Abstract

This somewhat terse draft discusses issues related to the application of peer to peer (P2P) technologies to SIP in particular, and Internet communications in general. While early work involving P2P and SIP proposes running P2P protocols over SIP messaging, this draft proposes the opposite layering - replacing SIP discovery and rendezvous functionality with a general P2P protocol. This layering of SIP on top of P2P has many advantages. A number of DHT
(Distributed Hash Table) P2P protocols that solve some similar functions are given as examples. Finally, an approach to the discovery of NAT traversal relays using a logically separate P2P network is proposed.

Table of Contents

1. Introduction .................................................. 3
2. SIP is P2P ................................................... 3
3. P2P over SIP .................................................. 3
4. SIP over P2P ................................................... 5
5. P2P NAT Traversal ............................................. 6
6. Conclusion .................................................... 7
7. Acknowledgements .............................................. 7
8. Security Considerations ....................................... 7
9. Informative References ....................................... 7
    Author’s Address ............................................. 8
    Intellectual Property and Copyright Statements ............ 9
1. Introduction

P2P technologies have been widely used on the Internet in file sharing and other applications including VoIP, IM, and presence. First, the lack of actual usage of the inherent P2P capabilities in SIP is bemoaned. Next, the draft compares two approaches to incorporating additional P2P functionality with SIP. One is to tunnel P2P messages over the SIP protocol. The other is to layer SIP on top of P2P protocols which provide discovery and rendezvous functions. Finally, an approach to the discovery of NAT traversal relays using a logically separate P2P network is proposed.

2. SIP is P2P

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.

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. P2P over SIP

Some preliminary work [3], [4] has been published on the use of SIP and a class of P2P algorithms known as Distributed Hash Tables (DHTs). 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. However, these SIP
approaches propose utilizing DHT algorithms, but tunneling all P2P messages over SIP.

For example, these approaches propose using SIP REGISTER requests to join, build, and communicate between peer nodes in the P2P overlay network. The rational for doing so is not clearly explained. Unfortunately, this is one area in which the use of SIP as a transport brings only overhead, delay, and poor scalability. (It also brings to mind the much-discredited DTMF over INFO layer violation.) One type of DHT known as Chord [5], and discussed in later sections, utilizes tables of peer nodes known as Finger Tables. A peer node running the Chord algorithm keeps track of at least N peer nodes, where the maximum network size is 2^N. Using the SHA-1 hashing algorithm, a value of 160 is used for N. In comparison, one Chord over SIP approach recommends instead a Finger Table size of only 32 [3]. Discussions with DHT implementers suggests that real DHT networks need the full 160 node hash table, with even more nodes contacted in a typical query for reasonable latency. Clearly maintaining SIP transaction state with 160 or more dialogs is a significant overhead.

Optimizations and caching schemes for DHT algorithms is an area of active research in the P2P community. For the SIP community to try to replicate this work over SIP would require following this work in great detail. However, since many of these optimizations involve communicating with additional nodes in the network beyond the base of 160, it is clear that due to the overhead of SIP tunneling, few of these enhancements could be utilized.

Instead, it makes much more sense for SIP to reuse the DHT protocols themselves, allowing them to optimize their own transport schemes. For example, DHT algorithms have been implemented and simulated using UDP and TCP transport, and with a variety of caching, striping, and congestion control algorithms [7].

In addition, a SIP-only approach limits the reach and scale of a P2P system. Instead, the SIP community should be working with the P2P community to standardize an interface between the discovery and rendezvous function (P2P) and session establishment and presence function (SIP). This discovery function could be utilized by any URI-based communications technology. For example, SIP, H.323, XMPP, and proprietary protocols could all share a single discovery P2P network based on DHT protocols.

As a result, the SIP community should not seek to replicate other work but instead develop this standard interface, reuse general P2P protocols, and concentrate on solving the security and identity issues specific to P2P.
The REGISTER method should not be used to tunnel P2P messages - its semantics are already overloaded enough and clearly client/server in nature. Defining new SIP header fields or URI conventions is also unnecessary. Instead, a simplified discovery/rendezvous paradigm should be used. That is, a peer can publish a device against a name, and that name is searchable in the P2P network returning the set of active devices. In SIP, this would be a routable Contact URI against an Address of Record URI.

Note: The failure to mention GRUU (Globally Routable User Agent URI) URI in this definition is explicit. This is because the GRUU concept is defined in client/server terms and hence is not applicable here. In addition, note that the current GRUU mechanism work actually suggests a permanent client/server communication mode with SIP - clearly not useful or desirable for P2P SIP.

4. SIP over P2P

An alternative architecture is to layer SIP above the P2P layer. SIP would use the P2P layer for discovery and rendezvous. The question of NAT traversal is discussed separately in the following section.

The search key in a general discovery and rendezvous mechanism for Internet communications would be a URI. This URI would be a name, and would require resolution to a particular device (URL) that must be performed in real time.

As an example set of functions that can be provided at the P2P layer (probably by an API call or similar) is:

- Lookup of a key. Returns a set of addresses of peer nodes that store information about the key.
- Retrieval of the data from a key node or nodes.
- Publishing data to key node or nodes.

For example, the Chord [5] distributed hash lookup primitive (http://www.pdos.lcs.mit.edu/chord/) is an active open source research project which provides the first of these functions. With the addition of the other two functions, complete P2P systems can be constructed.

For example, the CFS (Cooperative File System) 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 [9] is a Chord-based approach for providing DNS lookups. While these applications may not be exactly the discovery and rendezvous capabilities needed for Internet communications, they should be closely studied.

Besides DHTs, there are other classes of P2P algorithms including CAN (Content Addressable Network) [10], Pastry [11], and Tapestry [12]. 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.

5. P2P 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 discovery and rendezvous functions should be separate from the NAT traversal functions. A node agreeing to participate in the discovery and rendezvous P2P 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. This is how P2P protocols such as Chord operate today.

The discovery of NAT traversal relay services is a useful P2P functionality in itself, however. Such a P2P network will likely be a hierarchical one, with super-nodes offering TURN-like services to normal nodes, with a NAT friendly protocol used between nodes and super-nodes. Once a node behind a NAT acquires a relay, it may then participate in the discovery and rendezvous P2P network.

Note: The NAT traversal required is not identical to TURN [13] 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 Chord network, for example, will be similar, but not exactly the same as a TURN server.
6. Conclusion

This draft has discussed at a very high level the issues in layering SIP and P2P functions in an Internet communications system. Some Chord-based DHT P2P applications have been discussed as examples of protocols which may meet the discovery and rendezvous requirements of SIP. Finally, some observations were presented on the discovery of NAT traversal relays.

7. 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, and Adam Roach. In addition, thanks to the Chord developers especially Emil Sit for their feedback.

8. 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 [14] should be studied before jumping into SIP specific mechanisms.

9. Informative References


Author’s Address

Alan Johnston
MCI
100 South 4th Street
St. Louis, MO  63102

Email: alan.johnston@mci.com
Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The Internet Society (2005). This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

Acknowledgment

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