SIPCORE Working Group
Internet Draft
Expires: January 13, 2009
Intended Status: Standards Track (PS)

James Polk
Cisco Systems
Brian Rosen
NeuStar

July 13, 2009

Location Conveyance for the Session Initiation Protocol
draft-ietf-sipcore-location-conveyance-01.txt

Status of this Memo

Copyright Notice

rights and restrictions with respect to this document.

Legal

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 SIP servers make routing decisions based on the location of the user agent client.

Table of Contents

1. Conventions and Terminology 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].

The following terms and acronyms used throughout this document are defined here:

LbyR = Location-by-Reference

LbyV = Location-by-Value

Location Generator (LG): The entity that initially determines or gathers the location of the Target and creates Location Objects describing the location of the Target [RFC3693].

Location Inserter: a role created in this document describing the entity that included location in a SIP request, either by-value or by-reference (i.e., including a location URI).

Location Object (LO): An object conveying location information (and possibly privacy rules) to which Geopriv security mechanisms and privacy rules are to be applied [RFC3693].

Location Recipient (LR): The entity that receives location information. It may have asked for this location explicitly (by sending a query to a location server), or it may receive this location asynchronously [RFC3693].

Location Server (LS): The entity to which a LG publishes location objects, the recipient of queries from location receivers, and the entity that applies rules designed by the rule maker [RFC3693].

A Location Server is also an entity that retains Target Location Objects that are dereferenced by Location Recipients via SIP SUB/NOT transactions.

Target: A person or other entity whose location is communicated by a Geopriv Location Object [RFC3693].

Using Protocol: A protocol that carries a Location Object [RFC3693].

2. Introduction

This document describes how geolocation can be "conveyed" in a SIP request from one SIP entity unsolicited to another entity using SIP [RFC3261]. Here, "Location" is a description of the physical geographical area where something currently exists. Note that this information is not solicited by the entity that receives it. The mechanism in this document does not allow one SIP user to request the location of another SIP user to be returned in a response.

Geographic location in the IETF is discussed in RFC 3693 (Geopriv Requirements) [RFC3693]. It defines a "Target" as the entity whose location is being transmitted over IP. A [RFC3693]-defined "Using Protocol" describes 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 a SIP extension to carry a Location Object and how it complies with the Using Protocol requirements in RFC 3693.

Common terms are in Section 1. Section 3 provides an overview of SIP location conveyance. Section 4 details the extensions to SIP necessary to accomplish location conveyance. Section 5 gives decode examples of geolocation within SIP requests, both LbyV and LbyR. Section 6 articulates the SIP element type behaviors for location conveyance. Section 7 discusses Geopriv privacy considerations. Section 8 discusses security considerations. Section 9 IANA registers the modifications made to SIP by this document in section 4.

3. Overview of SIP Location Conveyance

The concept of conveying location in SIP is fairly straightforward. Location is conveyed directly or indirectly from a transmitting SIP entity to a receiving SIP entity. When location is conveyed directly, it is conveyed as a value contained within the SIP request, as in Figure 1.

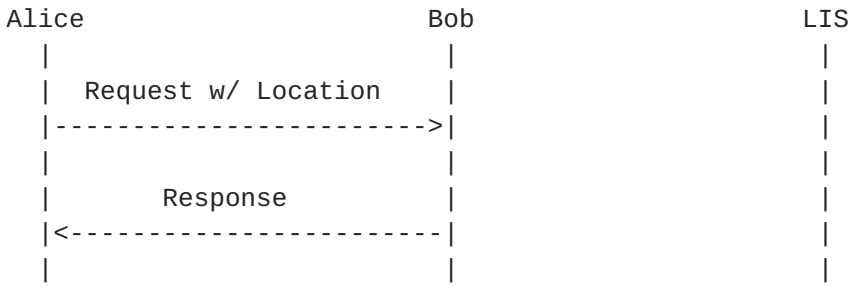

Figure 1. Location Conveyed by Value

When location is conveyed indirectly, analogous to Content

Indirection [RFC4483], Bob receives (from Alice) a location URI and must make an additional request - here called a dereference - to

learn Alice's actual location from a Location Server (LS) identified in the location URI, as in Figure 2.

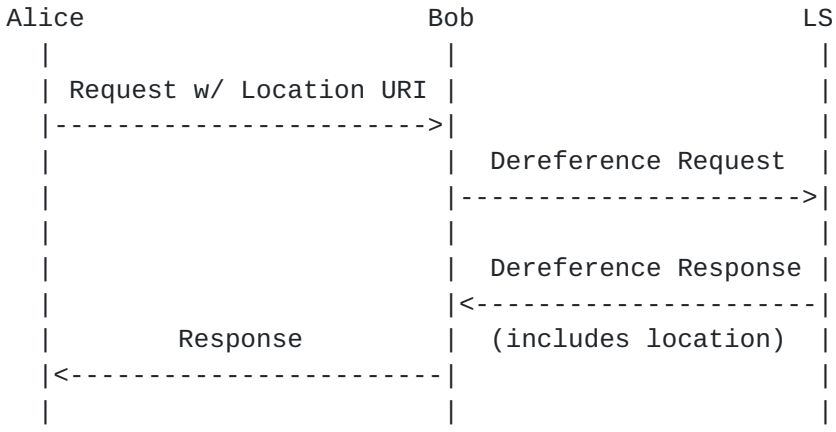

Figure 2. Location Conveyed by Reference

Many protocols can be used for this dereference transaction, but this is usually determined by the scheme of the location URI in the SIP request. In other words, if the location URI is a SIPS: URI, then SIPS would be used to contact the LS to make the dereference.

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 the geographic location of a Target. LbyV refers to a UA including a PIDF-LO as a message body part of a SIP request, sending that Location Object to another SIP element. LbyR refers to a UA including a location URI in a SIP request header field which can be dereferenced by a Location Recipient to retrieve Alice's Location Object, in the form of a PIDF-LO.

To accomplish location conveyance in SIP, a new SIP header field, Geolocation, is created and described in this document. The Geolocation header field contains a URI that points to where the location is for the location target, either in the body of the SIP request itself (LbyV), or on a Location Server (LbyR). A location URI that points to a message body is always a "cid:" URI (Content Identification), as defined in [RFC2392].

If the URI in the Geolocation header field is a scheme other than "cid:", a dereference transaction (see Figure 2) is necessary. This document describes how a SIP presence subscription [RFC3856] can be used as a dereference protocol.

Location can be inserted in a SIP request by a SIP server as well as by a UA. This document offers guidance on this practice. This document also describes how a location recipient can determine which

entity included a specific location, as more than one location can be conveyed in a given SIP request. Section 4 gets into guidance and limitations of this behavior.

A new error response (424 Bad Location Information) is also defined in this document. Within this response is a new header field indicating location-based errors, called the Geolocation-Error header field. This header field has various codes that provide additional information about the type of location error experienced by a Location Recipient, separated into actionable categories to be taken by the UAC.

Because more than one SIP entity can insert location, when considering SIP as an end-to-end protocol, there needs to be a means of identifying which location within a message of multiple locations was considered bad by a location recipient - if that were to occur. The ability to tell which entity (identified by host-id) inserted a specific location is extremely important. Not only does this allow each location error to be targeted at a particular inserter of a specific location object, but it also allows error recipients to understand when the location they inserted was not at fault, and that a received error is not meant for them. This optimization is necessary, otherwise each location error would be a blanket error to every entity upstream in the signaling path.

Just as location can be conveyed by more than one entity about the same target, there can be more than one location recipient along a request's path. It is possible to route SIP requests based on the location of the target (i.e., source based routing, instead of normal destination based routing). This means SIP servers can be location recipients. If this is not desired by a Location Inserter, then the Location Inserter can also include a separate indication in the Geolocation header field showing that this usage is not desired.

Location Inserters have the ability to provide instructional parameters about location it has inserted. These are hints to downstream entities on how the location information in the message was originated, intended and is to be used.

Transport Layer Security is expected when a request contains a target's location. Some implementations will choose to have S/MIME for integrity protection, or to encrypt message bodies from source to destination(s).

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

The new header field, the header parameters, the new option tag, the new error response, and Geolocation-Error codes are defined in Section 4, each of which are IANA registered by this document.

RFC 3693 demands that a transmitted location be required to maintain

privacy considerations. This document maintains all of the privacy considerations defined by RFC 3693, plus adds an intended usage indication within the SIP Geolocation header field. This increases

the considerations for recipients not to inspect a target's location when they are not the intended location recipient.

4. SIP Modifications for Geolocation Conveyance

The following sections detail the standards track modifications to SIP for Location Conveyance.

4.1 The Geolocation Header Field

This document defines Geolocation as a new SIP header field registered by IANA, with the following ABNF [RFC5234]:

```
Geolocation        =  "Geolocation" HCOLON (locationValue *(COMMA
                       locationValue)) (COMMA retrans-param)
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"
                       / generic-param ; (from RFC 3261)
geoloc-inserter    =  DQUOTE hostport DQUOTE
                       / gen-value ; (from RFC 3261)
retrans-param      =  "routing-allowed" EQUAL "yes" / "no"
```

sip-URI, sips-URI and absoluteURI are defined according to [RFC3261].

The pres-URI is defined in [RFC3859].

The cid-url is defined in [RFC2392] to locate message body parts. This URI type is present in a SIP request where location is conveyed as a value.

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

The Geolocation header field MAY have one or more locationValues. SIP servers inserting a locationValue MUST add the new value as the last locationValue in the Geolocation header field (i.e., the last locationValue in the header field is the most recent one added to the message). Placement of the "routing-allowed" parameter, when present, MUST be the last header field value in the Geolocation header field.

A locationValue has the following independent header field parameters,

o the "inserted-by=" parameter provides the hostport

(alice.example.com -- which is the same as the "sent-by"
parameter in a Via header field, with or without a port number)

of the SIP entity that inserted this locationValue into the request. If a Location Recipient has determined a supplied location is in error, as there can be more than one location in any request, the "inserted-by=" parameter is copied into the locationErrorValue in the response indicating the location error, and to whom the error is for. Hence, this "inserted-by=" parameter MUST be present in each locationValue. If an entity receives an Geolocation-Error with a hostport that does not identify this entity, the Geolocation-Error MUST 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. "used-for-routing" can occur more than once along the request's path. Because locationValues are inserted as last inserted is last in the header field, the last locationValue is the most recent one added to the message. This also gives the "used-for-routing" header field parameter added meaning - as the receiving SIP entity knows which location URI the message was routed upon.

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

There MUST NOT be more than one "inserted-by=" parameter or one "used-for-routing" parameter in the same locationValue. However, there can be more than one locationValue in the same Geolocation header field.

The "routing-allowed" header field parameter is a global parameter over any (and all/each) locationValues in the Geolocation header field. This is the reason why the placement of the header field parameter is outside any locationValue, appears only once, and is always last in the header field value.

This header field parameter only has the values "=yes" or "=no". When this parameter is "=yes", any locationValue can be used for routing decisions along the downstream signaling path by intermediaries. When this parameter is "=no", this means no locationValue (inserted by the originating UAC or any (or subsequent) intermediary(ies) along the signaling path) can be used by any SIP intermediary to make routing decisions. This behavior MUST be adhered to. Section 4.3 describes the details on what a routing intermediary does if it believes it needs to use the location in the SIP request in order to process the message further.

The practical implication is that when the "routing-allowed" parameter is set to "no", if an LbyV is present in the SIP request, intermediaries MUST NOT view the location (because it is not

for intermediaries to view), and if an LbyR is present, MUST NOT dereference it. UASs are allowed to view location in the SIP request even when the "routing-allowed" header field parameter is

set to "no".

The default behavior when this header field parameter is not present in a message is to treat the SIP request as if the parameter were present and its value is set to "no".

This document defines the Geolocation header field 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 is out of scope for this document since it is part of Presence, therefore, for completeness, Table 1 shows PUBLISH is to support Location Conveyance via this extension, but is not discussed further.

The following table extends the values in Tables 2 and 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 Field

The Geolocation header field MAY be included in any one of the above requests by a UAC. A proxy MAY add the Geolocation header field, but MUST NOT modify any pre-existing locationValue, including any associated header field parameters within an existing Geolocation header field value, unless one of the existing locationValues is used to retarget the request towards a new destination UAS. This is discussed in section 6.3.

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

A SIP proxy MAY add a Geolocation header field if one is not present, and MAY add the "routing-allowed" parameter if not yet

present in the SIP request. When a "routing-allowed" parameter is already present in the SIP request, a SIP server MUST NOT change the

value of the parameter (i.e., from 'yes' to 'no', or from 'no' to 'yes'). This would override the policy set by an upstream SIP entity (i.e., likely the UAC).

Adding a new locationValue to an in-transit request is NOT RECOMMENDED for at least two reasons,

#1 SIP Servers are not the best locators geographically of where a UAC is; the location information that a SIP server knows may not be the best location information available.

#2 this document gives limited guidance as to what a Location Recipient should do when receiving more than one location (no instructions are given about which locationValue to use when more than one is present), so adding a new locationValue may lead to undesirable results.

Location Recipients receiving a location object, whether received directly or as the result of a dereference, MUST honor the usage element rules within that XML document, as defined in [RFC4119]. Such entities MUST NOT alter the rule set.

4.2 424 (Bad Location Information) Response Code

This SIP extension creates a new location specific response code, defined as follows.

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 4.3 creates the Geolocation-Error header field to provide more detail about what was wrong with the location information in the request. This header field 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 (LbyV or LbyR), and the Location Recipient can process any of the locationValues, 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 an existing 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 any SIP intermediary as a result of this error message. The new 424 (Bad Location Information) error code is registered with IANA in Section 8 of this document. An initial set of IANA-registered Geolocation-Error codes are in Section 4.3 of this document.

4.3 The Geolocation-Error Header Field

As discussed in Section 4.2, 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 field is used for this purpose.

The Geolocation-Error header field is used to convey location-specific errors within a response. Additional IANA-registered values must be defined in an RFC (this is the "RFC Required" IANA policy defined in [RFC5226]). The Geolocation-Error header field has the following ABNF [RFC5234]:

```
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
                         / location-error-host-id
                         / location-error-code-text
                         / generic-param ; from RFC3261
location-error-node-id   = "node" EQUAL
                              DQOUTE hostport DQOUTE ; from RFC3261
location-error-host-id   = "inserter" EQUAL
                              DQOUTE hostport DQOUTE ; from RFC3261
location-error-code-text = "code" EQUAL quoted-string ; from RFC3261
```

The Geolocation-Error header field MUST contain at least one locationErrorValue to indicate what was wrong with the locationValue in the corresponding request the Location Recipient determined was bad. Each locationErrorValue contains a 3-digit error code indicating what was wrong with the location in the request. Each error type has a corresponding quoted error text string that is human understandable.

Each locationErrorValue contains the Location Recipient identifier (the "node=" parameter) which experienced the location error, as

well as an identifier of which SIP entity (the "inserter="
parameter) the Location Recipient is told (in the locationValue)
added this problematic locationValue to the request. The "node="

and "inserter=" are the domain identifier of a SIP entity, with the ability to have the same host communicate on different ports - and have port specific identification. This is the same information that would be entered in the "sent-by" parameter of the Via header field for that entity [RFC3261]. As stated in section 18 of RFC 3261, the usage of FQDN is RECOMMENDED. Here are examples of both locationErrorValue parameters,

node="bob.example.com"

inserter="alice.example.com"

Both the "node=" and "inserter=" parameters MUST be present in all locationErrorValues in a response, unless the locationValue of the request did not include the "inserted-by=" parameter (which is a violation of this document). The "inserter=" parameter value is copied from the "inserted-by=" parameter within the locationValue of the request.

This is required because a Location Recipient that experienced a problem with the location included in a request needs to tell the specific SIP entity which added the locationValue in error into the original request. Since more than one SIP entity can insert location into a request in transit, all other SIP elements may be confused by receiving this error header field, were it to remain generic to all entities in the response path. This requirement means that the header field identifies the Location Inserter who inserted the problematic locationValue, so that all other SIP entities that read the header field know to ignore it. 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 be interpreted by each SIP entity along the original path upstream and be processed by both the server that included the invalid locationValue and the UAC that did not, resulting in confusion at the UAC.

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

When more than one locationErrorValue is present in a Geolocation-Error header field, they are separated by commas.

If more than one locationErrorValue is present in a response, and intended for the same "inserter=", each error code MUST be unique to this "inserter=" entity, and the error codes MUST NOT conflict in meaning.

Here is an example of a Geolocation-Error header field:

```
Geolocation-Error: 200; code="Retry Location Later";
                        node="bob.example.com";
                        inserter="alice.example.com";
```

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 Field

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. For example, Alice includes her location in an INVITE to Bob. Bob can accept this INVITE, thus creating a dialog, even though his UA determined the location contained in the INVITE was bad. There is a Geolocation-Error header value in the 200 OK to the INVITE informing Alice the INVITE was accepted but the location provided was bad. The SIP requests included in table 2 above are the ones allowed to optionally contain the Geolocation header field (see section 4.1). That said, a UAC MUST ignore a Geolocation-Error header field value that did not contain its host-id..

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 field, is in Figure 3.

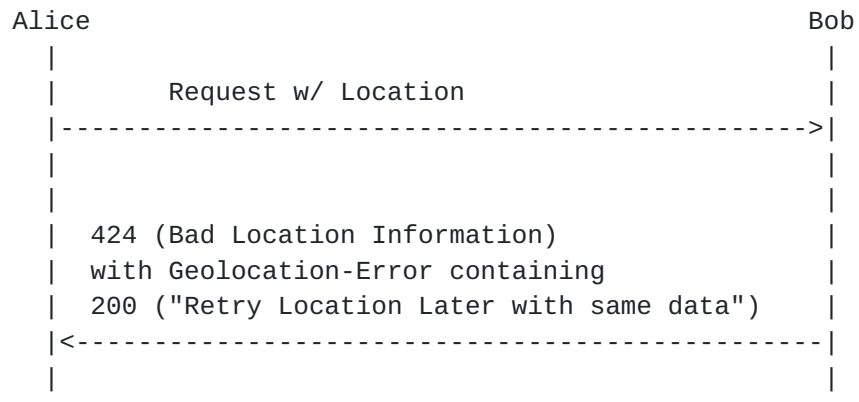

Figure 3. Basic Transaction with 424 and Geolocation-Error Header Field

The following subsections provide an initial list of location

based errors for any SIP non-100 response, including the new 424 (Bad Location Information) response. These error codes are divided

into 5 categories, based on how the response receiver should react to these errors.

- 1XX "Cannot Process Location"
- 2XX "Retry Location Later with same data"
- 3XX "Retry Location Later with updated device location"
- 4XX "Permission To Reveal Location Information to a Third Party"
- 5XX "Location Information Denial"

All 5 of the above error codes MUST be implemented to comply with this specification. Each of these actionable errors is given a 3 digit error code category, meaning any future 1XX, 2XX, 3XX, 4XX, and 5XX error codes defined will have the same action expected by X00 categories, although the future error code may provide more granular information. If another action is expected to occur with a newly defined error code, it MUST outside the 100-599 range.

4.3.1 Location Error: 100 "Cannot Process Location"

The location error 100 "Cannot Process Location" indicates to a Geolocation-Error recipient that the locationValue supplied in a request cannot be processed at this time. This only has to do with the location that the location "inserter=" added to the request, and not about the overall request that was sent.

Action(s) to be taken by Geolocation-Error receiver for a 1XX:
This error gives no guidance on what to do next. It is a general information indication to a SIP "inserter=" entity that there was an unspecific problem with the location supplied in the SIP request.

Implementations MAY choose to react as if the "inserter=" entity received a 2XX or 3XX location error. Implementations MUST NOT react as if the "inserter=" entity received a 4XX location error, as that error category involves human intervention to grant, or not, permission to reveal "inserter=" location when this is likely not desired.

The text string of "Cannot Process Location" is RECOMMENDED, but not mandatory for usage in this error. Implementations MAY use another text string.

An example 100 location error is:

```
Geolocation-Error: 100; code="Cannot Process Location";
```

```
  node="bob.example.com";
  inserter="alice.example.com";
```

This category covers all 1XX location errors. The same basic reaction is expected when a location "inserter=" entity receives any 1XX location error.

4.3.2 Location Error: 200 "Retry Location Later same data"

The location error 200 "Retry Location Later same data" indicates to a Geolocation-Error recipient that what they supplied in a request, as far as location is concerned, cannot be processed at this time, but there is no need to update the location at a later time in a new SIP request. For example, this location error is appropriate when the Location Recipient cannot process location at a specific time, or when there is there was a timeout when a location URI is dereferenced.

Action(s) to be taken by Geolocation-Error receiver for a 2XX:

Reactions to a 2XX location error are to try again after some period of time has elapsed. The "inserter=" has not identified problems with the location provided in the original request, so there is no need to update this location unless the requestor moves. A Retry-After header field MUST be present in the 424 response indicating after how long the LR thinks it can process the location, the error recipient MUST obey this instruction.

Implementations SHOULD choose to react by queuing another message with the same location information, unless the "inserter=" entity knows it has changed locations.

Implementations MAY inform the user of this error. The text string of "Retry Location Later same data" is RECOMMENDED, but not mandatory for this error. Implementations MAY use another text string to inform the user that location was not received by the UAS (i.e., the called party).

An example 200 location error is:

```
Geolocation-Error: 200; code="Retry Location Later same data";
                        node="bob.example.com";
                        inserter="alice.example.com";
```

This category covers all 2XX location errors. The same basic reaction is expected when a location "inserter=" entity receives any 2XX location error.

If a SIP request has the "routing-allowed" header field parameter set to "no", and the SIP server believes processing location is required in order to service the request properly, a 2XX location

error is sent towards the recipient. This error is the proper error even when there is no location in the SIP request, but the SIP

request contains a policy statement that location is not to be viewed during transit towards the ultimate destination.

4.3.2.1 Location Error: 201 "Linkable Target Identity Required"

The error code 201 "Linkable Target Identity Required" is specifically for the event in which a SIP request requires a binding of the Target's identity to the presence document in order to know this is the Target's location to make an appropriate routing decision. Because Alice could include Bob's location in her SIP request, the SIP server - in this specific case - needs to understand this message is routed based on Alice's location, and not Bob's. There is no absolute binding between presence documents and SIP signaling, hence a separate error with specific behaviors are necessary.

It is of particular importance is the emergency calling case, described here [ID-PHONE]. The presence document has a <presence> element containing an "entity=" attribute identifying who the presence document is about. The PIDF-LO is a presence document. [RFC3693] allows unlinkable pseudonyms to be in the "entity=" attribute. This is problematic for some (all?) source location based call routing situations.

The node= routing intermediary makes this locationErrorValue a 201 to resolve this problem. In the 424 response, the Retry-After header value of '0' is REQUIRED, indicating the new request be sent immediately, but with a target identification resolved in the PIDF-LO and SIP request. In presence, the entity= attribute is typically the URI of the presentity, meaning something like the Contact address of the UAC here.

If there is no Retry-After header value, the default is to resend the SIP request immediately with the corrected entity= attribute (i.e., emulating a Retry-After: 0 header value).

Action(s) to be taken by Geolocation-Error receiver for a 201:

> 201 location error indicate the "inserter=" did not properly identify the Target of this presence document. The Retry-After has been received - or is assumed to be 0 - meaning the new SIP request is to be sent immediately.

4.3.3 Location Error: 300 "Retry Location Later with device updated location"

The location error 300 "Retry Location Later with device updated location" indicates to a Geolocation-Error recipient that what they

supplied in a request, as far as location is concerned, cannot be processed. In order to retry this request in a new SIP request, the

location information must be updated.

Action(s) to be taken by Geolocation-Error receiver for a 3XX:

> 3XX location errors indicate the "inserter=" SIP entity needs to refresh its location, or make the location information supplied more complete, without notifying the user of this error. 3XX errors may be resolved without user intervention.

This document gives no guidance how this is accomplished, given the number of ways a UAC can learn its location, or a SIP intermediary can Sight a UAC, as defined in [RFC3693].

This 300 location error currently does not indicate what exactly was wrong with the location supplied, according to the Location Recipient. That is left for a future effort.

Implementations MAY choose whether or not to inform the user of this error. The text string of "Retry Location Later with device updated location" is RECOMMENDED, but not mandatory for usage in this error. Implementation MAY use another text string to inform the user that location was not received by the UAS (i.e., the called party).

A 3XX location error would be used where the Location Recipient cannot find or cannot parse the location supplied. 3XX location errors should cause a requestor to retry with refreshed location information, which might be sufficient to fix the problem. Also, a 3XX location error would be used when a Location Recipient was expecting to find location in a SIP request, but did not find it - perhaps an emergency request was made that did not contain location. The retry in this case would be in the form of an UPDATE Method request, containing location. If the 424 response contains a Retry-After value, there SHOULD NOT be a long delay associated with a new request - under the guise that if the location had been good, there would not have been an error to this request.

An example 300 location error is:

```
Geolocation-Error: 300; code="Retry Location Later with device
                          updated location";
                          node="bob.example.com";
                          inserter="alice.example.com";
```

This category covers all 3XX location errors. The same basic reaction is expected when a location "inserter=" entity receives any 3XX location error.

4.3.4 Location Error: 400 "Permission to Reveal Location Information to

The location error 400 "Permission to Reveal Location Information to a Third Party" indicates to a Geolocation-Error recipient that they sent a particular SIP Request including location in that request, without giving permission in the request for an intermediary SIP entity to look at that location information (i.e., the <retransmission-allowed> was set to "no" in the PIDF-LO for B2BUAs, or "routing-allowed=no" as a Geolocation header field parameter for proxy servers) to route the request toward the intended recipient (i.e., the UAS, or the called party).

Action(s) to be taken by Geolocation-Error receiver for a 4XX:

> 4XX location errors indicate to the UAC (i.e., the calling party) that they need to grant permission to a SIP intermediary server to look at the supplied location to complete the message routing. This indication MUST require human user intervention, acting as the ruleholder of the policy on whether or not location is to be revealed.

The user of the location "inserter=" device can choose to grant permission to this SIP intermediary server to allow this request to be routed, or the user can deny permission. It is the user's choice as ruleholder.

Implementations MUST provide the user, as ruleholder, a clear indication that permission to consume their location is sought by an entity that is not the entity that the user is calling. The text string of "Permission To Reveal Location Information to a Third Party" is RECOMMENDED, but not mandatory for usage in this error. Implementations MAY use another text string to inform the user that location is being sought by an intermediary (i.e., not the called party).

This document gives no guidance how the UAC can seek permission from the user, given the number of ways a UAC can accomplish this (i.e., audio prompt or toggle or keystroke on a UA).

This 400 location error indicates exactly which SIP server indicates that it needs the location by the "node=" FQDN address supplied, perhaps telling the user (via audio or video indication) which SIP entity wants this location. Perhaps the user can know in some circumstances whether this is an appropriate "node=" (domain). This latter feature is not described in this document.

An example 400 location error is:

```
Geolocation-Error: 400; code="Permission to Reveal Location
                          Information to a Third Party";
                          node="bob.example.com";
```

inserter="alice.example.com";

This category covers all 4XX location errors. The same resolution

is expected to be afforded to the UAC user, as ruleholder, to any 4XX location error.

4.3.5 Location Error: 500 "Location Information Denial"

The location error 500 "Location Information Denial" indicates to a Geolocation-Error recipient that what they supplied in a request, as far as location is concerned, has been denied at this time. This only has to do with the location that the location "inserter=" added to the request, and not about the overall request that was sent. If this were applied to the SIP request itself, this would equate to a 6XX Global error.

Action(s) to be taken by Geolocation-Error receiver for a 5XX:
This error gives no guidance on what to do next, other than to not try again with this same location supplied.

If the Location Recipient determined that merely refreshing, or in some other way alter or augment the location supplied would work in a new request, then a 3XX location error would have been more appropriate. An example of why this 5XX could have been returned is if location were sent as a location URI, and the LS denied the dereference request from the potential Location Recipient, this is the expected location error returned to the "inserter=" entity. In all likelihood, this is a dereference function error, meaning this will not occur when location is carried by-value in the request.

Implementations MUST NOT make any assumptions about the implications of this location error other than recognizing that a location based denial error has occurred. This does not mean the SIP request was denied, or even had an error, unless the response was a 424. Otherwise, this only has to do with the location part of the request.

The difference between a 1XX and a 5XX location error is simple. A 1XX location error is appropriate when a Location Recipient either does not know or cannot tell the "inserter=" entity what was wrong with the location supplied in a SIP request. A 5XX location error is appropriate when the Location Recipient was purposely and actively prevented from retrieving the location information. This could occur in a UAS or SIP server.

If implementations choose to inform the UAC user of this error, the text string of "Location Information Denial" is RECOMMENDED, but not mandatory for usage in this error. Implementations MAY use another text string.

An example of this location error is here:

```
Geolocation-Error: 500; code="Location Information Denial";
                        node="bob.example.com";
                        inserter="alice.example.com";
```

This category covers 5XX location errors. The same basic reaction is expected when a location "inserter=" entity receives any 5XX location error.

4.3.6 Which Scenario Matches Which Error Code?

The following scenario/error code mapping MUST be used for consistency,

- Scheme (sip:, or sips:, or pres:, etc.) of the location URI isn't supported - (100)
- Format (geo or civic) isn't supported - (100)
- Found where location should be, but do not understand what is there -(300)
- Cannot find LS in location URI (no access or no path) - (100) or denied access - (500))
- Dereference failed (timeout) - (200)
- Insufficient location info supplied - (300)

4.4 The 'geolocation' Option Tag

This document creates and IANA registers one new option tag: "geolocation". This option tag is to be used, as defined in [RFC3261], in the Require, Supported and Unsupported header fields.

4.5 Using sip/sips/pres as a Dereference Scheme

If an LbyR URI is included in a SIP request, it MUST be a SIP-, SIPS- or PRES-URI. When PRES: is used, if the resulting resolution, as defined in [RFC3856], resolves to a SIP: or SIPS: URI, this section applies.

This document IANA registers 3 mandatory-to-implement URI schemes for LbyR:

- SIP:
- SIPS:
- PRES:

These 3 are registered with IANA in Section 9.6.

These schemes MUST be implemented according to this document.

absoluteURI is not mandatory-to-implement.

Dereferencing a Target's location using SIP- or SIPS-URI is accomplished by treating the URI as a PRES-URI and generating a SUBSCRIBE request to a presence server as defined in [RFC3856] using the 'presence' event package. The resulting NOTIFY MUST contain a PIDF-LO. See Figure 4 for a basic message flow for a dereference. The NOTIFY contains Alice's PIDF-LO in Figure 4.

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

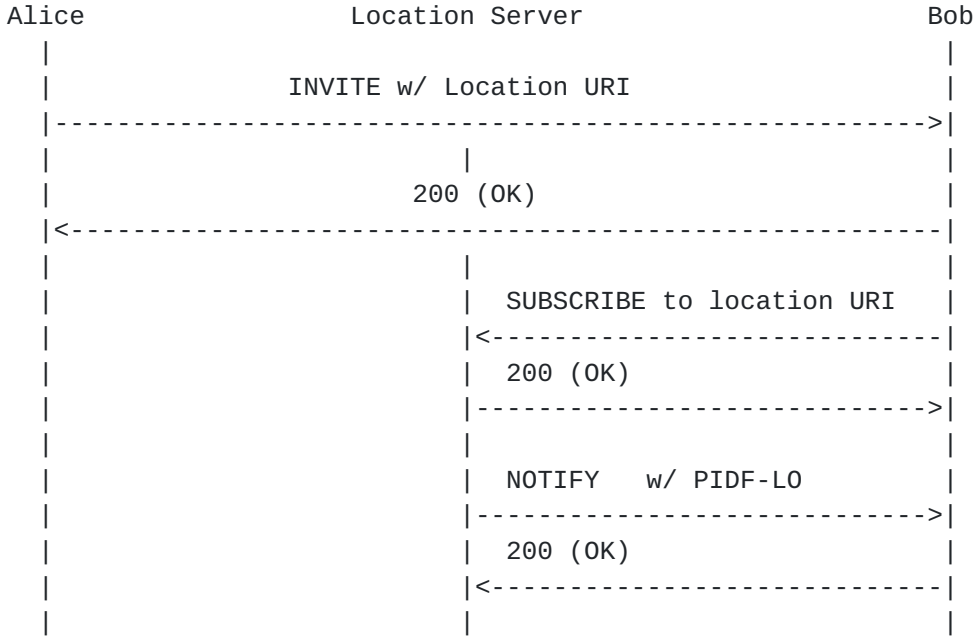

Figure 4. Location-by-Reference and Dereferencing

In Figure 4, Alice sends Bob her location in a location URI. Bob receives this location URI in the INVITE and generates a new transaction (SUBSCRIBE) to retrieve Alice's PIDF-LO. If accepted, the PIDF-LO will be returned in the NOTIFY request from the Location Server to Bob's UA. 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.

The SUBSCRIBE contains a geolocation option tag in either the Supported or Require header field (depending on what strength of support the UAC requires). The NOTIFY MUST match the subscribing UAC's option-tag strength for geolocation.

A dereference of an LbyR 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 this PIDF-LO. This is the purpose of this extension to SIP - to convey location to a specific

destination.

5. Geolocation Examples

This section contains are two examples of messages providing location. One shows LbyV with coordinates, the other shows dereferencable location URI. 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 appear within the <gp:location-info> in 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 field, which has a different type of URI, based on the different means of location conveyance. Section 5.1 shows a "cid:" URI, indicating this SIP request contains an LbyV message body - which is in the form of a PIDF-LO. Section 5.2 shows an LbyR URI indicating location is to be acquired via an indirection dereference mechanism, which is determined by the scheme of URI supplied.

5.1 Location-by-value (Coordinate Format)

This example shows an INVITE message with a coordinate location. In this example, the SIP request uses a sips-URI [RFC3261], meaning this message is protected using TLS on a hop-by-hop basis.

```
INVITE sips:bob@biloxi.example.com SIP/2.0
Via: SIPS/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:target123@atlanta.example.com>
  ;inserted-by="alice@atlanta.example.com"
  ;routing-allowed=no
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: <target123@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"
          xmlns:gml="http://www.opengis.net/gml"
          entity="pres:alice@atlanta.example.com">
        <tuple id="target123">
         <status>
         <timestamp>2009-07-13T09:00:00Z</timestamp>
          <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>2009-07-29T18:00:00Z</gp:retention-
                            expiry>
            </gp:usage-rules>
            <gp:method>802.11</gp:method>
a          </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. The "cid:" eases message body parsing.

If the Geolocation header field did not contain a "cid:" scheme, for example, like this location URI:

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

... the existence of a non-"cid:" scheme indicates this is a location URI, to be dereferenced to learn the target's location. Any node wanting to know where user "target123" is would subscribe to

server5 to dereference the sips-URI (see Figure 4 for this message flow, and Section 5.2 for this decoded example). 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.

5.2 Location-by-reference

Below is an INVITE request with a location URI that is not a "cid:" - instead of an LbyV PIDF-LO message body part as shown in Section 5.1. The Location Recipient will dereference Alice's location at the Atlanta Location Server the location URI is pointing towards. Dereferencing, if done using 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 request containing Alice's UA location, as a PIDF-LO message body (see Figure 4 for this message flow example). The NOTIFY, in this case, and not the INVITE, is the SIP request that is conveying location. 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"
  ;routing-allowed=no
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, it is RECOMMENDED that entities outside this domain be able to reach the dereference server, unless location is intentionally restricted within the atlanta.example.com domain.

6. SIP Element Behavior

Because a device's location is generally considered to be sensitive in nature, location information needs to be protected when transmitted. This can be addressed through securing the location information to prevent either viewing or changing the PIDF-LO.

Section 26 of [RFC3261] defines the SIPS security functionality by

transporting SIP messages with either TLS protection between SIP entities.

If a SIP entity wants to prevent all SIP entities in a request path that do not possess a decryption key from viewing or changing the contents of the PIDF-LO, the message body needs to be secure by a means such as S/MIME.

6.1 UAC Behavior

A UAC might choose to send location in a SIP request to facilitate location-based routing of the request, or for some other purpose. Alice communicating her location to Bob in a SIP request is a simple example of this. If Alice wanted to include her location as a message body in an INVITE that also has an SDP message body, the Content-Type: Multipart MUST be supported by both UAC and UAS. Multipart comes in many forms (/mixed, /alternative, etc), and this document does not limit which type of multipart is used, though future documents might specify or limit multipart to a subset of all the choices for a given use.

A UAC conveying location MUST include a locationValue in a Geolocation header (see section 4.1) with either an LbyV indication (a cid-URL), or a dereferencable location URI. Requests containing an LbyV message body sent MUST also contain a Geolocation header field. The UAC supporting this extension MUST include a Supported header with the 'geolocation' option tag.

More than one location format (civic and coordinate) can 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, identifying the same target. The same location in multiple formats, for example, a partial or complete geodetic and a partial or complete civic, allows the recipient to select the most preferred format for its use. Additional complementary location information can be in the second format as well.

Multiple PIDF-LOs are allowed in the same request, with each allowed to point at separate positions - however, each PIDF-LO MUST identify a different Target (i.e., in the entity= attribute in the <presence> element of the presence document). 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). A SIP request SHOULD include only one location per target in a single SIP request. More than one will likely lead to confusion by a Location Recipient because this extension does not provide guidance on what a recipient is to do with more than one location for the same target (which could point to different positions), nor does it give any preference regarding which location is more or less reliable than another location in the same request.

The presence of the 'geolocation' option tag in a Supported header field without a Geolocation header field in the same message informs

a SIP element receiving this request that 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. Indicating support with a geolocation option tag affects how a UAS or SIP server processes such a request. For example, it ought to understand that erroring the request because there was no location in the request is likely not going to result in the location appearing in the subsequent request.

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

A UAC inserting a locationValue MUST include an "inserted-by=" parameter to indicate its hostport. This is copied to the "inserter=" parameter of the Geolocation-Error header field in a response if a Location Recipient determines there is something wrong with the locationValue in this 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, and who the locationErrorValue is for. For example:

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

If a UAC does not learn and store its location locally (a GPS chip) or from the network (DHCP or LLDP-MED), the UAC MAY learn its location URI (from DHCP for example). If the latter is the case, the UAC can SUBSCRIBE to this location URI, using the 'presence' event package, to get and store its own location.

The Location Server will likely challenge requests to dereference a Target's location URI. The location URI SHOULD be treated as equivalent to possession of the location information itself and thus TLS SHOULD be used when transmitting any location URI hop-by-hop along the path to the Location Recipient, for protection reasons. This is not to be confused with a 'possession model', in which possessing the location URI grants authorization to dereference the URI. Any entity dereferencing the location URI MUST pass whatever authentication and authorization rules are on the LS to acquire this location. The Ruleholder from [RFC3693] is still very much in control - for any entity possessing the LbyR.

If the Location Generator wishes to control whether any location

included in the SIP request or added along the signaling path of this request can be viewed for routing decisions, the Location Generator adds a Geolocation header field value including the

"routing-allowed=no" parameter. This header field parameter provides specific policy rules for each locationValue (if more than one locationValue is inserted along the signaling path) within the SIP request. A UAC SHOULD include the "routing-allowed" header field parameter, with or without a locationValue, to each SIP request supporting this specification to ensure the UAC's policy for intermediaries which might add a locationValue of the Target downstream. The default behavior for SIP servers is to consider this value to be present, with a value of "no".

There is no feedback mechanism to inform the Target if a SIP server has included a locationValue downstream. If a UAC has already conveyed location in the original request of a transaction, and wants to update its location information (for whatever reason) after the original request is sent, or after a dialog is created, this is done by sending an UPDATE request [RFC3311]. The UPDATE will include a geolocation option tag and Geolocation header field with the new locationValue to the original destination UAS.

A presence document includes identity information (in the "entity=" attribute of the <presence> element), although the identity could be an unlinkable pseudonym [RFC3693]. Implementations of this extension SHOULD consider the appropriateness of including an unlinkable pseudonym as the identity in the location information where a real identity is not required. See the concerns raised in section 4.3.2 about unlinkable pseudonyms in relation to their potential problems with requests that need to route based on the location contained in the message.

When using LbyR, the location URI MUST NOT contain any user identifying information. For example, use something unidentifiable like

3fg5T5yqWowhGLn54wg4NgHlkDsFn@example.atlanta.com

rather than

aliceishere@example.atlanta.com).

Use of self-signed certificates is inappropriate for use in protecting a PIDF, as the sender does not have a secure identity of the recipient.

Although this is discussed in more detail in later in section 6.2, SIP entities MUST NOT bypass rules and behaviors conveyed in a PIDF-LO. UACs SHOULD take care when setting their <retransmission-allowed> flag to "yes". When Alice tells Bob her location with this flag set to "yes", Bob is free to tell Carol where Alice is (as long as the <retention-expiry> time has not

elapsed, indicating that the location information should be deleted). This is an implicit byproduct of the PIDF-LO rule-set, as of this writing. This decision is a configuration issue, but is

worth mentioning here.

6.1.1 UAC Receiving a Location Failure Indication

Location Recipients that use the location information for location-based routing decisions can be either destination UASs or intermediate servers. If a request fails based on the location information in the request, a 424 (Bad Location Information) response is sent back to the UAC. The 424 MUST have a Geolocation-Error header field containing one or more locationErrorValues in the response message. A locationErrorValue has a header field parameter indicating which entity inserted the locationValue correlated to this error, called the "inserter=" parameter. This "inserter=" parameter (in the locationErrorValue) is copied from the "inserted-by=" parameter (from the locationValue) by the Location Recipient (UAS or proxy) sending the error response. A UAC receiving a Geolocation-Error in any response type MUST check whether the "inserter=" parameter in the locationErrorValue indicates this UAC.

- If locationErrorValue does not match, the locationErrorValue MUST be ignored.

- If a locationErrorValue is in a 424, and the "inserter=" entity is not this UAC, the response SHOULD be treated as a 400 response.

- If locationErrorValue does indicate this UAC, this UAC MUST process the response, including the Geolocation-Error code (defined in section 4.3), taking the action described in that section for the received error code.

Additionally, the UAC MUST ignore a Geolocation-Error header field value, even for this UAC, even in a 424 response if the UAC did not include a Geolocation header field value (with locationValue) in the request part of the transaction.

A UAC MAY reattempt a new request if it can correct the stated failure in the Geolocation-Error header field, unless the location error is a 5XX level error - Section 4.3 clearly states not to do this. A UAC MUST follow all the guidance that pertains to UACs from Section 4.3 (Geolocation-Error Header Field), heeding what to do when it receives any of the error codes articulated in that section.

Any UAC that inserted location into a request MUST be prepared to receive the Geolocation-Error header field in any response, looking to determine if a locationErrorValue 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 reason other than location, any response MAY contain a Geolocation-Error header field containing a locationErrorValue with the details of the location error.

6.2 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 is in a message body (part) or requires an additional dereference transaction. If the location URI is sip:, sips: or pres:, and the UAS wants to learn the UAC's location, the UAS MUST SUBSCRIBE to the provided URI to retrieve the PIDF-LO being conveyed by the UAC as defined in [RFC3856]. If successful, the Target's PIDF-LO will be returned in the NOTIFY request from the remote host. The UAS is not REQUIRED to dereference the location URI if location is not needed to process the request. It is RECOMMENDED the UAS display the location to the user, or otherwise render the location appropriately.

A Require header field with the 'geolocation' option tag indicates the UAC requires that the UAS understand this extension, sending an error response if it does not. A 420 (Bad Extension) with a 'geolocation' option tag in an Unsupported header field would be the appropriate response in this case.

It is possible, but undesirable, for a message to arrive with a body containing an LbyV, 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 publication(s), a location URI only has meaning within the Geolocation header field and MUST NOT appear in any other SIP header field.

There are 3 Geolocation header field parameters,

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

The "inserted-by=" parameter informs a Location Recipient which SIP entity 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 6.2.1.

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 locationValues in the Geolocation header field, it is possible that different locationValues were used to route the message at different points along the path traversed by the request. This is allowed, as it is consistent with the rule that whenever a message is routed based upon a locationValue, a "used-for-routing" parameter is added to the applicable locationValue. This parameter MUST be present in each locationValue used along the path. A "used-for-routing" parameter MUST NOT be removed from a locationValue in a request.

The "routing-allowed" parameter is exclusively for SIP servers, and will be discussed in section 6.3.

Additional locationValues inserted into a request MUST be placed the order they were generated, 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.

Individual header field 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 server(s) cannot reply to a response with an error response. If the UAS wants to send its location to a UAC, it can do so in a new request within a separate transaction. This document gives no recommendation about which SIP request to use, but SIP MESSAGE is a viable choice.

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

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

Dereferencing for sip:, sips: and pres: URI schemes are mandatory-to-implement.

A UAS MUST be prepared to receive subsequent location updates from the UAC, either LbyV or LbyR (regardless of how the UAS received

location previously from this UAC). The UAC will convey updated location using the UPDATE [RFC3311] method to the UAS, and not a reINVITE. The UAS MUST NOT reject updated location arriving in a

reINVITE though, as other dialog parameters might be changing also (in which a reINVITE is appropriate).

If there is more than one location (any combination of LbyV and LbyR), this document does not give guidance about what a Location Recipient does with each location. There are no priority or more-trusted indications specified by this document. All this is considered application-specific, and out-of-scope for this document. If more than one location is present in the PIDF-LO, location elements in the same PIDF-LO MUST apply to the same Target (identified in the "entity=" attribute) and point at the same position on the earth. If there is more than one PIDF-LO with different Target identifiers, then the UAC is merely telling the UAS where more than one Target is, and there should not be any conflict.

Within any PIDF-LO, there is a <retransmission-allowed> policy element that can be set to "yes" or "no". These are the only possibilities. If Alice conveys her location to Bob (as has been described throughout this document) with a <retransmission-allowed> element set to "no", then Bob MUST NOT inform any other element where Alice is. If the <retransmission-allowed> element is set to "yes", then Bob can inform other elements where Alice is, but only as long as the <retention-expiry> policy element, also in the PIDF-LO, allows [RFC4119]. As a byproduct of this, that means that if Alice conveys her location to Bob with a <retransmission-allowed> element set to "yes", and the <retention-expiry> time does not require Bob to delete Alice's location yet, then Bob is free to tell anyone else where Alice is. The entity= attribute in the <presence> element identifies who is the target of each location, preventing confusion. Whenever Bob conveys Alice's location, <retention-expiry> timer MUST be maintained as is (i.e., not changed from when Bob received it). RFC 4119 implicitly allows this behavior, and the behavior does not change when SIP is the Using Protocol.

6.2.1 UAS Generating a Location Failure Indication

If a UAS receives location in a request, but determines there is a problem with the location in the request, it is the responsibility of the UAS to inform the entity that inserted the problematic location into that request. The Geolocation header field in the request has a locationValue, providing the UAS a location URI indicating where the Target's location is. The Location Target identified in the PIDF-LO may or may not be the location inserter, or the generator of the request. Ultimately, location is in a PIDF-LO. This is either in the request as a message body (LbyV), or obtained by initiating a dereference transaction on a Location Server identified in the location URI. Location Servers typically

challenge all dereference requests. All PIDF-LOs have a Location Target identifier. The "inserted-by=" parameter of the locationValue tells the UAS which SIP entity inserted that

locationValue. This "inserted-by=" parameter is copied into the "inserter=" parameter of the locationErrorValue generated by the Location Recipient (the UAS), in a response, when it wants to inform the location "inserter=" entity there was a problem with the location it received.

There can be more than one locationValue in a request. The "inserter=" parameter in the locationErrorValue will prevent entities that did not insert the errored location from misunderstanding whether the locationErrorValue applies to them.

If there is one valid locationValue in a request, even if all the others have errors with them, the Location Recipient MUST NOT send a 424 (Bad Location Information) response. The Location Recipient (the UAS) MUST send a locationErrorValue for each errored locationValue, with unique "inserter=" parameters to make sure the right entities know which locations were in error.

As hinted at, a location "inserter=" can be a UAC or it can be an in-signaling-path SIP server acting as a UAC locator. This means the SIP server is including its version of where it thinks the UAC is, geographically. This "inserter=" has to be in the form of an dereferencable location URI in a locationValue, because SIP servers are not allowed to insert message bodies.

Each locationErrorValue has an error code, letting the location "inserter=" entity know what was wrong with the location supplied by that entity. See Section 4.3 for the 5 actionable responses a UAC can take from a locationErrorValue.

If the location "inserted-by=" entity, meaning either the UAC or proxy in the message path chose to indicate that location was sufficiently important to include a 'geolocation' option tag in a Require header field, any error response SHOULD be a 424 (Bad Location Information) back to the "inserter=" entity (knowing the response will ultimately go to the UAC regardless) if there needs to be a good locationValue sent to properly process the request. Only entities identified in a locationErrorValue as the "inserter=" entity will pay attention to this locationErrorValue. All other entities MUST ignore any locationErrorValue not directed towards them. See section 4.3 for more information on this, including all the location-specific errors and Geolocation-Error header field parameters.

In the above scenario ('geolocation' option tag in a Require header field), the only other response can be a 420, if the UAS does not support this Geolocation extension to SIP.

If the "inserted-by=" location entity placed the 'geolocation'

option tag in a Supported header field, the response can be a 424 if it chooses, but also can be any other SIP response, including a 200 OK. A locationErrorValue in a Geolocation-Error header field that

is not in a 424 (Bad Location Information) response is considered informational by the Location Recipient, and does not cause the Location Recipient to reject the request solely because of bad location information.

For example, Alice INVITEs Bob to a dialog, and includes geolocation in the request. Bob can accept the INVITE with a 200 OK and still add a locationErrorValue in the 200 OK indicating "yes, I accept your request, and btw, something was wrong with the location you provided in the INVITE". The specific problem with the location is indicated by the Geolocation-Error code. The "inserter=" parameter identifies the Location Inserter this locationErrorValue is intended for.

Each locationErrorValue is destined for one "inserter=" entity. This gives a Location Recipient a mechanism to tell each inserter what the Location Recipient concluded was wrong with the location the "inserter=" entity included. 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 is intended for the inserter (and which to ignore).

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

- there MUST NOT be more than one locationErrorValue with the same "inserter=" entity 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 field

```
Geolocation-Error: 201; code="Linkable Target Identity Required";
                       node="server42.example.com";
                       inserter="alice.example.com";
```

The above example says that the Location Recipient is server42.example.com, and this entity cannot verify the Target's identity within this message. This is typically needed in order to make routing decisions for the SIP request where the entity= attribute has an unlinkable pseudonym obscuring a location target's identity from the signaling. This is not desire because if Alice's message is to be routed based on the location in the SIP request, server42 has to verify that this is Alice's location. The appropriate action is to send a 424 (Bad Location Information) response with the above (201) Geolocation-Error header value. We do

not want Alice's request routed based on Bob's location.

See Section 4.3 for further rules about the Geolocation-Error header field and the locationErrorValue.

This document says nothing about what a Location Recipient does with more than one 'good' locationValue in a request (i.e., which to choose to use). This scenario MAY be addressed in a future effort.

Further, more than one error code is allowed in the locationErrorValue - each having one "inserter=" parameter.

6.3 Proxy Behavior

[RFC3261] states message bodies cannot be added by proxies. However, proxies are permitted to add a header field to a request. This means that a proxy can add a Geolocation locationValue header field with a dereferencable location URI, but not a LbyV message body.

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 new locationValue in a Geolocation header field. Also described earlier, each locationValue MUST contain an "inserted-by=" value indicating to a Location Recipient the host that inserted a specific location into a particular request.

If, however, location is already in a SIP request, a SIP server SHOULD NOT add another location URI that identifies the same target in the PIDF-LO (in the entity= attribute) to the same request. This will likely cause confusion at the Location Recipient as to which to use.

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 field, which would be a local policy decision, the new locationValue MUST be added to the end of the header field (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 field.

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

6.3.1 Proxy Behavior with Geolocation Header Field Parameters

SIP servers MUST NOT delete any existing Geolocation locationValue (URI or header field parameter) from a request. An existing locationValue MAY be modified by adding a "used-for-routing" parameter to an existing locationValue, if the request was retargeted based on the location within that locationValue.

According to this specification, the default value of any Geolocation header value "routing-allowed" is "no". This parameter does not have to be present to deny SIP servers along the signaling path the ability to view the target's location. This parameter MAY be added in transit by a SIP server, and MUST be with a value of "no". Other modifications of the 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 the proxy can make no other modification.

A SIP server MAY add a new Geolocation locationValue to a SIP request. The server SHOULD NOT insert a locationValue of a Target unless it is reasonably certain it knows the actual geographic location of the Location Target (for example, if it thoroughly understands the topology of the underlying access network and it can identify the device location reliably, even in the presence of NAT or VPN). Routing errors are likely if the SIP server inserts an incorrect locationValue.

A locationValue added to an existing Geolocation header field would look like:

```
Geolocation: <cid:alice123@atlanta.example.com>;
              inserted-by="alice123@atlanta.example.com",
             <sips:3sdefrhy2jj7@ls7.atlanta.example.com>;
              inserted-by="ls7.atlanta.example.com"
```

Notice the locationValue added by proxy "ls7" is last among locationValues. Proxies MUST add locationValue at the end of all

locationValues that are already present in the request.

If this request was then retargeted by an intermediary using the locationValue inserted by server "ls7", 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@ls7.atlanta.example.com>;
              inserted-by="ls7.atlanta.example.com"; used-for-routing
```

It is conceivable that an initial routing decision is made on one locationValue, and subsequently another routing decision is made on a different locationValue further towards the ultimate destination. 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 field parameter. In this instance, the SIP server retargeting based on another locationValue MUST add the "used-for-routing" header field parameter to the locationValue used for retargeting by this server. This will result in a Geolocation header field indicating that the request has been retargeted more than once, which is allowed.

A Proxy that inserts or adds locationValue into a request MAY move a 'geolocation' option tag that is in a Supported header field into a Require header field if this proxy deems geolocation to be sufficiently important to Location Recipient(s) of this request.

A proxy can read any locationValue present, and the associated body, if not S/MIME protected, and can 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. Retargeting is defined in [RFC3261]. However, if the Geolocation header field parameter "routing-allowed" is present and set to "no", or is not present (the default behavior is "no" if "routing-allowed" is not present, whether or not a Geolocation header field is present), SIP servers MUST NOT view the contents of the location message body. Further, SIP servers MUST NOT attempt to dereference a location URI. This is because the Location Inserter (likely the originating UAC) did not give the SIP server permission to view the location within the SIP request. How a SIP server indicates it requires permission to view a request's location in order to properly process this request is described in section 6.3.2.

If the Geolocation header field parameter "routing-allowed" is present in a SIP request, the value MUST NOT be changed during processing of the request. If the Geolocation header field parameter "routing-allowed" is not present, SIP servers MUST treat the location within the request as if the header field parameter "routing-allowed" were present and set to "no".

B2BUAs and SBCs should also adhere to the above proxy guidance with respect to the "Routing-allowed" header field parameter. In other

words, if the particular type of SIP server mentioned here supports this SIP extension and is not the ultimate destination of this SIP request, each policy rule within the Geolocation header field MUST remain intact and unchanged.

No SIP server can delete a "Routing-allowed" header field parameter, but if one is not yet present, any SIP server MAY add a "Routing-allowed" header field parameter with the value set to "no" only.

6.3.2 Proxy Error Behavior for Sending or Receiving locationErrorValues

For proxies that receive a SIP request that contains a location error, all the rules applicable to a UAS apply (see Section 6.2.1.). The one point to add is that a proxy does not need to examine location contained in a request. Section 6.2.1. only applies to proxies that need to monitor or police location within requests (for whatever reason).

If a proxy inserted a locationValue into a request, it MUST 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 6.1.1.) that included a locationValue a request, which then receives an error to that locationValue.

This location-inserting proxy SHOULD be (at least) transaction stateful for the response. If the proxy is configured as a stateless proxy, and it inserts location, it MUST process and monitor all SIP responses, looking for locationErrorValues that indicate it was the "inserter=" to learn that the location it supplied was in error. It SHOULD react according to the error code received. This document gives no guidance what the proxy should do to rectify the bad location information, since the proxy is not the SIP response destination, but a future document could address this.

The "routing-allowed" parameter's purpose is to indicate if SIP servers along the signaling path should bother looking at the location message body or dereferencing the location URI. There are two values specified here for this parameter: "yes" and "no". If the "routing-allowed" parameter is set to "yes", and the SIP server determines this SIP request should be routed based on the target's location, this parameter gives the server permission to look at the location (or dereference it). If this parameter is set to "no", then the SIP server MUST NOT view the location message body or dereference the location URI within this SIP request. If the SIP server believes it cannot process this message properly because it needs to learn the target's location in order to route the message,

then it MUST return a 424 (Bad Location Information) response, indicating it requires permission (error code 400) to view the location.

Here is an example of a Geolocation-Error header field

```
Geolocation-Error: 400; code="Permission to Reveal Location
                        Information to a Third Party";
                        node="server42.example.com";
                        inserter="alice.example.com";
```

The above example says that the Location Recipient is server42.example.com, and this entity believes it cannot route this message without knowing permission to view the Target's location. Regardless of whether there is a Geolocation header value parameter, such as

routing-allowed=no

or this parameter is not present in the SIP request (as shown 400 error example above). The default behavior is to act as if the parameter is present and set to "no". Server42 MUST get permission to view the Target's location by reading a routing-allowed header parameter saying "yes", otherwise a 400 error is sent back to the inserter= entity to get permission.

Section 4.3 highlights this example, stating the user, Alice, MUST be made aware of this location revelation request. This document does not give any guidance how Alice is to be informed (i.e., audio, visual, etc). Alice can grant permission or choose not to, knowing this SIP request attempt (to this destination, at this time) will fail. The problem might not recur if a future SIP request were to travel through a different server than server42 (or it might again).

7. Geopriv Privacy Considerations

Location information is considered by most to be highly sensitive information, requiring protection from eavesdropping, and altering in transit. [RFC3693] articulates rules to be followed by any protocol wishing to be considered a "Using Protocol", specifying how a transport protocol meets 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 be large. Normal RFC 3261 procedures of reverting to TCP when the MTU size is exceeded would be invoked.

8. 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 be accomplished through secure mechanisms, like S/MIME protecting message bodies (although 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 LbyV 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. This extension is especially sensitive. That said, there is the ability, according to [RFC3693] to have an anonymous identity for the target's location. This is accomplished by use of an unlinkable pseudonym in the entity= attribute of the <presence> element [RFC4479]. Though, this can be problematic for routing messages

based on location (covered several times in the document above).

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

This SIP extension offers the default ability to require permission to view location while the SIP request is in transit. The default for this is set to "no", and there is an error explicitly describing how an intermediary asks for permission to view the Target's location.

Because target locations can be placed on a remote server, called a Location Server accessible with the possession of a location URI, this URI has its own security considerations. It is tempting to assume that the dereference of this URI 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 a target's location. Consider, specifically, a nomadic user connected to an access network in a hotel. The UAC has no way to provide a credential acceptable of 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.

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 specification does not constrain the dereference protocol to use TLS. That specification is left entirely to the dereferencing protocol documents.

There is no end-to-end integrity on any locationValue or locationErrorValue header field parameter (or middle-to-end if the value was inserted by a intermediary), so recipients of either header field need to implicitly trust the header field contents, and take whatever precautions each entity deems appropriate given this situation. Any idea of using SIP Identity is lost as soon as it is realized that the Geolocation header value can be added to by intermediaries in the signaling path.

9. 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).

[Editor's Note: RFC-Editor - within the IANA section, please replace "this doc" with the assigned RFC number, if this document reaches publication.]

9.1 IANA Registration for the SIP Geolocation Header Field

The SIP Geolocation Header Field is created by this document, with its definition and rules in Section 4.1 of this document, and should be added to the IANA sip-parameters registry, in the portion titled "Header Field Parameters and Parameter Values".

Header Field	Parameter Name	Predefined Values	Reference
Geolocation	inserted-by=	no	[this doc]
Geolocation	used-for-routing	yes	[this doc]
Geolocation	routing-allowed	yes	[this doc]

9.2 IANA Registration for New SIP Option Tag

The SIP option tag "geolocation" is created by this document, with the definition and rule in Section 4.4 of this document, to be added to the IANA sip-parameters registry.

9.3 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 4.2 of this document.

9.4 IANA Registration of New Geolocation-Error Header Field

The SIP Geolocation-error header field is created by this document, with its definition and rules in Section 4.3 of this document, to be added to the IANA sip-parameters registry, in the portion titled "Header Field Parameters and Parameter Values".

Header Field	Parameter Name	Predefined Values	Reference
Geolocation-Error	inserter=	no	[this doc]
Geolocation-Error	node=	no	[this doc]

Geolocation-Error	code=	yes*	[this doc]

* see section 9.5 for the newly created values.

9.5 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 in the IANA sip-parameters registry. Details of these error codes are in Section 4.3 of this document.

Geolocation-Error codes

Geolocation-Error codes provide reason for the error discovered by Location Recipients, categorized by action to be taken by error recipient to be placed into SIP responses to inform the location inserter of the error.

Code	Description	Reference
100	"Cannot Process Location" General location error, meaning location in the request cannot be processed at this time. No actionable guidance. Can be treated as a 200 or 300 error by error recipient.	[this doc]
200	"Retry Location Later with same data" The location cannot be processed at this time. Error recipient should try again with same data.	[this doc]
201	"Linkable Target Identity Required" Target's identity cannot be unlinkable within the presence element's entity= attribute. This is necessary for routing SIP requests based on Target's location (and some other entity's).	[this doc]
300	"Retry Location Later with device updated location" Location cannot be processed at this time. Error recipient should try again with same data.	[this doc]
400	"Permission To Reveal Location Information to a Third Party" Permission from calling user to reveal location in request before request can be processed. This is a routing by location error. User MUST be informed of permission request.	[this doc]
500	"Location Information Denial" Request was actively denied because of the location in the request. Recipient should not try again.	[this doc]

9.6 IANA Registration of Location URI Schemes

This document directs IANA to create a new set of parameters in a separate location from SIP and Geopriv, called the "Location Reference URI" registry, containing the URI scheme, the Content-Type, and the reference, as follows:

URI Scheme	Content-Type	Reference
SIP:		[this doc]
SIPS:		[this doc]
PRES:		[this doc]

Additions to this registry must be defined in a permanent and readily available specification (this is the "Specification Required" IANA policy defined in [RFC5226])..

10. 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, Ted Hardie, Shida Shubert, Umesh Sharma, Richard Barnes, Dan Wing, Matt Lepinski and Jacqueline Lee for constructive feedback and nits checking.

Special thanks to Paul Kyzivat for his help with the ABNF in this document and to Robert Sparks for many helpful comments and the proper construction of the Geolocation-Error header field.

And finally, to Spencer Dawkins for giving this doc a good scrubbing to make it more readable.

11. References

11.1 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 2392, 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.

[RFC2976] S. Donovan, "The SIP INFO Method", RFC 2976, Oct 2000

[RFC3515] R. Sparks, "The Session Initiation Protocol (SIP) Refer Method", RFC 3515, April 2003

[RFC3903] Niemi, A, "Session Initiation Protocol (SIP) Extension for Event State Publication", RFC 3903, October 2004.

[RFC5234] Crocker, D., Ed., and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.

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

[RFC5226] T. Narten, H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", RFC 5226, May 2008

[RFC4479] J. Rosenberg, "A Data Model for Presence", RFC 4479, July 2006

11.2 References - Informative

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

[RFC4483] E. Berger, "A Mechanism for Content Indirection in SIP", RFC 4483, May 2006

[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 ", RFC 4776, October 2006

[ID-PHONE] B. Rosen, J. Polk, "ECRIT Phone BCP", draft-ietf-ecrit-phonebcp, "work in progress", July 2009

Author Information

James Polk
Cisco Systems
3913 Treemont Circle
Colleyville, Texas 76034

33.00111N
96.68142W

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

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

40.70497N
80.01252W

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. If a

requirement is not obvious in intent, a motivational statement is included below it.

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: to guarantee 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: if a UAC has moved prior to the establishment of a dialog between UAs, the UAC must be able to send location information. If location has been conveyed, and the UA moves, the UAC must be able to update the location previously conveyed to other parties.

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 [RFC4119] is a mandatory to implement format for location conveyance within SIP, whether included LbyV or LbyR.

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 happen because the UAC does not implement this extension, or because the UAC implements the extension, but does not know where the Target is. This may be, for example, due to the failure of the access network to provide a location acquisition mechanism the UAC supports. 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 also 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 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.