IEEE P802.1AE YANG Instance Document

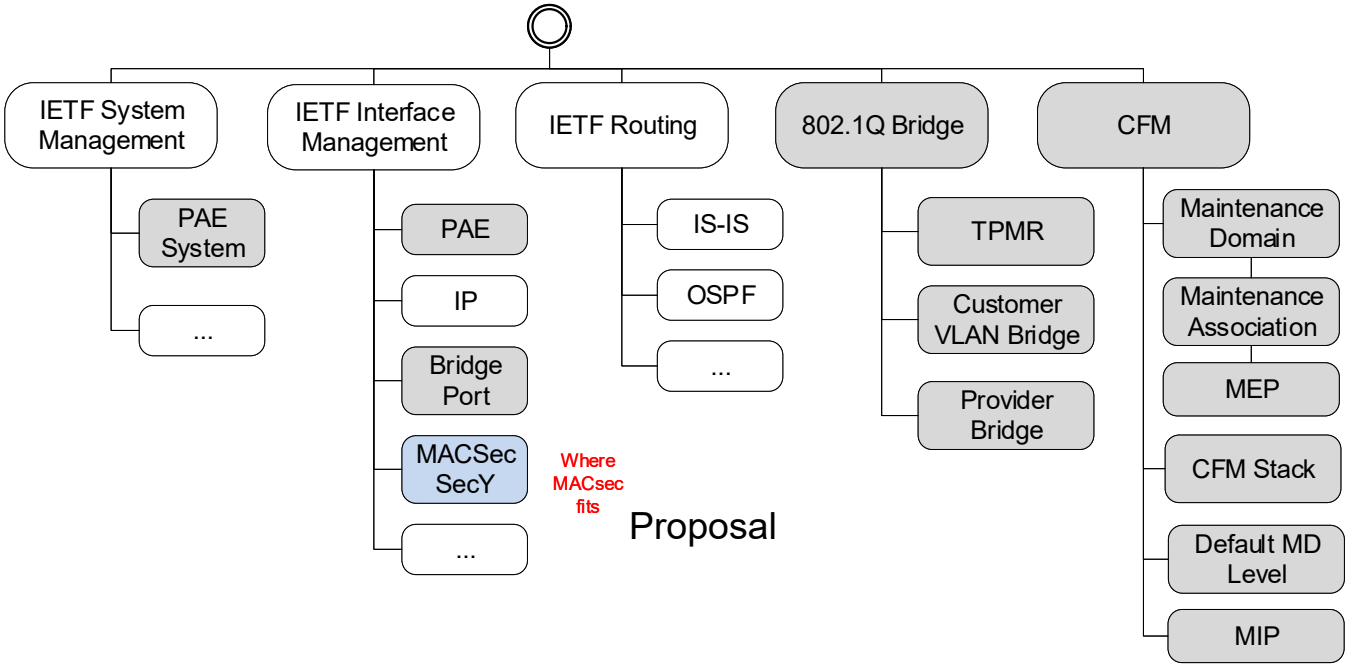
IEEE 802.1 Meeting

Don Fedyk
Version 0
July 3, 2019

Introduction

- 802.1dk is a proposed project to update 802.1AE-2018 with Yang and privacy options for MACsec
- 802.1dk is not yet approved.
- Part of the work that needs to be done for 802.1dk is a YANG model for the existing 802.1AE
- This slide deck is a discussion of how a YANG model for 802.1AE could fit with 802.1X
- For reference
  - dk-fedyk-ieee802-dot1ae-yang-0719-v00
  - dk-fedyk-ieee802-dot1ae-types-yang-0719-v00
  - dk-fedyk-ieee802-dot1ae-tree-0719-v00

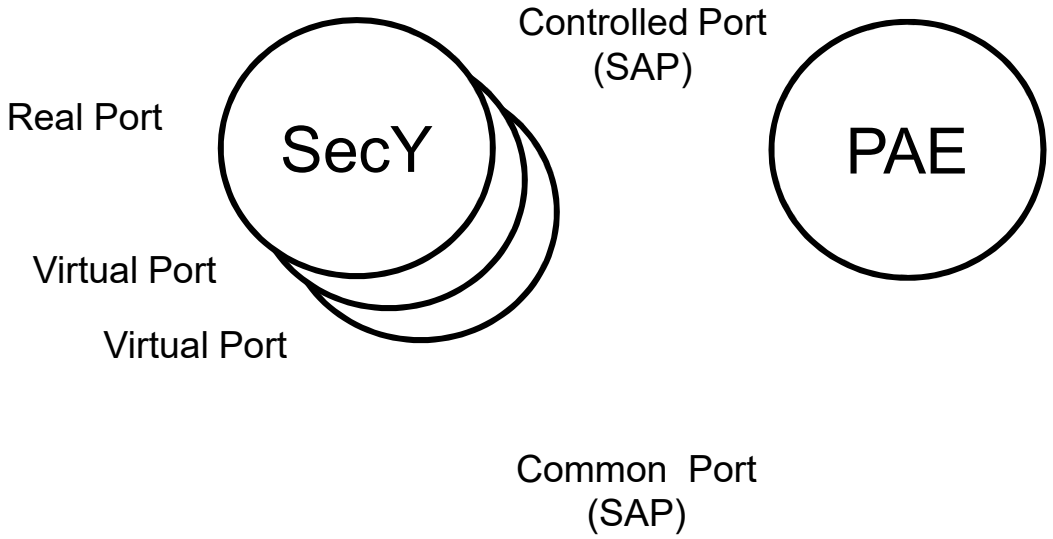
802.1 YANG Structure and Relationships



MAC SEC Information Model 802.1AE

- End stations (11.2)
- MAC Bridges (11.3)
- VLAN-aware Bridges (11.4)
- Systems that incorporate Link Aggregation (11.5)
- Systems that incorporate Link Layer Discovery Protocol (LLDP, 11.6)
- Provider Bridges and VLAN-aware Bridges attached to Provider Bridged Networks (11.7)
- LANs that provide independently secured access for multiple end stations (11.8).

Relationship to 802.1X



IETF Interface Stats

```
module: ietf-interfaces
  +--rw interfaces
  |  +--rw interface* [name]
  |     +--rw name                          string
  |     +--rw description?                  string
  |     +--rw type                          identityref
  |     +--rw enabled?                      boolean
  |     +--rw link-up-down-trap-enable?     enumeration {if-mib}?
  |     +--ro admin-status                  enumeration {if-mib}?
  |     +--ro oper-status                   enumeration
  |     +--ro last-change?                  yang:date-and-time
  |     +--ro if-index                      int32 {if-mib}?
  |     +--ro phys-address?                 yang:phys-address
  |     +--ro higher-layer-if*              interface-ref
  |     +--ro lower-layer-if*               interface-ref
  |     +--ro speed?                        yang:gauge64
  |     +--ro statistics
  |        +--ro discontinuity-time    yang:date-and-time
  |        +--ro in-octets?            yang:counter64
  |        +--ro in-unicast-pkts?      yang:counter64
  |        +--ro in-broadcast-pkts?    yang:counter64
  |        +--ro in-multicast-pkts?    yang:counter64
  |        +--ro in-discards?          yang:counter32
  |        +--ro in-errors?            yang:counter32
  |        +--ro in-unknown-protos?    yang:counter32
  |        +--ro out-octets?           yang:counter64
  |        +--ro out-unicast-pkts?     yang:counter64
  |        +--ro out-broadcast-pkts?   yang:counter64
  |        +--ro out-multicast-pkts?   yang:counter64
  |        +--ro out-discards?         yang:counter32
  |        +--ro out-errors?           yang:counter32
```

```
x--ro interfaces-state
     x--ro interface* [name]
        x--ro name                string
        x--ro type                identityref
        x--ro admin-status        enumeration {if-mib}?
        x--ro oper-status         enumeration
        x--ro last-change?        yang:date-and-time
        x--ro if-index            int32 {if-mib}?
        x--ro phys-address?       yang:phys-address
        x--ro higher-layer-if*    interface-state-ref
        x--ro lower-layer-if*     interface-state-ref
        x--ro speed?              yang:gauge64
        x--ro statistics
           x--ro discontinuity-time    yang:date-and-time
           x--ro in-octets?            yang:counter64
           x--ro in-unicast-pkts?      yang:counter64
           x--ro in-broadcast-pkts?    yang:counter64
           x--ro in-multicast-pkts?    yang:counter64
           x--ro in-discards?          yang:counter32
           x--ro in-errors?            yang:counter32
           x--ro in-unknown-protos?    yang:counter32
           x--ro out-octets?           yang:counter64
           x--ro out-unicast-pkts?     yang:counter64
           x--ro out-broadcast-pkts?   yang:counter64
           x--ro out-multicast-pkts?   yang:counter64
           x--ro out-discards?         yang:counter32
           x--ro out-errors?           yang:counter32
```

The Base Model for interfaces or ports

802.1X Yang Augment of Interfaces (Snipits)

```
augment /if:interfaces/if:interface:
   +--rw pae
      +--rw pae-system?                -> /sys:system/dot1x:pae-system/name
      +--rw vp-enable?                 boolean
      +--rw port-capabilities
      |  +--rw supp?                  boolean
      |  +--rw auth?                  boolean
      |  +--rw mka?                   boolean
      |  +--rw macsec?                boolean
      |  +--rw announcements?         boolean
      |  +--rw listener?              boolean
      |  +--rw virtual-ports?         boolean
      |  +--rw in-service-upgrades?   boolean
      +--ro port-name?                 if:interface-ref
      +--ro port-number?               dot1x-types:pae-if-index
      +--ro controlled-port-name?      if:interface-ref
      +--ro controlled-port-number?    dot1x-types:pae-if-index
      +--ro uncontrolled-port-name?    if:interface-ref
      +--ro uncontrolled-port-number?  dot1x-types:pae-if-index
      +--ro common-port-name?          if:interface-ref
      +--ro common-port-number?        dot1x-types:pae-if-index
      +--rw port-type?                 enumeration
      +--ro virtual-port
      |  +--ro max?            uint32
      |  +--ro current?        yang:gauge32
      |  +--ro start?          boolean
      |  +--ro peer-address?   ieee:mac-address
      +--rw supplicant
      |  +--rw held-period?     uint16
      |  +--rw retry-max?       uint32
      |  +--ro enabled?         boolean
      |  +--ro authenticate?    boolean
      |  +--ro authenticated?   boolean
      |  +--ro failed?          boolean
+--ro terminate-cause?   enumeration
```

PAE augments Interfaces and has references to controlled uncontrolled and common ports.

Port is either real or virtual implying the whole interface is real or virtual.

802.1X Yang Augment of Interfaces (Snipits)

```
+--rw supplicant
+--rw authenticator
+--rw kay
+--rw logon-nid
+--rw announcer
+--rw listener
```

```
+--ro eapol-statistics
      |  +--ro invalid-eapol-frame-rx?       yang:counter32
      |  +--ro eap-length-error-frames-rx?   yang:counter32
      |  +--ro eapol-announcements-rx?       yang:counter32
      |  +--ro eapol-announce-reqs-rx?       yang:counter32
      |  +--ro eapol-port-unavailable?       yang:counter32
      |  +--ro eapol-start-frames-rx?        yang:counter32
      |  +--ro eapol-eap-frames-rx?          yang:counter32
      |  +--ro eapol-logoff-frames-rx?       yang:counter32
      |  +--ro eapol-mk-no-cfn?              yang:counter32
      |  +--ro eapol-mk-invalid-frames-rx?   yang:counter32
      |  +--ro last-eapol-frame-source?      ieee:mac-address
      |  +--ro last-eapol-frame-version?     uint8
      |  +--ro eapol-supp-eap-frames-tx?     yang:counter32
      |  +--ro eapol-logoff-frames-tx?       yang:counter32
      |  +--ro eapol-announcements-tx?       yang:counter32
      |  +--ro eapol-announce-reqs-tx?       yang:counter32
      |  +--ro eapol-start-frames-tx?        yang:counter32
      |  +--ro eapol-auth-eap-frames-tx?     yang:counter32
      |  +--ro eapol-mka-frames-tx?          yang:counter32
      +--rw logon-process
         +--rw logon?                 boolean
         +--ro connect?               enumeration
         +--ro port-valid?            boolean
         +--ro session-statistics* [session-id]
            +--ro session-id          dot1x-types:pae-session-id
            +--ro user-name?          dot1x-types:pae-session-user-name
            +--ro octets-rx?          yang:counter64
            +--ro octets-tx?          yang:counter64
            +--ro frames-rx?          yang:counter64
            +--ro frames-tx?          yang:counter64
            +--ro time?               uint32
```

MACsec Yang Augment of Interfaces Stats Under SECY by controlled port

```
module: ieee802-dot1ae
  augment /if:interfaces/if:interface:
    +--rw secy
       +--rw secy* [controlled-port-number]
       |  +--rw controlled-port-number    dot1x-types:pae-if-index
       |  +--rw verification
       |  |  +--ro max-receive-channels?   uint8
       |  |  +--ro max-receive-keys?       uint8
       |  |  +--rw validate-frames?        enumeration
       |  |  +--rw replay-protect?         boolean
       |  |  +--rw replay-window?          uint32
       |  |  +--ro in-pkts-untagged?       yang:counter64
       |  |  +--ro in-pkts-no-tag?         yang:counter64
       |  |  +--ro in-pkts-bad-tag?        yang:counter64
       |  |  +--ro in-pkts-no-sa?          yang:counter64
       |  |  +--ro in-pkts-no-sa-error?    yang:counter64
       |  |  +--ro in-pkts-overrun?        yang:counter64
       |  |  +--ro in-octets-validated?    yang:counter64
       |  |  +--ro in-octets-decrypted?    yang:counter64
       |  |  +--ro receive-sc* [sci]
       |  |     +--ro sci                  dotlaetypes:sec-sci-type
       |  |     +--ro created-time?        yang:date-and-time
       |  |     +--ro started-time?        yang:date-and-time
       |  |     +--ro stopped-time?        yang:date-and-time
       |  |     +--ro receiving?           boolean
       |  |     +--ro in-pkts-ok?          yang:counter64
       |  |     +--ro in-pkts-unchecked?   yang:counter64
       |  |     +--ro in-pkts-delayed?     yang:counter64
       |  |     +--ro in-pkts-late?        yang:counter64
       |  |     +--ro in-pkts-invalid?     yang:counter64
       |  |     +--ro in-pkts-not-valid?   yang:counter64
       |  |     +--ro receive-sa* [rxa]
       |  |        +--ro in-use?           boolean
       |  |        +--ro ssci?             uint32
       |  |        +--ro next-pn?          dotlaetypes:sec-pn-type
       |  |        +--ro created-time?     yang:date-and-time
       |  |        +--ro started-time?     yang:date-and-time
       |  |        +--ro stopped-time?     yang:date-and-time
       |  |        +--ro rxa               dotlaetypes:sec-an-type
       |  |        +--ro lowest-pn?        dotlaetypes:sec-pn-type
       |  |        +--ro enable-receive?   boolean
       |  |        +--ro updt-next-pn?     dotlaetypes:sec-pn-type
       |  |        +--ro updt-lowest-pn?   dotlaetypes:sec-pn-type
       |  |        +--ro key-identifier?   dotlaetypes:sec-key-identifier-type
       |  +--rw generation
       |  |  +--ro sci-base?               string
       |  |  +--rw max-transmit-channels?  uint16
       |  |  +--rw max-transmit-keys?      uint16
       |  |  +--rw protect-frames?         boolean
       |  |  +--rw always-include-sci?     boolean
       |  |  +--rw use-es?                 boolean
       |  |  +--rw use-scb?                boolean
       |  |  +--ro including-sci?          boolean
       |  |  +--ro out-pkts-untagged?      yang:counter64
       |  |  +--ro out-pkts-too-long?      yang:counter64
       |  |  +--ro out-octets-protected?   yang:counter64
       |  |  +--ro out-octets-encrypted?   yang:counter64
       |  |  +--rw user-priority-0-7
       |  |  |  +--rw traffic-class?   uint8
I      |  |  |  +--rw traffic-class?   uint8
       |  |  +--rw user-pcp-ap* [user-pcp]
       |  |  |  +--rw user-pcp           uint8
       |  |  |  +--rw access-priority?   uint8
       |  |  +--ro transmit-sc* [sci]
       |  |     +--ro sci                  dotlaetypes:sec-sci-type
       |  |     +--ro created-time?        yang:date-and-time
       |  |     +--ro started-time?        yang:date-and-time
       |  |     +--ro stopped-time?        yang:date-and-time
       |  |     +--ro transmitting?        boolean
       |  |     +--ro encoding-sa?         dotlaetypes:sec-an-type
       |  |     +--ro out-pkts-protected?  yang:counter64
       |  |     +--ro out-pkts-encrypted?  yang:counter64
       |  |     +--ro transmit-sa* [txa]
       |  |        +--ro in-use?           boolean
       |  |        +--ro ssci?             uint32
       |  |        +--ro next-pn?          dotlaetypes:sec-pn-type
       |  |        +--ro created-time?     yang:date-and-time
       |  |        +--ro started-time?     yang:date-and-time
       |  |        +--ro stopped-time?     yang:date-and-time
       |  |        +--ro txa               dotlaetypes:sec-an-type
       |  |        +--ro confidentiality?  boolean
       |  |        +--ro key-identifier?   dotlaetypes:sec-key-identifier-type
```

As shown there are multiple SecYs per interface but could be 1 per real or virtual interface too.

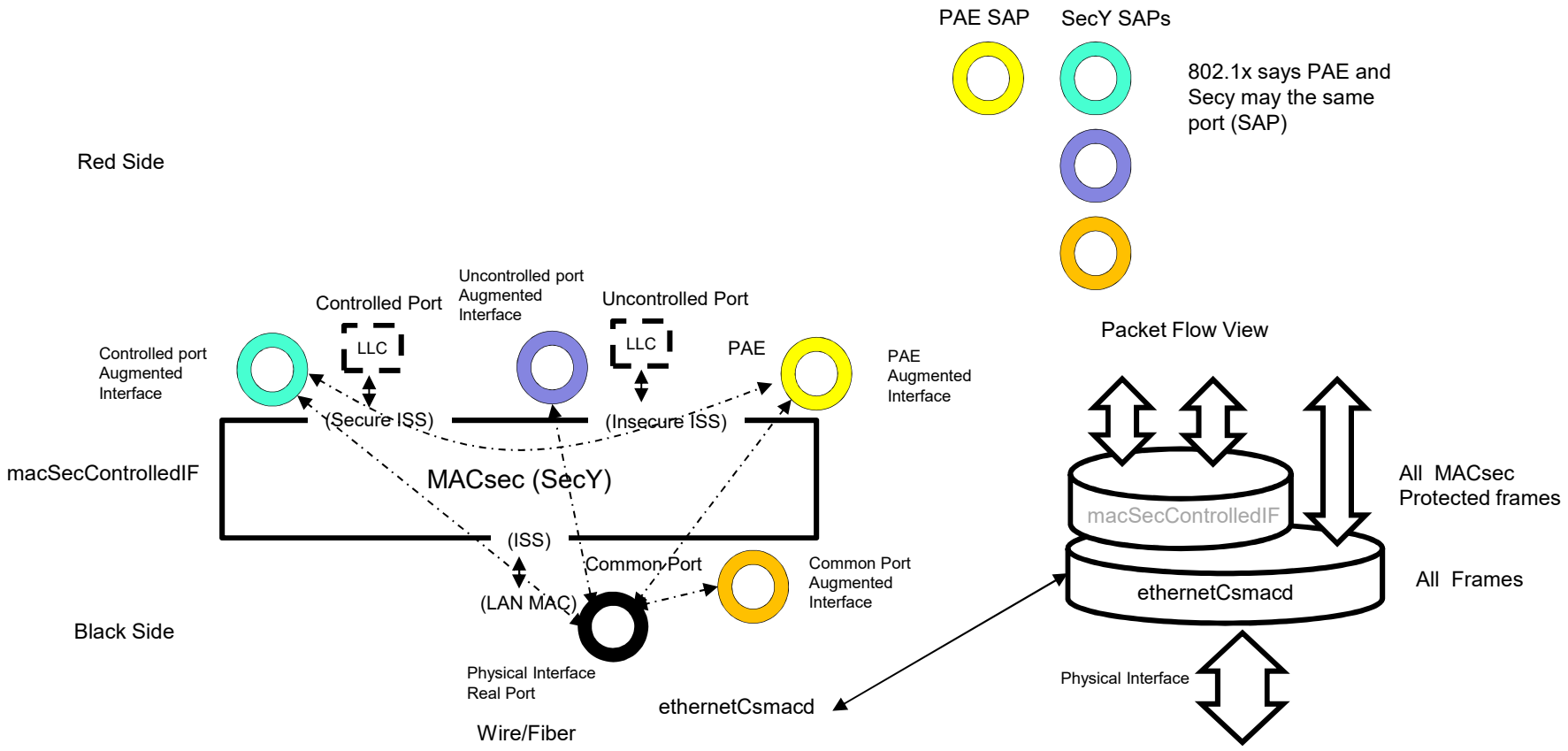
See dk-fedyk-ieee802-dot1ae-tree-0719-v00

MACsec Yang Augment of Interfaces Stats Under SECY by controlled port

```
|  +--rw current-cipher-suite
       |  |  +--rw cipher-suite-identifier?   dot1aetypes:sec-eui64-type
       |  |  +--rw data-key* [keys]
       |  |     +--rw keys              uint32
       |  |     +--ro key-identifier?   dot1aetypes:sec-key-identifier-type
       |  |     +--ro transmits?        boolean
       |  |     +--ro receives?         boolean
       |  +--rw controlled-interface
       |  |  +--ro provided-interface?         dot1x-types:pae-if-index
       |  |  +--ro mac-enabled?                boolean
       |  |  +--ro mac-operational?            boolean
       |  |  +--ro oper-point-to-point-mac?    boolean
       |  |  +--rw admin-point-to-point-mac?   enumeration
       |  |  +--ro controlled-port-enabled?    boolean
       |  +--rw uncontrolled-interface
       |  |  +--ro provided-interface?         dot1x-types:pae-if-index
       |  |  +--ro mac-enabled?                boolean
       |  |  +--ro mac-operational?            boolean
       |  |  +--ro oper-point-to-point-mac?    boolean
       |  |  +--rw admin-point-to-point-mac?   enumeration
       |  +--rw common-port
       |  |  +--ro common-port?   dot1x-types:pae-if-index
       |  +--rw cipher-suite-control* [implemented-cipher-suite]
       |     +--rw implemented-cipher-suite    dot1aetypes:sec-eui64-type
       |     +--rw enable-use?                 boolean
       |     +--rw require-confidentiality?    boolean
       +--rw cipher-suites* [cipher-suite]
          +--rw cipher-suite                  dot1aetypes:sec-eui64-type
          +--ro name?                         string
          +--ro integrity-protection?         boolean
          +--ro confidentiality-protection?   boolean
          +--ro offset-confidentiality?       boolean
          +--ro changes-data-length?          boolean
          +--ro icv-length?                   uint16
```

See dk-fedyk-ieee802-dot1ae-tree-0719-v00

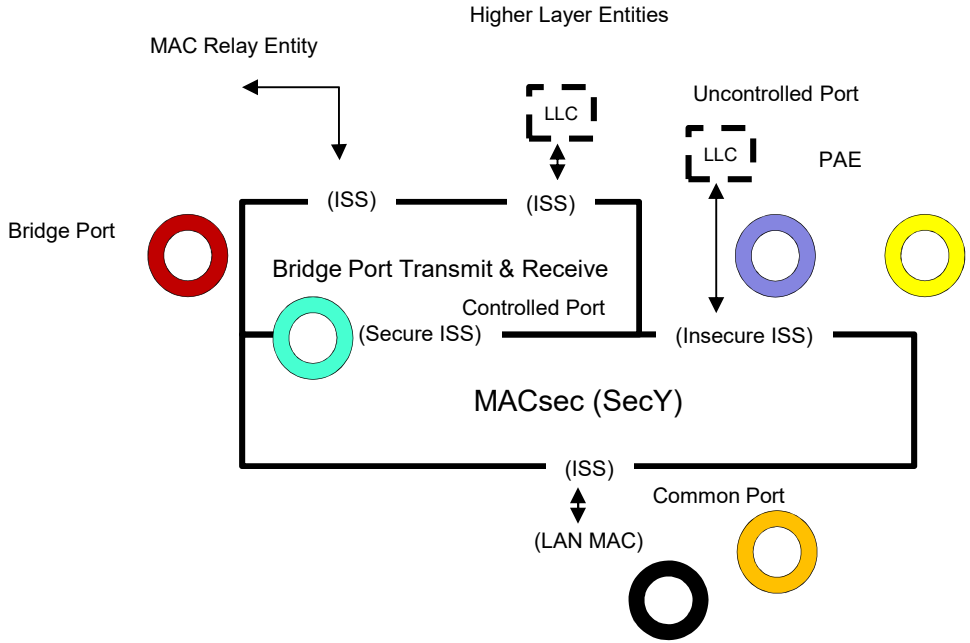
(Not so) Simple End Station Interface
All the little Pieces for PAE and SECY



Notes

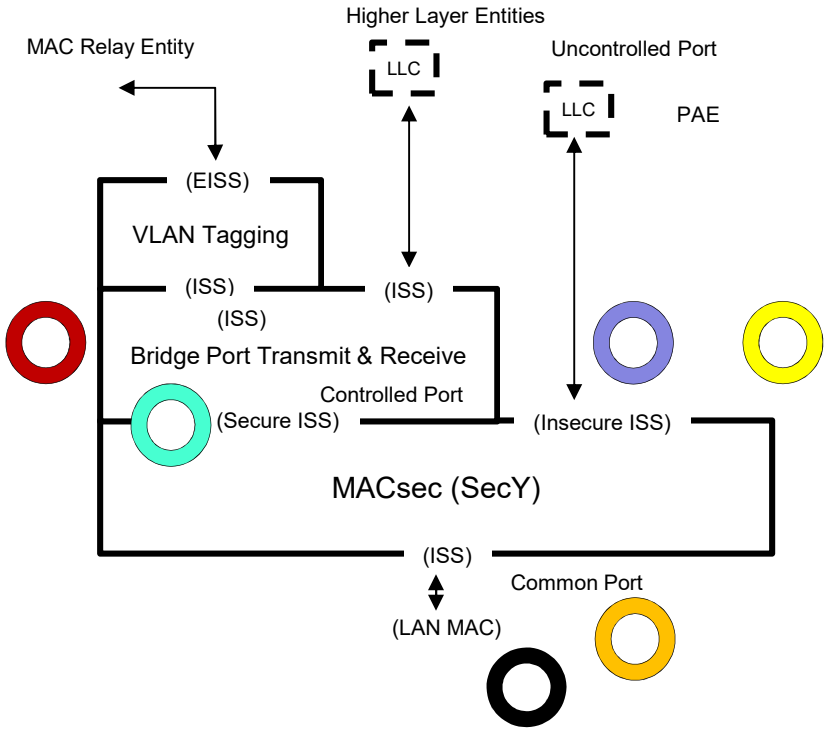
- Base Interfaces - ietf-interfaces are augmented by Ports (SAPs) PAE is a SAP, Controlled Interface is a SAP etc.
- (Division between interfaces and SAPs is blurry)
- Multiple SAPs are off the same Base Interface
- PAE and SecY share the same Base Interface but have different attributes.

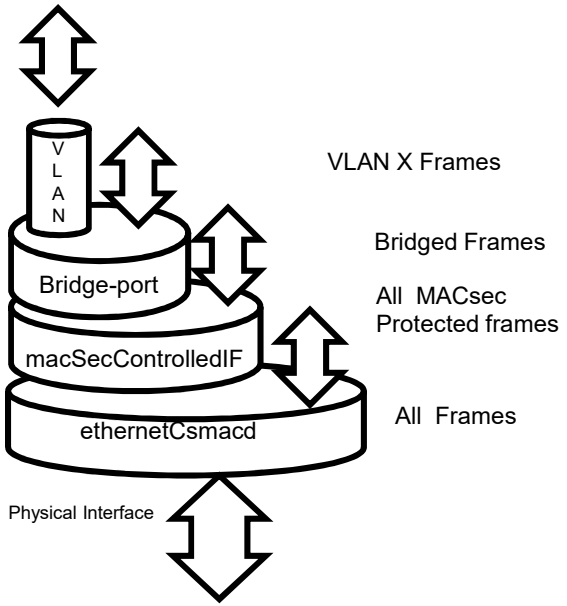
VLAN-unaware MAC Bridge Port with MACsec



Basically relationships do not change as we add Bridges VLANs etc.

VLAN-unaware MAC Bridge Port with MACsec

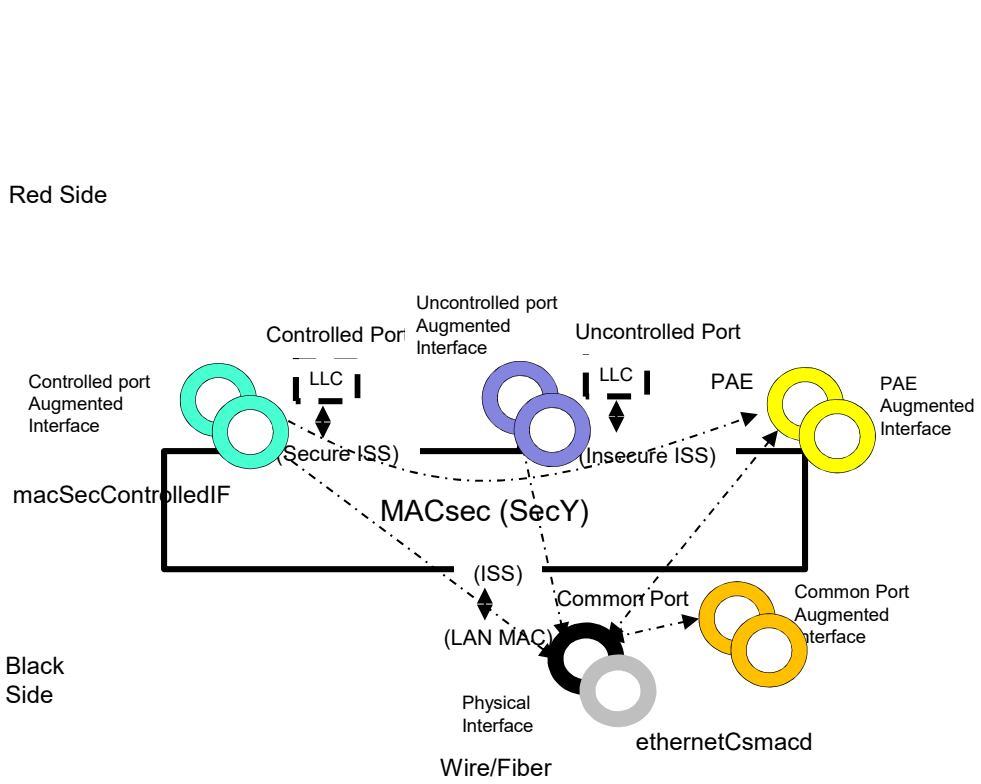


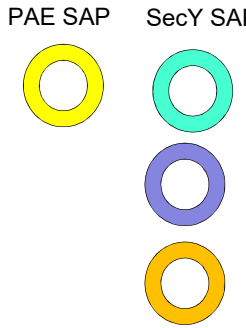


Virtual Ports

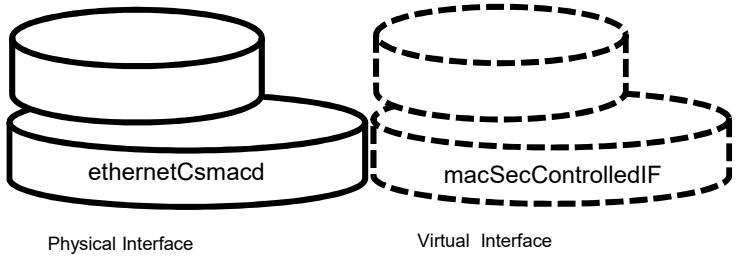
- One PAE supports a number of Virtual ports, but
  - As far as I can tell a Virtual port is a complete new virtual interface?
  - A PAE under an Interface is either a real port or a virtual port.
- How is a Single PAE with 1 real port and say 1 virtual port configured? (Two Interfaces with same PAE identifier?)
- A SecY per Virtual port.

Virtual Ports by Creating Virtual Interfaces



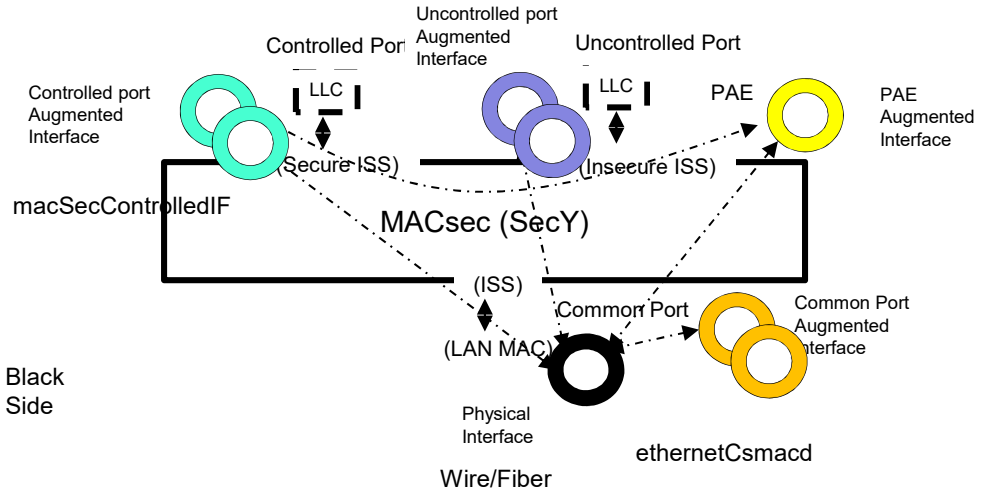


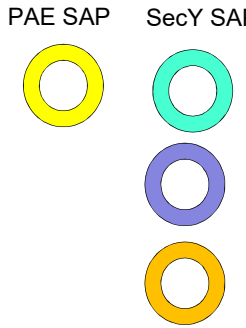
802.1x says PAE and Secy may the same port (SAP)



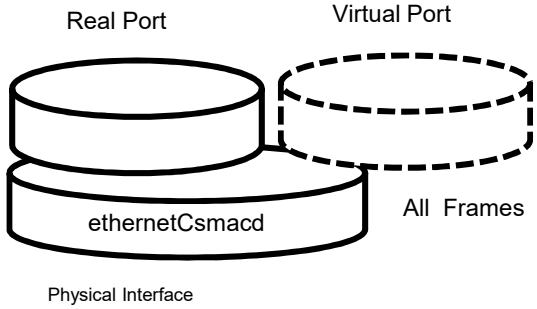
Seems to fit with current 802.1X Model

Virtual Ports by Multiple SecY





802.1x says PAE and Secy may the same port (SAP)



Slightly Different 802.1X Model No
Virtual Interface One PAE instance

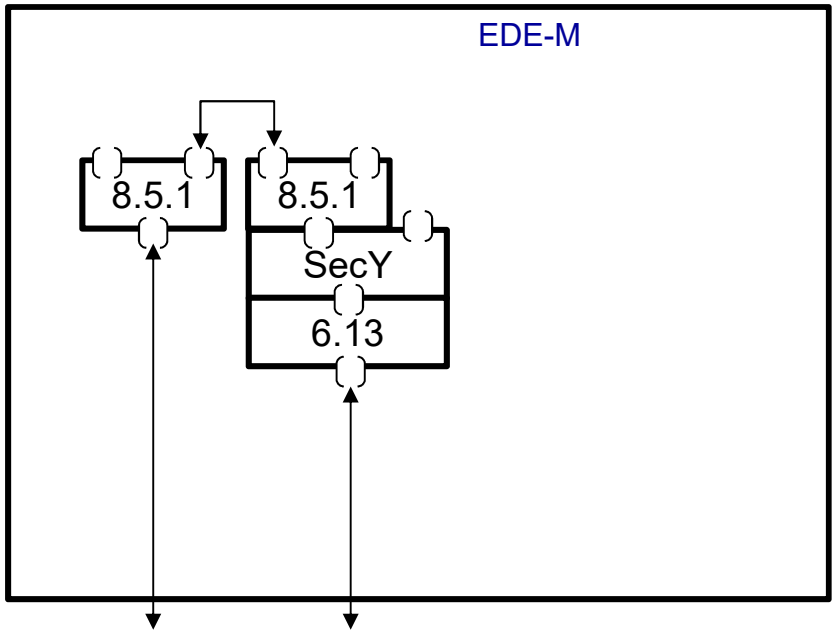
Questions

- Which way to go? Virtual Interfaces or Multiple SecY with real and Virtual Ports.
- It seems that 802.1X leans towards the Virtual Port is based on a Virtual Interface model but there is no mention of Virtual Interfaces just Virtual Ports. The PAE specifies the virtual or real interface characteristics.
- Is a macSecControlledIF a virtual Interface?
- It is not clear how or when the Virtual Interfaces are created.
- The other way to accomplish this is a Multiple SecY under real interfaces. In this case the Single PAE controls all SecYs under an Interface.
- It may be the two models are totally equivalent.
- See dk-fedyk-ieee802-dot1ae-yang-0719-v00

Ethernet Encryption Device (EDE)

- EDEs are part of 802.1AE
- The YANG model applies to EDEs as well.
- The following is for discussion of what is needed to configure EDEs.

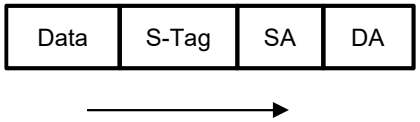
Ethernet Encryption Device
EDE-M

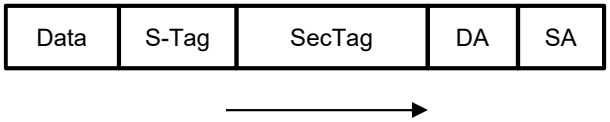


EDE-M needs no VLAN Config

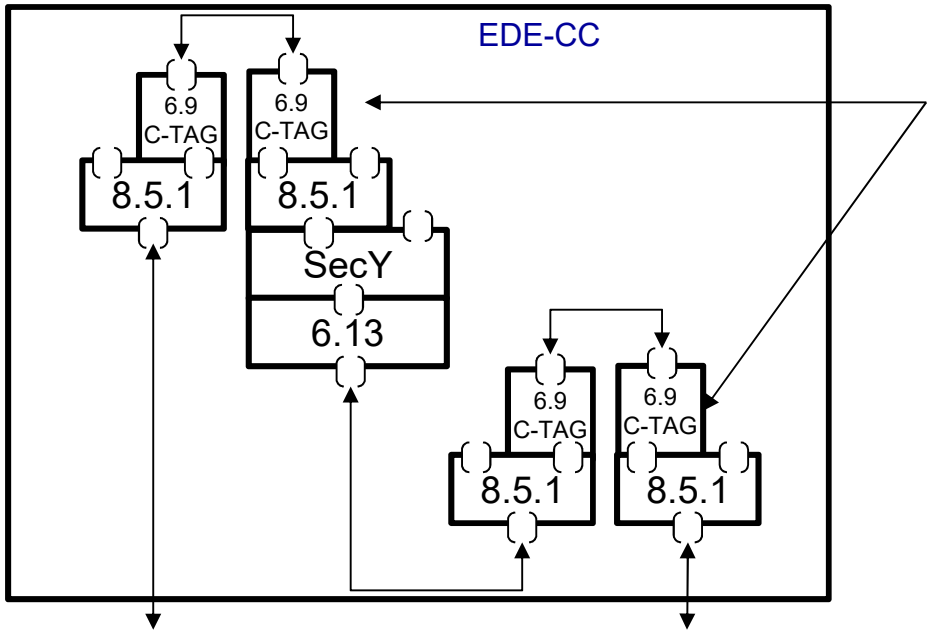
May use PVID

Current SecY Model is sufficient.





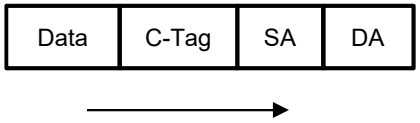
Ethernet Encryption Device
EDE-CC



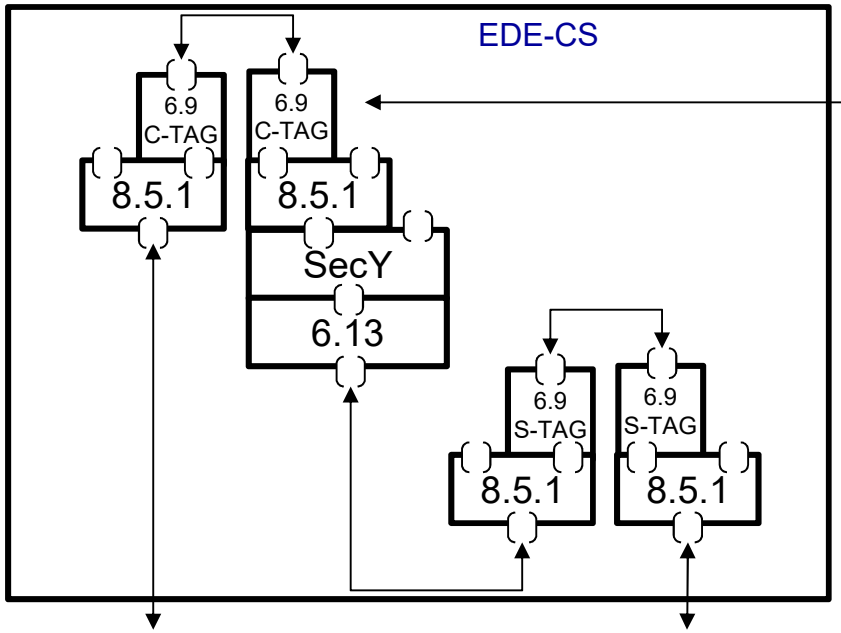
EDE –CC uses C-Tag Config from Upper Bridge

Current SecY Model is sufficient.

Inner and Outer C-Tag are identical



Ethernet Encryption Device
EDE-CS



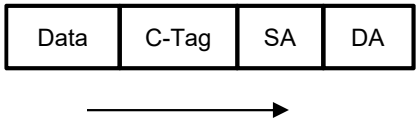
C-Tag Config

Where is S-TAG Management Interface defined?

Needs to be a Port Map?

Current SecY Model is not sufficient.

S-TAG Could come from from port VID



EDEs and YANG Summary

- EDE-M
  - VLAN Unaware same YANG
- EDE-CC
  - C-VLAN comes from Bridge and outer C-VID is from inner C-VID
- EDE-CS
  - C-VLAN comes From Bridge and outer S-VID comes from another Bridge but there is a port mapping function.