ï»¿

Internet Engineering Task Force (IETF)
Request for Comments: 8862
BCP: 228
Category: Best Current Practice
ISSN: 2070-1721

J. Peterson
Neustar
R. Barnes
Cisco
R. Housley
Vigil Security

January 2021

Best Practices for Securing RTP Media Signaled with SIP

Abstract

Although the Session Initiation Protocol (SIP) includes a suite of security services that has been expanded by numerous specifications over the years, there is no single place that explains how to use SIP to establish confidential media sessions. Additionally, existing mechanisms have some feature gaps that need to be identified and resolved in order for them to address the pervasive monitoring threat model. This specification describes best practices for negotiating confidential media with SIP, including a comprehensive protection solution that binds the media layer to SIP layer identities.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The Session Initiation Protocol (SIP) [RFC3261] includes a suite of security services, including Digest Authentication [RFC7616] for authenticating entities with a shared secret, TLS [RFC8446] for transport security, and (optionally) S/MIME [RFC8551] for body security. SIP is frequently used to establish media sessions -- in particular, audio or audiovisual sessions, which have their own security mechanisms available, such as the Secure Real-time Transport Protocol (SRTP) [RFC3711]. However, the practices needed to bind security at the media layer to security at the SIP layer, to provide an assurance that protection is in place all the way up the stack, rely on a great many external security mechanisms and practices. This document provides documentation to explain their optimal use as a best practice.

Revelations about widespread pervasive monitoring of the Internet have led to a greater desire to protect Internet communications [RFC7258]. In order to maximize the use of security features, especially of media confidentiality, opportunistic measures serve as a stopgap when a full suite of services cannot be negotiated all the way up the stack. Opportunistic media security for SIP is described in [RFC8643], which builds on the prior efforts of [Best-Effort-SRTP]. With opportunistic encryption, there is an attempt to negotiate the use of encryption, but if the negotiation fails, then cleartext is used. Opportunistic encryption approaches typically have no integrity protection for the keying material.

This document contains the SIP Best-practice Recommendations Against Network Dangers to privacY (SIPBRANDY) profile of Secure Telephone Identity Revisited (STIR) [RFC8224] for media confidentiality, providing a comprehensive security solution for SIP media that includes integrity protection for keying material and offers application-layer assurance that media confidentiality is in place. Various specifications that User Agents (UAs) must implement to support media confidentiality are given in the sections below; a summary of the best current practices appears in Section 8.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Security at the SIP and SDP Layer

There are two approaches to providing confidentiality for media sessions set up with SIP: comprehensive protection and opportunistic security (as defined in [RFC7435]). This document only addresses comprehensive protection.

Comprehensive protection for media sessions established by SIP requires the interaction of three protocols: the Session Initiation Protocol (SIP) [RFC3261], the Session Description Protocol (SDP) [RFC4566], and the Real-time Transport Protocol (RTP) [RFC3550] -- in particular, its secure profile SRTP [RFC3711]. Broadly, it is the responsibility of SIP to provide integrity protection for the media keying attributes conveyed by SDP, and those attributes will in turn identify the keys used by endpoints in the RTP media session(s) that SDP negotiates.

Note that this framework does not apply to keys that also require confidentiality protection in the signaling layer, such as the SDP "k=" line, which MUST NOT be used in conjunction with this profile.

In that way, once SIP and SDP have exchanged the necessary information to initiate a session, media endpoints will have a strong assurance that the keys they exchange have not been tampered with by third parties and that end-to-end confidentiality is available.

To establish the identity of the endpoints of a SIP session, this specification uses STIR [RFC8224]. The STIR Identity header has been designed to prevent a class of impersonation attacks that are commonly used in robocalling, voicemail hacking, and related threats. STIR generates a signature over certain features of SIP requests, including header field values that contain an identity for the originator of the request, such as the From header field or P-Asserted-Identity field, and also over the media keys in SDP if they are present. As currently defined, STIR provides a signature over the "a=fingerprint" attribute, which is a fingerprint of the key used by DTLS-SRTP [RFC5763]; consequently, STIR only offers comprehensive protection for SIP sessions in concert with SDP and SRTP when DTLS-SRTP is the media security service. The underlying Personal Assertion Token (PASSporT) object [RFC8225] used by STIR is extensible, however, and it would be possible to provide signatures over other SDP attributes that contain alternate keying material. A profile for using STIR to provide media confidentiality is given in Section 4.

4. STIR Profile for Endpoint Authentication and Verification Services

STIR [RFC8224] defines the Identity header field for SIP, which provides a cryptographic attestation of the source of communications. This document includes a profile of STIR, called the SIPBRANDY profile, where the STIR verification service will act in concert with an SRTP media endpoint to ensure that the key fingerprints, as given in SDP, match the keys exchanged to establish DTLS-SRTP. To satisfy this condition, the verification service function would in this case be implemented in the SIP User Agent Server (UAS), which would be composed with the media endpoint. If the STIR authentication service or verification service functions are implemented at an intermediary rather than an endpoint, this introduces the possibility that the intermediary could act as a man in the middle, altering key fingerprints. As this attack is not in STIR's core threat model, which focuses on impersonation rather than man-in-the-middle attacks, STIR offers no specific protections against such interference.

The SIPBRANDY profile for media confidentiality thus shifts these responsibilities to the endpoints rather than the intermediaries. While intermediaries MAY provide the verification service function of STIR for SIPBRANDY transactions, the verification needs to be repeated at the endpoint to obtain end-to-end assurance. Intermediaries supporting this specification MUST NOT block or otherwise redirect calls if they do not trust the signing credential. The SIPBRANDY profile is based on an end-to-end trust model, so it is up to the endpoints to determine if they support signing credentials, not intermediaries.

In order to be compliant with best practices for SIP media confidentiality with comprehensive protection, UA implementations MUST implement both the authentication service and verification service roles described in [RFC8224]. STIR authentication services MUST signal their compliance with this specification by including the "msec" claim defined in this specification to the PASSporT payload. Implementations MUST provide key fingerprints in SDP and the appropriate signatures over them as specified in [RFC8225].

When generating either an offer or an answer [RFC3264], compliant implementations MUST include an "a=fingerprint" attribute containing the fingerprint of an appropriate key (see Section 4.1).

4.1. Credentials

In order to implement the authentication service function in the UA, SIP endpoints will need to acquire the credentials needed to sign for their own identity. That identity is typically carried in the From header field of a SIP request and contains either a greenfield SIP URI (e.g., "sip:alice@example.com") or a telephone number (which can appear in a variety of ways, e.g., "sip:+17004561212@example.com;user=phone"). Section 8 of [RFC8224] contains guidance for separating the two and determining what sort of credential is needed to sign for each.

To date, few commercial certification authorities (CAs) issue certificates for SIP URIs or telephone numbers; though work is ongoing on systems for this purpose (such as [ACME-Auth-Token]), it is not yet mature enough to be recommended as a best practice. This is one reason why STIR permits intermediaries to act as an authentication service on behalf of an entire domain, just as in SIP a proxy server can provide domain-level SIP service. While CAs that offer proof-of-possession certificates similar to those used for email could be offered for SIP -- for either greenfield identifiers or telephone numbers -- this specification does not require their use.

For users who do not possess such certificates, DTLS-SRTP [RFC5763] permits the use of self-signed public keys. The profile of STIR in this document, called the SIPBRANDY profile, employs the more relaxed authority requirements of [RFC8224] to allow the use of self-signed public keys for authentication services that are composed with UAs, by generating a certificate (per the guidance in [RFC8226]) with a subject corresponding to the user's identity. To obtain comprehensive protection with a self-signed certificate, some out-of-band verification is needed as well. Such a credential could be used for trust on first use (see [RFC7435]) by relying parties. Note that relying parties SHOULD NOT use certificate revocation mechanisms or real-time certificate verification systems for self-signed certificates, as they will not increase confidence in the certificate.

Users who wish to remain anonymous can instead generate self-signed certificates as described in Section 4.2.

Generally speaking, without access to out-of-band information about which certificates were issued to whom, it will be very difficult for relying parties to ascertain whether or not the signer of a SIP request is genuinely an "endpoint". Even the term "endpoint" is a problematic one, as SIP UAs can be composed in a variety of architectures and may not be devices under direct user control. While it is possible that techniques based on certificate transparency [RFC6962] or similar practices could help UAs to recognize one another's certificates, those operational systems will need to ramp up with the CAs that issue credentials to end-user devices going forward.

4.2. Anonymous Communications

In some cases, the identity of the initiator of a SIP session may be withheld due to user or provider policy. Following the recommendations of [RFC3323], this may involve using an identity such as "anonymous@anonymous.invalid" in the identity fields of a SIP request. [RFC8224] does not currently permit authentication services to sign for requests that supply this identity. It does, however, permit signing for valid domains, such as "anonymous@example.com", as a way of implementing an anonymization service as specified in [RFC3323].

Even for anonymous sessions, providing media confidentiality and partial SDP integrity is still desirable. One-time self-signed certificates for anonymous communications SHOULD include a subjectAltName of "sip:anonymous@anonymous.invalid". After a session is terminated, the certificate SHOULD be discarded, and a new one, with fresh keying material, SHOULD be generated before each future anonymous call. As with self-signed certificates, relying parties SHOULD NOT use certificate revocation mechanisms or real-time certificate verification systems for anonymous certificates, as they will not increase confidence in the certificate.

Note that when using one-time anonymous self-signed certificates, any man in the middle could strip the Identity header and replace it with one signed by its own one-time certificate, changing the "mky" parameters of PASSporT and any "a=fingerprint" attributes in SDP as it chooses. This signature only provides protection against non-Identity-aware entities that might modify SDP without altering the PASSporT conveyed in the Identity header.

4.3. Connected Identity Usage

STIR [RFC8224] provides integrity protection for the fingerprint attributes in SIP request bodies but not SIP responses. When a session is established, therefore, any SDP body carried by a 200-class response in the backwards direction will not be protected by an authentication service and cannot be verified. Thus, sending a secured SDP body in the backwards direction will require an extra RTT, typically a request sent in the backwards direction.

[RFC4916] explored the problem of providing "connected identity" to implementations of [RFC4474] (which is obsoleted by [RFC8224]); [RFC4916] uses a provisional or mid-dialog UPDATE request in the backwards (reverse) direction to convey an Identity header field for the recipient of an INVITE. The procedures in [RFC4916] are largely compatible with the revision of the Identity header in [RFC8224]. However, the following need to be considered:

* The UPDATE carrying signed SDP with a fingerprint in the backwards direction needs to be sent during dialog establishment, following the receipt of a Provisional Response Acknowledgement (PRACK) after a provisional 1xx response.

* For use with this SIPBRANDY profile for media confidentiality, the UAS that responds to the INVITE request needs to act as an authentication service for the UPDATE sent in the backwards direction.

* Per the text in Section 4.4.1 of [RFC4916] regarding the receipt at a User Agent Client (UAC) of error code 428, 436, 437, or 438 in response to a mid-dialog request, it is RECOMMENDED that the dialog be treated as terminated. However, Section 6.1.1 of [RFC8224] allows the retransmission of requests with repairable error conditions. In particular, an authentication service might retry a mid-dialog rather than treating the dialog as terminated, although only one such retry is permitted.

* Note that the examples in [RFC4916] are based on [RFC4474] and will not match signatures using [RFC8224].

Future work may be done to revise [RFC4916] for STIR; that work should take into account any impacts on the SIPBRANDY profile described in this document. The use of [RFC4916] has some further interactions with Interactive Connectivity Establishment (ICE) [RFC8445]; see Section 7.

4.4. Authorization Decisions

[RFC8224] grants STIR verification services a great deal of latitude when making authorization decisions based on the presence of the Identity header field. It is largely a matter of local policy whether an endpoint rejects a call based on the absence of an Identity header field, or even the presence of a header that fails an integrity check against the request.

For this SIPBRANDY profile of STIR, however, a compliant verification service that receives a dialog-forming SIP request containing an Identity header with a PASSporT type of "msec", after validating the request per the steps described in Section 6.2 of [RFC8224], MUST reject the request if there is any failure in that validation process with the appropriate status code per Section 6.2.2 of [RFC8224]. If the request is valid, then if a terminating user accepts the request, it MUST then follow the steps in Section 4.3 to act as an authentication service and send a signed request with the "msec" PASSporT type in its Identity header as well, in order to enable end-to-end bidirectional confidentiality.

For the purposes of this profile, the "msec" PASSporT type can be used by authentication services in one of two ways: as a mandatory request for media security or as a merely opportunistic request for media security. As any verification service that receives an Identity header field in a SIP request with an unrecognized PASSporT type will simply ignore that Identity header, an authentication service will know whether or not the terminating side supports "msec" based on whether or not its UA receives a signed request in the backwards direction per Section 4.3. If no such requests are received, the UA may do one of two things: shut down the dialog, if the policy of the UA requires that "msec" be supported by the terminating side for this dialog; or, if policy permits (e.g., an explicit acceptance by the user), allow the dialog to continue without media security.

5. Media Security Protocols

As there are several ways to negotiate media security with SDP, any of which might be used with either opportunistic or comprehensive protection, further guidance to implementers is needed. In [RFC8643], opportunistic approaches considered include DTLS-SRTP, security descriptions [RFC4568], and ZRTP [RFC6189].

Support for DTLS-SRTP is REQUIRED by this specification.

The "mky" claim of PASSporT provides integrity protection for "a=fingerprint" attributes in SDP, including cases where multiple "a=fingerprint" attributes appear in the same SDP.

6. Relayed Media and Conferencing

Providing end-to-end media confidentiality for SIP is complicated by the presence of many forms of media relays. While many media relays merely proxy media to a destination, others present themselves as media endpoints and terminate security associations before re-originating media to its destination.

Centralized conference bridges are one type of entity that typically terminates a media session in order to mux media from multiple sources and then to re-originate the muxed media to conference participants. In many such implementations, only hop-by-hop media confidentiality is possible. Work is ongoing to specify a means to encrypt both (1) the hop-by-hop media between a UA and a centralized server and (2) the end-to-end media between UAs, but it is not sufficiently mature at this time to become a best practice. Those protocols are expected to identify their own best-practice recommendations as they mature.

Another class of entities that might relay SIP media are Back-to-Back User Agents (B2BUAs). If a B2BUA follows the guidance in [RFC7879], it may be possible for B2BUAs to act as media relays while still permitting end-to-end confidentiality between UAs.

Ultimately, if an endpoint can decrypt media it receives, then that endpoint can forward the decrypted media without the knowledge or consent of the media's originator. No media confidentiality mechanism can protect against these sorts of relayed disclosures or against a legitimate endpoint that can legitimately decrypt media and record a copy to be sent elsewhere (see [RFC7245]).

7. ICE and Connected Identity

Providing confidentiality for media with comprehensive protection requires careful timing of when media streams should be sent and when a user interface should signify that confidentiality is in place.

In order to best enable end-to-end connectivity between UAs and to avoid media relays as much as possible, implementations of this specification MUST support ICE [RFC8445] [RFC8839]. To speed up call establishment, it is RECOMMENDED that implementations support Trickle ICE [RFC8838] [RFC8840].

Note that in the comprehensive protection case, the use of connected identity [RFC4916] with ICE implies that the answer containing the key fingerprints, and thus the STIR signature, will come in an UPDATE sent in the backwards direction, a provisional response, and a PRACK, rather than in any earlier SDP body. Only at such a time as that UPDATE is received will the media keys be considered exchanged in this case.

Similarly, in order to prevent, or at least mitigate, the denial-of-service attack described in Section 19.5.1 of [RFC8445], this specification incorporates best practices for ensuring that recipients of media flows have consented to receive such flows. Implementations of this specification MUST implement the Session Traversal Utilities for NAT (STUN) usage for consent freshness defined in [RFC7675].

8. Best Current Practices

The following are the best practices for SIP UAs to provide media confidentiality for SIP sessions.

* Implementations MUST support the SIPBRANDY profile as defined in Section 4 and signal such support in PASSporT via the "msec" header element.

* Implementations MUST follow the authorization decision behavior described in Section 4.4.

* Implementations MUST support DTLS-SRTP for management of keys, as described in Section 5.

* Implementations MUST support ICE and the STUN consent freshness mechanism, as specified in Section 7.

9. IANA Considerations

This specification defines a new value for the "Personal Assertion Token (PASSporT) Extensions" registry called "msec". IANA has added the entry to the registry with a value pointing to this document.

10. Security Considerations

This document describes the security features that provide media sessions established with SIP with confidentiality, integrity, and authentication.

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, DOI 10.17487/RFC3261, June 2002, <https://www.rfc-editor.org/info/rfc3261>.

[RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, DOI 10.17487/RFC3264, June 2002, <https://www.rfc-editor.org/info/rfc3264>.

[RFC3323] Peterson, J., "A Privacy Mechanism for the Session Initiation Protocol (SIP)", RFC 3323, DOI 10.17487/RFC3323, November 2002, <https://www.rfc-editor.org/info/rfc3323>.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, DOI 10.17487/RFC3550, July 2003, <https://www.rfc-editor.org/info/rfc3550>.

[RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, DOI 10.17487/RFC3711, March 2004, <https://www.rfc-editor.org/info/rfc3711>.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, DOI 10.17487/RFC4566, July 2006, <https://www.rfc-editor.org/info/rfc4566>.

[RFC4568] Andreasen, F., Baugher, M., and D. Wing, "Session Description Protocol (SDP) Security Descriptions for Media Streams", RFC 4568, DOI 10.17487/RFC4568, July 2006, <https://www.rfc-editor.org/info/rfc4568>.

[RFC4916] Elwell, J., "Connected Identity in the Session Initiation Protocol (SIP)", RFC 4916, DOI 10.17487/RFC4916, June 2007, <https://www.rfc-editor.org/info/rfc4916>.

[RFC5763] Fischl, J., Tschofenig, H., and E. Rescorla, "Framework for Establishing a Secure Real-time Transport Protocol (SRTP) Security Context Using Datagram Transport Layer Security (DTLS)", RFC 5763, DOI 10.17487/RFC5763, May 2010, <https://www.rfc-editor.org/info/rfc5763>.

[RFC7258] Farrell, S. and H. Tschofenig, "Pervasive Monitoring Is an Attack", BCP 188, RFC 7258, DOI 10.17487/RFC7258, May 2014, <https://www.rfc-editor.org/info/rfc7258>.

[RFC7675] Perumal, M., Wing, D., Ravindranath, R., Reddy, T., and M. Thomson, "Session Traversal Utilities for NAT (STUN) Usage for Consent Freshness", RFC 7675, DOI 10.17487/RFC7675, October 2015, <https://www.rfc-editor.org/info/rfc7675>.

[RFC7879] Ravindranath, R., Reddy, T., Salgueiro, G., Pascual, V., and P. Ravindran, "DTLS-SRTP Handling in SIP Back-to-Back User Agents", RFC 7879, DOI 10.17487/RFC7879, May 2016, <https://www.rfc-editor.org/info/rfc7879>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8224] Peterson, J., Jennings, C., Rescorla, E., and C. Wendt, "Authenticated Identity Management in the Session Initiation Protocol (SIP)", RFC 8224, DOI 10.17487/RFC8224, February 2018, <https://www.rfc-editor.org/info/rfc8224>.

[RFC8225] Wendt, C. and J. Peterson, "PASSporT: Personal Assertion Token", RFC 8225, DOI 10.17487/RFC8225, February 2018, <https://www.rfc-editor.org/info/rfc8225>.

[RFC8226] Peterson, J. and S. Turner, "Secure Telephone Identity Credentials: Certificates", RFC 8226, DOI 10.17487/RFC8226, February 2018, <https://www.rfc-editor.org/info/rfc8226>.

[RFC8445] Keranen, A., Holmberg, C., and J. Rosenberg, "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal", RFC 8445, DOI 10.17487/RFC8445, July 2018, <https://www.rfc-editor.org/info/rfc8445>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

[RFC8838] Ivov, E., Uberti, J., and P. Saint-Andre, "Trickle ICE: Incremental Provisioning of Candidates for the Interactive Connectivity Establishment (ICE) Protocol", RFC 8838, DOI 10.17487/RFC8838, January 2021, <https://www.rfc-editor.org/info/rfc8838>.

[RFC8839] Petit-Huguenin, M., Nandakumar, S., Holmberg, C., KerÃ¤nen, A., and R. Shpount, "Session Description Protocol (SDP) Offer/Answer Procedures for Interactive Connectivity Establishment (ICE)", RFC 8839, DOI 10.17487/RFC8839, January 2021, <https://www.rfc-editor.org/info/rfc8839>.

[RFC8840] Ivov, E., Stach, T., Marocco, E., and C. Holmberg, "A Session Initiation Protocol (SIP) Usage for Incremental Provisioning of Candidates for the Interactive Connectivity Establishment (Trickle ICE)", RFC 8840, DOI 10.17487/RFC8840, January 2021, <https://www.rfc-editor.org/info/rfc8840>.

11.2. Informative References

[ACME-Auth-Token]
Peterson, J., Barnes, M., Hancock, D., and C. Wendt, "ACME Challenges Using an Authority Token", Work in Progress, Internet-Draft, draft-ietf-acme-authority-token-05, 9 March 2020, <https://tools.ietf.org/html/draft-ietf-acme-authority-token-05>.

[Best-Effort-SRTP]
Kaplan, H. and F. Audet, "Session Description Protocol (SDP) Offer/Answer Negotiation For Best-Effort Secure Real-Time Transport Protocol", Work in Progress, Internet-Draft, draft-kaplan-mmusic-best-effort-srtp-01, 25 October 2006, <https://tools.ietf.org/html/draft-kaplan-mmusic-best-effort-srtp-01>.

[RFC4474] Peterson, J. and C. Jennings, "Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP)", RFC 4474, DOI 10.17487/RFC4474, August 2006, <https://www.rfc-editor.org/info/rfc4474>.

[RFC6189] Zimmermann, P., Johnston, A., Ed., and J. Callas, "ZRTP: Media Path Key Agreement for Unicast Secure RTP", RFC 6189, DOI 10.17487/RFC6189, April 2011, <https://www.rfc-editor.org/info/rfc6189>.

[RFC6962] Laurie, B., Langley, A., and E. Kasper, "Certificate Transparency", RFC 6962, DOI 10.17487/RFC6962, June 2013, <https://www.rfc-editor.org/info/rfc6962>.

[RFC7245] Hutton, A., Ed., Portman, L., Ed., Jain, R., and K. Rehor, "An Architecture for Media Recording Using the Session Initiation Protocol", RFC 7245, DOI 10.17487/RFC7245, May 2014, <https://www.rfc-editor.org/info/rfc7245>.

[RFC7435] Dukhovni, V., "Opportunistic Security: Some Protection Most of the Time", RFC 7435, DOI 10.17487/RFC7435, December 2014, <https://www.rfc-editor.org/info/rfc7435>.

[RFC7616] Shekh-Yusef, R., Ed., Ahrens, D., and S. Bremer, "HTTP Digest Access Authentication", RFC 7616, DOI 10.17487/RFC7616, September 2015, <https://www.rfc-editor.org/info/rfc7616>.

[RFC8551] Schaad, J., Ramsdell, B., and S. Turner, "Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 4.0 Message Specification", RFC 8551, DOI 10.17487/RFC8551, April 2019, <https://www.rfc-editor.org/info/rfc8551>.

[RFC8643] Johnston, A., Aboba, B., Hutton, A., Jesske, R., and T. Stach, "An Opportunistic Approach for Secure Real-time Transport Protocol (OSRTP)", RFC 8643, DOI 10.17487/RFC8643, August 2019, <https://www.rfc-editor.org/info/rfc8643>.

Acknowledgements

We thank Eric Rescorla, Adam Roach, Andrew Hutton, and Ben Campbell for contributions to this problem statement and framework. We thank Liang Xia and Alissa Cooper for their careful review.

Authors' Addresses

Jon Peterson
Neustar, Inc.

Email: jon.peterson@team.neustar

Richard Barnes
Cisco

Email: rlb@ipv.sx

Russ Housley
Vigil Security, LLC

Email: housley@vigilsec.com