Network Performance Measurement for IPsec

draft-ietf-ippm-ipsec-00

Kostas Pentikousis, Yang Cui , Emma Zhang

IETF 87

Berlin, Germany

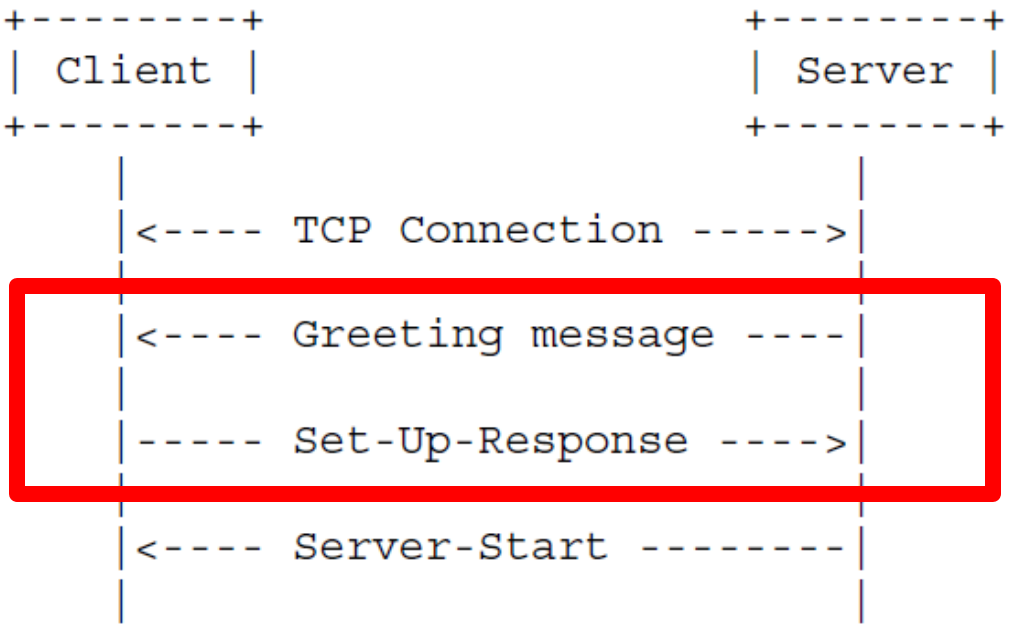
Background

- OWAMP [RFC 4656], TWAMP [RFC 5357]
  - Discussion on security protection in the past
  - Decision to develop a dedicated security mechanism and give up on TLS, DTLS, IPsec
  - Unauthenticated, authenticated, and encrypted modes
- Today: interested in stats about the actual deployment of the authenticated and encrypted modes in practice
  - Cf. IKEv2/IPsec deployment

Proposed Enhancement

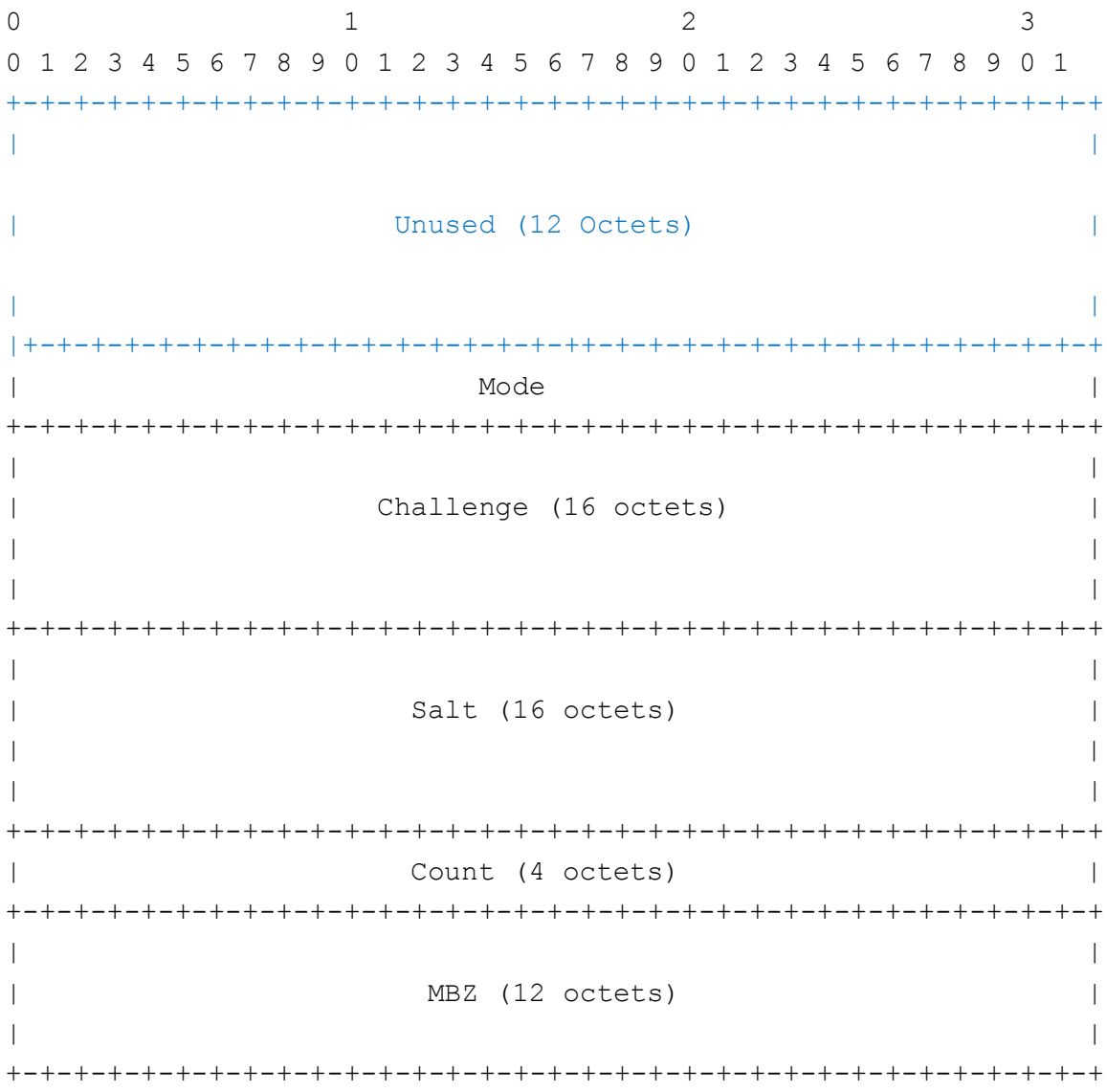
- Today: O/TWAMP security mechanism
  - Based on shared secret, does not support credential or certificates
  - Four (4) keys for integrity and encryption protection
    - AES keys: OWAMP-Control, OWAMP-Test
    - HMAC keys: OWAMP-Control, OWAMP-Test
- Proposal: Use IKEv2/IPsec to feed the key to O/TWAMP
  - Well-known and well-designed security mechanism
  - Enhance security protection, key negotiation
  - Support certificate based key exchange
  - Extend to automatic key management

OWAMP-Control Initiation

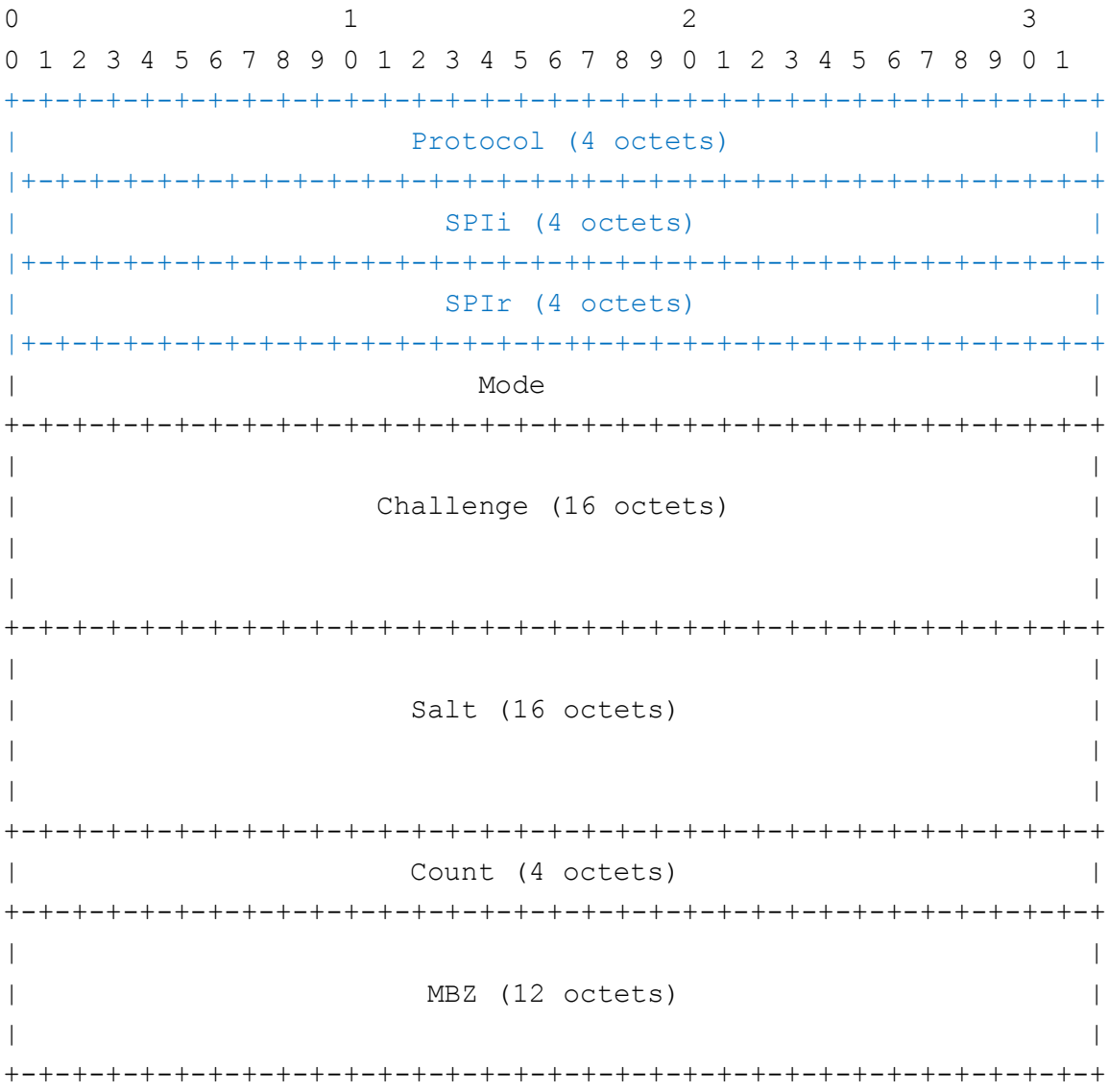


- Replace the current O/TWAMP security root (shared secret key) with keys derived from the existing IKEv2/IPsec SA
- In other words, instead of inventing a passphrase, take advantage of IKEv2/IPsec

Server Greeting [RFC 4656]



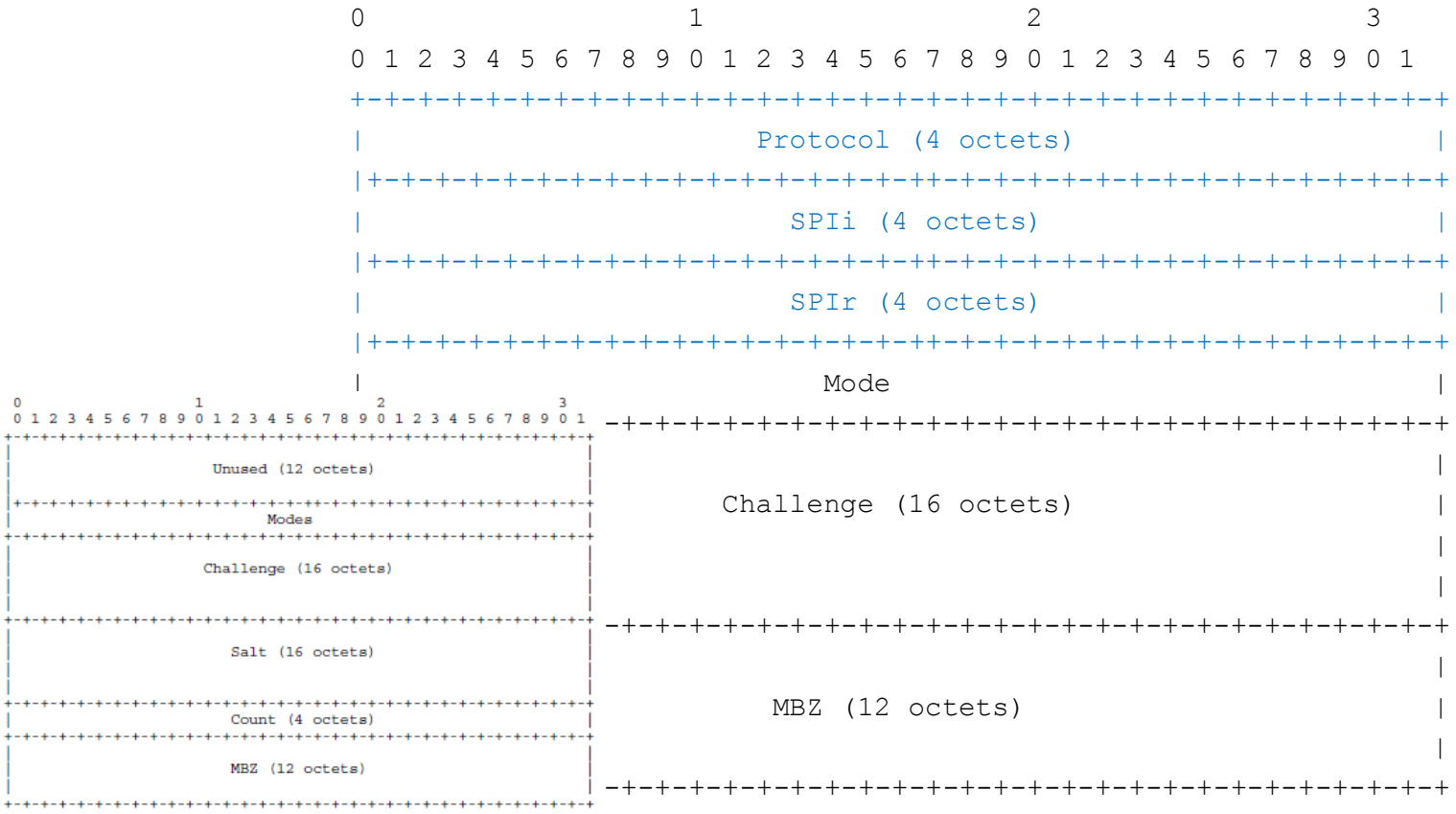
Server Greeting [-ippm-ipsec]



Key Derivation Update

- Up to now we only considered the key derivation part, taking advantage of IKEv2/IPsec
- The rest of the protocol operation remains unchanged, as do the other parts of the Server Greeting
- We could do more, however
- That is, we could reduce the number of keys used, and optimize the Greeting and Set-Up-Response messages

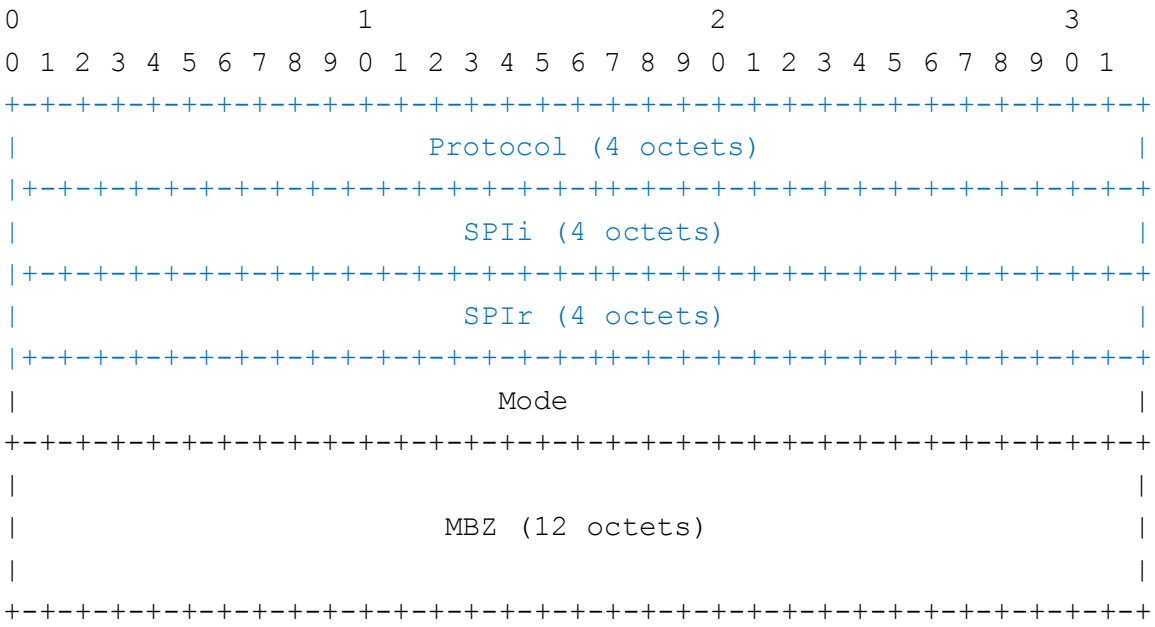
Greeting Optimization 1 [-ippm-ipsec]



Session key for enc = PRF{ root key of O/TWAMP, "O/TWAMP enc" }

Session key for auth = PRF{ root key of O/TWAMP, "O/TWAMP auth" }

Greeting Optimization 2 [-ippm-ipsec]



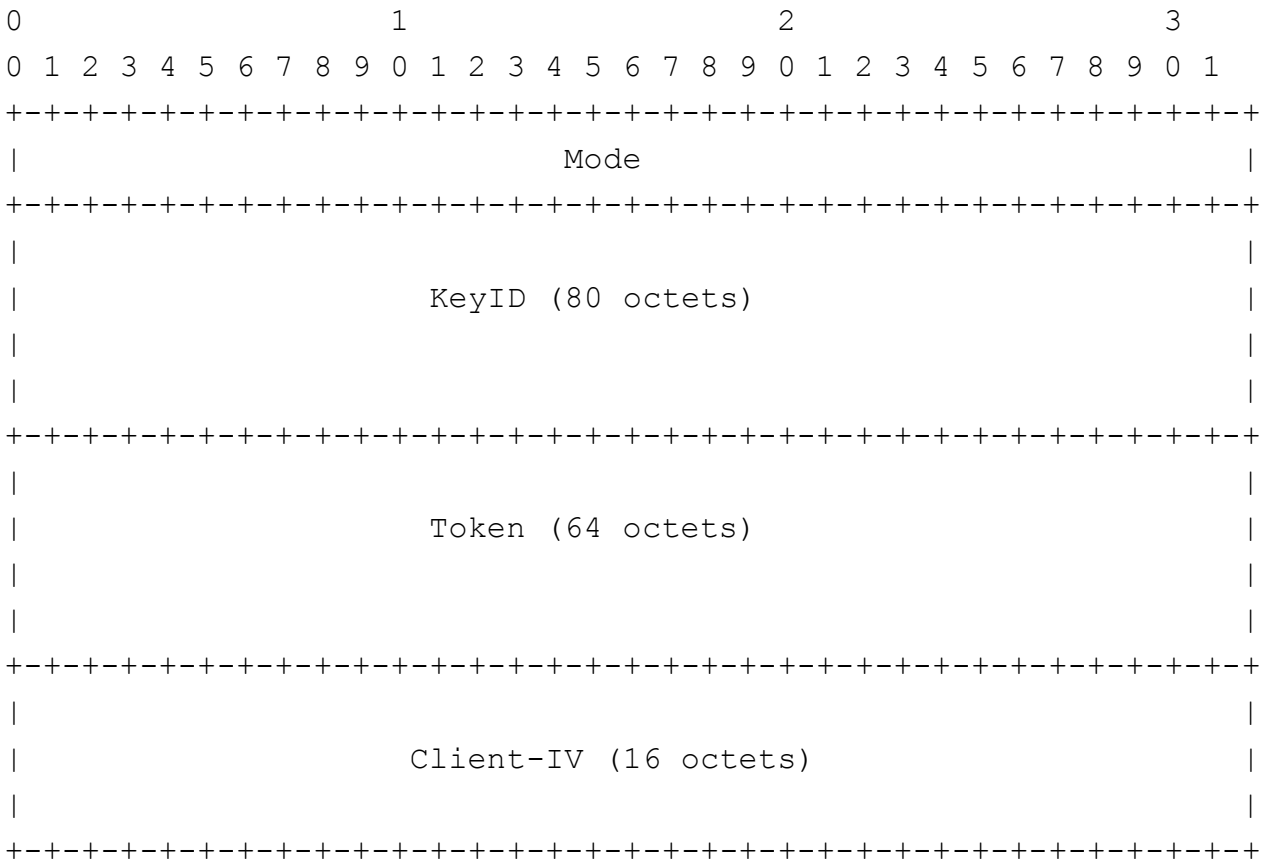
Session key for enc = encryption key of the IPsec SA

Session key for auth = integrity key of the IPsec SA

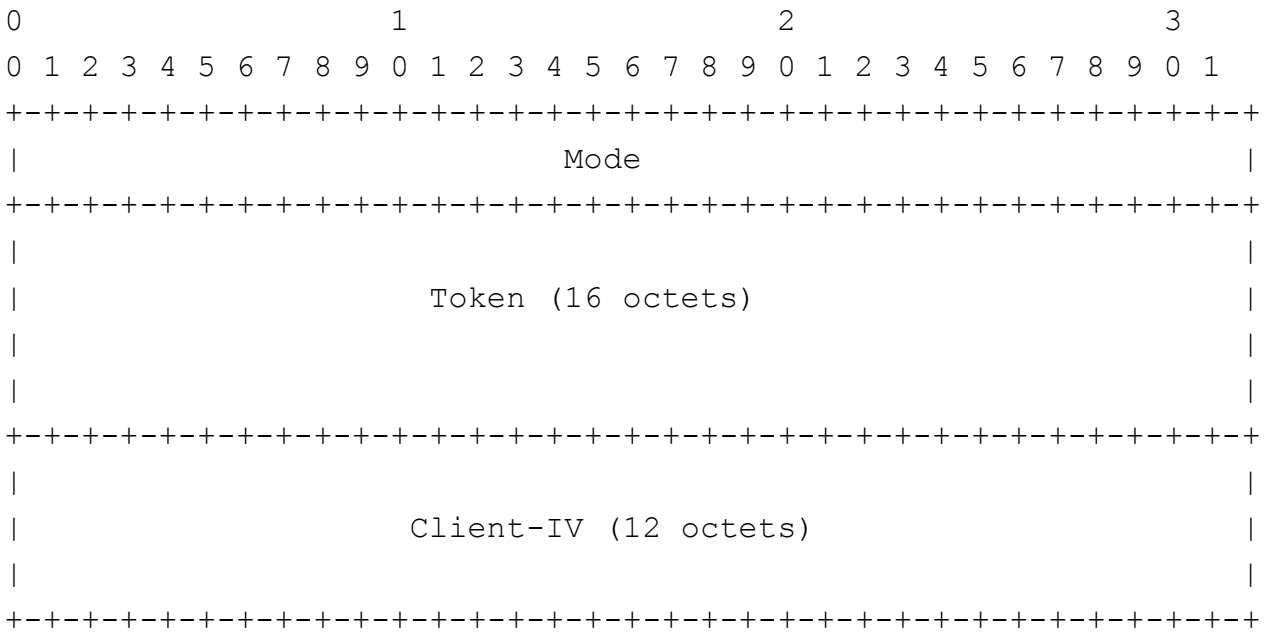
Optimizing Set-Up-Response

- KeyID is used for the shared secret
  - If keys are derived from IKEv2/IPsec, we do not need it.
  - Shave 80 octets
- Token is defined as a concatenation of Challenge (16) + AES Session Key (16) + HMAC-SHA1 (32 octets)
  - With IKEv2/IPsec in place, we do not need to generate the key at the client and send it to the server
  - Thus, the Token can be reduced in size (keep only the Challenge part in Optimization 1) or even be eliminated (Optimization 2)

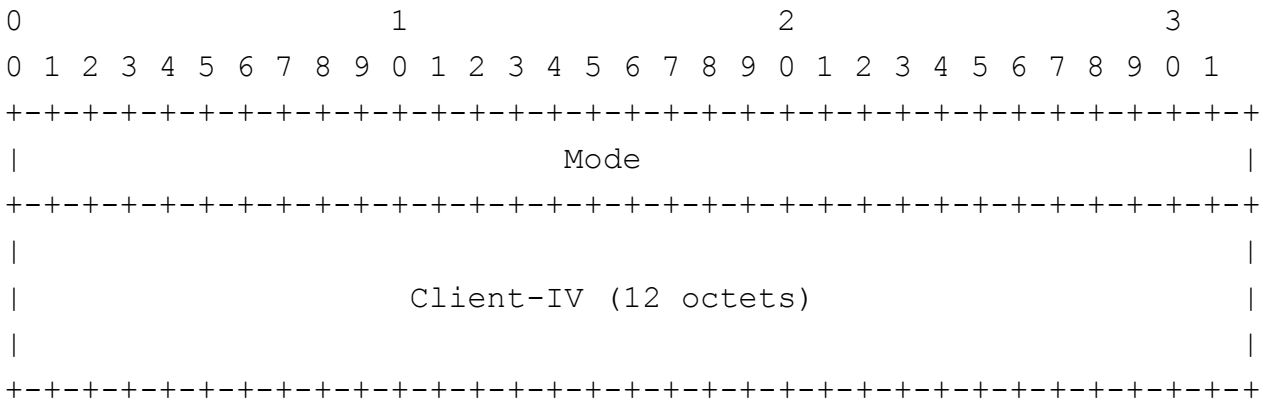
Set-UP-Response [RFC 4656]



Optimization 1 [-ippm-ipsec]



Optimization 2 [-ippm-ipsec]



Proposal Advantages

- Use of well-understood, widely-implemented IKEv2/IPsec to replace a specialized security mechanism
  - Enhance O/TWAMP security
- Support cert-based key exchange
  - More flexible in practice and more efficient
- Ease key management in shared secret model
  - The use of IKEv2/IPsec makes it easier to extend automatic key management.
- Reduce verbosity
  - Server Greeting from 64 octets down to 44 (Opt1) or 28 (Opt2)
  - Set-Up-Response from 164 octets down to 32 (Opt1) or 16 (Opt2)
- Community Document: please contribute!

Way Forward

- Feedback from WG
  - Key derivation
  - Alternative optimizations
  - Existing implementation impact (we argue it's minimal)
- Revision towards -01