P2PSIP C. Jennings
Internet-Draft Cisco
Intended status: Standards Track B. Lowekamp
Expires: January 30, 2014 Skype
E. Rescorla
RTFM, Inc.
S. Baset
H. Schulzrinne
Columbia University
T C. Schmidt, Ed.
HAW Hamburg
July 29, 2013

A SIP Usage for RELOAD
draft-ietf-p2psip-sip-11

Abstract

This document defines a SIP Usage for REsource LOcation And Discovery (RELOAD). The SIP Usage provides the functionality of a SIP proxy or registrar in a fully-distributed system and includes a lookup service for Address of Records (AORs) stored in the overlay. It also defines Globally Routable User Agent Uris (GRUUs) that allow the registrations to map an AOR to a specific node reachable through the overlay. After such initial contact of a peer, the AppAttach method is used to establish a direct connection between nodes through which SIP messages are exchanged.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

The REsource LOcation And Discovery (RELOAD) [I-D.ietf-p2psip-base] specifies a peer-to-peer (P2P) signaling protocol for the general use on the Internet. This document defines a SIP Usage of RELOAD that allows SIP [RFC3261] user agents (UAs) to establish peer-to-peer SIP (or SIPS) sessions without the requirement for permanent proxy or registration servers, e.g., a fully distributed telephony service. In such a network, the RELOAD overlay itself performs the registration and rendezvous functions ordinarily associated with such servers.

The SIP Usage involves two basic functions.

Registration: SIP UAs can use the RELOAD data storage functionality to store a mapping from their address-of-record (AOR) to their Node-ID in the overlay, and to retrieve the Node-ID of other UAs.

Rendezvous: Once a SIP UA has identified the Node-ID for an AOR it wishes to call, it can use the RELOAD message routing system to set up a direct connection for exchanging SIP messages.

Mappings are stored in the SipRegistration Resource Record defined in this document. All operations required to perform a SIP registration or rendezvous are standard RELOAD protocol methods.

For example, Bob registers his AOR, "bob@dht.example.com", for his Node-ID "1234". When Alice wants to call Bob, she queries the overlay for "bob@dht.example.com" and receives Node-ID 1234 in return. She then uses the overlay routing to establish a direct connection with Bob and can directly transmit a standard SIP INVITE. In detail, this works along the following steps.

1. Bob, operating Node-ID 1234, stores a mapping from his AOR to his Node-ID in the overlay by applying a Store request for "bob@dht.example.com -> 1234".
2. Alice, operating Node-ID 5678, decides to call Bob. She retrieves Node-ID "1234" by performing a Fetch request on "bob@dht.example.com".
3. Alice uses the overlay to route an AppAttach message to Bob's peer (ID 1234). Bob responds with his own AppAttach and they set up a direct connection, as shown in Figure 1. Note that mutual ICE checks are invoked automatically from AppAttach message exchange.


```
                          Overlay
    Alice       Peer1     ...          PeerN       Bob
    (5678)                                        (1234)
    -------------------------------------------------
    AppAttach ->
                AppAttach ->
                          AppAttach ->
                                      AppAttach ->
                                         <- AppAttach
                                <- AppAttach
                     <- AppAttach
         <- AppAttach

    <------------------ ICE Checks ----------------->
    INVITE ----------------------------------------->
    <-------------------------------------------- OK
    ACK -------------------------------------------->
    <------------ ICE Checks for media ------------->
    <-------------------- RTP ---------------------->
```

Figure 1: Connection setup in P2P SIP using the RELOAD overlay

It is important to note that here the only role of RELOAD is to set up the direct SIP connection between Alice and Bob. As soon as the ICE checks complete and the connection is established, ordinary SIP or SIPS is used. In particular, the establishment of the media channel for a phone call happens via the usual SIP mechanisms, and RELOAD is not involved. Media never traverses the overlay. After the successful exchange of SIP messages, call peers run ICE connectivity checks for media.

In addition to mappings from AORs to Node-IDs, the SIP Usage also allows mappings from AORs to other AORs. This enables an indirection useful for call forwarding. For instance, if Bob wants his phone calls temporarily forwarded to Charlie, he can store the mapping "bob@dht.example.com -> charlie@dht.example.com". When Alice wants to call Bob, she retrieves this mapping and can then fetch Charlie's AOR to retrieve his Node-ID. These mechanisms are described in Section 3.

Alternatively, Globally Routable User Agent URIs (GRUUs) can be used for directly accessing peers. They are handled via a separate mechanism, as described in Section 6.

The SIP Usage for RELOAD addresses a fully distributed deployment of session-based services among overlay peers. Two opposite scenarios of deploying P2P SIP services are in the focus of this document: A

highly regulated environment of a "single provider" that admits parties using AORs with domains from controlled namespace(s), only, and an open, multi-party infrastructure that liberally allows a registration and rendezvous for various or any domain namespace. It is noteworthy in this context that - in contrast to regular SIP - domain names play no role in routing to a proxy server. Once connectivity to an overlay is given, any name registration can be technically processed.

2. Terminology

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 [RFC2119].

We use the terminology and definitions from Concepts and Terminology for Peer to Peer SIP [I-D.ietf-p2psip-concepts] and the RELOAD Base Protocol [I-D.ietf-p2psip-base] extensively in this document.

In addition, term definitions from SIP [RFC3261] apply to this memo. The term AOR is the SIP "Address of Record" used to identify a user in SIP. For example, alice@example.com could be the AOR for Alice. For the purposes of this specification, an AOR is considered not to include the scheme (e.g sip:) as the AOR needs to match the rfc822Name in the X509v3 certificates. It is worth noting that SIP and SIPS are distinguished in P2PSIP by the Application-ID.

3. Registering AORs in the Overlay

3.1. Overview

In ordinary SIP, a UA registers its AOR and location with a registrar. In RELOAD, this registrar function is provided by the overlay as a whole. To register its location, a RELOAD peer stores a SipRegistration Resource Record under its own AOR using the SIP-REGISTRATION Kind, which is formally defined in Section 7. A RELOAD overlay MAY restrict the storage of AORs. Namespaces (i.e., the right hand side of the AOR) that are supported for registration and lookup can be configured for each RELOAD deployment as described in Section 3.4.

As a simple example, consider Alice with AOR "alice@dht.example.org" at Node-ID "1234". She might store the mapping "alice@dht.example.org -> 1234" telling anyone who wants to call her to contact node "1234".

RELOAD peers MAY store two kinds of SIP mappings,

- o from an AOR to a destination list (a single Node-ID is just a trivial destination list), or
- o from an AOR to another AOR.

The meaning of the first kind of mapping is "in order to contact me, form a connection with this peer." The meaning of the second kind of mapping is "in order to contact me, dereference this AOR". The latter allows for forwarding. For instance, if Alice wants her calls to be forwarded to her secretary, Sam, she might insert the following mapping "alice@dht.example.org -> sam@dht.example.org".

3.2. Data Structure

This section defines the SipRegistration Resource Record as follows:

```
      enum { sip_registration_uri(1), sip_registration_route(2),
         (255) } SipRegistrationType;

      select (SipRegistration.type) {
        case sip_registration_uri:
          opaque                 uri<0..2^16-1>;

        case sip_registration_route:
          opaque                 contact_prefs<0..2^16-1>;
          Destination            destination_list<0..2^16-1>;

        /* This type can be extended */

      } SipRegistrationData;

      struct {
         SipRegistrationType   type;
         uint16                length;
         SipRegistrationData   data;
     } SipRegistration;
```

The contents of the SipRegistration Resource Record are:

type

the type of the registration

length
the length of the rest of the PDU

data
the registration data

- o If the registration is of type "sip_registration_uri", then the contents are an opaque string containing the URI.
- o If the registration is of type "sip_registration_route", then the contents are an opaque string containing the callee's contact preferences and a destination list for the peer.

The encoding of contact_prefs - the callee's contact preferences - follows the media feature set syntax of [RFC2533] (see also [RFC2738]). As an example, a voicemail server that is a UA that supports audio and video media types and is not mobile would carry the following feature set description in its contact_prefs attribute:

```
(& (sip.audio=TRUE)
      (sip.video=TRUE)
      (sip.actor=msg-taker)
      (sip.automata=TRUE)
      (sip.mobility=fixed)
      (| (sip.methods=INVITE) (sip.methods=BYE) (sip.methods=OPTIONS)
         (sip.methods=ACK) (sip.methods=CANCEL)))
```

A callee MAY indicate that it prefers contact via a particular SIP scheme - SIP or SIPS - by using one of the following contact_prefs attribute:

```
      (sip.schemes=SIP)
      (sip.schemes=SIPS)
```

RELOAD explicitly supports multiple registrations for a single AOR. The registrations are stored in a Dictionary with Node-IDs as the dictionary keys. Consider, for instance, the case where Alice has two peers:

- o her desk phone (1234)
- o her cell phone (5678)

Alice might store the following in the overlay at resource "alice@dht.example.com".

- o A SipRegistration of type "sip_registration_route" with dictionary key "1234" and value "1234".
- o A SipRegistration of type "sip_registration_route" with dictionary key "5678" and value "5678".

Note that this structure explicitly allows one Node-ID to forward to another Node-ID. For instance, Alice could set calls to her desk phone to ring at her cell phone by storing a SipRegistration of type "sip_registration_route" with dictionary key "1234" and value "5678".

3.3. Access Control

In order to prevent hijacking or other misuse, registrations are subject to access control rules. Two kinds of restrictions apply:

- o A Store is permitted only for AORs with domain names that fall into the namespaces supported by the RELOAD overlay instance.
- o Storing requests are performed according to the USER-NODE-MATCH access control policy of RELOAD.

Before issuing a Store request to the overlay, any peer SHOULD verify that the AOR of the request is a valid Resource Name with respect to its domain name and the namespaces defined in the overlay configuration document (see Section 3.4).

Before a Store is permitted, the storing peer MUST check that:

- o The AOR of the request is a valid Resource Name with respect to the namespaces defined in the overlay configuration document.
- o The certificate contains a username that is a SIP AOR which hashes to the Resource-ID it is being stored at.
- o The certificate contains a Node-ID that is the same as the dictionary key it is being stored at.

Note that these rules permit Alice to forward calls to Bob without his permission. However, they do not permit Alice to forward Bob's calls to her. See Section 8.2.2 for additional descriptions.

3.4. Overlay Configuration Document Extension

The use of a SIP-enabled overlay MAY be restricted to users with AORs from specific domains. When deploying an overlay service, providers can decide about these use case scenarios by defining a set of namespaces for admissible domain names. This section extends the overlay configuration document by defining new elements for patterns that describe a corresponding domain name syntax.

A RELOAD overlay can be configured to accept store requests for any

AOR, or to apply domain name restrictions. For the latter, an enumeration of admissible domain names including wildcarded name patterns of the following form MAY be configured.

```
Example of Domain Patterns:
dht\.example\.com
.*\.my\.name
```

In this example, any AOR will be accepted that is either of the form <user>@dht.example.com, or ends with the domain "my.name". When restrictions apply and in the absence of domain patterns, the default behavior is to accept only AORs that exactly match the domain name of the overlay. Otherwise, i.e., when restrictions are not configured (attribute enable not set), the default behavior is to accept any AOR. In the absence of a <domain-restrictions> element, implementors SHOULD assume this default value. Encoding of the domain name complies to the restricted ASCII character set without character escaping as defined in Section 19.1 of [RFC3261].

The <domain-restrictions> element serves as a container for zero to multiple <pattern> sub-elements. A <pattern> element MAY be present if the "enable" attribute of its parent element is set to true. Each <pattern> element defines a pattern for constructing admissible resource names. It is of type xsd:string and interpreted as a regular expression according to "POSIX Extended Regular Expression" (see the specifications in [IEEE-Posix]).

The Relax NG Grammar for the AOR Domain Restriction reads:

```
<!-- AOR DOMAIN RESTRICTION URN SUB-NAMESPACE -->

namespace sip = "urn:ietf:params:xml:ns:p2p:config-base:sip"

<!--  AOR DOMAIN RESTRICTION ELEMENT -->

Kind-parameter &= element sip:domain-restriction {

    attribute enable { xsd:boolean }
```

```
     <!-- PATTERN ELEMENT -->

     element pattern { xsd:string }*
   }?
```


4. Looking up an AOR

4.1. Finding a Route to an AOR

A RELOAD user, member of an overlay, who wishes to call another user with given AOR SHALL proceed in the following way.

AOR is GRUU? If the AOR is a GRUU for this overlay, the callee can be contacted directly as described in Section 6.

AOR domain is hosted in overlay? If the domain part of the AOR matches a domain pattern configured in the overlay, the user can continue to resolve the AOR in this overlay. The user MAY choose to query the DNS service records to search for additional support of this domain name.

AOR domain not supported by overlay? If the domain part of the AOR is not supported in the current overlay, the user SHOULD query the DNS (or other discovery services at hand) to search for an alternative overlay that services the AOR under request. Alternatively, standard SIP procedures for contacting the callee SHOULD be used.

AOR inaccessible? If all of the above contact attempts fail, the call fails.

The procedures described above likewise apply when nodes are simultaneously connected to several overlays.

4.2. Resolving an AOR

A RELOAD user that has discovered a route to an AOR in the current overlay SHALL execute the following steps.

1. Perform a Fetch for Kind SIP-REGISTRATION at the Resource-ID

corresponding to the AOR. This Fetch SHOULD NOT indicate any dictionary keys, so that it will fetch all the stored values.

2. If any of the results of the Fetch are non-GRUU AORs, then repeat step 1 for that AOR.
3. Once only GRUUs and destination lists remain, the peer removes duplicate destination lists and GRUUs from the list and initiates SIP or SIPS connections to the appropriate peers as described in the following sections. If there are also external AORs, the peer follows the appropriate procedure for contacting them as well.

5. Forming a Direct Connection

5.1. Setting Up a Connection

Once the peer has translated the AOR into a set of destination lists, it then uses the overlay to route AppAttach messages to each of those peers. The "application" field MUST be either 5060 to indicate SIP or 5061 for using SIPS. If certificate-based authentication is in use, the responding peer MUST present a certificate with a Node-ID matching the terminal entry in the route list. Note that it is possible that the peers already have a RELOAD connection mutually established. This MUST NOT be used for SIP messages unless it is a SIP connection. A previously established SIP connection MAY be used for a new call.

Once the AppAttach succeeds, the peer sends plain or (D)TLS encrypted SIP messages over the connection as in normal SIP. A caller MAY choose to contact the callee using SIP or secure SIPS, but SHOULD follow a preference indicated by the callee in its contact_prefs attribute (see Section 3.2). A callee MAY choose to listen on both SIP and SIPS ports and accept calls from either SIP scheme, or select a single one. However, a callee that decides to accept SIPS calls, only, SHOULD indicate its choice by setting the corresponding attribute in its contact_prefs.

5.2. Keeping a Connection Alive

In many cases, RELOAD connections will traverse NATs and Firewalls that maintain states established from ICE [RFC5245] negotiations. It is the responsiblity of the Peers to provide sufficiently frequent traffic to keep NAT and Firewall states present and the connection alive. Keepalives are a mandatory component of ICE (see Section 10 of [RFC5245]) and no further operations are required. Applications that want to assure maintanance of sessions individually need to follow regular SIP means. Accordingly, a SIP Peer MAY apply keep-alive techniques in agreement with its transport binding as defined in Section 3.5 of [RFC5626].

6. Using GRUUs

Globally Routable User Agent Uris (GRUUs) [RFC5627] have been designed to allow direct routing without the indirection of a SIP proxy function. The concept is transferred to RELOAD overlays as follows. GRUUs in RELOAD are constructed by embedding a base64-encoded destination list in the gr URI parameter of the GRUU. The base64 encoding is done with the alphabet specified in table 1 of [RFC4648] with the exception that ~ is used in place of =.
Example of a RELOAD GRUU:
alice@example.com;gr=MDEyMzQ1Njc4OTAxMjM0NTY3ODk~

GRUUs do not require to store data in the Overlay Instance. Rather when a peer needs to route a message to a GRUU in the same P2P overlay, it simply uses the destination list and connects to that peer. Because a GRUU contains a destination list, it MAY have the same contents as a destination list stored elsewhere in the resource dictionary.

Anonymous GRUUs [RFC5767] are constructed analogously, but require either that the enrollment server issues a different Node-ID for each anonymous GRUU required, or that a destination list be used that includes a peer that compresses the destination list to stop the Node-ID from being revealed.

7. SIP-REGISTRATION Kind Definition

This section defines the SIP-REGISTRATION Kind.

Name SIP-REGISTRATION

Kind IDs The Resource Name for the SIP-REGISTRATION Kind-ID is the AOR of the user. The data stored is a SipRegistration, which can contain either another URI or a destination list to the peer which is acting for the user.

Data Model The data model for the SIP-REGISTRATION Kind-ID is dictionary. The dictionary key is the Node-ID of the storing peer. This allows each peer (presumably corresponding to a single device) to store a single route mapping.

Access Control USER-NODE-MATCH. Note that this matches the SIP AOR against the rfc822Name in the X509v3 certificate. The rfc822Name does not include the scheme so that the "sip:" prefix needs to be removed from the SIP AOR before matching.

Data stored under the SIP-REGISTRATION Kind is of type SipRegistration. This comes in two varieties:

sip_registration_uri
a URI which the user can be reached at.

sip_registration_route
a destination list which can be used to reach the user's peer.

8. Security Considerations

8.1. RELOAD-Specific Issues

This Usage for RELOAD does not define new protocol elements or operations. Hence no new threats arrive from message exchanges in RELOAD.

This document introduces an AOR domain restriction function that must be surveyed by the storing peer. A misconfigured or malicious peer could cause frequent rejects of illegitimate storing requests.

However, domain name control relies on a lightweight pattern matching and can be processed prior to validating certificates. Hence no extra burden is introduced for RELOAD peers beyond loads already present in the base protocol.

8.2. SIP-Specific Issues

8.2.1. Fork Explosion

Because SIP includes a forking capability (the ability to retarget to multiple recipients), fork bombs are a potential DoS concern. However, in the SIP usage of RELOAD, fork bombs are a much lower concern than in a conventional SIP Proxy infrastructure, because the calling party is involved in each retargeting event. It can therefore directly measure the number of forks and throttle at some reasonable number.

8.2.2. Malicious Retargeting

Another potential DoS attack is for the owner of an attractive AOR to retarget all calls to some victim. This attack is common to SIP and difficult to ameliorate without requiring the target of a SIP registration to authorize all stores. The overhead of that requirement would be excessive and in addition there are good use cases for retargeting to a peer without its explicit cooperation.

8.2.3. Misuse of AORs

A RELOAD overlay and enrollment service that liberally accept registrations for AORs of domain names unrelated to the overlay instance and without further justification, eventually store presence state for misused AORs. An attacker could hijack names, register a bogus presence and attract calls dedicated to a victim that resides within or outside the Overlay Instance.

A hijacking of AORs can be mitigated by restricting the name spaces admissible in the Overlay Instance, or by additional verification

actions of the enrollment service. To prevent an (exclusive) routing to a bogus registration, a caller can in addition query the DNS (or other discovery services at hand) to search for an alternative presence of the callee in another overlay or a normal SIP infrastructure.

8.2.4. Privacy Issues

All RELOAD SIP registration data is public. Methods of providing location and identity privacy are still being studied. Location privacy can be gained from using anonymous GRUUs.

9. IANA Considerations

9.1. Data Kind-ID

IANA shall register the following code point in the "RELOAD Data Kind-ID" Registry (cf., [I-D.ietf-p2psip-base]) to represent the SIP-REGISTRATION Kind, as described in Section 7. [NOTE TO IANA/RFC-EDITOR: Please replace RFC-AAAA with the RFC number for this specification in the following list.]

Kind	Kind-ID	RFC
SIP-REGISTRATION	1	RFC-AAAA

9.2. XML Name Space Registration

This document registers the following URI for the config XML namespace in the IETF XML registry defined in [RFC3688]

URI: urn:ietf:params:xml:ns:p2p:config-base:sip
Registrant Contact: The IESG
XML: N/A, the requested URI is an XML namespace

10. Acknowledgments

This document was generated in parts from initial drafts and discussions in the early specification phase of the P2PSIP base protocol. Significant contributions (in alphabetical order) were from David A. Bryan, James Deverick, Marcin Matuszewski, Jonathan Rosenberg, and Marcia Zangrilli, which is gratefully acknowledged.

Additional thanks go to all those who helped with ideas, discussions,

and reviews, in particular (in alphabetical order) Michael Chen, Marc Petit-Huguenin, Brian Rosen, and Matthias Waehlisch.

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[I-D.ietf-p2psip-base]
Jennings, C., Lowekamp, B., Rescorla, E., Baset, S., and H. Schulzrinne, "REsource LOcation And Discovery (RELOAD) Base Protocol", draft-ietf-p2psip-base-26 (work in progress), February 2013.

[RFC3261] 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.

[RFC2533] Klyne, G., "A Syntax for Describing Media Feature Sets", RFC 2533, March 1999.

[RFC2738] Klyne, G., "Corrections to "A Syntax for Describing Media Feature Sets"", RFC 2738, December 1999.

[RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, January 2004.

[RFC5245] Rosenberg, J., "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols", RFC 5245, April 2010.

[RFC5626] Jennings, C., Mahy, R., and F. Audet, "Managing Client-Initiated Connections in the Session Initiation Protocol (SIP)", RFC 5626, October 2009.

[RFC5627] Rosenberg, J., "Obtaining and Using Globally Routable User Agent URIs (GRUUs) in the Session Initiation Protocol (SIP)", RFC 5627, October 2009.

[RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, October 2006.

[IEEE-Posix]

"IEEE Standard for Information Technology - Portable Operating System Interface (POSIX) - Part 2: Shell and Utilities (Vol. 1)", IEEE Std 1003.2-1992, ISBN 1-55937-255-9, January 1993.

11.2. Informative References

[I-D.ietf-p2psip-concepts]
Bryan, D., Matthews, P., Shim, E., Willis, D., and S. Dawkins, "Concepts and Terminology for Peer to Peer SIP", draft-ietf-p2psip-concepts-05 (work in progress), July 2013.

[RFC5767] Munakata, M., Schubert, S., and T. Ohba, "User-Agent-Driven Privacy Mechanism for SIP", RFC 5767, April 2010.

[I-D.ietf-p2psip-share]
Knauf, A., Schmidt, T., Hege, G., and M. Waehlisch, "A Usage for Shared Resources in RELOAD (ShaRe)", draft-ietf-p2psip-share-01 (work in progress), February 2013.

Appendix A. Third Party Registration

In traditional SIP, the mechanism of a third party registration (i.e., an assistant acting for a boss, changing users register a role-based AOR, ...) is defined in Section 10.2 of [RFC3261]. This is a REGISTER which uses the URI of the third-party in its From header and cannot be translated directly into a P2PSIP registration, because only the owner of the certificate can store a SIP-REGISTRATION in a RELOAD overlay.

A way to implement third party registration is by using the extended access control mechanism USER-CHAIN-ACL defined in [I-D.ietf-p2psip-share]. Creating a new Kind "SIP-3P-REGISTRATION" that is ruled by USER-CHAIN-ACL allows the owner of the certificate to delegate the right for registration to individual third parties. In this way, original SIP functionality can be regained without weakening the security control of RELOAD.

Appendix B. Change Log

B.1. Changes since draft-ietf-p2psip-sip-09

- Added subsection on keepalive
- Updated references

B.2. Changes since draft-ietf-p2psip-sip-08

- Added the handling of SIPS
- Specified use of Posix regular expressions in configuration document
- Added IANA registration for namespace
- Editorial polishing
- Updated and extended references

B.3. Changes since draft-ietf-p2psip-sip-07

- Cleared open issues
- Clarified use cases after WG discussion
- Added configuration document extensions for configurable domain names
- Specified format of contact_prefs
- Clarified routing to AORs
- Extended security section
- Added Appendix on Third Party Registration
- Added IANA code points
- Editorial polishing
- Updated and extended references

B.4. Changes since draft-ietf-p2psip-sip-06

- Added Open Issue

Authors' Addresses

Cullen Jennings
Cisco
170 West Tasman Drive
MS: SJC-21/2
San Jose, CA 95134
USA

Phone: +1 408 421-9990
Email: fluffy@cisco.com

Bruce B. Lowekamp
Skype
Palo Alto, CA
USA

Email: bbl@lowekamp.net

Eric Rescorla
RTFM, Inc.
2064 Edgewood Drive
Palo Alto, CA 94303
USA

Phone: +1 650 678 2350
Email: ekr@rtfm.com

Salman A. Baset
Columbia University
1214 Amsterdam Avenue
New York, NY
USA

Email: salman@cs.columbia.edu

Henning Schulzrinne
Columbia University
1214 Amsterdam Avenue
New York, NY
USA

Email: hgs@cs.columbia.edu

Thomas C. Schmidt (editor)
HAW Hamburg
Berliner Tor 7
Hamburg 20099
Germany

Email: schmidt@informatik.haw-hamburg.de