Network Working Group
Internet-Draft
Obsoletes: 9206 (if approved)
Intended status: Informational
Expires: 28 February 2026

R. Guthrie
NSA

27 August 2025

Commercial National Security Algorithm (CNSA) Suite 2.0 Profile for IPsec
draft-guthrie-cnsa2-ipsec-profile-01

Abstract

This document defines a base profile for IPsec for use with the US Commercial National Security Algorithm (CNSA) 2.0 Suite, a cybersecurity advisory that outlines quantum-resistant cryptographic algorithm policy for national security applications. This profile applies to the capabilities, configuration, and operation of all components of US National Security Systems that employ IPsec. It is also appropriate for all other US Government systems that process high-value information. This memo is not an IETF standard, and has not been shown to have IETF community consensus. 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 28 February 2026.

Copyright Notice

Table of Contents

1. Introduction

This document specifies an Internet Protocol Security (IPsec) profile to comply with the National Security Agency's (NSA) Commercial National Security Algorithm (CNSA) 2.0 Suite [CNSA2]. This profile applies to the capabilities, configuration, and operation of all components of US National Security Systems (NSS) [SP80059] that employ IPsec. This profile is also appropriate for all other US 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.

This document does not specify how to use any cryptographic algorithm not currently supported by IPsec; instead, it profiles CNSA 2.0-compliant conventions for IPsec, and it uses only algorithms in the CNSA 2.0 suite already specified for use by IPsec.

This memo is not an IETF standard, and has not been shown to have IETF community consensus.

2. 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. Normative language does not apply beyond the scope of this profile.

AES: Advanced Encryption Standard

DH: Diffie-Hellman key establishment

ECDH: Elliptic Curve Diffie-Hellman

PSK/PPK: Pre-Shared Key/Post-Quantum Pre-Shared Key; in this document, used interchangeably

ML-KEM: Module-Lattice-Based Key Encapsulation Mechanism, defined in [FIPS203]

ML-DSA: Module-Lattice-Based Digital Signature Algorithm, defined in [FIPS204]

3. The Commercial National Security Algorithm Suite 2.0

This profile uses CNSA 2.0 compliant algorithms ML-KEM-1024 [FIPS203] for key establishment and ML-DSA-87 [FIPS204] as a digital signature. In addition, this profile also includes the key establishment algorithms allowed by the CNSA 1.0 IPsec profile [RFC9206]: ECDH with curve P-384 [SP80056A] and DH with prime modulus of 3072 bits or 4096 bits [RFC3526].

If ML-KEM-1024 were used in the IKE_SA_INIT exchange, the sizes of its public key and ciphertext would cause the initiator and responder messages to exceed the typical path MTU and necessitate in IP-level fragmentation, which can cause operational challenges and prevent the establishment of a connection.

To address this issue, [RFC9370] enables peers to perform multiple key exchanges. While the cryptographic content exchanged to facilitate the first key establishment algorithm (performed in IKE_SA_INIT) must be constrained enough in size as to not induce IP fragmentation, a subsequent key establishment algorithm can be performed in the IKE_INTERMEDIATE exchange [RFC9242], which precedes IKE_AUTH. This exchange, because it is encrypted by the initial key establishment algorithm, can now leverage the IKEv2-level fragmentation mechanism specified in [RFC7383]. Use of this mechanism allows public keys and ciphertexts to be exchanged in messages that exceed path MTU and avoids IP fragmentation.

This document profiles the use of [RFC9370] and [I-D.ietf-ipsecme-ikev2-mlkem] where the key exchanges are instantiated as follows:

* The key exchange performed in IKE_SA_INIT can use any of the following algorithms: 384-bit random ECP group, 3072-bit MODP group, 4096-bit MODP group.

* The additional key exchange performed in IKE_INTERMEDIATE MUST be ML-KEM-1024.

[EDNOTE: The CNSA 1.0 key establishment algorithms are permitted instead of, e.g., a smaller quantum resistant algorithm because they enable backwards compatibility with CNSA 1.0-compliant implementations of IPsec during the transition to the CNSA 2.0 Suite and facilitate interoperability among implementations, while introducing negligible security risks (described further in Security Considerations). Should this justification be included in the final text?]

4. IKE_SA_INIT Exchange

4.1. Overview

In the IKE_SA_INIT exchange, the initiator sends a message to the responder that includes the following payloads:

* SAi1 [RFC7296]

* KEi [RFC7296]

* Ni [RFC7296]

* N(IKEV2_FRAGMENTATION_SUPPORTED) [RFC7383]

* N(INTERMEDIATE_EXCHANGE_SUPPORTED) [RFC9242]

* N(SIGNATURE_HASH_ALGORITHMS)
  [RFC7427][I-D.ietf-ipsecme-ikev2-pqc-auth]

* [N(USE_PPK_INT)] [I-D.ietf-ipsecme-ikev2-qr-alt]

The responder sends the initiator a message that includes the following payloads:

* SAr1 [RFC7296]

* KEr [RFC7296]

* Nr [RFC7296]

* CERTREQ [RFC7296]

* N(IKEV2_FRAGMENTATION_SUPPORTED) [RFC7383]

* N(INTERMEDIATE_EXCHANGE_SUPPORTED) [RFC9242]

* N(SIGNATURE_HASH_ALGORITHMS)
  [RFC7427][I-D.ietf-ipsecme-ikev2-pqc-auth]

* [N(USE_PPK_INT)] [I-D.ietf-ipsecme-ikev2-qr-alt]

Note that optional payloads are indicated using brackets; some payloads, while optional in their respective specifications, are required by this profile. These differences are discussed in more detail throughout this section.

Section 4.2 discusses requirements concerning the contents of the SA payloads. Section 4.3 discusses requirements on the KE payloads. Section 4.4 discusses the Notify payloads included in the lists above. Section 4.5 discusses the CERTREQ payload sent by the responder.

No additional requirements are imposed for the Nonce payloads Ni and Nr beyond what is detailed in [RFC7296].

Note that it may be necessary for the initiator and responder to include Notify payloads other than those listed above in their respective IKE_SA_INIT messages for certain use cases (such as those used for NAT traversal). Such use cases are not discussed here as they do not require further profiling and should be implemented in accordance with the relevant RFCs.

4.2. Security Association Payloads

Security Association (SA) payloads are used to propose IPsec protocols and the cryptographic algorithms (indicated via Transform Types) that correspond to each protocol. A CNSA 2.0-compliant implementation of IPsec will use the Encapsulating Security Payload (ESP) protocol [RFC4303] and the IKEv2 protocol [RFC7296]. The SA[i/r]1 payloads exchanged in IKE_SA_INIT are used to present IKE proposals. Proposals for the ESP protocol are presented during the IKE_AUTH exchange and are discussed in Section 6.3.

The IKE proposal includes the following:

* Encryption Algorithm (Transform Type 1): ENCR_AES_GCM_16 (with key size 256 bits)

* Pseudorandom Function (Transform Type 2): PRF_HMAC_SHA2_512 or PRF_HMAC_SHA2_384

* Integrity Algorithm (Transform Type 3): NONE or not offered

* Key Exchange Method (Transform Type 4): 384-bit random ECP group or 3072-bit MODP group or 4096-bit MODP group

* Additional Key Exchange 1 (Transform Type 6): ML-KEM-1024

While the proposal requires negotiation of both a confidentiality algorithm and integrity algorithm, algorithms for Authenticated Encryption with Associated Data (AEAD) [RFC5116] provide integrity, and are incompatible with the use of a separate integrity algorithm. In particular, since the Advanced Encryption Standard [FIPS197] in Galois-Counter Mode [SP80038D] (AES-GCM) [RFC8247] is an AEAD algorithm, the IKE proposal offering AES-GCM as its encryption algorithm MUST either offer no integrity algorithm or an integrity algorithm of "NONE" with the former being the RECOMMENDED method, as per [RFC7296], Section 3.3.

AES-GCM-SIV [RFC8452] conforms with the requirements of this document and MAY be used instead of AES-GCM if a Federal Information Processing Standard (FIPS) validated implementation is available and a Transform Type 1 value is registered with IANA.

[EDNOTE: Please contact the document author to express interest in or identify implementations that support AES-GCM-SIV.]

Note that ML-KEM-1024 MAY be proposed using any one of the Additional Key Exchange Transform Types (ADDKEs) 1-7 (with Transform Type 6-12 respectively), but a proposal MUST only include a single ADDKE Transform Type, and that ADDKE Transform Type MUST be ML-KEM-1024.

An initiator proposal MUST be constructed using all of the Transform Types indicated above. For Transform Types that list more than one algorithm as an option, an initiator proposal MUST include at least one of the options listed and no options other than those listed above.

A responder MUST accept initiator proposals that fit this description. If none of the proposals offered by the initiator comply with the above description, then the responder MUST return a Notify payload with the error NO_PROPOSAL_CHOSEN when operating in CNSA-compliant mode.

4.3. Key Exchange Payloads

The Key Exchange Method field [RFC9370] in the initiator's Key Exchange payload MUST be one of the following:

* 384-bit random ECP group

* 3072-bit MODP group

* 4096-bit MODP group

A responder compliant with this profile MUST send a N(INVALID_KE_PAYLOAD) [RFC7296] if it receives a Key Exchange payload using a Key Exchange Method value corresponding to any algorithm not listed above.

The following requirements are restated from [RFC9206].

In every key establishment session, the CNSA-compliant initiator and responder MUST each generate ephemeral keys.

While IKEv2 allows for the reuse of ephemeral Diffie-Hellman private keys [RFC7296], Section 2.12, there are security concerns related to this practice. In order to address such concerns, [I-D.ietf-ipsecme-ikev2-mlkem] requires ephemeral keys to be generated per connection. Moreover, this profile REQUIRES CNSA 2.0-compliant IPsec implementations to align with [SP80056A]. In particular, an ephemeral private key MUST be used in exactly one key establishment transaction and MUST be destroyed (zeroized) as soon as possible. Any shared secret derived from key establishment MUST also be destroyed (zeroized) immediately after its use.

If the Elliptic Curve Diffie-Hellman (ECDH) key exchange is used, the initiator and responder both MUST generate an elliptic curve (EC) key pair using the P-384 elliptic curve. The ephemeral public keys MUST be stored in the key exchange payload as described in [RFC5903].

If the Diffie-Hellman (DH) key exchange is used, the initiator and responder both MUST generate a key pair using the appropriately sized MODP group as described in [RFC3526]. The size of the MODP group will be determined by the selection of either a 3072-bit or greater modulus for the SA.

As noted in [RFC5903], Section 7, the shared secret result of an ECDH key exchange is the 384-bit x value of the ECDH common value. The shared secret result of a DH key exchange is the number of octets needed to accommodate the prime (e.g., 384 octets for 3072-bit MODP group) with leading zeros as necessary [RFC2631], Section 2.1.2.

4.4. Notify Payloads

In order to use the Intermediate Exchange [RFC9242] and IKE-level fragmentation [RFC7383], initiator and responder MUST include both of the following Notify payloads in their respective IKE_SA_INIT messages:

* N(IKEV2_FRAGMENTATION_SUPPORTED) [RFC7383]

* N(INTERMEDIATE_EXCHANGE_SUPPORTED) [RFC9242]

[EDNOTE: Is there a way to signal that both of the above Notify payloads are required by this profile even though they are optional in their respective specifications?]

In order to use the ML-DSA-87 signature algorithm, initiator and responder MUST include the N(SIGNATURE_HASH_ALGORITHMS) [RFC7427], Section 4 payload in their respective IKE_SA_INIT messages as specified in Section 5 of [I-D.ietf-ipsecme-ikev2-pqc-auth].

The initiator and responder also MAY negotiate support for [I-D.ietf-ipsecme-ikev2-qr-alt] using N(USE_PPK_INT) as specified in [I-D.ietf-ipsecme-ikev2-qr-alt].

Note that while [RFC9206] supported the use of a PSK via [RFC8784], this profile instead aims to support the use of a PSK as specified in [I-D.ietf-ipsecme-ikev2-qr-alt]. This profile acknowledges that there may be a period of transition in which implementations support the use of a PSK via [RFC8784], but have not yet added support for [I-D.ietf-ipsecme-ikev2-qr-alt]. During this time, a CNSA 2.0-compliant initiator MAY include only N(USE_PPK) in its

IKE_SA_INIT message if it does not yet support [I-D.ietf-ipsecme-ikev2-qr-alt]. Because each PSK mechanism is negotiated independently (though only one is used in a given connection), a CNSA 2.0-compliant initiator also MAY include both N(USE_PPK_INT) and N(USE_PPK) in its IKE_SA_INIT message to accommodate the scenario where a responder who is otherwise CNSA 2.0-compliant has not yet added in support for [I-D.ietf-ipsecme-ikev2-qr-alt]. In either scenario, if the responder replies with N(USE_PPK) [RFC8784] but is otherwise CNSA 2.0-compliant, the initiator MAY proceed in establishing the SA using a PSK as specified by [RFC8784]. A CNSA 2.0-compliant responder that receives both and supports both N(USE_PPK_INT) and N(USE_PPK) MUST return N(USE_PPK_INT) and agree to use a PSK as specified by [I-D.ietf-ipsecme-ikev2-qr-alt].

4.5. Certificate Request Payload

While [RFC7296] treats the CERTREQ payload as optional, this profile requires its use.

[EDNOTE: [I-D.ietf-ipsecme-ikev2-pqc-auth] discusses mechanisms for signaling which digital signature algorithms are supported. In the case of CNSA 2.0, the inclusion of the N(SIGNATURE_HASH_ALGORITHMS) payload in IKE_SA_INIT is sufficient to indicate that ML-DSA is supported because it is the only CNSA 2.0-compliant signature algorithm that uses this Notify payload. However, one proposal from [I-D.ietf-ipsecme-ikev2-pqc-auth] uses the CERTREQ payload to indicate that a CA certificate signed by an ML-DSA key is trusted. Do future implementors have a preference between these two methods? Note that while [I-D.ietf-ipsecme-ikev2-pqc-auth] also proposes the use of N(SUPPORTED_AUTH_METHODS) [RFC9593] as another mechanism, CNSA 2.0 does not plan to support the use of this extension.]

[EDNOTE: Separate from the above discussion, this profile is considering requiring the use of the CERTREQ payload even though it is optional per [RFC7296]. Can implementations support this? Are there compelling use cases for not using the CERTREQ payload (sent by either the responder in IKE_SA_INIT or the initiator in IKE_AUTH) though X.509-based authentication is required?]

5. IKE_INTERMEDIATE Exchanges

5.1. Overview

The IKE_INTERMEDIATE exchange performs an additional key establishment with ML-KEM-1024.

The initiator sends an IKE_INTERMEDIATE message to the responder containing an encrypted KEi(1) [RFC9370] payload. In reply, the responder sends an IKE_INTERMEDIATE message containing an encrypted KEr(1) [RFC9370] payload. Requirements for Key Exchange payloads and ML-KEM are discussed in Section 5.2.

If [I-D.ietf-ipsecme-ikev2-qr-alt] is supported by initiator and responder, a second IKE_INTERMEDIATE exchange can also be used to mix in a pre-shared key. PSK guidance and requirements are discussed in detail in Section 5.3.

Implementations compliant with this profile MUST NOT use additional IKE_INTERMEDIATE exchanges to facilitate further key establishments. In other words, peers MUST NOT use IKE_INTERMEDIATE exchanges to send KE[i/r](n) payloads for values other than n=1 or that contain key establishment material for algorithms other than ML-KEM-1024.

5.2. Key Exchange Payloads

In IKE_INTERMEDIATE, initiator and responder messages each contain a Key Exchange payload in order to facilitate a second key establishment which uses ML-KEM-1024.

ML-KEM-1024 has Key Exchage Method value "TBD37" [EDNOTE: Will be assigned by IANA when [I-D.ietf-ipsecme-ikev2-mlkem] is published].

This profile strengthens normative key generation, encapsulation, and decapsulation guidance from [I-D.ietf-ipsecme-ikev2-mlkem], Section 2.3 as follows: Responders MUST perform the checks specified in Section 7.2 of [FIPS203] prior to performing Encaps(pk). If the checks fail, the responder MUST send a N(INVALID_SYNTAX) payload as a response to the request from the initiator. Initiators MUST perform the checks specified in Section 7.3 of [FIPS203] prior to performing Decaps(sk, ct). In this case, the initiator SHOULD send a N(INVALID_SYNTAX) payload to the responder using the IKE_INFORMATIONAL exchange. This is an exception for the general requirement to not begin a new exchange based on errors in responses.

As per [I-D.ietf-ipsecme-ikev2-mlkem], ephemeral ML-KEM private keys must be generated per connection. Additionally, a CNSA 2.0-compliant IPsec implementation MUST use ML-KEM ephemeral private keys in exactly one key establishment transaction, and MUST destroy (zeroize) said key as soon as possible. Any shared secret derived from key establishment MUST also be destroyed (zeroized) immediately after its use, as is required for (EC)DH in Section 4.3.

The following requirement is restated from [I-D.ietf-ipsecme-ikev2-mlkem] for emphasis: the ML-KEM public key generated by the initiator and the ciphertext generated by the responder use randomness (usually a seed) which MUST be independent of any other random seed used in the IKEv2 negotiation. For example, at the initiator, the ML-KEM and (EC)DH keypairs used in a PQ/T Hybrid key exchange should not be generated from the same seed.

[EDNOTE: Additional RBG requirements for ML-KEM may be included here.]

5.3. PSK

If initiator and responder signal support for [I-D.ietf-ipsecme-ikev2-qr-alt] in IKE_SA_INIT, the initiator MAY send N(PPK_IDENTITY_KEY) payload(s) to the responder using the IKE_INTERMEDIATE exchange, and the responder will reply, both in accordance with [I-D.ietf-ipsecme-ikev2-qr-alt], Section 3.1.

While [I-D.ietf-ipsecme-ikev2-qr-alt] indicates that the initiator MAY include the N(PPK_IDENTITY_KEY) payload(s) in the IKE_INTERMEDIATE exchange message used to transfer ML-KEM key material, it is RECOMMENDED by this profile that the initiator use a separate IKE_INTERMEDIATE exchange to agree on a pre-shared key such that the ML-KEM exchange precedes the PSK exchange.

Note that PSKs shall be at least 256 bits in length, and generated from a NIST approved random bit generator that supports 256-bits of entropy [SP80090C].

6. IKE_AUTH Exchange

6.1. Overview

In the IKE_AUTH exchange, the initiator sends a message to the responder that includes the following payloads:

* IDi [RFC7296]
* CERT [RFC7296]
* CERTREQ [RFC7296]
* [IDr] [RFC7296]
* AUTH [RFC7296] (modified by [RFC9242], [RFC9370], [I-D.ietf-ipsecme-ikev2-qr-alt], [I-D.ietf-ipsecme-ikev2-pqc-auth])

* SAi2 [RFC7296]

* TSi [RFC7296]

* TSr [RFC7296]

The responder sends a message to the initiator that includes the following payloads:

* IDr [RFC7296]

* CERT [RFC7296]

* AUTH [RFC7296] (modified by [RFC9242], [RFC9370], [I-D.ietf-ipsecme-ikev2-qr-alt], [I-D.ietf-ipsecme-ikev2-pqc-auth])

* SAr2 [RFC7296]

* TSi [RFC7296]

* TSr [RFC7296]

Note that optional payloads are indicated using brackets; some payloads, while optional in their respective specifications, are required by this profile. These differences are discussed in more detail throughout this section.

The initiator’s CERTREQ payload is discussed in Section 6.2. The contents of the initiator SAi2 payload and the responder SAr2 payload are discussed in Section 6.3. Section 6.4 discusses the initiator and responder AUTH payloads. The CERT payloads are discussed in Section 6.5.

Note that it may be necessary for the initiator and responder to include other Notify payloads in their respective IKE_AUTH messages for certain use cases. Such instances are not discussed here as they do not require further profiling and should be implemented in accordance with the relevant RFCs.

6.2. Certificate Request Payload

While [RFC7296] treats the CERTREQ payload as optional, this profile requires the CERTREQ payload to be used.

[EDNOTE: See related EDNOTE in Section 4.5.]

6.3. Security Association Payloads

The initiator and responder present their ESP proposals in SAi2 and SAr2 respectively.

The ESP proposal includes the following:

* Encryption Algorithm (Transform Type 1): ENCR_AES_GCM_16 (with key size 256 bits)

* Integrity Algorithm (Transform Type 2): NONE or not offered

While ESP as specified in [RFC7296] optionally supports an integrity algorithm, the use of AES-GCM [RFC4106] for an encryption algorithm requires that either no integrity algorithm or an algorithm NONE be offered.

AES-GCM-SIV [RFC8452] conforms with the requirements of this document and MAY be used instead of AES-GCM if a Federal Information Processing Standard (FIPS) validated implementation is available and a Transform Type 1 value is registered with IANA.

[EDNOTE: [RFC7296] states that DH (now Key Exchange Method) is optional for ESP, but the choice to do this would imply that a KE payload could optionally be included in IKE_AUTH messages, which is not addressed in [RFC7296]. Is this interpretation correct? If so, is this something that happens in practice?]

[EDNOTE: Guidance on Extended Sequence Numbers (Transform Type 5) [RFC7296], Section 3.3.2 in ESP proposals is forthcoming.]

6.4. Authentication Payloads

This profile REQUIRES the use of digital signature ML-DSA-87 [FIPS204]. If the relying party receives a message signed with any authentication method other than ML-DSA-87, it MUST return an AUTHENTICATION_FAILED Notify payload and stop processing the message.

In alignment with [I-D.ietf-ipsecme-ikev2-pqc-auth] and as defined by [FIPS204], this profile permits the use of both hedged and deterministic variants of ML-DSA.

Note that the calculation of the Authentication Data field in the AUTH payload [RFC7296] is updated by [I-D.ietf-ipsecme-ikev2-mlkem] (which leverages [RFC9370], which leverages [RFC9242], which leverages [RFC7383]). Additionally, if peers agreed on and mixed in a PSK as in [I-D.ietf-ipsecme-ikev2-qr-alt], this also impacts the calculation of the Authentication Data field.

[EDNOTE: Is the way that Authentication Data calculation is updated clear, or should this be explained?]

[EDNOTE: Clarification regarding the use of ML-DSA vs. ExternalMu-ML-DSA [I-D.ietf-lamps-dilithium-certificates], Part Appendix D is forthcoming. Note that HashML-DSA will not be permitted (also in accordance with [I-D.ietf-lamps-dilithium-certificates]).]

6.5. Certificate Payloads

While [RFC7296] treats the inclusion of CERT payloads in IKE_AUTH messages as optional, CNSA 2.0-compliant initiator and responder IKE_AUTH messages MUST include the requisite CERT payloads. All CERT payloads MUST comply with [I-D.jenkins-cnsa2-pkix-profile]. CERT payload(s) sent by both initiator and responder MUST include the Cert Encoding of X.509 Certificate - Signature (4). Note that if a chain of certificates needs to be sent, multiple CERT payloads are used, where only the first of which holds the public key used to validate the sender’s AUTH payload [RFC7296], Section 3.6. CERT payloads sent by initiator or responder may also optionally use other Cert Encodings (such as Certificate Revocation List (7)) as needed. Other public key formats (such as PGP Certificate=2, SPKI Certificate=9) MUST NOT be used. Peer authentication decisions MUST be based on the Subject or Subject Alternative Name from the certificate that contains the key used to validate the digital signature in the AUTH payload, rather than the Identification Data from the ID payload that is used to look up policy.

7. CREATE_CHILD_SA Exchanges

Forthcoming.

8. Additional Requirements

Forthcoming.

9. Security Considerations

Forthcoming.

10. IANA Considerations

None.

11. References

[CNSA2] National Security Agency, "The Commercial National Security Algorithm Suite 2.0 and Quantum Computing FAQ", December 2024, <https://media.defense.gov/2022/Sep/07/2003071836/-1/-1/0/CSI_CNSA_2.0_FAQ_.PDF>.

[FIPS197] National Institute of Standards and Technology, "Advanced Encryption Standard (AES)", May 2023.

[FIPS203] National Institute of Standards and Technology, "Module-Lattice-Based Key-Encapsulation Mechanism Standard", August 2024.

[FIPS204] National Institute of Standards and Technology, "Module-Lattice-Based Digital Signature Standard", August 2024.

[I-D.ietf-ipsecme-ikev2-mlkem] Kampanakis, P., "Post-quantum Hybrid Key Exchange with ML-KEM in the Internet Key Exchange Protocol Version 2 (IKEv2)", Work in Progress, Internet-Draft, draft-ietf-ipsecme-ikev2-mlkem-02, 6 August 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-ipsecme-ikev2-mlkem-02>.

[I-D.ietf-ipsecme-ikev2-pqc-auth] Reddy, T., Smyslov, V., and S. Fluhrer, "Signature Authentication in the Internet Key Exchange Version 2 (IKEv2) using PQC", Work in Progress, Internet-Draft, draft-ietf-ipsecme-ikev2-pqc-auth-03, 29 August 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-ipsecme-ikev2-pqc-auth-03>.

[I-D.ietf-ipsecme-ikev2-qr-alt] Smyslov, V., "Mixing Preshared Keys in the IKE_INTERMEDIATE and in the CREATE_CHILD_SA Exchanges of IKEv2 for Post-quantum Security", Work in Progress, Internet-Draft, draft-ietf-ipsecme-ikev2-qr-alt-10, 23 May 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-ipsecme-ikev2-qr-alt-10>.

[I-D.ietf-lamps-dilithium-certificates] Massimo, J., Kampanakis, P., Turner, S., and B. Westerbaan, "Internet X.509 Public Key Infrastructure - Algorithm Identifiers for the Module-Lattice-Based Digital Signature Algorithm (ML-DSA)", Work in Progress, Internet-Draft, draft-ietf-lamps-dilithium-certificates-12, 26 June 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-lamps-dilithium-certificates-12>.

[I-D.jenkins-cnsa2-pkix-profile]
Jenkins, M. J. and A. Becker, "Commercial National Security Algorithm Suite Certificate and Certificate Revocation List Profile", Work in Progress, Internet-Draft, draft-jenkins-cnsa2-pkix-profile-02, 21 April 2025, <https://datatracker.ietf.org/doc/html/draft-jenkins-cnsa2-pkix-profile-02>.

[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>.

[RFC2631] Rescorla, E., "Diffie-Hellman Key Agreement Method", RFC 2631, DOI 10.17487/RFC2631, June 1999, <https://www.rfc-editor.org/info/rfc2631>.

[RFC3526] Kivinen, T. and M. Kojo, "More Modular Exponential (MODP) Diffie-Hellman groups for Internet Key Exchange (IKE)", RFC 3526, DOI 10.17487/RFC3526, May 2003, <https://www.rfc-editor.org/info/rfc3526>.

[RFC4106] Viega, J. and D. McGrew, "The Use of Galois/Counter Mode (GCM) in IPsec Encapsulating Security Payload (ESP)", RFC 4106, DOI 10.17487/RFC4106, June 2005, <https://www.rfc-editor.org/info/rfc4106>.

[RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, DOI 10.17487/RFC4303, December 2005, <https://www.rfc-editor.org/info/rfc4303>.

[RFC5116] McGrew, D., "An Interface and Algorithms for Authenticated Encryption", RFC 5116, DOI 10.17487/RFC5116, January 2008, <https://www.rfc-editor.org/info/rfc5116>.

[RFC5903] Fu, D. and J. Solinas, "Elliptic Curve Groups modulo a Prime (ECP Groups) for IKE and IKEv2", RFC 5903, DOI 10.17487/RFC5903, June 2010, <https://www.rfc-editor.org/info/rfc5903>.

[RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T. Kivinen, "Internet Key Exchange Protocol Version 2 (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October 2014, <https://www.rfc-editor.org/info/rfc7296>.

[RFC7383] Smyslov, V., "Internet Key Exchange Protocol Version 2 (IKEv2) Message Fragmentation", RFC 7383, DOI 10.17487/RFC7383, November 2014, <https://www.rfc-editor.org/info/rfc7383>.

[RFC7427] Kivinen, T. and J. Snyder, "Signature Authentication in the Internet Key Exchange Version 2 (IKEv2)", RFC 7427, DOI 10.17487/RFC7427, January 2015, <https://www.rfc-editor.org/info/rfc7427>.

[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>.

[RFC8247] Nir, Y., Kivinen, T., Wouters, P., and D. Migault, "Algorithm Implementation Requirements and Usage Guidance for the Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 8247, DOI 10.17487/RFC8247, September 2017, <https://www.rfc-editor.org/info/rfc8247>.

[RFC8452] Gueron, S., Langley, A., and Y. Lindell, "AES-GCM-SIV: Nonce Misuse-Resistant Authenticated Encryption", RFC 8452, DOI 10.17487/RFC8452, April 2019, <https://www.rfc-editor.org/info/rfc8452>.

[RFC8784] Fluhrer, S., Kampanakis, P., McGrew, D., and V. Smyslov, "Mixing Preshared Keys in the Internet Key Exchange Protocol Version 2 (IKEv2) for Post-quantum Security", RFC 8784, DOI 10.17487/RFC8784, June 2020, <https://www.rfc-editor.org/info/rfc8784>.

[RFC9206] Corcoran, L. and M. Jenkins, "Commercial National Security Algorithm (CNSA) Suite Cryptography for Internet Protocol Security (IPsec)", RFC 9206, DOI 10.17487/RFC9206, February 2022, <https://www.rfc-editor.org/info/rfc9206>.

[RFC9242] Smyslov, V., "Intermediate Exchange in the Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 9242, DOI 10.17487/RFC9242, May 2022, <https://www.rfc-editor.org/info/rfc9242>.

[RFC9370] Tjhai, CJ., Tomlinson, M., Bartlett, G., Fluhrer, S., Van Geest, D., Garcia-Morchon, O., and V. Smyslov, "Multiple Key Exchanges in the Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 9370, DOI 10.17487/RFC9370, May 2023, <https://www.rfc-editor.org/info/rfc9370>.

[RFC9593] Smyslov, V., "Announcing Supported Authentication Methods in the Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 9593, DOI 10.17487/RFC9593, July 2024, <https://www.rfc-editor.org/info/rfc9593>.

[SP80038D] National Institute of Standards and Technology, "Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC", November 2007.

[SP80056A] National Institute of Standards and Technology, "Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography", April 2018.

[SP80059] National Institute of Standards and Technology, "Guideline for Identifying an Information System as a National Security System", August 2003.

[SP80090C] National Institute of Standards and Technology, "Recommendation for Random Bit Generator (RBG) Constructions", July 2024.

Author’s Address

Rebecca Guthrie
National Security Agency
Email: rmguthr@uwe.nsa.gov