SIPPING Working Group
Internet-Draft
Expires: September 6, 2006

A. Johnston
SIPStation
H. Sinnreich
pulver.com
March 5, 2006

SIP, P2P, and Internet Communications
draft-johnston-sipping-p2p-ipcom-02

Status of this Memo

Copyright Notice

Abstract

This draft discusses issues related to the application of peer to peer (P2P) technologies to SIP in particular, and Internet communications in general. After an analysis of the P2P and non-P2P capabilities of SIP, this draft proposes that a P2P protocol be standardized in the IETF as a protocol used between a Registrar/ Proxy/Redirect server and a Location Service. This allows the operator of a Registrar to decide how much registration state is be

stored locally and how much can be distributed using the P2P network to a distributed Location Server. A number of DHT (Distributed Hash Table) P2P protocols that solve some similar functions are given as examples and could be used as input to this work. Finally, existing SIP and P2P work is surveyed with respect to this proposal.

Table of Contents

1. Introduction

P2P technologies have been widely used on the Internet in file sharing and other applications including VoIP, IM, and presence. Other Internet-Drafts have been written which explore some of the requirements [4] and example implementations [3] of P2P SIP.

This draft attempts to analyze the current P2P capabilities in SIP, then discusses the non-P2P aspects of SIP. The high level operation of the proposed Location Service Protocol is outline. Some P2P research work using the Chord protocol is then surveyed. Finally, existing SIP and P2P work is compared to this proposal.

2. P2P Capabilities in SIP

SIP [1] actually already has quite a lot of inherent P2P capabilities, although most deployments of SIP today barely take advantage of them. For instance, all servers in SIP are optional, allowing User Agents (UAs) to directly communicate. Even when a server such as a proxy server is utilized, after the initial exchange, subsequent messaging is routed on a peer to peer basis using the Contact URI. Even presence can be published and retrieved in a peer to peer manner [2]. However, much development in SIP has been in the area of intermediaries. While the standard specification discusses in detail the roles of registrars, proxy servers, and redirect servers, many actual deployments of SIP have instead used B2BUA intermediaries which completely break the P2P properties of SIP by design.

Efforts in SIP to make Contact URIs routable outside dialogs seemed to be moving SIP in a P2P direction. However, the current work in this area (i.e. standardizing GRUUs, Globally Routable User Agent URIs [12]) unfortunately mixes the definition of the GRUU and a particular acquisition mechanism. The proposed mechanism actually permanently includes intermediaries in the signaling path. As such, the currently defined GRUU mechanism is not applicable to P2P SIP. Hopefully alternative GRUU mechanisms compatible with P2P will be developed in the future as the full implications of the current approach are realized.

As a result, SIP is not yet a true Internet protocol. It is used today in closed networks, within walled gardens, and in mediated-middle networks. Much of its complexity is a side effect of its deployment in these environments. Some of us, however, have used SIP in P2P mode over the Internet for our personal communication for years.

For SIP to truly become an Internet protocol, it needs to escape these closed networks and spread in the public Internet. The best way to make this happen is for SIP to take advantage of P2P technology and truly harness the P2P properties of SIP, rendering the closed networks and their intermediaries irrelevant.

3. Making SIP a True P2P Protocol

As hinted in the previous section, there are a variety of non-technical reasons why the P2P capabilities in SIP have not been widely utilized. In this section, some technical reasons why SIP is currently not truly P2P are discussed. This leads to a proposal to correct this.

Consider the typical routing of a SIP request over the SIP Trapezoid introduced in RFC 3261 [1] is reproduced in part in Figure 1 below.

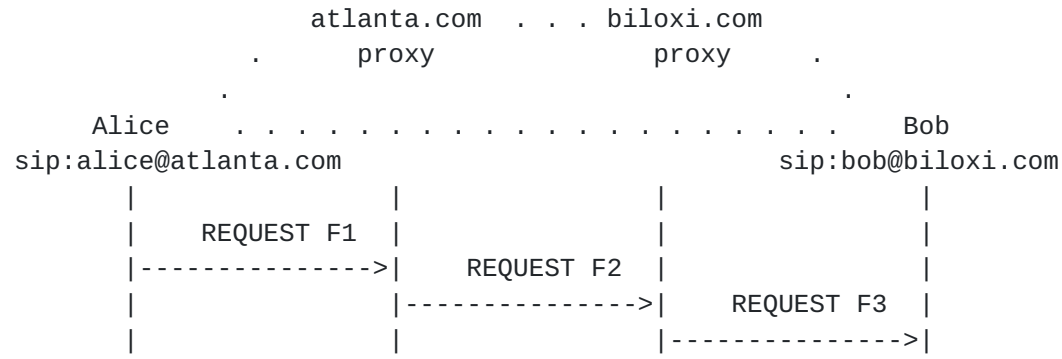

Figure 1. SIP Request Routing with the Trapezoid

An analysis of why this typical interdomain request flow involves multiple proxies provides some useful insights into how SIP can operate in a more P2P manner.

Alice wants to send a SIP request to Bob using Bob's Address of Record (AOR) sip:bob@biloxi.com. The request is shown as first routing through the atlanta.com proxy server before being routed to the biloxi.com proxy server. As such, atlanta.com is acting as a "outbound proxy" server. The use of a default outbound proxy server could be required if the UA, for example, does not support the DNS queries necessary to locate the biloxi.com domain. Another reason might be that the outbound proxy server is performing some NAT/ firewall traversal or other services needed to allow Alice to communicate with the outside world.

Today, all UAs support DNS queries including SRV, so the DNS argument is no longer valid - Alice is perfectly capable of locating the

biloxi.com server without assistance from the atlanta.com server. In addition, since the publication of RFC 3261, we have other NAT traversal techniques such as STUN, TURN, and ICE. As such, it is possible to remove the atlanta.com proxy server from this call flow, collapsing the trapezoid to a triangle.

However, the biloxi.com proxy server is not so easy to eliminate. The NAT/firewall traversal services provided can likely be migrated to STUN and TURN as in the outbound proxy case. However, the proxy server can only be bypassed if Alice can discover Bob's Contact URI (sip:bob@192.0.2.4 in this example) which routes directly to Bob's UA. UAs publish this AOR/Contact URI binding using registration, which is then maintained in the network as soft state.

Note that a domain can create AORs which utilized non-SIP methods to maintain this soft state. For example, if Bob uses an AOR URI which has a host part which resolves using dynamic DNS, then registrar and proxy servers can be removed from the call flow. This requires Bob's SIP devices to run a dynamic DNS client which sends a message to the dynamic DNS Server to update Bob's DNS A record each time Bob's IP address changes. In this mode, Bob no longer sends REGISTERs - the dynamic DNS updates take the place of registrations. (Note that in this context, dynamic DNS is not DNS Dynamic Updates [5],[6]. For information on dynamic DNS in this context, Google "dynamic DNS").

This dynamic DNS approach works, and has been used by a number of us for many years to successfully run P2P SIP. Note, however, that many SIP clients require a successful registration before permitting outgoing requests, even though this goes explicitly against RFC 3261 which makes it clear that a successful registration has no impact on outgoing requests. Note also that this approach is actually not P2P, as there is still a client/server dynamic DNS protocol being used.

Assuming that dynamic DNS is not used, let us examine closely the reasons why the biloxi.com proxy must remain in this call flow. Bob's UA generates registration messages which contain all the information needed to perform P2P SIP routing directly to Bob's UA. This registration information is sent by Bob to the registrar for the biloxi.com domain. The information is then stored in the Location Service for the biloxi.com domain.

RFC 3261 formally defines a Location Service as:

> A location service is used by a SIP redirect or proxy server to obtain information about a callee's possible location(s). It contains a list of bindings of address-of-record keys to zero or more contact addresses. The bindings can be created and removed in many ways; this specification

defines a REGISTER method that updates the bindings.

This information is then generally only available to proxy servers in the biloxi.com domain. Also, note that the protocol between a registrar and a Location Service or a proxy and a Location Service has never been standardized in the IETF. It is the one protocol in the SIP Trapezoid which is not standardized, and it is this lack of standardization that traps the registration data in the biloxi.com domain and precludes true P2P SIP.

In giving SIP tutorials, I am often asked why this protocol has not been standardized. There are a number of reasons given in the past:

1. There has been no push to standardize this protocol, and the SIP Trapezoid model does not require it to be standardized.

2. The IETF typically does not standardize decomposition protocols such as these, although there are notable exceptions such as MEGACO.

3. In addition to the AOR/Contact URI binding, the Location Service is often used to store service logic. Since the IETF does not standardize services, a standard protocol to access a Location Service would be difficult to define without also standardizing (and hence limiting) the services.

However, I believe that given the desire to make SIP more P2P, these reasons no longer apply. Specifically:

1. Standardizing a Location Service Protocol would allow the biloxi.com proxy server to be bypassed, allowing true interdomain P2P communication between Alice and Bob, a goal shared by many of us in the SIP community for a variety of scalability, privacy, and security reasons.

2. While this protocol could be used to decompose a SIP Server farm (and this is a very useful thing) within a domain, its use in an interdomain P2P network makes it an Internet protocol and hence of interest to the IETF.

3. In the pure P2P model, services are exclusively provided by endpoints, not intermediary servers, reducing the requirements of his protocol to purely AOR/Contact URI binding. The analysis in [4] confirms this view and suggests that even standard telephony services can be provided by peers without resorting to centralized servers.

As such, the IETF should standardize this protocol to enable true P2P SIP.

As an aside, the IETF has developed TRIP (Telephony Routing over IP) [7] which is an inter-Location Service protocol used to discover PSTN gateway routes. It is possible that TRIP could be extended beyond telephony routing to allow Location Servers to exchange AOR/Contact URI bindings. However, it is fundamentally a routing protocol rather than a URI discovery mechanism and as such not suitable for this application.

4. Operation of a Location Service Protocol

The new Location Service Protocol (LSP) (Editor's Note: we definitely need a better name so as to be clear that this protocol has nothing to do with GEOPRIV) would be used between a Registrar, Proxy, or Redirect Server and a Location Service. Other Internet Communication Protocols could also utilize this protocol. For example, in a H.323 network, the protocol could be used between a RAS Server and its database.

The basic functions are publication of AOR/Contact URI binding, and the querying of AOR/Contact URI binding. Publication would be performed by suitably authorized registrars for a domain, while querying could be performed by proxy servers, redirect servers, or even UAs in other domains.

In a conventional SIP network, only the SIP servers would need to support this new protocol - UAs would not need to support a new protocol to take advantage. SIP Servers performing lookups using the Location Service Protocol would redirect the results of the query.

A Registrar of a domain can decide if it wants to operate a local Location Service. If so, the protocol can be utilized to store and retrieve the information in the local Location Service. Request routing within the domain could consult this local Location Service for routing. DNS SRV records would be populated to point to the Proxy Servers which would query the local Location Service using the protocol. In short, normal SIP operation.

In addition, or instead of a local Location Service, the Registrar may opt to publish registration data in a public Location Service. The protocol could then be used by users in other domains to query the public Location Service, discover the Contact URI of a particular user, and route SIP requests directly, bypassing the domain's Proxy Servers.

The reasons why a domain might decide to only publish registration information to the public Location Service are obvious - the domain no longer needs to store this state and maintain SIP Proxy or

Redirect Servers for incoming requests.

The reasons why a domain might decide to publish this information both locally and publicly are similar. If the information is published locally then the domain still has to run SIP Proxy and Redirect Servers - however, the more users that utilize the public Location Service, the lighter the load on the SIP Servers, reducing cost and complexity for the domain.

In this hybrid mode, users have the choice of either performing a DNS query and routing though a SIP Server or using the Location Service Protocol

The most interesting scenario from a P2P perspective is when the Registrar decides to keep no registration state. In this case, the Registrar would simply statelessly translate a REGISTER request into the corresponding Location Service Protocol message and publish this information into the public Location Service P2P network.

Note that if a UA acts as its own Registrar, it can utilize the Location Service Protocol to publish its information directly into the public Location Service. This mode is the "pure" P2P operation. In this way, the Location Service Protocol does not replace DNS or provide conflicting information, as only the registrar authoritative for the domain would publish information. Once a request is sent, normal SIP identity and security mechanisms can be used to verify that the correct destination has been reached.

5. Location Service Protocol as a P2P Protocol

This proposed Location Service Protocol would not necessarily need to be a P2P protocol. None of the arguments in the previous sections requires the protocol to be P2P.

In the case where the protocol is used within a single domain, the protocol need not be P2P. However, the more interesting case where the protocol is used to establish a distributed public Location Service, or a confederation or clearing house Location Service, the scaling requirements point to a P2P protocol. The need for nodes to join and leave, and for distributed storage and retrieval points to a P2P protocol.

The protocol should allow a joining peer to learn the scaling size of the P2P network and provide a mechanism to insert itself into the overlay network.

6. NAT Traversal

In some P2P Internet communications networks, the discovery, rendezvous, and NAT traversal functions are combined into a single network and service. In fact, joining some networks require you to agree to provide all of these functions, despite the disparate resource requirements.

Logically, the Location Service Protocol proposed in this draft should be separate from any NAT traversal functions. A node agreeing to participate in a distributed Location Service network should not have to agree to participate in offering NAT traversal. Also, the underlying P2P protocols, optimized for discovery and rendezvous, should not be complicated and burdened with NAT traversal issues. Instead, these protocols should assume that peers manage their own NAT traversal.

The discovery of NAT traversal relay services is a useful P2P functionality in itself, however, and is an orthogonal topic. Once a node behind a NAT acquires a relay, it may then participate in the distributed Location Service P2P network.

Note: The NAT traversal required is not identical to TURN [17] as it is defined today. The "lock down" property of TURN that limits it to relaying between a pair of hosts is a useful security property in a peer-wise media session. However, this property will block the arbitrary inter-node communication needed for normal P2P communication. As such, a relay acquired for the purposes of allowing a node behind a NAT to participate in a P2P network, for example, will be similar, but not exactly the same as a TURN server.

7. P2P Implementation using Distributed Hash Tables

Distributed Hash Tables (DHTs) are an active research area in the P2P community that is highly scalable and offers efficient, low latency search and retrieval of data over an overlay network. These algorithms use a hash function to map a search key to a set of node numbers. Data related to that particular search key are then stored and retrieved from this set of nodes.

The search key in the distributed Location Service Protocol would be an Address of Record URI. This URI is effectively a name, and would require resolution to a particular device (URL) or Contact performed in real time.

As an example set of functions provided by the P2P protocol is:

o Lookup of a key. Returns a set of addresses of peer nodes that store information about the key.

o Retrieval of the data from a key node or nodes.

o Publishing data to key node or nodes.

Chord [9] is an example of a distributed hash lookup primitive. There is an active open source research project (http://www.pdos.lcs.mit.edu/chord/) which provides the first of these functions using Remote Procedure Calls (RPCs). With the addition of the other two functions, complete P2P systems can be constructed.

As another example, the CFS (Cooperative File System) [10] is a read-only file system built on top of Chord that utilizes P2P techniques to request the retrieval of data. CFS utilizes load balancing techniques to break stored data into chunks and randomly distribute them across a number of nodes. Chord is used to maintain routing tables identifying which node stores which blocks. The second and third functions are provided by DHash which stores the data reliably in a number of nodes. CFS layers on top a file system that puts the retrieved blocks together as a complete data file. In another example, DDNS [13] is a Chord-based approach for providing DNS lookups.

Besides DHTs, there are other classes of P2P algorithms including CAN (Content Addressable Network) [14], Pastry [15], and Tapestry [16], The problem statement for each of these algorithms bears a striking similarity to the P2P discovery and rendezvous capability that would be useful in a SIP P2P network.

This body of work should be taken as an input to any IETF Location Service Protocol standardization effort.

8. SIP P2P Implementations

Some preliminary work [3], [8] has been published on the use of SIP and Chord. However, these SIP approaches propose utilizing SIP as a transport, with all P2P messages tunneled over SIP. These implementations are good demonstrations of the use of DHT algorithms and the use of P2P and SIP, but are not a good starting point for the design of the Location Service Protocol.

Specifically, the approach of tunneling P2P messages over a SIP REGISTER request has the following issues:

- It is SIP specific. A distributed Location Service Protocol would be of use to more protocols than just SIP, and requiring non-SIP protocols to implement the REGISTER method is not likely to be acceptable.

- It is not a logical extension of the REGISTER method. REGISTER is only used in SIP between a UA and a Registrar Server. Redirection of REGISTER requests is uncommon, and likely to be interpreted as an attempt at registration hijacking. To interoperate with non-P2P SIP networks, this would require a Registrar Server to initiate REGISTER requests on behalf of a UA - undesirable for a number of architectural and security reasons.

- SIP transport has a high overhead and transactional state cost. For large P2P networks, this is likely to translate into long search delays and overloading of the query network.

9. Conclusion

This draft has discussed the P2P capabilities of SIP and identified P2P limitations in current SIP usage. An analysis of SIP message routing over the SIP Trapezoid was used to motivate the proposal to standardize a protocol to implement a distributed Location Service. The NAT traversal requirements of this protocol were briefly discussed. Finally, the draft included a brief discussion of the applicability of some SIP P2P approaches and this proposal.

10. Acknowledgements

I'd like to thank the members of the SIP community for their conversations about P2P SIP including Henry Sinnreich, Henning Schulzrinne, Robert Sparks, Cullen Jennings, Jon Peterson, Adam Roach, Adrian Georgescu, David Bryan, and Philip Matthews. In addition, thanks to the Chord developers especially Emil Sit for their feedback.

11. Security Considerations

SIP utilization of P2P discovery and rendezvous techniques will introduce a number of new security, identity, and privacy considerations that will need to be solved. As a starting point, general P2P security papers such as [18] should be studied, then considerations specific to the Internet communications application should be applied.

12. Informative References

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

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

[3] Bryan, D. and C. Jennings, "A P2P Approach to SIP Registration and Resource Location", draft-bryan-sipping-p2p-01 (work in progress), July 2005.

[4] Matthews, P., "Industrial-Strength P2P SIP", draft-matthews-sipping-p2p-industrial-strength-00 (work in progress), February 2005.

[5] Vixie, P., Thomson, S., Rekhter, Y., and J. Bound, "Dynamic Updates in the Domain Name System (DNS UPDATE)", RFC 2136, April 1997.

[6] Wellington, B., "Secure Domain Name System (DNS) Dynamic Update", RFC 3007, November 2000.

[7] Rosenberg, J., Salama, H., and M. Squire, "Telephony Routing over IP (TRIP)", RFC 3219, January 2002.

[8] Singh, K. and H. Schulzrinne, "Peer-to-peer Internet Telephony using SIP", Columbia University Technical Report CUCS-044-04, New York, NY October 2004.

[9] Stoica, I., Morris, R., Karger, D., Kaashoek, M., and H. Balakrishnan, "Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications", ACM SIGCOMM 2001, San Diego, CA August 2001, pp. 149-160.

[10] Dabek, F., Kaashoek, M., Karger, D., Morris, R., and I. Stoica, "Wide-area Cooperative Storage with CFS", Proceedings of the 18th SOSP 2001.

[11] Dabek, F., Kaashoek, M., Li, J., Morris, R., Robertson, J., and E. Sit, "Designing a DHT for Low Latency and High Throughput", NSDI 2004 March 2004.

[12] Rosenberg, J., "Obtaining and Using Globally Routable User Agent (UA) URIs (GRUU) in the Session Initiation Protocol (SIP)", draft-ietf-sip-gruu-06 (work in progress), October 2005.

[13] Cox, R., Muthitacharoen, A., and R. Morris, "Serving DNS using a Peer-to-Peer Lookup Service", First International Workshop on Peer-to-Peer Systems (Cambridge, MA, Mar. 2002).

[14] Ratmasamy, S., Francis, P., Handley, M., Karp, R., and S. Shenker, "A scalable content-addressable network", Proc. ACM SIGCOMM, San Diego, CA August 2001, pp. 161-172.

[15] Rowstron, A. and P. Druschel, "Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems", Proceedings of the 18th IFIP/ACM International Conference on Distributed Systems Platforms (Middleware 2001) (Nov. 2001), pp. 329-350.

[16] Zhao, B., Kubiatowicz, J., and A. Joseph, "Tapestry: An infrastructure for fault-tolerant wide-area location and routing", Tech. Rep. UCB/CSD-01-1141, Computer Science Division, U. C. Berkeley April 2001.

[17] Rosenberg, J., "Traversal Using Relay NAT (TURN)", draft-rosenberg-midcom-turn-08 (work in progress), September 2005.

[18] Sit, E. and J. Robertson, "Security Considerations for Peer-to-Peer Distributed Hash Tables", First International Workshop on Peer-to-Peer Systems (IPTPS 02) March 2002; Cambridge, MA.

Authors' Addresses

Alan Johnston
SIPStation
St. Louis, MO 63124

Email: alan@sipstation.com

Henry Sinnreich
pulver.com
115 Broadhollow Rd
Suite 225
Melville, NY 11747

Email: henry@pulver.com

Intellectual Property Statement

Disclaimer of Validity

Copyright Statement

Acknowledgment

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