ML-KEM Post-Quantum Key Agreement for TLS 1.3

draft-ietf-tls-mlkem-01

https://datatracker.ietf.org/doc/draft-ietf-tls-mlkem/

https://github.com/tlswg/draft-ietf-tls-mlkem/

A pure-PQ ciphersuite for TLS 1.3

- No purely post-quantum ciphersuites
- Fills in the other side of draft-ietf-tls-hybrid-design
- Adopted
- If PQ-only works for you, clean key agreement, no hybrid duplicate shares or mixing and matching logic

NamedGroups: MLKEM512, MLKEM768, MLKEM1024

```
enum {

     ...,

      /* ML-KEM Key Agreement Methods */
      mlkem512(0x0200),
      mlkem768(0x0201),
      mlkem1024(0x0202)

     ...,

} NamedGroup;
```

Codepoints allocated:

512	MLKEM512	Y	N	[draft-connolly-tls-mlkem-key-agreement-03]	FIPS 203 version of ML-KEM-512
513	MLKEM768	Y	N	[draft-connolly-tls-mlkem-key-agreement-03]	FIPS 203 version of ML-KEM-768
514	MLKEM1024	Y	N	[draft-connolly-tls-mlkem-key-agreement-03]	FIPS 203 version of ML-KEM-1024

Client sends encaps key, server replies with ciphertext

```
struct {
    NamedGroup group;
    opaque key_exchange<1..2^16-1>;
} KeyShareEntry;
```

```
These are transmitted in the `extension_data` fields of
`KeyShareClientHello` and `KeyShareServerHello` extensions:

~~~
    struct {
        KeyShareEntry client_shares<0..2^16-1>;
    } KeyShareClientHello;

    struct {
        KeyShareEntry server_share;
    } KeyShareServerHello;
~~~
```

KEM shared secret is input to Handshake Secret derivation

```
                       v
                 Derive-Secret(., "derived", "")
                       |
                       v
shared_secret -> HKDF-Extract = Handshake Secret
^^^^^^^^^^^^^          |
                       +------> Derive-Secret(...)
                       +------> Derive-Secret(...)
                       |
                       v
                 Derive-Secret(., "derived", "")
                       |
```

KEM shared secret is input to Handshake Secret derivation

```
                       v
                 Derive-Secret(., "derived", "")
                       |
                       v
shared_secret -> HKDF-Extract = Handshake Secret
^^^^^^^^^^^^^          |
                       +-----> Derive-Secret(...)
                       +-----> Derive-Secret(...)
                       |
                       v
                 Derive-Secret(., "derived", "")
                       |
```

Since 122 Bangkok

- Adopted
- Several reviews on-list
  - Removed unused normative ref
  - Removed security considerations section on fixed lengths
  - Remove paragraph on resilience to MAL and LEAK binding adversary model
  - Remove paragraph describing DH use in TLS 1.3 modeled as a KEM
  - Tighten up the failure rate section
  - Tighten up the requirements of TLS 1.3 key agreement and how ML-KEM satisfies
  - Tighten up section on reuse bounds
  - Redundant 'standard', ‘fully’ -> ‘purely’
  - Fix line around fixed lengths/choosing algorithms
  - Remove section on larger sizes of public stuff
  - Add acknowledgments
- Pushed -02

ML-KEM Post-Quantum Key Agreement for TLS 1.3

draft-ietf-tls-mlkem-01

https://datatracker.ietf.org/doc/draft-ietf-tls-mlkem/
https://github.com/tlswg/draft-ietf-tls-mlkem/