3KEM and KEM-Sign Brainstorming

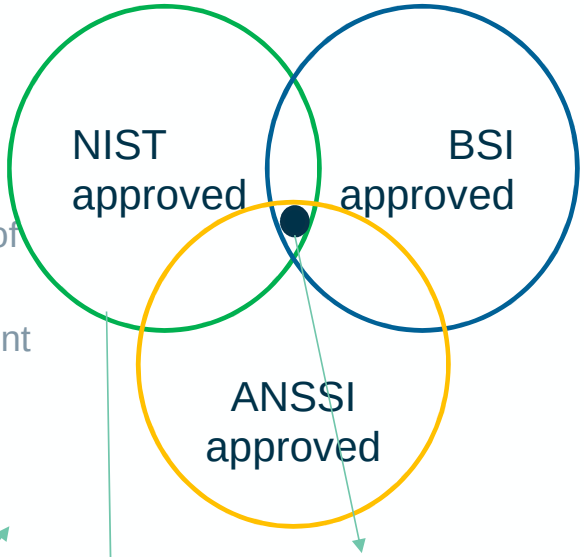
Oana Graur (TEC-ESS)

06/08/2024

Baseline configuration - derisking strategy

- Any CCSDS key negotiation protocol solution needs to incorporate **cryptographic primitives which are mature**
  - Shall avoid instantiation with cryptographic primitives known to be very complex to implement on standard space hardware, prone to side-channel attacks or not previously standardized by NIST, IETF, ANSI, etc.
  - When using well established standards for different instantiations, while not all “features” of those standards are relevant, the ones that are relevant should not be deviated from -> **interoperability**. In other words, minimal subset that does the job is OK, but do not reinvent the wheel where not needed.
- The proposed strategy here is to choose cryptographic protocols that lie at the **intersection between NIST, ANSSI and BSI recommendations and best practices.** Literature is available online and will be referenced at the end of the presentation.

How to choose the crypto primitive instantiations



Additional restrictions apply for space applications (space hardware limitations, bandwidth, etc)

Not a rigid strategy, but a good starting point. Additional partners may have additional constraints

Important for ESA partners (e.g., NASA) which are bound by US law to certify modules according to FIPS 140 if they process unclassified but sensitive data.

PQC KEMs: ANSSI / BSI / NCSC / NIST Recommendations

KEM	Standard available?	Recommended by NIST?	Recomm. in CNSA 2.0?	Recommended by BSI?	Recomm. by ANSSI?	Recomm. By NCSC?
Kyber (ML-KEM)	NIST FIPS 203	Yes	Yes, see [0]	Yes, BSI intends to include ML-KEM in the recommendations of [6] with the parameter sets for NIST Level 3 and Level 5.	Yes, see [2,3,5]	Yes, see [31], Level 5 [32]
FrodoKEM	No, NIST PQC round 3 alternate candidate [4], under ISO standardisation	Conservative security but performance worse than BIKE/HQC, not selected for NIST standardization [1]	Not present	Yes, see [6]	Yes, see [2,3,5,30]	Yes, see [32], Level 3 or higher
Classic McEliece	Submitted to NIST Round 4	TBD, too early to tell	Not at this stage	Yes, mceliece460896, mceliece6688128 and mceliece8192128, and the faster variants mceliece460896f, mceliece6688128f and mceliece8192128f in [6]	Not found, assume no at this stage. See [8], [9].	Yes, see [32], Level 3 or higher [32]
BIKE	Submitted to NIST Round 4	TBD, too early to tell	Not at this stage	TBD, too early to tell	TBD, too early to tell	TBD, too early to tell
HQC	Submitted to NIST Round 4	TBD, too early to tell	Not at this stage	TBD, too early to tell	TBD, too early to tell	TBD, too early to tell

ANSSI Recommendations

[2] ANSSI Plan for post-quantum transition (pkic.org)
[3] Follow_up_position_paper_on_post_quantum_cryptography.pdf (cyber.gouv.fr)
[5] ANSSI's recommendations on the migration plan (cyber.gouv.fr)
[30] ANSSI views on the Post-Quantum Cryptography transition | ANSSI (cyber.gouv.fr)

BSI Recommendations

[6] Cryptographic Mechanisms: Recommendations and Key Lengths, Version 2024-01 (bund.de)
[7] https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/Brochure/quantum-safe-cryptography.pdf?__blob=publicationFile&v=6

Additional References

[4] FrodoKEM-standard_proposal-20230314.pdf
[8] Faulting original McEliece's implementations is possible
[9] The syzygy distinguisher https://eprint.iacr.org/2024/1193.pdf
[31] NCSC whitepaper: Next steps in preparing for post-quantum cryptography
[32] NCSC: Guidelines for quantum-safe transport-layer encryption

NIST/NSA Recommendations

[1] Status Report on the 3rd Round of the NIST PQC Standardization Process
[0] The Commercial National Security Algorithm Suite 2.0 and Quantum Computing FAQ

PQC Sign: ANSSI / BSI / NCSC / NIST Recommendations

PQC Signatures	Standard available?	Recomm. by NIST?	Recomm. in CNSA 2.0?	Recommended by BSI?	Recomm. by ANSSI?	Recomm. By NCSC?
Dilithium (ML-DSA)	NIST FIPS 204	**Yes**	**Yes**, Level 5 [0]	**Yes**, "The BSI intends to include SLH-DSA and ML-DSA with the parameter sets corresponding to NIST Security Strength Categories 3 and 5 in " reference [6]	**Yes**, see [2,3,5], hybridization mandatory	**Yes**, [31], all Levels acceptable, Level 3 or higher recommended
SPHINCS+ (SLH-DSA)	NIST FIPS 205	Yes	Not present	**Yes,** "The BSI intends to include SLH-DSA and ML-DSA with the parameter sets corresponding to NIST Security Strength Categories 3 and 5 in " reference [6]". Note attack when SPHINCS+ instantiated with SHA-256. [6]	**Yes**, see [2,3,5], hybridization not needed	**Sometimes** [31] "They are **not** suitable for general purpose use as the signatures are large and the algorithms are much slower than **ML-DSA**. […] may be a good fit for use cases such as signing firmware and software where speed is not a bottleneck." For XMSS/LMS, careful with state [34]
XMSS /LMS	NIST SP 800 208, RFC 8391, ETSI TR 103 692	**Yes**	**Yes,** see [0]	**Yes,** see [6]	**Yes**, allowed for software updates, careful with the state, note the limited number of signatures [3]	
FALCON	Under NIST standardization	Draft not yet available	Not at this stage	No recommendation found in [6].	**Yes**, Level 5 recommended if used [2], "requires floating point operations, vulnerable to side channel attacks"	

N/A for 3KEM, relevant for KEM-Sign

ANSSI Recommendations
[2] ANSSI Plan for post-quantum transition (pkic.org)
[3] Follow_up_position_paper_on_post_quantum_cryptography.pdf (cyber.gouv.fr)
[5] ANSSI's recommendations on the migration plan (cyber.gouv.fr)
[30] ANSSI views on the Post-Quantum Cryptography transition | ANSSI (cyber.gouv.fr)

BSI Recommendations
[6] Cryptographic Mechanisms: Recommendations and Key Lengths, Version 2024-01 (bund.de)
[7] https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/Brochure/quantum-safe-cryptography.pdf?__blob=publicationFile&v=6

Additional References
[31] NCSC whitepaper: Next steps in preparing for post-quantum cryptography
[32] NCSC: Guidelines for quantum-safe transport-layer encryption
[43] ETSI TR 103 692 State management for stateful authentication mechanisms

NIST/NSA Recommendations
[1] Status Report on the 3rd Round of the NIST PQC Standardization Process
[0] The Commercial National Security Algorithm Suite 2.0 and Quantum Computing FAQ

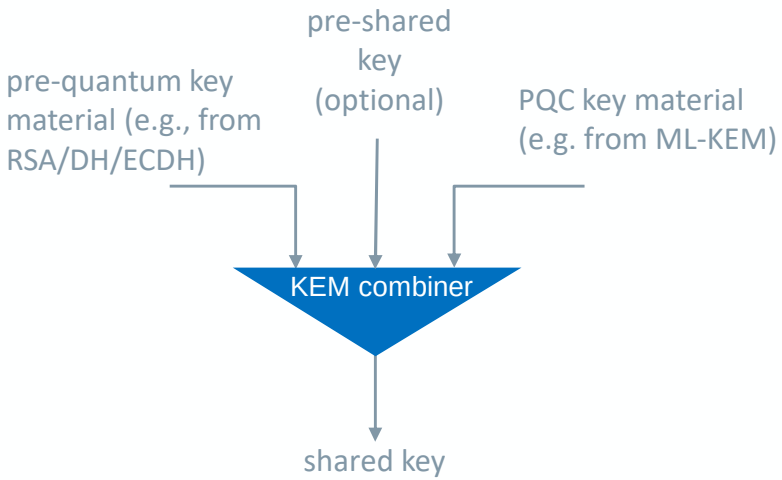
KEM Combiners

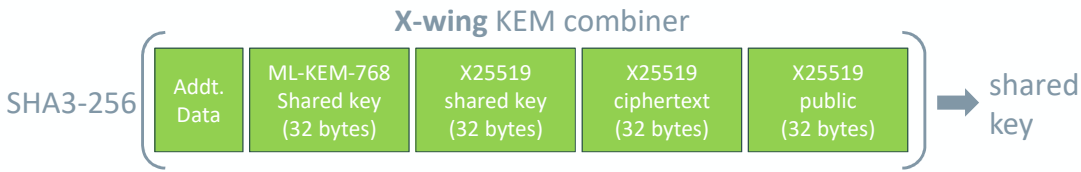
Use a secure (standardized) KEM combiner with a validated security proof, do not simply concatenate or XOR the symmetric keys resulting from the pre- and post-quantum key exchange schemes.

A KEM combiner uses a Key Derivation Function to combine the inputs from the pre- and post-quantum schemes and produce a common shared secret.

KEM Combiner options:

- See ANSSI recommendations for KEM combiners in [1]
- See BSI recommendations for KEM combiners in [5]
- CHEMPAT [3]
- X-wing (specific for Kyber) [2]
- draft-ounsworth-cfrg-kem-combiners [4]





CHEMPAT KEM combiner

```
H = SHA3-256

hybrid_pk = concat(receiver_pk_TKEM, receiver_pk_PQKEM)

hybrid_ct = concat(sender_ct_TKEM, sender_ct_PQKEM)

hybrid_ss = H(concat(ss_TKEM,
                     ss_PQKEM,
                     H(hybrid_ct),
                     H(hybrid_pk),
                     context))
```


[1] ANSSI views on Post-Quantum Cryptography Transition (2023 follow-up)
[2] IETF draft: X-Wing: general-purpose hybrid post-quantum KEM draft-connolly-cfrg-xwing-kem-04
[3] IETF draft: Chempat: Generic Instantiated PQ/T Hybrid Key Encapsulation Mechanisms
[4] IETF draft: Combiner function for hybrid key encapsulation mechanisms (Hybrid KEMs)
[5] https://pkic.org/events/2023/pqc-conference-amsterdam-nl/pkic-pqcc_stephan-ehlen_bsi_post-quantum-policy-and-roadmap-of-the-bsi.pdf
[6] Hybrid key exchange in TLS 1.3

Ounsworth CFRG combiner

Combiner function for hybrid key encapsulation mechanisms (Hybrid KEMs) (ietf.org) [16]

```
combined_secret = KDF(counter + c_dh + rlen_c_dh + key_dh + rlen_key_dh + c_pq_kem +
rlen_c_pq_kem + key_pq_kem + rlen_key_pq_kem + fixedInfo, outputBits)
```

```
    # KDF = KMAC128, with hashSize = 128 bit.
    # KDF = KMAC256, with hashSize = 256 bit.
    # KDF = SHA3-256, with hashSize = 256 bit.
    # KDF = SHA3-512, with hashSize = 512 bit
```

KEM Combiners X-Wing

[X-Wing (iacr.org)]

2 Design

X-Wing private key (2464 bytes):

ML-KEM-768 private key (2400 bytes)	X25519 private key (32 bytes)	X25519 public key (32 bytes)

X-Wing public key (1216 bytes):

ML-KEM-768 public key (1184 bytes)	X25519 public key (32 bytes)

X-Wing ciphertext (1120 bytes):

ML-KEM-768 ciphertext (1088 bytes)	X25519 ciphertext (32 bytes)

X-Wing shared key (32 bytes):

SHA3-256 (

\./ /^\ (6 bytes)	ML-KEM-768 shared key (32 bytes)	X25519 shared key (32 bytes)	X25519 ciphertext (32 bytes)	X25519 public key (32 bytes)

)

The initial X-Wing label is encoded as 6-byte ASCII string "\.//^\".

Figure 1: The X-Wing KEM private key, public key, ciphertext, and shared key.

Elliptic curve choices

Curves	Standard available?	OK for NIST?	OK for BSI?	OK for ANSSI?
x25519	IETF 7748 - Elliptic Curves for Security	Listed as recommended/alternative in NIST SP800-186, allowed for US government use in SP 800-186	Not present in the list of recommended mechanisms, assume not	Not present in the list of recommended mechanisms, assume not,
Brainpool brainpoolP224r1, brainpoolP256r1, brainpoolP320r1, brainpoolP384r1, and brainpoolP512r1	RFC 5639 - Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation	**Yes** for interoperability reasons for ECDSA and ECDH, for 112 bit sec strength or higher; allowed inclusion in FIPS validated products (see Appendix H.1 in NIST SP 800-186**). Not listed in allowed list for US government use** in NIST SP 800-186.	**Yes,** see BSI TR-03111 v 2.1	
secp256k1	[17] SEC 2: Recommended Elliptic Curve Domain Parameters	**Yes,** 128 bit sec strength (see Appendix H.2 in NIST SP 800-186), allowed for blockchain		
P-256 P-384 P-521	Yes	Yes	Yes	Yes
Curve 448	IETF 7748 - Elliptic Curves for Security	Yes, recommended in NIST SP 800-186		

Certification: NIST FIPS 140-3

FIPS 140-3 (Federal Information Processing Standard Publication 140-3) is a standard maintained by NIST (National Institute of Standards and Technology) which specifies the ***security requirements for a cryptographic module utilized within a security system protecting sensitive but unclassified information*** in computer and telecommunication systems (including voice systems) as defined in Section 5131 of the Information Technology Management Reform Act of 1996, (Public Law 104-106) and the Federal Information Security Management Act of 2002 (Public Law 107-347).

FIPS 140-3 supersedes FIPS140-2. FIPS 140-3 aligns with **ISO/IEC 19790:2012(E)** and **ISO/IEC 24759:2017** which form the technical requirements and the basis for the testing methods of FIPS.

SDLS EP PDUs in CCSDS Packets in CCSDS Frames

4.2.1 TRANSFER OF EXTENDED PROCEDURES SERVICE PDU OVER THE SPACE LINK

4.2.1.1 For transport of SDLS Extended Procedures PDUs on the TC link (TC data link protocol), the MAP packet service with a dedicated MAP shall be used (see references [6] and [8]).

4.2.1.2 For transport of SDLS Extended Procedures PDUs on the TM downlink (data link protocols), the VC packet service shall be used (see references [4] and [8]).

4.2.1.3 For transport of SDLS Extended Procedures PDUs using AOS (link or downlink), the VC packet service shall be used (see references [5] and [8]).

4.2.1.4 For transport of SDLS Extended Procedures PDUs using USLP (link or downlink), the MAP packet service shall be used (see references [9] and [8]).

NOTE – Grouping EP PDUs in one single packet is a way of ensuring that the related PDUs are transferred together.

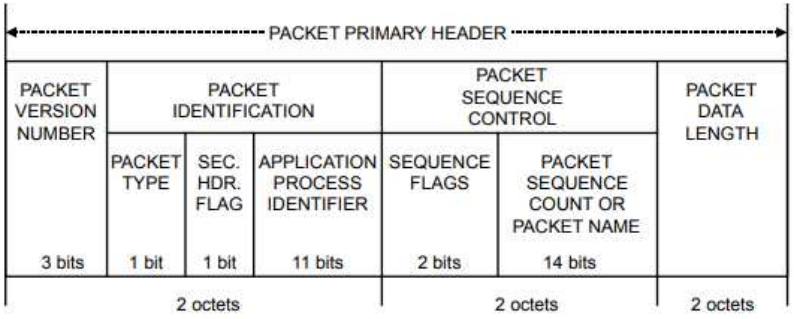


Figure 4-2: Packet Primary Header

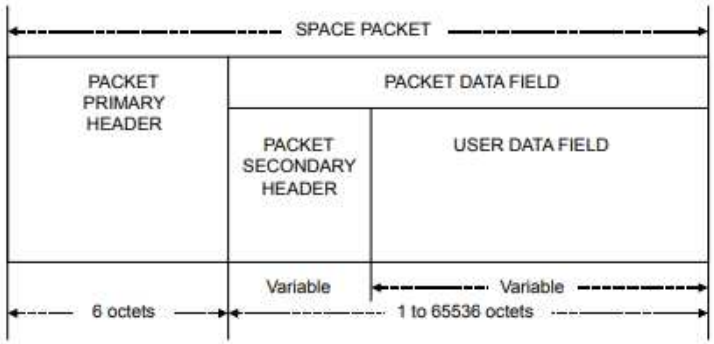


Figure 4-1: Space Packet Structural Components

[8] Space Packet Protocol. Issue 1. Recommendation for Space Data System Standards (Blue Book), CCSDS 133.0-B-1. Washington, D.C.: CCSDS, September 2003.

SDLS EXTENDED PROCEDURES

SDLS Extended Procedures commands and reports share a common message format based on the 'TLV' concept. The Tag field uniquely identifies the command or the report. The Length field indicates the length of the Value field (may be zero). The (optional) Value field contains additional data pertaining to the message

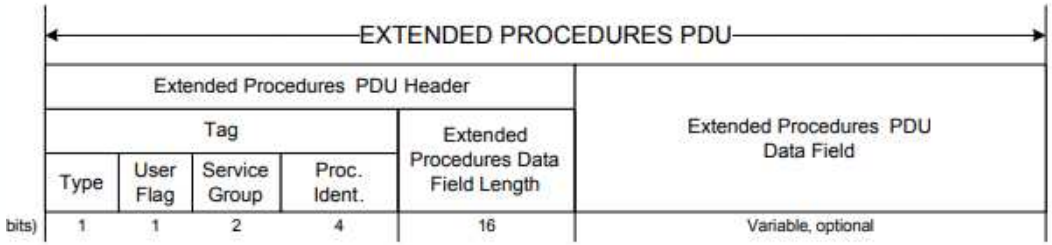


Figure 5-2: Extended Procedures PDU

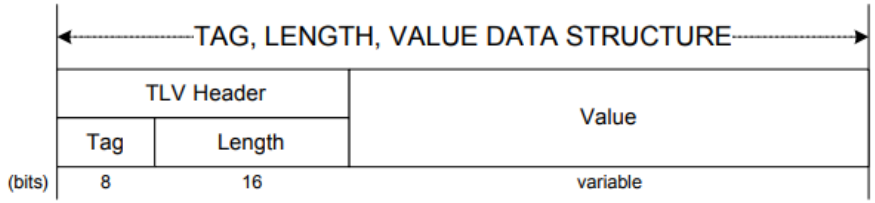


Figure 5-1: TLV Format Specification

SDLS EXTENDED PROCEDURES

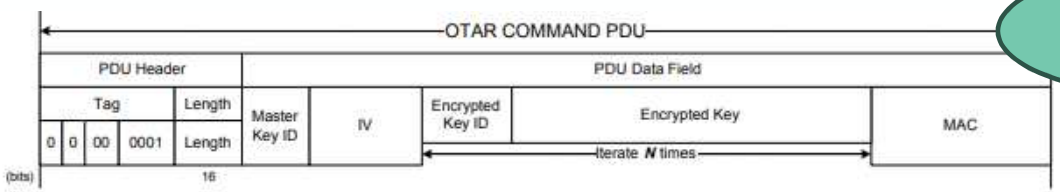


Figure 5-3: OTAR Command PDU

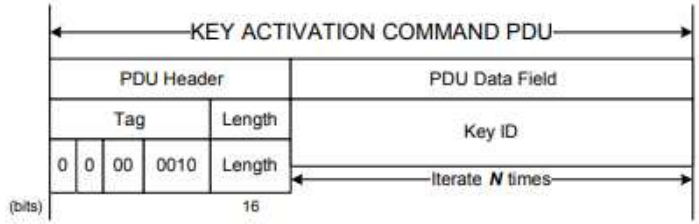


Figure 5-4: Key Activation Command PDU

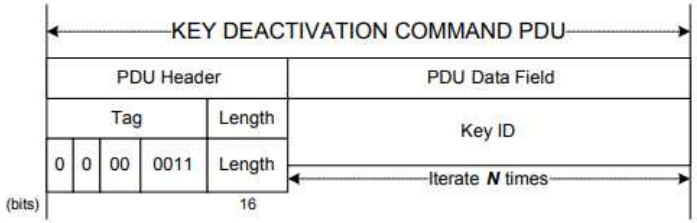


Figure 5-5: Key Deactivation Command PDU

Add AKE PDUs under Key Management Service

Table 5-1: Extended Procedures PDU Header Values

	Assignment	Service Group
	OTAR	00 (Key Management)
0010	Key Activation	00 (Key Management)
0011	Key Deactivation	00 (Key Management)
0100	Key Verification	00 (Key Management)
0110	Key Destruction	00 (Key Management)
0111	Key Inventory	00 (Key Management)
0001	Create SA	01 or 10 (SA Management)
0110	Rekey SA	01 or 10 (SA Management)
1011	Start SA	01 or 10 (SA Management)
1110	Stop SA	01 or 10 (SA Management)
1001	Expire SA	01 or 10 (SA Management)
0100	Delete SA	01 or 10 (SA Management)
1010	Set Anti-Replay Sequence Number	01 or 10 (SA Management)
0101	Set Anti-Replay Sequence Number Window	01 or 10 (SA Management)
0000	Read Anti-Replay Sequence Number	01 or 10 (SA Management)
1111	SA Status Request	01 or 10 (SA Management)
0001	Ping	11 (Security Monitoring & Control)
0010	Log Status Request	11 (Security Monitoring & Control)
0011	Dump Log	11 (Security Monitoring & Control)
0100	Erase Log	11 (Security Monitoring & Control)
0101	Self-Test	11 (Security Monitoring & Control)
0111	Reset Alarm Flag	11 (Security Monitoring & Control)

SDLS EXTENDED PROCEDURES

New SDLS EP PDUs will be required to be defined to support a key negotiation protocol.

AKE Header							
AKE Protocol Version Number	Initiator ID	Responder ID	Session ID	Timestamp	AKE Message Type	AKE Fragment Number	AKE Fragment size
4	32 TBC	32 TBC	32 TBC	64 TBC	4 TBC	TBC	TBC

Protocol Version Number	Indicates a protocol configuration (baseline vs alternate and 3KEM vs KEM-Sign) as well as the explicit instantiation of the crypto primitives/ parameter sizes
Initiator ID	ID of the entity initiating the AKE handshake
Responder ID	ID of the entity responding to the AKE handshake initiation message
Session ID	unique identifier for the session generated by the initiator
Timestamp	timestamp associated with the generation of the first negotiation message (Message Type 1)
KN Message Type	message identifier, for 3-KEM there are 3 message types

Add total number fragments TBD

Location of AKE PROTOCOL in the CCSDS stack

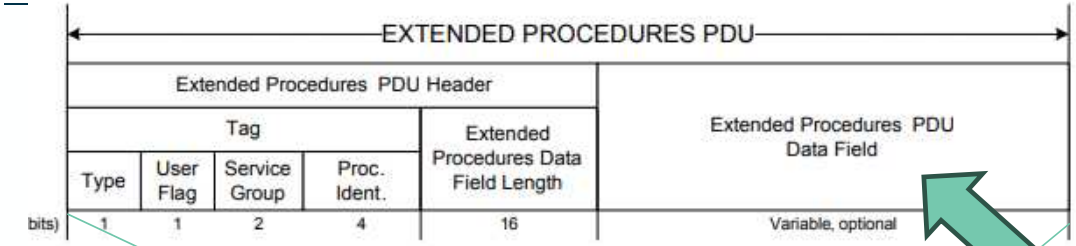


Figure 5-2: Extended Procedures PDU

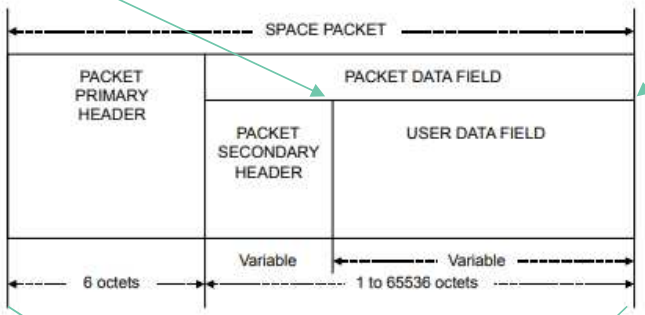
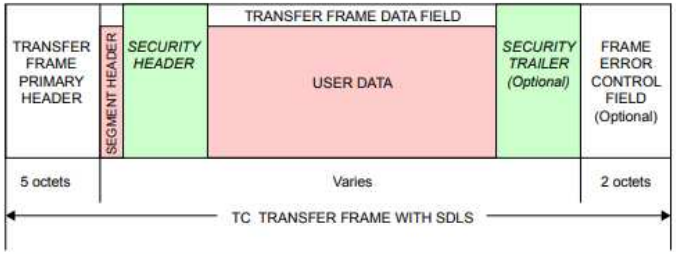


Figure 4-1: Space Packet Structural Components



AKE Header							
AKE Protocol Version Number	Initiator ID	Responder ID	Session ID	Timestamp	AKE Message Type	AKE Fragment Number	AKE Fragment size
	432 TBC	32 TBC	32 TBC	64 TBC	4 TBC	TBC	TBC

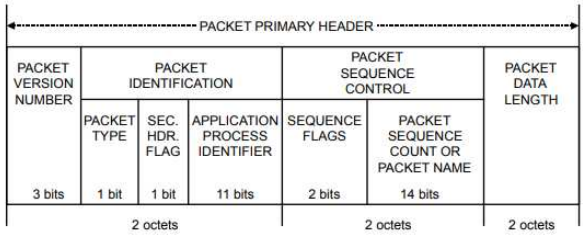


Figure 4-2: Packet Primary Header

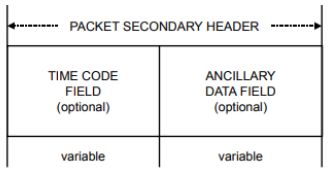


Figure 4-3: Packet Secondary Header

BACKUP SLIDES

Plan for Transitioning to Quantum-Resistant Crypto

Source: M-23-02 (whitehouse.gov)

EXECUTIVE OFFICE OF THE PRESIDENT
OFFICE OF MANAGEMENT AND BUDGET
WASHINGTON, D.C. 20503

THE DIRECTOR

November 18, 2022

M-23-02

MEMORANDUM FOR THE HEADS OF EXECUTIVE DEPARTMENTS AND AGENCIES

FROM: Shalanda D. Young
Director

SUBJECT: Migrating to Post-Quantum Cryptography

This memorandum provides direction for agencies to comply with National Security Memorandum 10 (NSM-10), *on Promoting United States Leadership in Quantum Computing While Mitigating Risk to Vulnerable Cryptographic Systems* (May 4, 2022).[1]

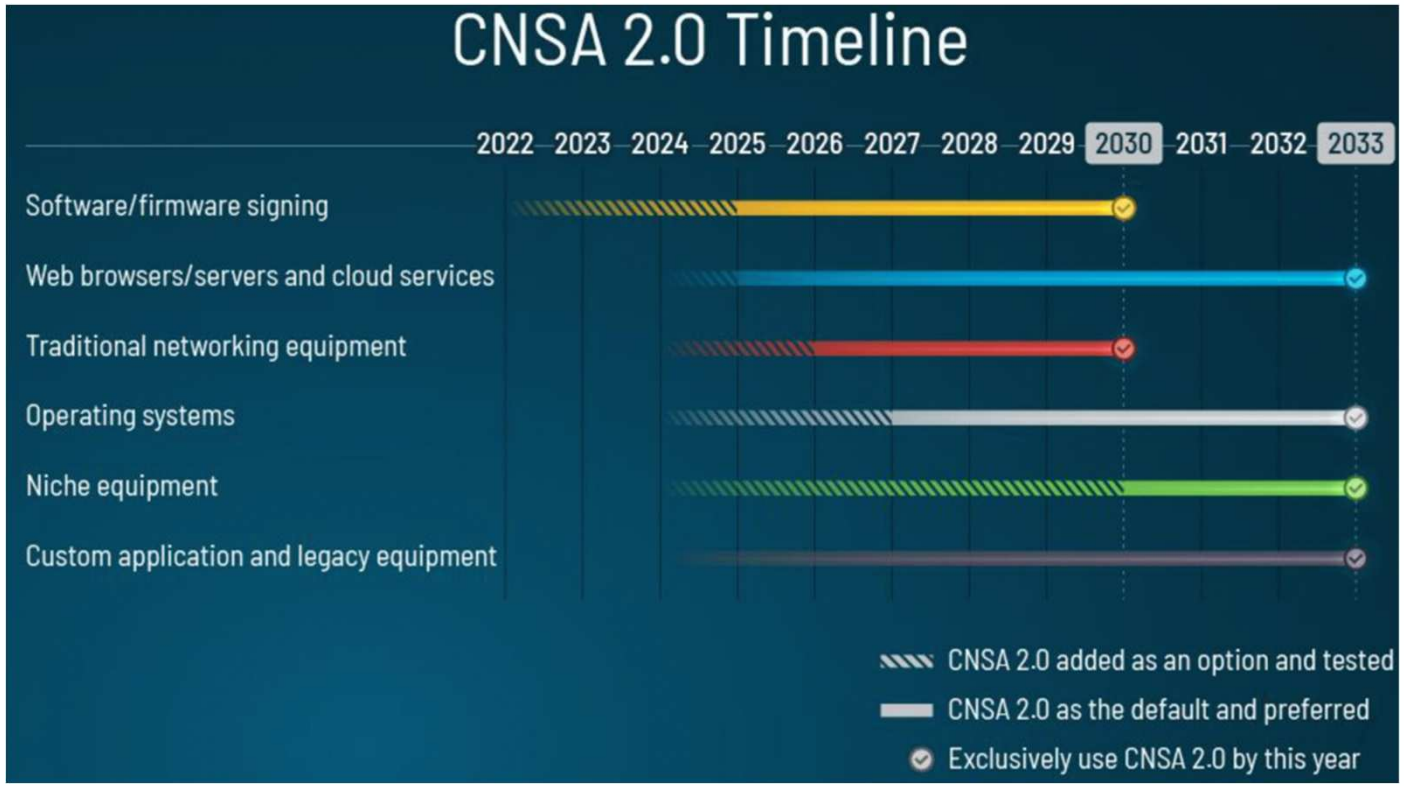
WH.GOV

MAY 04, 2022

National Security Memorandum on Promoting United States Leadership in Quantum Computing While Mitigating Risks to Vulnerable Cryptographic Systems

"the United States must prioritize the transition of cryptographic systems to quantum-resistant cryptography, with the goal of mitigating as much of the quantum risk as feasible ***by 2035****"*

Timeline for Transitioning to Quantum-Resistant Crypto



Source: [The Commercial National Security Algorithm Suite 2.0 and Quantum Computing FAQ]

CNSS (Committee on National Security Systems) Commercial National Security Algorithm Suite (CNSA Suite) is a set of cryptographic algorithms recommended for use by U.S. government agencies **to protect classified** and **sensitive information**.

It is specifically designed to provide security for information that will remain sensitive and require protection beyond the year 2030.

CNSA 2.0 replaces the earlier CNSA 1.0, which included older algorithms such as RSA, ECDSA and ECDH.

Plan for Transitioning to Quantum-Resistant Crypto

Source:[Recommendation on a Coordinated Implementation Roadmap for the transition to Post-Quantum Cryptography | Shaping Europe's digital future (europa.eu)]

EUROPEAN COMMISSION

Brussels, 11.4.2024
C(2024) 2393 final

COMMISSION RECOMMENDATION

of 11.4.2024

on a Coordinated Implementation Roadmap for the transition to Post-Quantum Cryptography

The future potential development of quantum computers capable of breaking today's encryption makes it necessary for Europe to look for stronger safeguards, ensuring the protection of sensitive communications and the long-term integrity of confidential information, i.e., by switching to Post-Quantum Cryptography as swiftly as possible. This new type of cryptography will remove the known vulnerabilities of current asymmetric cryptography and enhance the robustness against the threats posed by the malicious use of quantum computers.

This Commission Recommendation encourages Member States to develop a comprehensive strategy for the adoption of Post-Quantum Cryptography, to ensure a coordinated and synchronized transition among the different Member States and their public sectors. The strategy should define clear goals, milestones, and timelines resulting in the definition of a joint Post-Quantum Cryptography Implementation Roadmap. This should lead to the deployment across the Union of Post-Quantum Cryptography technologies into existing public administration systems and critical infrastructures via hybrid schemes that may combine Post-Quantum Cryptography with existing cryptographic approaches or with Quantum Key Distribution.

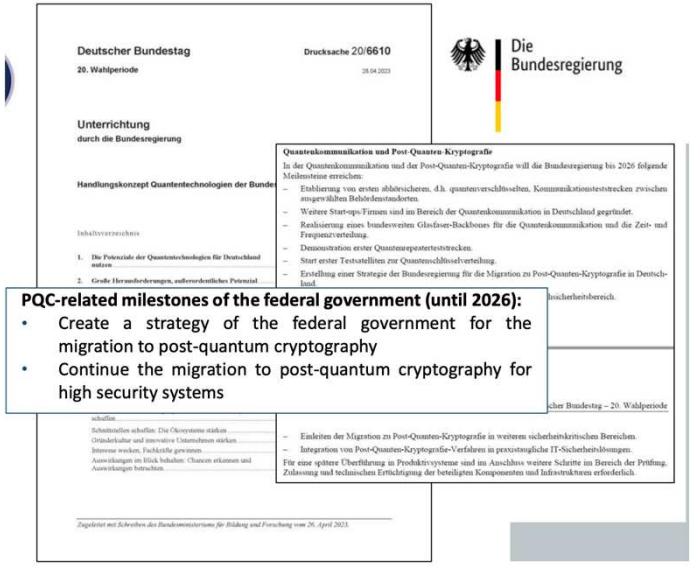
The Post-Quantum Cryptography Coordinated Implementation Roadmap should be available after a period of two years following the publication of this Recommendation, which will be followed by the development and further adaptation of Post-Quantum Cryptography transition plans of individual Member States, in accordance with the principles set out in the Post-Quantum Cryptography Coordinated Implementation Roadmap.

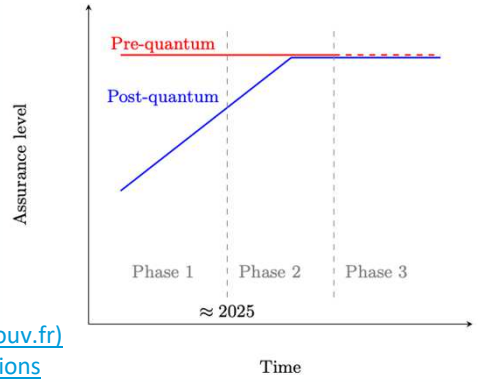
Timeline for Transitioning to PQC: 2030

For national security systems, BSI works under the hypothesis that cryptographically relevant quantum computers will be available in the early 2030s [BT19/25208], [BT19/26340]. It should be emphasised that this statement is **not to be** understood as a forecast of the availability of quantum computers, but rather represents a benchmark for risk assessment. BSI has therefore initiated the shift to quantum-safe cryptography in line with the federal government's framework programme "Quantum technologies - from basic research to market" [BMBF18].

Taking into account the working hypothesis that a cryptographically relevant quantum computer will be available in the early 2030s, BSI believes that it is already urgently necessary to take appropriate measures to switch to quantum-safe schemes. This urgency alone makes the migration to post-quantum cryptography, the standardisation of which is already well advanced in the NIST process, a clear priority from BSI's point of view. Furthermore, post-quantum algorithms are much more flexible, as they can be implemented in existing infrastructure, they are more cost-effective, do not require secret pre-distributed keys and offer end-to-end security.

ANSSI recommends introducing post-quantum defense-in-depth as soon as possible for security products aimed at offering a long-lasting protection of information (until after 2030) or that will potentially be used after 2030 without updates.





"In 2030 quantum computing resources will be made more widely available, allowing threat actors to use quantum computing to attack existing deployments of public key cryptography. Likewise, there is a risk that threat actors collect sensitive encrypted data now, aiming to decrypt it once quantum computing is accessible."

ENISA, March 2024

Source (FR): ANSSI views on the Post-Quantum Cryptography transition | ANSSI (cyber.gouv.fr)
Source (DE): BSI: Quantum-safe cryptography, current developments and recommendations
Source (DE): Post-Quantum Policy and Roadmap of the BSI

US NSA on PQC Transitioning: CNSA 1.0 to CNSA 2.0

NATIONAL SECURITY AGENCY
CENTRAL SECURITY SERVICE

Commercial National Security Algorithm (CNSA) Suite

Rapid and secure information sharing is important to protect our Nation, its citizens and its interests. Strong cryptographic algorithms are vital tools that contribute to our national security and help address the need for secure, interoperable communications. The National Security Agency (NSA) is responsible for approving solutions for protecting National Security Systems (NSS). Many systems in the NSS community are planned over decade timescales, have very long lifetimes after deployment, and are used to protect data that requires confidentiality for years beyond that.

Since 2005, a specific set of elliptical curve based algorithms, the CNSA cryptographic algorithms as specified by the National Institute of Standards and Technology (NIST), have been used by NSA in solutions approved for protecting classified and unclassified NSS. After observing the past decade of progress in quantum computing research, NSA endorses the increasing consensus that quantum computers will pose a threat in the future and that protocols using public key algorithms in the market place today will eventually need to be addressed. Given the longevity and unique nature of NSS and the costs of converting our existing public key based infrastructure to new algorithms, it is prudent to reconsider our strategic approach to the protection of data on NSS now.

To ensure the confidentiality of our customers' long life data, NSA is planning for an upcoming transition to quantum resistant algorithms and encouraging the design and analysis of quantum resistant public key algorithms. NSA plans to support NIST and other external standards bodies in developing standards for quantum resistant cryptography. In 2015, NSA announced a revised set of cryptographic algorithms that can be used to protect NSS while the algorithms that would be part of a quantum resistant suite are developed. For symmetric algorithms, options exist today that will be sufficient well into the future and beyond the development of a quantum computer. In the public key space, the intent is to give more flexibility to vendors and our customers in the present as we prepare for a quantum safe future.

Commercial cryptography approved to protect NSS systems up to the TOP SECRET level

Algorithm	Function	Specification	Parameters
Advanced Encryption Standard (AES)	Block cipher used for information protection	FIPS Pub 197	Use 256 bit keys
Elliptic Curve Diffie-Hellman (ECDH) Key Exchange	Asymmetric algorithm used for key establishment	NIST SP 800-56A	Use Curve P-384
Elliptical Curve Digital Signature Algorithm (ECDSA)	Asymmetric algorithm used for digital signatures	FIPS Pub 186-4	Use Curve P-384
Secure Hash Algorithm (SHA)	Used for computing a condensed representation of information	FIPS Pub 180-4	Use SHA-384
Diffie-Hellman (DH) Key Exchange	Algorithm used for key establishment	IETF RFC 3526	Minimum 3072-bit modulus
RSA	Algorithm used for key establishment	NIST SP 800-56B rev 1	Minimum 3072-bit modulus
RSA	Asymmetric algorithm used for digital signatures	FIPS PUB 186-4	Minimum 3072-bit modulus

CNSA 1.0

NSA | CNSA Suite 2.0 and Quantum Computing FAQ

Q: What is the Commercial National Security Algorithm Suite 2.0 (CNSA 2.0)?

A: CNSA 2.0 is the suite of QR algorithms approved for eventual NSS use. The following table lists the algorithms and their functions, specifications, and parameters.

Table: Commercial National Security Algorithm Suite 2.0

Algorithm	Function	Specification	Parameters
Advanced Encryption Standard (AES)	Symmetric block cipher for information protection	FIPS PUB 197	Use 256-bit keys for all classification levels.
ML-KEM (aka CRYSTALS-Kyber)	Asymmetric algorithm for key establishment	FIPS PUB 203	Use Category 5 parameter, ML-KEM-1024, for all classification levels.
ML-DSA (aka CRYSTALS-Dilithium)	Asymmetric algorithm for digital signatures in any use case, including signing firmware and software	FIPS PUB 204	Use Category 5 parameter, ML-DSA-87, for all classification levels.
Secure Hash Algorithm (SHA)	Algorithm for computing a condensed representation of information	FIPS PUB 180-4	Use SHA-384 or SHA-512 for all classification levels.
Leighton-Micali Signature (LMS)	Asymmetric algorithm for digitally signing firmware and software	NIST SP 800-208	All parameters approved for all classification levels. LMS SHA-256/192 is recommended.
Xtended Merkle Signature Scheme (XMSS)	Asymmetric algorithm for digitally signing firmware and software	NIST SP 800-208	All parameters approved for all classification levels.

CNSA 2.0

Q: Where should CNSA 2.0 algorithms be used?

A: CNSA 2.0 algorithms will be required for all products that employ public-standard algorithms in NSS, whether a future design or currently fielded. Any usage of Suite B or CNSA 1.0 algorithms will be required to transition to CNSA 2.0 usage. The Timeframe

Q: Can I use SLH-DSA (aka SPHINCS+) to sign software?

A: While SLH-DSA is hash-based, it is not part of CNSA and is not approved for any use in NSS.

Timeframe

Q: What timeframe information can NSA provide for adoption of CNSA 2.0?

A: NSA intends that all NSS will be quantum-resistant by 2035, in accordance with the goal espoused in NSM-10. NSA relies upon NIST-approved commercial cryptography for commercial solutions. After NIST has finalized the standards associated with CNSA 2.0, NSA will update CNSSP 15.

New cryptographic developments will be required to support CNSA 2.0 algorithms as an option once appropriate standards for the given technology are in place. All appropriate system components should be configured to prefer CNSA 2.0 algorithms. As products mature, those components should be configured to accept only CNSA 2.0 algorithms.

NSA will provide guidance and updated protection profiles as industry develops the appropriate standards because product lines may develop at different speeds. CNSA 1.0 algorithms will continue to be used until current solutions can operate in a CNSA 2.0 mode. NSA's current view on timing is as follows:

- **Software- and firmware-signing:** begin transitioning immediately, support and prefer CNSA 2.0 by 2025 where available, *exclusively* use CNSA 2.0 by 2030.
- **Web browsers/servers and cloud services**: support and prefer CNSA 2.0 by 2025, *exclusively*[6] use CNSA 2.0 by 2033.
- **Traditional networking equipment (e.g., virtual private networks, routers):** support and prefer CNSA 2.0 by 2026, *exclusively* use CNSA 2.0 by 2030.
- **Operating systems**: support and prefer CNSA 2.0 by 2027, *exclusively* use CNSA 2.0 by 2033.
- **Niche equipment (e.g., constrained devices, large public-key infrastructure systems)**: support and prefer CNSA 2.0 by 2030, *exclusively* use CNSA 2.0 by 2033.
- **Custom applications and legacy equipment**: update or replace by 2033.

Source: The Commercial National Security Algorithm Suite 2.0 and Quantum Computing FAQ

Pre- and Post-Quantum Hybridization

Hybrid mechanisms

In the context of PQC, hybridization refers to the combined use of both PQ and pre-quantum algorithms in the same protocol:

- hybridization of KEMs (key exchange/agreement)
- hybridization of signatures (signing the key exchange/agreement)

Hybrid mechanisms benefit from the resistance of the pre-quantum algorithm against classical attackers, and from the resistence of the post-quantum algorithm against quantum attackers.

- Hybridization of signatures is easy
- Hybridization of KEMs is more complex, it requires secure KEM combiners

Not to be confused with hybrid encryption (schemes that combine assymmetric and symmetric algorithms)

To Hybridize Or Not to Hybridize with Pre-Quantum

US NSA

NSA | CNSA Suite 2.0 and Quantum Computing FAQ

solutions can consist of many traditional or QR algorithms. "Component algorithms" are individual algorithms used in a hybrid solution.

Q: What is NSA's position on the use of hybrid solutions?

A: NSA has confidence in CNSA 2.0 algorithms and will not require NSS developers to use hybrid certified products for security purposes. Product availability and interoperability requirements may lead to adopting hybrid solutions.

NSA recognizes that some standards may require using hybrid-like constructions to accommodate the larger sizes of CRQC algorithms and will work with industry on the best options for implementation.

Q: What complications can using a hybrid solution introduce?

A: Hybrids add complexity to protocols, as designers need to incorporate additional negotiation and error handling and implementers need to modify API's and testing.

Rather than ease the transition to quantum resistance, hybrid deployments introduce additional interoperability concerns, now that all algorithms plus the method of hybridization must be features common to all parties to a communication. Similarly, hybrid deployments add a second transition later as users eventually move away from classical algorithms in the future.

At the same time, hybrid solutions make the implementations more complex, so one must balance the risk of flaws in an increasingly complex implementation with the risk of a cryptanalytic breakthrough. Because more security products fail due to implementation or configuration errors than failures in their underlying cryptographic algorithms, spending limited resources to add cryptographic complexity can at times weaken security rather than improve it.

Where NSA recognizes a need to support a hybrid solution, extensive work will be performed to ensure that it can be safely implemented, including engineering to a high degree of robustness, and facilitation to a straightforward transition to QR-only solutions.

Source: CSI_CNSA_2.0_FAQ_.PDF (defense.gov)

FR NSA (ANSSI)

Acknowledging the immaturity of PQC is important: ANSSI will not endorse any direct drop-in replacement of currently used algorithms in the short/medium term. However, this immaturity should not serve as an argument for postponing the first deployments. ANSSI encourages all industries to initiate in the next months a gradual overlap transition in order to progressively increase trust on the post-quantum algorithms and their implementations while ensuring no security regression as far as classical (pre-quantum) security is concerned.

What is the recommended post-quantum transition roadmap?

To support a gradual transition, ANSSI encourages the following 3-phase roadmap (see below for a detailed description):

- Phase 1 (today): hybridation to provide some additional post-quantum *defense-in-depth* to the pre-quantum security assurance.
- Phase 2 (not earlier than 2025): hybridation to provide *post-quantum security assurance* while avoiding any pre-quantum security regression.
- Phase 3 (probably not earlier than 2030): optional standalone post-quantum cryptography.

Source: ANSSI views on the Post-Quantum Cryptography transition | ANSSI (cyber.gouv.fr)

Start date: ⩾ **2025**.

- Hybridisation **remains mandatory**.
- Post-quantum safety becomes **mandatory** in some cases.

Source: ANSSI Plan for Post-Quantum Transition

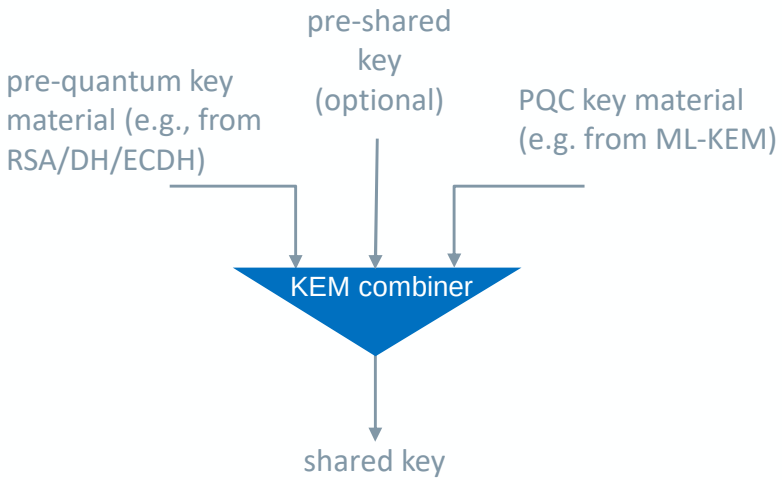
KEM Combiners

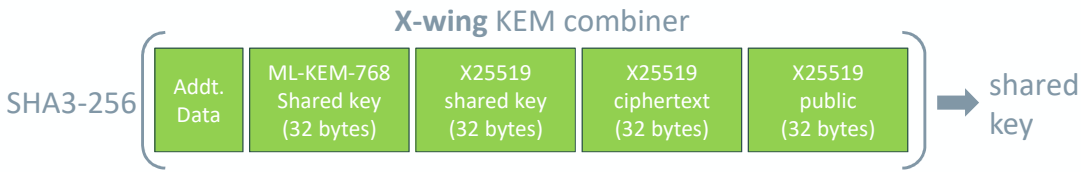
Use a secure (standardized) KEM combiner with a validated security proof, do not simply concatenate or XOR the symmetric keys resulting from the pre- and post-quantum key exchange schemes.

A KEM combiner uses a Key Derivation Function to combine the inputs from the pre- and post-quantum schemes and produce a common shared secret.

KEM Combiner options:

- See ANSSI recommendations for KEM combiners in [1]
- See BSI recommendations for KEM combiners in [5]
- CHEMPAT [3]
- X-wing (specific for Kyber) [2]
- draft-ounsworth-cfrg-kem-combiners [4]





CHEMPAT KEM combiner

```
H = SHA3-256

hybrid_pk = concat(receiver_pk_TKEM, receiver_pk_PQKEM)

hybrid_ct = concat(sender_ct_TKEM, sender_ct_PQKEM)

hybrid_ss = H(concat(ss_TKEM,
                     ss_PQKEM,
                     H(hybrid_ct),
                     H(hybrid_pk),
                     context))
```


[1] ANSSI views on Post-Quantum Cryptography Transition (2023 follow-up)
[2] IETF draft: X-Wing: general-purpose hybrid post-quantum KEM draft-connolly-cfrg-xwing-kem-04
[3] IETF draft: Chempat: Generic Instantiated PQ/T Hybrid Key Encapsulation Mechanisms
[4] IETF draft: Combiner function for hybrid key encapsulation mechanisms (Hybrid KEMs)
[5] https://pkic.org/events/2023/pqc-conference-amsterdam-nl/pkic-pqcc_stephan-ehlen_bsi_post-quantum-policy-and-roadmap-of-the-bsi.pdf
[6] Hybrid key exchange in TLS 1.3

Cryptographic Agility

- A product is crypto-agile if it includes the possibility of updating its cryptographic algorithms after it has been deployed, without recalling it.
- Cryptographic agility is very relevant in the context of emerging attacks and updates of cryptographic standards.

NIST SP 800-131A Rev 2

2019: NIST encourages implementers to plan for **cryptographic agility** to facilitate transitions to quantum-resistant algorithms where needed in the future.

"In practice, cryptoagility also means that in addition to the possibility of patching, products could include an extra surface for allowing potential updates in order to react to upcoming cryptographic recommendations and standard updates. Even though updates of the cryptographic algorithms should be much less frequent than patches, the cryptoagility feature is non-trivial to implement due to the need for retro-compatibility and the potential requirement for additional security visas if the product is certified. However, as the motivation for cryptoagility is very relevant nowadays, ANSSI believes that cryptoagility features should be taken into account during the benefit/risk analysis of future products"

.

Technical position paper – ANSSI Views on the Post-Quantum Cryptography Transition

NCSC Recommendations

Choosing algorithms and parameters for your use cases

The following table gives the NCSC recommended algorithms, their functions, and specifications:

Algorithm	Function	Specification
ML-KEM	Key establishment algorithm	NIST FIPS 203
ML-DSA	Digital signature algorithm	NIST FIPS 204
SLH-DSA	Digital signature algorithm for use cases such as signing firmware and software	NIST FIPS 205
LMS	Digital signature algorithm for use cases such as signing firmware and software	NIST SP 800-208
XMSS	Digital signature algorithm for use cases such as signing firmware and software	NIST SP 800-208

Source: NCSC whitepaper: Next steps in preparing for post-quantum cryptography

National Cyber Security Centre
Ministry of Justice and Security

Source: NCSC: Guidelines for quantum-safe transport-layer encryption

Guidelines for quantum-safe transport-layer encryption

These guidelines are written for an audience of architects responsible for specifying cryptographic requirements. They can also be used in R&D and prototyping as well as for contract negotiations. For a more general introduction, see NLNCSA's brochure and our own factsheet. For further details, follow NIST, ETSI, IETF, and ISO standardisation efforts and read publications by ENISA and TNO.

Our recommendations target the early adopters who follow our advice to apply quantum-safe cryptography to ensure long-term confidentiality against store-and-decrypt attacks. Signatures are not part of these guidelines as they are not vulnerable to such attacks. The guidelines recommend hybrid key exchange to mitigate potential vulnerabilities in novel post-quantum algorithms and implementations. Besides a list of algorithms and recommended parameters, this document also contains some questions to ask when choosing implementations.

Combine traditional algorithms with quantum-safe key encapsulation

Key agreement should rely on multiple algorithms. For other purposes, apply established methods. You should use algorithms that have stood the test of time and that are future-proof. However, post-quantum cryptography is a new and fast-moving field. As such, ensure that you can quickly replace any algorithms and implementations that you rely on – so-called cryptographic agility.

Use all of the following standard cryptographic algorithms*:

— AES-256-GCM or ChaCha20-Poly1305 (for bulk encryption)
— SHA-256 or SHA3-256 (for hashing, viz. key derivation)
— ECDSA-secp256r1 or Ed25519 (for certificate verification)
— ECDH-secp256r1 or ECDH-X25519 (for key exchange)

Combine these with at least one of the following quantum-safe key encapsulation mechanisms:

— FrodoKEM at level 3+ (frodokem976 or higher)
— Classic McEliece at level 3+ (mceliece460896 or higher)
— CRYSTALS-Kyber at level 5 (kyber1024)

Apply one of the following key derivation mechanisms to get a hybrid construction:

— Concatenation of shared secrets (as specified by NIST in SP 800-56C Rev. 2) using HKDF-256
— Cascade of shared secrets (as specified by ETSI in TS 103 744) using HKDF-256

Alternatively, protocol stacking is another possible approach, where at least one of the protocols supports the standard cryptographic algorithms given above and where one or more protocols provide a quantum-safe key encapsulation mechanism. In a situation where TLS is used as the protocol that implements standard cryptographic algorithms, note our guidelines for TLS.

* Longer hash functions and elliptic curves of the same type can be used, e.g. of 384 or 512 bits. Note that other AEAD modes which use a synthetic IV are less brittle, but also less performant.

Only choose a KEM with shorter keys following a risk assessment

For FrodoKEM and Classic McEliece, a minimum key length is specified based on previous NLNCSA, BSI, and ANSSI recommendations; level 5 parameters can also be used. Note that using Kyber or another structured-lattice algorithm is riskier. As such, longer keys are recommended until enough confidence has been gained in them. However, this does not exclude novel attacks that may be discovered on structured lattices. When shorter keys are used – as with kyber768 and sntrup761 – a risk assessment should be carried out with a decision taken by the asset owner, which should be recorded, tracked, and regularly revisited based on the level of uncertainty.†

As noted in our factsheet on migration planning, a clear view of information assets and data flows helps to feed risk assessment decisions. Additionally, the involvement of professionals in the area of applied cryptography may be valuable, especially when looking into the use of riskier parameters and when faced with specific constraints. Either way, consider applying mitigating measures such as over-provisioning systems (so that they support stronger but heavier cryptography) and testing that implementations can be replaced when necessary. This will also be useful in case any future standard ends up deviating from the most recent specifications.‡

Use production-grade implementations that have been suitably vetted

Due to the critical role that cryptography plays in securing information, implementations should be mature and assured commensurate with the sensitivity of the data involved. This applies both to traditional cryptography as well as to quantum-safe cryptography – although the latter is relatively immature, especially when it comes to implementations. Given this immaturity, it is vital that a system's architecture enables agility: easily replaceable implementations are a prudent safety net. Even so, using production-grade and vetted implementations is an important principle to aim for.

Besides implementation quality, there are various other aspects that require attention. Contextual factors include stakeholder acceptance of performance and latency overheads, their risk appetite, available budget, switching costs, and interoperability concerns. Such considerations influence how trade-offs are made. By addressing these issues proactively during the life-cycle of existing and future deployments, it will be easier to achieve flexible systems that remain aligned to standards.

Colophon

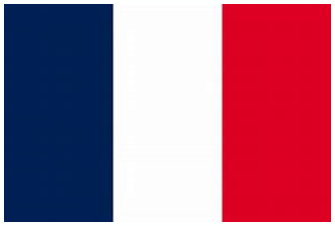
The following organisations and individuals have contributed to these guidelines: I&W, NLNCSA, Andreas Hülsing (TU/e), Anthony Hu (wolfSSL), Bas Westerbaan (Cloudflare), Marc Stevens (CWI).

Version 1.0, May 2022. This information is not legally binding. It is to be updated yearly.

National Cyber Security Centre (NCSC)
Turfmarkt 147, 2511 DP, The Hague
info@ncsc.nl
070 751 5555

† The same applies when choosing symmetric key lengths, e.g. AES-128, AES-192, or AES-256.
‡ Note that if such changes are made, experts should be given time to study these modifications.

ANSSI on FrodoKEM



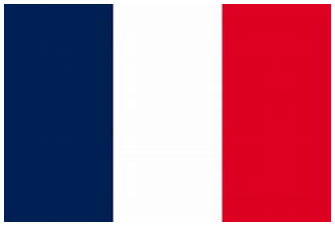
"While few exceptions are expected in practice, at least for mainstream cryptography, an algorithm that is not a NIST standard, but that is demonstrably stronger that a NIST standard, could constitute such an exception. For example, a developer should be able to obtain a security visa for a product implementing an hybrid FrodoKEM whether NIST decides that FrodoKEM will be one of the first PQC standards or not."

Source: Technical Position paper – ANSSI views on Post-Quantum Cryptography transition, March 30, 2022

"FrodoKEM [16]: this scheme is considered as a more conservative variant of CRYSTALS-Kyber. Its security is based on plain (and not module) learning with errors. The unstructured property of the underlying lattice makes it more secure in theory as attacks might potentially leverage the lattice structure of CRYSTALS-Kyber and might be defeated by the absence of structure in the lattice used by FrodoKEM. The price to pay for this more conservative security lies in the performance. FrodoKEM is heavier in terms of key sizes and slower than CRYSTALS-Kyber which makes it a less relevant option for many use cases. However, **ANSSI would encourage including FrodoKEM as a valid and conservative option in high security applications where the resulting performance penalty (in particular in terms of bandwidth) is not prohibitive**. If a designer chooses to include this conservative post-quantum algorithm in a cryptographic product, the recommendations for CRYSTALS-Kyber also apply for FrodoKEM."

Source: ANSSI views on the Post-Quantum Cryptography transition (2023 follow up)

ANSSI and NSA on FALCON

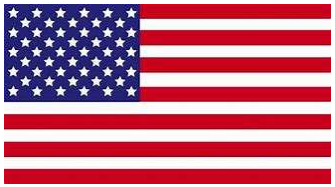


ANSSI

- **Falcon also called FN-DSA** [22]: this signature has been chosen by NIST as a future post-quantum standard. It is a compact and more efficient alternative to CRYSTALS-Dilithium. Since it is based on structured lattice problems, the same warning about the security applies. The design is here based on a more recent framework [8] with a hash-and-sign paradigm on lattices. It is more difficult to implement and needs intermediate variables to be defined as floats.

  For cryptographic products that may include this scheme, ANSSI makes the following recommendations:

  1. It is important to avoid modifying the parameters of the standardized instance. As implementing Falcon is not straightforward, we recommend to pay attention to stick to the design in order to avoid misuse attacks. We should also note that the Gaussian distributions in Falcon play an important role in the security and they should not be replaced.
  2. The parameters are defined for several minimum security levels. We recommend to use the highest level as possible, preferably level-5 (i.e. equivalent to AES-256).
  3. Please note that side-channel countermeasures are particularly difficult to apply and research has proved that side-channel attacks may defeat unprotected implementations of Falcon.

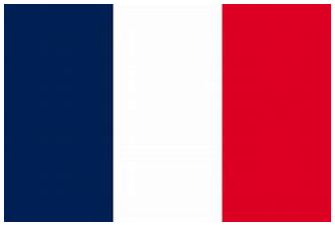


NSA

Q: Why did NSA choose ML-DSA over FN-DSA (aka Falcon)?

A: For NSS, NSA agrees with NIST: ML-DSA is preferred, as FN-DSA seems more susceptible to implementation errors that may affect security. As NIST has prioritized standardizing ML-DSA, it will likely be available sooner.

ANSSI on XMSS and LMS



ANSSI

- **XMSS** [12] / **LMS** [15]: these signature schemes were initially candidates in the NIST post-quantum standardization campaign but in 2018, they have been moved into a separate standardization process. The IETF version of their specification is cited above. These schemes are considered as conservative options because the underlying security hypothesis is very minimalist. Their security proofs are based on the security of hash functions. The particularity of these signatures is their statefulness and the potentially limited number of possible signatures per key pair.

  For contexts where the maximum number of signatures per key pair is restricted and where a state can be carefully stored, typically for software updates for example, ANSSI agrees that XMSS or LMS can be a relevant option, with the following recommendations:

  1. It is important to avoid modifying the parameters of the standardized instance including the underlying hash function.
  2. The parameters should provide the highest security level as possible.
  3. Hybridation (see Section 3 for more information) is optional for this signature.
  4. The state is a very critical data and should be protected in integrity. It should also be protected against replay attacks.

Certificates

Certification Authority (CA)

KeyGen: $k_{pr,\ CA,}\ k_{pub,CA}$

$Cert_{CA} = [(k_{pub,CA}\ ,\ IDca),\ sig_{k_{pr,\ CA}}(k_{pub,CA}\ ,\ IDca)]$

Only for KEM-Sign, N/A to 3KEM

Long term authentication keys for AKE

Satellite

KeyGen: $k_{pr,sat,}\ k_{pub,sat}$

REQUEST $_{k_{pr,sat}}(k_{pub,sat}\ ,\ IDsat)$

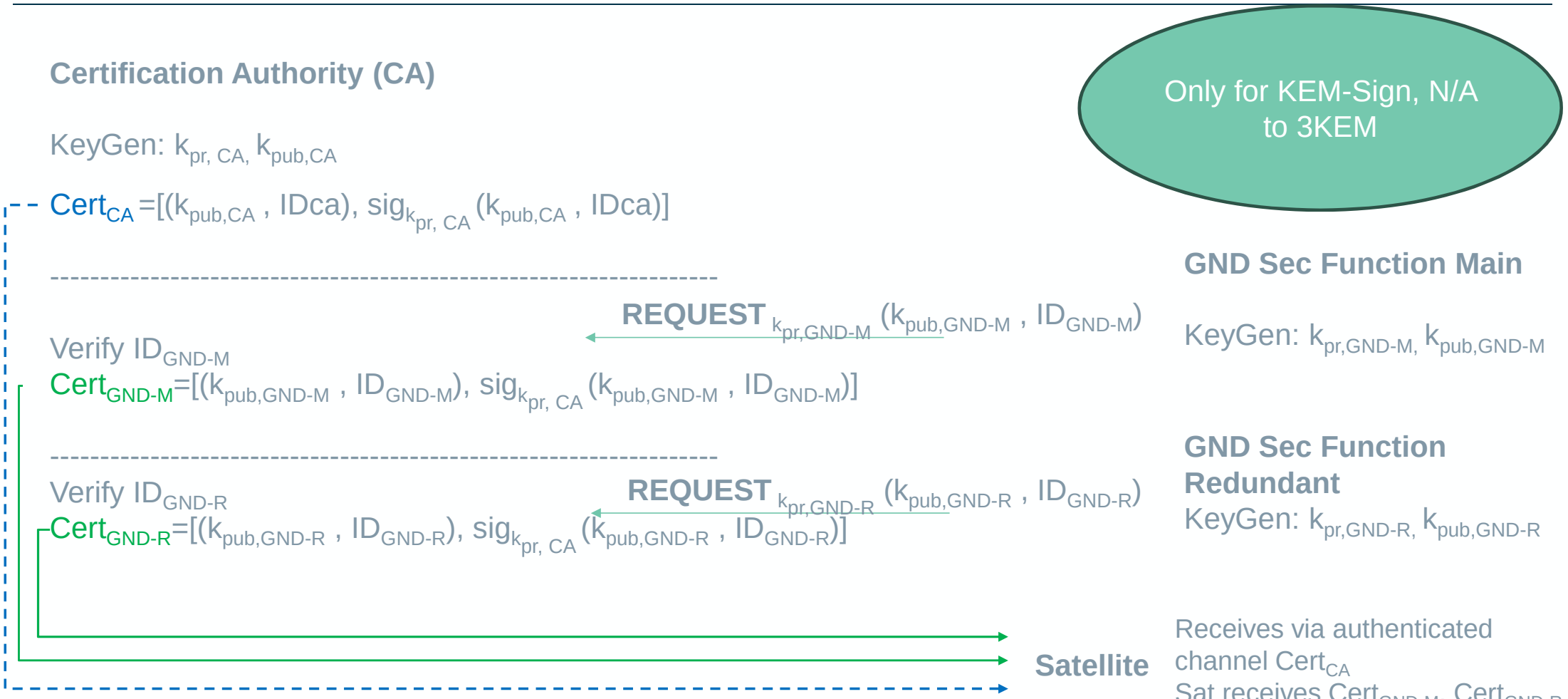
Verify IDsat

$Cert_{sat} = [(k_{pub,sat}\ ,\ IDsat),\ sig_{k_{pr,\ CA}}(k_{pub,sat}\ ,\ IDsat)]$

Ground Sec Function Main
Ground Sec Function Redundant

CertCA is distributed via authenticated channel

Certificates



Classic McEliece Security Strength vs Param Sets

	Parameter set	NIST Level	Public key size (bytes)	Secret key size (bytes)	Ciphertext size (bytes)	Shared secret size (bytes)
Level 1	Classic-McEliece-348864	1	261120	6492	96	32
Level 1	Classic-McEliece-348864f	1	261120	6492	96	32
Level 3	Classic-McEliece-460896	3	524160	13608	156	32
Level 3	Classic-McEliece-460896f	3	524160	13608	156	32
Level 5	Classic-McEliece-6688128	5	1044992	13932	208	32
Level 5	Classic-McEliece-6688128f	5	1044992	13932	208	32
Level 5	Classic-McEliece-6960119	5	1047319	13948	194	32
Level 5	Classic-McEliece-6960119f	5	1047319	13948	194	32
Level 5	Classic-McEliece-8192128	5	1357824	14120	208	32
Level 5	Classic-McEliece-8192128f	5	1357824	14120	208	32

LUTs – Classic McEliece vs Kyber vs Frodo

Table 2: Level I KEMs and PKEs on Artix-7 (default) and Zynq-7000 (indicated with the superscript ^{z})

Design	Algorithm	Type	Target	Max. Freq.	LUT	FF	Slice	DSP	BRAM	Key Generation		Encaps./Enc.^{cpa}		Decaps./(Dec.+Enc.)^{cpa}	
										cycles	μs	cycles	μs	cycles	μs
Security Level I															
[81]^{z}	NewHope-512^{cpa}	HW	HS	200	6,780	4,026	–	2	7.0	4,200	21.0	6,600	33.0	9,100	45.5
[75]	mceliece348864^{cpa}	HW	HS	106	81,339	132,190	–	0	236.0	202,787	1,920.3	2,720	25.8	12,743	120.7
[75]	mceliece348864^{cpa}	HW	LW	108	25,327	49,383	–	0	168.0	1,599,882	14,800.0	2,720	25.2	18,358	169.8
[26]^{z}	Kyber-512	SW/HW^{RV}	LW	–	23,925	10,844	–	21	32.0	150,106	–	193,076	–	204,843	–
[38]	FrodoKEM-640 16x	HW	HS	171	5,786	4,694	1,692	16	0						
				149	6,881	5,081	1,947	16	12.5	–	–	–	–	209,867	1,408.5
[13]	Kyber-512	SW/HW^{RV}	LW	25*	14,975	2,539	4,173	11	14.0	54,861	2,194.4	134,965	5,398.6	146,068	5,842.7
[26]^{z}	NewHope-512	SW/HW^{RV}	LW	–	23,925	10,844	–	21	32.0	123,860	–	207,299	–	226,742	–
[49]	SIKEp434	HW	HS	132	21,946	24,328	8,006	240	26.5	530,000	4,009.1	930,000	7,034.8	980,000	7,413.0
[49]	SIKEp503	HW	HS	130	24,610	27,759	9,186	264	33.5	640,000	4,926.9	1,140,000	8,776.0	1,200,000	9,237.9
[13]	NewHope-512	SW/HW^{RV}	LW	25*	14,975	2,539	4,173	11	14.0	97,969	3,918.8	236,812	9,472.5	258,872	10,354.9
[26]^{z}	LightSaber	SW/HW^{RV}	LW	–	23,925	10,844	–	21	32.0	366, 837	–	526, 496	–	657,583	–
[1]	Kyber-512	SW/HW^{RV}	LW	59	1,842	1,634	–	5	34.0	710,000	11,993.2	971,000	16,402.0	870,000	14,695.9
[1]	NewHope-512	SW/HW^{RV}	LW	59	1,842	1,634	–	5	34.0	904,000	15,270.3	1,424,000	24,054.1	1,302,000	21,993.2
[53]	SIKEp434	SW/HW^{c}	HS	162	22,595	11,558	7,491	162	37.0	1,474,200	9100	2,494,800	15,400.0	2,656,800	16,400.0
[53]	SIKEp503	SW/HW^{c}	HS	162	22,595	11,558	7,491	162	37.0	1,733,400	10,700.0	2,932,200	18,100.0	3,126,600	19,300.0
[38]	FrodoKEM-640 1x	HW	LW	191	971	433	290	1	0	3,237,288	16,949.2	–	–	–	–
				190	4,216	2,131	1,180	1	0	–	–	3,275,862	17,241.4	–	–
				162	4,446	2,152	1,254	1	12.5	–	–	–	–	3,306,122	20,408.2
[53]	SIKEp434	SW/HW^{c}	LW	143	10,976	7,115	3,512	57	21.0	2,187,902	15,300.0	3,718,004	26,000.0	3,946,804	27,600.0
[53]	SIKEp503	SW/HW^{c}	LW	143	10,976	7,115	3,512	57	21.0	2,602,603	18,200.0	4,390,104	30,700.0	4,676,105	32,700.0
[61]	BIKE Level 1	HW	LW	121	10,702	4,940	3,334	7	15.0	2,671,000	21,903.0	153,000	1,252.0	13,120,000	107,580.0
[61]	BIKE Level 1	HW	HS	96	29,448	5,498	8,419	7	28.0	259,000	2,691.0	12,000	127.0	13,120,000	136,443.0
[13]	FrodoKEM-640	SW/HW^{RV}	LW	25*	14,975	2,539	4,173	11	14.0	11,453,942	458,157.7	11,609,668	464,386.7	12,035,513	481,420.5
[40]	BIKE-1 Level 1es	HW	HS	165	1,907	1,049	608	0	7.0	95,500	578.0	–	–	–	–
[40]	BIKE-3 Level 1es	HW	HS	170	1,397	925	453	0	4.0	98,500	579.0	–	–	–	–
[40]	BIKE-2 Level 1es	HW	HS	160	3,874	2,141	1,312	0	10.0	2,150,000	13,437.0	–	–	–	–

^{z} Design implemented on Zynq-7000
^{cpa} Design of a PKE variant resistant against Chosen-Plaintext Attack (CPA)
^{es} Designs for the variants BIKE-1, BIKE-2, and BIKE-3 consolidated by submitters to BIKE on May 3, 2020
^{RV} co-design using RISC-V RV32IM
^{c} co-design using a custom processor
^{A9} co-design using ARM Cortex-A9
* Preliminary result

10281391 (nsf.gov)

LUTs – Classic McEliece vs Kyber vs Frodo

Table 5: Level 3 & 5 KEMs and PKEs on Virtex-7

Design	Algorithm	Type	Target	Max. Freq.	LUT	FF	Slice	DSP	BR AM	Key Generation		Encap./Enc.[cpa]		Decap./Dec.[cpa]	
										cycles	μs	cycles	μs	cycles	μs
Security Level III															
[75]	mceliece460896[cpa]	HW	HS	131	109,484	168,939		0	446.0	515,806	3,943.5	3,360	25.7	17,931	137.1
[19]	SIKEp610	HW	HS	239	16,226	26,757	7,461	270	38.5	–	–	2,280,000	9,550.0	2,280,000	9,550.0
[49]	SIKEp610	HW	HS	166	28,217	33,297	10,675	312	39.5	900,000	5,428.2	1,810,000	10,916.8	1,780,000	10,735.8
[53]	SIKEp610	SW/HW	HS	142	21,210	13,657	7,408	162	38.0	1,962,360	13,800.0	3,654,540	25,700.0	3,711,420	26,100.0
[53]	SIKEp610	SW/HW	LW	152	10,937	7,132	3,415	57	21.0	4,353,120	28,600.0	8,097,412	53,200.0	8,219,178	54,000.0
Security Level V															
[75]	mceliece6960119[cpa]	HW	HS	130	116,928	188,324		0	607.0	974,306	7,500.4	5,413	41.7	25,135	193.5
[75]	mceliece6688128[cpa]	HW	HS	137	122,624	186,194		0	589.0	1,046,139	7,658.4	5,024	36.8	29,754	217.8
[75]	mceliece8192128[cpa]	HW	HS	130	123,361	190,707		0	589.0	1,286,179	9,901.3	6,528	50.3	32,765	252.2
[75]	mceliece6960119[cpa]	HW	LW	141	44,154	88,963		0	563.0	11,179,636	79,570.4	5,413	38.5	46,141	328.4
[75]	mceliece6688128[cpa]	HW	LW	136	44,345	83,637		0	446.0	12,389,742	91,034.1	5,024	36.9	52,333	384.5
[75]	mceliece8192128[cpa]	HW	LW	134	45,150	88,154		0	525.0	15,185,314	113,154.4	6,528	48.6	55,330	412.3
[19]	SIKEp751	HW	HS	233	20,207	39,339	11,136	452	41.5	–	–	2,965,000	12,750.0	2,965,000	12,750.0
[49]	SIKEp751	HW	HS	163	39,953	50,079	15,834	512	43.5	1,250,000	7,664.0	2,210,000	13,550.0	2,340,000	14,347.0
[53]	SIKEp751	SW/HW	HS	142	21,210	13,657	7,408	162	38.0	2,516,940	17,700.0	4,166,460	29,300.0	4,479,300	31,500.0
[53]	SIKEp751	SW/HW	LW	152	10,937	7,132	3,415	57	21.0	7,960,426	52,300.0	13,150,685	86,400.0	14,185,693	93,200.0

[cpa] Design of a PKE variant resistant against Chosen-Plaintext Attack (CPA)

Table 23: Maximum frequency and resource utilization of hardware accelerators developed as a part of software/hardware co-designs targeting Zynq Ultrascale+

Algorithm	Security Category: Parameter Set	Max. Freq.	LUT	FF	Slice	DSP	BR AM
FrodoKEM	Frodo-640	402	7,213	6,647	1,186	32	13.5
FrodoKEM	Frodo-976	402	7,087	6,693	1,190	32	17.0
FrodoKEM	Frodo-1344	417	7,015	6,610	1,215	32	17.5
Kyber	KYBER512	410	12,034	10,532	2,327	8	14.0
Kyber	KYBER768	405	12,195	10,461	2,253	8	14.0
Kyber	KYBER1024	405	12,589	12,574	2,635	8	14.0

LUTs – Classic McEliece vs Kyber vs Frodo

Table 3: Level III & V KEMs and PKEs on Artix-7 (default) and Zynq-7000 (indicated with the superscript ^{z})

Design	Algorithm	Type	Target	Max. Freq.	LUT	FF	Slice	DSP	BRAM	Key Generation cycles	Key Generation µs	Encaps./Enc.^{cpa} cycles	Encaps./Enc.^{cpa} µs	Decaps./(Dec.+Enc.)^{cpa} cycles	Decaps./(Dec.+Enc.)^{cpa} µs
Security Level III															
[75]	mceliece460896^{cpa}	HW	LW	107	38,669	74,858		0	303.0	5,002,044	46,704.4	3,360	31.4	31,005	289.5
[38]	FrodoKEM-976 16x	HW	HS	169	2,869	3,000	908	16	0	476.05	2,816.9	–	–	–	–
				168	6,188	4,678	1782	16	0	–	–	479,993	2,857.1	–	–
				157	7,213	5,087	2042	16	19.0	–	–	–	–	483,073	3,076.9
[54]^{z}	Saber	SW/HW^{A9}	HS	125	7,400	7,331		28	2.0	–	3,273.0	–	4,147.0	–	3,844.0
[13]	Kyber-768	SW/HW^{RV}	LW	25*	14,975	2,539	4,173	11	14.0	84,110	3,364.4	184,080	7,363.2	198,011	7,920.4
[49]	SIKEp610	HW	HS	125	29,447	33,198	10,843	312	39.5	900,000	7,182.8	1,810,000	14,445.3	1,780,000	14,205.9
[53]	SIKEp610	SW/HW^{c}	HS	162	22,595	11,558	7,491	162	37.0	2,916,000	18,000.0	5,443,200	33,600.0	5,508,000	34,000.0
[38]	FrodoKEM-976 1x	HW	LW	189	1,343	441	363	1	0	7,560,000	40,000.0				
				187	4,650	2,118	1,272	1	0	–	–	7,480,000	40,000.0	–	–
				162	4,888	2,153	1,390	1	19.0	–	–	–	–	7,714,286	47,619.0
[53]	SIKEp610	SW/HW^{c}	LW	143	10,976	7,115	3,512	57	21.0	4,347,204	30,400.0	8,108,108	56,700.0	8,208,208	57,400.0
[61]	BIKE Level 3	HW	LW	122	9,808	5,075	2,996	7	23.0	11,600,207	95,122.6	601,099	4,929.1	37,596,111	308,291.2
[61]	BIKE Level 3	HW	HS	96	28,784	5,553	8,184	7	33.0	930,179	9,674.2	42,162	438.5	37,596,111	391,015.2
[13]	FrodoKEM-976	SW/HW^{RV}	LW	25*	14,975	2,539	4,173	11	14.0	26,005,326	1,040,213.0	29,749,417	1,189,976.7	30,421,175	1,216,847.0
Security Level V															
[81]^{z}	NewHope-1024^{cpa}	HW	HS	200	6,781	4,127	–	2	8.0	8,000	40.0	12,500	62.5	17,300	86.5
[41]^{z}	NewHope-1024^{cpa}	HW	HS	190	13,244	8,272	–	24	18.0	–	–	34,000	178.0	30,600	160.0
[13]	Kyber-1024	SW/HW^{RV}	LW	25*	14,975	2,539	4,173	11	14.0	116,841	4,673.6	236,886	9,475.4	256,828	10,273.1
[13]	NewHope-1024	SW/HW^{RV}	LW	25*	14,975	2,539	4,173	11	14.0	97,969	3,918.8	236,812	9,472.5	258,872	10,354.9
[26]	Kyber-1024	SW/HW	LW	–	23,925	10,844	–	21	32.0	349,673	–	405,477	–	424,682	–
[26]^{z}	NewHope-1024	SW/HW	LW	–	23,925	10,844	–	21	32.0	235,420	–	392,734	–	450,541	–
[49]	SIKEp751	HW	HS	127	40,792	49,982	15,794	512	43.5	1,250,000	9,842.5	2,210,000	17,401.6	2,340,000	18,425.2
[27]^{z}	NewHope-1024^{cpa}	SW/HW	HS	25	26,606	26,303	–	32	1.0	357,052	14,282.1	589,285	23,571.4	756,932	30,277.3
[26]^{z}	FireSaber	SW/HW	LW	–	23,925	10,844	–	21	32.0	1,300,272	–	1,622,818	–	1,898,051	–
[1]	Kyber-1024	SW/HW^{RV}	LW	59	1,842	1,634	–	5	34.0	2,203,000	37,212.8	2,619,000	44,239.9	2,429,000	41,030.4
[53]	SIKEp751	SW/HW^{c}	HS	162	22,595	11,558	7,491	162	37.0	3,742,200	23,100.0	6,188,400	38,200.0	6,658,200	41,100.0
[1]	NewHope-1024	SW/HW^{RV}	LW	59	1,842	1,634	–	5	34.0	1,776,000	30,000.0	2,742,000	46,317.6	2,528,000	42,702.7
[53]	SIKEp751	SW/HW^{c}	LW	143	10,976	7,115	3,512	57	21.0	7,965,108	55,700.0	13,156,013	92,000.0	14,185,614	99,200.0
[13]	FrodoKEM-1344	SW/HW^{RV}	LW	25*	14,975	2,539	4,173	11	14.0	67,994,170	2,719,766.8	71,501,358	2,860,054.3	72,526,695	2,901,067.8

^{z} Design implemented on Zynq-7000
^{cpa} Design of a PKE variant resistant against Chosen-Plaintext Attack (CPA)
^{RV} co-design using RISC-V RV32IM
^{c} co-design using a custom processor
^{A9} co-design using ARM Cortex-A9
* Preliminary result
^{KD} total execution time of Key Generation and Decryption
[41] only reports latency of Encapsulation and total latency of Key Generation and Decapsulation

HW/SW Codesign - Frodo

Table 8: All KEMs and PKEs on Zynq Ultrascale+

Design	Algorithm	Type	Target	Max. Freq.	LUT	FF	Slice	DSP	BRAM	Key Gen.		Encapsulation		Decapsulation	
										cycles	us	cycles	us	cycles	us
Security Level 1															
[18]	R5ND_1KEM_0d	SW/HW	HS	260	55,442	82,341	10,627	0	2	–	–	–	19.0	–	24.0
[18]	LightSaber	SW/HW	HS	322	12,343	11,288	1,989	256	3.5	–	–	–	53.0	–	56.0
[18]	FrodoKEM-640	SW/HW	HS	402	7,213	6,647	1,186	32	13.5	–	–	–	1,223.0	–	1,319.0
Security Level 3															
[63]	Saber	HW	HS	250	45,895	18,705	–	0	2	4,320	17.3	5,231	20.9	6,461	25.8
[63]	Saber	HW	HS	250	25,079	10,750	–	0	2	5,435	21.8	6,618	26.5	8,034	32.1
[18]	R5ND_3KEM_0d	SW/HW	HS	249	73,881	109,211	14,307	0	2	–	–	–	24.0	–	33.0
[18]	Saber	SW/HW	HS	322	12,566	11,619	1,993	256	3.5	–	–	–	60.0	–	65.0
[18]	FrodoKEM-976	SW/HW	HS	402	7087	6693	1190	32	17	–	–	–	1,642.0	–	1,866.0
Security Level 5															
[18]	R5ND_5KEM_0d	SW/HW	HS	212	91,166	151,019	18,733	0	2	–	–	–	32.0	–	42.0
[41]	NewHope-1024^{cpa}	HW	HS	406	13,961	8,149	–	25	18	–	–	34,000	83.0	30,600^{KD}	75.0^{KD}
[18]	FireSaber	SW/HW	HS	322	12,555	11,881	2,341	256	3.5	–	–	–	74.0	–	80.0
[18]	FrodoKEM-1344	SW/HW	HS	417	7,015	6,610	1,215	32	17.5	–	–	–	2,186.0	–	3,120.0

All SW/HW co-designs using ARM Cortex-A53

^{cpa} Design of a PKE variant resistant against Chosen-Plaintext Attack (CPA)

^{KD} total execution time of Key Generation and Decryption

ANSSI and NSA on FALCON

ANSSI

- **Falcon also called FN-DSA** [22]: this signature has been chosen by NIST as a future post-quantum standard. It is a compact and more efficient alternative to CRYSTALS-Dilithium. Since it is based on structured lattice problems, the same warning about the security applies. The design is here based on a more recent framework [8] with a hash-and-sign paradigm on lattices. It is more difficult to implement and needs intermediate variables to be defined as floats.

  For cryptographic products that may include this scheme, ANSSI makes the following recommendations:

  1. It is important to avoid modifying the parameters of the standardized instance. As implementing Falcon is not straightforward, we recommend to pay attention to stick to the design in order to avoid misuse attacks. We should also note that the Gaussian distributions in Falcon play an important role in the security and they should not be replaced.
  2. The parameters are defined for several minimum security levels. We recommend to use the highest level as possible, preferably level-5 (i.e. equivalent to AES-256).
  3. Please note that side-channel countermeasures are particularly difficult to apply and research has proved that side-channel attacks may defeat unprotected implementations of Falcon.

NSA

Q: Why did NSA choose ML-DSA over FN-DSA (aka Falcon)?

A: For NSS, NSA agrees with NIST: ML-DSA is preferred, as FN-DSA seems more susceptible to implementation errors that may affect security. As NIST has prioritized standardizing ML-DSA, it will likely be available sooner.

ANSSI on XMSS and LMS

- **XMSS** [12] / **LMS** [15]: these signature schemes were initially candidates in the NIST post-quantum standardization campaign but in 2018, they have been moved into a separate standardization process. The IETF version of their specification is cited above. These schemes are considered as conservative options because the underlying security hypothesis is very minimalist. Their security proofs are based on the security of hash functions. The particularity of these signatures is their statefulness and the potentially limited number of possible signatures per key pair.

  For contexts where the maximum number of signatures per key pair is restricted and where a state can be carefully stored, typically for software updates for example, ANSSI agrees that XMSS or LMS can be a relevant option, with the following recommendations:

  1. It is important to avoid modifying the parameters of the standardized instance including the underlying hash function.
  2. The parameters should provide the highest security level as possible.
  3. Hybridation (see Section 3 for more information) is optional for this signature.
  4. The state is a very critical data and should be protected in integrity. It should also be protected against replay attacks.

Additional statements from various NSAs

Netherlands: https://english.aivd.nl/publications/publications/2022/01/18/prepare-for-the-threat-of-quantumcomputers
https://english.aivd.nl/publications/publications/2023/04/04/the-pqc-migration-handbook

2022: "For PQC, we recommend the most secure algorithms, such as Frodo or McEliece" 2023 Kyber is recommended, Classic McEliece and FrodoKEM are acceptable

Germany: https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/Brochure/quantum-safe-cryptography.html
https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/TechGuidelines/TG02102/BSI-TR-02102-1.pdf

2023: FrodoKEM as well as Classic McEliece are assessed to be cryptographically suitable to protect confidential information on a long-term basis

France: https://www.ssi.gouv.fr/en/publication/anssi-views-on-the-post-quantum-cryptography-transition/
https://www.ssi.gouv.fr/uploads/2023/09/follow_up_position_paper_on_post_quantum_cryptography.pdf https://arxiv.org/pdf/2305.02855.pdf
2022: "FrodoKEM, Kyber, Dilithium or Falcon could be good options for first deployments“
2023: "Its competitive efficiency and simplicity are part of the reasons why Crystals-Kyber was selected as a first NIST post-quantum standard. Hence, Crystals-Kyber is expected to be the primary post-quantum KEM in security products and internet protocols." "However, ANSSI would encourage including FrodoKEM as a valid and conservative option in high security applications where the resulting performance penalty (in particular in terms of bandwidth) is not prohibitive."
2023: "On the other hand, NIST candidate Classic McEliece, has been the subject of such attacks [6] [19] [24]. The Goppa codes of a Niederreiter cryptosystem, which are derived from McEliece, have been the subject of a fault injection based attack explained in [10]. These elements make it opportune to follow the PQCRYPTO recommendations and focus on the original McEliece cryptosystem."

ANSSI on acceptable Elliptic Curves

TLS 1.3 a fait évoluer le protocole en donnant la possibilité au client de négocier les groupes DHE et les courbes elliptiques. Les groupes d'entiers et les courbes elliptiques utilisables ont également été inscrits dans les spécifications. Les groupes d'entiers retenus sont ceux définis dans [34] : `ffdhe2048`, `ffdhe3072`, `ffdhe4096`, `ffdhe6144` et `ffdhe8192`. Les courbes elliptiques retenues sont `secp256r1`, `secp384r1`, `secp521r1` (aussi appelées P-256, P-384 et P-521) ainsi que les courbes `x25519`, `x448`, `brainpoolP256r`, `brainpoolP384r` et `brainpoolP512r1`. La négociation de ces groupes est réalisée par l'extension `supported_groups`, détaillée dans la section 2.3.

Dans le cas d'ECDHE, pour une protection des données au-delà de 2020, le RGS préconise l'utilisation de groupes d'ordre multiple d'un nombre premier long d'au moins 256 bits [35]. L'ensemble des courbes retenues dans TLS 1.3 respectent le RGS.

R7 Échanger les clés avec l'algorithme ECDHE

Les échanges de clés ECDHE doivent être privilégiés, à l'aide des courbes `secp256r1`, `secp384r1`, `secp521r1`. Les courbes `x25519` et `x448` constituent des variantes acceptables. Les courbes `brainpoolP256r`, `brainpoolP384r` et `brainpoolP512r1` sont également acceptables.

Dans le cas de DHE, la sécurité de l'échange est liée à l'ordre du groupe multiplicatif en jeu. L'attaque Logjam [4] a illustré l'insuffisance des groupes de taille 512-bits, et pousse à déconseiller l'utilisation de groupes 1024-bits. Le RGS préconise l'utilisation de groupes 3072-bits ou plus, et tolère les groupes 2048-bits pour une protection des données jusqu'en 2030.

R7 – Échanger les clés avec l'algorithme DHE

Les échanges de clés DHE sont tolérés en utilisant les groupes 2048-bits ou plus (3072-bits ou plus si l'information doit être protégée au-delà de 2030) définis dans [34].

BSI on acceptable Elliptic Curves

3.4.2 Diffie-Hellman groups

In TLS 1.3, client and server can use the extension "supported_groups" to inform each other about the Diffie-Hellman groups they want to use for (EC)DHE.

The use of the following Diffie-Hellman groups is recommended:

Table 9: Recommended Diffie-Hellman groups for TLS 1.3

Diffie-Hellman group	*IANA no.*	*Specification*	*Use up to*
secp256r1	23	[RFC 8422]	2030+
secp384r1	24	[RFC 8422]	2030+
secp521r1	25	[RFC 8422]	2030+
brainpoolP256r1tls13	31	[RFC 8734]	2030+
brainpoolP384r1tls13	32	[RFC 8734]	2030+
brainpoolP512r1tls13	33	[RFC 8734]	2030+
ffdhe3072	257	[RFC 7919]	2030+
ffdhe4096	258	[RFC 7919]	2030+

Note: In general, the Brainpool curves are recommended.

Note: In [RFC 8446], the IANA numbers of some EC groups, that are either obsolete or have had little usage according to [RFC 8446], have been marked as "obsolete_RESERVED". Among those are the IANA numbers 26, 27, 28, which are allocated for the Brainpool curves for usage in TLS 1.2 and earlier TLS versions. For using the Brainpool curves in TLS 1.3, the IANA numbers 31, 32, 33 have been allocated (see [RFC 8734]).

[Technical Guideline TR-02102-2: Use of Transport Layer Security (TLS) (bund.de)]

3.6.2 Use of elliptic curves

When using elliptic curves, cryptographically strong curves over finite fields of the form F_p (p prime) are always recommended. In addition, it is recommended to only use *named curves* (see Section "Supported Groups Registry" in [IANA]) in order to avoid attacks via unverified weak domain parameters. The following named curves are recommended:

- brainpoolP256r1, brainpoolP384r1, brainpoolP512r1 (see [RFC 5639] and [RFC 7027])

If these curves are not available, the following curves can also be used:

- secp256r1, secp384r1, secp521r1

NSA on various algorithms

NSA | CNSA Suite 2.0 and Quantum Computing FAQ

Q: What is the Commercial National Security Algorithm Suite 2.0 (CNSA 2.0)?

A: CNSA 2.0 is the suite of QR algorithms approved for eventual NSS use. The following table lists the algorithms and their functions, specifications, and parameters.

Table: Commercial National Security Algorithm Suite 2.0

Algorithm	Function	Specification	Parameters
Advanced Encryption Standard (AES)	Symmetric block cipher for information protection	FIPS PUB 197	Use 256-bit keys for all classification levels.
ML-KEM (aka CRYSTALS-Kyber)	Asymmetric algorithm for key establishment	FIPS PUB 203	Use Category 5 parameter, ML-KEM-1024, for all classification levels.
ML-DSA (aka CRYSTALS-Dilithium)	Asymmetric algorithm for digital signatures in any use case, including signing firmware and software	FIPS PUB 204	Use Category 5 parameter, ML-DSA-87, for all classification levels.
Secure Hash Algorithm (SHA)	Algorithm for computing a condensed representation of information	FIPS PUB 180-4	Use SHA-384 or SHA-512 for all classification levels.
Leighton-Micali Signature (LMS)	Asymmetric algorithm for digitally signing firmware and software	NIST SP 800-208	All parameters approved for all classification levels. LMS SHA-256/192 is recommended.
Xtended Merkle Signature Scheme (XMSS)	Asymmetric algorithm for digitally signing firmware and software	NIST SP 800-208	All parameters approved for all classification levels.

CNSA 2.0

Q: Where should CNSA 2.0 algorithms be used?

A: CNSA 2.0 algorithms will be required for all products that employ public-standard algorithms in NSS, whether a future design or currently fielded. Any usage of Suite B or CNSA 1.0 algorithms will be required to transition to CNSA 2.0 usage. The Timeframe

NSA | CNSA Suite 2.0 and Quantum Computing FAQ

solutions can consist of many traditional or QR algorithms. "Component algorithms" are individual algorithms used in a hybrid solution.

Q: What is NSA's position on the use of hybrid solutions?

A: NSA has confidence in CNSA 2.0 algorithms and will not require NSS developers to use hybrid certified products for security purposes. Product availability and interoperability requirements may lead to adopting hybrid solutions.

NSA recognizes that some standards may require using hybrid-like constructions to accommodate the larger sizes of CRQC algorithms and will work with industry on the best options for implementation.

Q: What complications can using a hybrid solution introduce?

A: Hybrids add complexity to protocols, as designers need to incorporate additional negotiation and error handling and implementers need to modify API's and testing.

Rather than ease the transition to quantum resistance, hybrid deployments introduce additional interoperability concerns, now that all algorithms plus the method of hybridization must be features common to all parties to a communication. Similarly, hybrid deployments add a second transition later as users eventually move away from classical algorithms in the future.

At the same time, hybrid solutions make the implementations more complex, so one must balance the risk of flaws in an increasingly complex implementation with the risk of a cryptanalytic breakthrough. Because more security products fail due to implementation or configuration errors than failures in their underlying cryptographic algorithms, spending limited resources to add cryptographic complexity can at times weaken security rather than improve it.

Where NSA recognizes a need to support a hybrid solution, extensive work will be performed to ensure that it can be safely implemented, including engineering to a high degree of robustness, and facilitation to a straightforward transition to QR-only solutions.

CSI_CNSA_2.0_FAQ_.PDF (defense.gov)

NATIONAL SECURITY AGENCY
CENTRAL SECURITY SERVICE

Commercial National Security Algorithm (CNSA) Suite

Rapid and secure information sharing is important to protect our Nation, its citizens and its interests. Strong cryptographic algorithms are vital tools that contribute to our national security and help address the need for secure, interoperable communications. The National Security Agency (NSA) is responsible for approving solutions for protecting National Security Systems (NSS). Many systems in the NSS community are planned over decade timescales, have very long lifetimes after deployment, and are used to protect data that requires confidentiality for years beyond that.

Since 2005, a specific set of elliptical curve based algorithms, the CNSA cryptographic algorithms as specified by the National Institute of Standards and Technology (NIST), have been used by NSA in solutions approved for protecting classified and unclassified NSS. After observing the past decade of progress in quantum computing research, NSA endorses the increasing consensus that quantum computers will pose a threat in the future and that protocols using public key algorithms in the market place today will eventually need to be addressed. Given the longevity and unique nature of NSS and the costs of converting our existing public key based infrastructure to new algorithms, it is prudent to reconsider our strategic approach to the protection of data on NSS now.

To ensure the confidentiality of our customers' long life data, NSA is planning for an upcoming transition to quantum resistant algorithms and encouraging the design and analysis of quantum resistant public key algorithms. NSA plans to support NIST and other external standards bodies in developing standards for quantum resistant cryptography. In 2015, NSA announced a revised set of cryptographic algorithms that can be used to protect NSS while the algorithms that would be part of a quantum resistant suite are developed. For symmetric algorithms, options exist today that will be sufficient well into the future and beyond the development of a quantum computer. In the public key space, the intent is to give more flexibility to vendors and our customers in the present as we prepare for a quantum safe future.

Commercial cryptography approved to protect NSS systems up to the TOP SECRET level

Algorithm	Function	Specification	Parameters
Advanced Encryption Standard (AES)	Block cipher used for information protection	FIPS Pub 197	Use 256 bit keys
Elliptic Curve Diffie-Hellman (ECDH) Key Exchange	Asymmetric algorithm used for key establishment	NIST SP 800-56A	Use Curve P-384
Elliptical Curve Digital Signature Algorithm (ECDSA)	Asymmetric algorithm used for digital signatures	FIPS Pub 186-4	Use Curve P-384
Secure Hash Algorithm (SHA)	Used for computing a condensed representation of information	FIPS Pub 180-4	Use SHA-384
Diffie-Hellman (DH) Key Exchange	Algorithm used for key establishment	IETF RFC 3526	Minimum 3072-bit modulus
RSA	Algorithm used for key establishment	NIST SP 800-56B rev 1	Minimum 3072-bit modulus
RSA	Asymmetric algorithm used for digital signatures	FIPS PUB 186-4	Minimum 3072-bit modulus

NIST on acceptable security strength

5.6.3 Projected Security Strength Time Frames and Current Approval Status

Over time, cryptographic algorithms and their associated key lengths may become more vulnerable to successful attacks, requiring a transition to stronger algorithms or longer key lengths. Table 4 provides a projected time frame for applying cryptographic protection at a minimum security strength (e.g., at least 128 bits in 2031). In Table 4:

1. Column 1 is divided into two sub-columns. The first sub-column indicates the security strength to be provided; the second sub-column indicates whether cryptographic protection is being applied to data (e.g., encrypted) or whether cryptographically protected data is being processed (e.g., decrypted).
2. Columns 2 and 3 indicate the time frames during which the security strength is either acceptable, OK for legacy use, or disallowed.
   - "Acceptable" indicates that the algorithm or key length is currently considered to be secure.
   - "Legacy use" means that an algorithm or key length may be used because of its use in legacy applications (i.e., the algorithm or key length can be used to process cryptographically protected data).
   - "Disallowed" means that an algorithm or key length **shall not** be used for applying cryptographic protection (e.g., encrypting).

Table 4: Security strength time frames

Security Strength		Through 2030	2031 and Beyond
< 112	Applying protection	Disallowed	
	Processing	Legacy use	
112	Applying protection	Acceptable	Disallowed
	Processing		Legacy use
128	Applying protection and processing information that is already protected	Acceptable	Acceptable
192		Acceptable	Acceptable
256		Acceptable	Acceptable

NIST Curves

3. Recommended Curves for U.S. Federal Government Use

This section specifies the elliptic curves recommended for U.S. Federal Government use and contains choices for the private key length and underlying fields. This includes elliptic curves over prime fields (Section 3.2) and elliptic curves over binary fields (Section 3.3), where each curve takes one of the forms described in Section 3 (referred to as "*Type*" below).

Each recommended curve is uniquely defined by its domain parameters D, which indicate the field GF(q) over which the elliptic curve is defined, the parameters of its defining equation, and principal parameters, such as the cofactor h of the curve, the order n of its prime-order subgroup, and a designated point G on the curve of order n (i.e., the "base point"). In the case $q = 2^m$, the domain parameters also include a description of the representation chosen for GF(q).

Table 1. Approximate Security Strength of the Recommended Curves

Security Strength	Recommended Curves
112	P-224, K-233, B-233

Security Strength	Recommended Curves
128	P-256, W-25519, Curve25519, Edwards25519, K-283, B-283
192	P-384, K-409, B-409
224	W-448, Curve448, Edwards448, E448
256	P-521, K-571, B-571

Each elliptic curve specified in this recommendation is allowed for specific NIST approved cryptographic functions. The allowed usages for each curve are summarized in Table 2.

Table 2. Allowed Usage of the Specified Curves

Specified Curves	Allowed Usage
K-233, B-233 K-283, B-283 K-409, B-409 K-571, B-571	Deprecated
P-224 P-256 P-384 P-521	ECDSA, EC key establishment (see [SP_800-56A])
Edwards25519 Edwards448	EdDSA
Curve25519, W-25519 Curve448, E448, W-448	Alternative representations included for implementation flexibility. Not to be used for ECDSA or EdDSA directly.

Appendix H. Other Allowed Elliptic Curves

H.1. Brainpool Curves

This standard also allows the curves specified in *Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation* [RFC_5639] for ECDSA signatures as well as EC key establishment, which support a security strength of 112 bits or higher. In particular, this includes brainpoolP224r1, brainpoolP256r1, brainpoolP320r1, brainpoolP384r1, and brainpoolP512r1. These curves were pseudorandomly generated and are allowed to be used for interoperability reasons.

H.2. The Curve secp256k1

This standard also allows the curve secp256k1 specified in *SEC 2: Recommended Elliptic Curve Domain Parameters* [SEC_2], which supports a security strength of 128 bits. This curve is a Koblitz curve with coefficients selected for efficiency reasons. The curve secp256k1 is allowed to be used for blockchain-related applications.

Curve25519 was specified in IETF 7748 by the Crypto Forum Research Group (CFRG).

NIST Curves

Table 24: Appoved elliptic curves for ECC key-agreement.

Referenced in:	FIPS 186-4 SP 800-56A	TLS (RFC 4492) (SP 800-52)	IPsec w/ IKE v2 (RFC 5903)	Targeted Security Strengths that can be Supported
Specified in:	SP 800-186[29]	SEC 2	RFC 5903	
	P-224	secp224r1	-	$s = 112$
	P-256	secp256r1	secp256r1	$112 \le s \le 128$
	P-384	secp384r1	secp384r1	$112 \le s \le 192$
	P-521	secp521r1	secp521r1	$112 \le s \le 256$
	K-233	sect233k1	-	$112 \le s \le 128$
	K-283	sect283k1	-	$112 \le s \le 128$
	K-409	sect409k1	-	$112 \le s \le 192$
	K-571	sect571k1	-	$112 \le s \le 256$
	B-233	sect233r1	-	$112 \le s \le 128$
	B-283	sect283r1	-	$112 \le s \le 128$
	B-409	sect409r1	-	$112 \le s \le 192$
	B-571	sect571r1	-	$112 \le s \le 256$