Post-Quantum and Hybrid enhancements for EAP-AKA’

draft-ar-emu-pqc-eapaka-02

draft-ra-emu-pqc-eapaka-01

IETF 120 Vancouver

Aritra Banerjee

K Tirumaleswar Reddy

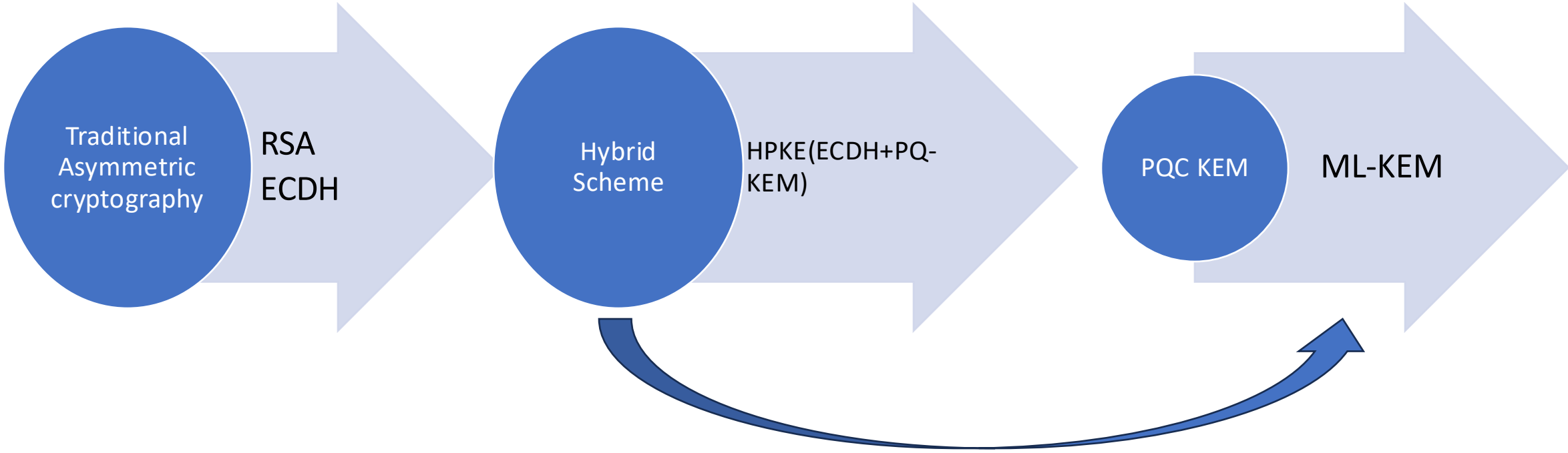
Motivation

- EAP-AKA' FS [I-D.ietf-emu-aka-pfs] provides updates to [RFC9048] with an optional extension that offers ephemeral key exchange using the traditional ECDHE key agreement algorithm for achieving perfect forward secrecy (PFS).

- However, it is susceptible to future threats from CRQCs, which could potentially help an attacker to derive the private key from the public key.

- If the adversary using CRQC has also obtained knowledge of the long-term key and ephemeral public key, it could compromise session keys generated as part of the authentication run in EAP-AKA'.

Transition Path to Post-Quantum EAP-AKA’



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 for more than 8 years and gone through many rounds of analysis
Hybrids can’ be used when CRQC arrive and adds to computational cost.

PQ/T Hybrid enhancements for EAP-AKA’

draft-ar-emu-pqc-eapaka-02

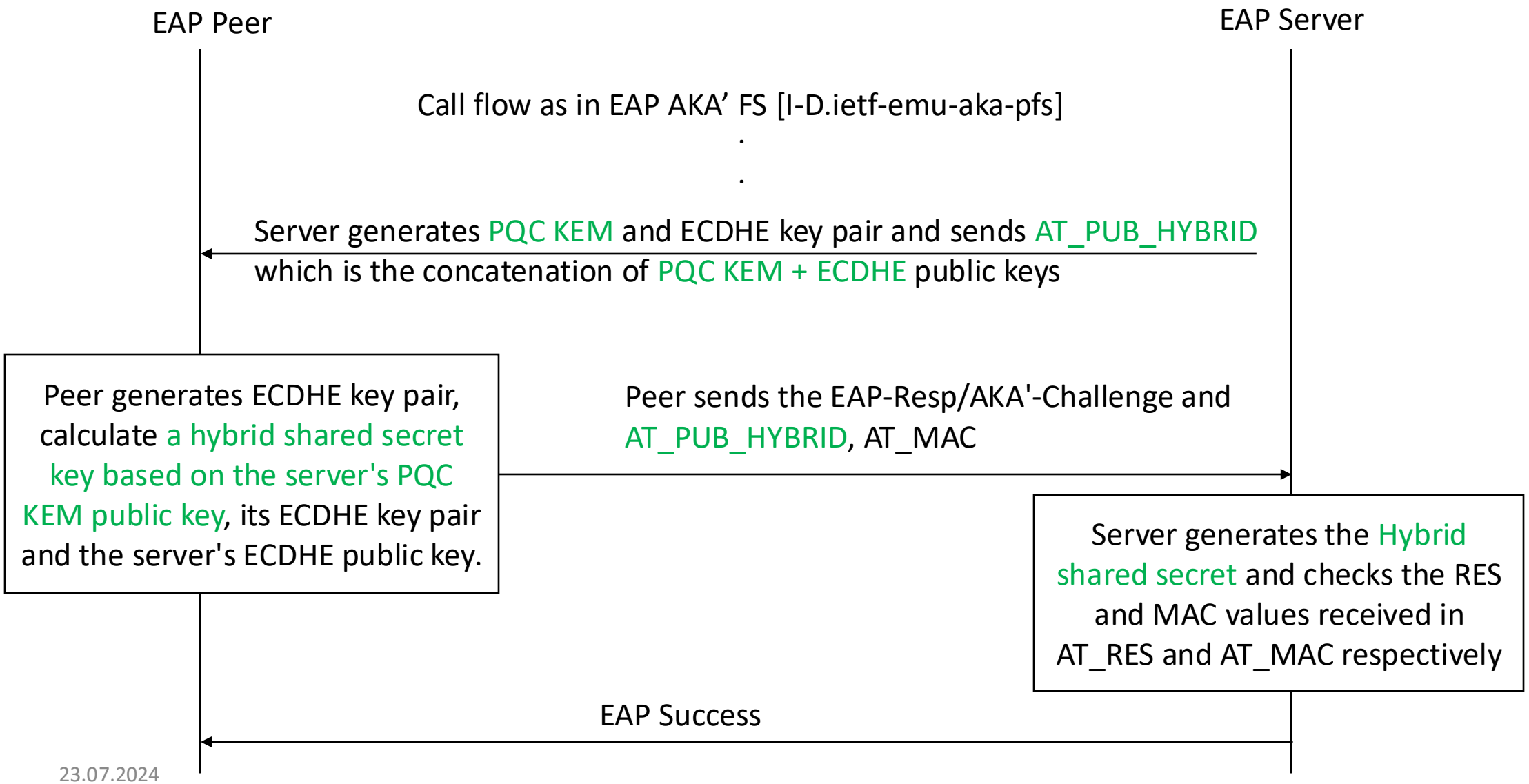
HPKE

- The HPKE specification provides a variant of public key encryption of arbitrary-sized plaintexts for a recipient public key.

- HPKE (Hybrid Public Key Encryption) emerged in the IETF as a prominent public key encryption scheme
  - https://www.rfc-editor.org/rfc/rfc9180.html (Developed by CFRG in IRTF)
  - Used by several protocols Oblivious HTTP, Encrypted Client Hello in TLS, MLS and COSE/JOSE

- HPKE interfaces are friendly to hybrid encryption

Overview of the protocol



Generating Hybrid Master Key

- MK = PRF'(IK'|CK',"EAP-AKA'"|Identity)

- HYBRID_SHARED_SECRET, enc = Encapsulate(pKR)
- MK_HYBRID = PRF'(IK'|CK'| HYBRID_SHARED_SECRET,"EAP-AKA' FS"| Identity)

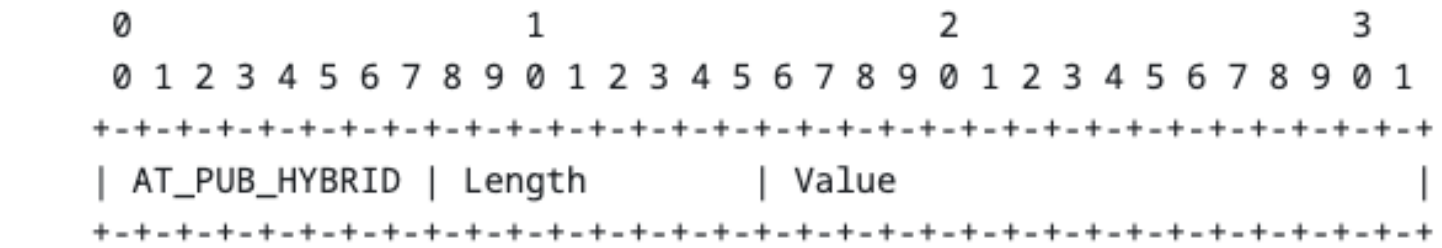
- K_encr = MK[0..127]
- K_aut = MK[128..383]
- K_re = MK_HYBRID [0..255]
- MSK = MK_HYBRID [256..767]
- EMSK = MK_HYBRID [768..1279]

Overview

- A new attribute, AT_PUB_HYBRID, is defined to carry the public key, which is the concatenation of traditional and PQC KEM public keys from the EAP server.
- The AT_PUB_HYBRID attribute will carry the encapsulated key, which is formed by concatenating the encapsulated key (enc) from the traditional KEM algorithm and the ciphertext (ct) from the PQC KEM Encapsulation function from the EAP peer.
- The AT_KDF_FS attribute is updated to indicate the HPKE KEM and HKDF for generating the Hybrid Master Key MK_HYBRID.
- The Hybrid key derivation function will be included first in the EAP-Request to indicate a higher priority than the traditional key derivation function.

AT_PUB_HYBRID attribute

The format of the AT_PUB_HYBRID attribute is shown below.



Value:

EAP-Request: It contains the public key, which is the concatenation of traditional and PQC KEM public keys from the EAP server.

EAP-Response: It contains the encapsulated key, which is formed by concatenating the encapsulated key (enc) from the traditional KEM algorithm and the ciphertext (ct) from the PQC KEM Encapsulation function from the EAP peer.

Next Steps

- Comments and Suggestions are welcome
- Consider for WG Adoption

Post Quantum KEM mechanisms for EAP-AKA’

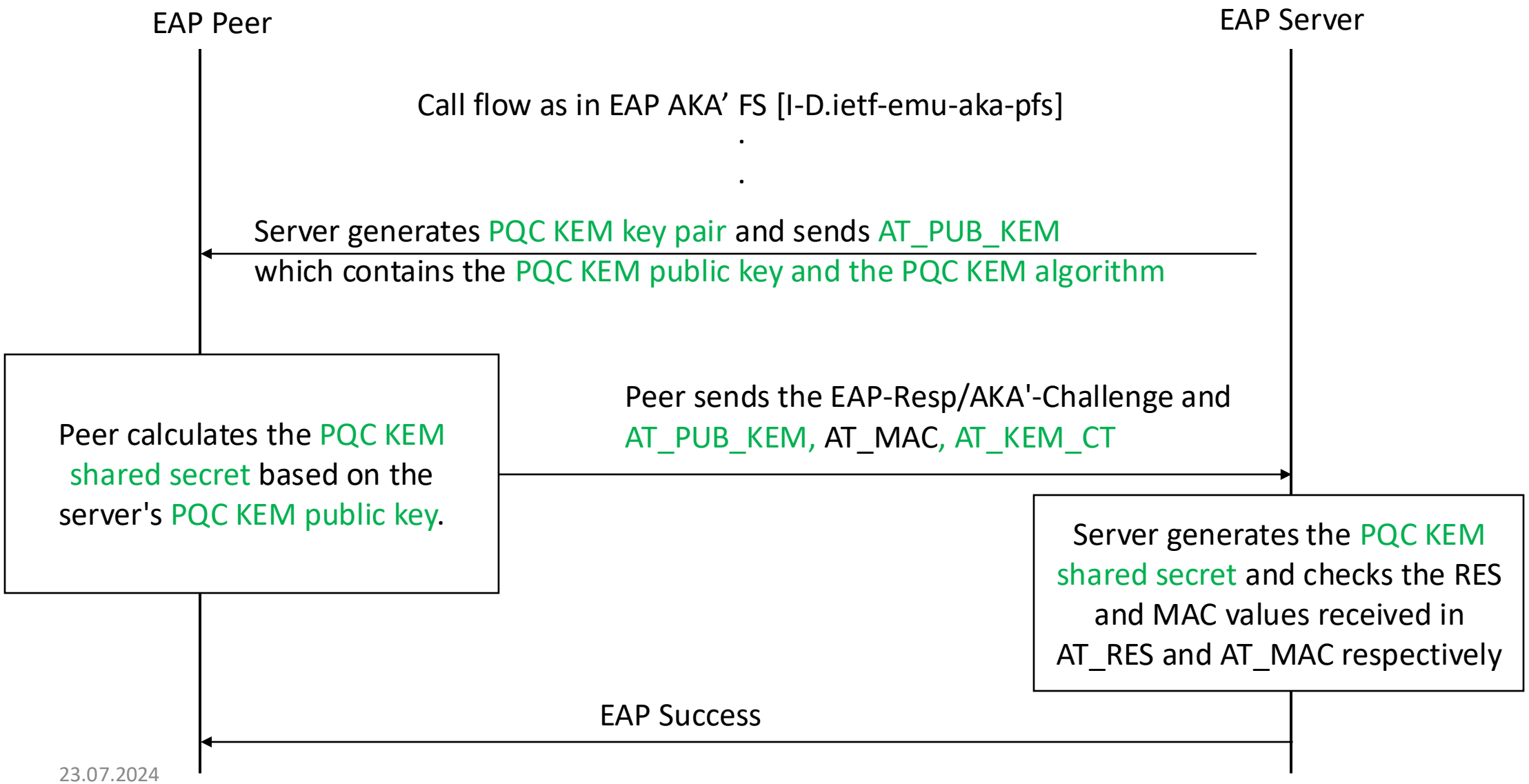
draft-ra-emu-pqc-eapaka-01

PQ Key Encapsulation Mechanism (KEMs)

- Key Encapsulation Mechanism (KEM) can be any asymmetric cryptographic scheme comprised of algorithms satisfying the following interfaces [PQCAPI].
  - def kemKeyGen() -> (pk, sk)
  - def kemEncaps(pk) -> (ct, ss)
  - def kemDecaps(ct, sk) -> ss

  where pk is public key, sk is secret key, ct is the ciphertext representing an encapsulated key, and ss is shared secret.

Overview of the protocol



Generating Post-Quantum Master Key

- MK = PRF'(IK'|CK',"EAP-AKA'"|Identity)
- ct, ss = kemEncaps(pKR)

- MK_PQ_SHARED_SECRET = PRF'(IK'|CK'|ss,"EAP-AKA' FS"| Identity | ct)

- K_encr = MK[0..127]
- K_aut = MK[128..383]
- K_re = MK_PQ_SHARED_SECRET [0..255]
- MSK = MK_PQ_SHARED_SECRET [256..767]
- EMSK = MK_PQ_SHARED_SECRET [768..1279]

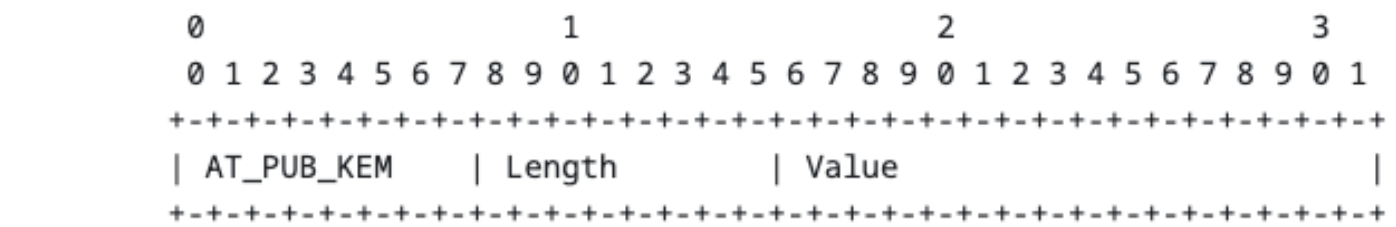
Overview

- A new attribute, AT_PUB_KEM, is defined to carry the PQC KEM public key from the EAP server.
- The AT_KEM_CT attribute will carry the ciphertext (ct) from the PQC KEM Encapsulation function from the EAP peer.
- The AT_KDF_FS attribute is updated to indicate the PQC KEM and HKDF for generating the PQC Master Key MK_PQ_SHARED_SECRET.
- The PQC key derivation function will be included first in the EAP-Request to indicate a higher priority than the traditional key derivation function.

AT_PUB_KEM attribute

9.1. AT_PUB_KEM

The format of the AT_PUB_KEM attribute is shown below.



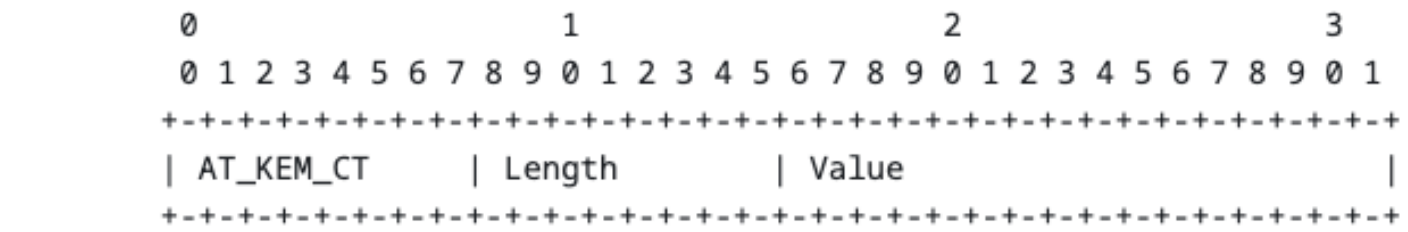
Value:

EAP-Request: It contains the public key, which is the PQC KEM public key from the EAP server. Because the length of the attribute must be a multiple of 4 bytes, the sender pads the Value field with zero bytes when necessary.
To retain the security of the keys, the sender SHALL generate a fresh value for each run of the protocol.

AT_KEM_CT attribute

9.2. AT_KEM_CT

The format of the AT_KEM_CT attribute is shown below.



Value:

EAP-Response: It contains the ciphertext (ct) from the PQC KEM Encapsulation function from the EAP peer. Because the length of the attribute must be a multiple of 4 bytes, the sender pads the Value field with zero bytes when necessary.
To retain the security of the keys, the sender SHALL generate a fresh value for each run of the protocol.

Next Steps

- Comments and Suggestions are welcome
- Consider for WG adoption