S/MIME Version 3 Certificate Handling

Status of this Memo

   This document specifies an Internet standards track protocol for the
   Internet community, and requests discussion and suggestions for
   improvements. Please refer to the current edition of the "Internet
   Official Protocol Standards" (STD 1) for the standardization state
   and status of this protocol. Distribution of this memo is unlimited.

Copyright Notice

1. Overview

   S/MIME (Secure/Multipurpose Internet Mail Extensions), described in
   [SMIME-MSG], provides a method to send and receive secure MIME
   messages. Before using a public key to provide security services, the
   S/MIME agent MUST certify that the public key is valid. S/MIME agents
   MUST use PKIX certificates to validate public keys as described in
   the Internet X.509 Public Key Infrastructure (PKIX) Certificate and
   CRL Profile [KEYM]. S/MIME agents MUST meet the certificate
   processing requirements documented in this document in addition to
   those stated in [KEYM].

   This specification is compatible with the Cryptographic Message
   Syntax [CMS] in that it uses the data types defined by CMS. It also
   inherits all the varieties of architectures for certificate-based key
   management supported by CMS.

1.1 Definitions

   For the purposes of this memo, the following definitions apply.

   ASN.1: Abstract Syntax Notation One, as defined in ITU-T X.680-689.

   Attribute Certificate (AC): An X.509 AC is a separate structure from
   a subject's public key X.509 Certificate. A subject may have
   multiple X.509 ACs associated with each of its public key X.509
   Certificates. Each X.509 AC binds one or more Attributes with one of
   the subject's public key X.509 Certificates. The X.509 AC syntax is
   defined in [X.509]

BER: Basic Encoding Rules for ASN.1, as defined in ITU-T X.690.

Certificate: A type that binds an entity's distinguished name to a
public key with a digital signature. This type is defined in the
Internet X.509 Public Key Infrastructure (PKIX) Certificate and CRL
Profile [KEYM]. This type also contains the distinguished name of the
certificate issuer (the signer), an issuer-specific serial number,
the issuer's signature algorithm identifier, a validity period, and
extensions also defined in that document.

Certificate Revocation List (CRL): A type that contains information
about certificates whose validity an issuer has prematurely revoked.
The information consists of an issuer name, the time of issue, the
next scheduled time of issue, a list of certificate serial numbers
and their associated revocation times, and extensions as defined in
[KEYM]. The CRL is signed by the issuer. The type intended by this
specification is the one defined in [KEYM].

DER: Distinguished Encoding Rules for ASN.1, as defined in ITU-T
X.690.

Receiving agent: software that interprets and processes S/MIME CMS
objects, MIME body parts that contain CMS objects, or both.

Sending agent: software that creates S/MIME CMS objects, MIME body
parts that contain CMS objects, or both.

S/MIME agent: user software that is a receiving agent, a sending
agent, or both.

1.2 Compatibility with Prior Practice of S/MIME

S/MIME version 3 agents should attempt to have the greatest
interoperability possible with S/MIME version 2 agents. S/MIME
version 2 is described in RFC 2311 through RFC 2315, inclusive. RFC
2311 also has historical information about the development of S/MIME.

1.3 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 [MUSTSHOULD].

2. CMS Options

   The CMS message format allows for a wide variety of options in
   content and algorithm support. This section puts forth a number of
   support requirements and recommendations in order to achieve a base
   level of interoperability among all S/MIME implementations. Most of
   the CMS format for S/MIME messages is defined in [SMIME-MSG].

2.1 CertificateRevocationLists

   Receiving agents MUST support the Certificate Revocation List (CRL)
   format defined in [KEYM]. If sending agents include CRLs in outgoing
   messages, the CRL format defined in [KEYM] MUST be used.

   All agents MUST be capable of performing revocation checks using CRLs
   as specified in [KEYM]. All agents MUST perform revocation status
   checking in accordance with [KEYM]. Receiving agents MUST recognize
   CRLs in received S/MIME messages.

   Agents SHOULD store CRLs received in messages for use in processing
   later messages.

   Agents MUST handle multiple valid Certificate Authority (CA)
   certificates containing the same subject name and the same public
   keys but with overlapping validity intervals.

2.2 CertificateChoices

   Receiving agents MUST support PKIX v1 and PKIX v3 certificates. See
   [KEYM] for details about the profile for certificate formats. End
   entity certificates MAY include an Internet mail address, as
   described in section 3.1.

   Receiving agents SHOULD support X.509 attribute certificates.

2.2.1 Historical Note About CMS Certificates

   The CMS message format supports a choice of certificate formats for
   public key content types: PKIX, PKCS #6 Extended Certificates and
   X.509 Attribute Certificates. The PKCS #6 format is not in widespread
   use. In addition, PKIX certificate extensions address much of the
   same functionality and flexibility as was intended in the PKCS #6.
   Thus, sending and receiving agents MUST NOT use PKCS #6 extended
   certificates.

2.3 CertificateSet

   Receiving agents MUST be able to handle an arbitrary number of
   certificates of arbitrary relationship to the message sender and to
   each other in arbitrary order. In many cases, the certificates
   included in a signed message may represent a chain of certification
   from the sender to a particular root. There may be, however,
   situations where the certificates in a signed message may be
   unrelated and included for convenience.

   Sending agents SHOULD include any certificates for the user's public
   key(s) and associated issuer certificates. This increases the
   likelihood that the intended recipient can establish trust in the
   originator's public key(s). This is especially important when sending
   a message to recipients that may not have access to the sender's
   public key through any other means or when sending a signed message
   to a new recipient. The inclusion of certificates in outgoing
   messages can be omitted if S/MIME objects are sent within a group of
   correspondents that has established access to each other's
   certificates by some other means such as a shared directory or manual
   certificate distribution. Receiving S/MIME agents SHOULD be able to
   handle messages without certificates using a database or directory
   lookup scheme.

   A sending agent SHOULD include at least one chain of certificates up
   to, but not including, a Certificate Authority (CA) that it believes
   that the recipient may trust as authoritative. A receiving agent
   SHOULD be able to handle an arbitrarily large number of certificates
   and chains.

   Agents MAY send CA certificates, that is, certificates that are
   self-signed and can be considered the "root" of other chains. Note
   that receiving agents SHOULD NOT simply trust any self-signed
   certificates as valid CAs, but SHOULD use some other mechanism to
   determine if this is a CA that should be trusted. Also note that in
   the case of DSA certificates the parameters may be located in the
   root certificate. This would require that the recipient possess the
   root certificate in order to perform a signature verification, and is
   a valid example of a case where transmitting the root certificate may
   be required.

   Receiving agents MUST support chaining based on the distinguished
   name fields. Other methods of building certificate chains may be
   supported but are not currently recommended.

Receiving agents SHOULD support the decoding of X.509 attribute
certificates included in CMS objects. All other issues regarding the
generation and use of X.509 attribute certificates are outside of the
scope of this specification.

3. Using Distinguished Names for Internet Mail

End-entity certificates MAY contain an Internet mail address as
described in [RFC-822]. The address must be an "addr-spec" as defined
in Section 6.1 of that specification. The email address SHOULD be in
the subjectAltName extension, and SHOULD NOT be in the subject
distinguished name.

Receiving agents MUST recognize email addresses in the subjectAltName
field. Receiving agents MUST recognize email addresses in the
Distinguished Name field in the PKCS #9 emailAddress attribute.

Sending agents SHOULD make the address in the From or Sender header
in a mail message match an Internet mail address in the signer's
certificate. Receiving agents MUST check that the address in the From
or Sender header of a mail message matches an Internet mail address
in the signer's certificate, if mail addresses are present in the
certificate. A receiving agent SHOULD provide some explicit alternate
processing of the message if this comparison fails, which may be to
display a message that shows the recipient the addresses in the
certificate or other certificate details.

All subject and issuer names MUST be populated (i.e. not an empty
SEQUENCE) in S/MIME-compliant PKIX certificates, except that the
subject DN in a user's (i.e. end-entity) certificate MAY be an empty
SEQUENCE in which case the subjectAltName extension will include the
subject's identifier and MUST be marked as critical.

4. Certificate Processing

A receiving agent needs to provide some certificate retrieval
mechanism in order to gain access to certificates for recipients of
digital envelopes. There are many ways to implement certificate
retrieval mechanisms. X.500 directory service is an excellent example
of a certificate retrieval-only mechanism that is compatible with
classic X.500 Distinguished Names. The PKIX Working Group is
investigating other mechanisms such as directory servers. Another
method under consideration by the IETF is to provide certificate
retrieval services as part of the existing Domain Name System (DNS).
Until such mechanisms are widely used, their utility may be limited
by the small number of correspondent's certificates that can be

retrieved. At a minimum, for initial S/MIME deployment, a user agent
could automatically generate a message to an intended recipient
requesting that recipient's certificate in a signed return message.

Receiving and sending agents SHOULD also provide a mechanism to allow
a user to "store and protect" certificates for correspondents in such
a way so as to guarantee their later retrieval. In many environments,
it may be desirable to link the certificate retrieval/storage
mechanisms together in some sort of certificate database. In its
simplest form, a certificate database would be local to a particular
user and would function in a similar way as a "address book" that
stores a user's frequent correspondents. In this way, the certificate
retrieval mechanism would be limited to the certificates that a user
has stored (presumably from incoming messages). A comprehensive
certificate retrieval/storage solution may combine two or more
mechanisms to allow the greatest flexibility and utility to the user.
For instance, a secure Internet mail agent may resort to checking a
centralized certificate retrieval mechanism for a certificate if it
can not be found in a user's local certificate storage/retrieval
database.

Receiving and sending agents SHOULD provide a mechanism for the
import and export of certificates, using a CMS certs-only message.
This allows for import and export of full certificate chains as
opposed to just a single certificate. This is described in [SMIME-
MSG].

4.1 Certificate Revocation Lists

In general, it is always better to get the latest CRL information
from a CA than to get information stored away from incoming messages.
A receiving agent SHOULD have access to some certificate-revocation
list (CRL) retrieval mechanism in order to gain access to
certificate-revocation information when validating certificate
chains. A receiving or sending agent SHOULD also provide a mechanism
to allow a user to store incoming certificate-revocation information
for correspondents in such a way so as to guarantee its later
retrieval.

Receiving and sending agents SHOULD retrieve and utilize CRL
information every time a certificate is verified as part of a
certificate chain validation even if the certificate was already
verified in the past. However, in many instances (such as off-line
verification) access to the latest CRL information may be difficult
or impossible. The use of CRL information, therefore, may be dictated
by the value of the information that is protected. The value of the

CRL information in a particular context is beyond the scope of this
memo but may be governed by the policies associated with particular
certificate hierarchies.

All agents MUST be capable of performing revocation checks using CRLs
as specified in [KEYM]. All agents MUST perform revocation status
checking in accordance with [KEYM]. Receiving agents MUST recognize
CRLs in received S/MIME messages.

4.2 Certificate Chain Validation

In creating a user agent for secure messaging, certificate, CRL, and
certificate chain validation SHOULD be highly automated while still
acting in the best interests of the user. Certificate, CRL, and chain
validation MUST be performed as per [KEYM] when validating a
correspondent's public key. This is necessary before using a public
key to provide security services such as: verifying a signature;
encrypting a content-encryption key (ex: RSA); or forming a pairwise
symmetric key (ex: Diffie-Hellman) to be used to encrypt or decrypt a
content-encryption key.

Certificates and CRLs are made available to the chain validation
procedure in two ways: a) incoming messages, and b) certificate and
CRL retrieval mechanisms. Certificates and CRLs in incoming messages
are not required to be in any particular order nor are they required
to be in any way related to the sender or recipient of the message
(although in most cases they will be related to the sender). Incoming
certificates and CRLs SHOULD be cached for use in chain validation
and optionally stored for later use. This temporary certificate and
CRL cache SHOULD be used to augment any other certificate and CRL
retrieval mechanisms for chain validation on incoming signed
messages.

4.3 Certificate and CRL Signing Algorithms

Certificates and Certificate-Revocation Lists (CRLs) are signed by
the certificate issuer. A receiving agent MUST be capable of
verifying the signatures on certificates and CRLs made with id-dsa-
with-sha1 [DSS].

A receiving agent SHOULD be capable of verifying the signatures on
certificates and CRLs made with md2WithRSAEncryption,
md5WithRSAEncryption and sha-1WithRSAEncryption signature algorithms
with key sizes from 512 bits to 2048 bits described in [PKCS#1V2].

4.4 PKIX Certificate Extensions

   PKIX describes an extensible framework in which the basic certificate
   information can be extended and how such extensions can be used to
   control the process of issuing and validating certificates. The PKIX
   Working Group has ongoing efforts to identify and create extensions
   which have value in particular certification environments. Further,
   there are active efforts underway to issue PKIX certificates for
   business purposes. This document identifies the minumum required set
   of certificate extensions which have the greatest value in the S/MIME
   environment. The syntax and semantics of all the identified
   extensions are defined in [KEYM].

   Sending and receiving agents MUST correctly handle the Basic
   Constraints Certificate Extension, the Key Usage Certificate
   Extension, authorityKeyID, subjectKeyID, and the subjectAltNames when
   they appear in end-user certificates. Some mechanism SHOULD exist to
   handle the defined certificate extensions when they appear in
   intermediate or CA certificates.

   Certificates issued for the S/MIME environment SHOULD NOT contain any
   critical extensions (extensions that have the critical field set to
   TRUE) other than those listed here. These extensions SHOULD be marked
   as non-critical unless the proper handling of the extension is deemed
   critical to the correct interpretation of the associated certificate.
   Other extensions may be included, but those extensions SHOULD NOT be
   marked as critical.

   Interpretation and syntax for all extensions MUST follow [KEYM],
   unless otherwise specified here.

4.4.1 Basic Constraints Certificate Extension

   The basic constraints extension serves to delimit the role and
   position of an issuing authority or end-entity certificate plays in a
   chain of certificates.

   For example, certificates issued to CAs and subordinate CAs contain a
   basic constraint extension that identifies them as issuing authority
   certificates. End-entity certificates contain an extension that
   constrains the certificate from being an issuing authority
   certificate.

   Certificates SHOULD contain a basicConstraints extension in CA
   certificates, and SHOULD NOT contain that extension in end entity
   certificates.

4.4.2 Key Usage Certificate Extension

   The key usage extension serves to limit the technical purposes for
   which a public key listed in a valid certificate may be used. Issuing
   authority certificates may contain a key usage extension that
   restricts the key to signing certificates, certificate revocation
   lists and other data.

   For example, a certification authority may create subordinate issuer
   certificates which contain a keyUsage extension which specifies that
   the corresponding public key can be used to sign end user certs and
   sign CRLs.

   If a key usage extension is included in a PKIX certificate, then it
   MUST be marked as critical.

4.4.2.1 Key Usage in Diffie-Hellman Key Exchange Certificates

   For Diffie-Hellman key exchange certificates (certificates in which
   the subject public key algorithm is dhpublicnumber), if the keyUsage
   keyAgreement bit is set to 1 AND if the public key is to be used to
   form a pairwise key to decrypt data, then the S/MIME agent MUST only
   use the public key if the keyUsage encipherOnly bit is set to 0. If
   the keyUsage keyAgreement bit is set to 1 AND if the key is to be
   used to form a pairwise key to encrypt data, then the S/MIME agent
   MUST only use the public key if the keyUsage decipherOnly bit is set
   to 0.

4.4.3 Subject Alternative Name Extension

   The subject alternative name extension is used in S/MIME as the
   preferred means to convey the RFC-822 email address(es) that
   correspond to the entity for this certificate. Any RFC-822 email
   addresses present MUST be encoded using the rfc822Name CHOICE of the
   GeneralName type. Since the SubjectAltName type is a SEQUENCE OF
   GeneralName, multiple RFC-822 email addresses MAY be present.

5. Security Considerations

   All of the security issues faced by any cryptographic application
   must be faced by a S/MIME agent. Among these issues are protecting
   the user's private key, preventing various attacks, and helping the
   user avoid mistakes such as inadvertently encrypting a message for
   the wrong recipient. The entire list of security considerations is
   beyond the scope of this document, but some significant concerns are
   listed here.

When processing certificates, there are many situations where the
processing might fail. Because the processing may be done by a user
agent, a security gateway, or other program, there is no single way
to handle such failures. Just because the methods to handle the
failures has not been listed, however, the reader should not assume
that they are not important. The opposite is true: if a certificate
is not provably valid and associated with the message, the processing
software should take immediate and noticable steps to inform the end
user about it.

Some of the many places where signature and certificate checking
might fail include:

- no Internet mail addresses in a certificate match the sender
  of a message
- no certificate chain leads to a trusted CA
- no ability to check the CRL for a certificate
- an invalid CRL was received
- the CRL being checked is expired
- the certificate is expired
- the certificate has been revoked

There are certainly other instances where a certificate may be
invalid, and it is the responsibility of the processing software to
check them all thoroughly, and to decide what to do if the check
fails.

A. References

    [CERTV2]   Dusse, S., Hoffman, P. and B. Ramsdell,"S/MIME Version 2
                   Certificate Handling", RFC 2312, March 1998.

    [CMS]   Housley, R., "Cryptographic Message Syntax", RFC 2630,
                   June 1999.

    [DSS]   NIST FIPS PUB 186, "Digital Signature Standard", 18 May
                   1994.

    [KEYM]   Housley, R., Ford, W., Polk, W. and D. Solo, "Internet
                   X.509 Public Key Infrastructure Certificate and CRL
                   Profile", RFC 2459, January 1999.

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

    [PKCS#1V2]   Kaliski, B., "PKCS #1: RSA Cryptography Specifications
                   Version 2.0", RFC 2437, October 1998.

    [RFC-822]   Crocker, D., "Standard For The Format Of ARPA Internet
                   Text Messages", STD 11, RFC 822, August 1982.

    [SMIME-MSG]   Ramsdell, B., Editor, "S/MIME Version 3 Message
                   Specification", RFC 2633, June 1999.

    [X.500]   ITU-T Recommendation X.500 (1997) | ISO/IEC 9594-1:1997,
                   Information technology - Open Systems Interconnection -
                   The Directory: Overview of concepts, models and
                   services.

    [X.501]   ITU-T Recommendation X.501 (1997) | ISO/IEC 9594-2:1997,
                   Information technology - Open Systems Interconnection -
                   The Directory: Models.

    [X.509]   ITU-T Recommendation X.509 (1997) | ISO/IEC 9594-8:1997,
                   Information technology - Open Systems Interconnection -
                   The Directory: Authentication framework.

    [X.520]   ITU-T Recommendation X.520 (1997) | ISO/IEC 9594-6:1997,
                   Information technology - Open Systems Interconnection -
                   The Directory: Selected attribute types.

B. Acknowledgements

Editor's Address

   Blake Ramsdell
   Worldtalk
   17720 NE 65th St Ste 201
   Redmond, WA 98052

   Phone: +1 425 376 0225
   EMail: blaker@deming.com