Using classic McEliece in IKEv2

draft-smyslov-ipsecme-ikev2-mceliece

What is McEliece?

- A Key Encapsulation Method (KEM) designed for IND-CCA2 security, even against quantum computers.
  - Quoting from draft-josefsson-mceliece-03.
- Introduced in 1978; current documents date from 2022.
- Key agreement requires a large public key, and a relatively small ciphertext.
  - For the most common parameter, 1 MB and 208 bytes respectively.
- A lot of interest in many places because of its tried-and-true nature compared to NIST algorithms.

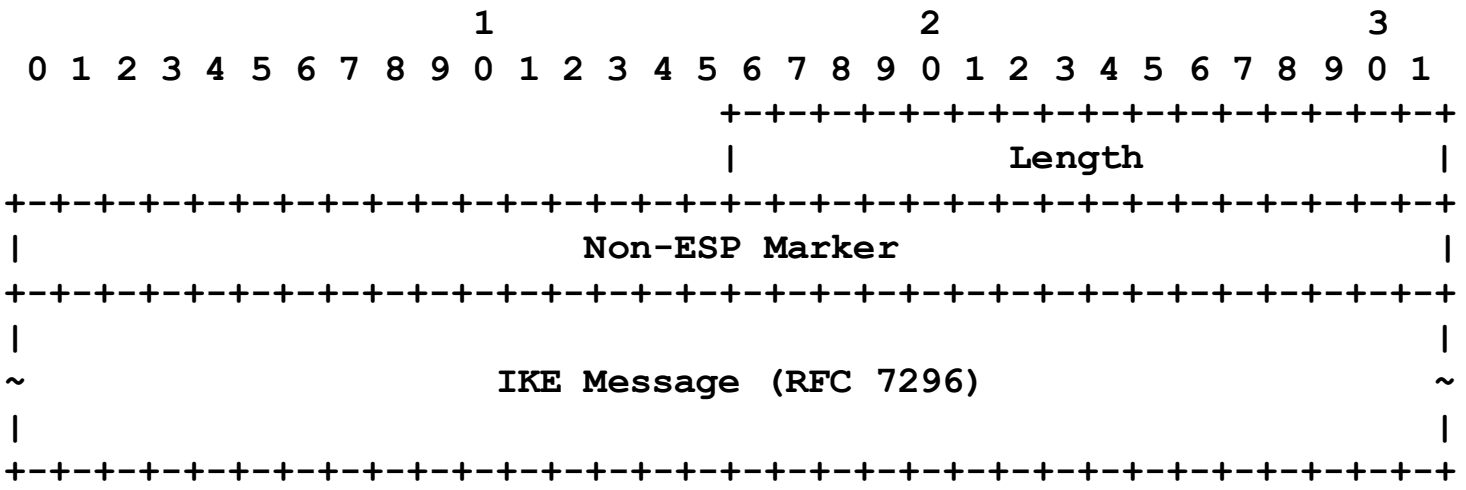
Challenges (1/3)

- Biggest challenge is sending the public key.
- An IKEv2 Payload has a 16-bit length. You can’t fit a 1-MB public key in a KE payload. We need a way to transport larger payloads.
  - There is draft-nir-ipsecme-big-payload.
- Even if you have your big payload, you can’t use it without fragmentation, because the limit of UDP and IP is 64K
  - So you need IKE message fragmentation (RFC 7383)

Challenges (2/3)

- Can TCP encapsulation (RFC 9329) save us? No, that Length field is 16-bit again.
- You still want TCP for reliability when transporting that 1-MB payload, though.

3.1. TCP-Encapsulated IKE Message Format



Challenges (3/3)

- But our problems with that 1-MB public key are not done yet.
- Both big payload and fragmentation need to be negotiated in the IKE_SA_INIT exchange, and used only later.
  - We need Intermediate exchange (RFC 9242)
  - We need the additional key exchange extension (RFC 9370)
- And if we're using both of these, we need some other key exchange (ECDH?) in the IKE_SA_INIT.

Other work in the IETF

- draft-josefsson-mceliece – Classic McEliece
- draft-josefsson-ssh-mceliece – “Secure Shell Key Exchange Method Using Chempat Hybrid of Classic McEliece and X25519 with SHA-512”
- draft-wagner-tls-keysharepqc – “New Key Share Extension for Classic McEliece Algorithms”

- Like I said, lots of interest, but still no adopted draft

So what are we asking?

- Not (yet) adoption.
- For the purposes of key exchange, other than payload length limits, there is nothing special about IPsec compared to TLS and SSH.
- Might as well follow the other drafts to see that we are reaching similar conclusions about formats and parameter sets.
- Two things we *can* consider adopting:
  - The big payload draft – draft-nir-ipsecme-big-payload
  - The transport separation draft – draft-smyslov-ipsecme-ikev2-reliable-transport – to allow TCP for IKE, but ESP for, well, ESP.

Questions?