LoST: A Protocol for Mapping Geographic Locations to Public Safety Answering Points

Henning Schulzrinne*, Hannes Tschofenig†, Andrew Newton‡, Ted Hardie§
*Columbia University, New York, NY (hgs@cs.columbia.edu)
†Siemens Networks GmbH & Co KG, Munich, Germany (hannes.tschofenig@siemens.com)
‡SunRocket, Vienna, VA (andy@hxr.us)
§Qualcomm, San Diego, CA (hardie@qualcomm.com)

***Abstract*— Public Safety Answering Points (PSAPs) serve limited geographic areas, so emergency callers must be directed to the most appropriate PSAP. As part of the overall Internet Engineering Task Force (IETF) emergency services architecture, we have developed a new protocol, LoST (Location-to-Service Translation), that allows end systems and VoIP proxies to map location data into URLs representing either PSAPs or other SIP proxies that perform a more fine-grained mapping. LoST is designed to operate globally, with a highly-distributed authority. We describe the protocol and its design rationale.**

I. Introduction

The transition from the PSTN to VoIP forces a reconsideration of how to provide emergency calling services, as many of the assumptions underlying the existing "9-1-1" system in North America and similar systems elsewhere in the world are no longer true. In particular, network access services and voice services may be provided by different entities; larger enterprises may operate their own voice services, without the involvement of a traditional telephone carrier. While for traditional landline telephones the telephone number can be used as a key to identify the caller's location, the caller's IP address is assigned dynamically and not tied to a specific location. Thus, the old approach to managing emergency calls is no longer viable.

The need to change the emergency calling architecture also affords an opportunity to offer improved services, such as bidirectional multimedia and text messaging, as well as to improve scaling, reliability and resiliency. Fig. 1 shows a simplified overview of the overall architecture [1, 2] that is emerging for providing next-generation emergency calling services, although this is only one example of many variations. The picture shows a split network, where PSAPs are located within a separately-managed emergency services network, protected by firewalls against some forms of Internet-based attacks. However, systems could have a single level of mapping, or more than two. The architecture defined there is a part of the NENA[1] long-term architecture for emergency services, sometimes referred to as NG911 or I3.

In this VoIP-based emergency calling model, end systems acquire location information, either directly through GPS or indirectly through link layer protocols such as LLDP-MED, configuration protocols such as DHCP [3, 4] or a layer-7 location configuration protocol yet to be defined [5]. The end system then contacts a mapping server to obtain the URL of a SIP [6] proxy that serves a PSAP (Public Safety Answering Point, i.e., the emergency call center) or some other organization, such as a state agency, that in turn performs a more fine-grained mapping and directs the call to the right PSAP. The location information is carried from the calling user agent to the destination either by value or by reference in a SIP header field [7].

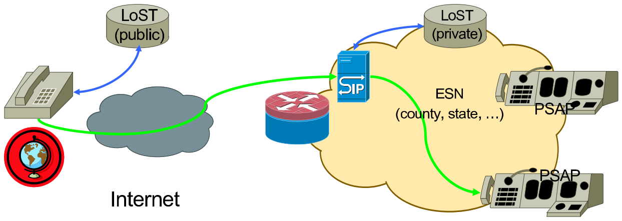

Fig. 1. Simplified rendition of IP-based emergency calling architecture

Fortunately, most of the components described above have either already been defined for non-emergency applications or are relatively simple additions to existing configuration, VoIP and IM protocols. However, there is currently no standardized protocol for mapping location information to (PSAP) URLs. Such a mapping protocol is the focus of this paper. Solving the mapping problem is the core mission of the Internet Engineering Task Force (IETF) Emergency Context Resolution with Internet Technologies (ECRIT) working group. After defining a set of general [8] and security threats and requirements [9], a number of solutions were proposed, which eventually converged to a protocol called LoST, for Location-to-Service URL Translation.

While LoST has been designed to serve as a mapping protocol for finding PSAPs, it is not limited to that particular function. We envision that LoST can be used as a mapping protocol for other location-dependent services, both governmental and commercial. For example, in the United States, there are other widely-known service numbers, such as 411 for directory assistance or 311 for non-emergency governmental services, where services are provided by different providers for each region. While the focus of this paper is on citizen-to-authority

[1] National Emergency Number Association, a professional association for the North American 9-1-1 system

communications, determining the areas served by first responders is an important part of an overall emergency services architecture that avoids manually-maintained and error prone records. For non-emergency services, one can imagine using LoST to find instances of commercial services with a limited service region, such as automatic teller machines, restaurants, towing services, gas stations or food delivery businesses. (As designed, the mapping is based on service regions, but it can be readily extended to provide proximity matching, by adding a new location profile.) Using LoST for other services has the advantage that it makes it more likely that robust commercial and open-source implementations emerge and that ISPs will offer resolver services.

The remainder of the paper is organized as follows. We introduce the overall LoST architecture in Section II, describe the client-server protocol in Section III and an early implementation in Section IV.

II. The LoST Architecture

Mapping locations to PSAP URLs is just one, albeit critical, function for a globally interoperable and robust emergency services networks. Thus, LoST is integrated into other components of an IP-based communications architecture for emergency calls [1]. We are generally assuming that emergency calls use SIP [6] for setting up and terminating calls, as this is the most widely-used standards-based VoIP protocol. However, LoST and the basic architecture are largely independent of the signaling protocol and would, for example, also work for XMPP [10], Skype, Jingle and other proprietary VoIP protocols. LoST itself is carried in HTTP messages.

In brief, users placing an emergency call dial either the local or home emergency service number, such as 911 in North America, where the number is configured through LoST, as described later. The user agent recognizes the call as an emergency call, inserts a special service URN [11], such as urn:service:sos, into the call setup request, and consults it internal table for the PSAP URL it should route the request to. The PSAP URL has been determined earlier by invoking LoST with the current location of the caller. (For non-emergency services, a client can determine the mapping on demand.)

Services are identified by service URN (uniform resource name) [11], which are globally unique and common names for emergency and other services. Currently, only names for emergency and counseling services have been defined, with examples such as 'urn:service:sos.fire' or 'urn:service:sos.police'. These identifiers are not generally visible to human users, who would continue to dial the familiar emergency numbers, such as 911 or 112.

A. Goals and Requirements

Mapping is a critical function for emergency calling. If the mapping process fails, is delayed excessively or yields the wrong answer, emergency calls either fail or are delivered to the wrong PSAP, which is unlikely to be able to dispatch first responder in a timely fashion. In particular, during mass-casualty events, the mapping function needs to survive even in the face of overload or intentional attack on the network infrastructure. For robustness and to avoid post-dialing delays, end systems should generally make mapping queries before they place an emergency call, i.e., any time they move to a new PSAP service area. (Since it is desirable that an emergency call reaches the PSAP within two seconds of completing dialing, the mapping function can take only about one second to complete.) To preserve the location privacy of callers, the identity of the user should not be part of the mapping query.

Unfortunately, location determination technology and manually-entered location information, particularly for street addresses, sometimes has errors, which could then deliver emergency calls to the wrong PSAP or cause the PSAP to dispatch aid to the wrong location. Thus, the mapping architecture makes use of the fact that the mapping can occur ahead of the emergency call to validate addresses, so that users can correct errors or contact their service providers.

For emergency calling, there are clear jurisdictional responsibilities for service regions. Each jurisdiction is responsible for its own service boundaries and the street addresses that it contains, but generally has limited ability to track the service boundaries in other jurisdictions. Indeed, some jurisdictions may well have a non-cooperative or hostile relationships with other jurisdictions.

Given that IP-based mobile devices can be bought anywhere and used anywhere Internet connectivity is available, the architecture needs to be international, rather than tied to one region or language.

Mappings can be performed by both end systems such as softphones on PC or mobile phones, in the Internet spirit of having intelligent end systems, or by SIP proxies, which are used by voice service providers to route calls. Thus, given that Internet access and voice services may be provided by different, possibly competing, organizations, mapping services must be able to be provided by either Internet access provider, voice service providers or third parties, without these having any direct business relationship to the ISP or VSP.

These requirements led to the development of a distributed architecture, with a protocol that relies heavily on caching, as discussed in more detail in the next section.

B. Resolvers, Trees and Forest Guides

We distinguish four core components in the LoST architecture: seekers, resolvers, authoritative mapping servers and forest guides. A simple example is shown in Fig. 2. *Seekers* are looking for location-to-URL mappings, but do not respond to queries. As discussed in Section II-C, they cache query results. Seekers cannot know all the trees that contain authoritative information from the numerous jurisdictions in the world. Instead of trying to configure seekers with that information, we introduce a special server, a *resolver*, whose role it is to know about jurisdictions and to cache mapping results. The resolver would likely be operated by an enterprise, the Internet service provider or the voice service provider, but could in principle be operated by anybody. Thus, there will likely be tens of thousands of such resolvers once LoST is widely deployed.

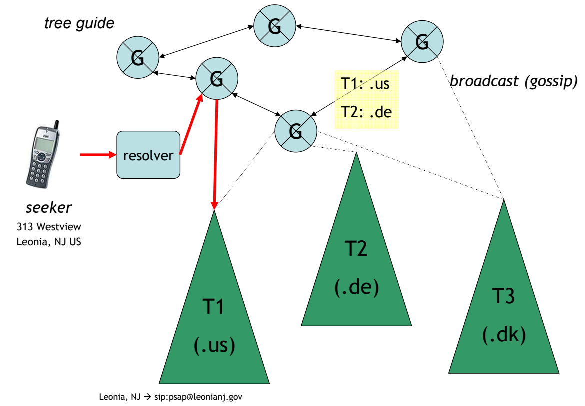

Fig. 2. LoST seekers, resolvers, authoritative mapping servers and tree guides

Authoritative mapping information can only be provided by jurisdictions or their designated service providers. They know which geographic area is served by which PSAP and how this PSAP should be reached. The service area can be described by a polygon with longitude and latitude coordinates or it can be a set of civic addresses, such as a county or city. However, authority for this mapping information is handled very differently in different countries, with some countries having a very small number of PSAPs and a nationwide management of location information, while others, such as the United States, have high degrees of local autonomy, with more than 6,000 PSAPs. (For example, in New Jersey, many boroughs and townships with a few thousand inhabitants run their own PSAP.) The authoritative mapping servers form a tree, with the node representing the largest coverage region at the logical top of the tree. Leaf nodes point only to PSAPs, while interior nodes generally point to other tree nodes.

While there is generally a hierarchy of jurisdictions within a country, there is no obvious global authority that can coordinate a mapping system. Given territorial disputes between countries, it may also be difficult to find a single global tree root. However, since VSPs can serve customers from all over the world and have customers that roam globally without changing their telephone number or SIP URL, users need to be able to locate the tree representing their current physical location.

While manually configuring resolvers with all the trees is theoretically possible, it is error-prone, particularly when the organization of trees changes. (For example, it is likely that during the initial deployment of LoST, smaller jurisdictions, such as states, that have more advanced emergency calling systems would operate their own authoritative trees, before national coordination mechanisms can be put in place.) Thus, we introduce a light-weight directory service for trees, called forest guides. Forest guides form a mesh and generally all contain the same set of records, one for each tree. Each record describes the service, the coverage region of the tree and the address where the top-most authoritative node of the tree can be contacted. Forest guides peer with neighbors and zero or more trees, and propagate new coverage information amongst themselves, in a fashion roughly analogous to BGP. (The protocol details for this synchronization mechanism remain to be worked out.) In theory, forest guides could, by policy, decide not to accept or distribute certain coverage regions, e.g., because it contradicts its national policies regarding the territory claimed by different countries. Since resolvers can connect to any forest guide, a resolver operator would presumably choose a forest guide that is likely to provide useful information for its customers.

The roles of seeker, resolver, authoritative mapping server and forest guides are logical roles and a single server can fulfill several of them at the same time. For example, a resolver could also act as an authoritative mapping server and the top of an authoritative tree could act as a forest guide or resolver. The protocol generally does not concern itself with roles; rather, the roles are defined by the protocol messages that a server understands and behavior that it exhibits.

As noted earlier, LoST can be used not just for emergency calls, but for locating first responders and non-emergency calls. In principle, each service could have its own hierarchy of LoST servers, operated by independent organizations or collections of organizations. It is more likely that only each top-level services, such as 'urn:service:sos', has its own hierarchy.

C. Caching

Any reliable emergency communication system needs to remove single points of failure and should limit the ability of malicious parties to interfere with services by taking down critical components. One crucial component of reliability is to distribute information as much as possible, so that, with high likelihood, a user can find the information close by. This principle underlies, for example, the Domain Name System (DNS) [12]. Thus, for LoST, we need to ensure that failure of the authoritative servers, be they attacker-induced or accidental, does not immediately prevent emergency calls for a whole region.

In the LoST architecture, all elements can, if they so desire, cache mappings. However, unlike other directory protocols, cache entries are located not by label match, but rather by finding matches by region. If a geographic location in a query falls within the shape of an existing entry, that entry is returned, as long as it has not expired. Initially, shapes are polygons, but more complicated shapes are feasible. (Finding point-in-polygon matches can be relatively efficient and is implemented by many GIS databases, such as PostGIS or the GIS extensions to mySQL.) Each mapping contains the coverage region, the service URL and other mapping-related information.

Caching can avoid having to query the LoST hierarchy during an actual emergency, when there is only a second or two available for all call routing and lookup actions. Thus, in this architecture, user agents query LoST servers when they boot up or when they discover a change in their location, e.g., for nomadic and mobile end systems, rather than waiting until the actual emergency call. This approach also has the advantage that servers are less likely to be overloaded during mass casualty events when hundreds of thousands of users might attempt to place emergency calls. Determining mappings ahead of time also facilitates placing test calls, specially marked as such, ahead of an emergency, to ensure that all components are working properly. This type of result caching is not directly feasible if mapping is delegated to proxies, but in some circumstances, proxies serving a limited geographic region, such as an enterprise or regional ISP, can issues queries on their own and thus prime their cache.

Caching is crucial to efficiently support mobile end systems. A mobile end system cannot know when it has left the coverage region of one PSAP and entered that for another, so in a naive protocol, it would have to continuously query the LoST server to make sure it has information that is appropriate for its current location. Clearly, this would impose a tremendous burden on the LoST servers and would drain the battery of mobile devices. With end system caching, the mobile end system simply checks periodically whether it has left the region indicated in the mapping entry and only then issues another query, assuming it does not already have an entry for its new location cached. We anticipate that mapping information for PSAPs has lifetimes on the order of days or weeks, so that end systems or proxies would only issue at most a few queries a day. Naturally, travelers traversing long distances would generate more queries. The volume of queries depends on the coverage region of PSAPs and whether queries return the PSAP URL itself or the address of some larger-region server, e.g., a state-wide emergency routing proxy.

To further increase reliability and deal with high query loads, we envision that most LoST servers are actually clusters of servers that can all provide the same mapping results. The members of each cluster synchronize their mapping databases and caches with each other. All of them are advertised within DNS NAPTR [13] records, so that clients can automatically load-balance queries among servers or pick a working server if one or more servers in the cluster have failed.

III. The LoST Query Protocol

The core component of the LoST architecture is the LoST client-server protocol. It is a relatively simple XML-based query-response protocol, currently defined to operate on top of HTTP or HTTPS, although other protocol mappings are also possible. (In particular, using SOAP envelopes is straightforward.) Fig. 3 illustrates a query and response for mapping a civic address, using the <findService> request and the corresponding <findServiceResponse> response.

As discussed earlier, the LoST architecture distributes mapping information globally and hierarchically. The user agent originates the query and asks its local resolver. If the resolver does not have the answer, it in turn queries the forest guide to find the tree for the particular region, such as a country. Servers for that country recursively issue the same query to the appropriate lower-level server, until it reaches a server that has the relevant information. The information then propagates back to the original querier, traversing the servers in reverse order. The path of the query and its response is captured in a set of <via> elements, to facilitate debugging.

```
<?xml version="1.0" encoding="UTF-8"?>
<findService xmlns="urn:ietf:params:xml:ns:lost1"
  recursive="true" validateLocation="true"
  mappingInclude="uri serviceNumber
    displayName serviceBoundary" >
  <location
    profile="civic">
    <civicAddress xmlns="urn:...:civicAddr">
      <country>Germany</country>
      <A1>Bavaria</A1>
      <A3>Munich</A3>
      <A6>Neu Perlach</A6>
      <HNO>96</HNO>
      <PC>81675</PC>
    </civicAddress>
  </location>
  <service>urn:service:sos.police</service>
</findService>
```

Fig. 3. Example LoST query

Instead of this recursive resolution mechanism, servers can also simply respond with the address of another server, iteratively guiding the original querier closer to the server that has the authoritative answer. Iterative queries require less state in each server, but impose more of a burden on the user agent, increasing the number of queries it has to issue. The query volume could be important for clients that are on low-bandwidth links, such as mobile phones.

The core query is naturally the <findService> query that maps a point describing the location of the querier to a service URL for a PSAP, the emergency service number and the service region which is served by the same URL. Since some features and data may consume additional bandwidth or processing time, the querier can specify which response elements it would like to receive. In the example, it asked for address validation using the <valid>, <invalid> and <unchecked> XML elements.

A. Location Profiles

The format of location information depends on the wireless location determination technique, particularly in how uncertainty is represented. These shapes can be encoded in a variety of ways, even if the GIS standard GML The location of end systems and coverage regions can be expressed in a variety of ways, even if the GIS standard GML [14] is used. Beyond the basic two-dimensional geospatial (longitude, latitude) and civic addresses, it might be desirable in the future to include three-dimensional locations that include altitudes or more complicated shapes. To facilitate evolving the protocol while ensuring interoperability, a query can contain multiple location objects, drawn from different location profiles. At least one of the location objects must be either a civic address or a two-dimensional point.

B. Service Numbers

Currently, landline phones rely on the first switch to recognize emergency number, while cell phones draw on a short list of emergency numbers configured through the cell network or the SIM card. For example, all GSM phones recognize 1-1-2 as an emergency number, regardless of the caller's location. For an international system that generalizes to other services, these mechanisms are insufficient. For emergency services alone, there are more than 100 national emergency numbers in the world, some of which have non-emergency uses in other countries. It is desirable that a mobile device is automatically configured to recognize the emergency number for the country that the device is currently in, as well as that for the country that a visitor calls home. LoST helps in configuring devices with local service numbers, as it returns the service number as part of the mapping response. (Longer term, we envision that the address book of mobile devices is pre-configured with emergency and other common services, such as directory assistance, but these entries simply use the service URNs, not numbers.)

C. Address Validation

In the current PSTN-based emergency services architecture, street addresses in the Automatic Location Identification (ALI) database can be verified against the master street address guide (MSAG), to make sure that every phone number is associated with a street address that actually exists and can be dispatched to. For next-generation emergency calling, it is desirable to offer at least the same functionality. LoST incorporates the ability to validate street addresses. (In the IETF, the term "civic address" is used instead of street addressees, which can be either jurisdictional or postal.) If a query contains a set of flags, the server resolving the query will compare all elements of the civic address, from the name of the country, to the county and city, down to the house number and maybe even suite or apartment number. Depending on the level of detail contained in the database, either all or part of the address will match. If there is no match, the server prunes back elements, starting from the bottom of the civic address hierarchy, e.g., the suite number. If the remainder of the civic address matches, it then indicates to the querier which parts of the address matched, which parts did not match and which parts were not compared since the database did not contain relevant information. This approach allows fine-grained validation once such data becomes available; today's MSAG only contains street names and house number ranges.

As noted earlier, the end system issues LoST queries independent of an emergency call, so that the user can be notified of any validation errors long before an emergency occurs. Validation failures are particularly likely if addresses have been entered manually, as may be unavoidable until robust location determination techniques have been universally deployed.

D. Service Lists

In addition to mappings, clients can also ask servers for the list of services they support, using the `<listServices>` request. For a resolver, this indicates which forest guides it knows about. When asked for a specific location, the response enumerates the (emergency) services that can be invoked for that region. This response could be used to create an emergency service list in the address book of a mobile phone, for example.

E. Finding LoST Servers

Emergency services need to work without users having to manually configure aspects of the service. Also, LoST resolvers can be operated by a number of entities, in particular Internet service providers (ISPs) and voice service providers (VSPs). To allow for both, LoST defines several mechanisms to discover suitable LoST resolvers. ISPs and enterprises often use DHCP to configure aspects of the Internet service, so a DHCP option for discovering LoST servers has been defined [15]. If operated by the VSP, SIP configuration mechanisms [16] used to configure other aspects of VoIP service can also indicate a suitable LoST server. Alternatively, the user name, known as the SIP address-of-record, can be used to locate, via DNS NAPTR records [13, 17], a suitable LoST server.

F. Service Boundaries

Earlier on, we mentioned that mapping responses indicate the service region that includes the query point, typically expressed as a polygon with longitude and latitude point coordinates. For civic addresses, the service region indicates the matching civic address elements. For example, a region of "country = US, state = NJ, county = Bergen" would indicate that all queries within Bergen county would fall within that service region. Since service regions may not align exactly with cities or counties, the service region returned may actually be only a subset of the real service region, but this only marginally affects the query rate. (Queries for civic addresses are only issued by stationary or nomadic devices.)

Unfortunately, some boundary polygons can contain around a thousand points, yielding boundaries that are several kilobytes in size. On the other hand, such service regions are expected to change very rarely, possibly on the order of years and decades. The remainder of the mapping record, such as the service URL, may change more frequently. Thus, in some deployments, such as mobile end systems, it may be desirable to avoid sending the geographic service boundary with each query response. Instead, the mapping response contains a reference to the boundary, identified by the server that can provide the actual polygon and a version identifier. The client can retrieve it using the `<getServiceBoundary>` request. If the boundary changes, the mapping response simply includes a new version number for the boundary and the client updates the cached version by contacting the owner of the boundary data or its local resolver, where it might already be cached.

IV. Implementation and Performance

At Columbia University, we have implemented a first prototype of a LoST client and a LoST authoritative server [18]. We have integrated a Tcl-based LoST client into sipc [19], our SIP-based multimedia client, to demonstrate how end systems can perform location mappings. The LoST server was implemented using Java, running on the Apache Axis2 SOAP stack running within Tomcat. As a backend database, we used PostGIS, which "adds support for geographic objects to the PostgreSQL object-relational database." PostGIS directly supports the necessary point-in-polygon representation to look up geospatial mappings. We have performed preliminary measurements of the system, using a dual Pentium 4 (3 GHz) with 1 GB of RAM running Linux 2.6. For civic data, the response size for our example is 768 bytes and it takes the server 87 ms to respond to a query. The client can parse the response in 32 ms. Geographic queries take longer (492 ms) to answer and parse (224 ms), partially because they are significantly longer in our test, at 9215 bytes. We achieved a query rate of 110 queries per second, apparently limited by the PostGIS database. We believe that this figure can be improved significantly by better use of database caching and indexing.

V. Related Work

There have been numerous proposals and standards for general Internet-scale directory protocols, including LDAP [20]. However, as far as we know, none of these have considered the special requirements for a global, distributed location-to-URL mapping service. While theoretically capable of operating in a distributed hierarchy, LDAP is mostly used in local area networks, as a single logical server, while LoST is designed from the beginning to answer questions in a distributed fashion.

LoST shares a number of features with DNS [12]. In particular, DNS also has a hierarchy of servers, starting with the root servers and employs caching in local resolvers. This shared architecture provides scalability and robustness. However, unlike DNS, LoST is able to provide more extensive responses, a hierarchy based on coverage regions rather than a label hierarchy and has location-based caching, rather than simply caching a single record identified by a label. Also, the forest guides mechanism generalizes the concept of the collection of root servers in DNS, allowing largely independent entities to operate parts of the tree. This will hopefully avoid the need for a global coordination body such as ICANN, and the concomitant political and legal overhead.

VI. Conclusion and Future Work

We have presented an overview of LoST, a protocol for mapping geographic locations to URLs, as a core component of an open-standards-based emergency calling system. LoST is designed for the special needs of geographic queries, supporting region-based caching and independent hierarchies for different services and regions, with forest guides as the glue at the top of the hierarchy. An initial implementation shows that standard web services and GIS database open-source software can be used to implement the system.

The core query protocol is being finalized within the context of the IETF ECRIT working group. We hope to conduct the first interoperability tests in early 2007. The protocol among forest guides and other components of the replication architecture are still under development, but the protocol can be deployed within a VSP, for example, without standardizing such a mechanism.

Acknowledgments

The LoST protocol was implemented by Jong Yul Kim and Wonsang Song.

References

[1] B. Rosen, H. Schulzrinne, J. Polk, and A. Newton, "Framework for emergency calling in internet multimedia," Internet Draft draft-ietf-ecrit-framework, Internet Engineering Task Force, Oct. 2006.

[2] B. Rosen and J. Polk, "Best current practice for communications services in support of emergency calling," Internet Draft draft-ietf-ecrit-phonebcp-00, Internet Engineering Task Force, Oct. 2006.

[3] H. Schulzrinne, "Dynamic host configuration protocol (dhcpv4 and dhcpv6) option for civic addresses configuration information," RFC 4776, Internet Engineering Task Force, Nov. 2006.

[4] J. Polk, J. Schnizlein, and M. Linsner, "Dynamic host configuration protocol option for coordinate-based location configuration information," RFC 3825, Internet Engineering Task Force, July 2004.

[5] H. Tschofenig and H. Schulzrinne, "GEOPRIV layer 7 location configuration protocol; problem statement and requirements," Internet Draft draft-tschofenig-geopriv-l7-lcp-ps-03, Internet Engineering Task Force, Oct. 2006.

[6] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, "SIP: session initiation protocol," RFC 3261, Internet Engineering Task Force, June 2002.

[7] J. Polk and B. Rosen, "Session initiation protocol location conveyance," Internet Draft draft-ietf-sip-location-conveyance-05, Internet Engineering Task Force, Oct. 2006.

[8] H. Schulzrinne and R. Marshall, "Requirements for emergency context resolution with internet technologies," Internet Draft draft-ietf-ecrit-requirements-12, Internet Engineering Task Force, Aug. 2006.

[9] T. Taylor, H. Tschofening, H. Schulzrinne, and M. Shanmugam, "Security threats and requirements for emergency call marking and mapping," Internet Draft draft-ietf-ecrit-security-threats-03, Internet Engineering Task Force, July 2006.

[10] P. Saint-Andre, "Extensible messaging and presence protocol (XMPP): core," RFC 3920, IETF, Oct. 2004.

[11] H. Schulzrinne, "A uniform resource name (URN) for services," Internet Draft draft-ietf-ecrit-service-urn, Internet Engineering Task Force, Aug. 2006.

[12] P. Mockapetris, "Domain names - concepts and facilities," RFC 1034, Internet Engineering Task Force, Nov. 1987.

[13] M. Mealling, "Dynamic delegation discovery system (DDDS) part three: The domain name system (DNS) database," RFC 3403, Internet Engineering Task Force, Oct. 2002.

[14] Open GIS Consortium, "Opengis geography markup language (GML) encoding specification," Specification OGC 03-105r1, OpenGIS, Feb. 2004.

[15] H. Schulzrinne, J. Polk, and H. Tschofenig, "A dynamic host configuration protocol (DHCP) based location-to-service translation protocol (lost) discovery procedure," internet draft, Internet Engineering Task Force, Sept. 2006.

[16] D. Petrie, "A framework for SIP user agent configuration," Internet Draft draft-ietf-sipping-config-framework-09, Internet Engineering Task Force, Oct. 2006. Work in progress.

[17] L. Daigle, "Domain-based application service location using uris and the dynamic delegation discovery service (DDDS)," internet draft, Internet Engineering Task Force, Oct. 2006.

[18] J. Y. Kim, W. Song, and H. Schulzrinne, "An enhanced voip emergency services prototype," in *Information Systems for Crisis Response and Management (ISCRAM)*, (Newark, New Jersey), ISCRAM, May 2006.

[19] X. Wu and H. Schulzrinne, "sipc, a multi-function SIP user agent," in *7th IFIP/IEEE International Conference on Management of Multimedia Networks & Services*, (San Diego, California, USA), Oct. 2004.

[20] J. Hodges and R. Morgan, "Lightweight directory access protocol (v3): Technical specification," RFC 3377, Internet Engineering Task Force, Sept. 2002.