S/MIME WG
Internet Draft
Intended Status: Standard Track
Expires: March 24, 2009

Sean Turner, IECA
September 24, 2008

Using SHA2 Algorithms with Cryptographic Message Syntax
draft-ietf-smime-sha2-08.txt

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with Section 6 of BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

This Internet-Draft will expire on March 24, 2008.

Copyright Notice

Copyright (C) The IETF Trust (2008).

Abstract

This document describes the conventions for using the Secure Hash Algorithm (SHA) message digest algorithms (SHA-224, SHA-256, SHA-384, SHA-512) with the Cryptographic Message Syntax (CMS). It also describes the conventions for using these algorithms with CMS and the Digital Signature Algorithm (DSA), Rivest Shamir Adleman (RSA), and Elliptic Curve DSA (ECDSA) signature algorithms.

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 [RFC2119].

Table of Contents

1. Introduction....2
2. Message Digest Algorithms....3
2.1. SHA-224....4
2.2. SHA-256....4
2.3. SHA-384....4
2.4. SHA-512....4
3. Signature Algorithms....5
3.1. DSA....5
3.2. RSA....6
3.3. ECDSA....6
4. Security Considerations....7
5. IANA Considerations....7
6. References....7
6.1. Normative References....7
6.2. Informative References....8

1. Introduction

This document specifies the algorithm identifiers and specifies parameters for the message digest algorithms SHA-224, SHA-256, SHA-384, and SHA-512 for use with the Cryptographic Message Syntax (CMS) [RFC3852]. The message digest algorithms are defined in [SHS] and reference code is provided in [RFC4634].

This document also specifies the algorithm identifiers and parameters for use of SHA-224, SHA-256, SHA-384, and SHA-512 with DSA [DSS], RSA [RFC2313], and ECDSA [X9.62].

This document does not define new identifiers; they are taken from [RFC3874], [RFC4055], [ECCADD], and [RFC3278]. Additionally, the parameters follow the conventions specified therein. Therefore, there is no Abstract Syntax Notation One (ASN.1) module included in this document.

Note that [RFC4231] specifies the conventions for the message authentication code (MAC) algorithms: HMAC with SHA-224, HMAC with SHA-256, HMAC with SHA-384, and HMAC with SHA-512.

In CMS, the various algorithm identifiers use the AlgorithmIdentifier syntax, which is included here for convenience:

```
  AlgorithmIdentifier  ::=  SEQUENCE  {
    algorithm   OBJECT IDENTIFIER,
    parameters  ANY DEFINED BY algorithm OPTIONAL  }
```

2. Message Digest Algorithms

Digest algorithm identifiers are located in the SignedData digestAlgorithms field, the SignerInfo digestAlgorithm field, the DigestedData digestAlgorithm field, and the AuthenticatedData digestAlgorithm field.

Digest values are located in the DigestedData digest field and the Message Digest authenticated attribute. In addition, digest values are input to signature algorithms.

The digest algorithm identifiers use the AlgorithmIdentifier syntax elaborated upon in Section 1.

The algorithm field is discussed in Sections 2.1-2.4 for each message digest algorithm.

There are two possible encodings for the SHA AlgorithmIdentifier parameters field. The two alternatives arise from the fact that when the 1988 syntax for AlgorithmIdentifier was translated into the 1997 syntax, the OPTIONAL associated with the AlgorithmIdentifier parameters got lost. Later the OPTIONAL was recovered via a defect report, but by then many people thought that algorithm parameters were mandatory. Because of this history some implementations encode parameters as a NULL element and others omit them entirely. The correct encoding is to omit the parameters field; however, implementations MUST also handle a SHA AlgorithmIdentifier parameters field which contains a NULL.

The AlgorithmIdentifier parameters field is OPTIONAL. If present, the parameters field MUST contain a NULL. Implementations MUST accept SHA2 AlgorithmIdentifiers with absent parameters. Implementations MUST accept SHA2 AlgorithmIdentifiers with NULL parameters. Implementations SHOULD generate SHA2 AlgorithmIdentifiers with absent parameters.

2.1. SHA-224

The SHA-224 message digest algorithm is defined in [SHS]. The algorithm identifier for SHA-224 is:

```
  id-sha224 OBJECT IDENTIFIER ::= {
     joint-iso-itu-t(2) country(16) us(840) organization(1) gov(101)
     csor(3) nistalgorithm(4) hashalgs(2) 4 }
```

The parameters are as specified in Section 2.

2.2. SHA-256

The SHA-256 message digest algorithm is defined in [SHS]. The algorithm identifier for SHA-256 is:

```
 id-sha256 OBJECT IDENTIFIER ::= {
     joint-iso-itu-t(2) country(16) us(840) organization(1) gov(101)
     csor(3) nistalgorithm(4) hashalgs(2) 1 }
```

The parameters are as specified in Section 2.

2.3. SHA-384

The SHA-384 message digest algorithm is defined in [SHS]. The algorithm identifier for SHA-384 is:

```
  id-sha384 OBJECT IDENTIFIER ::= {
    joint-iso-itu-t(2) country(16) us(840) organization(1) gov(101)
    csor(3) nistalgorithm(4) hashalgs(2) 2 }
```

The parameters are as specified in Section 2.

2.4. SHA-512

The SHA-512 message digest algorithm is defined in [SHS]. The algorithm identifier for SHA-512 is:

```
  id-sha512 OBJECT IDENTIFIER ::= {
    joint-iso-itu-t(2) country(16) us(840) organization(1) gov(101)
    csor(3) nistalgorithm(4) hashalgs(2) 3 }
```

The parameters are as specified in Section 2.

3. Signature Algorithms

This section specifies the conventions employed by CMS implementations that support DSA, RSA, and ECDSA with SHA2 algorithms.

Signature algorithm identifiers are located in the SignerInfo signatureAlgorithm field of SignedData. Also, signature algorithm identifiers are located in the SignerInfo signatureAlgorithm field of countersignature attributes.

Signature values are located in the SignerInfo signature field of SignedData. Also, signature values are located in the SignerInfo signature field of countersignature attributes.

3.1. DSA

[RFC3370] section 3.1 specifies the conventions for DSA with SHA-1 public key algorithm identifiers, parameters, public keys, and signature values. DSA with SHA2 algorithms uses the same conventions for these public key algorithm identifiers, parameters, public keys, and signature values. DSA MAY be used with SHA-224 and SHA-256.

DSA has not been specified with SHA-384 and SHA-512. SHA-384 and SHA-512 are not supported because the maximum bit length of p (specified as L) is 3072 for DSA. For consistent cryptographic strength, SHA-384 would be used with DSA where L is 7680, and SHA-512 would be used with DSA where L is 15360.

The algorithm identifier for DSA with SHA-224 signature values is:

```
id-dsa-with-sha224 OBJECT IDENTIFIER ::= { joint-iso-ccitt(2)
  country(16) us(840) organization(1) gov(101) csor(3)
  algorithms(4) id-dsa-with-sha2(3) 1 }
```

The algorithm identifier for DSA with SHA-256 signature values is:

```
id-dsa-with-sha256 OBJECT IDENTIFIER ::= { joint-iso-ccitt(2)
  country(16) us(840) organization(1) gov(101) csor(3)
  algorithms(4) id-dsa-with-sha2(3) 2 }
```

When either of these algorithm identifiers is used, the AlgorithmIdentifier parameters field MUST be absent.

3.2. RSA

[RFC3370] section 3.2 specifies the conventions for RSA with SHA-1 (PKCS #1 v1.5) public key algorithm identifiers, parameters, public keys, and signature values. RSA with SHA2 algorithms uses the same conventions for these public key algorithm identifiers, parameters, public keys, and signature values. RSA (PKCS #1 v1.5) MAY be used with SHA-224, SHA-256, SHA-384, or SHA-512.

The object identifier for RSA with SHA-224 signature values is:

```
  sha224WithRSAEncryption OBJECT IDENTIFIER ::= { iso(1)
    member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-1(1) 14 }
```

The object identifier for RSA with SHA-256 signature values is:

```
  sha256WithRSAEncryption  OBJECT IDENTIFIER  ::=  { iso(1)
    member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-1(1) 11 }
```

The object identifier for RSA with SHA-384 signature values is:

```
  sha384WithRSAEncryption  OBJECT IDENTIFIER  ::=  { iso(1)
    member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-1(1) 12 }
```

The object identifier for RSA with SHA-512 signature values is:

```
  sha512WithRSAEncryption  OBJECT IDENTIFIER  ::=  { iso(1)
    member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-1(1) 13 }
```

When any of these four object identifiers appears within an AlgorithmIdentifier, the parameters MUST be NULL. Implementations MUST accept the parameters being absent as well as present.

3.3. ECDSA

[RFC3278] section 2.1 specifies the conventions for ECDSA with SHA-1 public key algorithm identifiers, parameters, public keys, and signature values. ECDSA with SHA2 algorithms uses the same conventions for these public key algorithm identifiers, parameters, public keys, and signature values, except that the digestAlgorithm MUST include the corresponding message digest algorithm identifier, and not the sha-1 object identifier. ECDSA MAY be used with SHA-224, SHA-256, SHA-384, or SHA-512.

The algorithm identifier for ECDSA with SHA-224 signature values is:

```
ecdsa-with-SHA224 OBJECT IDENTIFIER ::= { iso(1) member-body(2)
  us(840) ansi-X9-62(10045) signatures(4) ecdsa-with-SHA2(3) 1 }
```

The algorithm identifier for ECDSA with SHA-256 signature values is:

```
ecdsa-with-SHA256 OBJECT IDENTIFIER ::= { iso(1) member-body(2)
  us(840)ansi-X9-62(10045) signatures(4) ecdsa-with-SHA2(3) 2 }
```

The algorithm identifier for ECDSA with SHA-384 signature values is:

```
ecdsa-with-SHA384 OBJECT IDENTIFIER ::= { iso(1) member-body(2)
  us(840) ansi-X9-62(10045) signatures(4) ecdsa-with-SHA2(3) 3 }
```

The algorithm identifier for ECDSA with SHA-512 signature values is:

```
ecdsa-with-SHA512 OBJECT IDENTIFIER ::= { iso(1) member-body(2)
  us(840) ansi-X9-62(10045) signatures(4) ecdsa-with-SHA2(3) 4 }
```

When any of these four object identifiers appears within an AlgorithmIdentifier, the parameters MUST omit the parameters field. That is, the AlgorithmIdentifier SHALL be a SEQUENCE of one component: the OID ecdsa-with-SHA224, ecdsa-with-SHA256, ecdsa-with-SHA384 or ecdsa-with-SHA512.

4. Security Considerations

The security considerations in [RFC3278], [RFC3370], [RFC3874], [RFC4055], and [ECCADD] apply. No new security considerations are introduced as a result of this specification.

5. IANA Considerations

None: All identifiers are already registered. Please remove this section prior to publication as an RFC.

6. References

6.1. Normative References

[ECCADD] Dang, S., Santesson, S., Moriarty, K., and Brown, "Internet X.509 Public Key Infrastructure: Additional Algorithms and Identifiers for DSA and ECDSA", work-in-progress.

[DSS] National Institute of Standards and Technology (NIST), FIPS Publication 186-3: Digital Signature Standard, March 2006.

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

[RFC2313] Kaliski, B., "PKCS #1: RSA Encryption Version 1.5", RFC 2313, March 1998.

[RFC3278] Blake-Wilson, S., Brown, D., and P. Lambert, "Use of Elliptic Curve Cryptography (ECC) Algorithms in Cryptographic Message Syntax (CMS)", RFC 3278, April 2002.

[RFC3370] Housley, R., "Cryptographic Message Syntax (CMS) Algorithms", RFC 3370, August 2002.

[RFC3852] Housley, R., "The Cryptographic Message Syntax (CMS)", RFC 3852. July 2004.

[RFC3874] Housley, R., "A 224-bit One Way Hash Function: SHA-224", RFC 3874. September 2004.

[RFC4055] Schaad, J., Kaliski, B., and R. Housley, "Additional Algorithms and Identifiers for RSA Cryptography for use in the Internet Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 4055. June 2005.

[SHS] National Institute of Standards and Technology (NIST), FIPS Publication 180-3: Secure Hash Standard, June 2003.

[X9.62] X9.62-2005, "Public Key Cryptography for the Financial Services Industry: The Elliptic Curve Digital Signature Standard (ECDSA)", November, 2005.

6.2. Informative References

[RFC4231] Nystrom, A. "Identifiers and Test Vectors for HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512", RFC4231. December 2005.

[RFC4634] Eastlake, D., and T. Hansen, "US Secure Hash Algorithms (SHA and HMAC-SHA)", RFC 4634, July 2006.

Author's Addresses

Sean Turner

IECA, Inc.
3057 Nutley Street, Suite 106
Fairfax, VA 22031
USA

EMail: turners@ieca.com

Full Copyright Statement

Copyright (C) The IETF Trust (2008).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).