MACsec hops

Mick Seaman

IEEE 802.1AE MAC Security (MACsec) cryptographically protects frames on a hop-by-hop basis, with the scope of each cryptographic operation and key limited to a 'single hop'. When traversing a provider bridged or provider backbone bridged (PBN or PBBN) network, or other 'virtual media', that 'single hop' can be supported by a number of intervening bridges. IEEE 802.1AE describes a number of use cases, and specifies the interface stack (including the use and positioning of MACsec and the associated addressing) for each use case. This note explains the sense in which each of the .1AE use cases represents a 'single hop', why that restriction is important, and how to deal with some additional cases. It has been written mainly for an audience that is not familiar with the original development of both MACsec and other 802.1 protocols, but would like to explore additional possibilities.

1. Overview and conclusions

MACsec could be deployed in a many different and 'interesting' ways to achieve different security results —at the risk of taking a narrow protocol-centric view, and setting aside data rate, control plane performance, and interoperability considerations as well as the existence, needs, and evolution of other network protocols. This note explains why IEEE 802.1AE–2006[1] does not include all these possibilities, why it may prove difficult to procure equipment that uses MACsec in such custom configurations, and points to some of the deployment problems. In doing so it explains what is meant by a 'single hop'. A companion note[2] describes MACsec bridging configurations that support or exploit more recent developments in IEEE 802.1 bridging technology.

The 'single hop' restriction is a result of adopting a coherent approach to the considerations alluded to above, more specifically:

a) MACsec aims at cryptographic protection of all traffic on a LAN (3).

b) Achieving the necessary data rates at reasonable cost means limiting the number of SAKs (the secret keys used to protect data) in use at any one time (4.1).

c) Network control plane protocol performance should not be impacted by MACsec. Specifically, it should not be necessary to agree and install fresh SAKs if the network paths are reconfigured (4.2).

d) Performing MACsec processing naturally involves accessing packet data and, if confidentiality protection is being provided, modifying that data. To avoid costs that would be incurred by additional memory accesses it is desirable to locate the MACsec processing within chips/modules that are already concerned with moving the data, such as network interfaces.

e) Interoperability naturally requires that communicating MACsec SecYs (Security Entities) be peers (at the same level/position in an interface stack) and cryptographically protect/validate the same frame fields as each other.

Each of these points is detailed below. The control plane considerations ((c) above, 4.2) dictate that any given SecY's peers be those[3] that support the peers of the control plane protocol entities making direct use of the instance of the secure MAC service provided by that SecY. For example, if LACP[4] is supported (see Figure 3), each SecY has as its (only) peer the SecY at the other end of the link. Similarly, if RSTP is supported by the SecY in a Customer Bridge, each SecY's peers are those supporting RSTP in the neighbouring Customer Bridges. Each SecY is supported by a MACsec KaY (Key Agreement Entity). Each KaY can ensure that it discovers and communicates with the correct peers by using the same MAC Group Address as the other supported control plane protocols. For example, if communication between Customer Bridges is to be

[1] See IEEE 802.1AE-2006 7.3.2, and in particular NOTE 1 and NOTE 2 in that clause.

[2] Under development at the timing of writing.

[3] Strictly speaking "those, or a superset of those,", but since a SecY does not transparently pass the entire frame it validates the difference can only arise from selective static filtering of other control protocols. Whether such filtering makes sense or not is their affair. The present discussion is unaffected. Of course it may be a deliberate decision to protect only part of the path (as far as the nearest provider bridge for example). What is out of the question is to protect the path through and beyond the control plane entity's peer(s).

[4] Link Aggregation Control Protocol.

secured irrespective of the presence of intervening Provider Bridges, then the Bridge Group Address[1] is used[2].

2. Bridging architecture

A brief review of architectural concepts[3] follows, as an aid to our subsequent examination of detailed scenarios that illustrate what MACsec (and possible alternatives) can and cannot do well.

2.1 Layering and the ISS

Layered protocol entities communicate with their peer or peers using the service provided by the protocol entities in the layer below. We are really concerned with just one service—the MAC Internal Sublayer Service (ISS)[4]. This is simply the service provided by a LAN (the MAC Service), stripped of the peculiarities introduced by one or other media access methods, but with the explicit inclusion of parameters necessary for describing the process of forwarding the frame (which might otherwise be thought particular to the way the service is provided, and not of concern to upper layers in the end stations). The job of each and every bridge can be summarized as supporting one or more instances of the ISS, with the desired efficiency/extent/manageability/etc. Each ISS instance provides connectivity[5] between a set of end stations and/or bridges. At the lowest layer, in end stations or intermediate systems (bridges), the ISS is mapped to (provided using) the particular media access method supported by the physical LAN—connecting just those stations and bridges. The basic bridging function is to concatenate two or more instances of the ISS Some bridges, Provider Edge Bridges (PEBs) and Backbone Edge Bridges (BEBs) for example, include protocol entities whose explicit function is to provide one instance of the ISS over another. The protocol entities of others (Provider Bridges, for example) are configured to relay all the frames transmitted by others (Customer Bridges, in this example) so what they see as a single instance of the ISS is supported by two or more.

Virtual LANs (VLANs) are really just a way (using a VLAN tag field added to each frame) of separating multiple instantiations of the ISS, and the EISS (Enhanced Internal Sublayer Service) used by VLAN-aware bridging components is a compact way of describing that multiplexing at a service interface. An EISS service access point functions just as a number (potentially 4094) of ISS service access points in parallel. Similarly the MAC address encapsulation provided by Backbone Edge Bridges separates and provides address independence between a higher and lower layered instances of the ISS.

The layered architecture of a Bridge is often drawn as in Figure 1. This shows the bridge's MAC Relay entity below the level of the MAC Service used by higher layer protocols (supported by LLC) in the end stations to the left and right—emphasizing the fact that the relay is transparent to those service users. For our present purposes it is more convenient to use diagrams that show the interface stacks supporting relay and higher layer entities in more detail (e.g. Figure 2).

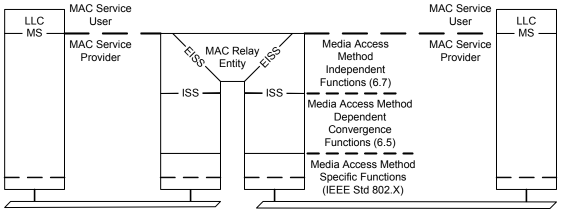

LLC
MS
MAC Service User
MAC Service Provider
EISS
MAC Relay Entity
EISS
ISS
ISS
Media Access Method Independent Functions (6.7)
Media Access Method Dependent Convergence Functions (6.5)
Media Access Method Specific Functions (IEEE Std 802.X)
MAC Service User
MAC Service Provider
LLC
MS

Figure 1—A VLAN bridge and end stations

2.2 Systems, networks, and components

Many protocols have been specified purely from the point of view of the rules governing the interaction of entities whose sole function is to provide the protocol (or worse by describing only the frame formats and—just possibly—individual field processing) and assuming that the relationship between those protocol entities and the rest of the system is obvious. In a layered system or network where the intent is often to provide or extend a service transparently (leaving the interactions between individual service unchanged, but increasing network throughput, extent, or the number of users) the same protocols can be used at many different layers, as can protocols and entities (such as those adding and removing VLAN tags) that are not truly transparent but serve to select between different instances of transparent service. For a standard to be useful—promoting the availability of equipment and interoperability between items of

[1] Also know as the "Nearest Customer Bridge group address" with the value 01-80-C2-00-00-00 (assigned in IEE Std 802.1Q-2011 Table 8-1).

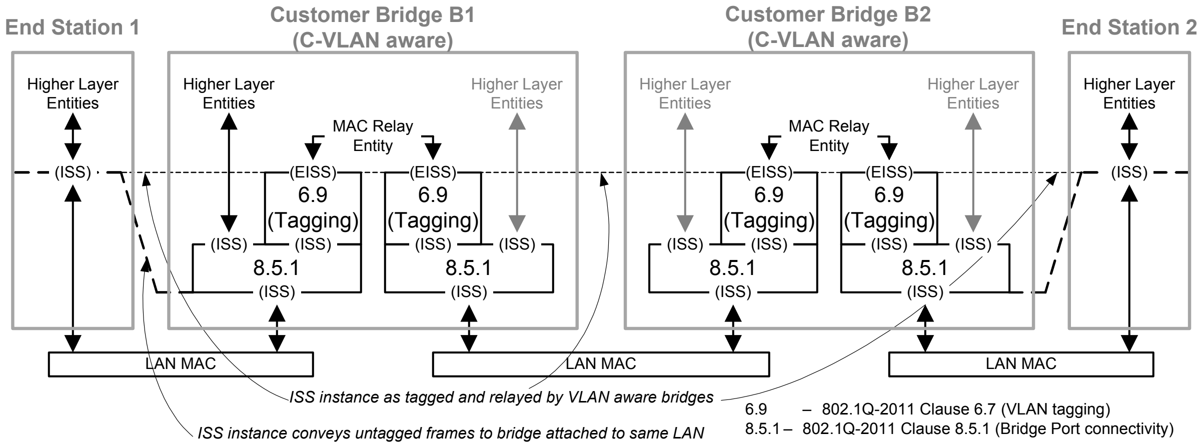
[2] See IEEE 802.1X-2010 clause 11.1 and Table 11-1, and

[3] The architectural concepts and terms are described in some detail in IEEE Std 802.1X Annex D.

[4] In most scenarios of interest Virtual LANs (VLANs) play a role, but VLANs are really just away (using a VLAN tag field added to each frame) of separating multiple instantiations of the ISS, and an EISS (Enhanced Internal Sublayer Service) service access point functions just as a number (potentially 4094) of ISS service access points in parallel. IEEE Std 802.1Q specifies trivial protocol entities that can be used to split/recombine an EISS interface into/from component ISS interfaces so that other protocol entities specified just for use with the latter can be used without respecification.

[5] IEEE Std 802.1AE and 802.1X-2010 Annex D.8 formalizes this notion as a Connectivity Association (CA), following RFC 787.

equipment developed by different organizations—it has to specify (or at least suggest) the layering relationship between (and concomitant configuration aspects of) protocol entities.



ISS instance as tagged and relayed by VLAN aware bridges

ISS instance conveys untagged frames to bridge attached to same LAN

6.9 – 802.1Q-2011 Clause 6.7 (VLAN tagging)
8.5.1 – 802.1Q-2011 Clause 8.5.1 (Bridge Port connectivity)

Figure 2—Interface stacks for VLAN-aware bridges and end stations

When very similar functions have to be performed at different sub-layers it is more useful to re-use an existing protocol entity, and the same applies to entire combination of protocol entities. The functionality of a Provider Edge Bridge is (for example) conveniently expressed (see Figure 3) as the concatenation of (a number of) C-VLAN aware components with an S-VLAN aware component, each of these internal system components having functionality that could be instantiated in separate systems (compare the left and right sides of Figure 3). There was some initial reluctance to follow this approach—a fear that it would mandate more functionality than necessary—but experience has shown its value. Some functionality that was initially explicitly removed has had to be reinstated to address real needs, and it is now a useful way of specifying new functionality, and of actually providing that functionality by interconnecting appropriately configured systems that are already available—rather than requiring custom engineering that might prove uneconomic in cases where relatively few systems are required. Part of the strength of this component based design is its natural inheritance and preservation of the essential arrangements for protocols that are not the immediate focus of the designer.

The lower part of Figure 3 is a plan diagram of the connectivity between the systems and their components. This view ‘from above’ serves to emphasize that the interface stack picture is best at showing a single path, and that other end to end paths can join and share part of this path. In some cases it is convenient to restrict the properties of one of the system components (requiring a component to serve a single customer, or to only have two ports, for example). Such a restriction permits some adjustment of the way that configuration protocols are supported by the system.

2.3 Connectivity and address scopes

In general, a configuration protocol needs to know its immediate peer neighbours in a network, and to be able to transmit and receive particular frames to and from those neighbours, while the same frames should not reach more distant participants in the protocol[1]. When the same protocol is to be deployed at different layers within the architecture, the set of protocol entity peers for each protocol instance naturally differs, and has to be kept separate. This separation is enforced by using different destination MAC addresses for each protocol instance. Specific group MAC addresses are used for this purpose—it is either impossible or impractical to manually configure the individual addresses of each entity’s peers before protocol operation begins. This use of group addresses is a general feature of LAN-based protocols, but the use of reserved group addresses that are always filtered by particular types of bridges is specific to the layered architecture of bridged networks.

Each MAC Relay entity includes a Filtering Database (FDB). FDB entries are used to ensure that frames with given destination MAC addresses (or given combinations of MAC address and VLAN ID) are not forwarded through any or some bridge ports. FDB entries can be created by management, by the operation of network configuration protocols (such as

[1] The protocol might need to transmit frames that reach all participants in a particular instance of the protocol as well. That is not our focus at present.

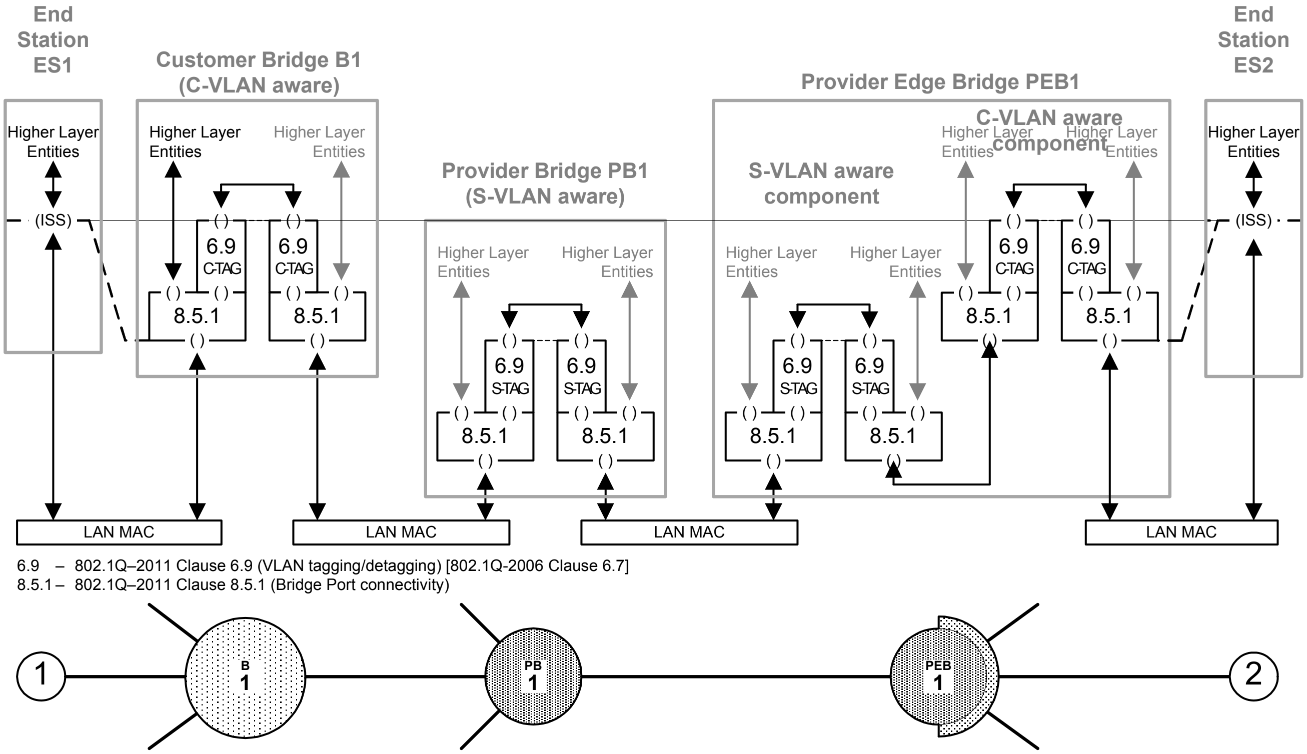

End Station ES1
Customer Bridge B1 (C-VLAN aware)
Provider Bridge PB1 (S-VLAN aware)
Provider Edge Bridge PEB1
S-VLAN aware component
C-VLAN aware component
End Station ES2
Higher Layer Entities
(ISS)
6.9 C-TAG
6.9 S-TAG
8.5.1
LAN MAC
6.9 – 802.1Q–2011 Clause 6.9 (VLAN tagging/detagging) [802.1Q-2006 Clause 6.7]
8.5.1 – 802.1Q–2011 Clause 8.5.1 (Bridge Port connectivity)
1
B 1
PB 1
PEB 1
2

Figure 3—Interface stacks for a path through a Provider Bridged Network

ISIS-SPB), or by learning the (relative) location of bridges by observing the source MAC addresses of frames. Permanent FDB entries are made for the Reserved Addresses used by protocol entities to discover their peers (at the appropriate layer), see Table 1 and Figure 4.

Table 1—Reserved addresses for bridge components

Value	Assignment	Filtered by		
		C-[1]	S-B-[2]	T-
01-80-C2-00-00-00	Bridge Group Address, Nearest Customer Bridge group address[3]	Y		
01-80-C2-00-00-01	IEEE MAC-specific Control Protocols group address	Y	Y	Y
01-80-C2-00-00-02	IEEE Std. 802.3 Slow_Protocols_Multicast address	Y	Y	Y
01-80-C2-00-00-03	Nearest non-TPMR Bridge group address	Y	Y	
01-80-C2-00-00-04	IEEE MAC-specific Control Protocols group address	Y	Y	Y
01-80-C2-00-00-05 01-80-C2-00-00-06	Reserved for future standardization - media access method specific	Y	Y	
01-80-C2-00-00-07	Metro Ethernet Forum ELMI protocol group address[4]	Y	Y	
01-80-C2-00-00-08	Provider Bridge Group Address	Y	Y	
01-80-C2-00-00-09 01-80-C2-00-00-0A	Reserved for future standardization	Y	Y	
01-80-C2-00-00-0B 01-80-C2-00-00-0C	Reserved for future standardization	Y		
01-80-C2-00-00-0D	Provider Bridge MVRP Address	Y		
01-80-C2-00-00-0E	Individual LAN Scope group address, Nearest Bridge group address[5]	Y	Y	Y
01-80-C2-00-00-0F	Reserved for future standardization	Y		

[1] Filtered by C-VLAN aware components in Customer Bridges and Provider Edge Bridges, and by VLAN-unaware MAC Bridges (IEEE 802.1D).

[2] B-components (in Backbone and Backbone Edge Bridges) behave exactly as S-components (in Provider and Provider Edge Bridges). The MAC address encapsulation provided by PEBs separates the address spaces for these components.

[3] As stated in 802.1Q-2011 (clause 13.39, and Table 8-1) a C-VLAN component (within a Provider Edge Bridge) that relays frames from a single Customer Edge Port to a single Provider Edge Port (see 802.1Q-2011 clause 15.4) may forward (not filter) frames with this destination address.

[4] This address is not exclusively reserved for this purpose; other uses are reserved for future standardization.

[5] It is intended that no IEEE 802.1 relay device will be defined that will forward frames that carry this destination address. Protocol uses include controlling Power over Ethernet.

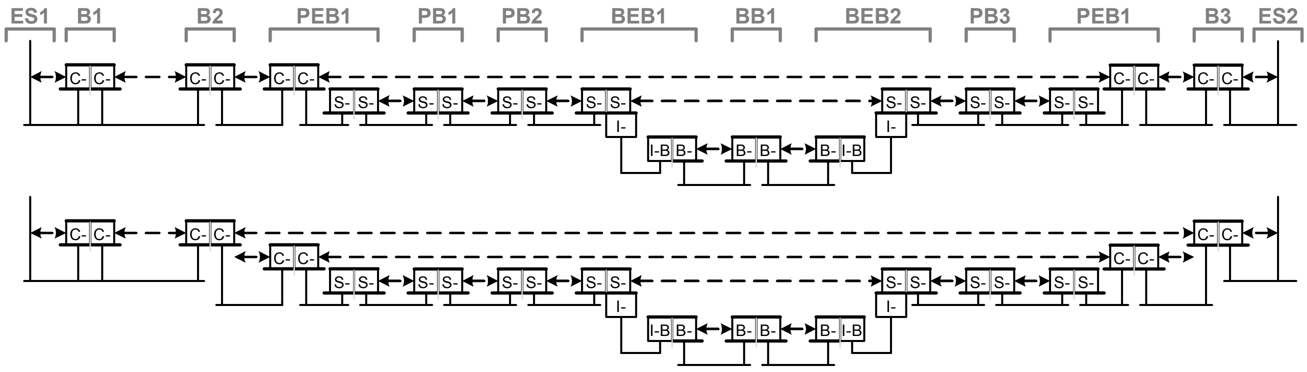


Figure 4—Address scopes in a Bridged Local Area Network

Figure 4 depicts a cross-section through a Bridged Local Area Network, using a simplified interface stack diagrams to illustrate the various address scopes provided by Table 1. TPMRs (Two Port MAC Relays) and the corresponding scope (bounded by the limits of real physical media and not passing through any bridge component) are not shown. Two sets of interface stack are shown, with the latter depicting the additional scope that can be provided if the C-VLAN components of the Provider Edge Bridges forward the Nearest Customer Bridge group address (see footnote 3 to Table 1). Backbone Edge Bridges (BEBs) do not currently support a similar scope for the directly attached Provider Bridges (PBs), and this could be consider a deficiency of the current standards since BEBs are necessarily administered by the backbone network provider while the Provider Bridges might well be administered by an entirely separate organization.

3. Security threats and requirements

As 802.1AE-2006 says in its opening paragraph:

"IEEE 802® Local Area Networks (LANs) are often deployed in networks that support mission-critical applications. These include corporate networks of considerable extent, and public networks that support many customers with different economic interests. The protocols that configure, manage, and regulate access to these networks typically run over the networks themselves. Preventing disruption and data loss arising from transmission and reception by unauthorized parties is highly desirable, since it is not practical to secure the entire network against physical access by determined attackers."

and (later in 1.1 Introduction):

"MACsec protects communication between trusted components of the network infrastructure, thus protecting ... network operation. MACsec cannot protect against attacks facilitated by the trusted components themselves, and is complementary to, rather than a replacement for, end-to-end application-to-application security protocols. The latter can secure application data independent of network operation, but cannot necessarily defend the operation of network components, or prevent attacks using unauthorized communication from reaching the systems that operate the applications."

Thus, although MACsec can provide confidentiality and data origin authenticity, it has more to do than just hiding the data transmitted and received on behalf of network users from prying eyes. Indeed, because end-to-end transmission is usually supported by IP, there is no way that security at or just above the MAC layer could ensure that user data is accessed by only the original transmitter and the final receiver.

A significant motivation for the standardization of MACsec was the desire to avoid the need to design a protocol-specific security mechanism for each and every protocol used as part of network control and configuration. It can be readily appreciated that such a protocol-specific security approach would most likely lead to a delay in the development of the necessary protocols, or in forcing a choice on every network designer between being able to use the latest (but insecure) or secure (but older) technology.[1] Satisfying the desire for MACsec to be capable of protecting all our MAC layer control protocols, including those yet to be designed, without serial development or deployment delays imposes additional requirements:

[1] This would mirror the early experience of those wishing to use MIBs as part of operational practice (i.e. to really manage their networks) in an era of MIB development as a separate arcane skill, quite separate from the rest of system and network protocol design. Under these conditions MIB development only starts when everything else is almost complete—ensuring that the first release product is only fully manageable through the console interface.

- the MACsec specification itself needs to remain unchanged when it is to be incorporated into a new type of system (or network);
- the specification of the new system needs to naturally provide the right interfaces and opportunities for the addition of MACsec, even if no thought has been given to the subject by the system's designers;
- existing MACsec capable systems, within the network or available off the shelf, should not require modification to operate in a network with the new control protocols or systems unless they explicitly need to use the new protocols.

One consequence of the above is that MACsec needs to protect all the traffic transmitted and received on a given LAN, if the traffic is to be protected at all. A more or less general way might be devised to subset this protection (identifying those protocols that do have their own security mechanisms and skipping protection for those frames, for example) but would not reduce the frame protection and validation performance requirements (high throughput, very low delay) since such frames constitute a very low percentage of the potential load.

The point has to be made that when (as is the case for the vast majority of bridges[1]) each bridge learns from the source MAC address of forwarded frames, then each frame forwarded is de facto a control frame, potentially altering the configuration of the network, as well as being a data frame. If such frames are not validated by the forwarding bridge, an attacker with LAN access can selectively deny service by transmitting frames (with a source address that is being used by legitimate traffic) on the 'wrong' LAN. A crude DoS attack, aimed at simply denying all service and possibly carried out by sending large number of frames to overwhelm a switch's control processor, might be easily detected, but a learning attack offers more possibilities. The attacker might, for example, attack a source only when it transmitted frames secured by IPsec (or some other protocol) and thus persuade the frustrated user to turn security off. Note: this attack could be carried out with frames with an Ethertype reserved for an 'end-to-end' authentication protocol if individual LANs on the path are not secure—existing bridges will (and should) learn from the source addresses of such frames.

While MACsec cannot protect end-to-end if IP routers lie along the path, the requirement is often to protect only part of the path even if each end uses the MAC address of the other directly as the destination in the frames it transmits. If part of the path is known to be physically secure (within a cage in a co-location facility, for example) there is no particular need to require MACsec capability on the end equipment (which might be a router without MACsec capability, for example). In such cases there is a positive requirement to protect only those LANs in the path that an attacker might be able to access[2]. In general a network might comprise a number of trusted regions, each under the secure control of a single administration, connected by LANs or LAN services (such as provider bridged network) that may be controlled by a different administration and that are not (or are not trusted to be) physically secure. Requirements naturally arise to secure particularly exposed LANs in any network, to authenticate and secure connectivity between different administrations, or to secure connectivity 'end-to-end' where the ends are those of a path provided by a subcontracted administration and each 'end' of that path lies within equipment administered by the same organization—be that the organization providing the connectivity or the organization using it. IEEE 802.1AE–2006 Figure 11–12 provides some examples.

In the main the technical requirement for MAC security discussed here pertains to integrity protection (a frame that passes validation checks on reception has not been modified since its transmission), and to data origin (the frame was originally transmitted by an authenticated peer, or at least—in the case of group communication, as occurs necessarily though not exclusively with multicast on shared media—by a peer whose authentication has resulted in its ability to transmit secured frames. However it remains the case that the general perception of security is one of confidentiality (e.g. 'no one else knows you are selling me this diamond ring so cheaply?') rather than integrity (e.g. 'this is a diamond ring, right?'), so no security standard can be without it.

There will of course be cases where confidentiality is really required, and where IPsec may be impractical (?), cannot be applied until later in the envisaged

[1] Backbone bridges that only support PBB-TE or that only support shortest path operation using ISIS-SPB would be an exception, but even in a backbone one or two B-VLANs are likely to be dedicated to providing local management connectivity or other services that use station location learning. Even in 'exclusively routed' networks learning bridges (switches) can be found, playing a valuable (if largely transparent) role expanding interface port counts etc.

[2] This definite requirement has been a problem for proposed layer 2 and-to-layer 2 end schemes, such as the (never deployed and now withdrawn) 802.10 'interoperable LAN security'. In that standard bridges that sought to terminate the scope of protection on a path had to acquire the (secret) cryptographic keys from the end stations that authenticated their mutual communication, with the burden of a large number (potentially thousands) of keys in bridges that connected trusted and untrusted regions of the network. These keys might have to be acquired in a hurry if a network reconfiguration resulted in end to end paths traversing different bridges.

transaction (?), or has significantly worse price/performance in a particular scenario. In some cases confidentiality is far more important than delivery (if the network itself is under attack the user will choose other means of delivering data) or the operation of the network is the responsibility of a separate organization, and is secured independently of the user data conveyed. Such cases are discussed later in this note.

4. Securing connectivity

This section discusses how connectivity is to be secured, given the bridging architecture, systems, and networks (2. above), and the threats and requirements (3. above). It reviews relevant aspects of the available cryptographic technology (4.1), particularly the use of secret keys, before considering how those keys are to be used to protect frames (4.2), and how such protection is to be included within the network and system architecture (4.3).

4.1 Cryptographic technology

<Secret key, AES, data movement, data per key, changing the frame, when and how to process, choice of architectures based on how many keys, practical experience.>

4.2 Protecting frames

Consider the network fragment shown in Figure 5 (a). End stations ES1 and ES2 are connected by VLAN bridges B1 thru B3. Each of the connecting LANs/links is shown in red, to indicate that they are exposed to attack, carrying data that ought to be protected. How should this be done?

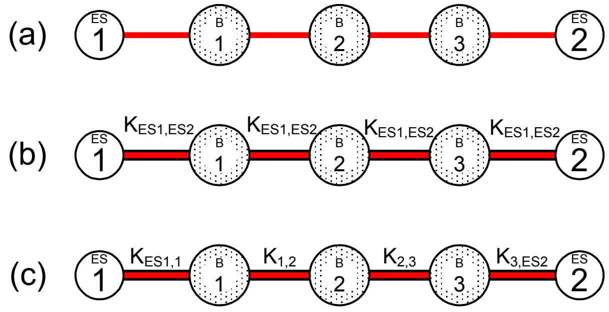


Figure 5—Protection along a path

An 'end-to-end' approach (Figure 5 (b)) protects the frame (shown in the figure by covering the links along the path with black) with a key ($K_{1,5}$) agreed between, and known only to the two end stations. This has a few downsides:

- The bridges cannot check the frame integrity, and thus safely learn (or refresh previously learnt information[1]) from the MAC source address (3. above), unless each of them also possesses $K_{1,5}$.
- A bridge cannot make (and protect) a change to the frame that would be permitted in (and may be essential to the operation of) an unsecured network, unless it also possesses $K_{ES1,ES2}$. For example, ES1 might transmit a frame without a VLAN tag, with B1 adding it (with the appropriate VLAN ID). Other stations, in different parts of the network, might be assigned to different VLANs, with all these frames (including their VLAN IDs) being destined for the ES2. In one typical network arrangement the ES2 is a router, and the VLANs correspond to IP subnets. Two stations, on different VLANs/subnets, might have the same MAC address. The assigned VLAN IDs have to be protected if one station is to be prevented from masquerading as another.

MACsec uses the 'hop-by-hop' approach shown in Figure 5 (c). A different key is used for each hop (LAN), with each participant validating the frame using the key for the reception LAN, and reprotecting the frame for onward transmission. The receiving station has to trust not only the transmitting end station, but also the intervening bridges, but this is also the case for the end-to-end approach (b) — once the latter is extended to permit and protect normal bridge functions (learning, VLAN assignment, ...). In both cases a secure infrastructure has to be established, and the users of the network provided by that infrastructure need to trust it.

The difference between these approaches is readily apparent when communication from ES1 to an additional station, ES3, is considered (Figure 6).

The end-to-end approach (b) requires a key for each communicating pair of stations, and each such key needs to be known by intervening learning bridges or by any bridge that needs to modify the frame[2].

The hop-by-hop approach facilitates incremental deployment. Initially it may be important to secure one link in the network (B3–B4 in Figure 7 (a)), while

[1] This is a vital part of a bridge's handling of network reconfiguration or end station movement.

[2] Addition of a tag is not the only potential modification. The priority bits in the tag can also be changed as part of normal class of service handling. After such a modification the frame would need to be reprotected with a key acceptable to the recipient.

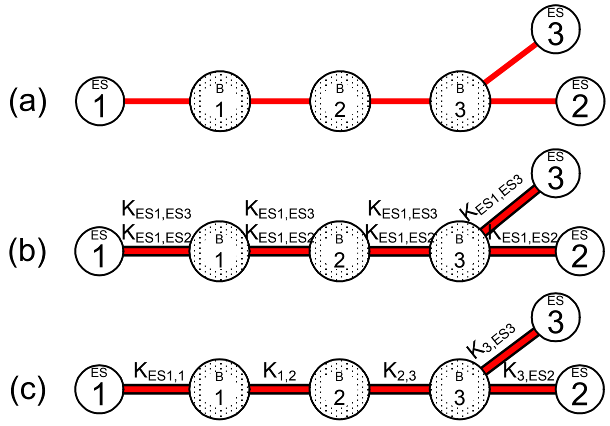


Figure 6—Protection to multiple stations

others are considered immune from attack even if not explicitly secured (shown as green in the figure).

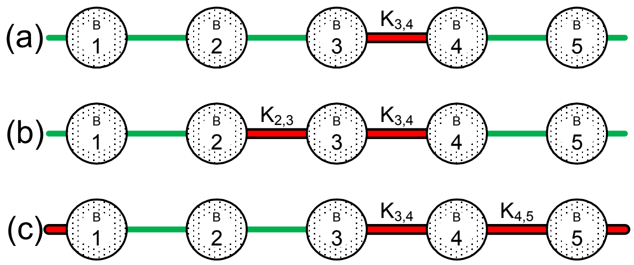


Figure 7—Protected and trusted LANs

The protected region or portion of the network path can then be extended, as in Figure 7 (b), though protected traffic within the region cannot be trusted unless appropriate controls/policies are applied to traffic entering it. If part of the network is truly immune from direct interference (the LANs connecting B1, B2, and B3 in Figure 7 (c) might be completely contained in a locked closet, for example) then they can form part of the trusted region.

The case where the physical connectivity to B2 is actually exposed to attack, and only unprotected because B2 itself lacks the capability, is more difficult. If B2 does not modify any of the frames it forwards (and those it originates are readily identifiable and subject to sufficient ingress policy controls by the adjacent bridges) it is tempting to protect the path from B1–B2–B3 with a key agreed by B1 and B3 (Figure 8 (a)). However this can pose problems. Almost all network designs provide alternate paths to protect against device or link failure (as in (Figure 8 (b)). A failure of the link B2–B3 should divert traffic from B2–B3–B4 to B2–B30–B4, and may well be supported by rapid reconfiguration protocol mechanisms (aimed at meeting or bettering a 50 millisecond service restoration time). Notifying B1 of the failure and having B1 and B3 agree and install the new key[1] is not currently part of such mechanisms and is unlikely to fit within the time budget.

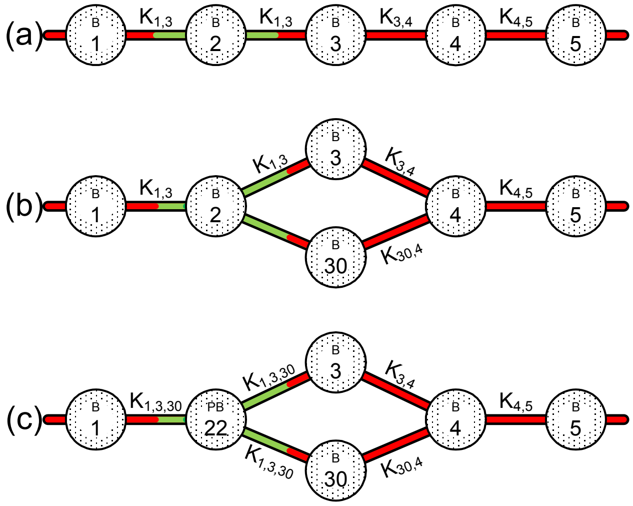


Figure 8—Network reconfiguration

However if B2 really does not modify forwarded frames it may be possible to treat it as operating at a lower (sub-)layer, as a Provider Bridge (Figure 8 (c), for example). In that case B1, B3, and PB30 are really one hop apart from the point of view of their network configuration protocols—they are all attached to the (virtual) shared medium supported by PB22—and can agree a group key.

Two systems may not be immediate neighbours for an instance of a configuration protocol in which they both participate, but the path protected by that configuration protocol may be constrained to pass between the two systems—if it passes through either of them. In that case it may be possible to omit the intervening systems from the configuration protocol, effectively placing them at a lower (sub-)layer, as in the above, and making the systems immediate neighbours. Simply forwarding Nearest Customer Bridge group addressed frames (see Table 1) through the C-VLAN components of Provider Edge Bridges has that effect.

4.3 The MACsec shim

The paradigm of connectionless networking, in which communicating peers can exchange data without previously participating in an explicit exchange to setup a connection (as required by X.25 or TCP), is now so prevalent as to pass without comment[2]. The

[1] In some cases the path B2–B30–B4 would not exist prior to the failover while the required key cannot be agreed until it exists, in other cases the path only exists for the purposes of supervisory traffic, which would not naturally report to B1, and in any case does not support key agreement protocol.

[2] See RFC 787 for a useful tutorial.

notion of a ‘connectivity association’ as an a priori association between communicating peers—that is to say an association created without their explicit knowledge[1]—remains useful. What concerns us is the connectivity association between neighbouring peers at a given (sub-)layer—the simple ability of a set of protocol entities to exchange frames without the need for the frame to be relayed at that (sub-)layer. This is the ‘single hop’ that we wish to secure, and in general we wish to secure it without the explicit involvement of the communicating peers—otherwise we would fail in our goal (see 3 above) to keep pace with the new protocol development that continues largely indepently of security concerns.

This secure connectivity association (CA, as defined by 802.1AE) formalizes our notion of a secured instance of the ISS, defining its extent and participants. The connectivity association implied by the existence of the ISS at any point in a protocol interface stack can be secured by the insertion of protocol entities whose operation is transparent to that of the existing protocol entities above and below. Such transparent protocol entities are known as ‘shims’. Figure 9 (compare with Figure 2) provides an example.

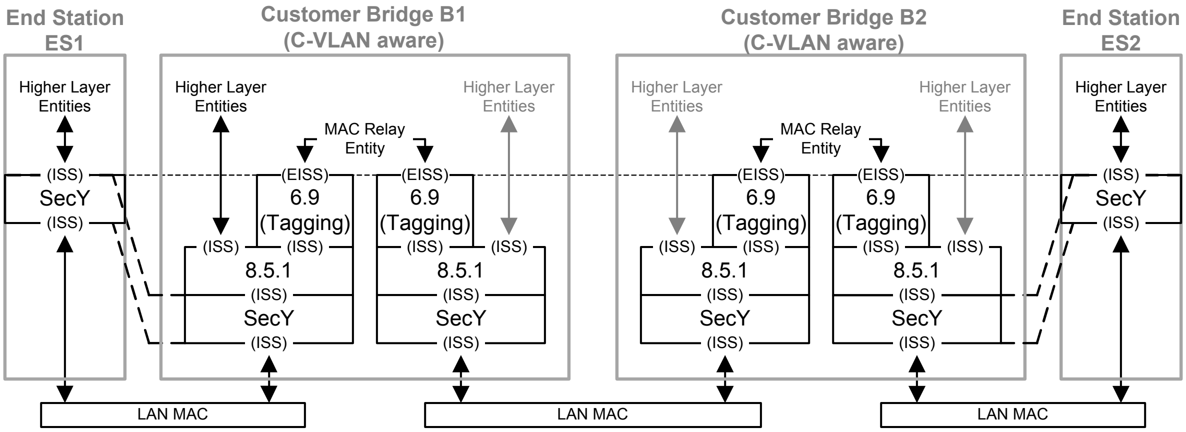


Figure 9—Securing the ISS in VLAN-aware bridges and end stations

Three connectivity associations (ES1–B1, B1–B2, B1–ES2) have been secured by the addition of the MAC Security Entities (SecYs). In addition to cryptographically protecting frames that pass between its upper (Controlled, or secured) port and its lower (Common Port), each SecY also supports transmission and reception of unprotected frames (through an Uncontrolled Port) so that companion protocol entities (PAEs and KaYs) specified in 802.1X-2010 can authenticate, reauthenticate, and agree keys with the other participants or potential participants in the CA[2].

The scope of the CA (if connectivity is permitted at all) is thus determined by the scope of the addresses used for authentication and key agreement. Management controls for the SecY determine whether insecure connectivity is permitted, or indeed whether the connectivity is to be secured at all. Moreover received frames are (at least notionally) passed both to the Controlled Port (for possible relay, if the system is a bridge) and the Uncontrolled Port (for use by authentication and key agreement). The destination address of the frame determines whether the relay’s FDB will result in forwarding, and whether the PAE or KaY will wish to process the frame. Consider the network path shown in Figure .

4.4

[1] The term ‘a priori’ does not simply mean ‘prior’, nor is it restricted to discussions of probability. See the Wikepedia discussion of ‘a priori’ (knowledge independent of experience)and ‘a posteriori’, and A.C.Grayling’s ‘An Introduction to Philosophical Logic’ (Chapter 3). In the present case examples of events and actions outside the media and subnetwork independent experience of protocol entities include plugging an Ethernet cable into a network, and setting up an ATM connection that will subsequently carry UDP packets. All that the protocol entities know is that they can, once active, transmit packets.

[2] Clearly no participant is to be given a key to participate in secure communication until mutual authentication has taken place. Authentication thus either implies, or is followed by, authorization. Authorization may result in changes to the management variables of other protol entities—permitting or denying access to certain VLANs, for example.