Keyfactor
Tech Days
2023

Planning for Post-Quantum Cryptography: Evolution of Internet Standards

Russ Housley

Founder

Vigil Security, LLC

Motivation

- If large-scale quantum computers are ever built, these computers will be able to break the public key cryptosystems currently in use.
- A post-quantum cryptosystem (PQC) is secure against large-scale quantum computers.
- It is open to conjecture when it will be feasible to build such computers; however, RSA, DSA, DH, ECDH, ECDSA, and EdDSA are all vulnerable if a large-scale quantum computer is developed.

NIST Hash-based Signature Algorithms

The U.S. National Institute of Standards and Technology (NIST) has already approved two PQC hash-based signature algorithms and published their specifications:

https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-208.pdf

- Digital Signatures:

  HSS/LMS (RFC 8554) and XMSS (RFC 8391)

Note: NIST chose to adopt these two algorithms that the IETF had already specified

NIST Competition – Four Winners (so far)

The U.S. National Institute of Standards and Technology (NIST) is conducting a multi-round competition for PQC public key algorithms:

https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/

Winners at the end of the third round:

- Key Encapsulation Mechanism (KEM) for key establishment:

  CRYSTALS-KYBER

- Digital signatures:

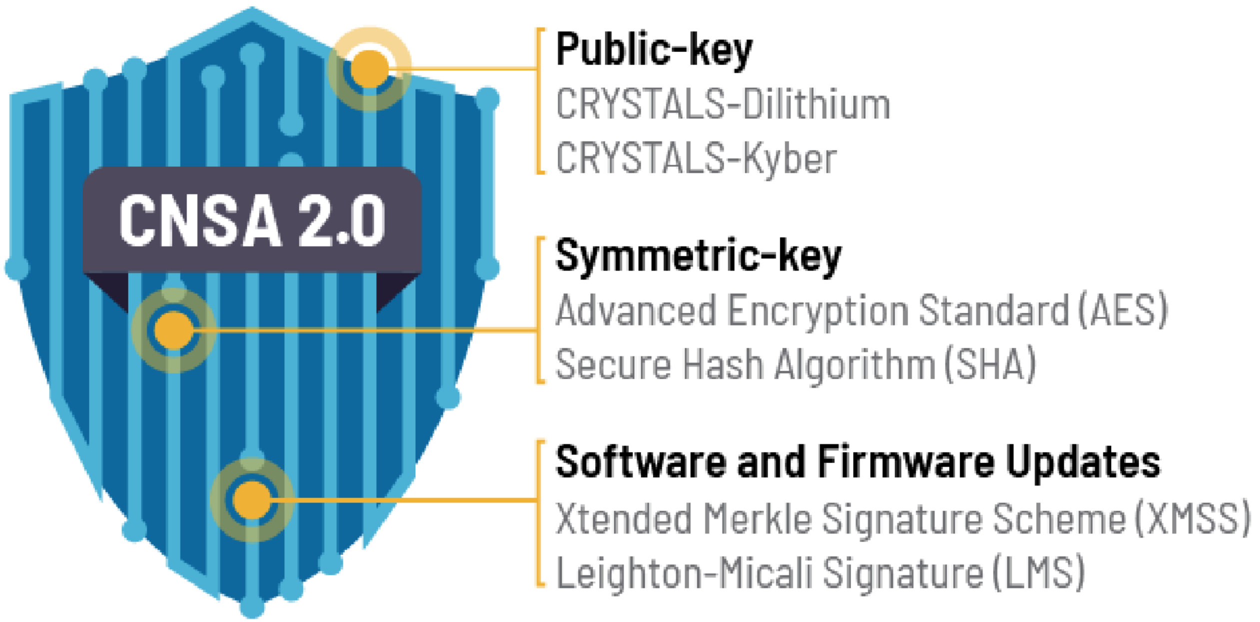
  CRYSTALS-DILITHIUM, FALCON, and SPHINCS+

NIST is considering additional rounds to select additional algorithms.

NSA Announced Direction

About a month after NIST announced the winning algorithms, NSA announced that National Security Systems should begin planning to implement:

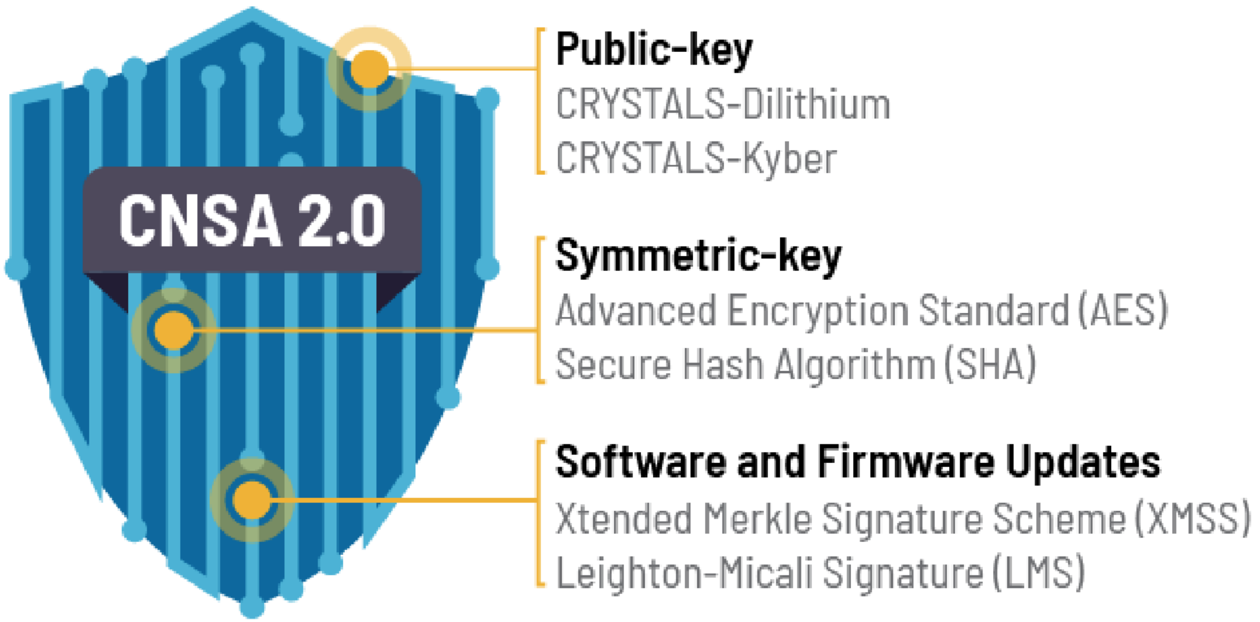
- Prefer HSS/LMS for software signing
- Prefer CRYSTALS-Dilithium for other signing
- Prefer CRYSTALS-Kyber for key management



NSA Announced Direction

About a month after NIST announced the winning algorithms, NSA announced that National Security Systems should begin planning to implement:

- Prefer HSS/LMS for software signing
- Prefer CRYSTALS-Dilithium for other signing
- Prefer CRYSTALS-Kyber for key management



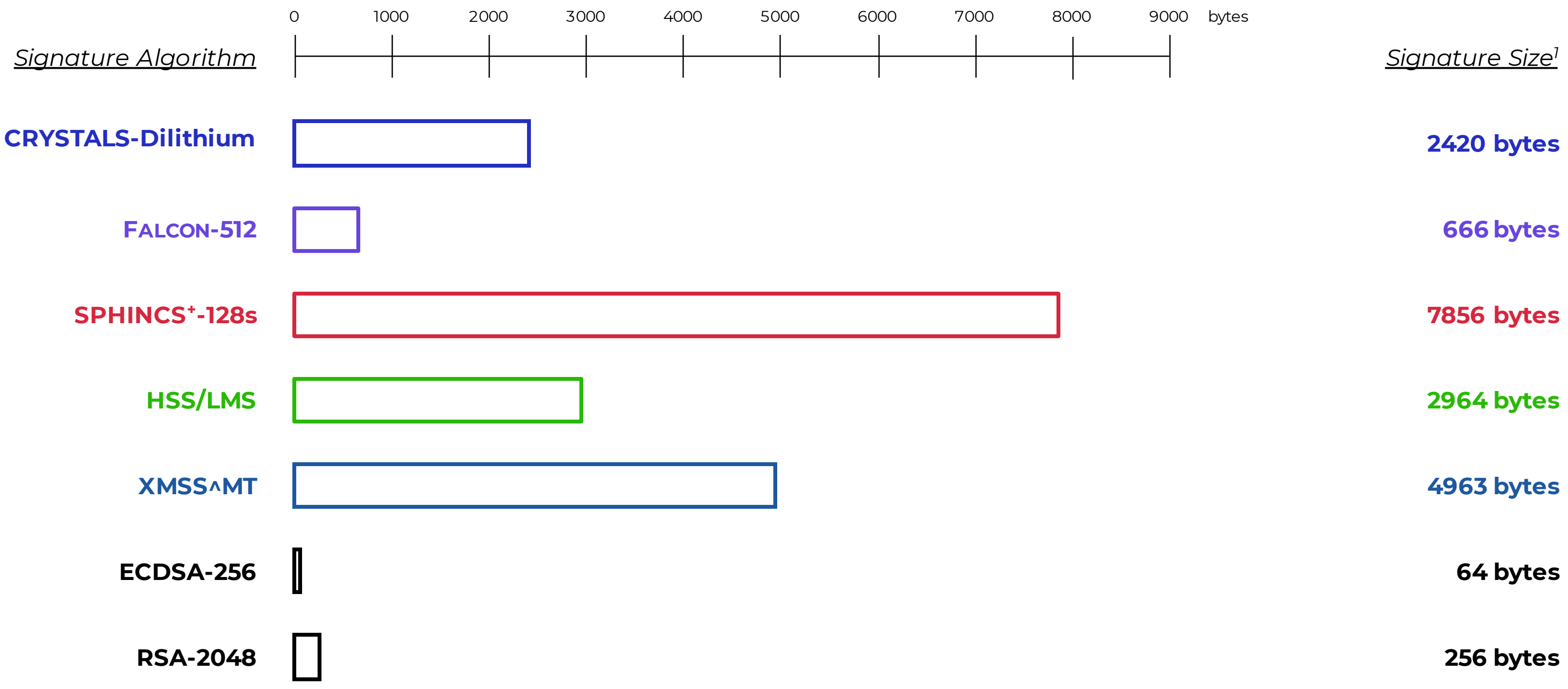
Transition is going to take a very long time. Let's get started!

IETF Security Protocols

Many security protocols are used in the Internet; all need to support PQC:

- IPsec
- TLS
- SSH
- S/MIME
- OpenPGP
- …
- Internet profile for X.509 certificates

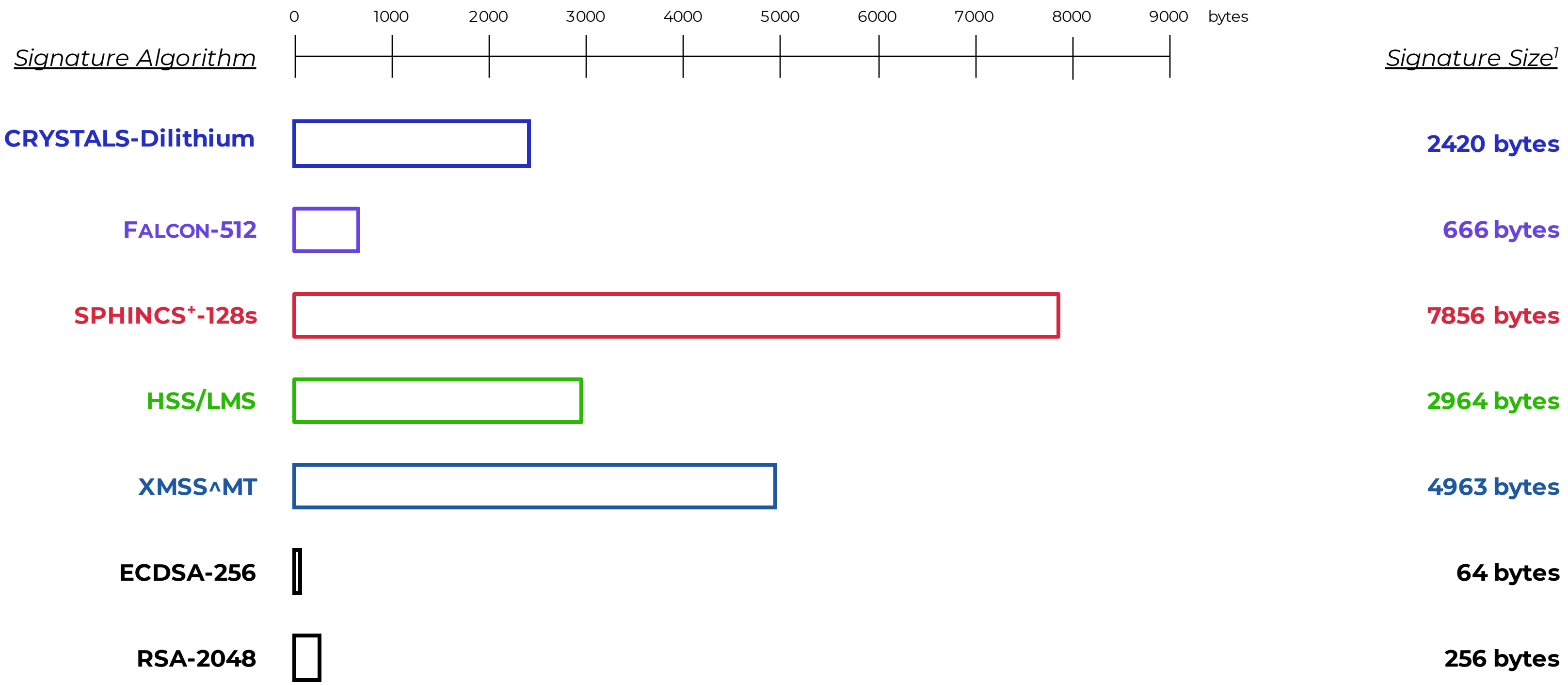
Large Public Key and Signature Size



[1]*with example parameters*

Thanks to Verisign for the graph

Large Public Key and Signature Size



[1] *with example parameters*

Plan for an increase of 10X in protocols …

Thanks to Verisign for the graph

Priorities

Confidentiality – The attacker can record today's traffic, and then break it when a large-scale quantum computer is eventually developed.

Authentication – Tends to be real-time interaction, so not a concern until a large-scale quantum computer is imminent.

Signature – Tends to be archival, so a notary or archivist can resign with a PQC signature at some point before a large-scale quantum computer is available. (See RFC 4998: Evidence Record Syntax.)

PQC Algorithms and Certificates

Goal – Deploy PQC algorithms before there is a large-scale quantum computer that is able to break the public key algorithms in widespread use today.

Assumption – While people gain confidence in the PQC algorithms and their implementations, security protocols will mix traditional algorithms and PQC algorithms.

Recognize – Such transitions take a long time—at least a decade.

Two Possible Certificate Approaches

Two certificates, each with one public key and one signature:

- one certificate traditional algorithm, signed with traditional algorithm
- one certificate PQC algorithm, signed with PQC algorithm

One certificate, containing multiple public keys and multiple signatures:

Public Key

SEQUENCE OF	Traditional public key
	PQC public key

Signature

SEQUENCE OF	Traditional signature
	PQC signature

Gaining Confidence (session-oriented)

While people gain confidence in the PQC algorithms and their implementations, security protocols are expected to mix traditional and PQC algorithms

IPsec and TLS, use a KDF to compute shared secret from two inputs:

$$\mathbf{SS = KDF(\ SS_T, SS_{PQC}\ )}$$

Gaining Confidence (session-oriented)

While people gain confidence in the PQC algorithms and their implementations, security protocols are expected to mix traditional and PQC algorithms

IPsec and TLS, use a KDF to compute shared secret from two inputs:

$$\mathbf{SS = KDF(\ SS_T,\ SS_{PQC}\ )}$$

For example:

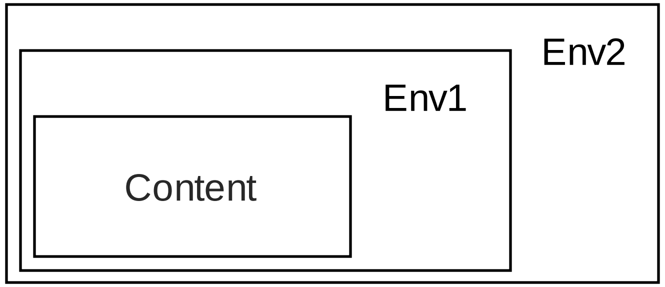
CRYSTALS-Kyber

Diffie-Hellman

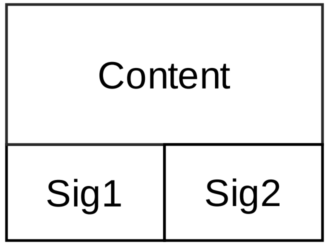
Gaining Confidence (store and forward)

S/MIME could so the same as IPsec and TLS, *or* more likely, S/MIME use double encapsulation:

Encryption



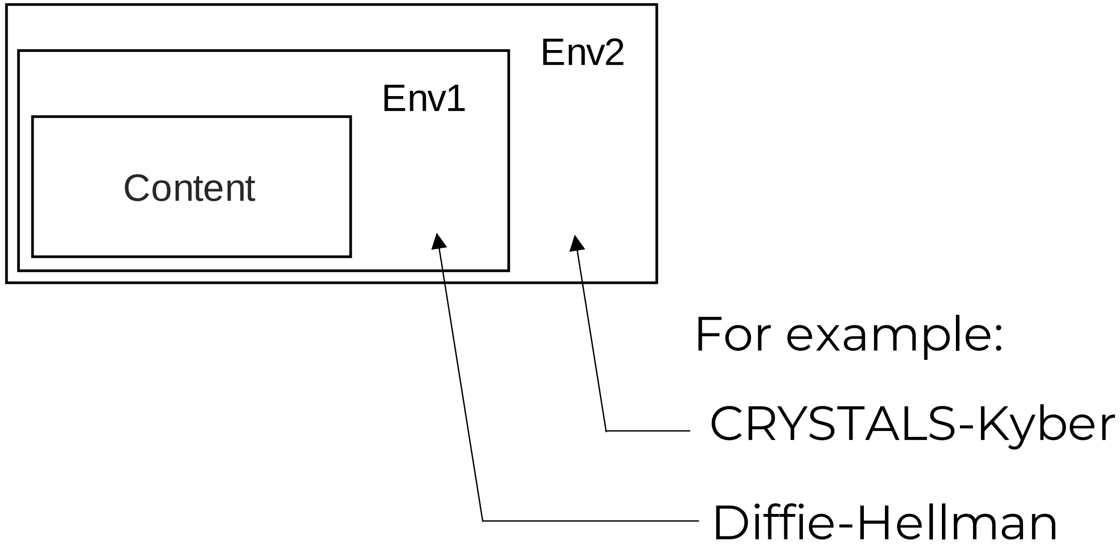
Signature



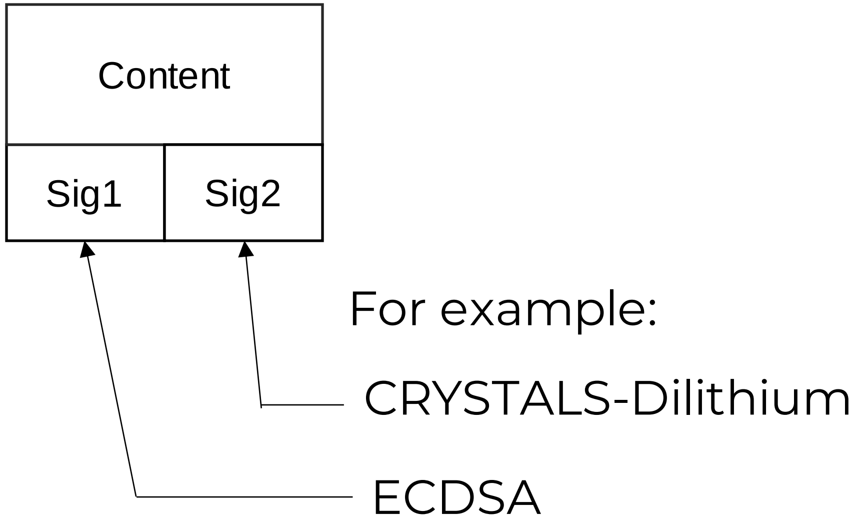
Gaining Confidence (store and forward)

S/MIME could so the same as IPsec and TLS, _or_ more likely,
S/MIME use double encapsulation:

Encryption



Signature



One Certificate, but Two Flavors

COMPOSITE

Composite encryption uses at least one of the public keys in the certificate

Composite decryption uses *at least one* of the private keys associated with the certified public keys (OR)

COMBINED

Combined encryption uses all of the public keys in the certificate

Combined decryption uses all of the private keys associated with all of the certified public keys (AND)

IETF SUIT Working Group

The IETF SUIT WG has specified a signed manifest for software updates. A PQC signature will be one of the mandatory to implement algorithms:

- Signing the software with a PQC algorithm offers a way to deploy other PQC algorithms, even if a large-scale quantum computer is invented soon
- Current draft specification requires implementation of HSS/LMS

IETF IPsecME Working Group

The IETF IPsecME WG has already specified a way for IKEv2 peers perform multiple successive key exchanges:

- **IKE_SA_INIT**: Always a traditional algorithm
- **IKE_INTERMEDIATE**: Allows PQC algorithms, and supports message fragmentation to handle the large public key sizes
- If any of the key exchange methods is a PQC algorithm, then the final keying material is post-quantum secure
- After NIST publishes their standards, IPsecME WG will specify their use with IKEv2

IETF TLS Working Group

The IETF TLS WG is defining the *hybrid* key exchange, which uses two or more algorithms to produce a final session key that is secure as long as at least one of the component key exchange algorithms remains unbroken.

- Client and server send the key shares, then the construct the **concatenated_shared_secret** by:

  shared_secret_1 || shared_secret_2 || … || shared_secret_n

- Compute the Handshake Secret in the TL 1.3 key schedule:

  concatenated_shared_secret -> HKDF-Extract = Handshake Secret

- After NIST publishes their standards, TLS WG will specify their use with this hybrid key exchange

IETF LAMPS Working Group

The IETF LAMPS WG will explore the transition to PQC for both certificates and S/MIME:

- specify the use of the NIST PQC public key algorithms using the object identifiers that are assigned by NIST
- specify formats, identifiers, enrollment, and operational practices for “hybrid key establishment”
- specify formats, identifiers, enrollment, and operational practices for “dual signature”

Russ Housley

housley@vigilsec.com

+1 703 435 1775