Network Working Group
Request for Comments: 5638
Category: Informational

H. Sinnreich, Ed.
Adobe
A. Johnston
E. Shim
Avaya
K. Singh
Columbia University Alumni

September 2009

Simple SIP Usage Scenario for Applications in the Endpoints

Abstract

For Internet-centric usage, the number of SIP-required standards for presence and IM and audio/video communications can be drastically smaller than what has been published by using only the rendezvous and session-initiation capabilities of SIP. The simplification is achieved by avoiding the emulation of telephony and its model of the intelligent network. 'Simple SIP' relies on powerful computing endpoints. Simple SIP desktop applications can be combined with rich Internet applications (RIAs). Significant telephony features may also be implemented in the endpoints.

This approach for SIP reduces the number of SIP standards with which to comply -- from roughly 100 currently, and still growing, to about 11.

References for NAT traversal and for security are also provided.

Status of This Memo

This memo provides information for the Internet community. It does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Table of Contents

1. Introduction

The Session Initiation Protocol (SIP) has become the global standard for real-time multimedia communications over the Internet and in private IP networks, due to its adoption by service providers and in enterprise networks alike. The cost of this success has been a continuing increase in complexity to accommodate the various requirements for such networks. At the same time, the World Wide Web has become the platform for a boundless variety of rich Internet applications (RIAs), both in the browser and on the desktop. For SIP to be useful for RIAs, requirements for legacy voice-service providers that add unnecessary complexity may be avoided by delegating the interworking to telephony gateway endpoints. This usage scenario for SIP requires following the end-to-end principle of the Internet architecture at the application level or, in other words, placing SIP applications in the endpoints.

There are several reasons, from the Web service's perspective, to place most or all SIP applications in the endpoints and just use the client-server (CS) or peer-to-peer (P2P) rendezvous function for SIP:

1. Value proposition: SIP applications in the endpoints can be easily mixed with RIAs and thus enable service providers to offer new services in a scalable and flexible manner. Mixing SIP applications with RIAs also significantly enhances the value of SIP applications. Rich Internet applications support unrestricted user choice as an alternative that is beyond what is traditionally prepackaged as network-based communication service plans.

2. Eliminating the problems associated with distributed SIP applications in various feature servers across the network allows us to greatly simplify SIP. There is also the Internet end-to-end principle, which argues that network intermediaries cannot completely understand the applications and their state in the endpoints.

'Simple SIP' in this document refers the SIP functions necessary to support only the rendezvous and session-setup functions of SIP, voice, video, basic presence, instant messaging, and also security. Simple SIP is focused on providing a basic multimedia, real-time communications "call". This includes presence, instant messaging, voice, and video for point-to-point and various conference applications. One or a very small number of additional servers may also be provided; for example, a voice-mail server may be provided as an auxiliary to make a simple one-to-one call to voice mail if the callee does not answer or to check voice mail.

Once the applications in the endpoints have established basic communications, it is up to them to support available features selected by users. This paper is targeted to such scenarios. In telephony, most of the value to users and service providers alike is added by signaling. By contrast, on the Web, RIAs add most of the value. The integrated use of SIP and RIAs in the endpoints can combine the best of both.

This approach limits the number of SIP standards to roughly 11 that are listed here as the core for simple SIP. At the time of this writing, the Real-Time Applications and Infrastructure (RAI) area of the IETF is focused on a dedicated working group for the core SIP protocol, separate from various SIP applications. We anticipate this emerging work will also be the core of what is termed here as simple SIP and will actually further reduce the number of references that reflect the present core SIP standards.

This memo aims to shield Web application developers from the need to know or understand more than the core SIP protocol. The total number of references has been kept to a minimum and includes other related topics, such as examples for providing telephony services in the endpoints, NAT traversal, and security. The referenced papers are, however, entry points to these knowledge resources. Readers interested in a more detailed list of SIP topics, especially telephony, can follow up the short list here with the extensive list in "A Hitchhikers' Guide to SIP", RFC 5411 [12]. The guide has over 140 references for understanding most, but not all, of the published features of SIP in the IETF and elsewhere. There is also a Web site that automatically tracks the number of SIP-related RFCs [13]. Other standards and commercial organizations have greatly enlarged the published features of SIP as well. We could not actually provide a complete count on everything that has been published as some form of SIP-standard document.

NAT traversal is also a basic requirement for simple SIP. However, given the potential option of using the Host Identity Protocol (HIP) in SIP-enabled endpoints, as shown in Section 4, simple SIP may not require any standards other than those mentioned here. The alternative to HIP is to use SIP-specific protocols for NAT traversal, such as STUN (Simple Traversal of the UDP Protocol through NAT), TURN (Traversal Using Relay NAT), and ICE (Interactive Connectivity Establishment), as discussed in Section 4.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT","SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

2. The Endpoint in the SIP and Web Architectures

SIP has been defined in RFC 3261 for rendezvous and session initiation. The usual example is the trapezoid model for communications between two endpoints placed in two different SIP service-provider domains. SIP is also flexible, since SIP applications beyond the rendezvous function can reside either in the SIP networks in additional feature and media servers or in the endpoints. SIP endpoints are our focus in this memo.

Since SIP has been invented, with much initial similarity between SIP and HTTP, the Web has evolved from a global access mechanism to static documents to a universal platform with rich interaction between the user and client. In most cases, the client is the browser, though recently dedicated Web desktop clients have emerged as well.

The Web provides access to applications as well as to documents. It is beyond the scope of this memo to describe the application and network architectures of the Web. We will note, however, some of the new application and communication forms that have emerged on the Web as a result of a Darwinian evolution [30] rather than as a result of being defined in standards organizations. They are referred to as Rich Internet Applications.

Examples of RIAs include social networks, blogs, wikis, web-based office and collaboration tools, as well as task-related apps for creating to-do lists, tracking time, combining geographic information with various applications (such as tracking exercise paths and recording the metrics), tracking airline flights, combining live video from events with results and comments, etc.

More information can be found at [31] and in the vast collection of books about RIAs.

RIAs have positioned the browser (and associated Web desktop applications) as the dominant platform for a large variety of applications. They are universal application platforms, independent of network location, operating system, processor, or display size.

Behind the better-known Web applications are a wealth of new technologies that can enhance SIP-based communications, for example, the aggregation of data at runtime from several resources on the Internet. A variety of RIA components, such as found on interactive Web pages, can significantly improve the user experience of SIP-based communications. This is in contrast to the fixed interfaces found in most SIP user agents (UA), such as phones and desktop clients.

The Web network and application architecture is very different from SIP service-provider networks at present, but the one point where they both meet is the end-user device of any shape: fixed or mobile.

The desire of SIP service providers to support new services in a scalable and flexible manner is incidentally easier to implement by the loose service coupling on the Web, as it is possible to characterize a service, or actually a mix of several service components (such as in a mash-up), with a URI. This is in contrast to network services registration being done by a central registrar. The Web architecture is also better suited for users to select and configure their applications and interaction mode with the client. The boundless variety of configurations of services and client settings on the Web is in contrast with the prepackaged services and fixed user-agent configurations in present SIP services.

Last but not least, program execution locally on the client is faster if the interaction with servers across the network is minimized.

The motivation behind this memo is the potential of integrating SIP-based multimedia communications with access to RIAs on the Web. To mention a few scenarios: adding SIP- and RTP-based real-time communications to RIAs, integrating (from a user perspective) the SIP location service (not to be confused with geographic location services) with other desktop- and network-based geographic location services, using social networks as part of the contact list, etc.

2.1. The Telephony Gateway as a SIP Endpoint

In order to accomplish interoperability with the installed base of telephone networks of various kinds, integrating SIP communications into RIAs precludes, in our opinion, carrying legacy telephony features over to the Web. Interoperability between the Internet and telephone networks is best left to gateways that look to the Web as special endpoints serving large numbers of users. Plain one-to-one phone calls are already supported by Internet-to-telephony gateways. If added, PSTN (Public Switched Telephone Network) or ISDN telephony features must be exposed to Web users; visual Web display and interaction with the user is preferable to carrying the extremely complex SIP equivalents over into the Internet. On the Internet side of telephony gateways, simple SIP is all that needs to be deployed, in our opinion. Additional telephony features can be just another RIA hosted in the gateway. The market is the best indicator to show if such an effort is worthwhile to be productized.

Overloading simple SIP with telephony features is a non-objective, as detailed in Section 3.

3. Applicability for Simple SIP in the Endpoints

This section aims to clarify the scope of applicability by considering what can be done better in the endpoints, what simple SIP for user agents can and cannot accomplish, and what is out of scope. We will use emergency calls as an example to illustrate these points on applicability. Emergency calls are also a good example for considering if and when SIP-plus-RIA applications could be used as emergency telephony enhancements or even replacements.

3.1. What Simple SIP Can Accomplish

The main goal for SIP applications on the desktop or in the browser is to support the integration of SIP- and RTP-based real-time communications with RIAs. This assumes powerful endpoints, such as PC/laptop, smart mobile phones, or various dedicated devices.

Example of better functionality: emergency calls not limited to a Public Safety Access Point (PSAP), but extended to a medical service taking care of patients or elderly people.

In this example, besides alerting the right medical provider of the emergency, vital body-sign data and video can also be transmitted. In the opposite direction, the caller may get visual and audio information and instructions for instant self-help. In this scenario, there is no need to invoke a PSAP service. A dedicated device for such scenarios may actually have an emergency medical call button, though for telephone calls to a PSAP this is not recommended [14]. Powerful endpoints may also have various means to determine the geographic location of the caller and transmit it to the emergency care provider. In this and other examples, SIP voice may be a component of several other communications means, but not always the central one; some emergency communications and data transfer may actually be performed without voice, such as instances when the "caller" cannot speak for some reason.

3.2. Baseline for Simple SIP

The focus of the memo is to define the baseline for simple SIP: the establishment of a one-to-one real-time multimedia communication session for presence, IM, voice, and video. Adequate security must also be provided; authentication and encryption for the media and for parts of the signaling should be done in a manner consistent with the routing of SIP messages.

3.3. What Simple SIP May or May Not Accomplish

There are border cases where simple SIP may or may not accomplish some necessary legacy function. Example: an emergency call to a PSAP over the Internet may be supported using the SOS URN [15] and the LoST protocol [16] to determine where to route the call. If, however, emergency calls must be routed over the PSTN to a country-specific telephone number, the assistance of a SIP proxy and also of a SIP-PSTN gateway is required to recognize and route the emergency call. Depending on the local jurisdiction, emergency calls from a SIP UA may require other features that are beyond the scope of this memo.

3.4. What Is Out of Scope for Simple SIP

The simple usage of SIP is applicable when avoiding the traditional voice-provider approaches for charging (or monetizing) that aim to provide, manage, and charge for what is referred to as services (not applications); some examples of such approaches to charging are listed here. Simple SIP means to avoid placing any functions in the network other than the rendezvous function of SIP. This includes avoiding:

- support of legacy telephony functions, such as emulating public-telephone-switch services and voice-only private branch exchanges.

- SIP network architectures designed to support telephony-type network models. Examples include long chains of SIP proxies and feature servers (more than the two SIP servers shown in RFC 3261) that may be encountered inside and between closed Voice over IP (VoIP) networks and in-transit VoIP networks in between. Long chains of intermediaries of any type not only add complexity, they pose a security risk that increases with the number of SIP network elements. Complex server-based networks also make it more difficult to introduce new services. A special problem in SIP server chains is forking, which leads to the well-known problems of concurrency in computing; the so-called race conditions in telephony. This is amplified by redesigning the whole network every time there is a new SIP routing requirement.

- support for legacy telephony models, such as identifying end-user devices for the purpose of differentiated charging by type of service or for charging for roaming between networks.

- policies and the associated policy servers and network elements for Quality of Service (QoS) to enforce service-rate-specific policies for real-time communications.

- o design considerations for SIP for compatibility with legacy telephony networks, traditional telephony services, and various telephone numbering plans. This pushes the responsibility of mapping the URI to telephone numbers to edge networks where the IP-PSTN gateway functions are performed. The handling of telephony-specific functions, such as early media, are also pushed to edge gateway networks. Other design considerations for interworking with the PSTN and 'looking like the PSTN' are also avoided.

This list is not exhaustive, but conveys the concept of what to avoid when using SIP as a simpler protocol to understand and to implement.

3.5. Borderline Cases

There are also some interesting borderline cases for what to avoid, such as Provisional Response Acknowledgements (PRACKs), specified in RFC 3262. PRACK is targeted for multi-hop SIP server networks and PSTN interworking, especially to assure reliable early media. PRACK can be delegated, albeit with some limitations to the SIP-PSTN gateway. PRACK does little to improve the user experience and has no relevance on true broadband networks with minimal SIP hop counts. Using PRACK may therefore be a decision best left to designers.

Another interesting example of a borderline case are the issues with SIP's Non-Invite transactions as discussed in RFC 4320 [17]. Long chains of SIP intermediaries complicate the handling of provisional responses and may create several problems, such as storms of late responses from forked SIP forwarding paths. We mentioned that long chains of SIP intermediaries are out of scope for simple SIP, but since designers may encounter various scenarios, even those they don't like, the decision to conform the user agent (UA) to RFC 4320 is best left to them.

The list of borderline cases is also not exhaustive and the above are only examples. So where is the borderline? We believe that SIP usage on the Internet, without any intermediaries designed to support closed VoIP networks, eliminates the borderline cases. Enterprise SIP networks are also most useful when designed to work with the Internet model in mind, by giving enterprise users the benefit of SIP-enhanced Web applications for productivity. Handling of SIP in enterprise firewalls is out of the scope of this memo.

4. Mandatory SIP References for Internet-Centric Usage

Here is the minimal set of mandatory references to support the Internet-centric approach to SIP, outlined above. The minimal set of references defines simple SIP.

The proposed change process [29] for SIP in the IETF RAI area will define the updated SIP core specification and thus reduce even more the required SIP standards for what is referred to here as simple SIP.

4.1. RFC 3261: "SIP: Session Initiation Protocol"

RFC 3261 [1] is the core specification for SIP. The trapezoid model for SIP, found in RFC 3261, is only an example and a use case applicable to two service providers featuring an outgoing SIP proxy and an incoming SIP proxy in each domain respectively. However, SIP can also work in peer-to-peer (P2P) communications without SIP servers.

4.2. RFC 4566: "SDP: Session Description Protocol"

SDP [2] is the standard format for the representation of media parameters, transport addresses, and other session data irrespective of the protocol used to transport the SDP data. SIP is one of the protocols used to transport SDP data, to enable the setting up of multimedia communication sessions. Other Internet application protocols use SDP as well.

4.3. RFC 3264: "An Offer/Answer Model with Session Description Protocol (SDP)"

Though SDP has the capability to describe SIP sessions, how to arrive at a common description by two SIP endpoints requires a negotiation procedure to agree on common media codecs, along with IP addresses and ports where the media can be received. This negotiation procedure is specified in RFC 3264 [3]. As will be seen in Section 6, this negotiation is usually considerably complicated due to the existence of NAT between the SIP endpoints.

4.4. RFC 3840: "Indicating User Agent Capabilities in the Session Initiation Protocol (SIP)"

A SIP UA can convey its capability in the Contact header field, indicating if it can support presence, IM, audio, or video, and if the device is fixed, mobile, or other, such as the endpoint being an automaton (voice mail for example). Which SIP methods are supported may also be indicated as specified in RFC 3840 [4]. SIP registrars (SIP servers or the P2P SIP overlay) can be informed of endpoint capabilities. Missing capabilities can be displayed for the user by, for example, grayed out or missing icons.

4.5. RFC 3263: "Session Initiation Protocol (SIP): Locating SIP Servers"

RFC 3263 [5] adds key clarifications to the base SIP specification in RFC 3261 by specifying how a SIP user agent (UA) or SIP server can determine with DNS queries not only the IP addresses of the target SIP servers, but also which SIP servers can support UDP or TCP transport, as required. TCP may be required to support secure SIP (SIPS) using Transport Layer Security (TLS) transport or when SIP messages are too large to fit into UDP packets without fragmentation. Successive DNS queries yield finer-grain location by providing NAPTR, SRV, and A type records. Note that finding a SIP server requires several successive DNS queries to access these records.

Locating SIP servers is also required for P2P SIP when a peer node wishes to communicate with a SIP UA outside its own P2P SIP overlay network.

4.6. RFC 3265: "Session Initiation Protocol (SIP)-Specific Event Notification"

RFC 3265 [6] provides an extensible framework by which SIP nodes can request notification from remote nodes indicating that certain events have occurred. The most prominent event notifications are those used for presence, though SIP events are used for many other SIP services, some of which can be useful for simple SIP.

4.7. RFC 3856: "A Presence Event Package for the Session Initiation Protocol (SIP)"

RFC 3856 [7] defines the usage of SIP as a presence protocol and makes use of the SUBSCRIBE and NOTIFY methods for presence events. SIP location services already contain presence information in the form of registrations and, as such, can be reused to establish connectivity for subscriptions and notifications. This can enable either endpoints or servers to support rich applications based on presence.

4.8. RFC 3863: "Presence Information Data Format (PIDF)"

RFC 3863 [8] defines the Presence Information Data Format (PIDF) and the media type "application/pidf+xml" to represent the XML MIME entity for PIDF. PIDF is used by SIP to carry presence information.

4.9. RFC 3428: "Session Initiation Protocol (SIP) Extension for Instant Messaging"

The SIP extension for IM in RFC 3428 [9] consists in the MESSAGE method (defined in RFC 3428) only for the pager model of IM, based on the assumption that an IM conversation state exists in the client interface in the endpoints or in the mind of the users.

4.10. RFC 4474: "Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP)"

RFC 4474 [10] defines (1) an identity header and (2) an identity info header for SIP requests that carry, respectively, the signature of the issuer over parts of the SIP request and the signed identity information. The signature includes the FROM header and the identity of the sender. The associated identity info header identifies the sender of the SIP request, such as INVITE. The issuer of the signature can present their certificate as well. It is assumed the issuer may be the domain owner. Strong authentication is thus provided for SIP requests. Authentication for SIP responses is not defined in this document.

4.11. RFC 3581: "An Extension to the Session Initiation Protocol (SIP) for Symmetric Response Routing"

RFC 3581 [11] specifies an extension to SIP called "rport" so that responses are sent back to the source IP address and port from which the request originated. This correction to RFC 3261 is helpful for NAT traversal, debugging, and support of multi-homed hosts.

4.12. Updates to SIP-Related Protocols

Several of the above are being updated to benefit from the experience of large deployments and frequent interoperability testing. We recommend readers to constantly check for revisions. One update example is "Correct Transaction Handling for 200 Responses to the Session Initiation Protocol INVITE Requests" [18]. This is an update to RFC 3261; the added security risk for misbehaving SIP UAs is handled in the forwarding SIP proxy.

5. SIP Applications in the Endpoints

Although the present adoption of SIP is mainly due to telephony applications, its roots are in the Web and it has initial similarity to HTTP. As a result, SIP may play other roles in adequately powerful endpoints (their number keeps increasing with Moore’s law). SIP-based multimedia communications may be linked with various other applications on the Web. Either some non-SIP application or the

communication feature may be perceived as the primary usage. An example is mixing SIP-based real-time communications with some Web content of high interest to the user.

Examples:

1. In a conversation between a consumer and the contact center, a Web conference can be invoked to present to the user buying options or help information. This information can make use of mashups to combine real-time data from various sources on the Web.

2. In a social network, multimedia conversations combined with Web mashups can be invoked, thus strengthening the bond between its members.

3. Conversations can be invoked while watching some events on the Web in real time. However, the main beneficiary in this case may be the Web site, since the conversation can prolong the time for users watching that Web site.

This shows the value of combining RIAs with SIP-based communications.

It is a matter for the end user's judgment whether the Web content or the associated communication capability is more important, or if a mix of both is most attractive.

Example: a Web-based enterprise directory where employees can find a wealth of data. Adding SIP multimedia communications to the enterprise directory to call someone (if online and not too busy) enhances its usefulness, but is not critical to the directory.

SIP applications in the endpoints can, however, accomplish most telephony functions as well. This has been amply documented in SIP-related work in the IETF, such as:

- "A Call Control and Multi-party usage framework for SIP" [19] presents a large assortment of telephony applications where the call control resides in the participating endpoints that use the peer-to-peer feature invocation model. The peer-to-peer design and its principles are based on multiparty call control.

- "Session Initiation Protocol Service Examples" [20] contains a collection of SIP call flows for traditional telephony, many of which require no server support for the respective features. The SIP service examples for telephony are extremely useful since they illustrate in detail the concepts and applications supported by the core simple SIP references.

In conclusion, SIP applications in the endpoints can support both a mix of real-time communications with new rich Internet applications and traditional telephony features as well.

6. NAT Traversal

SIP devices behind one or more NATs are, at present, the rule rather than the exception.

"Best Current Practices for NAT Traversal for SIP" [22] comprehensively summarizes the use of STUN, TURN, and ICE, and provides a definitive set of 'Best Common Practices' to demonstrate the traversal of SIP and its associated RTP media packets through NAT devices.

The use of ICE has been developed mainly for SIP. Other proposals, such as NICE (generic for non-SIP) and "D-ICE" for Real Time Streaming Protocol (RTSP) streaming media, have also been proposed. Internet games have different NAT traversal techniques of their own. This list is not exhaustive and such approaches are based on different NAT traversal protocols for each application protocol, separately.

A general, non-application-protocol-specific approach for NAT traversal is therefore highly desirable.

One approach for NAT traversal that is generic and applicable for all application protocols is to deploy the Host Identity Protocol (HIP) and solve NAT traversal only once, at the HIP level. HIP has many other useful features (such as support for the IPv6 transition in endpoints, mobility, and multihoming) that are beyond the scope of this paper. "Basic HIP Extensions for Traversal of Network Address Translators" [23] provides an extensive coverage of the use of HIP for NAT traversal.

Using HIP-enabled endpoints can provide the functions required for NAT traversal [24] for all applications, for both IPv4 and IPv6. HIP can thus simplify the SIP UA since it takes away the burden of NAT traversal from the SIP UA and moves it to the HIP protocol module in the endpoint.

7. Security Considerations

All protocols discussed in this paper have their own specific security requirements that MUST be considered. The special security considerations for SIP signaling security and RTP media security are discussed here.

SIP security has two main parts: transport security and identity.

- Transport security for SIP is specified in RFC 3261. Secure SIP has the notation SIPS in the request URI and uses TLS over TCP. Note that SIP over UDP cannot be secured in this way. Transport security works only hop by hop. Specifying SIPS requires the user to trust all intermediate servers and no end-to-end media encryption is assumed. There is no insurance for misbehaving intermediaries in the path. SIPS is therefore really adequate only in single-hop scenarios.

- RFC 4474, "Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP)", which is mentioned previously, specifies the use of certificates for secure identification of the parties involved in SIP signaling requests.

- The Datagram Transport Layer Security (DTLS) specified in RFC 4347 [25] has wide applicability for other applications that require UDP transport. DTLS has been designed to have maximum commonality with TLS, yet does not require TCP transport and works over UDP. The DTLS-SRTP (Secure Realtime Transport Protocol) Framework [26] can support encrypted communications between endpoints using self-signed certificates whose fingerprints are exchanged over an integrity-protected SIP signaling channel. The SRTP master secret is derived using the DTLS exchange as described in [27].

- ZRTP [28] provides key agreement for SRTP for multimedia communication with voice without depending on SIP signaling, though it can utilize an integrity-protected SIP signaling path for authentication. ZRTP does not require the use of certificates or any Public Key Infrastructure (PKI). ZRTP provides best-effort SRTP encryption without any additional SIP extensions.

8. Acknowledgements

The authors would like to thank Cullen Jennings, Ralph Droms, and Adrian Farrel for helpful comments in the most recent stage of this memo.

Special thanks are due to Paul Kyzivat for challenging the authors to clarify the role of telephony network gateways and also to Keith Drage for challenging them to discuss the use of emergency calls using simple SIP.

Robert Sparks has pointed to some missing references, which we have added.

The authors would also like to thank Jiri Kuthan, Adrian Georgescu, and others for the detailed discussion on the SIPPING WG list. As a result, we have added clarification of what simple SIP can do, what it does not aim to do, and some scenarios in between. We would also like to thank Wilhelm Wimmreuter for the detailed review of the initial draft and to Arjun Roychaudhury for the comments regarding the need to clarify the difference between network-based services and endpoint applications.

9. References

9.1. Normative 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] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, July 2006.

[3] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, June 2002.

[4] Rosenberg, J., Schulzrinne, H., and P. Kyzivat, "Indicating User Agent Capabilities in the Session Initiation Protocol (SIP)", RFC 3840, August 2004.

[5] Rosenberg, J. and H. Schulzrinne, "Session Initiation Protocol (SIP): Locating SIP Servers", RFC 3263, June 2002.

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

[7] Rosenberg, J., "A Presence Event Package for the Session Initiation Protocol (SIP)", RFC 3856, August 2004.

[8] Sugano, H., Fujimoto, S., Klyne, G., Bateman, A., Carr, W., and J. Peterson, "Presence Information Data Format (PIDF)", RFC 3863, August 2004.

[9] Campbell, B., Ed., Rosenberg, J., Schulzrinne, H., Huitema, C., and D. Gurle, "Session Initiation Protocol (SIP) Extension for Instant Messaging", RFC 3428, December 2002.

[10] Peterson, J. and C. Jennings, "Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP)", RFC 4474, August 2006.

[11] Rosenberg, J. and H. Schulzrinne, "An Extension to the Session Initiation Protocol (SIP) for Symmetric Response Routing", RFC 3581, August 2003.

9.2. Informative References

[12] Rosenberg, J., "A Hitchhiker's Guide to the Session Initiation Protocol (SIP)", RFC 5411, February 2009.

[13] Ohlmeier, N., "VoIP RFC Watch", http://rfc3261.net/.

[14] Rosen, B. and J. Polk, "Best Current Practice for Communications Services in support of Emergency Calling", Work in Progress, July 2009.

[15] Schulzrinne, H., "A Uniform Resource Name (URN) for Emergency and Other Well-Known Services", RFC 5031, January 2008.

[16] Hardie, T., Newton, A., Schulzrinne, H., and H. Tschofenig, "LoST: A Location-to-Service Translation Protocol", RFC 5222, August 2008.

[17] Sparks, R., "Actions Addressing Identified Issues with the Session Initiation Protocol's (SIP) Non-INVITE Transaction", RFC 4320, January 2006.

[18] Sparks, R. and T. Zourzouvillys, "Correct Transaction Handling for 200 Responses to Session Initiation Protocol INVITE Requests", Work in Progress, July 2009.

[19] Mahy, R., Sparks, R., Rosenberg, J., Petrie, D., and A. Johnson, "A Call Control and Multi-party usage framework for the Session Initiation Protocol (SIP)", Work in Progress, March 2009.

[20] Johnston, A., Ed., Sparks, R., Cunningham, C., Donovan, S., and K. Summers, "Session Initiation Protocol Service Examples", BCP 144, RFC 5359, October 2008.

[22] Boulton, C., Rosenberg, J., Camarillo, G. and F. Audet, "Best Current Practices for NAT Traversal for Client-Server SIP", Work in Progress, September 2008.

[23] Komu, M., Henderson, T., Tschofenig, H., Melen, J. and A. Keraenen, "Basic HIP Extensions for Traversal of Network Address Translators", Work in Progress, June 2009.

[24] Moskowitz, R., "HIP Experimentation using Teredo", July 2008, http://www.ietf.org/proceedings/08jul/slides/HIPRG-3.pdf.

[25] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security", RFC 4347, April 2006.

[26] Fischl, J., Tschofenig, H. and E. Rescorla, "Framework for Establishing an SRTP Security Context using DTLS", Work in Progress, March 2009.

[27] McGrew, D. and E. Rescorla, "Datagram Transport Layer Security (DTLS) Extension to Establish Keys for Secure Real-time Transport Protocol (SRTP)", Work in Progress, February 2009.

[28] Zimmerman, P., Johnston, A. and J. Callas, "ZRTP: Media Path Key Agreement for Secure RTP", Work in Progress, March 2009

[29] Peterson, J., Jennings, C. and R. Sparks, "Change Process for the Session Initiation Protocol (SIP)", Work in Progress, July 2009.

[30] Raman, T.V., "Toward 2 exp(W), Beyond Web 2.0", Communications of the ACM, Vol. 52, No.2, p. 52-59, February 2009.

[31] Wikipedia, "Rich Internet application", http://en.wikipedia.org/wiki/Rich_Internet_Applications.

Authors' Addresses

Henry Sinnreich
Adobe Systems, Inc.
601 Townsend Street,
San Francisco, CA 94103, USA

EMail: henrys@adobe.com

Alan Johnston
Avaya
Saint Louis, MO, USA

EMail: alan@sipstation.com

Eunsoo Shim
Avaya Labs Research
233 Mount Airy Road
Basking Ridge, NJ 07920 USA

EMail: eunsooshim@gmail.com

Kundan Singh
Columbia University Alumni
1214 Amsterdam Ave., MC0401
New York, NY, USA

EMail: kns10@cs.columbia.edu