PQUIP
Internet-Draft
Intended status: Standards Track
Expires: 23 April 2026

L. Prabel
S. Sun
G. Zeng
G. Wang
Huawei
20 October 2025

Post-Quantum Algorithms guidance
draft-prabel-pquip-pqc-guidance-01

Abstract

This document provides general information (such as parameter sizes, security assumptions, and targeted security models) on a range of widely studied post-quantum cryptographic algorithms, including Key Encapsulation Mechanisms (KEMs) and digital signature schemes.

The cryptographic schemes described in this document are among those recommended by major security agencies and/or standardization bodies, and are believed to be secure against Cryptographically Relevant Quantum Computer (CRQC).

The goal of this document is to offer a high-level overview of these schemes and their distinguishing features, to help implementers, protocol designers, standards developers, and policymakers in understanding and selecting appropriate post-quantum cryptographic primitives for use in protocols and systems.

By aggregating and presenting this information in a unified format, this document aims to facilitate informed decision-making and interoperability during the migration to post-quantum cryptography, and to encourage consistent practices when evaluating and deploying Post-Quantum Cryptography (PQC) schemes in cryptographic protocols.

About This Document

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

Status information for this document may be found at https://datatracker.ietf.org/doc/draft-prabel-pquip-pqc-guidance/.

Discussion of this document takes place on the Post-Quantum Use In Protocols Working Group mailing list (mailto:pqc@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/pqc/. Subscribe at https://www.ietf.org/mailman/listinfo/pqc/.

Source for this draft and an issue tracker can be found at https://github.com/USER/REPO.

Status of This Memo

This Internet-Draft will expire on 23 April 2026.

Copyright Notice

Table of Contents

1. Introduction

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.

This document follows the terminology for post-quantum hybrid schemes defined in [I-D.draft-ietf-pquip-pqt-hybrid-terminology-04].

This section recalls some of this terminology, but also adds other definitions used throughout the whole document:

Traditional Asymmetric Cryptographic Algorithm: An asymmetric cryptographic algorithm based on integer factorisation, finite field discrete logarithms, elliptic curve discrete logarithms, or related mathematical problems. They can also be called classical or conventional algorithms.

Post-Quantum Asymmetric Cryptographic Algorithm: An asymmetric cryptographic algorithm that is intended to be secure against attacks using quantum computers as well as classical computers. They can also be called quantum-resistant or quantum-safe algorithms.

As with all cryptography, it always remains the case that attacks, either quantum or classical, may be found against post-quantum algorithms. Therefore it should not be assumed that just because an algorithm is designed to provide post-quantum security it will not be compromised. Should an attack be found against a post-quantum algorithm, it is commonly still referred to as a post-quantum algorithm as they were designed to protect against an adversary with access to a CRQC and the labels are referring to the designed or desired properties.

IND-CCA2: Indistinguishability under Adaptive Chosen-Ciphertext Attack. It is the standard security notion for KEM schemes.

EUF-CMA: Existential Unforgeability under Chosen-Message Attack. It is the standard security notion for digital signature schemes.

SUF-CMA: Strong Existential Unforgeability under Chosen-Message Attack. It is a stronger security notion than EUF-CMA.

3. Parameter Sizes

This section is divided into two different subsections, one focused on Key Encapsulation Mechanism, and the other on signature schemes.

The "claimed security level" in each table refers to the NIST Post-Quantum Cryptography Evaluation Criteria. We summarize this classification in Table 1 below. Additional details are available at [IR.8547].

Security Category	Attack Type	Example
1	Key search on a block cipher with a 128-bit key	AES-128
2	Collision search on a 256-bit hash function	SHA-256
3	Key search on a block cipher with a 192-bit key	AES-192
4	Collision search on a 384-bit hash function	SHA3-384
5	Key search on a block cipher with a 256-bit key	AES-256

Table 1: NIST Post-Quantum Cryptography Classification

3.1. Key Encapsulation Mechanism (KEM) Schemes

A Key Encapsulation Mechanism (KEM) is a cryptographic primitive that can be used as a building block within a broader key establishment protocol. Therefore, while KEMs are often employed to achieve the same end goal as a traditional key exchange, they do not, by themselves, define the interactive procedures, message flows, or authentication steps that a full key exchange protocol requires.

This distinction is particularly relevant for implementers and developers to avoid confusion:

* A KEM provides the mechanism for securely deriving and encapsulating a shared secret.

* A Key Exchange Protocol defines the interaction between parties that uses one or more KEMs (and possibly other primitives) to securely establish a secret key in context.

3.1.1. ML-KEM

ML-KEM, formerly known as CRYSTALS-Kyber, is a structured lattice-based KEM, the first PQC KEM standardized by NIST. The security of ML-KEM is based on the computational hardness of the Module Learning with Errors problem.

NIST recommends Security Level 3 by default, and European security agencies recommend a minimum of the same security level.

The NIST specification of ML-KEM is available at [MLKEM.SPEC].

Scheme	Public Key	Private Key	Ciphertext	Shared Secret	Claimed Security Level
ML-KEM-512	800	1632	768	32	1
ML-KEM-768	1184	2400	1088	32	3
ML-KEM-1024	1568	2168	1568	32	5

Table 2: ML-KEM Parameter Sizes (in bytes)

[MLKEM.SPEC] also allows to use a 64-bytes seed to represent the private key.

3.1.2. FrodoKEM

FrodoKEM is a lattice-based KEM whose security is based on the plain Learning with Errors (LWE) hardness assumption, unlike ML-KEM which is based on structured lattices. This makes FrodoKEM a more conservative KEM scheme.

It is considered for standardization by ISO, and it is recommended by European security agencies.

Each security level allows the choice between AES and SHAKE as the underlying symmetric primitive. The AES variant is beneficial on devices with AES hardware acceleration, while the SHAKE variant generally provides better performance if hardware acceleration is not available.

The FrodoKEM specification is available at [FRODOKEM.SPEC].

Scheme	Public Key	Private Key	Ciphertext	Shared Secret	Claimed Security Level
FrodoKEM-640-AES	9616	19888	9720	16	1
FrodoKEM-640-SHAKE	9616	19888	9720	16	1
FrodoKEM-976-AES	15632	31296	15744	24	3
FrodoKEM-976-SHAKE	15632	31296	15744	24	3
FrodoKEM-1344-AES	21520	43088	21632	32	5
FrodoKEM-1344-SHAKE	21520	43088	21632	32	5

Table 3: FrodoKEM Parameter Sizes (in bytes)

3.1.3. Classic McEliece

Classic McEliece is a conservative, code-based KEM, based on the original McEliece cryptosystem from 1978.

It requires larger key sizes compared to the other KEMs discussed here, but relatively small ciphertext sizes.

Each security level includes an ’f’ variant that is more complex internally than the ’non-f’ variant but enables faster key generation.

It has withstood extensive cryptanalysis over several decades, and several European security agencies have expressed confidence in its security. It is considered for standardization by ISO.

The Classic McEliece specification is available at [CLASSICMCELIECE.SPEC].

Scheme	Public Key	Private Key	Ciphertext	Shared Secret	Claimed Security Level
Classic-McEliece-348864	261120	6492	96	32	1
Classic-McEliece-348864f	261120	6492	96	32	1
Classic-McEliece-460896	524160	13608	156	32	3
Classic-McEliece-460896f	524160	13608	156	32	3
Classic-McEliece-6688128	1044992	13932	208	32	5
Classic-McEliece-6688128f	1044992	13932	208	32	5
Classic-McEliece-6960119	1047319	13948	194	32	5
Classic-McEliece-6960119f	1047319	13948	194	32	5
Classic-McEliece-8192128	1357824	14120	208	32	5
Classic-McEliece-8192128f	1357824	14120	208	32	5

Table 4: Classic-McEliece Parameter Sizes (in bytes)

3.1.4. HQC

HQC is a code-based KEM relying on the decisional Quasi-Cyclic Syndrome Decoding (QCSD) hardness assumption.

HQC offers small public key and ciphertext sizes, although these are larger than those of ML-KEM.

It will be standardized by NIST.

The HQC specification is available at [HQC.SPEC].

Scheme	Public Key	Private Key	Ciphertext	Shared Secret	Claimed Security Level
HQC-128	2249	2305	4433	64	1
HQC-192	4522	4586	8978	64	3
HQC-256	7245	7317	14421	64	5

Table 5: HQC Parameter Sizes (in bytes)

3.1.5. NTRU

NTRU is a structured lattice-based KEM. It has a long, well-established history and has been widely analyzed.

It is considered for standardization by ISO.

The NTRU specification is available at [NTRU.SPEC].

Scheme	Public Key	Private Key	Ciphertext	Shared Secret	Claimed Security Level
ntruhps2048509	699	935	699	32	1
ntruhps2048677	930	1235	930	32	3
ntruhps4096821	1230	1592	1230	32	5
ntruhrss701	1138	1452	1138	32	3
ntruhrss1373	2401	2983	2401	32	5

Table 6: NTRU Parameter Sizes (in bytes)

3.2. Signature Schemes

Digital signatures are cryptographic primitives used to provide authenticity, integrity, and non-repudiation of messages or data.

In the context of post-quantum cryptography, signature schemes are designed to remain secure against adversaries with quantum computing capabilities. They can be used in various scenarios, including authentication of protocol messages, code signing, and certificates, and are often combined with key establishment mechanisms in secure communication protocols.

3.2.1. ML-DSA

ML-DSA, formerly known as CRYSTALS-Dilithium, is a structured lattice-based signature scheme, now standardized by NIST. The security of ML-DSA is based on the computational hardness of the Module Learning with Errors problem as well as the SelfTargetMSIS problem, a variant of the Module Short Integer Solution problem.

European security agencies recommend at least Security Level 3.

The NIST specification of ML-DSA is available at [MLDSA.SPEC].

Scheme	Public Key	Private Key	Signature	Claimed Security Level
ML-DSA-44	1312	2560	2420	2
ML-DSA-65	1952	4032	3309	3
ML-DSA-87	2592	4896	4627	5

Table 7: ML-DSA Parameter Sizes (in bytes)

[MLDSA.SPEC] also allows to use a 32-bytes seed to represent the private key.

3.2.2. FN-DSA

FN-DSA, formerly known as Falcon, is a lattice-based signature scheme that was selected by NIST for standardization.

It relies on floating-point arithmetic, which is considered challenging to implement securely, especially with respect to side-channel attacks.

The Falcon specification is available at [FNDSA.SPEC].

Scheme	Public Key	Private Key	Signature	Claimed Security Level
Falcon-512	897	1281	752	1
Falcon-1024	1793	2305	1462	5
Falcon-padded-512	897	1281	666	1
Falcon-padded-1024	1793	2305	1280	5

Table 8: FN-DSA Parameter Sizes (in bytes)

3.2.3. SLH-DSA

SLH-DSA, formerly known as SPHINCS+, is a stateless hash-based signature scheme now standardized by NIST.

Each security level offers two possible hash function families (SHA-2 or SHAKE) and for both, two specific variants. The ’s’ variant has smaller signature sizes, while the ’f’ variant has faster signature generation.

The NIST specification of SLH-DSA is available at [SLHDSA.SPEC].

Scheme	Public Key	Private Key	Signature	Claimed Security Level
SLH-DSA-SHA2-128s	32	64	7856	1
SLH-DSA-SHAKE-128s	32	64	7856	1
SLH-DSA-SHA2-128f	32	64	17088	1
SLH-DSA-SHAKE-128f	32	64	17088	1
SLH-DSA-SHA2-192s	48	96	16224	3
SLH-DSA-SHAKE-192s	48	96	16224	3
SLH-DSA-SHA2-192f	48	96	35664	3
SLH-DSA-SHAKE-192f	48	96	35664	3
SLH-DSA-SHA2-256s	64	128	29762	5
SLH-DSA-SHAKE-256s	64	128	29762	5
SLH-DSA-SHA2-256f	64	128	49856	5
SLH-DSA-SHAKE-256f	64	128	49856	5

Table 9: SLH-DSA Parameter Sizes (in bytes)

3.2.4. LMS

Leighton-Micali Signatures (LMS) is a stateful hash-based signature scheme that uses LM-OTS for one-time signatures, and is based on Merkle hash trees.

It requires careful state management. [I-D.draft-ietf-pquip-hbs-state] provides guidance and security considerations on state management for stateful hash-based signature schemes.

The NIST specification of LMS is available at [LMS.SPEC].

3.2.4.1. LMS with SHA-256

The signatures' sizes for the LMS_SHA256_M32_H{5, 10, 15, 20, 25} signature scheme depend on the choice of the underlying LMOTS scheme and in particular on the value of the Winternitz parameter W. Therefore, the signatures' sizes of LMS_SHA256_M32_H{5, 10, 15, 20, 25} are given in a 4-element array where values correspond to the value of W = 8, 4, 2, 1 in that order.

Scheme	Public Key	Private Key	Signature	Claimed Security Level
LMOTS_SHA256_N32_W1	56	8504	8516	x
LMOTS_SHA256_N32_W2	56	4280	4292	x
LMOTS_SHA256_N32_W4	56	2168	2180	x
LMOTS_SHA256_N32_W8	56	1112	1124	x
LMS_SHA256_M32_H5	56	1796	[1296, 2352, 4464, 8688]	5
LMS_SHA256_M32_H10	56	57348	[1456, 2512, 4624, 8848]	5
LMS_SHA256_M32_H15	56	1835012	[1616, 2672, 4784, 9008]	5
LMS_SHA256_M32_H20	56	58720260	[1776, 2832, 4944, 9168]	5
LMS_SHA256_M32_H25	56	1879048196	[1936, 2992, 5104, 9328]	5

Table 10: LMS with SHA256 Parameter Sizes (in bytes)

3.2.4.2. LMS with SHA-256/192

The signatures’ sizes for the LMS_SHA256/192_M24_H{5, 10, 15, 20, 25} signature scheme depend on the choice of the underlying LMOTS scheme and in particular on the value of the Winternitz parameter W. Therefore, the signatures’ sizes of LMS_SHA256/192_M24_H{5, 10, 15, 20, 25} are given in a 4-element array where values correspond to the value of W = 8, 4, 2, 1 in that order.

Scheme	Public Key	Private Key	Signature	Claimed Security Level
LMOTS_SHA256_N24_W1	56	4824	4828	x
LMOTS_SHA256_N24_W2	56	2448	2452	x
LMOTS_SHA256_N24_W4	56	1248	1251	x
LMOTS_SHA256_N24_W8	56	648	652	x
LMS_SHA256_M24_H5	56	1796	[784, 1384, 2584, 4960]	5
LMS_SHA256_M24_H10	56	57348	[904, 1504, 2704, 5080]	5
LMS_SHA256_M24_H15	56	1835012	[1024, 1624, 2824, 5200]	5
LMS_SHA256_M24_H20	56	58720260	[1144, 1744, 2944, 5320]	5
LMS_SHA256_M24_H25	56	1879048196	[1264, 1864, 3064, 5440]	5

Table 11: LMS with SHA256/192 Parameter Sizes (in bytes)

3.2.4.3. LMS with SHAKE256/256

The signatures’ sizes for the LMS_SHA256_M32_H{5, 10, 15, 20, 25} signature scheme depend on the choice of the underlying LMOTS scheme and in particular on the value of the Winternitz parameter W. Therefore, the signatures’ sizes of LMS_SHA256_M32_H{5, 10, 15, 20, 25} are given in a 4-element array where values correspond to the value of W = 8, 4, 2, 1 in that order.

Scheme	Public Key	Private Key	Signature	Claimed Security Level
LMOTS_SHAKE_N32_W1	56	8504	8516	x
LMOTS_SHAKE_N32_W2	56	4280	4292	x
LMOTS_SHAKE_N32_W4	56	2168	2180	x
LMOTS_SHAKE_N32_W8	56	1112	1124	x
LMS_SHAKE_M32_H5	56	1796	[1296, 2352, 4464, 8688]	5
LMS_SHAKE_M32_H10	56	57348	[1456, 2512, 4624, 8848]	5
LMS_SHAKE_M32_H15	56	1835012	[1616, 2672, 4784, 9008]	5
LMS_SHAKE_M32_H20	56	58720260	[1776, 2832, 4944, 9168]	5
LMS_SHAKE_M32_H25	56	1879048196	[1936, 2992, 5104, 9328]	5

Table 12: LMS with SHAKE256/256 Parameter Sizes (in bytes)

3.2.4.4. LMS with SHAKE256/192

The signatures’ sizes for the LMS_SHA256/192_M24_H{5, 10, 15, 20, 25} signature scheme depend on the choice of the underlying LMOTS scheme and in particular on the value of the Winternitz parameter W. Therefore, the signatures’ sizes of LMS_SHA256/192_M24_H{5, 10, 15, 20, 25} are given in a 4-element array where values correspond to the value of W = 8, 4, 2, 1 in that order.

Scheme	Public Key	Private Key	Signature	Claimed Security Level
LMOTS_SHAKE_N24_W1	56	4824	4828	x
LMOTS_SHAKE_N24_W2	56	2448	2452	x
LMOTS_SHAKE_N24_W4	56	1248	1252	x
LMOTS_SHAKE_N24_W8	56	648	652	x
LMS_SHAKE_M24_H5	56	1796	[784, 1384, 2584, 4960]	5
LMS_SHAKE_M24_H10	56	57348	[904, 1504, 2704, 5080]	5
LMS_SHAKE_M24_H15	56	1835012	[1024, 1624, 2824, 5200]	5
LMS_SHAKE_M24_H20	56	58720260	[1144, 1744, 2944, 5320]	5
LMS_SHAKE_M24_H25	56	1879048196	[1264, 1864, 3064, 5440]	5

Table 13: LMS with SHAKE256/192 Parameter Sizes (in bytes)

3.2.5. XMSS / XMSS^MT

The eXtended Merkle Signature Scheme (XMSS) is a stateful hash-based signature scheme that uses WOTS+ for one-time signatures, and is based on Merkle hash trees. XMSS^MT is a variant that has multiple hash trees.

It requires careful state management. [I-D.draft-ietf-pquip-hbs-state] provides guidance and security considerations on state management for stateful hash-based signature schemes.

The NIST specification of XMSS is available at [XMSS.SPEC].

Scheme	Public Key	Private Key	Signature	Claimed Security Level
XMSS-SHA2_10_256	64	1793	2500	5
XMSS-SHA2_16_256	64	2093	2692	5
XMSS-SHA2_20_256	64	2573	2820	5
XMSSMT-SHA2_20/2_256	64	5998	4963	5
XMSSMT-SHA2_20/4_256	64	10938	9251	5
XMSSMT-SHA2_40/2_256	64	9600	5605	5
XMSSMT-SHA2_40/4_256	64	15252	9893	5
XMSSMT-SHA2_40/8_256	64	24516	18469	5
XMSSMT-SHA2_60/3_256	64	16629	8392	5
XMSSMT-SHA2_60/6_256	64	24507	14824	5
XMSSMT-SHA2_60/12_256	64	38095	27688	5
XMSS-SHA2_10_192	48	1053	1492	5
XMSS-SHA2_16_192	48	1605	1636	5
XMSS-SHA2_20_192	48	1973	1732	5
XMSS-SHAKE256_10_256	64	1373	2500	5
XMSS-SHAKE256_16_256	64	2093	2692	5
XMSS-SHAKE256_20_256	64	2573	2820	5
XMSSMT-SHAKE256_20/2_256	64	5998	4963	5

XMSSMT-SHAKE256_20/4_256	64	10938	9251	5
XMSSMT-SHAKE256_40/2_256	64	9600	5605	5
XMSSMT-SHAKE256_40/4_256	64	15252	9893	5
XMSSMT-SHAKE256_40/8_256	64	24516	18469	5
XMSSMT-SHAKE256_60/3_256	64	24516	8392	5
XMSSMT-SHAKE256_60/6_256	64	24507	14824	5
XMSSMT-SHAKE256_60/12_256	64	38095	27688	5
XMSS-SHAKE256_10_192	48	1053	1492	5
XMSS-SHAKE256_16_192	48	1605	1636	5
XMSS-SHAKE256_20_192	48	1973	1732	5

Table 14: XMSS Parameter Sizes (in bytes)

4. Security Properties

4.1. Quantum-Vulnerable Asymmetric Cryptography

Table 15 gives a list of asymmetric cryptographic schemes that are vulnerable to quantum computers and are planned to be deprecated and/or disallowed in the future by various organizations or security agencies. In particular, NIST provides deprecation and disallowance timelines in [IR.8547].

The EU PQC Workstream also published its roadmap for the transition to post-quantum cryptography in [EU.Roadmap]. It distinguishes between low, medium and high quantum risk levels, and recommends completing the PQC transition for high-risk use cases before 2031, for medium-risk use cases before 2036, and for low-risk use cases before 2036, as much as feasible.

Scheme	Hardness assumption	Disallowed (NIST)
ECDSA	Discrete Logarithm	after 2035
EdDSA	Discrete Logarithm	after 2035
RSA	Factorisation	after 2035
(EC)DH	Decisional Diffie Hellman	after 2035

Table 15: Quantum-Vulnerable Asymmetric Cryptographic Schemes

4.2. Quantum-Safe Asymmetric Cryptography

Table 16 gives a brief summary of the security properties of various KEM algorithms.

Scheme	SDO	Hardness assumption	Security Model	Comments
ML-KEM	NIST	Module LWE	IND-CCA-2	xxx
FrodoKEM	ISO	Unstructured LWE	IND-CCA2	xxx
HQC	NIST	Decisional Quasi-Cyclic Syndrome Decoding Problem	IND-CCA2	xxx
Classic McEliece	ISO	Syndrome Decoding Problem, Goppa code recovery	IND-CCA2	xxx
NTRU	ISO	NTRU	IND-CCA2	xxx

Table 16: Properties of KEM schemes

FrodoKEM is believed to have conservative security compared to schemes based on structured lattices like ML-KEM or NTRU.

Table 17 gives a summary of the security properties of different signature algorithms.

Scheme	SDO	Hardness assumption	Security Model	Comments
ML-DSA	NIST	Module LWE, SelfTargetMSIS	SUF-CMA	xxx
FN-DSA	NIST	SIS over NTRU lattices	EUF-CMA	Uses floating point arithmetic
SLH-DSA	NIST	Second-preimage resistance	SUF-CMA (*)	xxx
LMS	NIST	Collision resistance	SUF-CMA (*)	Need state management
XMSS	NIST	Collision resistance	SUF-CMA (*)	Need state management

Table 17: Properties of signatures schemes

(*) There is no known attack on the SUF-CMA security of those schemes, which are widely believed to be SUF-CMA secure. However, no formal proof exists yet.

Hash-based signature schemes such as SLH-DSA, LMS, and XMSS are believed to offer more conservative security compared to lattice-based schemes like ML-DSA or FN-DSA.

5. IANA Considerations

This document has no IANA action.

6. References

6.1. Normative References

[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>.

[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>.

6.2. Informative References

[CLASSICMCELIECE.SPEC]
"Classic McEliece: conservative code-based cryptography", October 2022, <https://classic.mceliece.org/mceliece-spec-20221023.pdf>.

[EU.Roadmap]
EU PQC Workstream, "A Coordinated Implementation Roadmap for the Transition to Post-Quantum Cryptography", June 2025, <https://digital-strategy.ec.europa.eu/en/library/coordinated-implementation-roadmap-transition-post-quantum-cryptography>.

[FNDSA.SPEC]
"Falcon: Fast-Fourier Lattice-based Compact Signatures over NTRU", October 2020, <https://falcon-sign.info/falcon.pdf>.

[FRODOKEM.SPEC]
"FrodoKEM: Learning With Errors Key Encapsulation", December 2024, <https://frodokem.org/files/FrodoKEM_standard_proposal_20241205.pdf>.

[HQC.SPEC] "Hamming Quasi-Cyclic (HQC)", February 2025, <https://pqc-hqc.org/doc/hqc-specification_2025-02-19.pdf>.

[I-D.draft-ietf-pquip-hbs-state]
Wiggers, T., Bashiri, K., K旦lbl, S., Goodman, J., and S. Kousidis, "Hash-based Signatures: State and Backup Management", Work in Progress, Internet-Draft, draft-ietf-pquip-hbs-state-00, 17 June 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-pquip-hbs-state-00>.

[I-D.draft-ietf-pquip-pqt-hybrid-terminology-04]
D, F., P, M., and B. Hale, "Terminology for Post-Quantum Traditional Hybrid Schemes", Work in Progress, Internet-Draft, draft-ietf-pquip-pqt-hybrid-terminology-04, 10 September 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-pquip-pqt-hybrid-terminology-04>.

[IR.8547] National Institute of Standards and Technology (NIST), "Transition to Post-Quantum Cryptography Standards", November 2024, <https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf>.

[LMS.SPEC] "Stateless Hash-Based Digital Signature Standard", August 2024, <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-208.pdf>.

[MLDSA.SPEC] "Module-Lattice-Based Digital Signature Standard", August 2024, <https://nvlpubs.nist.gov/nistpubs/fips/nist.fips.204.pdf>.

[MLKEM.SPEC] National Institute of Standards and Technology (NIST), "Module-Lattice-Based Digital Signature Standard", August 2024, <https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.204.pdf>.

[NTRU.SPEC] "NTRU Key Encapsulation", March 2025, <https://www.ietf.org/id/draft-fluhrer-cfrg-ntru-02.html>.

[SLHDSA.SPEC] "Stateless Hash-Based Digital Signature Standard", August 2024, <https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.205.pdf>.

[XMSS.SPEC] "Recommendation for Stateful Hash-Based Signature Schemes", October 2020, <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-208.pdf>.

Authors’ Addresses

Lucas Prabel
Huawei
Email: lucas.prabel@huawei.com

Shuzhou Sun
Huawei
Email: sunshuzhou@huawei.com

Guang Zeng
Huawei
Email: zengguang13@huawei.com

Guilin Wang
Huawei
Email: wang.guilin@huawei.com