HIPRG
Internet-Draft
Expires: April 18, 2005

H. Tschofenig
Siemens
V. Torvinen
Ericsson
J. Ott
Universitaet Bremen
H. Schulzrinne
Columbia U.
T. Henderson
The Boeing Company
G. Camarillo
Ericsson

October 18, 2004

Exchanging Host Identities in SIP
draft-tschofenig-hiprg-host-identities-00.txt

Status of this Memo

Copyright Notice

Abstract

This document proposes to exchange Host Identities (or Host Identity Tags) in SIP/SDP for later usage in the Host Identity Protocol Protocol (HIP) between the SIP user agents. As such, it is a first step in investigating the interaction between SIP and HIP and mainly a discussion document.

Table of Contents

1. Introduction

SIP [1] allows to establish and maintain sessions between two user agents. The communication typically involves SIP proxies before direct communication between the end points takes place. As part of the initial communication exchange a number of parameters are exchanged including security relevant parameters, such as keying material and cryptographic information to establish a security association for subsequent data traffic protection.

HIP (see [2] and [3]) creates an architecture with a new, cryptographic namespace and a new layer between the network and the transport layer to shield applications from the impact of multi-homing, readdressing and mobility. A protocol, the Host Identity Protocol, is used to establish state at the two end hosts. This state includes the establishment of IPsec SAs.

In order to provide security between two HIP end hosts beyond opportunistic encryption it is necessary to securely retrieve the Host Identities. A number of mechanisms can be used including directories (such as DNS) or more advanced concepts for example based on Distributed Hash Tables typically used in peer-to-peer networks.

This document suggests to exchange the Host Identities (or Host Identity Tags) as part of the initial SIP exchange inside the SDP payload. As such, the Host Identities can also be bound to the user identities - a concept not used in HIP.

The figure below illustrates the main idea:

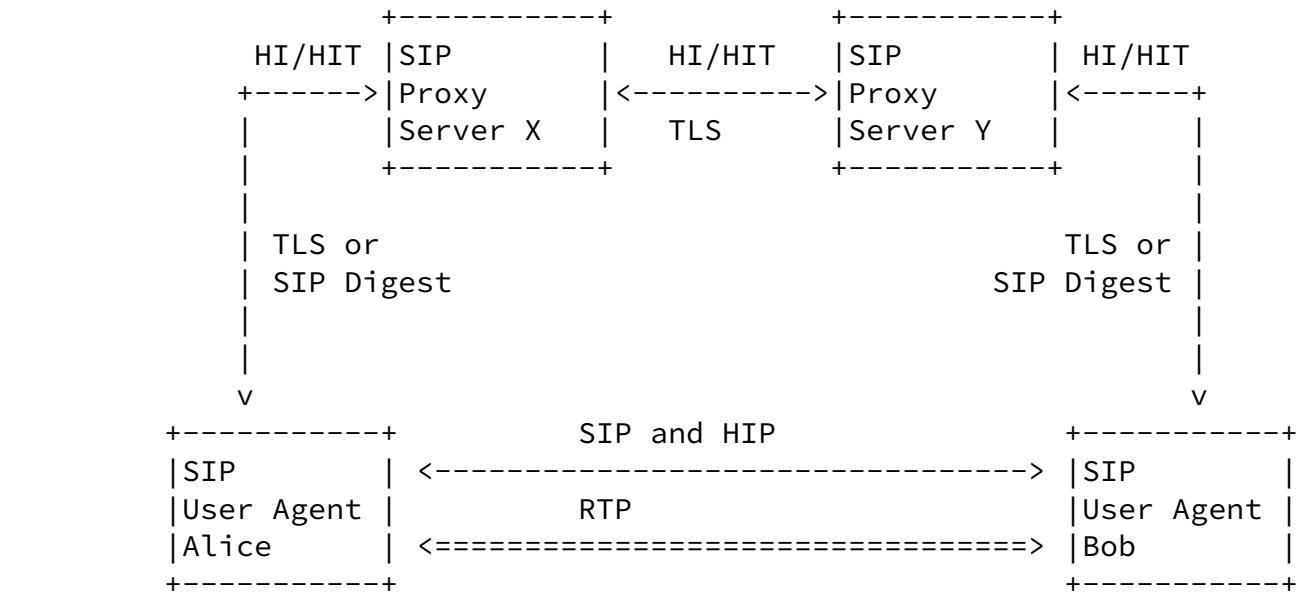

Figure 1: SIP Trapezoid

The initial SIP signaling messages between Alice and Bob often take place via the proxy servers. This exchange may be protected with TLS (between SIP proxies but also between SIP UAs and SIP proxies) or with SIP digest authentication between SIP UAs and the outbound proxy. Furthermore, SIP end-to-end security mechanisms are also

available with S/MIME.

This allows two hosts to securely exchange keys even if there are only domain-level public and private keys, as well as secure associations within a domain, thus avoiding the need for a global user-level PKI.

This initial message exchange is used to exchange Host Identities between the end points within the SDP payload.

Subsequently, when both user agents Alice and Bob communicate directly with each other they are able to reuse the Host Identity for the HIP message exchange.

If the SIP communication does not involve third parties (i.e., SIP proxies) and is therefore executed directly between the two SIP UAs then it is not useful to exchange Host Identities in the SDP payloads since the HIP exchange already took place before the first SIP message can be exchanged between the two peers. Still HIP might provide some advantages for the end-to-end communication, such as

providing security at the lower layer and mobility and multi-homing support.

The security of this approach relies on two properties:

- The signaling messages and the data traffic traverse a different path. Hence, an adversary needs to be located where it is able to see both, the signaling and the the data traffic.
- The signaling traffic is often protected.

2. SDP Extension

This document proposes to enhance the SDP [4] 'k' parameter.

This parameter has the following structure: k=<method><encryption key>. This document defines two new method fields:

k=host-identity:<HIP Host Identity>
k=host-identity-tag:<hash of the public key>

Both, the Host Identity and the Host Identity Tag are defined in [3]. The Host Identity contains the public key and a number of cryptographic parameters (such as used algorithms and Diffie-Hellmann public parameters). The Host Identity is base64 encoded.

FOR DISCUSSION:

The usage of the k parameter as defined in [5] is deprecated. [4] is more appropriate but like 'k=', they come with the caveat that they require a secured e2e signaling path (or SDP is S/MIME protected). One alternative is the usage of MIKEY for the exchange as defined in [6].

Furthermore, and probably more important, it is important to said what the Host Identity is supposed to be used with. They may help avoiding re-INVITEs when underlying IP addresses change to update the 'Contact:' address as well as the addresses in the 'c=' lines for the various media.

However, multiple devices may take part in the different media sessions (your laptop doing video in parallel to your hardware IP phone). To support these cases, it may be necessary to exchange _several_ HI(T)s within SDP and denote what they shall be used for. Such a mapping could naturally be achieved for each media stream (even using 'k=' attributes); at simple 'a=' attributes (or the mechanisms from [4]/ [6] would be preferred.

SDP only deals with media streams and does not have a notion of user or main device in the background. Hence, the SIP HI(T) may need to go into SIP signaling (rather than be carried in SDP).

Logically, this appears to belong to the 'Contact:' header which may be conveyed protected in an S/MIME body (signed and encrypted).

3. Example

This example contains the full details of the example session setup taken from Section 4 of [1]. The message flow is shown in Figure 1 of [1] and resembles the architecture shown in Figure 1. Note that these flows show the minimum required set of header fields; some other header fields such as Allow and Supported would normally be

present.

In our example Alice uses the following Host Identity Tag (7214148E0433AFE2FA2D48003D31172E) and Bob uses (44A5C522D7EDEDF962E55A0677DB1346) as the HIT. These HITs correspond to the following Host Identities (for convenience we reuse the XML representation format used by the Boeing implementation).

```
------
Alice:
------

<host_identity alg="DSA" alg_id="3" length="128"
 anon="no" incoming="yes">

<name>sip:alice@atlanta.com</name>

<P>D757262C4584C44C211F18BD96E5F061C4F0A423F7FE6B6B85B34CEF72CE14
A0D3A222FE08CECE65BE6C265854889DC1EDBD13EC8B274DA9F75BA26CCB98772
3602787E92BA84421F22C3C89CB9B06FD60FE01941DDD77FE6B12893DA76EEBC1
D128D97F0678D772B5341C8506F358214B16A2FAC4B368950387811C7DA33</P>

<Q>C773218C737EC8EE993B4F2DED30F48EDACE915F</Q>

<G>82269009E14EC474BAF2932E69D3B1F18517AD9594184CCDFCEAE96EC4D5EF
9313384B47093C52B20CD35D02492B3959EC6499625BC4FA5082E22C5B374E16D
D00132CE71020217091AC717B612391C76C1FB2E88317C1BD8171D41ECB83E210
C03CC9B32E81056C21621C73D6DAAC028F4B1585DA7F42519718CC9B09EEF</G>

<PUB>A4666AED5F5E753773DC961EDD0412A03F1F8D7CEC70A057076062804B86
619D3DA4E7610EBBDB05F44C5784622D1B86600DFCC1431BC4451D4FD31329354
07A9B24718CB82BAE93A4CDD9CC4C8B9A41C000AB53D52A65E8383F54F5BF92A8
21EA776A207C6991EF23808C00DB820977D97CAC01CB96307274E2386001327
</PUB>

<HIT>7214148E0433AFE2FA2D48003D31172E</HIT>
</host_identity>
```

```
----
Bob:
----

<host_identity alg="DSA" alg_id="3" length="96"
 anon="no" incoming="yes">

<name>sip:bob@biloxi.com</name>

<P>F13ACC1693AFD04B9E1E8D2A9DEA6DE8DE4C276BE2BF15B6CFF6E269B0169
378CB0DDDE23D187827015DC67E6768193914B823BDF215D0DAD7A151E434F9E
128DAFB9DEFAE07874621E70D7ED2D34B80A95FA8312B9564E4D118FB525664C
77D</P>

<Q>C773218C737EC8EE993B4F2DED30F48EDACE915F</Q>

<G>241F32CF48F424B1A75D33B7AE6088E745D9E24E653AE2CAEBE67E4AA1C11
15BA0CC25055A63C139235A95B36EFBC2064AF304C0F8A431D151B2B5854DE61
5168B45B9EAEBF9A88354CA7876E52D169E14E502BEA0CBB98B55AD2AB61620F
498</G>

<PUB>E481C20D8FBAA84F9C7ED8B5598F60F5A7D03951CA4783841EB8ADDC63D
DE11A2F3555C5641F465160AB1E016756D826B0F8CE4FDE33BA17F6FFFA751DA
1389A10E5599802AB1EBE4FD943405819A74FD6F1C9EA2815EE6B651610DF107
5D19F</PUB>

<HIT>44A5C522D7EDEDF962E55A0677DB1346</HIT>

</host_identity>
```

F1 INVITE Alice -> atlanta.com proxy

```
INVITE sip:bob@biloxi.com SIP/2.0
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds8
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: ...

v=0
o=alice 53655765 2353687637 IN IP4 pc33.atlanta.com
s=Session SDP
t=0 0
c=IN IP4 pc33.atlanta.com
m=audio 3456 RTP/AVP 0 1 3 99
a=rtpmap:0 PCMU/8000
k=host-identity-tag:7214148E0433AFE2FA2D48003D31172E
```

F2 100 Trying atlanta.com proxy -> Alice

```
SIP/2.0 100 Trying
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds8
 ;received=192.0.2.1
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Content-Length: 0
```

F3 INVITE atlanta.com proxy -> biloxi.com proxy

```
INVITE sip:bob@biloxi.com SIP/2.0
Via: SIP/2.0/UDP bigbox3.site3.atlanta.com
  ;branch=z9hG4bK77ef4c2312983.1
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds8
 ;received=192.0.2.1
Max-Forwards: 69
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: ...

v=0
o=alice 53655765 2353687637 IN IP4 pc33.atlanta.com
s=Session SDP
t=0 0
c=IN IP4 pc33.atlanta.com
m=audio 3456 RTP/AVP 0 1 3 99
a=rtpmap:0 PCMU/8000
k=host-identity-tag:7214148E0433AFE2FA2D48003D31172E
```

F4 100 Trying biloxi.com proxy -> atlanta.com proxy

```
SIP/2.0 100 Trying
Via: SIP/2.0/UDP bigbox3.site3.atlanta.com
 ;branch=z9hG4bK77ef4c2312983.1
 ;received=192.0.2.2
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds8
 ;received=192.0.2.1
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
```

```
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Content-Length: 0
```


F5 INVITE biloxi.com proxy -> Bob

```
INVITE sip:bob@192.0.2.4 SIP/2.0
Via: SIP/2.0/UDP server10.biloxi.com;branch=z9hG4bK4b43c2ff8.1
Via: SIP/2.0/UDP bigbox3.site3.atlanta.com
 ;branch=z9hG4bK77ef4c2312983.1
 ;received=192.0.2.2
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds8
 ;received=192.0.2.1
Max-Forwards: 68
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: ...

v=0
o=alice 53655765 2353687637 IN IP4 pc33.atlanta.com
s=Session SDP
t=0 0
c=IN IP4 pc33.atlanta.com
m=audio 3456 RTP/AVP 0 1 3 99
a=rtpmap:0 PCMU/8000
k=host-identity-tag:7214148E0433AFE2FA2D48003D31172E
```

F6 180 Ringing Bob -> biloxi.com proxy

```
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP server10.biloxi.com;branch=z9hG4bK4b43c2ff8.1
 ;received=192.0.2.3
Via: SIP/2.0/UDP bigbox3.site3.atlanta.com
 ;branch=z9hG4bK77ef4c2312983.1
 ;received=192.0.2.2
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds8
 ;received=192.0.2.1
To: Bob <sip:bob@biloxi.com>;tag=a6c85cf
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
Contact: <sip:bob@192.0.2.4>
CSeq: 314159 INVITE
Content-Length: 0
```


F7 180 Ringing biloxi.com proxy -> atlanta.com proxy

```
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP bigbox3.site3.atlanta.com
 ;branch=z9hG4bK77ef4c2312983.1
 ;received=192.0.2.2
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds8
 ;received=192.0.2.1
To: Bob <sip:bob@biloxi.com>;tag=a6c85cf
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
Contact: <sip:bob@192.0.2.4>
CSeq: 314159 INVITE
Content-Length: 0
```

F8 180 Ringing atlanta.com proxy -> Alice

```
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds8
 ;received=192.0.2.1
To: Bob <sip:bob@biloxi.com>;tag=a6c85cf
```

```
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
Contact: <sip:bob@192.0.2.4>
CSeq: 314159 INVITE
Content-Length: 0
```


F9 200 OK Bob -> biloxi.com proxy

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP server10.biloxi.com;branch=z9hG4bK4b43c2ff8.1
 ;received=192.0.2.3
Via: SIP/2.0/UDP bigbox3.site3.atlanta.com
 ;branch=z9hG4bK77ef4c2312983.1
 ;received=192.0.2.2
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds8
 ;received=192.0.2.1
To: Bob <sip:bob@biloxi.com>;tag=a6c85cf
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Contact: <sip:bob@192.0.2.4>
```

```
Content-Type: application/sdp
Content-Length: ...

v=0
o=bob 2890844527 2890844527 IN IP4 192.0.2.4
s=Session SDP
c=IN IP4 192.0.2.4
t=3034423619 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
k=host-identity-tag:44A5C522D7EDEDF962E55A0677DB1346
```


F10 200 OK biloxi.com proxy -> atlanta.com proxy

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP bigbox3.site3.atlanta.com
 ;branch=z9hG4bK77ef4c2312983.1
 ;received=192.0.2.2
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds8
 ;received=192.0.2.1
```

```
To: Bob <sip:bob@biloxi.com>;tag=a6c85cf
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Contact: <sip:bob@192.0.2.4>
Content-Type: application/sdp
Content-Length: ...

v=0
o=bob 2890844527 2890844527 IN IP4 192.0.2.4
s=Session SDP
c=IN IP4 192.0.2.4
t=3034423619 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
k=host-identity-tag:44A5C522D7EDEDF962E55A0677DB1346
```

F11 200 OK atlanta.com proxy -> Alice

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds8
 ;received=192.0.2.1
To: Bob <sip:bob@biloxi.com>;tag=a6c85cf
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Contact: <sip:bob@192.0.2.4>
Content-Type: application/sdp
Content-Length: ...

v=0
o=bob 2890844527 2890844527 IN IP4 192.0.2.4
s=Session SDP
c=IN IP4 192.0.2.4
t=3034423619 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
k=host-identity-tag:44A5C522D7EDEDF962E55A0677DB1346
```


F12 ACK Alice -> Bob

```
ACK sip:bob@192.0.2.4 SIP/2.0
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds9
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>;tag=a6c85cf
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 ACK
Content-Length: 0
```

The media session between Alice and Bob is now established.

The exchanged HITs are now placed in the pool of known HITs at both end hosts. As such there is also a binding established between URI and HIT at this point.

Next a regular HIP base exchange between Alice and Bob is started. As part of the exchange the two end hosts inspect their known-HITs pool and find the previously exchanged parameters.

```
Alice -> Bob: I1: Trigger exchange

Alice <- Bob: R1: {Puzzle, D-H(R), HI(R), ESP Transform,
              HIP Transform }SIG

Alice -> Bob: I2: {Solution, LSI(I), SPI(I), D-H(I),
              ESP Transform, HIP Transform, {H(I)}SK }SIG

Alice <- Bob: R2: {LSI(R), SPI(R), HMAC}SIG
```

As a result of this exchange, two IPsec SAs (one for each direction) is established. RTP media traffic can be exchanged between the two end hosts, Alice and Bob, protected by IPsec. If end host mobility takes place then a HIP readdressing exchange takes place which is not detected at the upper layer by UDP/RTP or SIP.

4. Security Considerations

This proposal is closely aligned towards the usage of the 'k' parameter in SDP [5]. As a difference, an asymmetric key is exchanged unlike the proposals illustrated in Section 6 of [5]. Section 5.12 of [8] is relevant for this discussion.

If an adversary aims to impersonate one of the SIP UAs in the subsequent HIP exchange then it is necessary to replace the Host Identity/Host Identity Tag exchanged in the SIP/SDP messages.

Please note that this approach is in a certain sense a re-instantiation of the Purpose-Built-Key (PBK) idea (see [9]). With PBK a hash of a public key is sent from node A to node B. If there was no adversary between A and B at that time to modify the transmitted hash value then subsequent communication interactions which use the public key are secure. This proposal reuses the same idea but focuses on the interworking between different protocols. In fact it would be possible to use the same approach to exchange the hash of an S/MIME certificate which can later be used in subsequent SIP signaling message exchanges.

If Host Identities for HIP can be retrieved using a different, more secure method then the Host Identities exchanged with SIP/SDP MUST NOT be used.

5. Open Issues

The authors came accross a number of open issues while thinking about this topic:

- The authors discussed the usage of SUBSCRIBE/NOTIFY to distribute Host Identities. This approach is particularly interesting, if Host Identities are subject to frequent change. As such, it would resemble the proposal provided with SIPPING-CERT [10]. Thereby the user agent would be allowed to upload its own Host Identity to the Credential Server. Other user agents would use the SUBSCRIBE method to retrieve Host Identities of a particular user. With the help of the NOTIFY message it is possible to learn about a changed Host Identity (e.g., a revoked HI). It is for further study whether this is more useful than the already described proposal.
- Is an IANA registration for the method field required?
- Is it possible to carry more than one Host Identity/Host Identity Tag in the SDP payload by listing more than one 'k' parameter?
- Further investigations are required with regard to the mobility functionality provided by HIP and the potential benefits for end-to-end signaling using SIP, RTP etc. between the SIP UAs.
- Middlebox traversal functionality discussed in the context of HIP (such as STUN, TURN, ICE) could potentially be replaced by the HIP middlebox traversal functionality.

6. Acknowledgments

The authors would like to thank Steffen Fries, Aarthi Nagarajan, Murugaraj Shanmugam, Franz Muenz, Jochen Grimminger and Joachim Kross for their feedback.

7. References

7.1 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] Moskowitz, R., "Host Identity Protocol Architecture", draft-moskowitz-hip-arch-06 (work in progress), June 2004.

[3] Moskowitz, R., "Host Identity Protocol", draft-ietf-hip-base-00 (work in progress), June 2004.

[4] Andreasen, F., Baugher, M. and D. Wing, "Session Description Protocol Security Descriptions for Media Streams", draft-ietf-mmusic-sdescriptions-07 (work in progress), July 2004.

[5] Handley, M. and V. Jacobson, "SDP: Session Description Protocol", RFC 2327, April 1998.

[6] Arkko, J., Carrara, E., Lindholm, F., Naslund, M. and K. Norrman, "Key Management Extensions for Session Description Protocol (SDP) and Real Time Streaming Protocol (RTSP)",

draft-ietf-mmusic-kmgmt-ext-11 (work in progress), April 2004.

[7] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", March 1997.

7.2 Informative References

[8] Handley, M., Jacobson, V. and C. Perkins, "SDP: Session Description Protocol", draft-ietf-mmusic-sdp-new-20 (work in progress), September 2004.

[9] Bradner, S., Mankin, A. and J. Schiller, "A Framework for Purpose-Built Keys (PBK)", draft-bradner-pbk-frame-06 (work in progress), June 2003.

[10] Jennings, C., "Certificate Management Service for SIP", draft-jennings-sipping-certs-04 (work in progress), July 2004.

Authors' Addresses

Hannes Tschofenig
Siemens
Otto-Hahn-Ring 6
Munich, Bayern 81739
Germany

EMail: Hannes.Tschofenig@siemens.com

Vesa Torvinen
Ericsson
Joukahaisenkatu 1
Turku 20520
Finland

EMail: vesa.torvinen@ericsson.com

Joerg Ott
Universitaet Bremen
Bibliothekstr. 1
Bremen 28359
Germany

EMail: jo@tzi.org

Henning Schulzrinne
Columbia University
Department of Computer Science
450 Computer Science Building
New York, NY 10027
USA

Phone: +1 212 939 7042
EMail: schulzrinne@cs.columbia.edu
URI: http://www.cs.columbia.edu/~hgs

Thomas R. Henderson
The Boeing Company
P.O. Box 3707
Seattle, WA
USA

EMail: thomas.r.henderson@boeing.com

Gonzalo Camarillo
Ericsson
Hirsalantie 11
Jorvas 02420
Finland

EMail: Gonzalo.Camarillo@ericsson.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

Copyright Statement

Acknowledgment

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