More Thoughts on Encryption for SIEPON

CableLabs

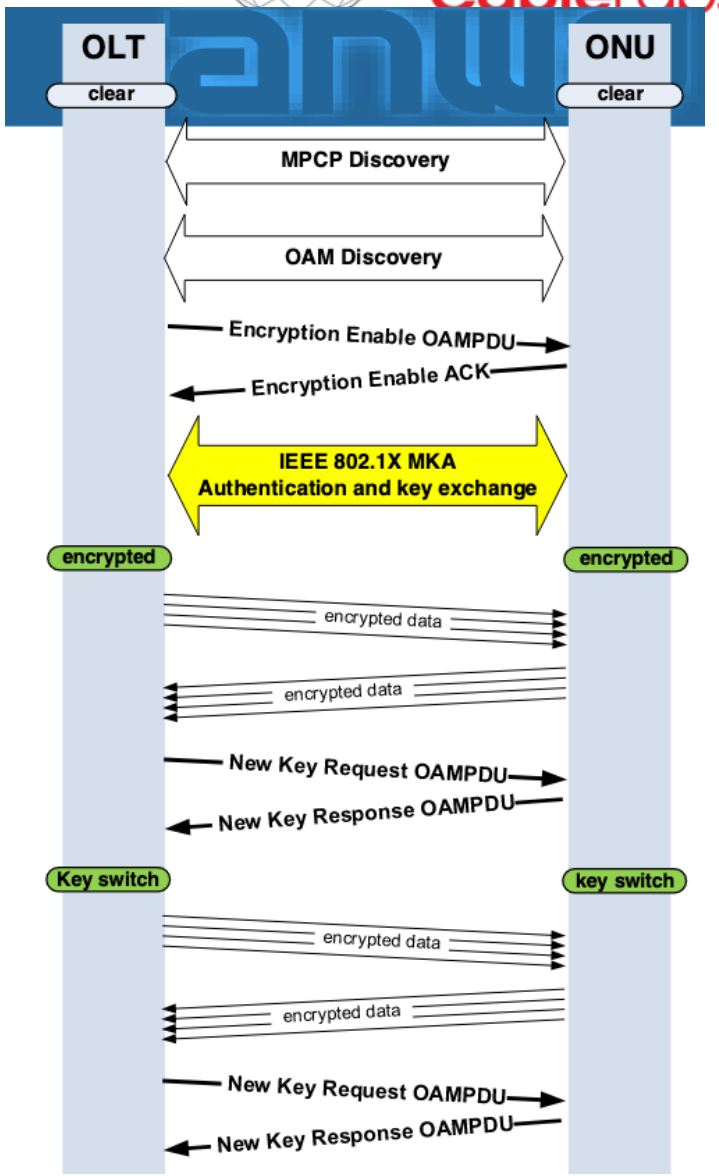
Steve Goeringer | s.goeringer@cablelabs.com

Encryption Initialization (re Kramer 02/06/2023 pg 2)

- Considering implications/issues of MKA
- Not sure of the EAP and MKA encapsulation here – EAPOL for EAP and MKPDU/EAPOL for MKA
- What's the purpose of the encryption enable handshake?
- Using MKA absent MACSec may require design
- MKA session negotiation determines the key server, whether devices are MACSec capable, the key server generates a key name and SAK, and the devices start doing encryption using the SAK
- New key request OAMPDU – does this contain the new SAK or is a trigger for an MKA message exchange?
  - When the SAK exhausts (~5mins at 10Gbps), a new SAK is determined when using MACSec with AES-128-GCM
  - What triggers the key request? Key agreement MUST occur BEFORE the current key exhausts.

Security & Privacy Technologies

CableLabs

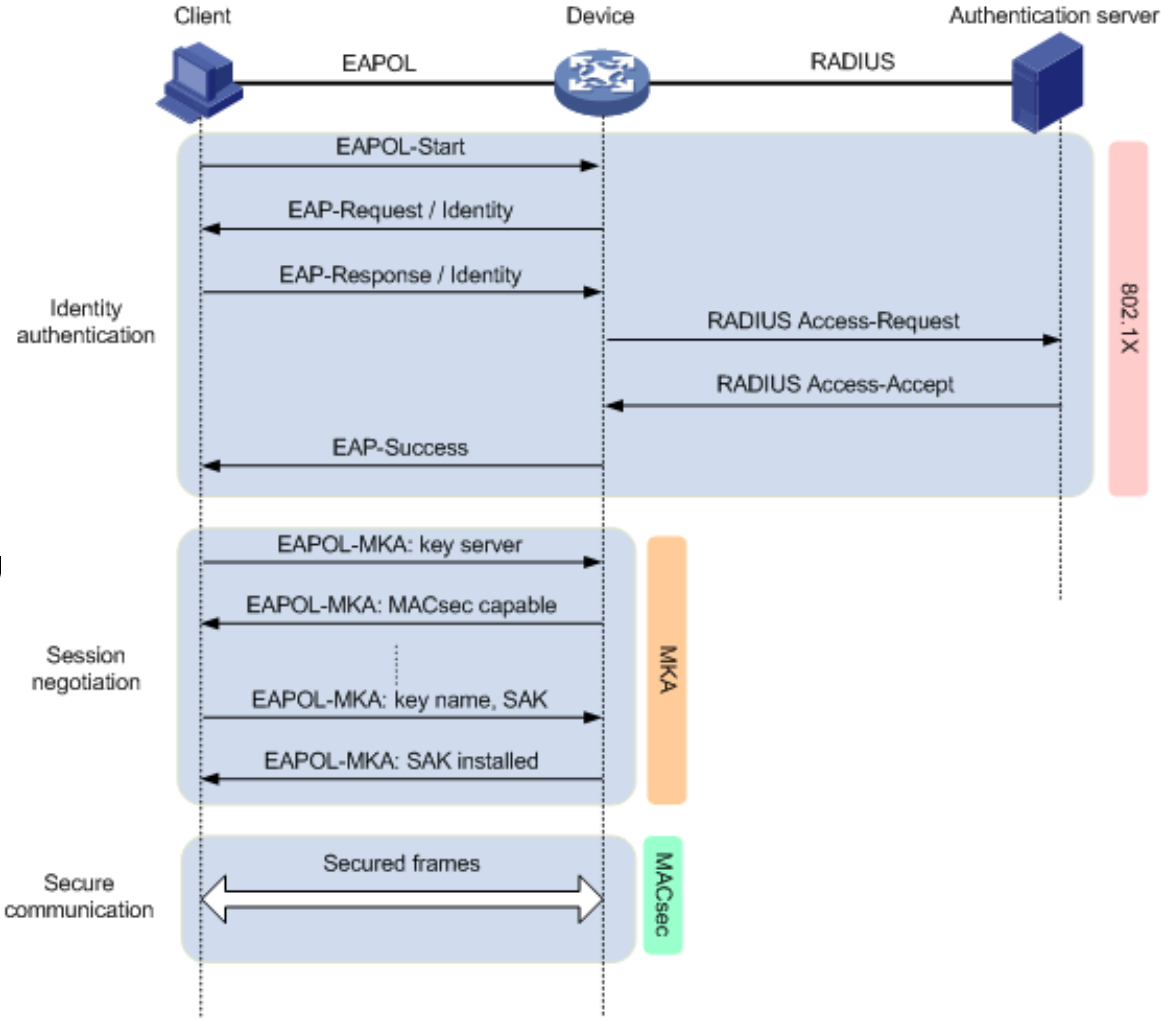


MACsec operating mechanism

Security & Privacy Technologies

CableLabs®

- Operating mechanism for client-oriented mode
- https://techhub.hpe.com/eginfolib/networking/docs/switches/5510hi/5200-0019b_security_cg/content/471724305.htm



DPoEv2 SEC

Security & Privacy Technologies

CableLabs®

- Little nit – I think both the OLT and ONU derive the KEK and CAK
- SAK key server is implicitly the OLT

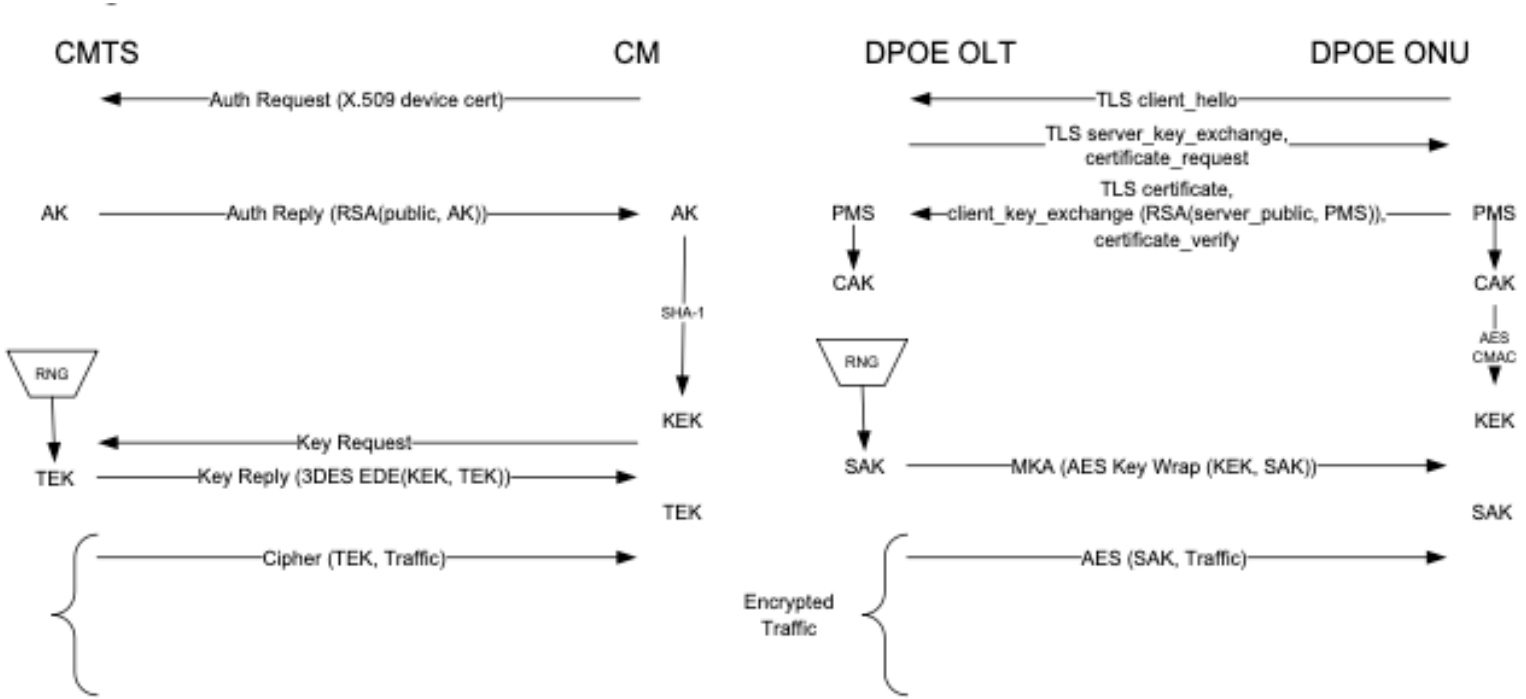


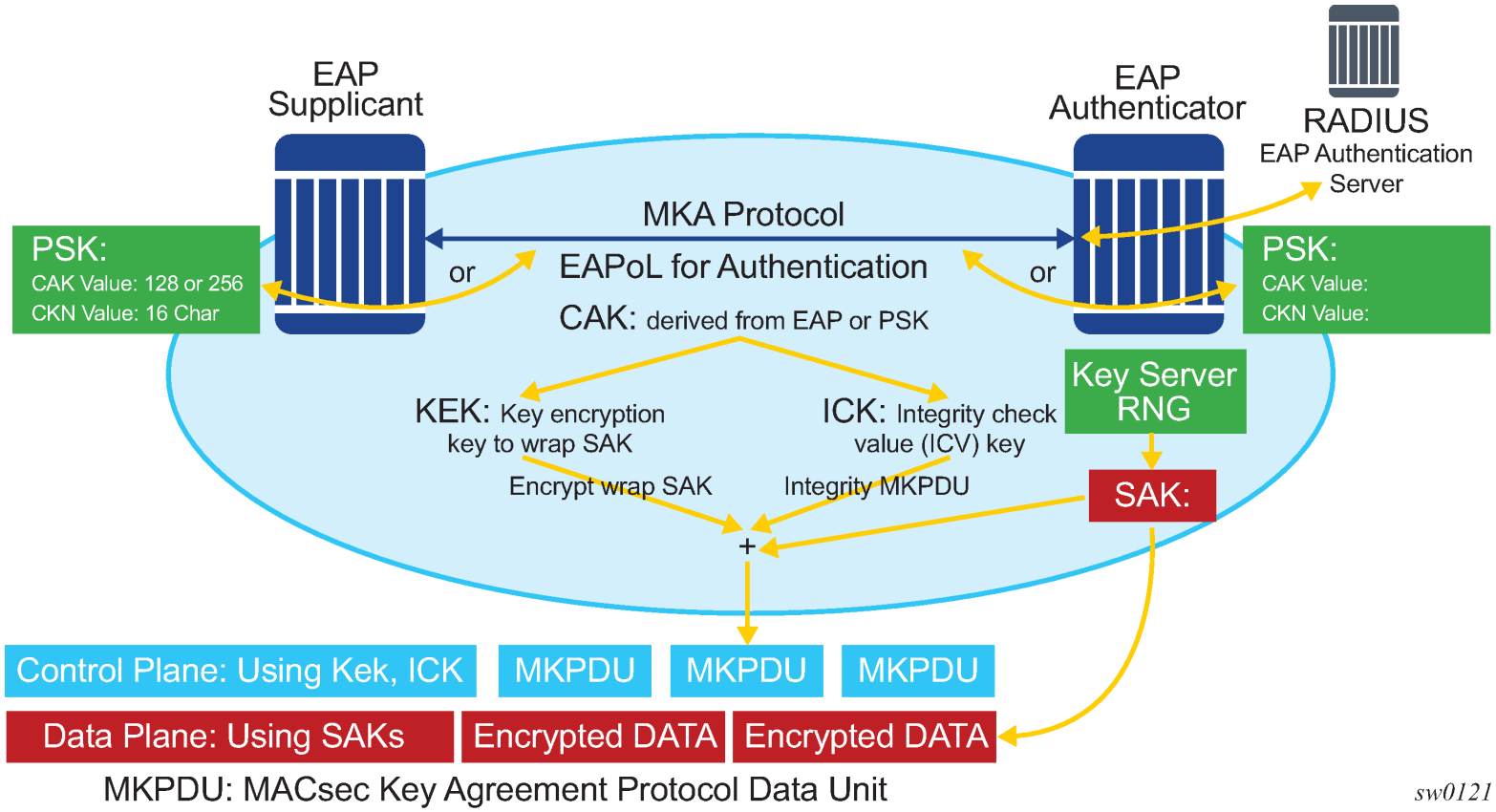
Figure 16 - Authentication in Bidirectional Methods

Lots and lots and lots of keys…

Security & Privacy Technologies

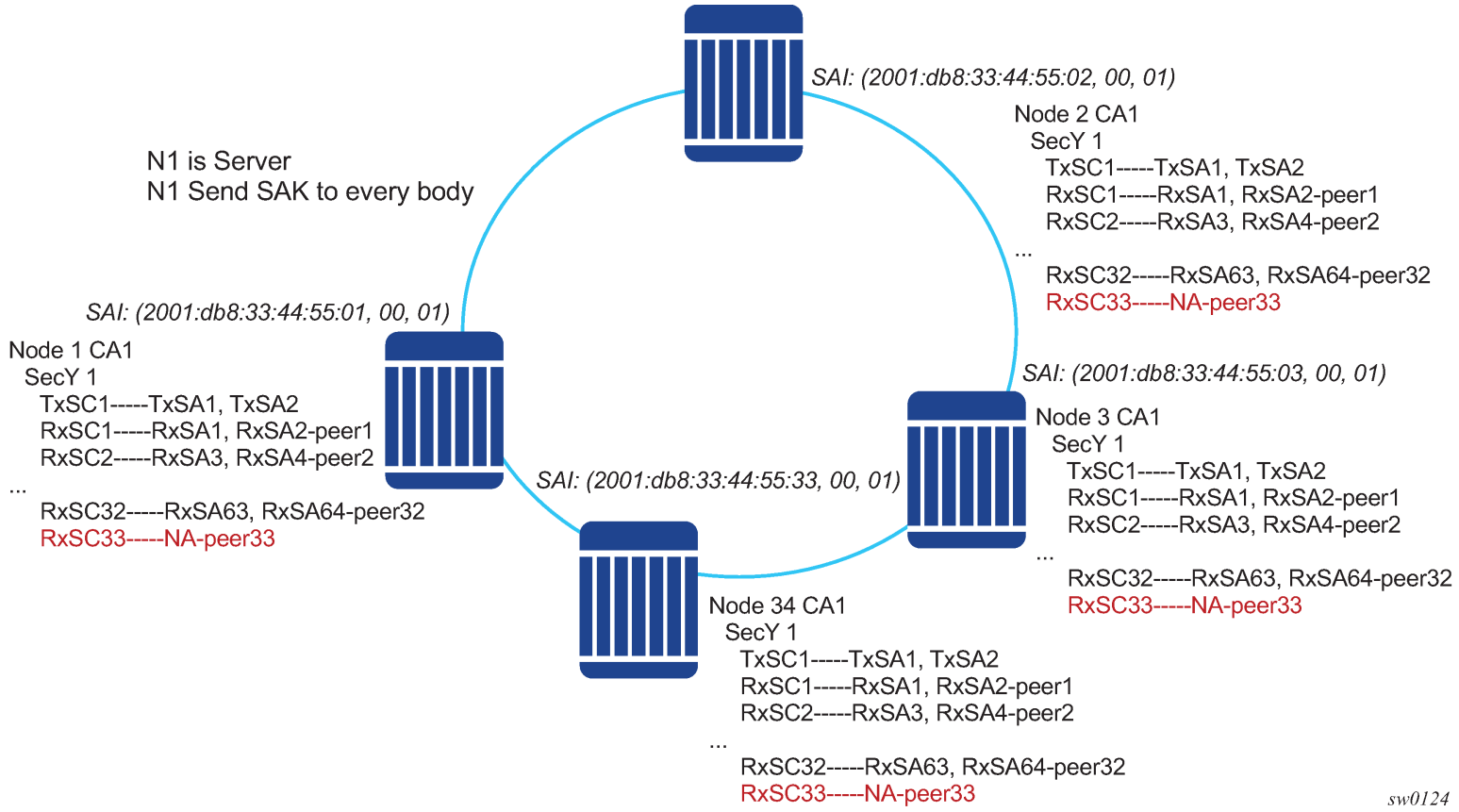
CableLabs®

- https://infocenter.nokia.com/public/7750SR217R1A/index.jsp?topic=%2Fcom.nokia.Interface_Configuration_Guide_21.7.R1%2Fmacsec_static_c_c-ai9emdynxp.html



Point to Multi-point

- https://infocenter.nokia.com/public/7750SR217R1A/index.jsp?topic=%2Fcom.nokia.Interface_Configuration_Guide_21.7.R1%2Fmacsec_static_c-ai9emdynxp.html



From 802.1ae-2018

Security & Privacy Technologies

CableLabs

- 7.1 -- NOTE— An SC can be required to last for many years without interruption, since interrupting the MAC Service can cause client protocols to re-initialize and recalculate aggregations, spanning trees, and routes (for example). An SC lasts through a succession of SAs, each using a new SAK, to defend against a successful attack on a key while it is still in use. In contrast it is desirable to use a new SAK at periodic intervals to defend against a successful attack on a key while it is still in use. In addition, the MACsec protocol (Clause 8 and Clause 9) only allows 2^{32}–1 frames to be protected with a single key unless a Cipher Suite that supports extended packet numbering is used. Since 2^{32} minimum-sized IEEE 802.3 frames can be sent in approximately 5 min at 10 Gb/s, this can force the use of a new SA.

Summary, thoughts, questions…

Security & Privacy Technologies

CableLabs

- Authentication happens first with EAPOL
  - PAE peers
  - Multiple EAP protocols – TLS is only one
  - May use an authentication server
  - May result in CAK related parameters being distributed to the client (how is this protected?) if PSK is not being used
  - I don't know how to securely execute PSK CAK
- MKA executes after EAP authentication
  - Continues to use EAPOL as transport (MKPDU – how does this map to MPCP or OAMPDUs?)
  - Key server is negotiated (should we have that normatively be the OLT? How many keys will the OLT be generating? Only a few a second.)
  - Key server generates an SAK and KEK from the CAK for packet encryption and distributes the SAK
  - Key server also advertises the cipher suite (GCM-AES-128 is default for MACSec)
- Note for further study: MACSec does support point to multi-point
- MKA Keepalive/renew SAK?
  - Per 802.1ae, each Secure Channel is supported by an overlapped sequence of Security Associations and tach SA uses a fresh Secure Association Key. See note below.
  - Is this only semantics? Is the SAK message encrypted using the KEK or in the previous SAK encrypted messages? Are MKPDUs encrypted by the SAK?
- Concerned about high speed SAK rollover – renegotiating keys every few minutes seems bad
  - Depending on implementation, 100-500k SAKs per year. A single OLT may generate 40-180M SAKs per year (12 PONs, 32 ONUs each)
  - How are the keys encrypted?
  - KEK is derived from the CAK. How often is the CAK renewed? Does the KEK derivation change per use? Mark advises there may be a nonce which will need coordination. Maybe the risk is acceptable…
- Note for consideration: If we only support high-speed line rate protocols, and they will be rekeying every few seconds or minutes, a key expiration timer seems unnecessary