Preemption and MACsec replay protection

Mick Seaman

Use of the proposed IEEE 802.3 Ethernet frame preemption capability could result in frame reordering from MACsec's point of view. In the absence of some change in the MACsec specification (IEEE Std 802.1AE) or its use it would not be possible to use MACsec to provide strict replay protection. Moreover it is currently thought to be difficult or undesirable to place tight bounds on the degree to which apparent misordering can occur. This note builds on 802.1 Security Task Group discussions[1] at the July 2014 meeting to examine the issues and a number of possible solutions or mitigations. The conclusions may affect the detailed changes to be made in amendments to IEEE 802.3 and IEEE 802.1Q (in P802.1Qbu) as well as to IEEE 802.1AE.

---

1. Introduction

Before discussing 'solutions' it is as well to be clear about the following:

- The 'threat model' assumed by MACsec (2.)
- What MACsec attempts to do (3.)
- How MACsec provides replay protection, and why that can be useful (4.)
- How preemption can misorder frames (5.)
- MACsec replay protection processing (6.)
- What additional threats might arise from enabling preemption

This note discusses some potential approaches to the issues raised by preemption:

1) No change to the current MACsec specification (7.1).
2) Relying only on the existing preliminary recheck, removing the check performed when a received frame has been validated (7.2).
3) Separate SCIs for low priority (subject to preemption) and higher priority (possibly prempting frames) (7.3).
4) Revising the MACSec packet format to separate the PN's cryptographic nonce and replay protection functions, replacing the later with an additional sequence number carried at the end of the protected frame, immediately prior to the ICV.

Each of these approaches is detailed below, together with some explanation of terms and concepts for those not familiar with MACsec. At present I prefer the second (7.2), which would require only modest changes to IEEE 802.1AE. P802.1AEcg is already likely to makes changes in the relevant areas and could include those for preemption. The last approach (revising the MACsec packet format) is only mentionned for completeness, least it be suggested at a late stage in the standards process. I believe it should be considered and rejected now.

2. Threat model

MACsec assumes that an attacker can copy, modify, remove, and add frames at will.

3. MACsec goals

IEEE 802.1AE details MACsec's goals, but for our present purposes a somewhat higher level view is useful: in addition to addressing issues of confidentiality, MACsec attempts to avoid the need for each and every higher level protocol designer to craft protocol-specific mechanisms to counter attacks using the posited threats (2. above). MACsec operates in the context of a single LAN[2] and of perimeter security, to reduce or remove effects that attacks made on that LAN can have on the operation of the rest of the network. An attacker possessing access to a LAN and the capabilities assumed in the threat model can clearly make that LAN unusable, but if MACsec ensures that it is localized and hence easier to contain, investigate, and counter.

4. Replay protection

MACsec adds a SecTAG to each frame to carry, amongst other parameters, a packet number (PN) that is incremented with each frame transmitted and provides each instantiation of the symmetric cryptography primitives with the unique nonce it requires. On receipt MACsec provides a configurable replay window, the leading edge of which is determined by the PNs of successfully validated

[1] I am indebted to Brian Weis' notes of our discussions, though this note contains new and changed material.Any mistakes are my fault.
[2] Though that LAN can be large and complex, see MACsec hops http://www.ieee802.org/1/files/public/docs2013/ae-seaman-macsec-hops-0626-v03.pdf

frames. If the replay window is zero then a subsequent frame is only accepted if its PN is greater. In that case MACsec provides strict replay protection. Otherwise frames within the trailing window are accepted, even if repeated.

Some protocol stacks behave logically correctly but with significantly reduced performance if frames are received out of order. For example, end station implementations of TCP/IP used to (and may still) suffer in this way. A misordering attack thus has at least some nuisance value at a distance[3].

Other protocol entities are more dependent on in order reception. For example, simple registration protocols with idempotent messages generally assume that the state of each entity's peers is contained in the last received message. Of course replay protection cannot defeat attacks that could be equally made by simply removing frames from the communication—simple lack of registration for example. The additional risk that replay protection can guard against is that of a flapping attack—alternating repetition of old and new messages that cause the receiving MACsec protected entity to propagate alternating states to distant entities.

MACsec can protect against the indefinite repetition of such messages by bounding the transit delay of protected frames. MKA (MACsec Key Agreement) carries the necessary PN information to advance the lower edge of the replayWindow if the transmission rate itself is not sufficient to guard against replay. The periodic transmissions used by simple protocols to guarantee convergence after loss thus suffice to limit the time during which flapping attacks can be made: when registrations (or similar demands placed on the network) are stable there are no old messages that are sufficiently recent to be accepted by MACsec.

5. Preemption and misordering

Preemption in the Ethernet MAC, as currently proposed[4], allows a frame that is currently being transmitted to be interrupted by one or more preempting frames. Once the prempting frames have been transmitted, in their entirety, transmission of the preempted frame resumes (without any retransmission of the data already sent) though it may be prempted once more. Preempting frames cannot themselves be prempted, nor can a frame that has the attribute of being able to preempt another if it happened to be transmitted while the latter was still 'on the wire'[5]. There is a lower limit to the size of fragments that a preempted frame can be broken into, of the order of 64 octets, with the exception of final fragment[6]. At present there is no bound on the number of preempting frames that can be sent.

If there are no intervening bridges, two commmunicating systems experience misordering as a consequence of the complete reception of a preempting frame occuring before the complete reception of the frame it preempts. The initial octets of both frames are received in order.

However if there is an intervening bridge, possibly operating transparently to the attached systems (as would a TPMR between two Customer Bridges) then that would receive and reorder the frames before transmitting them. Thus the receiving system might receive the two frames in their entirety, without one seeming to preempt the other. The attribute 'preempting' is not, therefore, firmly associated with a transmitted frame and cannot be used as part of a general solution to the problem of retaining strict replay protection capability when preemption is being used. The TPMR or even more invisible device might have been inserted by an attacker.

On the other hand preemption is of use in scenarios where timing and resource allocation are tightly controlled and is probably of marginal utility if there are intervening buffering systems that cannot know (because MACsec is rendering the contents of each frame confidential) which frames are candidates for preemption or being preempted. So we do not neccessarily need a solution that will not discard otherwise good frames because they have passed through some intermediate low-level bridge.

6. MACsec processing details

6.1 Receive

Details of MACsec's receive processing are modelled as shown in Figure 1 (a copy of Figure 10-5 of 802.1AE-2006 as amended by 802.1AEbw-2013). The standard does not assume that all valid implementations can carry out cryptographic validation of receive frames irrespective of LAN utilization and frame size, however desirable that might be, and thus its model[7] of receive processing includes a receive fifo (shown halfway up the figure) prior to validation. A number of operations can or

[3]I use 'at a distance' in contrast to 'localized' to mean an attack that can have a network effect that extends beyond the receiving MAC Security Entity.
[4]So far as I understand it.
[5]
[6]This needs checking, it would be nice from an implementation performance point of view if the minimum size of the final fragment was also constrained.

need to be performed prior validation and there was no reason to model those as being implemented at other than full line rate. They also, conveniently for the purposes of the current discussion, involve no frame data other than that present in the SecTAG and thus present in the initial octets of any received frame, whether pre-empted or not[8]. Contrariwise frame validation, and hence the operations shown in the upper half of the figure, cannot be completed until a entire frame has been received. The receive model can therefore be taken (if desired) as applying to a stream of prempting and preempted frames with in order receipt in the bottom half of the figure followed by some reordering in the fifo as preempting frames overtake their immediate preempted predecessor.

Replay protection is modelled as occuring both before and after frame validation, that is to say both in the lower half of the figure before the receive fifo and in the upper half after validation. The lowest acceptable PN can only be updated by a received frame (to the value of the PN carried plus one minus the size of the replay window[9]) after that frame has been successfully validated. If the replay window size is to be accurately enforced, the PN of each frame has to be checked against a lowest acceptable PN that could have been updated by the immediately prior frame.

6.2 Transmit

Details of MACsec's transmit processing are modelled as shown in Figure 2. As with receive processing, the standard does not assume that all valid implementations can carry out cryptographic protection at line rate and the model includes a transmit fifo. It matches the rate at which Controlled Port service requests occur, which could be arbitrary, with protection and line rates. This fifo is arguably unneccessary: service primitives are observed eventsT not procedure calls or other local interface operations —if the MACsec Controlled Port does not accept a frame from its service user the corresponding request primitive does not happen—the buffering is forced to exist elsewhere in the model of the overall system. However, when we come to modelling processing at a finer granularity than that of complete frames, buffering might play a role.

The PN acts as a nonce for the symmetric cryptography used in the protection operation, and is thus assigned as soon as the cipher suite[10] starts to calculate, AES block by AES block, the octets of a frame that will be transmitted and the integrity check value (ICV) that will be appended to the frame. The cipher suite currently standardized for use with .1AE use Galois Counter Mode (GCM) which has the advantage that the basic operations that it uses can be performed in parallel at Ethernet line rates[11]. This makes low latency implementations possible—octets of earlier blocks can be transmitted while those later in the frame are yet to be calculated. Such implementations have also been designed to exhibit constant latency, irrespective of frame size mix, thus supporting the use of .1AS/IEEE-1588.

To support strict replay protection, as specified by current standards, the MACsec transmit processing has to transmit frames in the order that it assigns PNs to those frames, and naturally computes the octets of each transmitted protected frame in that order. This is even true if the MACsec implementation is not tightly coupled to the Ethernet MAC but completes its calculation of the octets of the protected frame to be transmitted before submitting any of them for transmission. One could imagine an implementation of a bridge using the GCM and AES instructions now available on a general purpose processor, part way through the protection computation for a long frame and receiving a short high priority frame. The protection operation on the long frame could be suspended in favour of working on and transmitiing the short frame first. Thus the frames would be reordered in the transmit fifo, even in the absence of preemption capability in the Ethernet MAC. As previously mentionned, .1AE attempts to be reasonably tolerant of implementation diversity. The maximum permitted MACsec latency and jitter allow for processing a maximum sized frame and four minimum sized frames, and thus accomodate the loosely coupled implementation described. However it

[7]As always the standard points out that its model of operation is simply a basis for describing functionality and real implementations may adopt any internal model of operation compatible with the externally visible behavior that the standard specifies.

[8]The one test where this might be in doubt is that labelled in the Figure 'if(invalid_tag_or_icv(rx)'. At this point the only test applied to the ICV is to ensure that the received frame is of sufficient length to contain a correctly formatted SecTAG and a ICV of corresponding length. If MACsec is permitted to make the initial fragment of any premepted or preemptible frame at least 80 octets long (unless the entire frame is shorter) then this test can remain unchanged. Otherwise the test for ICV presence and size could be delayed until the frame is to be validated, after the fifo discussed. The only externally visible change resulting from such a change would be the incrementing of one error count rather than another if a frame is both too short to contain the specified ICV and in error so far as another test to be performed prior to the fifo is concerned.

[9]Obviously the lowest acceptable PN is not updated if the received value is, while acceptable, so low as to lower the value of the lowest acceptable.

[10]Each cipher suite specifies how the 16-octet block oriented AES is used to encode a longer sequence of octets and how that sequence is composed from the fields of the SecTag and the original frame data.

[11]Implementations processing at 100 Gb/s or more were reported some years ago.

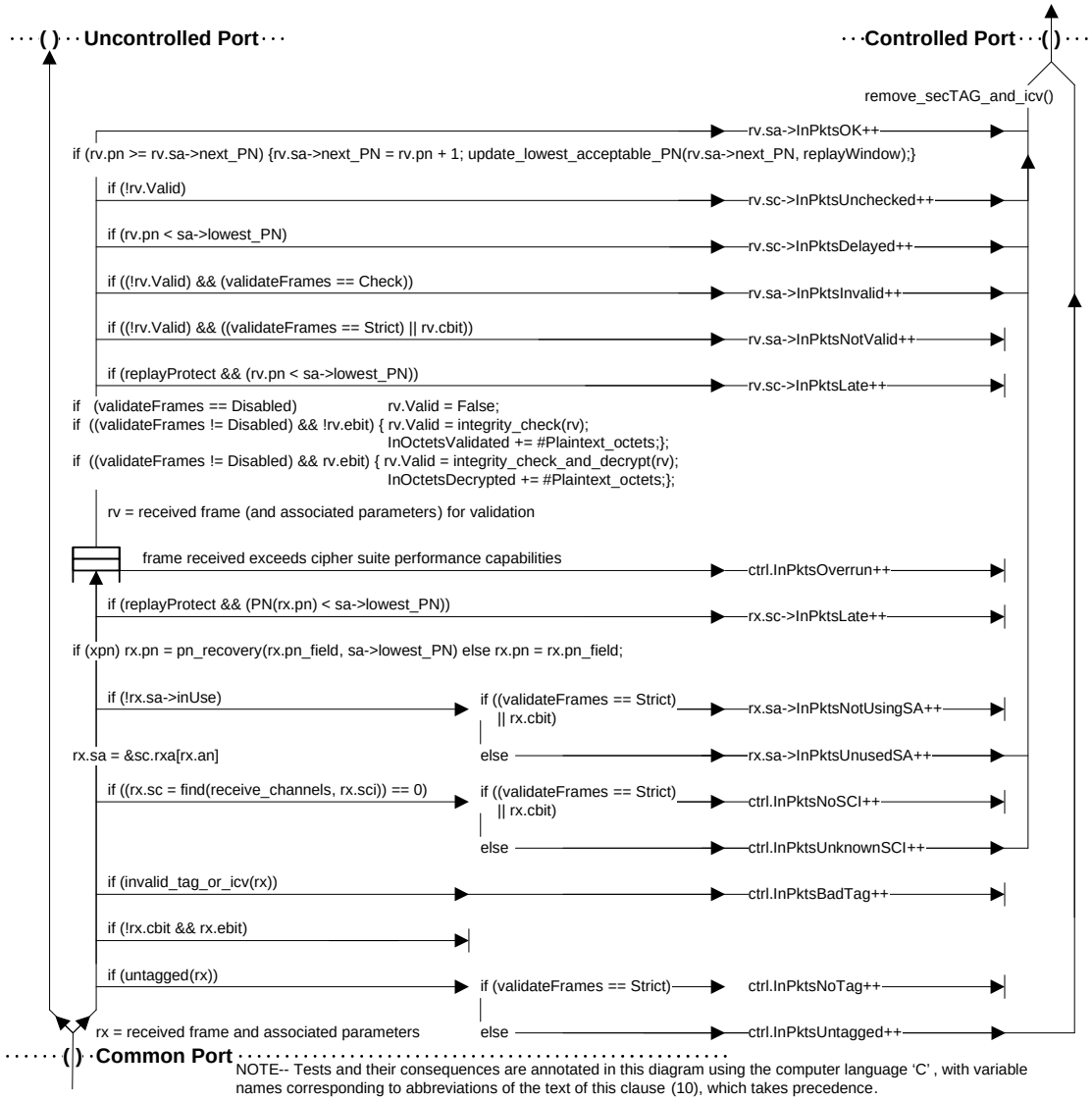

Figure 1—MACsec receive processing (.1AE-2006 amended by .1AEbw-2013)

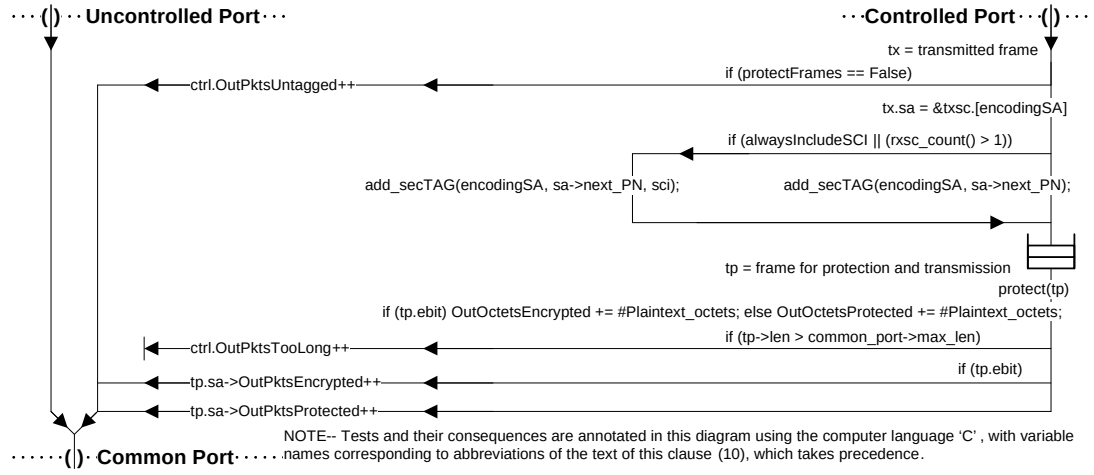

Figure 2—MACsec transmit processing

is unlikely that an implementation with such latency and jitter would meet the timing requirements of the systems that are the intended beneficiaries of preemption capability in the MAC. The latter are highly likely to use tightly coupled implementations that offer minimal jitter and a latency that is somewhat smaller than that corresponding to the transmission of a maximum sized frame. In any case the current .1AE standard does not provide reordering between the assignment of a PN and transmission of the corresponding frame, so (in the absence of intermediate bridges between transmitter and receiver) the initial octets of each frame should be received in transmission order even with preemption, though receipt of the complete frames may be misordered (see 6.2. above).

7. 'Solutions'

7.1 No change

One approach is to leave the current specification unchanged, forcing the user of preemption to choose a suitably large replay window. If this has to be done on a case by case basis because a tight window is deemed desirable then the standardized management counters provide some help: if the management variable replayProtect is false then frames outside the replay window are counted as late but not discarded. This allows a network manager to get some sense of what the replay window should be, before discarding frames that are late.

While the Ethernet MAC itself might not limit the number of preempting frames transmitted while reception of a preempted frame is suspended, it is not plausible that one hundred per cent of the bandwidth has been allocated to high priority or time sensitive traffic for any other a short period. As soon as a preemptable but unpreempted frame is received the received stream will be at the leading edge of the window. The necessary replay window values will, therefore, be quite small.

This 'no change' approach is made more attractive by the standard's procedures for bounding the time delay of transmitted frames—this reduces the intervals during which 'flapping' and related attacks might be carried out.

7.2 Preliminary check only

In the anticipated preemption use case scenario, no further bridges are interposed between the MACsec transmitter and receiver and the initial fragments of each frame are received in PN order (see 6. above). Therefore the preliminary replay check, just before the fifo in Figure 1, can be used. While this will not enforce strict replay protection at all times, the receive fifo is bound to empty frequently since it is not possible to arrange for the applied load to match the service rate exactly for extended periods without risking overrun (7.2.). In terms of changes to .1AE all that is required is to remove or turn off the 'if (replayProtect) && (rv.pn < sa->lowestPN))' check that occurs after the receive fifo.

7.3 Separate SCIs

<<Minimize associated changes by deriving SAKs for both SCIs from the existing distributed SAK (?). Works through intermediate bridges. Definitely more work than just relying on the preliminary replay check. Objections in terms of resources used as well as scope of changes to existing standards for arguably minor benefit. Review of management counters.>>