Network Working Group
Internet-Draft
Obsoletes: 9212 (if approved)
Intended status: Informational
Expires: 4 September 2025

A. Becker
M. Jenkins
NSA

3 March 2025

Commercial National Security Algorithm (CNSA) Suite Profile for Secure/ Multipurpose Internet Mail Extensions (S/MIME) draft-becker-cnsa2-smime-profile-00

Abstract

This document defines a base profile of S/MIME for use with the U.S. Commercial National Security Algorithm (CNSA) 2.0 Suite, a cybersecurity advisory published by the United States Government which outlines quantum-resistant cryptographic algorithm policy for U.S. national security applications.

This profile applies to the capabilities, configuration, and operation of all components of U.S. National Security Systems that employ S/MIME. It is also appropriate for all other U.S. Government systems that process high-value information.

This profile is made publicly available for use by developers and operators of these and any other system deployments.

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 4 September 2025.

Copyright Notice

Table of Contents

1. Introduction

This document specifies a profile of S/MIME [RFC8551] to comply with the National Security Agency's (NSA) Commercial National Security Algorithm (CNSA) 2.0 Suite [annccnsa]. This profile applies to the capabilities, configuration, and operation of all components of U.S. National Security Systems (NSS) that employ S/MIME. U.S. National Security Systems are described in NIST Special Publication 800-59 [SP80059]. This profile is also appropriate for all other U.S. Government systems that process high-value information, and is made publicly available for use by developers and operators of these and any other system deployments.

The reader is assumed to have familiarity with [RFC8551].

[ED NOTE: This document uses some details from [I-D.ietf-lamps-cms-kyber-08] to specify use of ML-KEM in CMS, and [I-D.ietf-lamps-cms-ml-dsa-02] to specify use of ML-DSA in CMS. We note that these drafts are not yet RFCs, and we may need to adjust this text accordingly based on the progress of these documents.]

All usage of the term "CNSA" in this document refers to CNSA 2.0 [annccnsa], unless otherwise stated.

2. Terminology

Text from RFC2119.

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] if and only if they appear in all capitals, as shown here.

All MUST-level requirements of [RFC8551], apply throughout this profile; they are generally not repeated. In cases where a MUST-level requirement is repeated for clarity, the source document prevails. This profile may contain changes that elevate or downgrade some SHOULD- or MUST-level options from a specified RFC. All options from [RFC8551], that are not mentioned in this profile remain at the requirement level of the reference.

All usage of the term "CNSA" in this document refers to CNSA 2.0 [annccnsa], unless otherwise stated.

3. The Commercial National Security Algorithm Suite

The National Security Agency (NSA) profiles commercial cryptographic algorithms and protocols as part of its mission to support secure, interoperable communications for U.S. Government National Security Systems (NSS). To this end, it publishes guidance both to assist with the U.S. Government transition to new algorithms and to provide vendors - and the internet community in general - with information concerning their proper use and configuration within the scope of U.S. Government National Security Systems.

The Commercial National Security Algorithm (CNSA) Suite is the set of approved commercial algorithms that can be used by vendors and IT users to meet cybersecurity and interoperability requirements for NSS. The initial suite of CNSA Suite algorithms, Suite B, established a baseline for use of commercial algorithms to protect classified information. The following suite, CNSA 1.0, served as a bridge between the original set and a fully quantum-resistant cryptographic capability. The current suite, CNSA 2.0, establishes fully quantum-resistant protection [annccnsa].

Pursuant to the National Security Memorandum on Promoting United States Leadership in Quantum Computing While Mitigating Risks to Vulnerable Cryptographic Systems [NSM-10], the National Institute for

Standards and Technology (NIST) selected several quantum-resistant, or post-quantum, asymmetric algorithms for standardization. From these, NSA has included two in CNSA 2.0: ML-DSA-87 [FIPS204] for signing, and ML-KEM-1024 [FIPS203] for key establishment. ML-DSA-87 and ML-KEM-1024 together with SHA384, SHA512, AES-256, LMS, and XMSS comprise the CNSA Suite 2.0.

The NSA is authoring a set of RFCs, including this one, to provide updated guidance for using the CNSA 2.0 Suite in certain IETF protocols. These RFCs can be used in conjunction with other RFCs and cryptographic guidance (e.g., NIST Special Publications) to properly protect Internet traffic and data-at-rest for U.S. Government National Security Systems.

4. Requirements and Assumptions

CMS values are generated using ASN.1 [X208], the Basic Encoding Rules (BER) [X209], and the Distinguished Encoding Rules (DER) [X509].

For key agreement, CNSA compliant implementations MUST use ML-KEM-1024. Details provided in Section 7.

For digital signature, CNSA compliant implementations MUST use ML-DSA-87.

For CNSA Suite applications, public key certificates used to verify S/MIME signatures MUST be compliant with the CNSA Suite Certificate and Certificate Revocation List (CRL) profile specified in [I-D.jenkins-cnsa2-pkix-profile].

Within the CMS signed-data content type, signature algorithm identifiers are located in the signatureAlgorithm field of SignerInfo structures contained within the SignedData. In addition, signature algorithm identifiers are located in the SignerInfo signatureAlgorithm field of countersignature attributes. Specific requirements for digital signatures are given in Section 6; compliant implementations MUST consider signatures not meeting these requirements as invalid.

This document assumes that the required trust anchors have been securely provisioned to the client.

All implementations MUST use SHA-384 or SHA-512 for hashing and AES-GCM for encryption, the requirements for which are given in the following sections.

5. Message Digest

CNSA 2.0 allows either SHA-384 or SHA-512 to be used as a message digest algorithm. [RFC5754] specifies the conventions for using SHA-384 or SHA-512 with the Cryptographic Message Syntax (CMS). CNSA Suite-compliant S/MIME implementations follow the conventions in [RFC5754].

Within the CMS signed-data content type, message digest algorithm identifiers are located in the SignedData digestAlgorithms field and the SignerInfo digestAlgorithm field.

The SHA-384 message digest algorithm is defined in FIPS 180 [SHS], and the algorithm identifier for SHA-384 is defined in [RFC5754] as follows

```
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 SHA-512 message digest algorithm is defined in FIPS 180 [SHS], and the algorithm identifier for SHA-512 is defined in [RFC5754] as follows:

```
id-sha512 OBJECT IDENTIFIER ::= {

joint-iso-itu-t(2) country(16) us(840) organization(1) gov(101)

csor(3) nistalgorithm(4) hashalgs(2) 3 }
```

[RFC5754] defines the AlgorithmIdentifier parameters field as OPTIONAL. Implementations MUST accept SHA-384 and SHA-512 AlgorithmIdentifiers with absent parameters, or with NULL parameters. As specified in [RFC5754], implementations MUST generate SHA-384 or SHA-512 AlgorithmIdentifiers with absent parameters.

6. Digital Signature

CNSA 2.0 requires the use of ML-DSA-87 as the digital signature algorithm in S/MIME. [I-D.ietf-lamps-cms-ml-dsa-02] is ongoing work detailing the conventions for using ML-DSA with the Cryptographic Message Syntax (CMS). Note that [I-D.ietf-lamps-cms-ml-dsa-02] only specifies use of the pure mode (not pre-hash) of ML-DSA in CMS, which is also in accordance with CNSA. The “message digest” supplied to the signature algorithm is the entire message.

[EDNOTE: This guidance is subject to change as conventions for External Mu, as defined in [I-D.ietf-lamps-dilithium-certificates], are further defined.]

The algorithm identifier for ML-DSA-87 is defined in Section 2 of [I-D.ietf-lamps-cms-ml-dsa-02], and this document will be updated accordingly.

7. Key Establishment

7.1. KEM

For key agreement, CNSA 2.0 requires the use of ML-KEM-1024. [I-D.ietf-lamps-cms-kyber-08] is ongoing work detailing the conventions for implementing ML-KEM in the KEMRecipientInfo structure in CMS to securely transfer the content-encryption key from the originator to the recipient. We also refer to guidance from [RFC9629].

A CMS originator MUST implement the Encapsulate algorithm from ML-KEM, and a CMS responder MUST implement the Decapsulate algorithm from ML-KEM.

CNSA compliant implementations MUST NOT include the ukm input to the key-derivation function.

7.2. KDF

A CNSA compliant implementation MUST support SHA-384 or SHA-512 for KDF computation. We note the KDF used to process the KEMRecipientInfo structure MAY be different from the KDF used in the ML-KEM-1024 algorithm.

7.3. AES Key Wrap

Within the CMS enveloped-data content type, key wrap algorithm identifiers are located in the KeyWrapAlgorithm parameters within the EnvelopedData RecipientInfos KeyAgreeRecipientInfo keyEncryptionAlgorithm field.

For CNSA-compliant implementations, the KeyWrapAlgorithm MUST be

```
id-aes256-wrap-pad
```

The required algorithm identifier, specified in [RFC5649] is:

```
id-aes256-wrap-pad OBJECT IDENTIFIER ::= { joint-iso-itu-t(2)
```

```
country(16) us(840) organization(1) gov(101) csor(3)

nistAlgorithm(4) aes(1) 48 }
```

8. Content Encryption

CNSA-compliant S/MIME implementations using the authenticated-enveloped-data content type [RFC5083] MUST use AES [AES] in Galois Counter Mode (GCM) [SP80038D] as the content authenticated encryption algorithm and MUST follow the conventions for using AES-GCM with the CMS defined in [RFC5084].

Within the CMS authenticated-enveloped-data content type, content-authenticated encryption algorithm identifiers are located in the AuthEnvelopedData EncryptedContentInfo contentEncryptionAlgorithm field. The content-authenticated encryption algorithm is used to encipher the content located in the AuthEnvelopedData EncryptedContentInfo encryptedContent field.

The AES-GCM content-authenticated encryption algorithm is described in [GCM] and [SP80038D]. The algorithm identifier for AES-256 in GCM mode is:

```
id-aes256-GCM OBJECT IDENTIFIER ::= { joint-iso-itu-t(2)

 country(16) us(840) organization(1) gov(101) csor(3)

 nistAlgorithm(4) aes(1) 46 }
```

The AlgorithmIdentifier parameters field MUST be present, and the parameters field must contain GCMParameters:

```
GCMParameters ::= SEQUENCE { )

 aes-nonce OCTET STRING,

 aes-ICVlen AES-GCM-ICVlen DEFAULT 12 }
```

The authentication tag length (aes-ICVlen) SHALL be 16 (indicating a tag length of 128 bits).

The initialization vector (aes-nonce) MUST be generated in accordance with Section 8.2 of [SP80038D]. AES-GCM loses security catastrophically if a nonce is reused with a given key on more than one distinct set of input data. Therefore, a fresh content-authenticated encryption key MUST be generated for each message.

9. Security Considerations

Most of the security considerations for this document are described in [RFC8551]. Additional security considerations for the use of ML-KEM and ML-DSA in S/MIME can be found in [I-D.ietf-lamps-cms-kyber-08] and [I-D.ietf-lamps-cms-ml-dsa-02], respectively.

10. IANA Considerations

This document has no IANA actions

11. References

[AES] National Institute of Standards and Technology, "Announcing the ADVANCED ENCRYPTION STANDARD (AES)", FIPS 197, DOI 10.6028/NIST.FIPS.197, November 2001, <https://nvlpubs.nist.gov/nistpubs/fips/NIST.FIPS.197.pdf>.

[annccnsa] National Security Agency, "Announcing the Commercial National Security Algorithm Suite 2.0", September 2022, <https://media.defense.gov/2022/Sep/07/2003071834/-1/-1/0/CSA_CNSA_2.0_ALGORITHMS_.PDF>.

[FIPS203] National Institute of Standards and Technology (2024), "Module-Lattice-Based Key-Encapsulation Mechanism Standard", (Department of Commerce, Washington, D.C.), Federal Information Processing Standards Publication (FIPS), NIST FIPS 203, <https://doi.org/10.6028/NIST.FIPS.203>.

[FIPS204] National Institute of Standards and Technology (2024), "Module-Lattice-Based Digital Signature Standard", (Department of Commerce, Washington, D.C.), Federal Information Processing Standards Publication (FIPS), NIST FIPS 204, <https://doi.org/10.6028/NIST.FIPS.204>.

[GCM] Dworkin, M., "Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC", NIST Special Publication 800-38D, DOI 10.6028/NIST.SP.800-38D, November 2007, <https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38d.pdf>.

[I-D.ietf-lamps-cms-kyber-08]
Prat, Julien., Ounsworth, Mike., and Daniel. Van Geest, "Use of ML-KEM in the Cryptographic Message Syntax (CMS)", January 2025, <https://datatracker.ietf.org/doc/draft-ietf-lamps-cms-kyber/08/>.

[I-D.ietf-lamps-cms-ml-dsa-02]
S, Ben., R, Adam., and Daniel. Van Geest, "Use of the ML-DSA Signature Algorithm in the Cryptographic Message Syntax (CMS)", January 2025, <https://datatracker.ietf.org/doc/draft-ietf-lamps-cms-ml-dsa/>.

[I-D.ietf-lamps-dilithium-certificates]
Massimo, Jake., "Internet X.509 Public Key Infrastructure: Algorithm Identifiers for ML-DSA", February 2025, <https://datatracker.ietf.org/doc/draft-ietf-lamps-dilithium-certificates/>.

[I-D.jenkins-cnsa2-pkix-profile]
Jenkins, M. and A. Becker, "Commercial National Security Algorithm Suite Certificate and Certificate Revocation List Profile", January 2025, <https://datatracker.ietf.org/doc/draft-jenkins-cnsa2-pkix-profile/>.

[NSM-10] United States, The White House, "National Security Memorandum on Promoting United States Leadership in Quantum Computing While Mitigating Risks to Vulnerable Cryptographic Systems", NSM 10, May 2022, <https://www.whitehouse.gov/briefing-room/statements-releases/2022/05/04/national-security-memorandum-on-promoting-united-states-leadership-in-quantum-computing-while-mitigating-risks-to-vulnerable-cryptographic-systems/>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC5083] Housley, R., "Cryptographic Message Syntax (CMS) Authenticated-Enveloped-Data Content Type", RFC 5083, DOI 10.17487/RFC5083, November 2007, <https://www.rfc-editor.org/info/rfc5083>.

[RFC5084] Housley, R., "Using AES-CCM and AES-GCM Authenticated Encryption in the Cryptographic Message Syntax (CMS)", RFC 5084, DOI 10.17487/RFC5084, November 2007, <https://www.rfc-editor.org/info/rfc5084>.

[RFC5649] Housley, R. and M. Dworkin, "Advanced Encryption Standard (AES) Key Wrap with Padding Algorithm", RFC 5649, DOI 10.17487/RFC5649, September 2009, <https://www.rfc-editor.org/info/rfc5649>.

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

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

[RFC8551] Schaad, J., Ramsdell, B., and S. Turner, "Secure/ Multipurpose Internet Mail Extensions (S/MIME) Version 4.0 Message Specification", RFC 8551, DOI 10.17487/RFC8551, April 2019, <https://www.rfc-editor.org/info/rfc8551>.

[RFC9629] Housley, R., Gray, J., and T. Okubo, "Using Key Encapsulation Mechanism (KEM) Algorithms in the Cryptographic Message Syntax (CMS)", RFC 9629, DOI 10.17487/RFC9629, August 2024, <https://www.rfc-editor.org/info/rfc9629>.

[SHS] National Institute of Standards and Technology (NIST), "Secure Hash Standard (SHS)", DOI 10.6028/NIST.FIPS.180-4, FIPS PUB 180-4, August 2015, <https://nvlpubs.nist.gov/nistpubs/FIPS/ NIST.FIPS.180-4.pdf>.

[SP80038D] National Institute of Standards and Technology, "Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, NIST Special Publication 800-38D", Special Publication 800-38D, DOI 10.6028/NIST.SP.800-38D, November 2007, <https://doi.org/10.6028/NIST.SP.800-38D>>.

[SP80056A] National Institute of Standards and Technology, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography, Revision 3, NIST Special Publication 800-56A", Special Publication 800-56A, DOI 10.6028/NIST.SP.800-56Ar3, April 2018, <https://doi.org/10.6028/NIST.SP.800-56Ar3>>.

[SP80059] National Institute of Standards and Technology, "Guideline for Identifying an Information System as a National Security System", Special Publication 59, DOI 10.6028/NIST.SP.800-59, August 2003, <https://csrc.nist.gov/publications/detail/sp/800-59/final>.

[X208] CCITT, "Specification of Abstract Syntax Notation One (ASN.1)", CCITT Recommendation X.208 1998, <https://www.itu.int/rec/T-REC-X.208-198811-W/en>.

[X209] CCITT, "Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)", CCITT Recommendation X.209 1998, <https://www.itu.int/rec/T-REC-X.209-198811-W/en>.

[X509] CCITT, "The Directory - Authentication Framework", CCITT Recommendation X.509 1998, <https://www.itu.int/rec/T-REC-X.509-198811-S>.

Authors' Addresses

Alison Becker
National Security Agency
Email: aebecke@uwe.nsa.gov

Michael Jenkins
National Security Agency
Email: mjjenki@cyber.nsa.gov