PQ KEMs for COSE AND JOSE

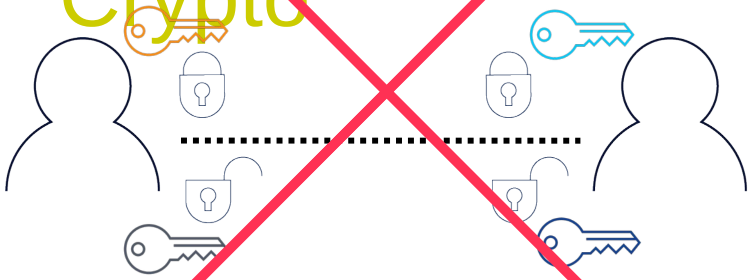
draft-reddy-cose-jose-pqc-kem

Tirumaleswar Reddy, Hannes Tschofenig, Aritra Banerjee

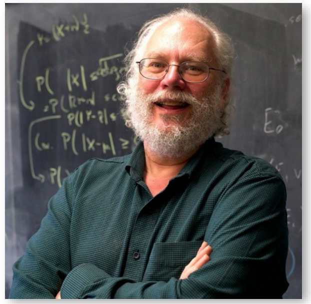
IETF 119, Brisbane

Impact of Quantum Computers in Cryptography

Asymmetric Crypto



Broken



Peter Shor

Algorithm for prime factorization of large integers

NIST Candidates Selected for Standardization

	Finalists	**4th round**
KEM/Encryption	ML-KEM (CRYSTALS-KYBER)	BIKE Classic McEliece HQC ~~SIKE~~
Signatures	CRYSTALS-Dilithium FALCON SPHINCS+	

Lattice Based Cryptography

Code-Based Cryptography

Hash Based Signatures

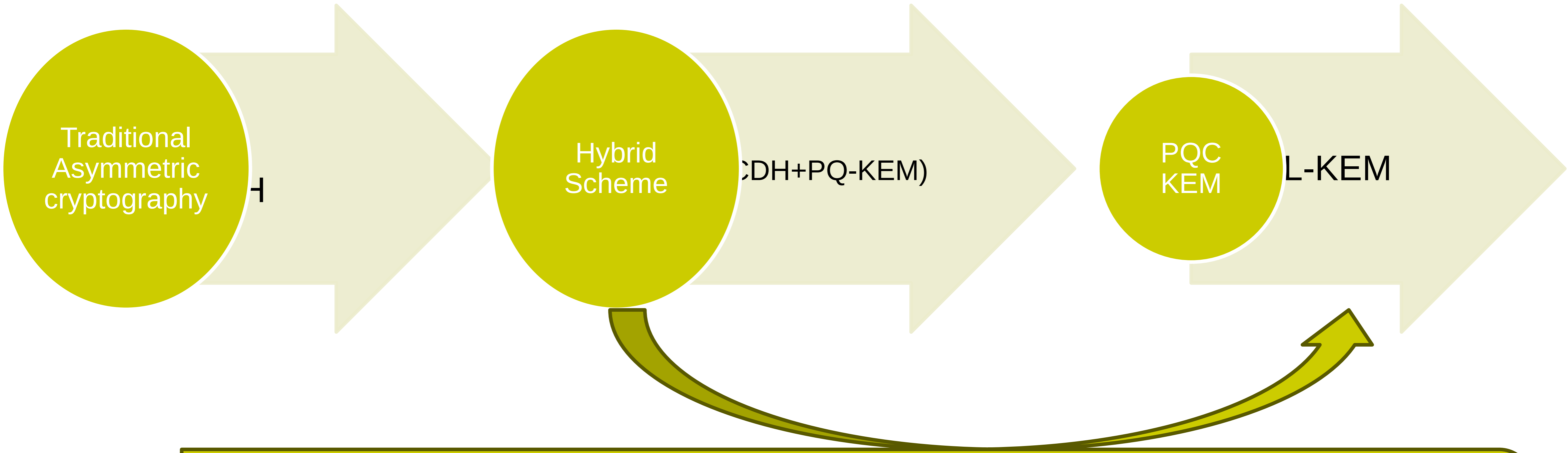
Isogeny based Cryptography : Broken!

KEM

```
                      +---------+ +---------+
                      | Client  | | Server  |
                      +---------+ +---------+
  ---------------------\ |           |
  | sk, pk = kemKeyGen()  |-|          |
  |---------------------| |           |
                         |           |
                         | pk        |
                         |---------->|
                         |           | -----------------------\
                         |           |-| ss, ct = kemEncaps(pk) |
                         |           | |----------------------|
                         |           |
                         |        ct |
                         |<----------|
-----------------------\ |           |
| ss = kemDecaps(ct, sk) |-|           |
|----------------------| |           |
                         |           |
```

Transition path

[PQ/T Hybrid KEM: HPKE with JOSE/COSE]()



FIPS 203 standard (ML-KEM) is a new CNSA 2.0 standard for PQ-KEM via lattice-based key establishment mechanism.
ML-KEM has been around 7 years and gone through many rounds of analysis
Hybrids can' be used when CRQC arrive and adds to computational cost.

PQ-KEM Encapsulation

- (SS', CT) = kemEncaps(recipPubKey)
- SS = KDF(SS', SSLen)

PQ-KEM Decapsulation

- SS' = kemDecaps(recipPrivKey, CT)
- SS = KDF(SS', SSLen)

JOSE Ciphersuite Registration

alg	Description
MLKEM512-KMAC128	ML-KEM-512 + KMAC128
MLKEM768-KMAC256	ML-KEM-768 + KMAC256
MLKEM1024-KMAC256	ML-KEM-1024 + KMAC256

Figure 1: Direct Key Agreement: Algorithms.

alg	Description
MLKEM512-KMAC128-AES128KW	ML-KEM-512 + KMAC128 + AES128KW
MLKEM768-KMAC256-AES256KW	ML-KEM-768 + KMAC256 + AES256KW
MLKEM1024-KMAC256-AES256KW	ML-KEM-1024 + KMAC256 + AES256KW

Figure 2: Key Agreement with Key Wrapping: Algorithms.

PQ KEM in JOSE

- Direct Key Agreement
- Key Agreement with Key Wrapping

Direct Key Agreement

PQ-KEM+KDF

```
{
  "alg": "MLKEM768-SHAKE256",
  "kid": "urn:ietf:params:oauth:jwk-thumbprint:sha-
56:adjwW6fyyZ94ZBjGjx_OpDEKHLG",
  “kem-ct": "yDVZLsO7-ecy_GCgEluwn9U723TCHNAzeYRRQPOfpHM",
  "enc": "A256GCM"
}
```

KEM Ciphertext

Key Agreement with Key Wrapping

```
{
  "protected": "eyJlbmMiOiJBMTI4R0NNIn0",
  "ciphertext": "S0qqrM3xXPUavbmL9LQkgUKRBu8BZ7DQwoT-mdNIZYU-ip_V-
fbMokiGwp2aPM57DX3cXCK3TKHqdhZ8rSNduUja",
  "iv": "AzaXpooLg3ZxEASQ",
  "aad": "8J-SgCBhYWQ",
  "tag": "S0omWw35S0H7tyEHsmGLDw",
  "recipients": [
    {
      "encrypted_key": "yDVZLsO7-ecy_GCgEluwn9U723TCHNAzeYRRQPOfpHM",
      "header": {
        "kid": "urn:ietf:params:oauth:jwk-thumbprint:sha-256:adjwW6fyyZ94ZBjGjx_OpDEKHLG",
        "alg": "MLKEM768-SHAKE256+A256KW",
       "kem-ct": "yDVZLsO7-ecy_GCgEluwn9U723TCHNAzeYRRQPOfpHM"
      }
    }
  ]
}
```

{"enc": "A128GCM"}

KEM Ciphertext

PQ-KEM+KDF

Next Steps

- Consider for WG adoption
- Comments and suggestions are welcome