Crypto Forum
Internet-Draft
Intended status: Informational
Expires: 8 January 2026

D. Connolly
SandboxAQ
R. Barnes
Cisco

7 July 2025

Concrete Hybrid PQ/T Key Encapsulation Mechanisms
draft-irtf-cfrg-concrete-hybrid-kems-00

Abstract

PQ/T Hybrid Key Encapsulation Mechanisms (KEMs) combine "post-quantum" cryptographic algorithms, which are safe from attack by a quantum computer, with "traditional" algorithms, which are not. CFRG has developed a general framework for creating hybrid KEMs. In this document, we define concrete instantiations of this framework to illustrate certain properties of the framework and simplify implementors' choices.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://cfrg.github.io/draft-irtf-cfrg-concrete-hybrid-kems/draft-irtf-cfrg-concrete-hybrid-kems.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-irtf-cfrg-concrete-hybrid-kems/.

Discussion of this document takes place on the Crypto Forum Research Group mailing list (mailto:cfrg@ietf.org), which is archived at https://mailarchive.ietf.org/arch/search/?email_list=cfrg. Subscribe at https://www.ietf.org/mailman/listinfo/cfrg/.

Source for this draft and an issue tracker can be found at https://github.com/cfrg/draft-irtf-cfrg-concrete-hybrid-kems.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

PQ/T Hybrid Key Encapsulation Mechanisms (KEMs) combine "post-quantum" cryptographic algorithms, which are safe from attack by a quantum computer, with "traditional" algorithms, which are not. Such KEMs are secure against a quantum attacker as long as the PQ algorithm is secure, and remain secure against traditional attackers even if the PQ algorithm is not secure.

[HYBRID-KEMS] defines a general framework for creating hybrid KEMs. It includes multiple specific mechanisms for combining a PQ algorithm with a traditional algorithm, with different performance properties and security requirements for the underlying algorithms.

In this document, we describe instances of these different specific combiners, with specific choices for the underlying algorithms. The choices described here illustrate the security analysis required to make choices that meet the requirements of the general framework, and can serve as a baseline for application designers. We also provide test vectors for these instances so that implementors can verify the correctness of their implementations.

2. Conventions and Definitions

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.

We make extensive use of the terminology in [HYBRID-KEMS].

3. Concrete Nominal Group and KEM Instances

This document introduces concrete hybrid KEM instances that in turn depend on concrete KEM and nominal group instances. This section introduces the nominal groups and KEM instances used for concrete hybrid KEM instances, specified in line with the abstraction from [HYBRID-KEMS]. Section 3.1 defines the concrete nominal groups, and Section 3.2 defines the nominal KEMs.

3.1. Nominal Groups

This section specifies concrete nominal groups that implement the abstraction in [HYBRID-KEMS]. It includes groups based on the NIST curves P-256 and P-384, as well as a group based on Curve25519.

3.1.1. P-256 and P-384 Nominal Groups

The NIST P-256 and P-384 elliptic curves are defined in [SP800-186]. They are widely used for key agreement and digital signature. In this section, we define how they meet the Nominal Group interface described in [HYBRID-KEMS].

Group elements are elliptic curve points, represented as byte strings in the compressed representation defined by the Elliptic-Curve-Point-to-Octet-String function in [SEC1].

The Nominal Group algorithms are the same for both groups:

* Exp(p, x) -> q: This function computes scalar multiplication between the input element (or point) p and the scalar x, according to the group law for the curve specified in [SP800-186].

* RandomScalar(seed) -> k: Implemented by converting seed to an integer using the Octet-String-to-Integer function in [SEC1], and then reducing the resulting integer modulo the group order.

* ElementToSharedSecret(p) -> ss: The shared secret is the X coordinate of the elliptic curve point p, encoded as an Nss-byte string using the Field-Element-to-Octet-String function in [SEC1].

The group constants for the P-256 group are as follows:

* Nseed: 48

* Nscalar: 32

* Nelem: 33

* Nss: 32

The group constants for the P-384 group are as follows:

* Nseed: 72

* Nscalar: 48

* Nelem: 49

* Nss: 48

3.1.2. Curve25519 Nominal Group

The following functions for the Curve25519 nominal group are defined:

* Exp(p, x) -> q: Implemented by X25519(x, p) from [RFC7748].

* RandomScalar(seed) -> k: Implemented by sampling and outputting 32 random bytes from a cryptographically secure pseudorandom number generator.

* ElementToSharedSecret(p) -> ss: Implemented by the identity function, i.e., by outputting P.

The following constants are also defined.

* Nseed: 32

* Nscalar: 32

* Nelem: 32

* Nss: 32

3.2. Concrete KEM Instances

This section specifies concrete KEM instances that implement the KEM abstraction from [HYBRID-KEMS].

3.2.1. ML-KEM-768 and ML-KEM-1024

The ML-KEM-768 and ML-KEM-1024 KEMs are defined in [FIPS203]. The algorithms defined in that specification map to the KEM abstraction in [HYBRID-KEMS] as follows:

* GenerateKeyPair() -> (ek, dk): Implemented as KeyGen in Section 7.1 of [FIPS203].

* DeriveKeyPair(seed) -> (ek, dk): Implemented as KeyGen_internal(seed[0:32], seed[32:64]), where KeyGen_internal is defined in Section 6 of [FIPS203].

* Encaps(ek) -> (ct, ss): Implemented as Encaps in Section 7.2 of [FIPS203].

* Decaps(dk, ct) -> ss: Implemented as Encaps in Section 7.3 of [FIPS203].

The KEM constants for ML-KEM-768 are as follows:

* Nseed: 64
* Nek: 1216
* Ndk: 32
* Nct: 1120
* Nss: 32

The KEM constants for ML-KEM-1024 are as follows:

* Nseed: 64
* Nek: 1629
* Ndk: 32
* Nct: 1629
* Nss: 32

3.3. Concrete PRG instances

This section specifies concrete PRG instances that implement the PRG abstraction from [HYBRID-KEMS] and meet the required security definitions.

3.3.1. SHAKE256

SHAKE256 is an extendable-output function (XOF) defined in the SHA-3 specification [FIPS202]. It can be used as a PRG for arbitrary values of Nout. When SHAKE256 is used as the PRG component in a hybrid KEM, it is implcit that Nout == KEM_T.Nseed + KEM_PQ.Nseed or Nout == Group_T.Nseed + KEM_PQ.Nseed as appropriate.

3.4. Concrete KDF instances

This section specifies concrete KDF instances that implement the KDF abstraction from [HYBRID-KEMS] and meet the required security definitions.

3.4.1. SHA-3

The SHA-3 hash function is defined in [FIPS202]. It produces a 32-byte output, so it is appropriate for use in hybrid KEMs with Nss = 32.

4. Concrete Hybrid KEM Instances

This section instantiates the following concrete KEMs:

QSF-MLKEM768-P256-SHA3256-SHAKE256: A hybrid KEM composing ML-KEM-768 and P-256 using the QSF scheme, with SHAKE256 as the PRG and SHA3-256 as the KDF.

QSF-MLKEM768-X25519-SHA3256-SHAKE256: A hybrid KEM composing ML-KEM-768 and Curve25519 using the QSF scheme, with SHAKE256 as the PRG and SHA3-256 as the KDF. This construction is identical to the X-Wing construction in [XWING-SPEC].

QSF-MLKEM1024-P384-SHA3256-SHAKE256: A hybrid KEM composing ML-KEM-1024 and P-384 using the QSF scheme, with SHAKE256 as the PRG and SHA3-256 as the KDF.

Each instance specifies the PQ and traditional KEMs being combined, the combiner construction from [HYBRID-KEMS], the label to use for domain separation in the combiner function, as well as the PRG and KDF functions to use throughout.

4.1. QSF-MLKEM768-P256-SHA3256-SHAKE256

This hybrid KEM is heavily based on [XWING], using the QSF combiner from [HYBRID-KEMS]. In particular, it has the same exact design but uses P-256 instead of X25519 as the the traditional component of the algorithm. It has the following parameters.

* Group_T: P-256 Section 3.1.1
* KEM_PQ: ML-KEM-768 Section 3.2.1
* PRG: SHAKE-256 [FIPS202]
* KDF: SHA3-256 [FIPS202]
* Label: QSF-P256-MLKEM768-SHAKE256-SHA3256

The KEM constants for the resulting hybrid KEM are as follows:

* Nseed: 32
* Nek: 1217
* Ndk: 32
* Nct: 1121

* Nss: 32

4.2. QSF-MLKEM768-X25519-SHA3256-SHAKE256

This hybrid KEM is identical to X-Wing [XWING-SPEC]. It has the following parameters.

* Group_T: Curve25519 Section 3.1.2

* KEM_PQ: ML-KEM-768 Section 3.2.1

* PRG: SHAKE-256 [FIPS202]

* KDF: SHA3-256 [FIPS202]

* Label: \.//^\

(This label does not follow the same pattern as the other KEMs here, but was chosen for compatibility with the X-Wing specification.)

The following constants for the hybrid KEM are also defined:

* Nseed: 32

* Nek: 1216

* Ndk: 32

* Nct: 1120

* Nss: 32

4.3. QSF-MLKEM1024-P384-SHA3256-SHAKE256

QSF-MLKEM1024-P384-SHA3256-SHAKE256 has the following parameters:

* Group_T: P-384 Section 3.1.1

* ‘KEM_PQ: ML-KEM-1024 Section 3.2.1

* PRG: SHAKE-256 [FIPS202]

* KDF: HKDF-SHA-256 [RFC5869]

* Label: QSF-P384-MLKEM1024-SHAKE256-SHA3256

The following constants for the hybrid KEM are also defined:

* Nseed: 32
* Nek: 1629
* Ndk: 32
* Nct: 1629
* Nss: 32

5. Security Considerations

[[TODO]]

6. IANA Considerations

This document has no IANA actions.

7. References

7.1. Normative References

[FIPS202] "SHA-3 standard :: permutation-based hash and extendable-output functions", National Institute of Standards and Technology (U.S.), DOI 10.6028/nist.fips.202, 2015, <https://doi.org/10.6028/nist.fips.202>.

[FIPS203] "Module-lattice-based key-encapsulation mechanism standard", National Institute of Standards and Technology (U.S.), DOI 10.6028/nist.fips.203, August 2024, <https://doi.org/10.6028/nist.fips.203>.

[HYBRID-KEMS] Connolly, D., "Hybrid PQ/T Key Encapsulation Mechanisms", Work in Progress, Internet-Draft, draft-irtf-cfrg-hybrid-kems-03, 25 February 2025, <https://datatracker.ietf.org/doc/html/draft-irtf-cfrg-hybrid-kems-03>.

[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/rfc/rfc2119>.

[RFC5869] Krawczyk, H. and P. Eronen, "HMAC-based Extract-and-Expand Key Derivation Function (HKDF)", RFC 5869, DOI 10.17487/RFC5869, May 2010, <https://www.rfc-editor.org/rfc/rfc5869>.

[RFC7748] Langley, A., Hamburg, M., and S. Turner, "Elliptic Curves for Security", RFC 7748, DOI 10.17487/RFC7748, January 2016, <https://www.rfc-editor.org/rfc/rfc7748>.

[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/rfc/rfc8174>.

[SP800-186] Chen, L., Moody, D., Regenscheid, A., Robinson, A., and K. Randall, "Recommendations for Discrete Logarithm-based Cryptography:: Elliptic Curve Domain Parameters", National Institute of Standards and Technology, DOI 10.6028/nist.sp.800-186, February 2023, <https://doi.org/10.6028/nist.sp.800-186>.

7.2. Informative References

[ANSIX9.62] ANS, "Public Key Cryptography for the Financial Services Industry: the Elliptic Curve Digital Signature Algorithm (ECDSA)", ANS X9.62-2005, November 2005.

[CDM23] Cremers, C., Dax, A., and N. Medinger, "Keeping Up with the KEMs: Stronger Security Notions for KEMs and automated analysis of KEM-based protocols", 2023, <https://eprint.iacr.org/2023/1933.pdf>.

[KSMW2024] Kraemer, J., Struck, P., and M. Weishaupl, "Binding Security of Implicitly-Rejecting KEMs and Application to BIKE and HQC", n.d., <https://eprint.iacr.org/2024/1233>.

[SCHMIEG2024] Schmieg, S., "Unbindable Kemmy Schmidt: ML-KEM is neither MAL-BIND-K-CT nor MAL-BIND-K-PK", 2024, <https://eprint.iacr.org/2024/523.pdf>.

[SEC1] "Elliptic Curve Cryptography, Standards for Efficient Cryptography Group, ver. 2", 2009, <https://secg.org/sec1-v2.pdf>.

[XWING] "X-Wing: The Hybrid KEM You’ve Been Looking For", 2024, <https://eprint.iacr.org/2024/039.pdf>.

[XWING-SPEC]
Connolly, D., Schwabe, P., and B. Westerbaan, "X-Wing: general-purpose hybrid post-quantum KEM", Work in Progress, Internet-Draft, draft-connolly-cfrg-xwing-kem-07, 3 May 2025, <https://datatracker.ietf.org/doc/html/draft-connolly-cfrg-xwing-kem-07>.

Appendix A. Test Vectors

[[TODO]]

Acknowledgments

[[TODO]]

Authors' Addresses

Deirdre Connolly
SandboxAQ
Email: durumcrustulum@gmail.com

Richard Barnes
Cisco
Email: rlb@ipv.sx