

NETWORK WG  
Internet Draft  
Intended Status: Informational  
Expires: June 3, 2010

Sean Turner  
IECA  
December 3, 2009

Additional S/MIME Capabilities  
draft-turner-additional-smimecaps-02.txt

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of [BCP 78](#) and [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/lid-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 June 3, 2008.

Copyright Notice

Copyright (c) 2009 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents in effect on the date of publication of this document (<http://trustee.ietf.org/license-info>). Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

Internet-Draft

Additional S/MIME Capabilities

December 2009

## Abstract

This document lists values for the S/MIME Capabilities Attribute. The attribute itself is defined in RFC TBD1, but the values for each are defined in separate algorithm documents and in some cases not at all. The SMIME Capability values can be included in S/MIME messages as a signed attribute and in public key certificates as an extension.

//RFC EDITOR: Replace TBD1 with the # assigned to [draft-ietf-smime-3851bis-10.txt](#).

## 1. Introduction

There has been and continues to be some confusion about an algorithm's parameter values. RFCs that define how an algorithm is used with CMS also define the algorithm's parameter values, e.g., [[RFC3370](#)]. Additionally, these RFCs should define the SMIMECapabilities attribute values; however, some have failed to do so and some have failed to do so correctly. Now, the situation may exist where implementations emit SMIMECapabilities attribute values that are the same as algorithm's parameters when used in CMS instead of following the SMIMECapabilities requirements from [[RFCTBD1](#)]: in "the event that there are no differentiating parameters for a particular OID, the parameters MUST be omitted, and MUST NOT be encoded as NULL." For example, ECDSA with SHA-1 from [[RFCTBD3](#)] includes NULL parameters when they should have been omitted.

//RFC EDITOR: Replace TBD3 with the # assigned to [draft-ietf-smime-3278bis-07.txt](#).

This document lists values for the S/MIME Capabilities Attribute. The attribute itself is defined in [[RFCTBD1](#)], but the values for each are defined in separate algorithm documents and in some cases not at all. Capability values can be included in S/MIME messages as an attribute and in public key certificates as an extension [[RFC4262](#)].

The majority of the values in this document are defined in other documents, and this document references those documents before the SMIME Capability. Values are encoded using the Distinguished Encoding Rule (DER) [[X.690](#)] and are a sequence of algorithm object identifier plus any parameters. The values provided in this document are values for one algorithm parameter pair. The syntax for the attribute is as follows and is included for convenience:

SMIMECapabilities ::= SEQUENCE OF SMIMECapability

Turner

Expires June 3, 2010

[Page 2]

---

Internet-Draft

Additional S/MIME Capabilities

December 2009

```
SMIMECapability ::= SEQUENCE {
    capabilityID OBJECT IDENTIFIER,
    parameters ANY DEFINED BY capabilityID OPTIONAL }
```

As specified in [\[RFC2811\]](#): "the object identifiers (OIDs) are listed in order of their preference, but SHOULD be separated logically along the lines of their categories (signature algorithms, symmetric algorithms, key encipherment algorithms, etc.)" As the "structure of the SMIMECapabilities attribute is [designed] to facilitate simple table lookups and binary comparisons in order to determine matches", the values are given in encoded format.

The DER [\[X.690\]](#) values for the capabilities are preceded by the algorithm's name and if they were previously defined a reference for the document in which they are defined.

### [1.1](#). Requirements 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](#). Message Digest Algorithms

[\[RFC3370\]](#) and [\[RFC2812\]](#) define the following message digest algorithms for use with CMS:

MD5: 300a 0608 2a86 4886 f70d 0205

NOTE: Though [\[RFC3370\]](#) allows NULL parameters for SHA-1, SHA-224, SHA-256, SHA-384, SHA-512, parameters MUST NOT be included as per [\[RFC2811\]](#) because there is no differentiating parameters for SHA-1 (e.g., output length).

SHA-1: 3007 0605 290e 0302 1a

[\[RFC2812\]](#) SHA-224: 300b 0609 6086 4801 6503 0402 04

[RFCTBD2] SHA-256: 300b 0609 6086 4801 6503 0402 01

[RFCTBD2] SHA-384: 300b 0609 6086 4801 6503 0402 02

[RFCTBD2] SHA-512: 300b 0609 6086 4801 6503 0402 03

### [3](#). Digital Signature Algorithms

[[RFC3370](#)], [[RFC4056](#)], [[RFCTBD2](#)], and [[RFCTBD3](#)] define the following digital signature algorithms for use with CMS:

RSA Encryption: 3009 0608 2a86 4886 f70d 0101 01

RSA With MD5: 3009 0608 2a86 4886 f70d 0101 04

RSA With SHA-1: 3009 0608 2a86 4886 f70d 0101 05

RSA With SHA-224: 3009 0608 2a86 4886 f70d 0101 0e

RSA With SHA-256: 3009 0608 2a86 4886 f70d 0101 0b

RSA With SHA-384: 3009 0608 2a86 4886 f70d 0101 0c

RSA With SHA-512: 3009 0608 2a86 4886 f70d 0101 0d

RSASSA-PSS: Add values here.

DSA With SHA-1: 3009 0607 2a86 48ce 3804 03

[RFCTBD2] DSA With SHA-224: 300b 0609 6086 4801 6503 0403 01

[RFCTBD2] DSA With SHA-256: 300b 0609 6086 4801 6503 0403 02

NOTE: [[RFCTBD3](#)] shows the ECDSA with SHA-1 with NULL parameter values, but the NULL parameters should not have been included according to [[RFCTBD1](#)]. The NULL is retained for backwards compatibility.

[RFCTBD3] ECDSA With SHA-1: 300b 0607 2a86 48ce 3d04 01 05 00

[RFCTBD3] ECDSA With SHA-224: 300a 0608 2a86 48ce 3d04 0301

[RFCTBD3] ECDSA With SHA-256: 300a 0608 2a86 48ce 3d04 0302

[RFCTBD3] ECDSA With SHA-384: 300a 0608 2a86 48ce 3d04 0303

[RFCTBD3] ECDSA With SHA-512: 300a 0608 2a86 48ce 3d04 0304

#### 4. Key Transport Algorithms

[RFC3370], [RFC3560], [RFCTBD4] define the following key transport algorithms for use with CMS:

Turner

Expires June 3, 2010

[Page 4]

---

Internet-Draft

Additional S/MIME Capabilities

December 2009

RSA Encryption: 300d 0608 2a86 4886 f70d 0101 01

[RFC3560] RSAES-OAEP Default: 300D 0609 2a86 4886 f70d 0101 0730 00

NOTE: [RFC3560] shows the RSAES-OAEP with SHA-224, SHA-256, 384, and 512 with NULL parameter values for the SHA algorithms, but the NULL parameters should not have been included according to [RFCTBD1].

[RFC3560] RSAES-OAEP SHA-224: 3038 0609 2a86 4886 f70d 0101 0730 2b30  
0d06 0960 8648 0165 0304 0201 0500 301a 0609 2a86 4886 f70d  
0101 0830 0d06 0960 8648 0165 0304 0204 0500

[RFC3560] RSAES-OAEP SHA-256: 3038 0609 2a86 4886 f70d 0101 0730 2b30  
0d06 0960 8648 0165 0304 0201 0500 301a 0609 2a86 4886 f70d  
0101 0830 0d06 0960 8648 0165 0304 0201 0500

[RFC3560] RSAES-OAEP SHA-384: 3038 0609 2a86 4886 f70d 0101 0730 2b30  
0d06 0960 8648 0165 0304 0202 0500 301a 0609 2a86 4886 f70d  
0101 0830 0d06 0960 8648 0165 0304 0202 0500

[RFC3560] RSAES-OAEP SHA-512: 3038 0609 2a86 4886 f70d 0101 0730 2b30  
0d06 0960 8648 0165 0304 0202 0500 301a 0609 2a86 4886 f70d  
0101 0830 0d06 0960 8648 0165 0304 0203 0500

[RFCTBD4] RSA-KEM:

#### 5. Key Agreement Algorithms

[[RFC2876](#)], [[RFC3370](#)], and [[RFCTBD3](#)] define the following key agreement algorithms for use with CMS:

NOTE: The parameters for key agreement algorithms are the key wrap algorithm (see [Section 6](#)).

[RFC2876] KEA: 3018 0609 6086 4801 6502 0101 1830 0b06 0960 8648  
0165 0201 0117

KA=DH S-S Wrap=Triple-DES: 301c 060d 2a86 4886 f70d 0109 1003 0a30  
0d06 0d2a 8648 86f7 0d01 0910 0306

KA=DH S-S Wrap=RC2 Para=40-bit: 3020 060d 2a86 4886 f70d 0109 1003  
0a30 1106 0d2a 8648 86f7 0d01 0910 0306 0202 00a0

KA=DH S-S Wrap=RC2 Para=64-bit: 301f 060d 2a86 4886 f70d 0109 1003  
0a30 1006 0d2a 8648 86f7 0d01 0910 0306 0201 78

KA=DH S-S Wrap=RC2 Para=128-bit: 301f 060d 2a86 4886 f70d 0109 1003  
0a30 1006 0d2a 8648 86f7 0d01 0910 0306 0201 3a

KA=DH E-S Wrap=Triple-DES: 301c 060d 2a86 4886 f70d 0109 1003 0530  
0d06 0d2a 8648 86f7 0d01 0910 0306

KA=DH E-S Wrap=RC2 Para=40-bit: 3020 060d 2a86 4886 f70d 0109 1003  
0530 1106 0d2a 8648 86f7 0d01 0910 030a 0202 00a0

KA=DH E-S Wrap=RC2 Para=64-bit: 301f 060d 2a86 4886 f70d 0109 1003  
0530 1006 0d2a 8648 86f7 0d01 0910 030a 0201 78

KA=DH E-S Wrap=RC2 Para=128-bit: 301f 060d 2a86 4886 f70d 0109 1003  
0530 1006 0d2a 8648 86f7 0d01 0910 030a 0201 3a

NOTE: [[RFCTBD3](#)] shows the ECDH with SHA-1|3 DES wrap capabilities with NULL parameter values, but the NULL parameters should not have been included according to [[RFCTBD1](#)]. The NULL is retained for backwards compatibility.

[RFCTBD3] KA=ECDH standard KDF=SHA-1 Wrap=Triple-DES: 301c 0609 2b81  
0510 8648 3f00 0230 0f06 0b2a 8648 86f7 0d01 0910 0306  
0500

[RFCTBD3] KA=ECDH standard KDF=SHA-224 Wrap=Triple-DES: 3017 0606  
2b81 0401 0b00 300e 060b 2a86 4886 f70d 0109 1003 06

[RFCTBD3] KA=ECDH standard KDF=SHA-256 Wrap=Triple-DES: 3017 0606  
2b81 0401 0b01 300e 060b 2a86 4886 f70d 0109 1003 06

[RFCTBD3] KA=ECDH standard KDF=SHA-384 Wrap=Triple-DES: 3017 0606  
2b81 0401 0b02 300e 060b 2a86 4886 f70d 0109 1003 06

[RFCTBD3] KA=ECDH standard KDF=SHA-512 Wrap=Triple-DES: 3017 0606  
2b81 0401 0b03 300e 060b 2a86 4886 f70d 0109 1003 06

[RFCTBD3] KA=ECDH standard KDF=SHA-1 Wrap=AES-128: 3018 0609 2b81  
0510 8648 3f00 0230 0b06 0960 8648 0165 0304 0105

[RFCTBD3] KA=ECDH standard KDF=SHA-224 Wrap=AES-128: 3015 0606 2b81  
0401 0b00 300b 0609 6086 4801 6503 0401 05

[RFCTBD3] KA=ECDH standard KDF=SHA-256 Wrap=AES-128: 3015 0606 2b81  
0401 0b01 300b 0609 6086 4801 6503 0401 05

[RFCTBD3] KA=ECDH standard KDF=SHA-384 Wrap=AES-128: 3015 0606 2b81  
0401 0b02 300b 0609 6086 4801 6503 0401 05

[RFCTBD3] KA=ECDH standard KDF=SHA-512 Wrap=AES-128: 3015 0606 2b81  
0401 0b03 300b 0609 6086 4801 6503 0401 05

[RFCTBD3] KA=ECDH standard KDF=SHA-1 Wrap=AES-192: 3018 0609 2b81  
0510 8648 3f00 0230 0b06 0960 8648 0165 0304 0119

[RFCTBD3] KA=ECDH standard KDF=SHA-224 Wrap=AES-192: 3015 0606 2b81  
0401 0b00 300b 0609 6086 4801 6503 0401 19

[RFCTBD3] KA=ECDH standard KDF=SHA-256 Wrap=AES-192: 3015 0606 2b81  
0401 0b01 300b 0609 6086 4801 6503 0401 19

[RFCTBD3] KA=ECDH standard KDF=SHA-384 Wrap=AES-192: 3015 0606 2b81  
0401 0b02 300b 0609 6086 4801 6503 0401 19

[RFCTBD3] KA=ECDH standard KDF=SHA-512 Wrap=AES-192: 3015 0606 2b81  
0401 0b03 300b 0609 6086 4801 6503 0401 19

[RFCTBD3] KA=ECDH standard KDF=SHA-1 Wrap=AES-256: 3018 0609 2b81  
0510 8648 3f00 0230 0b06 0960 8648 0165 0304 012d

[RFCTBD3] KA=ECDH standard KDF=SHA-224 Wrap=AES-256: 3015 0606 2b81  
0401 0B00 300b 0609 6086 4801 6503 0401 2d

[RFCTBD3] KA=ECDH standard KDF=SHA-256 Wrap=AES-256: 3015 0606 2b81  
0401 0b01 300b 0609 6086 4801 6503 0401 2d

[RFCTBD3] KA=ECDH standard KDF=SHA-384 Wrap=AES-256: 3015 0606 2b81  
0401 0b02 300b 0609 6086 4801 6503 0401 2d

[RFCTBD3] KA=ECDH standard KDF=SHA-512 Wrap=AES-256: 3015 0606 2b81  
0401 0B03 300b 0609 6086 4801 6503 0401 2d

NOTE: [[RFCTBD3](#)] shows the ECMQV with SHA-1 and 3 DES wrap capabilities with NULL parameter values, but the NULL parameters should not have been included according to [[RFCTBD1](#)]. The NULL is retained for backwards compatibility.

[RFCTBD3] KA=ECDH cofactor KDF=SHA-1 Wrap=Triple-DES: 301c 0609 2b81  
0510 8648 3f00 0330 0f06 0b2a 8648 86f7 0d01 0910 0306  
0500

[RFCTBD3] KA=ECDH cofactor KDF=SHA-224 Wrap=Triple-DES: 3017 0606  
2b81 0401 0e00 300d 060b 2a86 4886 f70d 0109 1003 06

[RFCTBD3] KA=ECDH cofactor KDF=SHA-256 Wrap=Triple-DES: 3017 0606  
2b81 0401 0e01 300d 060b 2a86 4886 f70d 0109 1003 06

[RFCTBD3] KA=ECDH cofactor KDF=SHA-384 Wrap=Triple-DES: 3017 0606  
2b81 0401 0e02 300d 060b 2a86 4886 f70d 0109 1003 06

[RFCTBD3] KA=ECDH cofactor KDF=SHA-512 Wrap=Triple-DES: 3017 0606  
2b81 0401 0e03 300d 060b 2a86 4886 f70d 0109 1003 06

[RFCTBD3] KA=ECDH cofactor KDF=SHA-1 Wrap=AES-128: 3018 0609 2b81  
0510 8648 3f00 0330 0b06 0960 8648 0165 0304 0105

[RFCTBD3] KA=ECDH cofactor KDF=SHA-224 Wrap=AES-128: 3015 0606 2b81  
0401 0e00 300b 0609 6086 4801 6503 0401 05

[RFCTBD3] KA=ECDH cofactor KDF=SHA-256 Wrap=AES-128: 3015 0606 2b81



0401 0e01 300b 0609 6086 4801 6503 0401 05

[RFCTBD3] KA=ECDH cofactor KDF=SHA-384 Wrap=AES-128: 3015 0606 2b81  
0401 0e02 300b 0609 6086 4801 6503 0401 05

[RFCTBD3] KA=ECDH cofactor KDF=SHA-512 Wrap=AES-128: 3017 0606 2b81  
0401 0e03 300b 0609 6086 4801 6503 0401 05

[RFCTBD3] KA=ECDH cofactor KDF=SHA-1 Wrap=AES-192: 30 18 06 09 2b 81  
0510 8648 3f00 0330 0b06 0960 8648 0165 0304 0119

[RFCTBD3] KA=ECDH cofactor KDF=SHA-224 Wrap=AES-192: 3015 0606 2b81  
0401 0e00 300b 0609 6086 4801 6503 0401 19

[RFCTBD3] KA=ECDH cofactor KDF=SHA-256 Wrap=AES-192: 3015 0606 2b81  
0401 0e01 300b 0609 6086 4801 6503 0401 19

[RFCTBD3] KA=ECDH cofactor KDF=SHA-384 Wrap=AES-192: 3015 0606 2b81  
0401 0e02 300b 0609 6086 4801 6503 0401 19

[RFCTBD3] KA=ECDH cofactor KDF=SHA-512 Wrap=AES-192: 3015 0606 2b81  
0401 0e03 300b 0609 6086 4801 6503 0401 19

[RFCTBD3] KA=ECDH cofactor KDF=SHA-1 Wrap=AES-256: 3015 0609 2b81  
0510 8648 3f00 0330 0b06 0960 8648 0165 0304 012d

[RFCTBD3] KA=ECDH cofactor KDF=SHA-224 Wrap=AES-256: 3015 0606 2b81  
0401 0e00 300b 0609 6086 4801 6503 0401 2d

[RFCTBD3] KA=ECDH cofactor KDF=SHA-256 Wrap=AES-256: 3015 0606 2b81  
0401 0e01 300b 0609 6086 4801 6503 0401 2d

[RFCTBD3] KA=ECDH cofactor KDF=SHA-384 Wrap=AES-256: 3015 0606 2b81  
0401 0e02 300b 0609 6086 4801 6503 0401 2d

[RFCTBD3] KA=ECDH cofactor KDF=SHA-512 Wrap=AES-256: 3015 0606 2b81  
0401 0e03 300b 0609 6086 4801 6503 0401 2d

NOTE: [[RFCTBD3](#)] shows the ECMQV with SHA-1 and 3 DES wrap capabilities with NULL parameter values, but the NULL parameters should not have been included according to [[RFCTBD1](#)]. The NULL is retained for backwards compatibility.

[RFCTBD3] KA=ECMQV 1-Pass KDF=SHA-1 Wrap=Triple-DES: 301c 0609 2b81  
0510 8648 3f00 1030 0f06 0b2a 8648 86f7 0d01 0910 0306  
0500

[RFCTBD3] KA=ECMQV 1-Pass KDF=SHA-224 Wrap=Triple-DES: 3017 0606  
2b81 0401 0f00 300d 060b 2a86 4886 f70d 0109 1003 06

[RFCTBD3] KA=ECMQV 1-Pass KDF=SHA-256 Wrap=Triple-DES: 3017 0606  
2b81 0401 0f01 300d 060b 2a86 4886 f70d 0109 1003 06

[RFCTBD3] KA=ECMQV 1-Pass KDF=SHA-384 Wrap=Triple-DES: 3017 0606  
2b81 0401 0f02 300d 060b 2a86 4886 f70d 0109 1003 06

[RFCTBD3] KA=ECMQV 1-Pass KDF=SHA-512 Wrap=Triple-DES: 3017 0606  
2b81 0401 0f03 300d 060b 2a86 4886 f70d 0109 1003 06

[RFCTBD3] KA=ECMQV 1-Pass KDF=SHA-1 Wrap=AES-128: 3018 0609 2b81  
0510 8648 3f00 1030 0b06 0960 8648 0165 0304 0105

[RFCTBD3] KA=ECMQV 1-Pass KDF=SHA-224 Wrap=AES-128: 3015 0606 2b81  
0401 0f00 300b 0609 6086 4801 6503 0401 05

[RFCTBD3] KA=ECMQV 1-Pass KDF=SHA-256 Wrap=AES-128: 3015 0606 2b81  
0401 0f01 300b 0609 6086 4801 6503 0401 05

[RFCTBD3] KA=ECMQV 1-Pass KDF=SHA-384 Wrap=AES-128: 3015 0606 2b81  
0401 0f02 300b 0609 6086 4801 6503 0401 05

[RFCTBD3] KA=ECMQV 1-Pass KDF=SHA-512 Wrap=AES-128: 3015 0606 2b81  
0401 0f03 300b 0609 6086 4801 6503 0401 05

[RFCTBD3] KA=ECMQV 1-Pass KDF=SHA-1 Wrap=AES-192: 3018 0609 2b81  
0510 8648 3f00 1030 0b06 0960 8648 0165 0304 0119

[RFCTBD3] KA=ECMQV 1-Pass KDF=SHA-224 Wrap=AES-192: 3015 0606 2b81  
0401 0f00 300b 0609 6086 4801 6503 0401 19

[RFCTBD3] KA=ECMQV 1-Pass KDF=SHA-256 Wrap=AES-192: 3015 0606 2b81  
0401 0f01 300b 0609 6086 4801 6503 0401 19

[RFCTBD3] KA=ECMQV 1-Pass KDF=SHA-384 Wrap=AES-192: 3015 0606 2b81  
0401 0f02 300b 0609 6086 4801 6503 0401 19

[RFCTBD3] KA=ECMQV 1-Pass KDF=SHA-512 Wrap=AES-192: 3015 0606 2b81  
0401 0f03 300b 0609 6086 4801 6503 0401 19

[RFCTBD3] KA=ECMQV 1-Pass KDF=SHA-1 Wrap=AES-256: 3018 0609 2b81  
0510 8648 3f00 1030 0b06 0960 8648 0165 0304 012d

[RFCTBD3] KA=ECMQV 1-Pass KDF=SHA-224 Wrap=AES-256: 3015 0606 2b81  
0401 0f00 300b 0609 6086 4801 6503 0401 2d

[RFCTBD3] KA=ECMQV 1-Pass KDF=SHA-256 Wrap=AES-256: 3015 0606 2b81  
0401 0f01 300b 0609 6086 4801 6503 0401 2d

[RFCTBD3] KA=ECMQV 1-Pass KDF=SHA-384 Wrap=AES-256: 3015 0606 2b81  
0401 0f02 300b 0609 6086 4801 6503 0401 2d

[RFCTBD3] ECMQV 1-Pass KDF=SHA-512 Wrap=AES-256: 3015 0606 2b81 0401  
0f03 300b 0609 6086 4801 6503 0401 2d

## 6. Key Wrap Algorithms

[RFC2876], [RFC3058], [RFC3370], [RFC3565], [RFC3657], [RFC4010], [RFC5649] define the following key agreement algorithms for use with CMS:

NOTE: In most instances, the key wrap algorithm is included in the capabilities set as part of the key agreement algorithm.

[RFC2876] FORTEZZA Wrap 80: 300b 0609 6086 4801 6502 0101 17

[RFC3058] IDEA: 300D 060B 2B06 0104 0181 3C07 0101 02

3-DES Wrap: 300e 060b 2a86 4886 f70d 0109 1003 06

RC2 40-bit: 3011 060d 2a86 4886 f70d 0109 1003 0602 0200 a0

RC2 64-bit: 3010 060d 2a86 4886 f70d 0109 1003 0602 0178

RC2 128-bit: 3010 060d 2a86 4886 f70d 0109 1003 0602 013a

AES-128 Key Wrap: 300b 0609 6086 4801 6503 0401 05

AES-196 Key Wrap: 300b 0609 6086 4801 6503 0401 19

AES-256 Key Wrap: 300b 0609 6086 4801 6503 0401 2d

AES-128 Key Wrap with Padding: 300b 0609 6086 4801 6503 0401 08

AES-196 Key Wrap with Padding: 300b 0609 6086 4801 6503 0401 1c

AES-256 Key Wrap with Padding: 300b 0609 6086 4801 6503 0401 30

Camellia 128-Wrap: 300d 060b 2a83 088c 9a4b 3d01 0103 02

Camellia 196-Wrap: 300d 060b 2a83 088c 9a4b 3d01 0103 03

Camellia 256-Wrap: 300d 060b 2a83 088c 9a4b 3d01 0103 03

SEED Wrap: 300c 060a 2a83 1a8c 9a44 0701 0101

## 7. Content Encryption Algorithms

[[RFC2876](#)], [[RFC3058](#)], [[RFC3370](#)], [[RFC3565](#)], [[RFC3657](#)], [[RFC5084](#)], and [[RFC5649](#)] define the following content encryption algorithms for use with CMS:

RC2-CBC 40-bit: 300d 0608 2a86 4886 f70d 0302 0201 28

RC2-CBC 64-bit: 300d 0608 2a86 4886 f70d 0302 0201 40

RC2-CBC 128-bit: 300e 0608 2a86 4886 f70d 0302 0202 0080

3-DES-CBC: 300a 0608 2a86 4886 f70d 0307

NOTE: [[RFC2876](#)] incorrectly included 00 at the end of the SMIMECapability.

[[RFC2876](#)] SKIPJACK: 300b 0609 6086 4801 6502 0101 04

[[RFC3058](#)] IDEA-CBC: 300d 060b 2b06 0104 0181 3c07 0101 02

[[RFC3565](#)] AES-CBC-128: 300b 0609 6086 4801 6503 0401 02

[[RFC3565](#)] AES-CBC-196: 300b 0609 6086 4801 6503 0401 16

[[RFC3565](#)] AES-CBC-256: 300b 0609 6086 4801 6503 0401 2a

AES-CCM-128: 300b 0609 6086 4801 6503 0401 07

AES-CCM-196: 300b 0609 6086 4801 6503 0401 1b

AES-CCM-256: 300b 0609 6086 4801 6503 0401 2f

Internet-Draft

Additional S/MIME Capabilities

December 2009

AES-GCM-128: 300b 0609 6086 4801 6503 0401 06

AES-GCM-196: 300b 0609 6086 4801 6503 0401 1a

AES-GCM-256: 300b 0609 6086 4801 6503 0401 2e

AES-128 Key Wrap: 300b 0609 6086 4801 6503 0401 05

AES-196 Key Wrap: 300b 0609 6086 4801 6503 0401 19

AES-256 Key Wrap: 300b 0609 6086 4801 6503 0401 2d

AES-128 Key Wrap with MLI: 300b 0609 6086 4801 6503 0401 08

AES-196 Key Wrap with MLI: 300b 0609 6086 4801 6503 0401 1c

AES-256 Key Wrap with MLI: 300b 0609 6086 4801 6503 0401 30

NOTE: Camellia defines their capability parameters as NULL.

[RFC3657] Camellia 128-CBC: 300f 060b 2a83 088c 9a4b 3d01 0101 0205  
00

[RFC3657] Camellia 196-CBC: 300f 060b 2a83 088c 9a4b 3d01 0101 0305  
00

[RFC3657] Camellia 256-CBC: 300f 060b 2a83 088c 9a4b 3d01 0101 0405  
00

NOTE: SEED defines their capability parameters as NULL.

[RFC4010] SEED CBC: 300C 0608 2a83 1a8c 9a44 0104 0500

## 8. Message Authentication Code Algorithms

[[RFC3370](#)], [[RFC4231](#)], and [[RFC4490](#)] define the following message authentication code algorithms for use with CMS:

HMAC SHA-1: 3009 0608 2b0601 0505 0801 02

HMAC SHA-224: 300a 0608 2a86 4886 f70d 0208

HMAC SHA-256: 300a 0608 2a86 4886 f70d 0209

HMAC SHA-384: 300a 0608 2a86 4886 f70d 020a

HMAC SHA-512: 300a 0608 2a86 4886 f70d 020b

Turner

Expires June 3, 2010

[Page 12]

---

Internet-Draft

Additional S/MIME Capabilities

December 2009

[RFC4490] HMAC GOST: 3008 0606 2A85 0302 0209

## [9](#). Compression Algorithms

[RFC3274] define the following compression algorithms for use with CMS:

[RFC3274] ZLIB: 300D 060B 2A86 4886 F70D 0109 1003 08

## [10](#). Security Considerations

This document does not advocate the use of any particular algorithm. The strength of the algorithms and applicability to their use in a particular environment is defined in the algorithms specifications.

## [11](#). IANA Considerations

There are no IANA considerations.

## [12](#). References

### [12.1](#). Normative References

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

[RFCTBD1] Turners, S., and B. Ramsdell, "Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.2 Message Specification", [draft-ietf-smime-3851bis-11.txt](#), work-in-progress.

//RFC EDITOR: Replace TBD1 with the # assigned to [draft-ietf-smime-3851bis-11.txt](#).

[X.690] ITU-T Recommendation X.690 (2002) | ISO/IEC 8825-1:2002, Information technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules

(DER).

## 12.2. Informative References

- [RFC2876] Pawling, J., "Use of the KEA and SKIPJACK Algorithms in CMS", [RFC 2876](#), July 2000.
- [RFC3058] Teiwe, S., Hartmann, P., Kuenzi, D., "Use of the IDEA Encryption Algorithm in CMS", [RFC 3058](#), February 2001.

Turner

Expires June 3, 2010

[Page 13]

---

Internet-Draft

Additional S/MIME Capabilities

December 2009

- [RFC3274] Gutmann, P., "CompressedData Content Type for Cryptographic Message Syntax", [RFC3274](#), June 2002.
- [RFC3370] Housley, R., "Cryptographic Message Syntax (CMS) Algorithms", [RFC 3370](#), August 2002.
- [RFC3560] Housley, R., "Use of the RSAES-OAEP Key Transport Algorithm in the Cryptographic Message Syntax (CMS)", [RFC 3560](#), July 2003.
- [RFC3565] Schaad, J., "Use of the Advanced Encryption Standard (AES) Encryption Algorithm in Cryptographic Message Syntax (CMS)", [RFC 3565](#), July 2003.
- [RFC3657] Moriai, S, Kato, A., "Use of the Camellia Encryption Algorithm", [RFC 3657](#), January 2004.
- [RFC4010] Park, J. Lee, S., Kim, J., and J. Lee, "Use of the SEED Encryption Algorithm in Cryptographic Message Syntax (CMS)", [RFC 4010](#), February 2005.
- [RFC4056] Schaad, J., " Use of the RSASSA-PSS Signature Algorithm in Cryptographic Message Syntax", [RFC 4056](#), June 2005.
- [RFC4231] Nystrom, M., "Identifiers and Test Vectors for HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512", [RFC 4231](#), December 2005
- [RFC4262] Santesson, S., "X.509 Certificate Extension for Secure/Multipurpose Internet Mail Extensions (S/MIME) Capabilities," [RFC 4262](#), December 2005.

- [RFC4490] Leontiev, S., and G. Chudov, Ed. "Using the GOST R 34.10-94, and GOST R 34.10-2001 Algorithms with Cryptographic Message Syntax (CMS)", [RFC 4490](#), May 2006.
- [RFC5084] Housley, R., "Using AES-CCM and AES-GCM Authenticated Encryption in the Cryptographic Message Syntax (CMS)", [RFC 5084](#), November 2007.
- [RFCTBD2] Turners, S., "Using SHA2 Algorithms with Cryptographic Message Syntax", [draft-ietf-smime-sha2-11.txt](#), work-in-progress.

Turner

Expires June 3, 2010

[Page 14]

---

Internet-Draft

Additional S/MIME Capabilities

December 2009

//RFC EDITOR: Replace TBD2 with the # assigned to [draft-ietf-smime-sha2-11.txt](#).

- [RFCTBD3] Turners, S., and D. Brown, "Use of Elliptic Curve Cryptography (ECC) Algorithms in Cryptographic Message Syntax (CMS)", [draft-ietf-smime-3278bis-09.txt](#), work-in-progress.

//RFC EDITOR: Replace TBD3 with the # assigned to [draft-ietf-smime-3278bis-09.txt](#).

- [RFCTBD4] Randall, J., and B.Kaliski, "Use of the RSA-KEM Key Transport Algorithm in CMS", [draft-ietf-smime-cms-rsa-kem-07.txt](#), work-in-progress.

//RFC EDITOR: Replace TBD4 with the # assigned to [draft-ietf-smime-cms-rsa-kem-07.txt](#).

- [RFC5649] Housley, R., and M. Dworkin, "Advanced Encryption Standard (AES) Key Wrap with Padding Algorithm", [RFC 5649](#), August 2009.

## Appendix A Revision History

[[ This entire section is to be removed upon publication. ]]

[A.1](#) Changes between [draft-turner-additional-smimecaps-00](#) and -01



Removed NULL parameters from RSA algorithms in [Section 3](#).

Corrected length in ECDH cofactor KDF=SHA-224, SHA-256, and SHA-512 Wrap Triple-DES in [Section 5](#).

Added RC2-CBC 64-bit in [Section 7](#).

#### [A.2](#) Changes between [draft-turner-additional-smimecaps-01](#) and -02

Provided a specific example of which algorithms included NULL as parameters when it shouldn't have.

Added an introduction paragraph to state which RFCs are being updated.

Removed and reworded many of the notes.

Added RSA-KEM and RSASSA-PSS capabilities.

Turner

Expires June 3, 2010

[Page 15]

---

Internet-Draft

Additional S/MIME Capabilities

December 2009

Removed NULL from SKIPJACK capability.

#### Authors' Addresses

Sean Turner

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

Email: [turners@ieca.com](mailto:turners@ieca.com)

Turner

Expires June 3, 2010

[Page 16]