GEOPRIV
Internet-Draft
Expires: April 19, 2004

H. Schulzrinne
Columbia U.
J. Morris
CDT
H. Tschofenig
J. Cuellar
Siemens
J. Polk
Cisco

October 20, 2003

Policy Rules for Disclosure and Modification of Geographic Information
draft-ietf-geopriv-policy-00

Status of this Memo

Copyright Notice

Abstract

This document describes an XML schema for governing the disclosure and transformation of geographic location information. It also describes the goals and the non-goals of the design considerations for the policy rules and the details of the attributes used within the policy rules.

Table of Contents

1. Introduction

Location information needs to be protected against unauthorized access to preserve privacy of the owner of the location information. The GEOPRIV working group has defined a protocol-independent model for access to geographic information. The model includes a location generator (LG) that produces location information, a location server (LS) that authorizes access to location information, a location recipient (LR) that requests and receives information, and a rulemaker (RM) that provides policy rules to the LS which enforce access control policies on access to a target.

Two policy rule namespaces have been defined. The first basic rule set [10] can restrict how long the receiver can retain the information and it can prohibit any further distribution of the information. It does not allow to customize information to specific receivers, for example. This document describes an enhanced rule set that provides richer constrains on location objects, including the basic rules in [10].

We refer to any element that uses the rules in this document to restrict the retention or distribution of information as a rule enforcer. Typically, these are LS and LR.

This rule set allows the rule enforcer to enforce access restrictions on location data, including prohibiting any dissemination to particular individuals or during particular times. The rule maker can also stipulate that only certain parts of the location object are distributed to recipients or that the resolution of parts of the location object is limited.

Below, we describe how different types of using protocols might employ the rules described in this document. We assume that a basic location object [10] can contain a reference to additional rule sets. Note that the protocols used to query location information (between

LS and LR), update policies at the location server and the protocol between the LG to LS and from LS to LR do not have to be the same.

In all cases, the abstract sequence of operations is similar. The LS receives either a query for location data or receives an update for the location of the target, via the using protocol. The using protocol provides the identity of the requestor, either at the time of the query or subscription to the location information. This verified location recipient identity, together with other information contained in the location information or generally available to the server, is then used to search the rule set. All matching rules are combined according to a merging algorithm described in this document. The resulting rule is applied to the location data, yielding a

possibly modified location object that is delivered to the location recipient.

A single LS may serve location data in more than one mode. Rather than having different rulesets for different modes, this document takes the approach of supporting all three modes in one ruleset schema. Specific instances of the ruleset can omit elements that are only applicable to the subscription model. The three different modes are explained below.

1.1 Passive Request-Response - LS as Server (Responder)

In a passive request-response scenario, the LR queries the LS for location information about the target. Examples of protocols following this mode of operation include HTTP, ftp, LDAP, finger or various RPC protocols, including Sun RPC, DCE, DCOM, Corba and SOAP.

The LS uses the rule set to prevent the transmission of information to the LR by refusing the request or to filter the information by removing elements or by reducing the resolution of elements.

1.2 Request-Response - LS as Client (Initiator)

Alternatively, the LS may contact the LR and convey the location information, e.g., to obtain some location-specific service. Examples include HTTP, SIP session setup (INVITE request), H.323 session setup or SMTP.

1.3 Event Notification

Event notification adds a subscription phase to the "LS as client" mode of operation. A watcher or subscriber asks to be added to the notification list for a particular presentity [cite] or event. When the presentity changes state or the event occurs, the LS sends a message to the LR containing the updated state. (Presence is a special case of event notification; thus, we often use the term interchangeably.)

In addition, the subscriber may itself add a filter to the subscription, limiting the rate [cite] or content [cite] of the notifications. If an event, after filtering by the rulemaker-provided rules and by the subscriber-provide rules, only produces the same notification content that was sent previously, no event notification is sent.

For SIP, the model is described in more detail in [cite].

Thus, for this model, a policy is needed for both subscription and

notification operations. While both can be expressed in separate policy rulesets, it is often helpful to synchronize the two. For example, it allows the location server to indicate to the subscriber what kind of information the subscriber can obtain. A subscriber may decide that the information is too limited to be of use and thus drop the subscription before wasting its own resources or those of the LS. (This does not reveal any additional information since the notification has to indicate the data received.)

2. Goals and Assumptions

Below, we summarize our design goals and constraints.

Table representation: Each rule must be representable as a row in a relational database. This design goal should allow efficient policy rule implementation by utilizing standard database optimization techniques.

Permit only: Allowing both 'permit' and 'deny' actions requires some rule ordering which has implications to the update operations executed on these rules. Additionally it makes distributed rule sets more complicated. An advantage is the more efficient rule handling. However, to avoid complex conflict resolution with

these rules only 'permit'. This also implies that rule ordering is not important. Consequently, to make achieve a policy decision it is necessary to process all policy rules.

Additive permissions: A query of the location recipient is matched against the rules in the rule database. If several rules fire then, the location recipient obtains the permissions of all rules.

Upgradeable: It should be possible to add additional rule elements later, without breaking location servers that have not been upgraded. Any such upgrades must not degrade privacy constraints, but may reveal less information than the rulemaker would have chosen.

Versioning support: In addition to the previous goal, a rule maker should be able to determine which types of rule elements are supported by the LS. We assume that the number of different versions of rule sets is best kept modest, so that a capability indication rather than an enumeration of fields is sufficient. This capability indication may take the form of indicating support for an XML namespace.

Protocol-independent: The rule set supports constraints on both notifications or queries as well as subscriptions for event-based systems such as presence systems.

No false assurance: It appears more dangerous to give the user the impression that the system will prevent disclosure automatically, but fail to do so with a significant probability of operator error or misunderstanding, than to force the user to explicitly invoke simpler rules. Among the rules proposed in earlier working group discussion, rules based on weekday and time-of-day ranges seem particularly subject to misinterpretation and false assumptions on part of the rulemaker. (For example, a non-technical rulemaker

would probably assume that the rules are based on the timezone of his current location, which may not be known to other components of the system.)

3. Non-Goals

We explicitly decided that a number of possibly worthwhile capabilities are beyond the scope of this first version. Future versions may include these capabilities, using the extension mechanism described in this document. Non-goals include:

No queries: Following the model being pursued for presence information, we do not support the notion of queries where the LR requests only a subset of the information available. This can readily be separated into a separate rule set and has different properties and authentication models.

No external references: Attributes within specific rules cannot refer to external rule sets, databases, directories or other network elements. Any such external reference would make simple database implementation impossible and would severely impact the scalability of the LS. External references also add significant failure handling complexity. An LS would have to make decisions on how long or often to attempt, what credentials to use and how to query. Thus, any such queries would have to be constrained to queries expressible as URIs.

No regular expression or wildcard matching: Conditions are matched on equality or 'greater-than'-style comparisons, not regular expressions, partial matches such as the SQL LIKE operator (e.g., LIKE "%foo%") or glob-style matches ("*@example.com"). Most of these are better expressed as explicit elements. For example, if it were desired to allow domain matches, the domain of the LR should be identified as a separate condition element.

No all-except conditions: It is not possible to express exclusion conditions based on identities such as "everybody except Alice". This is not only a ruleset limitation, but a limitation inherent in many access control systems (such as [8]). Limitations on the identity of this type do not provide rich functionality since it is easy for users such as Alice to acquire a different identity that is not blacklisted.

No repeat times: Repeat times are difficult to make work correctly, due to the different time zones that LG, LS and LR may occupy. It appears that suggestions for including time intervals are often based on supporting work/non-work distinctions, which unfortunately are difficult to capture by time alone.

4. Basic Data Model

A rule set consists of zero or more rules. The ordering of these rules is immaterial. The rule set can be stored in the LS and conveyed from RM to LS as a single document, in subsets or as individual rules. In addition, the LO can carry these rules as a data object, e.g., as multi-part MIME, or reference them via a URI. These alternatives are described in more detail in Section Section 4.2.

While the rules are encoding in XML, this is purely an exchange format between RM and LS. Additionally, if rules are attached to the location object then the rules are encoded in XML. The format does not imply that the RM or the LS use this format internally, e.g., in matching a query with the policy rules. The rules are designed so that a LS may translate the rules into a relational database table, with each rule represented by one row in the database. The database representation is by no means mandatory; we will use it as a convenient and widely-understood example of an interna representation. The database model has the advantage that operations on rows have tightly defined meanings. In addition, it appears plausible that larger-scale implementations will employ a backend database to store and query rules, as they can then benefit from existing optimized indexing, access control, scaling and integrity constraint mechanisms. Smaller-scale implementations may well choose different implementations, e.g., a simple traversal of the set of rules.

A rule consists of three types of elements (or fields): conditions, transformations and actions. (Some people have also used the term matching rules, matching conditions or constraints for what we term conditions.) One can think of the "conditions" part as the 'if' part of a statement, the transformations and actions as the "then" part of the statement. Since only one type of action i.e. 'permit' is allowed it does not need to be represented.

Transforming permissions ask the entity using the rules to modify the location object, e.g., to remove or modify a particular element. Actions cause the location server to take a certain action, such as adding a subscription.

4.1 Civil Location

TBD: The description below is only included for concreteness, since

a civil location object has not yet been defined and the description does not make sense without. It will migrate to a separate document.

This designation offers street-level precision.

The civil location elements are as follows:

Label	Description	Example
country	The country is identified by the two-letter ISO 3166 code.	US
A1	national subdivisions (state, region, province, prefecture)	New York
A2	county, parish, gun (JP), district (IN)	King's County
A3	city, township, shi (JP)	New York
A4	city division, borough, city district, ward, chou (JP)	Manhattan
A5	neighborhood, block	Morningside Heights
A6	street	Broadway
PRD	Leading street direction	N, W
POD	Trailing street suffix	SW

STS	Street suffix	Avenue, Platz, Street
HNO	House number, numeric part only.	123
HNS	House number suffix	A, 1/2
LMK	Landmark or vanity address	Low Library
LOC	Additional location	Room 543

	information	
FLR	Floor	5
NAM	Name (residence, business or office occupant)	Joe's Barbershop
PC	Postal code	10027-0401

Table 1

4.2 Rule Transport

We make no assumption as to how rules are conveyed to entities within the network. Purely as examples, we below describe a few plausible options. None of the elements depend on the properties of how rule sets are conveyed to an LS or LR. Mechanism may allow partial updates of rule sets. To simplify such partial updates, we include an identifier in each rule. This identifier is only unique among all the rules for a single target.

Transaction semantics for policy rule update is not required since 'permit only' and 'additive permissions' properties have to be used (as described in Section 2). These properties also prevent inconsistency during concurrent query and update operations.

4.2.1 HTTP

A rule set could be uploaded to the LS via an HTTP POST operation or more fully-featured WEBDAV. Each rule could be modeled as a single 'file', with the rule identifier as a file name. (Since multiple rules may have the LR identity in the condition part of the rule, the LR identity cannot be used.) One example of this approach includes XCAP [6].

The rule set can also be referenced from within a location object. The attribute 'ruleset-reference' specified in Section 2.2.2 of [10] and contains a URI that indicates where a fuller ruleset of policies related to this object can be found. The URI MAY alternatively use the CID URI scheme in which case it MUST denote a MIME body carried with the Location Object by the using protocol.

4.2.2 SIP Message Body

The rule set can be carried, as a separate MIME message body, in the

SIP message that conveys location information from a LG (a SIP UAC) via an LS (a SIP proxy) to an LR (a SIP UAS). The ruleset would then govern the behavior expected of the LR.

4.2.3 Location Object

The ruleset can be carried in location objects in two ways: by reference and by value. In any case the 'ruleset-reference' attribute inside the LO [10] points to to the location of the rules.

Instead the LG or LS can include the ruleset itself.

One of the transformations of the ruleset is the removal of the ruleset described here before further transmission. Only the whole ruleset can be removed.

4.3 Securing the Location Object

The Geopriv requirements draft [2] addresses the minimal security protection required for the Location Object: Mutual end-point authentication, data object integrity, data object confidentiality and replay protection. These security properties are implemented via

S/MIME and between elements. Protection for the LO includes any attached authorization rules.

Protection is likely to be necessary against adversaries who eavesdrop on the communication between the LS and the LR or the LG and the LS, who tamper with the location object or who replay previously recorded LOs. Securing the communication between rule maker and the LS depends on the protocol which is used to perform the desired actions (e.g., https). The communication between the LG and the LS can also be secured using channel security.

If the LO is integrity and confidentiality-protected then the receiving entity (LS or LR) has to be able to decrypt and to verify the LO. Since the policy rules described in this document allow the modification of the LO (via granularity reduction or by setting flags), it is not possible to forward the LO without reapplying the cryptographic protection. This is particularly true for the LS as it is not the final consumer of the LO.

When the LS protects the LO for transmission to the LR (after successful authorization), then the authenticated identity can be used to select a security association to apply proper protection of the location object. Securing the LO for multiple LRs is not provided.

Instead of encrypting the LO, the LG could digitally sign the LO,

offering integrity, but no confidentiality. However, if the LS needs to perform modifications on the LO, then it would have to break the digital signature and may apply its own digital signature.

Since the LO is generally distributed to more than one LR, the LG lacks the necessary information about the recipient and thus cannot usually apply confidentially protections.

By default, the LS re-signs LOs if the signed LO has been modified according to the rule set. If the LS receives an LO that it cannot decrypt, it discards it if and only if the rule requires modification of the content.

It remains for further study whether there should be an action that discards an LO that is signed or encrypted and needs to be modified

according to the matching rule set.

4.4 Extensions

The format is meant to be extensible. Each new extension needs to define a new namespace for its conditions and actions. A rule-enforcing entity MUST drop any rules where one or more of the <applies-to> elements or actions has an unknown namespace. It SHOULD simply ignore any transformations with unknown elements.

This behavior is privacy-safe since it prevents adding any permissions where the rule enforcer would not test for a particular condition. Consider an example: If a condition "Dow Jones Industrial Average" is added and a rule enforcer would ignore this condition, a watcher may obtain additional information even though the stock market condition is not met. Similarly, rules with unknown actions must be dropped since these actions may provide additional privacy protections, such as logging. Omitting transformations is safe, however, since this will only prevent the inclusion of data.

4.5 Identification

Each rule has an identifier, using the 'id' parameter of the rule element, which serves the purpose of partial updates as mentioned in Section 4.2. The identifier is an opaque token chosen by the rulemaker. A rule maker MUST NOT use the same identifier for two rules that are available to the LS at the same time for a given target. The combination <target identity, rule-id> uniquely identifies a rule.

4.6 Conditions

Conditions are identified by the <applies-to> element in a rule. In

all cases, conditions elements that are missing or empty are treated as if they contain a NULL value and always match.

4.6.1 Identity of the requestor (URI)

Location recipients are identified by URIs and matched by string matching on the <uri> element. Matching rules are governed by the URI scheme.

For some using protocols, the identity is encoded in the URI. It does not have to be the same identity that the protocol-specific authentication mechanism uses. For example, SIP URIs are sufficient to describe a requestor, but the user name for the SIP Digest authentication may differ from that identifier. (Some domains, for example, only require a name as the user identifier, while other use the full user@host form.)

For HTTP, the user name is not encoded in the URI. Thus, we define a new pseudo-URI scheme, http-auth, that carries the user name found in the authentication operation. For example, http-auth:alice@example.com.

It is left to the URI scheme and the using protocol to designate an identifier that denotes an 'anonymous user', i.e., a user that has not authenticated themselves. This allows to restrict anonymous access to users of a particular protocol, for example.

As an example, SIP can use the anonymous Digest authentication mechanism to grant access to holders of a particular secret. The rule sip:anonymous:myticket@anonymous.invalid would then match a requestor that includes the 'secret' myticket in the password part of the SIP request URI. This requires no special capabilities in the ruleset described here.

The definition of pseudonyms is left to each URI scheme and the related using protocol and any domain that it implies. For example, in SIP, a pseudonym might be 'sip:user42@coldmail.example', assuming that coldmail.example offers pseudonym accounts.

Anonymous users are treated by omitting this attribute in the rule which causes a 'NULL' value to be created in the ruleset table of a relational database. Any request for a location object (for a given target) would match with respect to this attribute in a rule.

4.6.2 Validity

The rule validity period is expressed as a two elements, a starting and ending time. Times are expressed in XML schema dateTime format.

An example of a rule fragment is shown below:

```
<valid-from>2003-08-15T10:20:00.000-05:00</valid-from>
<valid-until>2003-09-15T10:20:00.000-05:00</valid-until>
```

4.6.3 Sphere

The rule matches only if the target is currently in the state indicated. The state may be conveyed by manual configuration of the LS or by some protocol. For example, RPID provides the LG with the ability to inform the LS of its current sphere. The 'sphere' element is an XML schema token. An example of a rule fragment is shown below:

```
<sphere>work</work>
```

4.6.4 Civil Location

The Civil Location matches if the current civil location of the target matches all elements in the description given in the rule.

The civil location match includes a number of fields, including the timezone, the country (expressed as a two-letter ISO 3166 code), and the administrative units of [12] A1 through A6. This designation offers street-level precision.

If the civil location of the target is not known, rules that contain a civil location never match. (This case may occur, for example, if location information has been removed by earlier transmitters of location information or if only the geospatial location is known.)

If any of the elements <a1> through <a6> are specified, <country> also MUST to be specified. The schema does not enforce that the specification uniquely identifies a particular location. For example, it would be possible to omit the region and match only on city name, so that any city sharing that name within a country would match. This 'feature' is primarily designed to deal with users that may not know the administrative divisions between country and city level. For example, many users may not know the county for cities in the United States.

An example of a civil location condition fragment is shown below:

```
<country>US</country>
<a1>NJ</a1>
<a2>Bergen</a2>
<a3>Leonia</a3>
<a6>Westview</a6>
```

4.6.5 Geospatial Location

The geospatial location conditions make the rule apply if the target is currently located within the area bounded by a set of two longitude (longitude1, longitude2) and two latitude (latitude1, latitude2) values, describing a spherical trapezoid.

These four elements define a spherical trapezoid that is characterized as follows: the northern boundary of the spherical trapezoid is on the latitude given by the latitude1 element, and the southern boundary is on the latitude given by the latitude2 element - the western boundary is on the longitude given by the longitude1 element, and the eastern boundary is on the longitude given by the longitude2 element.

4.7 Conditions

Conditions are in conjunctive normal form, i.e., all defined elements in the rule have to match the query. It is possible to leave some attributes within a row empty or omit the element altogether; these attributes then assume the default value of NULL. NULL entries always match.

Expressed in database terms, queries on the rule set can be modeled as SELECT queries, as in

```
SELECT permission1,permission2, ... FROM ruleset
WHERE (condition1='p1' OR condition1 IS NULL)
AND (condition2='p2' OR condition2 IS NULL)
AND (...)
```

Here, p1, p2, etc. are properties of the querier, time of day or properties of the target, such as its current location. The attributes available for these conditions are described in Section 4.6. After determining the rules which fire for a given query the permissions have to be combined. Combining the permissions is described in Section 7 and leads to a final result which is applied to the location object.

Expressed in programming terms, the process of determining a result

can be written as (the first part represents the query):

```
  for each policy rule in ruleset do {
    if ((condition1 == 'p1' || condition1 == NULL)
      && (condition2 == 'p2' || condition2 == NULL)
      ,..
    then {
      collect permissions and add them to the firing-ruleset
    } // end if
  } // end for
  result = Apply permission-combining algorithm on firering-ruleset
  Transform location object based on result
```

Conditions can be of four datatypes:

a string (e.g. civil location)

an integer

a date (e.g. validity)

Naturally, sets are equivalent from an implementation perspective to integers.

In addition to the equality operator, range operators ("is between"), set intersection and inequality operators are also supported. Only one type of operator is defined for each element.

4.8 Procedure for combining Permissions

This section describes the procedure for combining permissions in case that multiple rules fire. As indicated in Section 2 permissions are additive i.e. a LR obtains permissions of multiple firing rules. The assumption is made that the attributes are ordered and that the value of one attribute does not depend on the value of another (different) attribute.

Combining permissions depends on the following four datatypes:

undefined (NULL)

an integer or enumeration

a boolean (true or false)

A query will be matched against all rules and any number of rules might fire. The permissions of all fireing rules are combined

according to permission-specific combining rules. The combining rules are simple and depend on the datatype:

Boolean: If any boolean row for a transformation is true, the result is true. If all rows are either undefined or false, the result is false.

Integer or enumeration: The result is the maximum across all rows. If all rows are 'NULL', the interpretation depends on the permission type.

5. Access Control for Policies

Each rule set is logically associated with exactly one target; we can consider this target to be another condition column in the rule set. A single rule maker can manage rule sets for a large number of targets, and a location server can be provided with rules for different targets by different rule makers.

Access restrictions to the rule set itself are beyond the scope of this document. A complete geo-privacy system would need to specify how a location server can verify the identity and authorization of a rule maker to insert, update or delete a particular rule.

A simple mechanism, followed in XCAP Section 2, restricts modifications to the target itself, but third-party authorizations are likely to be useful.

6. Actions

Actions describe what the recipient of the rule is allowed to do if the rule matches. Note that subscriptions are automatically allowed for any subscriber that matches a ruleset, possibly after confirmation. To refuse a subscription, the rulemaker simply omits the undesirable subscriber from the ruleset.

Actions that modify the location object have a default value of NULL. The behavior of NULL actions differs for parts of the LO describing the location and parts of the LO describing usage rules. For the former, only components that are explicitly included through non-NULL actions are kept by the LS. For the usage rules, elements are left unchanged unless a non-NULL action modifies the rule.

This approach ensures that extensions in the capabilities of the location server do not suddenly change the behavior of the location server for the same rule set. The behavior is also

privacy-preserving, under the assumption that removing location objects can only enhance privacy and that keeping unknown usage rules also does not diminuish privacy.

The operations defined in Section 4.8 are also applicable in this context if multiple rules fire.

6.1 Subscription Duration

The Subscription Duration action only applies to event-based or presence systems. It indicates the duration, measured in seconds, that the watcher is allowed to subscribe to this event. This action is ignored for queries. The default value is 3600 seconds (one hour).

6.2 Confirm Subscription

This action only applies to using protocols that follow a subscription model.

If the Confirm Subscription flag is set, the principal whose information is being desired has to approve the subscription. The subscription is marked as 'pending' while the server waits for the presentity to decide. The approval mechanism depends on the using protocol and is beyond the scope of this document. As an example, SIP defines a mechanism where the presentity is notified of the subscription attempt [cite] and then updates the ruleset to either allow or refuse the subscription.

It is an error if both the Confirm Subscription and Subscription

Duration action are non-null.

The default value for this attribute is 'true'.

7. Transformations

In addition to the transformations below, LS MAY translate and add location information. For example, they may add timezone information

based on civil information.

All transformations are privacy-safe, i.e., if a transformation is NULL (i.e., if the attribute is not present or empty in a policy rule), the LS removes the corresponding location information from the LO and leaves the LO flags undisturbed.

Extensions to this document may define other transformations.

7.1 Set D (Distribute) Flag

This transformation sets the D flag in the location object to either 'true' or 'false'. A value of 'true' means the recipient of the LO is allowed to further distribute it. A value of 'false' prevents further distribution.

The value NULL keeps the D flag in the LO as is.

7.2 Set R (Retention) Time

The retention transformation sets the retention value in the location object to the current time plus the time provided in the element, measured in seconds.

The value NULL keeps the retention time in the LO as is.

7.3 Keep Rule (RR)

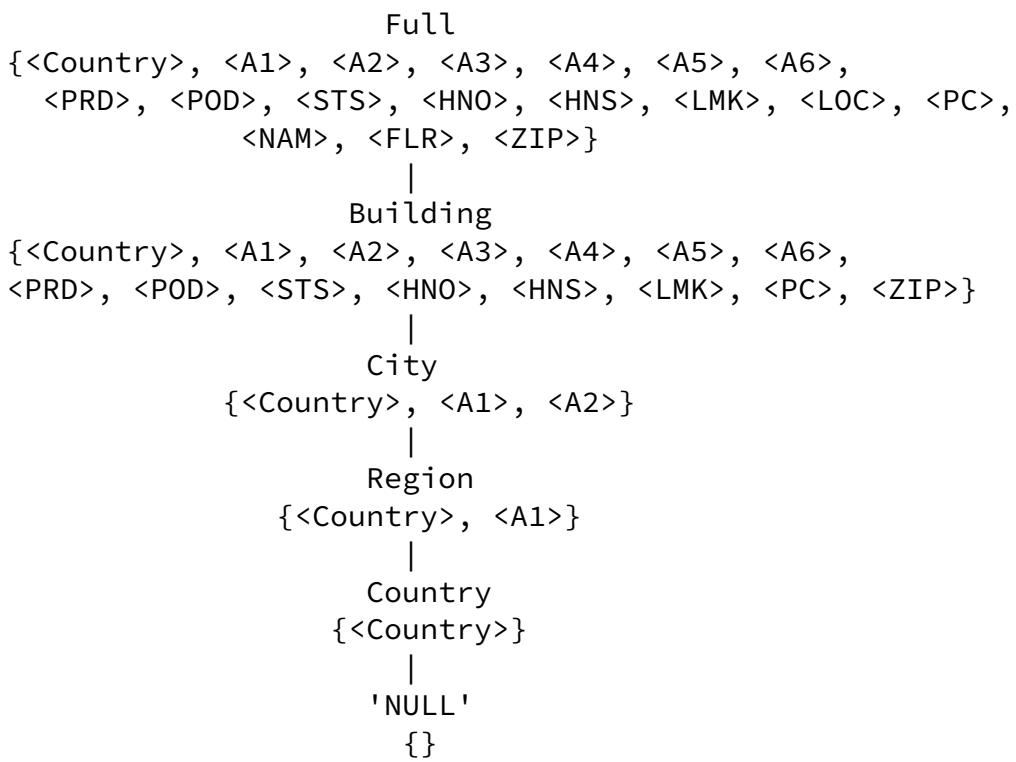
If the Keep Rule (RR) flag is set, any extended rules included in the location object are kept.

7.4 Provide Civil Location

The Provide Civil Location transformation restricts the civil location to one of six levels, from lowest to highest: null, country, region, city, building, full. Each level includes all elements included by the lower levels. The 'country' level includes only the <country> element; the 'region' level adds the <a1> element; the 'city' level adds the <a2> and <a3> elements; the 'building' level and the 'full' level add further civil location data as shown below.

If this action is NULL, all civil information is removed from the LO.

The lattice for this attribute has the following shape:



7.5 Provide Geospatial Location

The Provide Geospatial Location transformation restricts the resolution of the geospatial location information to the number of bits provided, separately for longitude and latitude. The default value is zero.

For purposes of this transformation, longitude and latitude are treated as a 34 bit fixed point value consisting of 9 bits of integer and 25 bits of fraction. Altitude is treated as a fixed-point 22-bit integer part with a 8-bit fraction, measured in meters. This corresponds to the representation in [7], but does not constrain the representation in the location object.

If the transformation value is NULL, all geospatial location information is removed from the LO.

7.6 Provide Timezone Flag

The Provide Timezone transformation includes the timezone of the target, i.e., the offset from UTC. The value of 'false' causes

timezone information to be excluded from the LO.

If the transformation value is NULL, all timezone information is removed from the LO.

8. Example

This section lists some basic examples to show the functionality.

8.1 Example for a rule

This example rule illustrates a ruleset with a single rule. The rule consists of three parts: an <applies-to> part which represents the conditions, an <action> part and a <transformations> part. The conditions match to a location requestor named alice@example.com. The rule is valid for one month (2003-08-15 to 2003-09-15). Requests only match if the target has set its sphere identifier to "work" and it is currently located at the indicated civil location. The <transformations> indicate that the granularity of the location information is reduced for both civil and for geospatial location information. The D flag is set to 'true' and the rules included in the LO are kept.

```
<ruleset>
  <rule id="1234567890">
    <applies-to>
      <uri>pres:alice@example.com</uri>
      <valid-from>2003-08-15T10:20:00.000-05:00</valid-from>
      <valid-until>2003-09-15T10:20:00.000-05:00</valid-until>
      <sphere>work</work>
      <country>US</country>
      <a1>NJ</a1>
      <a2>Bergen</a2>
      <a3>Leonia</a3>
       <a6>Westview</a6>
    </applies-to>
```

```
      <actions>
        <subscription>1800</subscription>
        <confirm>true</confirm>
      </actions>

      <transformations>
        <civil>region</civil>
        <set-retention>10</set-retention>
        <set-distribution>true</set-distribute>
        <keep-rules>true</keep-rules>
        <latitude-resolution>9<latitude-resolution>
        <longitude-resolution>9<longitude-resolution>
        <altiude-resolution>6<altitude-resolution>
      </transformations>
    </rule>
```



```
  </ruleset>
```

8.2 Permission-combining Example

TBD: Example should go in here.

9. XML Schema

This section describes an XML schema for the authorization policies described in the previous sections. It does not extend other schemas. It is a preliminary version primarily to show the simplicity of the policies. In [5] the XML schema in the XCAP Usages for Setting Presence Authorization draft [4] to have a better alignment with policies used in SIP presence. It is expected that future versions of [4] will experience some changes. We do not abandon either one of the approaches.

```
<?xml version="1.0" encoding="UTF-8"?>
  <xs:schema
    targetNamespace="urn:ietf:params:xml:ns:ruleset"
    xmlns:rs="urn:ietf:params:xml:ns:ruleset"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    elementFormDefault="qualified"
    attributeFormDefault="unqualified">
```

```
<xs:element name="ruleset">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="rs:rule" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="rule">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="applies-to" type="rs:applies-toType"/>
      <xs:element name="actions" type="rs:actionsType"/>
      <xs:element name="transformations" type="rs:transformationsType"/>
    </xs:sequence>
  <xs:attribute name="id" type="xs:string" use="required"/>
  </xs:complexType>
</xs:element>


<xs:complexType name="applies-toType">
  <xs:all>
    <xs:element name="uri" minOccurs="0" type="xs:anyURI" />
    <xs:element name="valid-from" minOccurs="0" type="xs:dateTime" />
    <xs:element name="valid-to" minOccurs="0" type="xs:dateTime" />
    <xs:element name="sphere" minOccurs="0" type="xs:string" />

     <xs:element name="country" minOccurs="0" type="xs:string" />
```



```
    <xs:element name="a1" minOccurs="0" type="xs:string" />
    <xs:element name="a2" minOccurs="0" type="xs:string" />
    <xs:element name="a3" minOccurs="0" type="xs:string" />
    <xs:element name="a4" minOccurs="0" type="xs:string" />
    <xs:element name="a5" minOccurs="0" type="xs:string" />
    <xs:element name="a6" minOccurs="0" type="xs:string" />

    <xs:element name="prd" minOccurs="0" type="xs:string" />
    <xs:element name="pod" minOccurs="0" type="xs:string" />
    <xs:element name="sts" minOccurs="0" type="xs:string" />
    <xs:element name="hno" minOccurs="0" type="xs:string" />
    <xs:element name="hns" minOccurs="0" type="xs:string" />
```

```
        <xs:element name="lmk" minOccurs="0" type="xs:string" />
        <xs:element name="pc" minOccurs="0" type="xs:string" />
        <xs:element name="zip" minOccurs="0" type="xs:string" />
        <xs:element name="nam" minOccurs="0" type="xs:string" />
        <xs:element name="flr" minOccurs="0" type="xs:string" /

        <xs:element name="geospatial-location">
          <xs:complexType>
            <xs:all>
              <xs:element name="longitude1" type="xs:double" />
              <xs:element name="longitude2" type="xs:double" />
              <xs:element name="latitude1" type="xs:double" />
              <xs:element name="latitude2" type="xs:double" />
         </xs:all>
          </xs:complexType>
        </xs:element>

      </xs:all>
    </xs:complexType>

    <xs:complexType name="actionsType">
      <xs:all>
        <xs:element name="subscription" minOccurs="0" type="xs:integer" />
        <xs:element name="confirm" minOccurs="0" type="xs:boolean" />
      </xs:all>
    </xs:complexType>


    <xs:complexType name="transformationsType ">
      <xs:all>
        <xs:element name="civil" minOccurs="0">
          <xs:complexType>
            <xs:choice>
              <xs:element name="full" type="xs:string" />
              <xs:element name="building" type="xs:string" />
              <xs:element name="city" type="xs:string" />
```



```
              <xs:element name="region" type="xs:string" />
              <xs:element name="country" type="xs:string" />
            </xs:choice>
          </xs:complexType>
        </xs:element>
```

```
        <xs:element name="set-retention" minOccurs="0" type="xs:integer" />
        <xs:element name="set-distribution" minOccurs="0" type="xs:boolean" />
        <xs:element name="keep-rules" minOccurs="0" type="xs:boolean " />
        <xs:element name="set-retention" minOccurs="0" type="xs:integer" />
        <xs:element name="longitude-resolution" minOccurs="0" type="xs:integer
        <xs:element name="latitude-resolution" minOccurs="0" type="xs:integer"
        <xs:element name="altitude-resolution" minOccurs="0" type="xs:integer"
        <xs:element name="provide-timezone" minOccurs="0" type="xs:boolean" />
      </xs:all>
    </xs:complexType>

  </xs:schema>
```

10. Security Considerations

This document aims to make it simple for users to prevent the unintended disclosure of private information to third parties. The described policies accomplish this task. Threats applicable to this draft are described in [3] and requirements are addressed in [2]. Section 4.3 addresses issues of protecting the policy rules within the LO and location information itself. Aspects of privacy-safe combining permissions is illustrated in Section 7.

11. Open Issues

Some open issues have been identified during the process of working on this draft:

Transformation: A default behavior for transformation attributes (see Section 7) has to be defined to indicate what behavior has to be assumed for omitted values (i.e. for values how appear as 'NULL' values in an attribute of row. Currently it seems that it is not sufficient to specify a single default behavior rule for all transformation attributes in order for them to be privacy-safe. Instead at least a differentiation between

Attributes whose removal may lessen privacy (the R/D/microwave flags) and

Attributes whose removal can only increase privacy (location objects).

Extensions: More text needs to be provided on how to deal with extensions and version conflicts.

Logging: Functionality for triggering logging has been started with [14] and continued in [5]. For this draft it was decided not to provide encryption and logging functionality as part of the policy rules. For logging a simple mechanism would be to just have this be a binary flag. It is then up to the LS to decide where to log this and how. One problem with logging is that it does not apply if the rule set is included in the LO. Asking the LR to log the information is meaningless in most cases since the target/RM would not be able to access this information. While logging seems simple, it also causes interactions with the retention mechanism. If someone sends a LO that says "do no retain" and "log", you have a problem. Additionally, you may want to specify exactly what gets logged: Just the fact that user A got location information at a certain time? The precise location information he did get, after filtering? Just the fact that location information was distributed to some set of users, which you can derive by looking at the rule set? Finally, logging would be done in general for each target, not just for one recipient. Hence there is a question why it should be done based on one particular rule.

Encryption: For encryption similar concerns are applicable as for logging. It needs to be decided whether encryption should be handled on a per-rule basis and whether a single flag would be sufficient. Some further issues deserve attention such as:

What happens if the public key is not known for the recipient?

What encryption technique should I use?

Should an identity be provided within the rule?

What happens if this identity specifies a SIPS URI (i.e., requiring that the request be via SIP-over-TLS) and the flag is not set, should it be refused?

If encryption is specified and the request was HTTP (not HTTPS), should the request be refused?

Identities: A number of issues have been discussed with regard to identities of the LR (specified in the URI attribute).

Authentication Types: Element E (Permission to disclose only to someone presenting a specified key) of [14] was implemented in [5] as authentication levels (see Section 2.1 in). [4] specifies the elements <auth-mechanism> and <anonymous> inside the acceptance permission which allows to refer to SIP specific authentication mechanisms such as None, TLS, Digest, SMIME and P-Asserted-ID. For this draft is was decided to omit these attributes since they might be difficult to understand for end users, difficult to realize since authentication levels are not universally defined and in case of specific authentication mechanisms it is difficult to imply the meaning for Geopriv.

Notifications: A concept which appears to be simple is to require notice to the rule maker if location is provided (Element K of [14]. The concept of 'notice' might be meaningless without saying how. Currently there is no such mechanism define in Geopriv and therefore there are no parameters which might be needed for this operation.There are certainly many possible protocols on how to notify the RM, including event notification (using SIP, Jabber,

SOAP events, etc.), email, some kind of RPC mechanism, an HTTP request to a specific address, syslog. Currently we think that this issues deserves further discussion.

Permission-Combining Example: An example describing the permission-combining algorithm has to be provided for Section 8. Working on this example the concept of a 'default' rule was discussed. The term default rule refers to a rule where the condition elements are all set to 'NULL'. This rule will fire with every query.

Lying: Functionality for lying by the LS is not supported.

A number of other minor issues are still buried in the drafts [5], [14] and in [4].

Normative References

[1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[2] Cuellar, J., Morris, J., Mulligan, D., Peterson, J. and J. Polk, "Geopriv requirements", draft-ietf-geopriv-reqs-03 (work in progress), March 2003.

[3] Morris, J., Danley, M. and J. Peterson, "Threat Analysis of the geopriv Protocol", draft-ietf-geopriv-threat-analysis-01 (work in progress), September 2003.

Informative References

[4] Rosenberg, J., "Extensible Markup Language (XML) Configuration Access Protocol (XCAP) Usages for Setting Presence Authorization", draft-ietf-simple-xcap-auth-usage-00 (work in progress), June 2003.

[5] Tschofenig, H., Morris, J., Cuellar, J., Polk, J. and H. Schulzrinne, "Location Object Authorization Policies", draft-tschofenig-geopriv-authz-00 (work in progress), August 2003.

[6] Rosenberg, J., "The Extensible Markup Language (XML) Configuration Access Protocol (XCAP)",

draft-ietf-simple-xcap-00 (work in progress), June 2003.

[7] Polk, J., Schnizlein, J. and M. Linsner, "DHC Location Object within GEOPRIV", draft-ietf-geopriv-dhcp-lo-option-00 (work in progress), January 2003.

[8] Gong, L., "Inside Java 2 Platform Security", Addison Wesley, Reading, Massachusetts Addison Wesley, Reading, Massachusetts, June 1999.

[9] Mealling, M., "The IETF XML Registry", draft-mealling-iana-xmlns-registry-05 (work in progress), June 2003.

[10] Peterson, J., "A Presence-based GEOPRIV Location Object Format", DRAFT draft-peterson-geopriv-pidf-lo-01, September 2003.

[11] Schulzrinne, H., "RPID - Rich Presence Information Data Format", draft-ietf-simple-rpid-00 (work in progress), July 2003.

[12] Schulzrinne, H., "DHCP Option for Civil Location", draft-ietf-geopriv-dhcp-civil-00 (work in progress), July 2003.

[13] Sugano, H. and S. Fujimoto, "Presence Information Data Format (PIDF)", draft-ietf-impp-cpim-pidf-08 (work in progress), May 2003.

[14] Morris, J., Mulligan, D. and J. Cuellar, "Core Privacy Protections for Geopriv Location Object", draft-morris-geopriv-core-02 (work in progress), July 2003.

Authors' Addresses

Henning Schulzrinne
Columbia University
Department of Computer Science
450 Computer Science Building
New York, NY 10027

US

Phone: +1 212 939 7042
EMail: hgs+geopriv@cs.columbia.edu
URI: http://www.cs.columbia.edu

John B. Morris, Jr.
Center for Democracy and Technology
1634 I Street NW, Suite 1100
Washington, DC 20006
US

EMail: jmorris@cdt.org
URI: http://www.cdt.org

Hannes Tschofenig
Siemens
Otto-Hahn-Ring 6
Munich, Bayern 81739
Germany

EMail: Jorge.Cuellar@siemens.com
URI: http://www.cdt.org

Jorge R. Cuellar
Siemens
Otto-Hahn-Ring 6
Munich, Bayern 81739
Germany

EMail: Jorge.Cuellar@siemens.com
URI: http://www.cdt.org

James Polk
Cisco
2200 East President George Bush Turnpike
Richardson, Texas 75082
US

EMail: jmpolk@cisco.com

Appendix A. Contributors

Jonathan Rosenberg
dynamicsoft
600 Lanidex Plaza
Parsippany, NJ 07054-2711
USA
Email: jdrosen@dynamicsoft.com

Appendix B. Acknowledgments

This document is based on the discussions within the IETF GEOPRIV working group. Discussions at the Geopriv Interim Meeting 2003 in Washington, D.C. helped the working group to make progress on the authorization policies based on the discussions among the participants. We would to particularly thank Jon Peterson for his helpful comments. Additionally, we would like to thank Christian Guenther for his work on the XML schema.

Intellectual Property Statement

Full Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.