Dispatch
Internet-Draft
Intended status: Standards Track
Expires: July 13, 2014

O. Johansson
Edvina AB

January 9, 2014

TLS sessions in SIP using DNS-based Authentication of Named Entities (DANE) TLSA records
draft-johansson-dispatch-dane-sip-01

Abstract

Use of TLS in the SIP protocol is defined in multiple documents, starting with RFC 3261. The actual verification that happens when setting up a SIP TLS connection to a SIP server based on a SIP URI is described in detail in RFC 5922 - SIP Domain Certificates.

In this document, an alternative method is defined, using DNS-Based Authentication of Named Entities (DANE). By looking up TLSA DNS records and using DNSsec protection of the required queries, including lookups for NAPTR and SRV records, a SIP Client can verify the identity of the TLS SIP server in a different way, matching on the SRV host name in the X.509 PKIX certificate instead of the SIP domain. This provides more scalability in hosting solutions and make it easier to use standard CA certificates (if needed at all).

Table of Contents

1. Introduction

RFC 3261 [RFC3261] defines how to use TLS in the SIP protocol, but doesn't describe the actual verification between a SIP request and a TLS server certificate in detail. RFC 5922 [RFC5922] updates RFC 3261 with a definition of how a SIP client matches a PKIX X.509 [RFC5280] certificate provided by a TLS-enabled SIP server with the domain of a SIP request that caused the connection to be set up. Verification is done using the domain part of the SIP URI and the X.509 SubjectAltName extension of type dNSName or uniformResourceIdentifier. This is called "domain verification" as opposed to "host verification" in RFC 5922.

Including all domains hosted by a server in a server's certificate doesn't provide for a scalable and easy-managed solution. Every time

a service adds a domain, a new certificate will need to be provided, unless TLS Server Name Identification (SNI) is used, where each domain can have it's own certificate. Having one certificate per domain and subdomain adds to the administration of a service. In addition, no known commercial CA offers certificate services with SIP URI's in the certificates.

Using DNSsec and DNS-based Authentication of Named Entities (DANE)[RFC6698] the chain from a SIP uri to a TLS certificate changes, as outlined in this document. With DNSsec, the DNS lookups are authenticated and can be verified and trusted. [I-D.ietf-dane-srv] describes a DANE-based chain of trust, matching the SRV host name with the contents of the certificate.

This document describes how a SIP implementation can use DANE to set up a secure connection to a SIP server with TLS support. In addition, we describe how a server can provide support for RFC 5922-style clients with the same certificate, if needed.

This document adds an alternative to RFC5922 so that SIP implementations supporting DANE can validate a SIP domain identity using secure DNS queries and the identity of the SIP host by verifying the certificate using the SRV host name found in a SubjectAltName extension of type DNSName in the certificate. The domain verification will now happen based on DNSsec and the TLS verification will be based on host names (host verification in RFC 5922).

In order to learn about DANE and the different ways a TLSA record can be constructed, readers of this document needs to also read RFC 6698 [RFC6698].

2. Terminology and Conventions Used in This Document

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].

RFC 3261 [RFC3261] defines additional terms used in this document that are specific to the SIP domain such as "proxy"; "registrar"; "redirect server"; "user agent server" or "UAS"; "user agent client" or "UAC"; "back-to-back user agent" or "B2BUA"; "dialog"; "transaction"; "server transaction".

This document uses the term "SIP Server" that is defined to include the following SIP entities: user agent server (UAS), registrar, redirect server, a SIP proxy in the role of user agent server, and a B2BUA in the role of a user agent server.

This document uses the term "SIP client" that is defined to include the following SIP entities: user agent client(UAC), a SIP proxy in the role of user agent client, and a B2BUA in the role of a user agent client.

3. Using DNS in the SIP protocol

RFC 3263[RFC3263] describes how a SIP implementation use DNS to find the next hop server. The first step is to look up a DNS NAPTR record for the domain part of the URI. NAPTR records are used by the target domain to indicate reachability using different transports. NAPTR may be used to indicate a preference for TLS/TCP connections.

The result of the NAPTR lookup is a DNS name used to query for DNS SRV records. The list of DNS SRV records indicate host names that are queried for to find A or AAAA records with IP addresses.

SIP SRV records for TLS/TCP are using the prefix _sips._tcp, as in the DNS name _sips._tcp.example.com.

A SIP implementation with no support for NAPTR may, based on configuration or URI scheme, choose to set up a TLS session to the target domain.

> In rare cases, no SRV lookup is done. This means that the implementation lacks capability to do load balancing and failover based on the information in the DNS. These type of clients are not considered in this document.

4. Why DNSsec is important for SIP

DNS relies on DNS lookups not only to find the next hop server, but also for server administrators to provide failover and to load balance clients. The result of querying for one domain may need to SRV records or host names in another domain. Without DNSsec, an attacker can forge DNS replies and issue bogus DNS records, directing traffic to a bad server. This applies to calls as well as instant messaging, chat and presense.

5. Secure delegation is required for DANE to apply

It is important for implementors to understand the concept of "secure" DNSsec validation according to RFC 4033[RFC4033]. For this specification to take effect, all DNS RRsets in the chain from SIP URI to IP address and TLSA record must be secure. (This corresponds to the A.D. bit being set in the responses).

If any RRset is not secure, this specification doesn't apply and the implementation should fall back to RFC 5922[RFC5922] behaviour. If any of the responses are "bogus" according to DNSsec validation, the client MUST abort the connection.

6. TLSA record name

For the SIP protocol DANE usage, TLSA records are to be found in accordance with [I-D.ietf-dane-srv]. If the domain example.com's TLS SRV records points to sip01.example.com port 5042 then the corresponding TLSA record will be found using the name _5042._tcp.sip01.example.com.

7. Procedures for DANE-capable SIP implementations

DANE capable SIP implementations follow the procedures above to find a SRV host name and look for a TLSA record. If no TLSA record is found, the client should fall back to RFC 5922 behaviour.

If a TLSA record is found, the client should never fall back to RFC 5922 behaviour. If TLSA-based validation fails, the client MUST abort the connection attempt.

8. X.509 certicate validation

When using DANE-based validation the client validates the SRV hostname with the certificate using RFC 5922 rules. A DANE-capable SIP implementation looks for the SRV hostname in the list of SubjAltName DNSName extension fields. Only if there are no SubjAltName extension fields may the client look in the CN of the X.509 certificate (according to RFC 5922).

If the SRV host name is not found in the certificate, DANE validation fails and the client MUST abort the connection.

Using the SRV host name for validation of a SIP domain identity is an update to RFC 5922

9. Backward Compatibility with RFC 5922

RFC 5922[RFC5922] implementations with no DANE support will be able to connect with the matching described in that document. SIP Servers can use certificates that are compatible with both this specification and RFC5922.

[I-D.ietf-dane-srv] requires use of the TLS Server Name Indication (SNI) extension [RFC6066]. This is not a requirement in this

document, since SIP certificates can support both RFC 5922 style validation and DANE-based validation with the same certificate.

10. Examples on certificate content

This section gives examples on certificate content and how the match a given URI. The X.509 PKIX Subject field CN value is abbreviated as "CN", the SubjectAltName extension DNSName and uniformResourceIdentifier are abbreviated as "SAN-DNS" and "SAN-URI". The certificates are tested with three different clients. A DANE-aware client, a RFC 5922 client with no DANE support and a client that matches the SIP domain with the Common Name in the Subject of the certificate. The last example is not really covered by any SIP-related RFC and should be avoided.

10.1. Example 1: johansson.example.com

- Domain: johansson.example.com
- DNS SRV host for TLS: siphosting.example.net

Certificate content:

- CN: siphosting.example.net
- SAN-URI: -
- SAN-DNS: -
- Matching for DANE-aware SIP clients: Yes
- Matching for only RFC 5922 SIP clients: No
- Matching on CNAME only: No

10.2. Example 2: lundholm.example.com

- Domain: lundholm.example.com
- DNS SRV host for TLS: sipcrew.example.net

Certificate content:

- CN: randomname.example.net
- SAN-URI: sip:lundholm.example.com
- SAN-DNS: lundholm.example.com

- Matching for DANE-aware SIP clients: Yes

- Matching for only RFC 5922 SIP clients: Yes

Note: More examples is coming here.

11. Security Considerations

This document use already published solutions for providing credentials for setting up a secure connection to a SIP server. By depending on secure lookups of DNS NAPTR and SRV records as well as using TLSA records to verify a SIP servers TLS certificate it describes a secure method for making sure that a SIP request for a domain is sent to an authoritative server.

In addition to this document, many security considerations are covered in ID.ietf-dane-srv.

12. IANA Considerations

This document does not require actions by IANA.

13. Acknowledgements

The author wishes to acknowledge Jakob Schlyter for inspiration and .SE for promoting DNSsec and DANE. Victor Dubovn

14. References

14.1. Normative References

[I-D.ietf-dane-srv] Finch, T., Miller, M., and P. Saint-Andre, "Using DNS-Based Authentication of Named Entities (DANE) TLSA records with SRV and MX records.", draft-ietf-dane-srv-03 (work in progress), December 2013.

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

[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.

[RFC3263] Rosenberg, J. and H. Schulzrinne, "Session Initiation Protocol (SIP): Locating SIP Servers", RFC 3263, June 2002.

[RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, March 2005.

[RFC4346] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.1", RFC 4346, April 2006.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, May 2008.

[RFC5922] Gurbani, V., Lawrence, S., and A. Jeffrey, "Domain Certificates in the Session Initiation Protocol (SIP)", RFC 5922, June 2010.

[RFC6066] Eastlake, D., "Transport Layer Security (TLS) Extensions: Extension Definitions", RFC 6066, January 2011.

[RFC6698] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA", RFC 6698, August 2012.

14.2. Informative References

[I-D.ogud-dane-vocabulary] Gudmundsson, O., "Harmonizing how applications specify DANE-like usage", draft-ogud-dane-vocabulary-01 (work in progress), October 2013.

[RFC5589] Sparks, R., Johnston, A., and D. Petrie, "Session Initiation Protocol (SIP) Call Control - Transfer", BCP 149, RFC 5589, June 2009.

Appendix A. Appendix A. Implementation notes

Developers of SIP implementations are strongly encouraged to implement RFC 5922 and this document for secure verification of a SIP domain with a TLS server. This document also encourages implementation of TLS SNI both in client and server implementations. In order to get support of this function, update to new versions of the TLS libraries and make sure that the implementation supports new versions of TLS - TLS 1.1 [RFC4346] and TLS 1.2 [RFC5246].

Implementatinos that do support TLS are encouraged to always start with attempting TLS, even if the URI is a SIP: uri. If there are NAPTR records for the domain and the domain indicates support of TLS, use it. If there are no NAPTR records, start SRV lookup with the _sips._tcp prefix. This way, the SIP network will gradually shift to always using secure and authenticated TLS sessions.

Author's Address

Olle E. Johansson
Edvina AB
Runbovaegen 10
Sollentuna SE-192 48
SE

Email: oej@edvina.net

SIPCORE
Internet-Draft
Intended status: Standards Track
Expires: April 9, 2015

O. Johansson
Edvina AB
G. Salgueiro
Cisco Systems

October 6, 2014

Locating Session Initiation Protocol (SIP) Servers in a Dual-Stack IP Network
draft-johansson-sip-dual-stack-03

Abstract

RFC 3263 defines how a Session Initiation Protocol (SIP) implementation, given a SIP Uniform Resource Identifier (URI), should locate the next hop SIP server using Domain Name System (DNS) procedures. The specification repeatedly states that the implementation should look up IPv4 or IPv6 addresses. This is not a suitable solution and one that can cause severely degraded user experience dual-stack clients, as detailed in the Happy Eyeballs specification. This document specifies amended procedures for dual-stack SIP implementations so that they look up both IPv4 and IPv6 addresses. This way, the SIP implementation can find the preferred network path and protocol with an improved chance of successfully reaching the desired service. This document also clarifies DNS SRV usage for single-stack clients.

Table of Contents

1. Introduction

The core SIP [RFC3261] RFCs were written with support for both IPv4 and IPv6 in mind, but they were not fully equipped to handle highly hybridized environments during this transitional phase of migration from IPv4 to IPv6 networks, where many server and client implementations run on dual stack hosts. In such environments, a dual-stack host will likely suffer greater connection delay, and by extension an inferior user experience, than an IPv4-only host. The need to remedy this diminished performance of dual-stack hosts led to the development of the Happy Eyeballs [RFC6555] algorithm, that has since been implemented in many applications.

RFC 6157[RFC6157] focuses on handling media in a dual-stack network path between two SIP user agents (UAs). This doesn't solve the signalling issues that can occur when trying to find the best network path to the next hop SIP server.

This document updates RFC 3263[RFC3263] procedures so that a dual-stack client SHOULD look up both A and AAAA records in DNS and then select the best way to set up a network flow. The details of how the latter is done is considered out of scope for this document. See the Happy Eyeballs algorithm and implementation and design considerations in RFC 6555[RFC6555] for more information about issues with setting up dual-stack network flows.

2. Terminology and Conventions Used in This Document

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].

RFC 3261 [RFC3261] defines additional terms used in this document that are specific to the SIP domain such as "proxy"; "registrar"; "redirect server"; "user agent server" or "UAS"; "user agent client" or "UAC"; "back-to-back user agent" or "B2BUA"; "dialog"; "transaction"; "server transaction".

This document uses the term "SIP Server" that is defined to include the following SIP entities: user agent server, registrar, redirect server, a SIP proxy in the role of user agent server, and a B2BUA in the role of a user agent server.

This document also uses the following terminology to make clear distinction between SIP entities supporting only IPv4, only IPv6 or supporting both IPv4 and IPv6.

IPv4-only UA/UAC/UAS: An IPv4-only UA/UAC/UAS supports SIP signaling and media only on the IPv4 network. It does not understand IPv6 addresses.

IPv6-only UA/UAC/UAS: An IPv6-only UA/UAC/UAS supports SIP signaling and media only on the IPv6 network. It does not understand IPv4 addresses.

IPv4/IPv6 UA/UAC/UAS: A UA/UAC/UAS that supports SIP signaling and media on both IPv4 and IPv6 networks; such a UA/UAC/UAS is known (and will be referred to in this document) as a "dual-stack" [RFC4213] UA/UAC/UAS.

3. DNS Procedures in a Dual-Stack Network

This specification introduces two normative DNS lookup procedures. These are designed to improve the performace of dual-stack clients in IPv4/IPv6 networks.

3.1. Dual-Stack SIP UA DNS Record Lookup Procedure

Once the transport protocol has been determined, the procedure for discovering an ip address if the TARGET is not a numeric IP address but the port is explicitly stated in the URI, is detailed in Section 4.2 of RFC 3263[RFC3263]. The piece relevant to to this discussion is:

> "If the TARGET was not a numeric IP address, but a port is present in the URI, the client performs an A or AAAA record lookup of the domain name. The result will be a list of IP addresses, each of which can be contacted at the specific port from the URI and transport protocol determined previously."

Section 4.2 of RFC 3263 [RFC3263] also goes on to describe the complete procedure for discovering an ip address if the TARGET is not a numeric IP address, and no port is present in the URI. The piece relevant to to this discussion is:

> "If no SRV records were found, the client performs an A or AAAA record lookup of the domain name. The result will be a list of IP addresses, each of which can be contacted using the transport protocol determined previously, at the default port for that transport. Processing then proceeds as described above for an explicit port once the A or AAAA records have been looked up."

Happy Eyeballs [RFC6555] has proven that looking up the "A or AAAA record" is not an effective practice for dual-stack clients that can add significant connection delay and greatly degrade user experience. A dual-stack client SHOULD perform an A and AAAA record lookup of the domain name and add the respective IPv4/IPv6 addresses to the list of IP addresses to be contacted. This is a normative update to the procedures described in Section 4.2 of RFC 3263 [RFC3263].

3.2. Indicating Address Family Preference in DNS SRV Records

The Happy Eyeballs algorithm is particularly effective when dual-stack client applications have have significant performance differences in their IPv4 or IPv6 network paths. In this common scenario it is often necessary for a dual-stack client to indicate a preference for either IPv4 or IPv6. A service may use DNS SRV records to indicate such a preference for an address family. This way, a server with a high-latency and/or low-capacity IPv4 tunnel may indicate a preference for being contacted using IPv6. A server that wishes to do this can use the lowest SRV priority to publish hostnames that only resolve in IPv6 and the next priority with host names that resolve in both address families.

When indicating address family preference through SRV, a single stack-clients should be prepared to handle SRV record sets that don't resolve into an ip address in the address family used by the client. In such a case, the client should simply proceed to the next priority and try those hostnames.

4. Security Considerations

This document makes two normative changes to the existing DNS procedures used to locate SIP servers in a dual-stack network. One change updates the procedure described in RFC 3263 for dual-stack clients and recommends both A and AAAA record lookups of the domain name. The other update is simply the ability to indicate preference for a particular address family in SRV records. While both of these changes to current procedures are optimizations designed to improve the performance of dual-stack clients, neither introduces any new security considerations. The specific security vulnerabilities, attacks and threat models of the various protocols discussed in this document (SIP, DNS, SRV records, etc.) are well documented in their respective specifications.

5. IANA Considerations

This document does not require any actions by IANA.

6. Acknowledgements

The author would like to acknowledge the support and contribution of the SIP Forum IPv6 Working Group. This document is based on a lot of tests and discussions at SIPit events, organized by the SIP Forum.

This document has benefited from the expertise and review feedback of Dan Wing, Brett Tate, Rifaat Shekh-Yusef and Carl Klatsky.

7. References

7.1. Normative References

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

[RFC3263] Rosenberg, J. and H. Schulzrinne, "Session Initiation Protocol (SIP): Locating SIP Servers", RFC 3263, June 2002.

7.2. Informative References

[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.

[RFC4213] Nordmark, E. and R. Gilligan, "Basic Transition Mechanisms for IPv6 Hosts and Routers", RFC 4213, October 2005.

[RFC6157] Camarillo, G., El Malki, K., and V. Gurbani, "IPv6 Transition in the Session Initiation Protocol (SIP)", RFC 6157, April 2011.

[RFC6555] Wing, D. and A. Yourtchenko, "Happy Eyeballs: Success with Dual-Stack Hosts", RFC 6555, April 2012.

Authors' Addresses

Olle E. Johansson
Edvina AB
Runbovaegen 10
Sollentuna SE-192 48
SE

Email: oej@edvina.net

Gonzalo Salgueiro
Cisco Systems
7200-12 Kit Creek Road
Research Triangle Park, NC 27709
US

Email: gsalguei@cisco.com

SIP Core
Internet-Draft
Updates: 3261 (if approved)
Intended status: Standards Track
Expires: March 22, 2018

R. Shekh-Yusef
Avaya

September 18, 2017

The Session Initiation Protocol (SIP) Digest Authentication Scheme
draft-yusef-sipcore-digest-scheme-06

Abstract

This document updates the Digest Access Authentication scheme used by the Session Initiation Protocol (SIP) to add support for SHA2 digest algorithms to replace the MD5 algorithm.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on March 22, 2018.

Copyright Notice

Table of Contents

1. Introduction

The SIP protocol [RFC3261] uses the same mechanism used by the HTTP protocol for authenticating users, which is a simple challenge-response authentication mechanism that allows a server to challenge a client request and allows a client to provide authentication information in response to that challenge.

The SIP protocol uses the Digest Authentication scheme that is used with the HTTP authentication mechanism, which by default uses MD5 as the default algorithm.

The HTTP Digest Access Authentication [RFC7616] document defines the Digest Authentication scheme and defines a few algorithms that could be used with the Digest Authentication scheme, and establishes a registry for these algorithms to allow for additional algorithms to be added in the future.

This document updates the Digest Access Authentication scheme used by SIP to add support for SHA2 digest algorithms to replace the MD5 algorithm.

1.1. 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 [RFC2119].

2. The SIP Digest Authentication Scheme

This section describes the modifications to the operation of the Digest mechanism as specified in [RFC3261] in order to support the SHA- 256 and SHA-512/256 algorithms as described in [RFC7616], and also to require support for the "qop" option."

2.1. Hash Algorithms

The Digest scheme has an ’algorithm’ parameter that specifies the algorithm to be used to compute the digest of the response. The IANA registry named "HTTP Digest Hash Algorithms" specifies the algorithms that correspond to ’algorithm’ values, and specifies a priority for each algorithm.

[RFC3261] specifies only one algorithm, MD5, which is used by default. This document extends [RFC3261] to allow use of any registered algorithm.

The priority of the algorithm defines its usage preference. UAs SHOULD prefer algorithms with higher priorities.

Note that [RFC7616] defines a -sess variant for each algorithm; the -sess variants are not used with SIP.

2.2. Representation of Digest Values

The size of the digest depends on the algorithm used. The bits in the digest are converted from the most significant to the least significant bit, four bits at a time to the ASCII representation as follows. Each four bits is represented by its familiar hexadecimal notation from the characters 0123456789abcdef, that is binary 0000 is represented by the character ’0’, 0001 by ’1’ and so on up to the representation of 1111 as ’f’. If the MD5 algorithm is used to calculate the digest, then the digest will be represented as 32

hexadecimal characters, SHA-256 and SHA-512/256 by 64 hexadecimal characters.

2.3. The Authenticate Response Header

When a UAS receives a request from a UAC, and an acceptable Authorization header is not sent, the UAS can challenge the originator to provide credentials by rejecting the request with a 401/407 status code with the WWW-Authenticate/Proxy-Authenticate header field. The UAS MAY include multiple WWW-Authenticate/Proxy-Authenticate headers to allow the UAS to utilize the best available algorithm supported by the client.

If the UAS challenges with multiple WWW-Authenticate/Proxy-Authenticate headers with the same realm, then each one of these headers MUST use a different digest algorithm. The UAS MUST add these headers to the response in the order that it would prefer to see them used, starting with the most preferred algorithm at the top, followed by the less preferred algorithms.

2.4. The Authorization Request Header

When the UAC receives a response with multiple headers with the same realm it SHOULD use the topmost header that it supports, unless a local policy dictates otherwise. The client MUST ignore any challenge it does not understand.

When the UAC receives a 401 response with multiple WWW-Authenticate headers with different realms it SHOULD retry and include an Authorization header containing credentials that match the topmost header of any one of the realms.

If the UAC cannot respond to any of the challenges in the response, then it should abandon attempts to send the request; e.g., if the UAC does not have credentials for any of the realms.

2.5. Forking

Section 22.3 of [RFC3261] discusses the operation of the proxy-to-user authentication, which describes the operation of the proxy when it forks a request. This section introduces some clarification to that operation.

If a request is forked, various proxy servers and/or UAs may wish to challenge the UAC. In this case, the forking proxy server is

responsible for aggregating these challenges into a single response. Each WWW-Authenticate and Proxy-Authenticate value received in responses to the forked request MUST be placed into the single response that is sent by the forking proxy to the UA.

When the forking proxy places multiple WWW-Authenticate and Proxy-Authenticate header fields from one received response into the single response it MUST maintain the order of these header fields. The ordering of the header field values from the various proxies is not significant.

2.6. HTTP Modifications

This section describes the modifications and clarifications required to apply the HTTP Digest authentication scheme to SIP. The SIP scheme usage is similar to that for HTTP. The changes specified here are mostly copied from section 22.4 of [RFC3261] with few changes.

SIP clients and servers MUST NOT accept or request Basic authentication.

The rules for Digest authentication follow those defined in HTTP, with "HTTP/1.1" replaced by "SIP/2.0" in addition to the following differences:

1. The URI included in the challenge has the following BNF:

   URI = Request-URI

2. The ’uri’ parameter of the Authorization header field MUST be enclosed in quotation marks.

3. The BNF for digest-uri-value is:

   digest-uri-value = Request-URI

4. The example procedure for choosing a nonce based on Etag does not work for SIP.

5. The text in [RFC7234] regarding cache operation does not apply to SIP.

6. [RFC7616] requires that a server check that the URI in the request line and the URI included in the Authorization header field point to the same resource. In a SIP context, these two URIs may refer to different users, due to forwarding at some proxy. Therefore, in SIP, a server MAY check that the Request-URI in the

Authorization header field value corresponds to a user for whom the server is willing to accept forwarded or direct requests, but it is not necessarily a failure if the two fields are not equivalent.

7. As a clarification to the calculation of the A2 value for message integrity assurance in the Digest authentication scheme, implementers should assume, when the entity-body is empty (that is, when SIP messages have no body) that the hash of the entity-body resolves to the hash of an empty string:

```
H(entity-body) = <algorithm>("")
```

For example, when the chosen algorithm is SHA-256, then:

```
H(entity-body) = SHA-256("") =
"e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855"
```

8. Servers MUST be able to properly handle "qop" parameter received in an authorization header field, and clients MUST be able to properly handle "qop" parameter received in WWW-Authenticate and Proxy-Authenticate header fields. Servers MUST always send a "qop" parameter in WWW-Authenticate and Proxy-Authenticate header field values, and clients MUST send the "qop" parameter in any resulting authorization header field.

The usage of the Authentication-Info header field continue to be allowed, since it provides integrity checks over the bodies and provides mutual authentication.

3. Augmented BNF for the SIP Protocol

This document updates the Augmented BNF for the SIP Protocol as follows.

It extends the request-digest as follows to allow for different digest sizes:

```
request-digest = LDQUOT *LHEX RDQUOT
```

The number of hex digits must be specified by the specification of the algorithm used.

It extends the algorithm parameter as follows to allow for SHA2 algorithms to be used:

```
algorithm = "algorithm" EQUAL ( "MD5" / "SHA-512-256" / "SHA-256"
/ token )
```

4. Security Considerations

This specification adds new secure algorithms to be used to with the Digest mechanism to authenticate users, but leaves the broken MD5 algorithm for backward compatibility.

This opens the system to the potential of a downgrade attack by man-in-the-middle. The most effective way of dealing with this type of attack is to remove the support for backward compatibility.

See section 5 of [RFC7616] for a detailed security discussion of the Digest scheme.

5. IANA Considerations

[RFC7616] defines an IANA registry named "Hash Algorithms for HTTP Digest Authentication" to simplify the introduction of new algorithms in the future. This document will use the algorithms defined in that registry.

6. Acknowledgments

The author would like to thank the following individuals for their careful reviews, comments, and suggestions: Paul Kyzivat, Olle Johansson, Dale Worley, Michael Procter, Inaki Baz Castillo, and Tolga Asveren.

7. Normative References

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

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, H., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

[RFC7234] Fielding, R., Nottingham, M., and J. Reschke, "Hypertext Transfer Protocol (HTTP/1.1): Caching", RFC 7234, June 2014.

[RFC7616] Shekh-Yusef, R., Ahrens, D., and S. Bremer, "HTTP Digest Access Authentication", RFC 7616, September 2015.

Author's Address

Rifaat Shekh-Yusef
Avaya
250 Sidney Street
Belleville, Ontario
Canada

Phone: +1-613-967-5176
EMail: rifaat.ietf@gmail.com