IEEE 802.1 Security MACsec Privacy Rate Specification Deep Dive

Don Fedyk – don.Fedyk@labn.net

Disclaimer

- This is a work in progress. The material here is for discussion purposes and may contain errors.

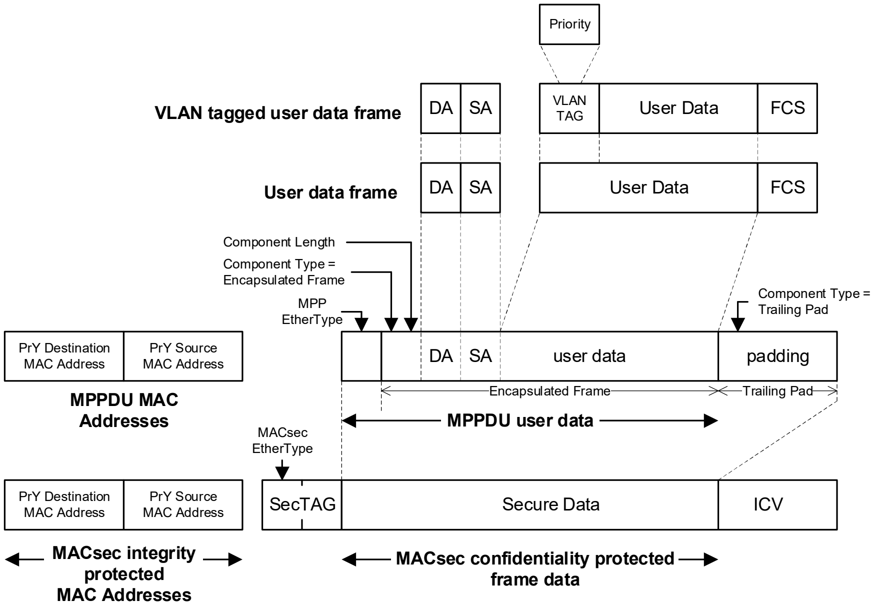
Configuration parameters for MAC Privacy 802.1AEdk

- How big should an MPPDU be?
  - Encapsulation Arithmetic
- Bandwidth, Interval, Delay, Burst
- Based on some ideas in Mick’s Updated Clause 20 text

Determining Frames Size and Rates for MAC Privacy Channels and Frames

- PrY Channel Frames:
  - Fragmentation or No fragmentation?
  - Fragmentation is default enabled. (No config option)
    - Allows Higher efficiency, allows late addition.
  - Setting an MPPDU too small can force fragmentation when Max size user frames are encountered.
  - Determine the maximum user frame size “The Layer 2 MTU”.
- PRY Frames
  - No Fragmentation
  - Determine the maximum user frame size “The Layer 2 MTU”.
  - This must be greater that or equal to the L2 MTU or larger frames could be dropped.

MAC Privacy – Which Length?



This is the source of the defining length

MAC Privacy

MAC Privacy & MACsec

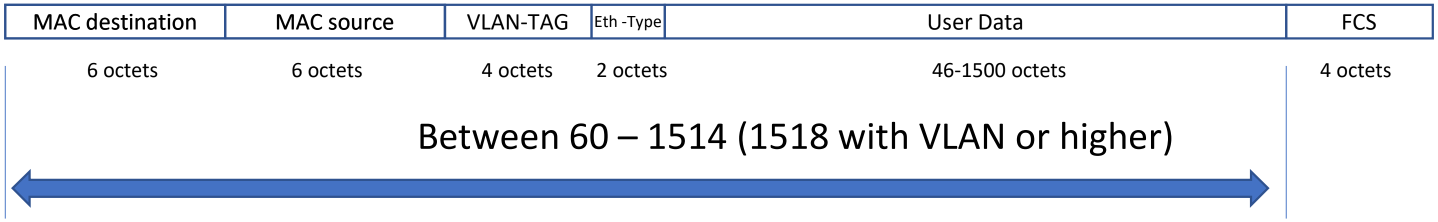
Clause 17 Draft figure

Standard Ethernet Frames

Standard Ethernet encapsulation:

- Frame sizes are dependent on media
- Ethernet Standard are 1500 octets of user data
- Ethernet Jumbo is 9000 octets of user data
- Uses the Media overhead bytes

Goal is to determine the MTU for the situation



MAC destination	MAC source	VLAN-TAG	Eth -Type	User Data	FCS
6 octets	6 octets	4 octets	2 octets	46-1500 octets	4 octets

Between 60 – 1514 (1518 with VLAN or higher)

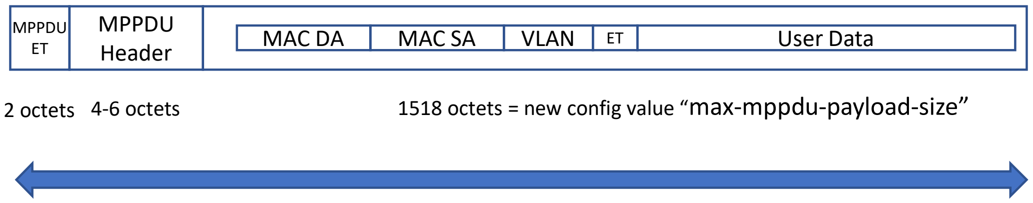
User data no FCS
a.k.a. L2 MTU

MAC Privacy allows for an unfragmented Max User data frame
This means encapsulating nominally up to 1518 octets but possibly higher.
Other formats (e.g. LLC, SNAP) are supported as well but are less than or equal to 1500 octets. IEEE 802.3 allows up to 2000 for envelope frames 802.3as-2006

MAC Privacy MPPDU

- MAC Privacy Encapsulation adds
- 6 octets for Fixed Full frames
- +2 octets for a fragment – but that can be absorbed since fragments are variable
- Therefore 6 octets plus user data frame 1518 + 6.

max-mppdu-payload-size



1524 octets (6 octets overhead)

This is not a frame. A PrY Frame in non collocated PrY adds 16 octets more.

Notes

- I argue that max-mppdu-payload-size is what a user should configure. This value is what an MPPDU should carry.
- Logically this is simply 1500 plus the Ethernet L2 overhead but since we can have various formats and various MTUs I think configuring the actual payload of the MPPDU makes sense. **It is a discussion point though.**
- However, there are other values which relate to the total bandwidth on the wire coming next.
- We need to keep configuration to a minimum – yet accurately represent bits if we are to follow the algorithms currently suggest by the text.

Summary

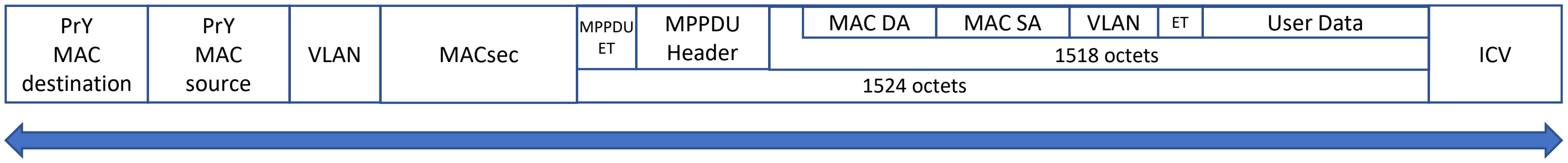
- Only need to configure the PrY L2 MTU e.g. 1518 or similar (e.g. 9018)
    - This number is whatever the base traffic user traffic format is for example PBB frames would use a larger number.
    - MPPDU length will be fixed at 1518 + 6 octets. (1524)
- As far as MAC privacy 1518 is the configurable encapsulation payload.
    - max-mppdu-payload-size, maxMppduPayloadSize
- If this number is configured smaller than source user traffic, some large frames may be fragmented – for channels.
- This number must be supported for privacy frames or the 1500 MTU of the user frame is impacted.
- If a smaller MTU is required for other reasons for PrY channels this number can be adjusted downwards this guideline is merely to prevent fragmentation of whole frames, but implementations may fragment anyway in the interest of reducing delay or increasing efficiency.

Part 2 Channel Frame Timing

- Whatever the Channel MPPDU length is, the MAC Pry Frames size determines the interval for a given data rate.
- We need to determine the size of the MAC PrY channel frame on the wire.

MAC Privacy Frame (using MACsec)

- MACsec has 8-16 octets of Sec TAG header
- Additional addresses (MAC Privacy) 12 octets
- Plus ICV 8-16 Octets



PrY MAC destination

PrY MAC source

VLAN

MACsec

MPPDU ET

MPPDU Header

MAC DA

MAC SA

VLAN

ET

User Data

1518 octets

1524 octets

ICV

1552-1568 octets (16-32 octets MACsec encapsulation + 12 PrY MAC) (28 or 36 or 44)
4 octets VLAN likely
Maximum as high as ~1592 (all well within 2000 octet maximum)

How to compute a Frame size Base on Pry MTU

- Frame size is the PrY User data frame with no FCS
  - Ethernet with 1 VLAN Tag = 1518 (from previous slides)
  - For Rate computation the system adds:
    - 6 octets MAC PrY + 4 octets VLAN 44 octets MAC Sec+ Constant for a configuration
    - Media overhead (20 octets). Media dependent
    - 1588 MAC PrY

Translation to a Media Rate/Interval

- 802.3 Media example
  - 8 Octets of preamble
  - 12 Octets of Inter Frame Gap
- These values plus the frame size determine the frame rate.
- I.E. 10Bbit/s = 10,000,000,000/(20+1572))*8))

  = 785175 Frames per second or interval of 1274 nanoseconds.

  (original was 10,000,000,000/((20+1518)*8) = 812,743 (1230 ns))

  ~4.7 percent overhead.

Arithmetic for Interval

Description	Value	Octets	Bits	Bandwidth bps	Interval ns	Actual BW
		Tally				
L2 MTU User		1500	12000	100,000	127,360,000	100000
L2 MTU Size Includes Address and Ethertype		1514	12112	500,000	25,472,000	500000
Number of User VLAN tags	1	1518	12144	1,000,000	12,736,000	1000000
Other Header Overhead	0	1518	12144	10,000,000	1,273,600	10000000
MAC Pry MTU max-mppdu-payload-size		1518	12144	100,000,000	127,360	100000000
MAC PrY MPPDU Header (constant)	6	1524	12192	1,000,000,000	12,736	1000000000
MAC DA+SA (constant)	12	1536	12288	10,000,000,000	1,274	9996860283
MACsec VLAN	1	1540	12320	40,000,000,000	319	39924764890
MACsec Header (constant)	8	1548	12384	100,000,000,000	128	99500000000
SCI Additional (Configured by Environment)	8	1556	12448			
ICV Size (Configured by Cipher Suite)	16	1572	12576			
Media Preamble	8	1580	12640			
Media Interframe Gap	12	1592	12736			
Total Bits per Frame on the wire			12736			

User Configures:

- Payload for MAC Pry to enable full frame.
- Interval for packet transmission

(Bandwidth is equivalent to interval but interval is friendly to TSN configuration.)

Note that total bits per frame maps to interval but a User does necessarily know this immediately . This is one reason bandwidth was in the original proposal. Interval can be computed but it is not solely determined by MAC PrY parameters alone.

Notes: For User Configuration

- Payload for MAC PrY
- And either:
  - An interval (which requires we know exact frame on the wire sizes). Or,
    - Interval is friendly to TSN intervals – need to map to priority and set equivalent number for TSN. It maps to a timescale of nanoseconds, but equipment may support less granular intervals.
    - If the interval is too small –exceeds the link rate ---the system should just use the smallest interval. But in many cases, we will configure intervals that are less than the maximum line rate.
  - A bandwidth that allows us to compute the size and interval to suite.
    - An interval is calculated and rounded to the nearest nanosecond or whatever timescale is supported. A corresponding actual bandwidth is calculated from the interval and bits on the wire.
    - If the bandwidth is too large it – exceeds capability it can be reduced
- The Shaping algorithm
  - Next,

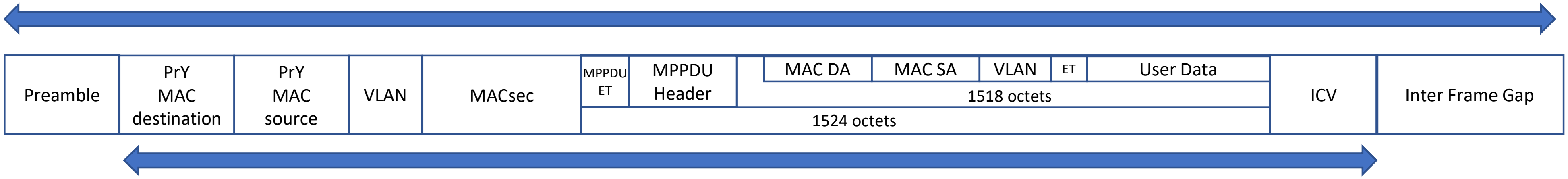
Burst Size

- We want a burst-size that reflects a frame sent on the wire.
- “The notional bucket contains a number of bits. When a Privacy Channel MPPDU is generated, 8 bits are subtracted from the bucket for each of the maxMppduPayloadSize plus overhead octets. Bits are added to the bucket at the ChannelInformationRate, expressed in bits per second, up to a maximum of ChannelBurstSize octets. “
- If the burst size was based on frames, we would not need to know the actual bits on the wire –it can be computed
- Current text suggests we know that 1518 octets MMPDU results in ~1592 octets on the wire (give or take 16 octets).
- Does the burst size need to be exact?
  - Trying to avoid configuring multiple packet sizes.
- Not that if multiple frames sizes are being encapsulated this is the largest Frame size bits on the wires time burst frames.

Burst Size

Burst Size Should be based on this if it is in bits

Bits on the Wire – 1592 octets



Preamble

PrY MAC destination

PrY MAC source

VLAN

MACsec

MPPDU ET

MPPDU Header

MAC DA

MAC SA

VLAN

ET

User Data

1518 octets

1524 octets

ICV

Inter Frame Gap

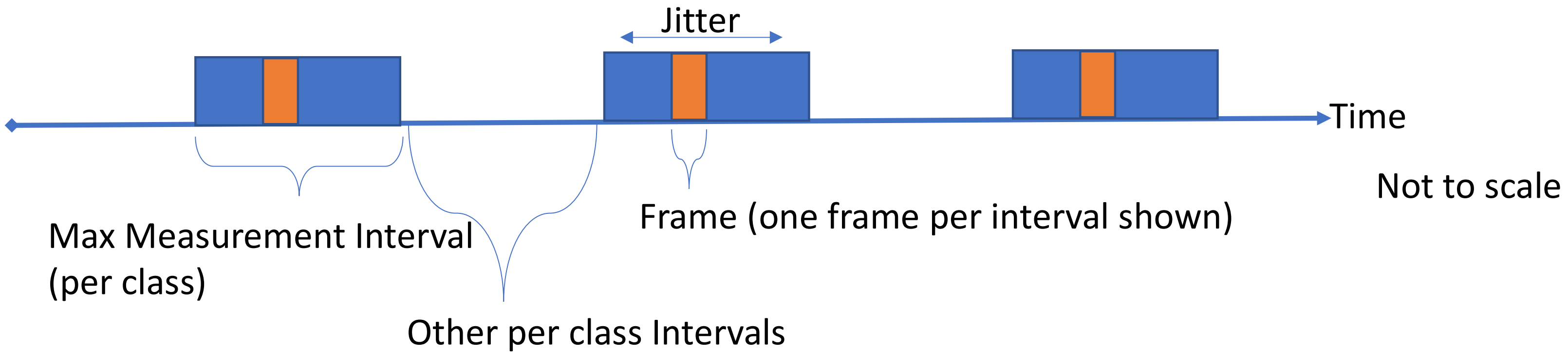
MACsec Size 1572 octets fixed includes padding

Burst Size could be in integral frames and then it does not matter?

Transmission Cases

- MAC PrY Channel is the whole link
  - Rate is line rate
  - Interval can be computed
  - Burst rate – Not really required ?
- MAC PrY Channel is a portion of the link
  - Interval is configured and rate can be computed.
  - Rate is some fraction of the line rate (interval could be computed)
  - Burst rate must be specified to average the interval/rate over a time period
- Burst size is limited by delay at lower speeds and memory buffers at high speeds.

Transmission Timing
(Time Sensitive Networking for example)



Jitter

Time

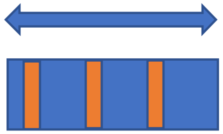
Not to scale

Frame (one frame per interval shown)

Max Measurement Interval
(per class)

Other per class Intervals

Controls / Per Class

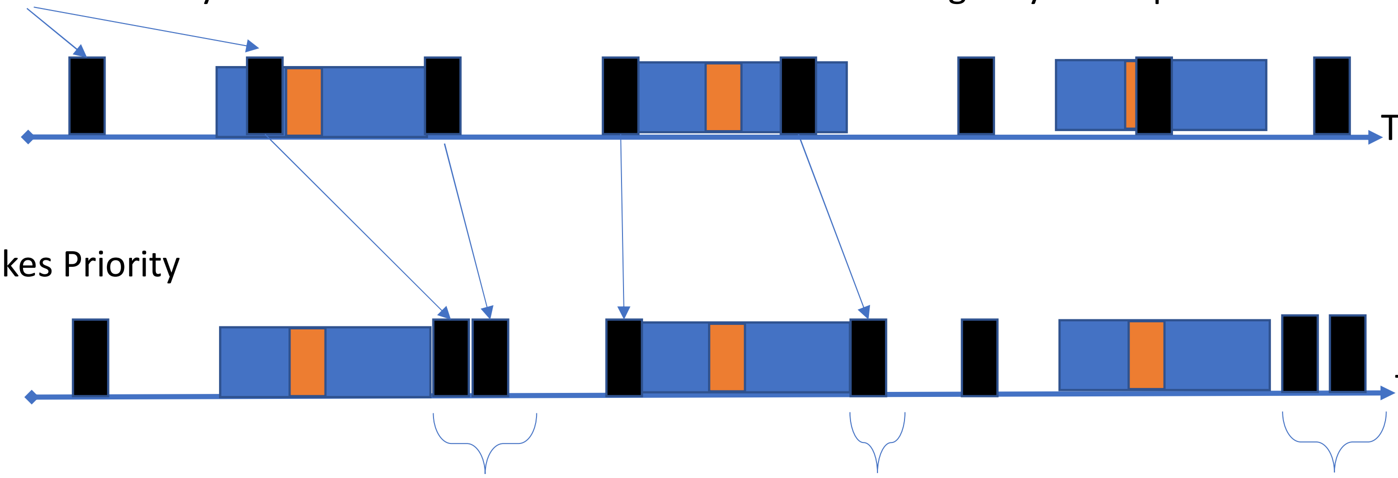


- Measurement interval 500us – 250us Typical
- Class Bandwidth - Percent of port bandwidth
(locked or shared with other classes)
- Integral number of Frames per interval

Note this is my interpretation

One Example Delays that cause Burst

MAC PrY Interval may be different than TSN intervals and scheduling may overlap:



Time

Not to scale

TSN takes Priority

Time

Time shifting creates a burst

Burst

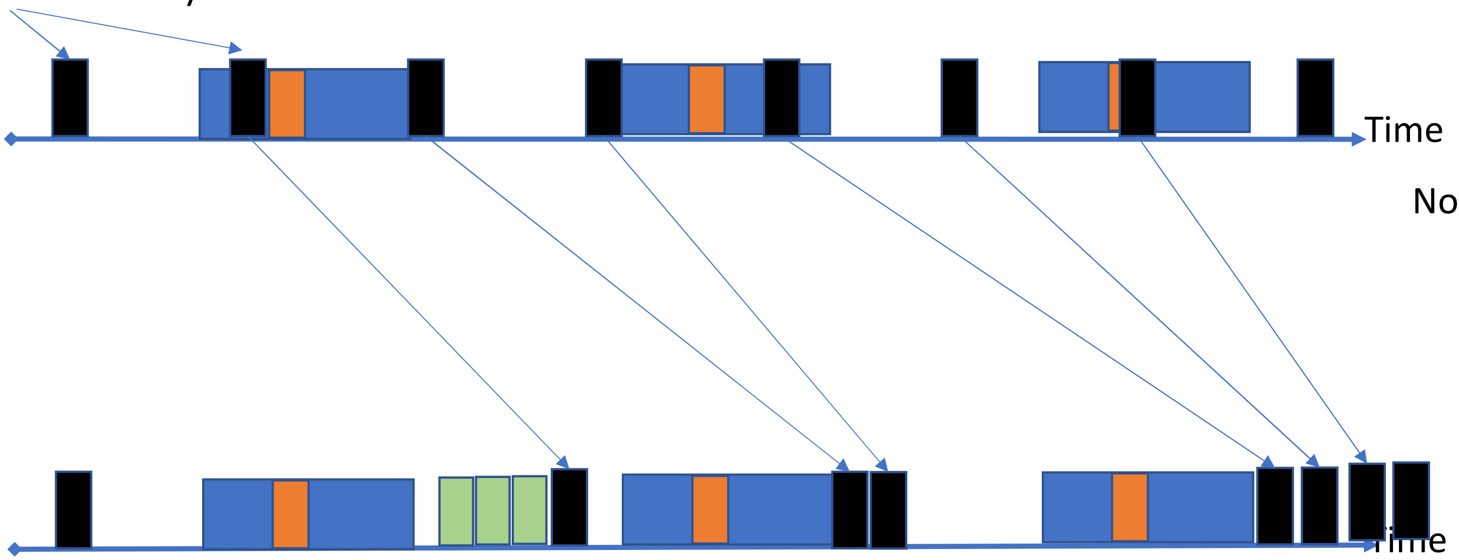
Burst

Burst

Average transmission rate is the same over a time span
need to configure a burst limit.

One Example Delays that cause Bursts

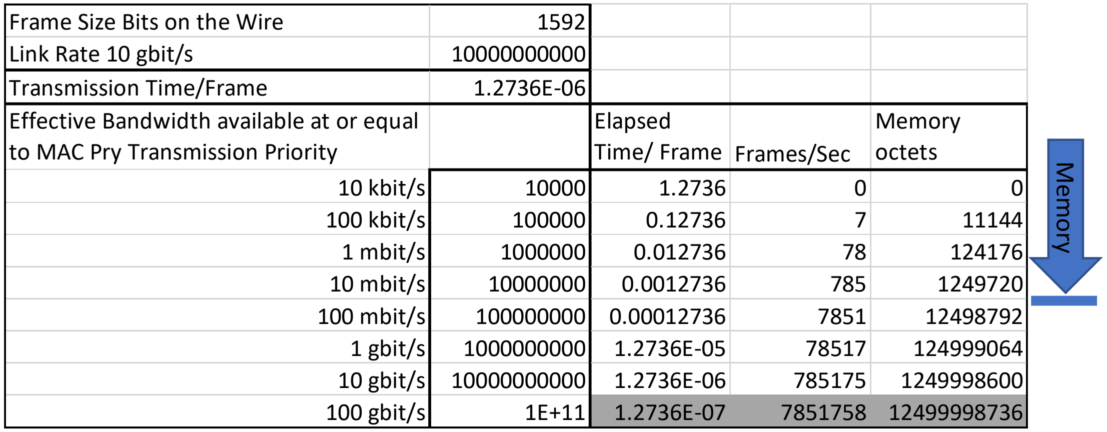
MAC PrY Interval may be different than TSN intervals



Other Traffic May cause the MAC Pry Frames to be pushed even further out. A maximum Burst size limits the number of frames that can be delayed from the original transmission time. Burst is based on effective bandwidth to the MAC PrY class.

Arithmetic for Channel Burst

Frames or Octets?



Frame Size Bits on the Wire	1592			
Link Rate 10 gbit/s	10000000000			
Transmission Time/Frame	1.2736E-06			
Effective Bandwidth available at or equal to MAC Pry Transmission Priority		Elapsed Time/ Frame	Frames/Sec	Memory octets
10 kbit/s	10000	1.2736	0	0
100 kbit/s	100000	0.12736	7	11144
1 mbit/s	1000000	0.012736	78	124176
10 mbit/s	10000000	0.0012736	785	1249720
100 mbit/s	100000000	0.00012736	7851	12498792
1 gbit/s	1000000000	1.2736E-05	78517	124999064
10 gbit/s	10000000000	1.2736E-06	785175	1249998600
100 gbit/s	1E+11	1.2736E-07	7851758	12499998736

Relative Delay
Aggregation Delay/Elapsed Time + Transmission Delay

What determines Burst octets or Frames?

- It is effective BW not Actual link rate when effective bandwidth is less than the link rate.

What else ?

- Burst Limit Memory?
- Burst limit aggregation delay?

Offer configuration of burst frames
0 to Max 100 Thousand frames?
~20 MB of memory

Channel Configuration Parameters

- max-mppdu-payload-size
- Bitrate
- Interval – in nanoseconds

Use a choice statement.
Configure one Compute the other

- Channel Burst-size
  - Uint32 octets (or Frames) Not using range in YANG.
- Actual bitrate and actual Interval can be computed by the system and reported. The actual frame size on the wire is used to compute the rate = interval x effective frame size on the wire. System has to add the MACsec frame sizes and media overhead.
- One problem – Non–collocated MAC PrY does not know MACsec/Media Overhead
  - If entering Interval, a slightly higher value (~2%) can be used. If entering bandwidth, a slight smaller value (~2%) can be used. Only a factor when using full link bandwidth.

Frame Configuration Parameters

- max-mppdu-payload-size
- Frame burst-frames
  - Range 0 – 100,000 usually a much smaller number 10-100. Not using Range in YANG
- nearest-multiple-pad
  - range "0|64|128|256|384|512|768|1024";
  - Pad to “512” means anything <= 512 = 512, anything >512 = 1024

Revised Prototype YANG Model (snippet)

```
augment /if:interfaces/if:interface:
 +--rw pry {macsec-priv}?
    +--rw mac-privacy-enabled?         boolean
    +--rw pry-source-address?          ieee:mac-address
    +--rw pry-destination-address?     ieee:mac-address
    +--rw user-priority-to-pry* [user-priority]
    |  +--rw user-priority    uint8
    |  +--rw privacy-type?    union
    +--rw privacy-channel* [channel-id]
    |  +--rw channel-id                identityref
    |  +--rw max-mppdu-payload-size?   uint16
    |  +--rw mppdu-priority?           dot1q-types:priority-type
    |  +--rw (interval-or-bitrate)?
    |  |  +--:(interval)
    |  |  |  +--rw interval
    |  |  |     +--rw requested-interval?   decimal64
    |  |  +--:(bitrate)
    |  |     +--rw bitrate
    |  |        +--rw requested-rate?   uint64
    |  +--ro actual-interval?          decimal64
    |  +--ro actual-bitrate?           uint64
    |  +--rw burst-size?               uint32
    +--rw privacy-frame* [frame-id]
    |  +--rw frame-id                  identityref
    |  +--rw max-mppdu-payload-size?   uint16
    |  +--rw mppdu-priority?           dot1q-types:priority-type
    |  +--rw nearest-multiple-pad?     uint16
```

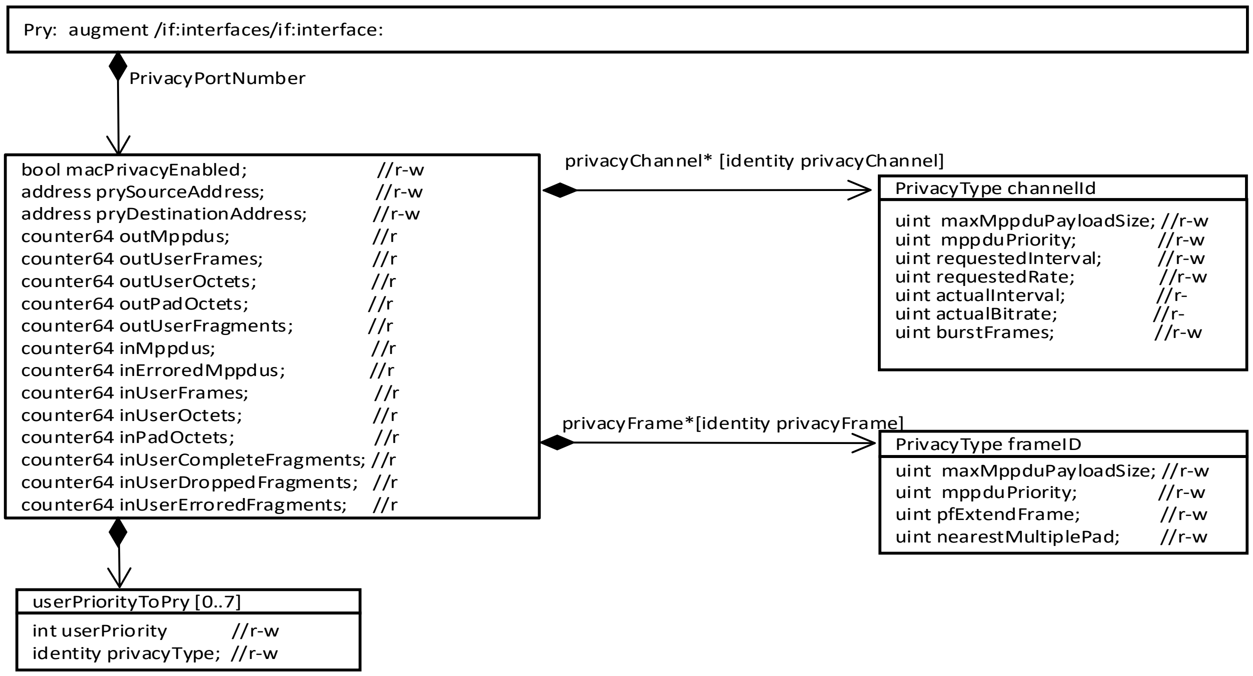
What gets configured

What gets configured

What gets used – System may adjust

Burst size as octets or frames?

UML Config



Pry: augment /if:interfaces/if:interface:

PrivacyPortNumber

```
bool macPrivacyEnabled;                  //r-w
address prySourceAddress;                //r-w
address pryDestinationAddress;           //r-w
counter64 outMppdus;                     //r
counter64 outUserFrames;                 //r
counter64 outUserOctets;                 //r
counter64 outPadOctets;                  //r
counter64 outUserFragments;              //r
counter64 inMppdus;                      //r
counter64 inErroredMppdus;               //r
counter64 inUserFrames;                  //r
counter64 inUserOctets;                  //r
counter64 inPadOctets;                   //r
counter64 inUserCompleteFragments; //r
counter64 inUserDroppedFragments;  //r
counter64 inUserErroredFragments;  //r
```

privacyChannel* [identity privacyChannel]

PrivacyType channelId

```
uint  maxMppduPayloadSize; //r-w
uint  mppduPriority;       //r-w
uint requestedInterval;    //r-w
uint requestedRate;        //r-w
uint actualInterval;       //r-
uint actualBitrate;        //r-
uint burstFrames;          //r-w
```

privacyFrame*[identity privacyFrame]

PrivacyType frameID

```
uint  maxMppduPayloadSize; //r-w
uint  mppduPriority;       //r-w
uint pfExtendFrame;        //r-w
uint nearestMultiplePad;   //r-w
```

userPriorityToPry [0..7]

```
int userPriority        //r-w
identity privacyType;  //r-w
```

Another Possible Channel config

- An Exact bits on the wire MAC Pry plus MAC Sec
  - This would allow accurate interval nearest nanosecond computation and entry
  - This value would have to include media overhead.
  - Systems could override if this number was wrong
    - real size is determined by max-mppdu-payload-size + overhead.
    - This number could override the max-mppdu-payload-size
  - Is this number a constant?
  - Or leave it to systems if they want this?

What if MAC PrY traffic is TSN?

- Should use a MAC Privacy Frame and timing has little impact (but frame sizes are slightly larger).
  - ieee8021SrpStreamTspecMaxFrameSize would be adjusted to account for MAC PrY and MACsec overhead.
  - No interval specified
- What if the TSN traffic is in a channel ?
  - Usually Not recommended. This would add delay to TSN frames. Any frame interval would be mapped to TSN parameters.
  - If the Channel used a small frame size (~256-512) octets and fragmentation was enabled? Not intended operation but possible.

Summary Channel config – Need to decide

- Two options:
    - Interval or Bandwidth
    - max-mppdu-payload-size
    - Should we configure Frame size as bits on the Wire?
        - Interval would specify an exact bandwidth.
    - Burst as bits on the wire.
- Or:
    - Interval or Bandwidth
        - Interval is TSN friendly, however actual rate is dependent on system size of Frame size bits on the wire
    - max-mppdu-payload-size
    - Burst as number of frames

Comments?

Thank You