Using Secure DNS to Associate Certificates with Domain Names for S/MIME

Abstract

   This document describes how to use secure DNS to associate an S/MIME
   user's certificate with the intended domain name, similar to the way
   that DNS-Based Authentication of Named Entities (DANE), RFC 6698,
   does for TLS.

Status of This Memo

   This document is not an Internet Standards Track specification; it is
   published for examination, experimental implementation, and
   evaluation.

   This document defines an Experimental Protocol for the Internet
   community. This document is a product of the Internet Engineering
   Task Force (IETF). It represents the consensus of the IETF
   community. It has received public review and has been approved for
   publication by the Internet Engineering Steering Group (IESG). Not
   all documents approved by the IESG are a candidate for any level of
   Internet Standard; see Section 2 of RFC 7841.

   Information about the current status of this document, any errata,
   and how to provide feedback on it may be obtained at
   http://www.rfc-editor.org/info/rfc8162.

Table of Contents

1. Introduction

   S/MIME [RFC5751] messages often contain a certificate (some messages
   contain more than one certificate). These certificates assist in
   authenticating the sender of the message and can be used for
   encrypting messages that will be sent in reply. In order for the
   S/MIME receiver to authenticate that a message is from the sender
   identified in the message, the receiver's Mail User Agent (MUA) must
   validate that this certificate is associated with the purported
   sender. Currently, the MUA must trust a trust anchor upon which the
   sender's certificate is rooted and must successfully validate the
   certificate. There are other requirements on the MUA, such as
   associating the identity in the certificate with that of the message,
   that are out of scope for this document.

   Some people want to authenticate the association of the sender's
   certificate with the sender without trusting a configured trust
   anchor. Others to want mitigate the difficulty of finding
   certificates from outside the enterprise. Given that the DNS
   administrator for a domain name is authorized to give identifying
   information about the zone, it makes sense to allow that
   administrator to also make an authoritative binding between email
   messages purporting to come from the domain name and a certificate
   that might be used by someone authorized to send mail from those
   servers. The easiest way to do this is to use the DNS.

   This document describes a mechanism for associating a user's
   certificate with the domain that is similar to that described in DANE
   itself [RFC6698], as updated by [RFC7218] and [RFC7671]; it is also
   similar to the mechanism given in [RFC7929] for OpenPGP. Most of the
   operational and security considerations for using the mechanism in
   this document are described in RFC 6698 and are not described here at
   all. Only the major differences between this mechanism and those
   used in RFC 6698 are described here. Thus, the reader must be
   familiar with RFC 6698 before reading this document.

1.1. 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.

   This document also makes use of standard PKIX, DNSSEC, and S/MIME
   terminology. See PKIX [RFC5280], DNSSEC [RFC4033] [RFC4034]
   [RFC4035], and S/MIME [RFC5751] for these terms.

1.2. Experiment Goal

   This specification is one experiment in improving access to public
   keys for end-to-end email security. There are a range of ways in
   which this can reasonably be done for OpenPGP or S/MIME, for example,
   using the DNS, SMTP, or HTTP. Proposals for each of these have been
   made with various levels of support in terms of implementation and
   deployment. For each such experiment, specifications such as this
   will enable experiments to be carried out that may succeed or that
   may uncover technical or other impediments to large- or small-scale
   deployments. The IETF encourages those implementing and deploying
   such experiments to publicly document their experiences so that
   future specifications in this space can benefit.

   This document defines an RRtype whose use is Experimental. The goal
   of the experiment is to see whether encrypted email usage will
   increase if an automated discovery method is available to Mail
   Transfer Agents (MTAs) and if MUAs help the end user with email
   encryption key management.

   It is unclear if this RRtype will scale to some of the larger email
   service deployments. Concerns have been raised about the size of the
   SMIMEA record and the size of the resulting DNS zone files. This
   experiment hopefully will give the IETF some insight into whether or
   not this is a problem.

If the experiment is successful, it is expected that the findings of
the experiment will result in an updated document for Standards Track
approval.

2. The SMIMEA Resource Record

The SMIMEA DNS resource record (RR) is used to associate an end
entity certificate or public key with the associated email address,
thus forming a "SMIMEA certificate association". The semantics of
how the SMIMEA resource record is interpreted are given later in this
document. Note that the information returned in the SMIMEA record
might be for the end entity certificate, or it might be for the trust
anchor or an intermediate certificate. This mechanism is similar to
the one given in [RFC7929] for OpenPGP.

The type value for the SMIMEA RRtype is defined in Section 8. The
SMIMEA resource record is class independent.

The SMIMEA wire format and presentation format are the same as for
the TLSA record as described in Section 2.1 of [RFC6698]. The
certificate usage field, the selector field, and the matching type
field have the same format; the semantics are also the same except
where RFC 6698 talks about TLS as the target protocol for the
certificate information.

3. Location of the SMIMEA Record

The DNS does not allow the use of all characters that are supported
in the "local-part" of email addresses as defined in [RFC5322] and
[RFC6530]. Therefore, email addresses are mapped into DNS using the
following method:

1. The "left-hand side" of the email address, called the "local-
    part" in both the mail message format definition [RFC5322] and in
    the specification for internationalized email [RFC6530]) is
    encoded in UTF-8 (or its subset ASCII). If the local-part is
    written in another charset, it MUST be converted to UTF-8.

2. The local-part is first canonicalized using the following rules.
    If the local-part is unquoted, any comments and/or folding
    whitespace (CFWS) around dots (".") is removed. Any enclosing
    double quotes are removed. Any literal quoting is removed.

3. If the local-part contains any non-ASCII characters, it SHOULD be
    normalized using the Unicode Normalization Form C from [UNICODE].
    Recommended normalization rules can be found in Section 10.1 of
    [RFC6530].

   4. The local-part is hashed using the SHA2-256 [RFC5754] algorithm,
       with the hash truncated to 28 octets and represented in its
       hexadecimal representation, to become the left-most label in the
       prepared domain name.

   5. The string "_smimecert" becomes the second left-most label in the
       prepared domain name.

   6. The domain name (the "right-hand side" of the email address,
       called the "domain" in [RFC5322]) is appended to the result of
       step 5 to complete the prepared domain name.

   For example, to request an SMIMEA resource record for a user whose
   email address is "hugh@example.com", an SMIMEA query would be placed
   for the following QNAME: "c93f1e400f26708f98cb19d936620da35eec8f72e57
   f9eec01c1afd6._smimecert.example.com".

4. Email Address Variants and Internationalization Considerations

   Mail systems usually handle variant forms of local-parts. The most
   common variants are upper and lower case, often automatically
   corrected when a name is recognized as such. Other variants include
   systems that ignore "noise" characters such as dots, so that local-
   parts 'johnsmith' and 'John.Smith' would be equivalent. Many systems
   allow "extensions" such as 'john-ext' or 'mary+ext' where 'john' or
   'mary' is treated as the effective local-part, and the 'ext' is
   passed to the recipient for further handling. This can complicate
   finding the SMIMEA record associated with the dynamically created
   email address.

   [RFC5321] and its predecessors have always made it clear that only
   the recipient MTA is allowed to interpret the local-part of an
   address. Therefore, sending MUAs and MTAs supporting this
   specification MUST NOT perform any kind of mapping rules based on the
   email address. In order to improve the chances of finding SMIMEA
   resource records for a particular local-part, domains that allow
   variant forms (such as treating local-parts as case-insensitive)
   might publish SMIMEA resource records for all variants of local-
   parts, might publish variants on first use (for example, a webmail
   provider that also controls DNS for a domain can publish variants as
   used by owner of a particular local-part), or might just publish
   SMIMEA resource records for the most common variants.

   Section 3 above defines how the local-part is used to determine the
   location in which one looks for an SMIMEA resource record. Given the
   variety of local-parts seen in email, designing a good experiment for
   this is difficult as a) some current implementations are known to
   lowercase at least US-ASCII local-parts, b) we know from (many) other

situations that any strategy based on guessing and making multiple
DNS queries is not going to achieve consensus for good reasons, and
c) the underlying issues are just hard -- see Section 10.1 of
[RFC6530] for discussion of just some of the issues that would need
to be tackled to fully address this problem.

However, while this specification is not the place to try to address
these issues with local-parts, doing so is also not required to
determine the outcome of this experiment. If this experiment
succeeds, then further work on email addresses with non-ASCII local-
parts will be needed, and that would be better based on the findings
from this experiment, rather than doing nothing or starting this
experiment based on a speculative approach to what is a very complex
topic.

5. Mandatory-to-Implement Features

   S/MIME MUAs conforming to this specification MUST be able to
   correctly interpret SMIMEA records with certificate usages 0, 1, 2,
   and 3. S/MIME MUAs conforming to this specification MUST be able to
   compare a certificate association with a certificate offered by
   another S/MIME MUA using selector types 0 and 1, and matching type 0
   (no hash used) and matching type 1 (SHA-256), and SHOULD be able to
   make such comparisons with matching type 2 (SHA-512).

   S/MIME MUAs conforming to this specification MUST be able to
   interpret any S/MIME capabilities (defined in [RFC4262]) in any
   certificates that it receives through SMIMEA records.

6. Application Use of S/MIME Certificate Associations

   The SMIMEA record allows an application or service to obtain an
   S/MIME certificate or public key and use it for verifying a digital
   signature or encrypting a message to the public key. The DNS answer
   MUST pass DNSSEC validation; if DNSSEC validation reaches any state
   other than "Secure" (as specified in [RFC4035]), the DNSSEC
   validation MUST be treated as a failure.

   If no S/MIME certificates are known for an email address, an SMIMEA
   DNS lookup MAY be performed to seek the certificate or public key
   that corresponds to that email address. This can then be used to
   verify a received signed message or can be used to send out an
   encrypted email message. An application whose attempt fails to
   retrieve a DNSSEC-verified SMIMEA resource record from the DNS should
   remember that failed attempt and not retry it for some time. This
   will avoid sending out a DNS request for each email message the
   application is sending out; such DNS requests constitute a privacy
   leak.

7. Certificate Size and DNS

   Due to the expected size of the SMIMEA record, applications SHOULD
   use TCP -- not UDP -- to perform queries for the SMIMEA resource
   record.

   Although the reliability of the transport of large DNS resource
   records has improved in the last years, it is still recommended to
   keep the DNS records as small as possible without sacrificing the
   security properties of the public key. The algorithm type and key
   size of certificates should not be modified to accommodate this
   section.

8. IANA Considerations

   This document uses a new DNS RRtype, SMIMEA, whose value (53) was
   allocated by IANA from the "Resource Record (RR) TYPEs" subregistry
   of the "Domain Name System (DNS) Parameters" registry.

9. Security Considerations

   Client treatment of any information included in the trust anchor is a
   matter of local policy. This specification does not mandate that
   such information be inspected or validated by the domain name
   administrator.

   DNSSEC does not protect the queries from pervasive monitoring as
   defined in [RFC7258]. Since DNS queries are currently mostly
   unencrypted, a query to look up a target SMIMEA record could reveal
   that a user using the (monitored) recursive DNS server is attempting
   to send encrypted email to a target.

   Various components could be responsible for encrypting an email
   message to a target recipient. It could be done by the sender's MUA,
   an MUA plugin, or the sender's MTA. Each of these have their own
   characteristics. An MUA can ask the user to make a decision before
   continuing. The MUA can either accept or refuse a message. The MTA
   might deliver the message as is or encrypt the message before
   delivering. Each of these components should attempt to encrypt an
   unencrypted outgoing message whenever possible.

   In theory, two different local-parts could hash to the same value.
   This document assumes that such a hash collision has a negligible
   chance of happening.

   If an obtained S/MIME certificate is revoked or expired, that
   certificate MUST NOT be used, even if that would result in sending a
   message in plaintext.

Anyone who can obtain a DNSSEC private key of a domain name via
coercion, theft, or brute-force calculations can replace any SMIMEA
record in that zone and all of the delegated child zones. Any future
messages encrypted with the malicious SMIMEA key could then be read.
Therefore, a certificate or key obtained from a DNSSEC-validated
SMIMEA record can only be trusted as much as the DNS domain can be
trusted.

Organizations that are required to be able to read everyone's
encrypted email should publish the escrow key as the SMIMEA record.
Mail servers of such organizations MAY optionally re-encrypt the
message to the individual's S/MIME key. This case can be considered
a special case of the key-replacement attack described above.

9.1. Response Size

To prevent amplification attacks, an Authoritative DNS server MAY
wish to prevent returning SMIMEA records over UDP unless the source
IP address has been confirmed with DNS Cookies [RFC7873]. If a query
is received via UDP without source IP address verification, the
server MUST NOT return REFUSED but answer the query with an empty
answer section and the truncation flag set ("TC=1").

9.2. Email Address Information Leak

The hashing of the local-part in this document is not a security
feature. Publishing SMIMEA records will create a list of hashes of
valid email addresses, which could simplify obtaining a list of valid
email addresses for a particular domain. It is desirable to not ease
the harvesting of email addresses where possible.

The domain name part of the email address is not used as part of the
hash so that hashes can be used in multiple zones deployed using
DNAME [RFC6672]. This makes it slightly easier and cheaper to brute-
force the SHA2-256 hashes into common and short local-parts, as
single rainbow tables [Rainbow] can be reused across domains. This
can be somewhat countered by using NSEC3 [RFC5155].

DNS zones that are signed with DNSSEC using NSEC [RFC4033] for denial
of existence are susceptible to zone walking, a mechanism that allows
someone to enumerate all the SMIMEA hashes in a zone. This can be
used in combination with previously hashed common or short local-
parts (in rainbow tables) to deduce valid email addresses. DNSSEC-
signed zones using NSEC3 for denial of existence instead of NSEC are
significantly harder to brute-force after performing a zone walk.

10. References

10.1. 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,
              <http://www.rfc-editor.org/info/rfc2119>.

   [RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S.
              Rose, "DNS Security Introduction and Requirements",
              RFC 4033, DOI 10.17487/RFC4033, March 2005,
              <http://www.rfc-editor.org/info/rfc4033>.

   [RFC4034] Arends, R., Austein, R., Larson, M., Massey, D., and S.
              Rose, "Resource Records for the DNS Security Extensions",
              RFC 4034, DOI 10.17487/RFC4034, March 2005,
              <http://www.rfc-editor.org/info/rfc4034>.

   [RFC4035] Arends, R., Austein, R., Larson, M., Massey, D., and S.
              Rose, "Protocol Modifications for the DNS Security
              Extensions", RFC 4035, DOI 10.17487/RFC4035, March 2005,
              <http://www.rfc-editor.org/info/rfc4035>.

   [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, DOI 10.17487/RFC5280, May 2008,
              <http://www.rfc-editor.org/info/rfc5280>.

   [RFC5751] Ramsdell, B. and S. Turner, "Secure/Multipurpose Internet
              Mail Extensions (S/MIME) Version 3.2 Message
              Specification", RFC 5751, DOI 10.17487/RFC5751, January
              2010, <http://www.rfc-editor.org/info/rfc5751>.

   [RFC5754] Turner, S., "Using SHA2 Algorithms with Cryptographic
              Message Syntax", RFC 5754, DOI 10.17487/RFC5754, January
              2010, <http://www.rfc-editor.org/info/rfc5754>.

   [RFC6698] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication
              of Named Entities (DANE) Transport Layer Security (TLS)
              Protocol: TLSA", RFC 6698, DOI 10.17487/RFC6698, August
              2012, <http://www.rfc-editor.org/info/rfc6698>.

   [RFC7671] Dukhovni, V. and W. Hardaker, "The DNS-Based
              Authentication of Named Entities (DANE) Protocol: Updates
              and Operational Guidance", RFC 7671, DOI 10.17487/RFC7671,
              October 2015, <http://www.rfc-editor.org/info/rfc7671>.

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

10.2. Informative References

   [Rainbow] Oechslin, P., "Making a Faster Cryptanalytic Time-Memory
              Trade-Off", DOI 10.1007/978-3-540-45146-4_36, 2003,
              <http://www.iacr.org/cryptodb/archive/2003/
              CRYPTO/1615/1615.ps>.

   [RFC4262] Santesson, S., "X.509 Certificate Extension for Secure/
              Multipurpose Internet Mail Extensions (S/MIME)
              Capabilities", RFC 4262, DOI 10.17487/RFC4262, December
              2005, <http://www.rfc-editor.org/info/rfc4262>.

   [RFC5155] Laurie, B., Sisson, G., Arends, R., and D. Blacka, "DNS
              Security (DNSSEC) Hashed Authenticated Denial of
              Existence", RFC 5155, DOI 10.17487/RFC5155, March 2008,
              <http://www.rfc-editor.org/info/rfc5155>.

   [RFC5321] Klensin, J., "Simple Mail Transfer Protocol", RFC 5321,
              DOI 10.17487/RFC5321, October 2008,
              <http://www.rfc-editor.org/info/rfc5321>.

   [RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322,
              DOI 10.17487/RFC5322, October 2008,
              <http://www.rfc-editor.org/info/rfc5322>.

   [RFC6530] Klensin, J. and Y. Ko, "Overview and Framework for
              Internationalized Email", RFC 6530, DOI 10.17487/RFC6530,
              February 2012, <http://www.rfc-editor.org/info/rfc6530>.

   [RFC6672] Rose, S. and W. Wijngaards, "DNAME Redirection in the
              DNS", RFC 6672, DOI 10.17487/RFC6672, June 2012,
              <http://www.rfc-editor.org/info/rfc6672>.

   [RFC7218] Gudmundsson, O., "Adding Acronyms to Simplify
              Conversations about DNS-Based Authentication of Named
              Entities (DANE)", RFC 7218, DOI 10.17487/RFC7218, April
              2014, <http://www.rfc-editor.org/info/rfc7218>.

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

   [RFC7873] Eastlake 3rd, D. and M. Andrews, "Domain Name System (DNS)
              Cookies", RFC 7873, DOI 10.17487/RFC7873, May 2016,
              <http://www.rfc-editor.org/info/rfc7873>.

   [RFC7929] Wouters, P., "DNS-Based Authentication of Named Entities
              (DANE) Bindings for OpenPGP", RFC 7929,
              DOI 10.17487/RFC7929, August 2016,
              <http://www.rfc-editor.org/info/rfc7929>.

   [UNICODE] The Unicode Consortium, "The Unicode Standard",
              <http://www.unicode.org/versions/latest/>.

Acknowledgements

   A great deal of material in this document is copied from [RFC7929].
   That material was created by Paul Wouters and other participants in
   the IETF DANE WG.

   Brian Dickson, Stephen Farrell, Miek Gieben, Martin Pels, and Jim
   Schaad contributed technical ideas and support to this document.

Authors' Addresses

   Paul Hoffman
   ICANN

   Email: paul.hoffman@icann.org

   Jakob Schlyter
   Kirei AB

   Email: jakob@kirei.se