SECURE BRIDGING IN LARGE SCALE DEPLOYMENT OF ETHERNET

Khan Ferdous Wahid
Department of ICT, Universitat Pompeu Fabra, Barcelona, Spain

Javier Rubio-Loyola
CINVESTAV Tamaulipas, Mexico City, Mexico

Keywords: Layer-2 security, Link security, Secure communication, Secure channel, ID-based security.

Abstract: Considering the dominance of Ethernet with upcoming 100 Gbps line rate, service providers want to reduce their transport networks to simpler Layer-2 networks. Since existing Ethernet security mechanisms protect links in hop-by-hop basis, they cannot control access for disloyal authorized users in virtual or logical shared media infrastructure LANs. Also they leave data in clear inside intermediate systems, which increase threats when these systems are placed in public places. To address these critical security issues, we propose an authenticated on-demand secure bridging solution that can provide a point-to-point secure channel between Ingress and Egress Bridges across Bridged Ethernet network. To build such secure channel, we use Identity-based authenticated key agreement and signature protocol. Experimental results using our prototype software on a small multi-segment Ethernet network suggest that our solution is feasible, and guarantees secure bridging.

1 INTRODUCTION

The need for broadband networks and high transfer rate is rapidly increasing with progressive deployment of new technologies. Communication between Servers in Data Centers and usage of multimedia applications (voice, video etc.) by end users are the driving force behind the growth. Also companies have to share large scale information as fast as possible without any disruption. Service providers must adopt these demands in a cost effective manner. Ethernet is considered as a promising solution for the transport protocol because of its simplicity, high capacity, easy management, easy deployment and cost effectiveness. So Ethernet has evolved towards Metropolitan Area Network (MAN) although it was originally designed for Local Area Network (LAN), and unprotected components of the LAN are leaving secure premises. Hence Layer-2 (Data Link Layer) security is a rising concern in Computer Networks.

Uncontrolled access to LAN transmission and source unauthentication can help a station to masquerade the sender and to use unauthorized resources. Also non-recipient stations can get data in promiscuous mode and can change without an acknowledgement. Moreover, flat nature of Layer-2 address format validates any standard Media Access Control (MAC) address on a LAN, so intrusion becomes a tradition. The network community has faced many well known attacks on Layer-2 over the last several years. We are not describing any of these threats, because they have enjoyed rigorous analysis by community including manufacturers (Altunbasak, Krasser, Owen, Grimminger, Huth and Sokol, 2005; Vyncke and Paggen, 2007) and many of them only exist in unsecured networks, but our main target is to address the existing gaps in current security mechanisms. Also recently published surveys (Furnell, 2004; Bhandari, Grewal and Jha, 2006; Kolodgy, Pintal and Burke, 2008) depict the rise of insider attacks in organizations. Thus, one could certainly imagine the risks in the large scale deployment of Ethernet. Furthermore, no Layer-2 security approach requires a change in the upper layer business applications and this transparent behavior is a dominant factor to its growth.

The IEEE 802 LAN/MAN Standards Committee provided the Media Access Control Security (MACsec) (IEEE802.1AE, 2006) to encrypt and decrypt frames, whereas the key agreement scheme is provided by Port-based Network Access Control (IEEE

Ferdous Wahid K. and Rubio-Loyola J. (2010).
SECURE BRIDGING IN LARGE SCALE DEPLOYMENT OF ETHERNET.
In *Proceedings of the International Conference on Security and Cryptography*, pages 251-260
DOI: 10.5220/0002992202510260

P802.1X-2010, 2009). Besides, Secure Device Identity (IEEE P802.1AR, 2009) cryptographically binds an identity to a device. So the device should identify itself to a central authority before it can enable operation to get access in the network. These security mechanisms mainly control the unauthorized access, but authorized users have full power inside their secure domain, which facilitates disloyal users to abuse services. How they can abuse is described in detail at the next section. These mechanisms secure each link (physical and virtual) separately, which leaves data in plain text format inside the intermediate systems (e.g., Bridges). Thus, physical access to these systems provides full wiretapping facility.

Here, we present an Identity-based (ID) authenticated on-demand secure bridging mechanism which provides a point-to-point secure channel between Ingress Bridges (IB) and Egress Bridges (EB) over multiple links in Bridged Ethernet network. Our proposal aims to address the unsecured zones inside Bridges, thus limits the nature and extent of attacks. Our system uses several well-known security protocols (Yuan and Li, 2005; Hess, 2003; FIPS-197, 2001), which are comprehensively described in Section 3. The point-to-point nature of our solution provides full confidentiality between participating entities, and our multi-link secure channel eliminates unsecured zones. Co-operating with existing standard mechanisms and providing minimal infrastructure change while securing the existing gaps make the design and implementation of our solution a challenging problem. We clearly elaborate our implementation on a small multi-link Ethernet network in fourth section, also we clarify security, efficiency and limitation of our system. Then we describe related work in Section 5. Lastly, we conclude our paper in Section 6.

2 BACKGROUND

Here, we provide brief descriptions of existing Layer-2 security mechanisms and elaborate on the limitations that we have mentioned in previous section.

2.1 MACsec and Port-based Network Access Control

The Media Access Control Security (MACsec) mainly describes encoding mechanisms of the secure frame, incorporation of the secure entity in systems and operations of the supported secure services (IEEE802.1AE, 2006). On the other hand, Port-based Network Access Control provides the authentication, secure association and key agreement features. Together they provide the complete operations to secure a link. So each device incorporates MACsec and Port-based Network Access Control feature and support either group or point-to-point security on a segment. Group security requires the capability of virtual ports creation inside each port, which is recommended only in the Bridge ports for easy management of the Bridged network.

The Extensible Authentication Protocol over LAN (EAPOL) helps the Supplicant (end-station) to get the authentication through Authenticator (Bridge), whereas the communication between the Authenticator and Authentication Server (AS) is done through the Authentication, Authorization and Accounting (AAA) protocol, typically RADIUS. On successful authentication, AS sends a master key which is used to generate a secure connectivity association between Supplicant and Authenticator. Then the Authenticator sends the randomly generated Secure Association Key (SAK) to the supplicant over that connectivity association. When the group security is needed, the bridge creates individual connectivity association to each member of a shared medium to individually send the same SAK. Finally, MACsec encrypts the user data using the SAK and attaches a MACsec security tag (SecTAG) in front of the secure data. This SecTAG includes necessary information for the recipient, like MACsec EtherType, version number, secure data length, packet number (provides replay protection) etc. The whole frame, from the destination MAC address (DA) to the end of secure data, is then integrity protected using Integrity Check Value (ICV), which is added after the secure data. The Advanced Encryption Standard (AES) is used both in the data encryption and the ICV creation.

2.2 Secure Device Identity

According to (IEEE P802.1AR, 2009), every network interface should be manufactured with a secure Device Identifier (DevID) module that contains Management interface, Service interface, Asymmetric cryptography (RSA/ECC), Random Number Generator (RNG), Hash Algorithm (SHA-256) and Storage components. The Storage component stores all cryptographic keys and certificate credentials including the credential chain up to root credential (manufacturer's root certificate credential). The DevID secret installed by manufacturer are called Initial Secure Device Identifier (IDevID).

When a user starts his device, it automatically authenticates itself to the manufacturer's Certificate Authority (CA) and stores certified credentials in Storage component. The Locally-significant Secure Device Identifier (LDevID) is generated by DevID module on Network Administrator's request and a Certificate Signing Request (CSR) is issued to the Local CA. When Local CA replies with the signed certificate credentials, the DevID module stores it in storage component. The LDevID (cryptographically bind to IDevID) is used as device identifier in the LAN. If there is no LDevID, the IDevID works as device identifier although it is not recommended to use directly.

2.3 Limitation of Current Security Mechanisms

The hop-by-hop security forces the ingress port of a Bridge to decrypt frames before it relays them to egress port. Hence packet remains in clear inside every Bridge on the communication path. In large Ethernet network, there would be huge numbers of Bridges in public places, where physical access to these systems are easy enough to get disclosed information.

Analysis of the attacks in physical shared media is not necessary, because the use of physical shared media has declined over the years (IEEE P802.1X-2010, 2009), but virtual shared media supported in provider Bridged network can introduce similar threats as the physical shared media. When customers want to connect two or more geographically dispersed sites, they usually use Customer VLAN (C-VLAN) aware Bridge at the edge of each site where the service provider is connected, whereas Service Provider uses Service VLAN (S-VLAN) aware Bridges. If the Customer does not trust the Service Provider, normal MACsec cannot guarantee secure communication between two sites over the provider network, because the provider has access to every Bridge inside the Bridged Ethernet network where data are in clear inside each Bridge. To protect this, (IEEE802.1AE, 2006) provides two MACsec at each edge port of the C-VLAN Bridges. But this special solution requires double encapsulation and decapsulation, whereas normal Bridges do not support this feature. More than two sites may require group Security and it is not possible to maintain a point-to-point security between two group members, hence private communication is impossible in group mode, and this limitation may hinder secure communication between two similar departments on two sites. This situation becomes more critical if one site belongs to another Enterprise. Then either one of the Enterprises should trust the other's Authentication Server or they need to agree on a pre-shared key, but in real life it is rarely possible to trust other's Authentication Server, and pre-shared key installation for every other organization is a complex solution. Also small companies do not have their own authentication server, which leaves us in an inconsistent state. Moreover, a receiver needs to contact the CA online to verify the authenticity of sender. In our opinion, this introduces overhead and complexity in plug-and-play feature of Ethernet and disrupts continuous communication.

3 DESIGN OVERVIEW

This section describes our main contribution, an ID-based authenticated on-demand secure bridging mechanism that addresses the above critical problems by providing point-to-point security between the IB and EB (Figure 1). So whenever the source sends a frame, the IB encrypts the payload and protects the integrity of the whole frame with an Integrity Check Value similar to MACsec. Only the EB can decrypt the frame and check the integrity because it holds the same secret key. Hence the encrypted frame is transmitted over the Bridged Ethernet Network, and no Bridge on the communication path can handle the frame in clear. A change in the source or destination host generates another secret key inside the Bridges which provides separate secure channel for each pair and protects the pair-wise communication from other systems. We exclude the host systems to support unauthenticated non-MACsec host devices (backward compatibility), but every Bridge should support MACsec to build the secure infrastructure.

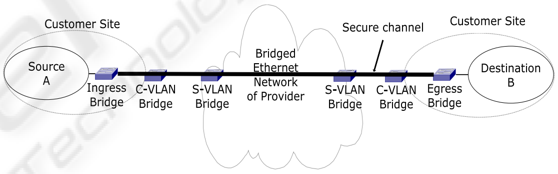

Figure 1: Secure channel over Bridged Ethernet network.

To create a secure channel like Figure 1, we must provide the same secret key between two Bridges, but the prediction of all pair-wise communications with separate secure channels is not possible in a large network because of huge storage requirement, complex key management and high communication overhead. For this reason, we want an on-demand scheme that can easily generate actual amount of keys when necessary and thus can reduce burdens. We can easily find the upcoming two-party communication if we look for an Address Resolution Protocol (ARP) request. So we should only initiate a secure channel when a host generates an ARP request message to find the destination for communication over the Bridged Ethernet network. To cope with this requirement and the plug-and-play capability of Ethernet, we need an on-demand key agreement technology without pre-shared key management. For this reason, we choose ID-based key agreement (Yuan and Li, 2005) and signature scheme (Hess, 2003). We now briefly review each technology before presenting our technical detail.

3.1 Review of Yuan-Li ID-based Authenticated Key Agreement Protocol

The ID-based cryptographic concept was introduced by Shamir (Yuan and Li, 2005) but the practical schemes were started after the proposal of Boneh and Franklin (Boneh and Franklin, 2001). We choose (Yuan and Li, 2005) because it has no security weaknesses compared to other ID-based solutions. They modified the Ryu, Yoon and Yoo's Protocol (Ryu, Yoon and Yoo, 2004), which is based on the bilinear pairing (e.g., Weil (Boneh and Franklin, 2001) or Tate (Frey and Ruck, 1994) pairing). The Bilinear MAP is a pairing function called $e : G_1 \times G_1 \to G_2$ that maps points of an additive group G_1 to a multiplicative group G_2, where both groups are cyclic groups of prime order p, and satisfies the following properties (Yuan and Li, 2005)-

- Bilinear: we have $e(xQ, yR) = e(Q,R)^{xy}$ for any Q,R of G_1 and for any x,y of Z
- Non-degeneracy: $e(P,P) \neq 1$, where P is a generator of G_1
- Computability: An efficient algorithm must exist to compute $e(Q,R)$, for any Q,R of G_1

The security of (Yuan and Li, 2005) is based on the Bilinear Diffie-Hellman(BDH) problem and the Computational Diffie-Hellman(CDH) problem. Ryu, Yoon and Yoo's Protocol has three steps: setup, extract and key agreement, where (Yuan and Li, 2005) modifies only the key agreement step to address the Key Compromise Impersonation Attack and Key Reveal Attack. In the setup stage, the Key Generation Center (KGC), which is the central authority to generate all private keys, selects $G_1,\ G_2,\ e : G_1 \times G_1 \to G_2$, and P as we described above. The KGC also selects a key derivation function H (a secure hash function), a master secret key $s \in Z_p^*$ and an encoding function $H_1 : \{0,1\}^* \to G_1$ that maps an arbitrary string to a non-zero point in G_1. Lastly, the KGC generates $P_{pub} = sP$ and publishes $< G_1;\ G_2;\ e;\ P;\ P_{pub};\ H_1;\ H >$. In the extract stage the public keys are derived from the corresponding IDs like $Q_{ID} = H_1(ID)$ and the KGC produces the private key for each ID as $S_{ID} = sQ_{ID}$. In the new key agreement step of (Yuan and Li, 2005), the user A chooses a random number $a \in Z_p^*$, computes $T_A = aP$, and sends T_A to B. On the other hand, B chooses a random number $b \in Z_p^*$, computes $T_B = bP$, and sends T_B to A. Then A and B individually compute a hash and the secret key, as shown in Equation 1 and Equation 2 (Yuan and Li, 2005):

$$A:\ h = aT_B = abP \text{ and } K_{AB} = e(aP_{pub} + S_A;\ T_B + Q_B) \quad (1)$$

$$B:\ h = bT_A = abP \text{ and } K_{BA} = e(T_A + Q_A;\ bP_{pub} + S_B) \quad (2)$$

Actually these two secret keys are same $K_{AB} = K_{BA} = e(P,P)^{abs} e(P,Q_B)^{as} e(Q_A,P)^{bs} e(Q_A,Q_B)^{s}$, so both parties have the same secret to calculate the same session key: $H(A,B,h,K_{AB})$.

3.2 Review of Hess ID-based Signature Scheme

If the sender knows the receiver's ID in the above scheme, he can generate the receiver's public key $Q_{ID} = H_1(ID)$ because H_1 is a public function, so sending authenticated T_A to B at the first place is easy. But it is complicated when the sender has no information about the ID of receiver, which is possible in Ethernet network that generates ARP request. So we need an ID-based signature scheme to cope with this requirement. For this reason we choose a very efficient ID-based signature scheme (Hess, 2003), which is backed by the survey of (Dutta, Barua and Sarkar, 2004). This scheme has four steps, setup, extract, sign and verify. The KGC chooses a hash function $H : \{0,1\}^* \times G_2 \to Z_p^*$ in the setup stage, and this is the extra operation in setup and extract steps than (Yuan and Li, 2005). To sign a message m the sender chooses a generator P_1 of G_1, a random number $k \in Z_p^*$ and computes $r = e(P_1,P)^k,\ v = H(m,r)$ and $u = vS_{ID} + kP_1$, where the pair (u,v) is the signature. On receiving the signed message, the receiver computes $r = e(u,P) \cdot e(Q_{ID}, -P_{pub})^v$ in the verify stage and validates the signature if $v = H(m,r)$.

3.3 Our Solution

Here, we divide the description of our solution in different useful properties to show how it fulfills each criterion.

3.3.1 On-demand Key Agreement

When we combine the above two schemes, we find that the aggregated solution requires two message transmissions to build the secure channel- 1) From the sender to the receiver that comprises the signed

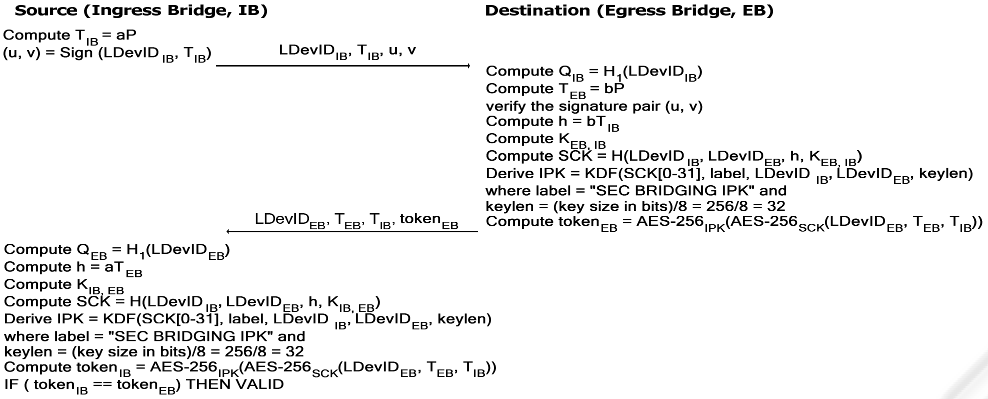
Source (Ingress Bridge, IB)
Destination (Egress Bridge, EB)
Compute T_{IB} = aP
(u, v) = Sign (LDevID $_{IB}$, T_{IB})
LDevID$_{IB}$, T_{IB}, u, v
Compute Q_{IB} = H_1(LDevID$_{IB}$)
Compute T_{EB} = bP
verify the signature pair (u, v)
Compute h = bT_{IB}
Compute $K_{EB, IB}$
Compute SCK = H(LDevID $_{IB}$, LDevID$_{EB}$, h, $K_{EB, IB}$)
Derive IPK = KDF(SCK[0-31], label, LDevID $_{IB}$, LDevID$_{EB}$, keylen)
where label = "SEC BRIDGING IPK" and
keylen = (key size in bits)/8 = 256/8 = 32
Compute token$_{EB}$ = AES-256$_{IPK}$(AES-256$_{SCK}$(LDevID$_{EB}$, T_{EB}, T_{IB}))
LDevID$_{EB}$, T_{EB}, T_{IB}, token$_{EB}$
Compute Q_{EB} = H_1(LDevID$_{EB}$)
Compute h = aT_{EB}
Compute $K_{IB, EB}$
Compute SCK = H(LDevID $_{IB}$, LDevID$_{EB}$, h, $K_{IB, EB}$)
Derive IPK = KDF(SCK[0-31], label, LDevID $_{IB}$, LDevID$_{EB}$, keylen)
where label = "SEC BRIDGING IPK" and
keylen = (key size in bits)/8 = 256/8 = 32
Compute token$_{IB}$ = AES-256$_{IPK}$(AES-256$_{SCK}$(LDevID$_{EB}$, T_{EB}, T_{IB}))
IF (token$_{IB}$ == token$_{EB}$) THEN VALID

Figure 2: Computation and message exchange in our protocol.

ID and T_{ID} of the sender, and 2) From the receiver to the sender that comprises signed ID and T_{ID} of receiver. The source needs ID-based signature scheme to send the message to an unknown destination, but when the destination replies, she has complete information about the sender, so she can use symmetric key based signature operation to avoid expensive computation. Also there is no acknowledgement between the communication parties in (Yuan and Li, 2005). For this reason, we combine the ID-based key agreement and signature scheme with symmetric encryption in our Key Agreement solution to provide faster computation. We assume that all the Bridges inside LAN get their private keys from the KGC. How to get these keys with minimum hardware change is explained later in the document. We already have the AES (AES-256) in our possession from MACsec. So the source Bridge sends his signed ID (LDevID) and T_{ID} to the destination. On receiving the message, the destination computes the session key, which is called Secure Channel Key (SCK) in our contribution, and derives an Integrity Protect Key (IPK) from this SCK as MACsec derives the connectivity association key (CAK) from the master key using Key Derivation Function (KDF), as shown in Figure 2. If we do not sign the first message to the unknown destination, any adversary could replace the T_A inside the message and thus the receiver unnecessarily computes everything and sends the reply to the source, which forces denial of service (DoS). So, in our opinion, the signing is required, but it does not prevent all type of DoS.

The destination encrypts her ID, T_{ID} and source's T_{ID} using the SCK, and the encrypted output is integrity protected using the IPK. The result of the integrity protection is attached to the replied message as a token of authentication and key confirmation. We attach the source's T_{ID} in the replied message to ensure liveness. On reception, the source can individually follow the same procedure to calculate the token and can match the token with the received one to verify the authenticity of the destination along with the confirmation of SCK. However, two extra message exchanges to build a secure channel for each pair-wise communication can reduce the Ethernet performance. So we are interested in minimizing the overhead.

3.3.2 Minimal Overhead

Normal ARP protocol works with exchange of ARP Request and Reply messages, and each packet is 42 bytes long. Such small size of ARP packets gives us an opportunity to add more bits at the end as data or payload bits, because maximum data field length is 1500 bytes in an Ethernet frame. Incorporation of our protocol messages in ARP packets provides an efficient way to minimize huge overhead, because we do not require any extra message exchange to build a secure channel. Thus, the message from source to destination (Figure 2) is embedded in ARP request and the message from destination to source is embedded in ARP reply. Our message format for the additional part of ARP packets comprises 12 bytes header and a variable length payload (Figure 3).

At first we want to write zeroes in the additional part of ARP messages to avoid information leakage which is described in (Arkin and Anderson, 2003) and incited by (Bruschi, Ornaghi and Rosti, 2003). The header starts with 4 bytes Protocol ID `0xCABFE001.` The next one byte is divided in two 4 bits value. The first or left four bits indicate availability of MACsec in source and destination host machines. We include this option for future Link Layer based Firewalls, it may help those Firewalls to force strict rules for unauthenticated devices. The last or right four bits indicate which type of pairing is used in the ID-based operations because there are different types of Elliptic curves and pairings (Lynn, 2007; PBC Manual, n.d.). The next one byte contains ID length of the generator of this frame and following two bytes store T_{ID}

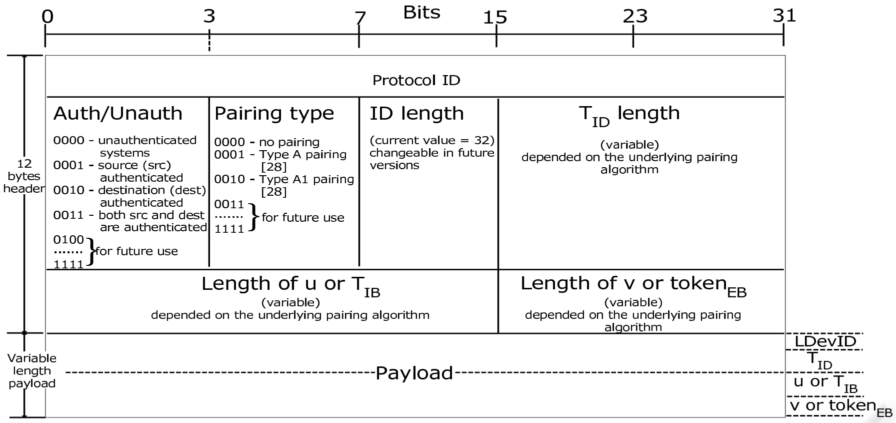
Bits
0
3
7
15
23
31
Protocol ID
Auth/Unauth
0000 - unauthenticated systems
0001 - source (src) authenticated
0010 - destination (dest) authenticated
0011 - both src and dest are authenticated
0100 .... 1111 for future use
Pairing type
0000 - no pairing
0001 - Type A pairing [28]
0010 - Type A1 pairing [28]
0011 .... 1111 for future use
ID length
(current value = 32) changeable in future versions
T_{ID} length
(variable) depended on the underlying pairing algorithm
Length of u or T_{IB}
(variable) depended on the underlying pairing algorithm
Length of v or $token_{EB}$
(variable) depended on the underlying pairing algorithm
12 bytes header
Variable length payload
Payload
LDevID
T_{ID}
u or T_{IB}
v or $token_{EB}$

Figure 3: Additional frame format in ARP packets (not to scale).

length, whereas the rest 4 bytes of the 12 bytes header are totally different for ARP request and reply messages. In ARP request message, the IB generates the signature and stores the length of (u, v) pair in these 4 bytes (2 bytes for each value). On the other hand, the EB includes the length of T_{IB} and $token_{EB}$ in these bytes of ARP reply messages. The payload part starts with the ID (LDevID) of the generator, whose length is presented in the header. For simplicity we do not include the Ethernet frame fields in Figure 3, but in actual operation the IB signs the whole Ethernet frame, from the Destination Address to the end of T_{IB}, and attaches the (u, v) pair at the end. This operation binds the source host address to the T_{IB} value. It is easy to know the (u, v) length prior the signing operation because the values are fixed for a specific type of pairing. EB signs the whole Ethernet frame with AES-256, which binds the hosts' addresses to the T_{EB} and T_{IB}. Lastly, both IB and EB calculate the same SCK. There is one question- how the EB distinguish him as the EB, because ARP request is a broadcast message. The EB is directly connected to the real destination, and each port of the bridge knows the destination MAC through (IEEE P802.1X-2010, 2009) operations. If we store IP address along with the MAC inside each port, the EB can easily understand the adjacent destination. Although physical shared media is deprecated, each physical port unnecessarily supports several virtual ports (65534 virtual ports) to accommodate several secure channels for the shared end stations, which require a significant amount of unused memory. On the other hand, IP storage requires only 32 bit memory, that is, in our opinion, far more reasonable than huge unused memory.

Moreover, we can optimize some operations (in Figure 2) by pre-computing their values. Any Bridge can pre-generate a random value $c \in Z_p^*$ and calculate $T_{ID} = cP$ and $(cP_{pub} + S_{ID})$ for the next key and hash values. Moreover, it can pre-compute $r = e(P_1, P)^k$ and kP_1 for signing any upcoming ARP request. We cannot pre-compute $e(Q_{ID}, -P_{pub})$ (Dutta, Barua and Sarkar, 2004), because we verify the signature of a sender only once for an ARP request.

3.3.3 Secure Communication

The IB and EB should store the SCK with a mapping to associated hosts' MAC addresses. Whenever host A and host B send frames to each other, these Bridges must use this SCK in AES (AES-256) encryption until a new key is generated for the pair, but the encrypted frame should attach an information TAG, like SecTAG of MACsec, so that the receiver can understand the Frame Type, payload size, and also can detect any replay attack. Our TAG field comprises EtherType, Short Length (SL) and Packet Number (PN), as described in Figure 4.

The EtherType for our proposal is `0xACEB`, which can be registered using the guidelines of (Eastlake, 2008) and the SL is similar to (IEEE802.1AE, 2006). MACsec has 4 bytes (32 bits) PN that must be used as an Initialization Vector (IV) in each encryption to generate different encrypted outputs of the identical clear texts, but 2^{32} minimum-sized frames can be sent in approximately 5 min on a 10 Gbps link (IEEE802.1AE, 2006), so it forces MACsec to change the key after each 5 minutes to avoid reusing the same value of PN. We increase the PN size by one byte, which gives approximately a 21 hour lifetime for each

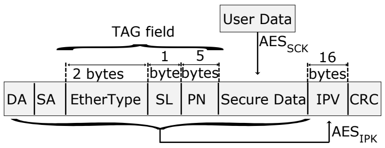

Figure 4: Frame format in our solution.

key when PN is increased sequentially for each frame. It is necessary for our solution because ARP cache generally expires after 20 minutes, whereas some operating systems do not expire the cache entries until cache becomes full (Bruschi, Ornaghi and Rosti, 2003). Moreover, AES needs a 16 byte IV, so we provide the concatenation result of Source Address and TAG field padded with 2 more bytes of zeroes as IV in each encryption. The IV usage defends against any successful attack on SCK. Lastly, we need an Integrity value to help the receiver to verify the correctness of the frame. For this reason, we provide an Integrity Protect Value (IPV) which is generated using IPK (Figure 4). Whenever one host's MAC address is expired in the CAM table, the Bridges should never use the associated key to secure further communication, because CAM expiration indicates that the host is not alive for long time.

3.3.4 Parameter Distribution and Key Revocation

The EAPOL could be used to distribute parameters from KGC, and we can initiate it as soon as MACsec starts its operation without any explicit steps. Moreover, key can be revoked with ARP cache timeout. Otherwise the revocation time is 21 hours as it is the maximum time limit for PN exhaustion. As Bridges may reside far away from each other, they could be non-synchronized, and counting of 20 minutes for ARP cache starts differently in two machines. So one machine can delete the cache and related secret key much earlier than the other end, and it brings inconsistency. Thus our system should store the key for several minutes to correctly receive encrypted frames from other end. The machine could store it in a temporary cache after elapse of 20 minutes (or 21 hours). To force key revocation, the machine can send an UNARP (Malkin, 1995) like message as soon as it receives a frame encrypted with a key of temporary cache (including a correct sequence number).

3.3.5 Minimal Hardware Change

DevID module in (IEEE P802.1AR, 2009) fulfils most of our needs. It contains Hash Algorithm, Random Number Generator and Elliptic Curve Cryptography. When the module requests for the certificate credentials of LDevID, the Local CA can act as the KGC and can generate the private key, like $S_{ID} = S_{LDevID} = sLDevID$, for each system. The generation of S_{LDevID} can be done concurrently with the generation of certificate credentials, so there is no need for extra pairing-based initialization step. We only need a component inside DevID module that handles all pairing operations and stores S_{LDevID}. Our encryption operations do not require any new functionality because MACsec has the AES. We need to store and bind the pair-wise key with associated CAM entries, and the KGC should issue new private keys after a certain period of time to prevent any successful attack on the key. The Local CA, which acts as the KGC in the Metropolitan Area should be accredited by Trusted Computing Group (TCG, 2003), Government's Security Division or similar kind of organization.

3.3.6 Support for Unauthenticated Systems

Every port of a Bridge should provide a flag in our solution. By default the flag value should be *zero* which means it is connected to an unauthenticated host. After successful authentication of a host by MACsec operations, the flag value automatically becomes *one* which represents authenticated status of the connected host. If flag value is *zero*, the Bridge always decrypts every frame leaving that port to the unauthenticated host and encrypts each frame upon reception. If flag value is *one*, the Bridge decrypts the frame by SCK and again encrypts using local SAK to send to an authenticated host. It does opposite operations when it received a frame from the adjacent host.

4 PROTOCOL ANALYSIS AND IMPLEMENTATION

4.1 Security and Efficiency

Our AES-based secure communication is partially evaluated in (Wahid, 2010b). Although the hosts are excluded here from the ID-based scheme, we assume that somehow one attacker gets the private-public key pair associated with his ID from the KGC and connects to a link between two Bridges, as shown in Figure 5. As the attacker has ID-based key now, he could easily build separate secure channels (channel 1 and 2) with Bridge 1 and Bridge 2, but it is not possible to take control of the secure channel 3, which is built between the two Bridges, because our system is based on the Yuan-Li protocol (Yuan and Li, 2005), which prevents the Key Compromise Impersonation and Key Reveal attacks (a.k.a. know key attack).

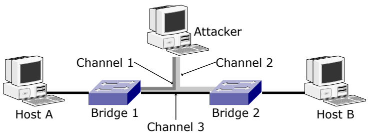

Figure 5: Impenetrable secure channel between two Bridges.

Table 1: Computational efficiency comparison between MACsec and our system.

Bridges between source and destination	Number of service provider in the network	MACsec	Our solution
5	0	10m	4m
6	0	12m	4m
6	1	14m	4m

If host A sends an ARP request to host B, the Bridge 1 embeds his T_{ID} value in the ARP message, signs it and relays to Bridge 2. The attacker can capture this message but cannot modify any value because he has no information about the private key of Bridge 1. Also any modification assures invalid signature. Besides, he cannot masquerade as the Bridge 1 because MACsec and Yuan-Li protocol prevent any ID forgery. We see here that our solution along with MACsec totally prevents data modification, masquerade and unauthorized disclosure. Also there is no unsecured zone in our solution because it provides end-to-end security, and the inclusion of T_{ID} guarantees liveness, hence, protects replay attacks.

Here the computational efficiency of our system is compared with standard MACsec operations. In MACsec, every Bridge decrypts and encrypts the frame in ingress and egress ports respectively, whereas our system does those operations only in IB and EB. We assume that m represents one encryption or one decryption operation inside Bridges, and both source and destination hosts are authenticated. Also we assume that source and destination are in two customer sites when there is a service provider. We can see from Table 1 that our solution depicts high computational efficiency compared to MACsec, and the encryption/decryption operations are constant irrespective of the number of intermediate Bridges or Service Providers. In MACsec, each edge Bridge to service provider does double encryption/decryption with computational overhead. Our system can provide point-to-point security inside the MACsec group secure domain because it is based on ARP.

4.2 Implementation

We have implemented the ARP-based on-demand key agreement feature in a closed environment, as shown in Figure 6. Our test bed comprises four Linux machines where two of them act like Bridges. All four machines have the same specification and they are running on CentOS Linux 5.0 kernel-2.6.18-128.1.6.el5. Each one comprises a 3GHz AMD Athlon 64 Dual Core processor, 2GB RAM and 1Gb Network interface. The Bridges are actually coded in C with raw sockets and open source libraries. For the pairing-based operation, we use PBC library and modify the given examples (PBC, n.d.). Besides, we use the Barreto's High-speed AES implementation in C (Krovetz, n.d.).

The average round trip delay in this small Ethernet network with custom Linux Bridges is *0.265ms* for normal ARP messages. On the other hand, our solution has a latency of *55.099ms*. We take both results from the average of 100 thousands ARP messages. We calculate the time inside host A, where the latency is measured by capturing the time interval between the ARP request transmission and corresponding ARP reply reception. We get a large difference, because here we focus on feasibility and functional capability, rather than optimized performance. Our key agreement is a one time setup for each pair in every 20 minutes because the operating systems generally expire the ARP cache in 20 minutes, so we think our solution is very effective and reasonable. Here, both bridges compute the same SCK, which will be passed to AES encryption operation for secure communication. The FPGA-based pairing implementation (Huang, Lee and Li, 2008) can dramatically reduce the latency because it speeds up the computation by 152 times compared to the software prototype and the operation time reaches μs (Junjie, 2007). For this reason, we plan to implement our solution on NetFPGA (NetFPGA, n.d.).

The KGC part can be easily provided with Secure Device Identity in real network and the secure communication using AES is already standardized, where the hardware based implementation gives 34Gbps throughput (Yang, Mishra and Karri, 2005). For this reason, we do not test them in our implementation.

4.3 Limitation

Our solution transmits the encrypted frame over the Bridged Ethernet Network, which prevents traffic analysis or intrusion detection, but we think the incorporation of the Network Interface Card based Firewalls (Friedman and Nagle, 2001) or Intrusion Detection system (Otey, Parthasarathy, Ghoting, Li, Narravula and Panda, 2003) can detect any intrusion at the IB and EB.

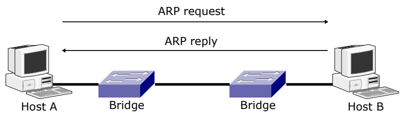

Figure 6: Our small platform to test the functionality.

5 RELATED WORK

Security in Layer-2 is an emerging technology. So there are very few literatures on this topic. Reference (Altunbasak, Krasser, Owen, Sokol, Grimminger and Huth, 2004) mainly discuss the IP and MAC address binding problems. Also they focus on Data Link layer security issues in IP over Ethernet networks and propose some extensions in MACsec frame format at Layer-3 devices to improve security in LANs (Altunbasak et al., 2005). On the other hand, Altunbasak and Owen propose a security inter-layering concept for future network architecture (Altunbasak and Owen, 2007). They argue that data link layer could utilize the secure namespace of upper layers instead of MAC addresses. They also point out two incompleteness: no available protocol for secure binding of IP and MAC, and no available Layer-2 secure identifier. But the latter could be achieved by Secure Device Identity. Also binding IP and MAC address can be done by DHCP snooping technique (DHCP snooping, n.d.), but it is not an embedded solution. However, their work focuses only on future network and does not consider threats in current standards. Switch-based single secure domain (Wahid, 2010a) depends on (IEEE P802.1AR, 2009) for safety of its global domain, so any failure of the identity puts total infrastructure in danger. Moreover, this approach is only applicable in internal network.

Usage of the additional part of ARP messages to convey new protocol information is practiced in (Bruschi, Ornaghi and Rosti, 2003; Lootah, Enck and McDaniel, 2005; Jerschow, Lochert, Scheuermann and Mauve, 2008; Zúquete and Marques, 2006), but only Cryptographic Link Layer (CLL) (Jerschow, Lochert, Scheuermann and Mauve, 2008) and Secure LAN (SLAN) (Zúquete and Marques, 2006) use it for LAN security. CLL uses several cryptographic mechanisms (e.g., MD5, Twofish, SHA-1, AES, RSA) to provide link security. In our opinion, such solution is not a practical approach for real-life systems because of its large number of modules. Our system only adds ID-based scheme to the current standards, thus enforces little modification. On the other hand, the source and destination need to communicate with the Key Distribution Center (KDC) to securely transmit a frame in SLAN and such extra communication overhead becomes a drawback in large network. Obviously our solution does not require any extra message exchange. Furthermore, (Park, Lee and Rhee, 2007; Park and Rhee, 2009) proposes ID-based on-demand authenticated key agreement protocol to discover authenticated route in wireless networks, but their solution assumes that the source contains the destination ID, therefore, they can avoid the computationally expensive ID-based signature scheme. We cannot avoid the signature scheme at the first phase, because we do not know the ID (LDevID) of the destination.

6 CONCLUDING REMARKS AND FUTURE WORK

Our paper has presented an efficient mechanism to overcome the insecurities induced by hop-by-hop and group security features of MACsec. We have demonstrated that it is possible to generate on demand pairwise keys within Bridges to secure all real-time end-to-end communications without new key agreement messages, therefore, our solution is capable of protecting private communication over multi-link with constant cryptographic computations. The feasibility and functional capability of our design has been validated with a software prototype deployed on a small multi-segment Ethernet Network. Although the speed is less than the line speed, we believe that the future hardware-based (NetFPGA) solution will be able to operate at a very high rate. We also hope that the ideas presented here may encourage the network security research community to address pure Layer-2 security problems which are at an initial stage.

REFERENCES

Altunbasak, H., Krasser, S., Owen, H. L., Grimminger, J., Huth, H-P. and Sokol, J. (2005). Securing Layer 2 in Local Area Networks. In *4th International Conference on Networking*. Reunion, France.

Vyncke, E. and Paggen, C. (2007). *LAN Switch Security: What Hackers Know About Your Switches*. Cisco Press, Indianapolis, 1 edition.

Furnell, S. (2004). Enemies within: the problem of insider attacks. *Computer Fraud & Security*, 7, 6–11. doi:10.1016/S1361-3723(04)00087-9

Bhandari, R., Grewal, K. and Jha, P. (2006). Scalable, High Speed Layer 2 security using IEEE 802.1AE/af. http://whitepapers.techrepublic.com.com /abstract.aspx?docid=267180

Kolodgy, C. J., Pintal, G. and Burke, B. E. (2008). *Oracle Database Security: Preventing Enterprise Data Leaks at the Source*. Retrieved December 11, 2009, from http://www.oracle.com/corporate/analyst/reports/infrastructure/sec/209752.pdf

IEEE Std 802.1AE – 2006. *IEEE Standard for Local and Metropolitan Area Networks- Media Access Control (MAC) Security*.

IEEE P802.1AR/D2.0 - 2009. *IEEE draft standard for Local and Metropolitan Area Networks- Secure Device Identity*.

IEEE P802.1X-2010 - Revision of 802.1X-2004. *IEEE draft standard for Local and Metropolitan Area Networks- Port Based Network Access Control*.

Hess, F. (2003). Efficient Identity Based Signature Schemes Based on Pairings. In *Nyberg, K., Heys, H. (eds.) SAC 2002. LNCS, vol. 2595, pp. 310-324*. Springer, Heidelberg.

Yuan, Q. and Li, S. (2005). A New Efficient ID-Based Authenticated Key Agreement Protocol. In *Cryptology ePrint Archive: Report 2005/309*.

Boneh, D. and Franklin, M. (2001). Identity-based encryption from the Weil pairing. In *Kilian, J. (ed.) CRYPTO 2001, LNCS, vol. 2139, pp. 213-229*. Springer, Heidelberg.

Frey, G. and Ruck, H. G. (1994). A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves. In *Mathematics of Computation, vol. 62, no. 206, 865-874*.

Ryu, E., Yoon, E. and Yoo, K. (2004). An efficient ID-based authenticated key agreement protocol from pairings. In *Mitrou, N., et al. (eds.) Networking 2004, LNCS, vol. 3042, pp. 1458-1463*. Springer, Heidelberg.

Park, Y., Lee, W. Y. and Rhee, K. H. (2007). Authenticated On-Demand Ad Hoc Routing Protocol without Pre-shared Key Distribution. In *Proceedings of the 2007 ECSIS Symposium on Bio-inspired, Learning, and Intelligent Systems for Security, Edinburg, pp. 41–46*. Edinburgh, UK.

Dutta, R., Barua, R. and Sarkar, P. (2004). Pairing-Based Cryptographic Protocols : A Survey. In *Cryptology ePrint Archive: Report 2004/064*.

Arkin, O. and Anderson, J. (2003). EtherLeak: Ethernet frame padding information leakage. http://www.packetstormsecurity.org/advisories/atstake /atstake_etherleak_report.pdf

Bruschi, D., Ornaghi, A. and Rosti, E. (2003). S-ARP: a Secure Address Resolution Protocol. In *Proceedings of the 19th Annual Computer Security Applications Conference, pp. 66*. Las Vegas, USA.

Lynn, B. (2007). *On The Implementation of Pairing-based Cryptosystems*. Ph.D. Thesis. Stanford University. Retrieved March 29, 2009, from http://crypto.stanford .edu/pbc/thesis.pdf

Lynn, B. (n.d.). *PBC Library Manual 0.5.0*. Retrieved March 29, 2009, from http://crypto.stanford.edu/pbc /manual.pdf

Eastlake, D. (2008). RFC: 5342 IANA Considerations and IETF Protocol Usage for IEEE 802 Parameters.

Malkin, G. (1995). RFC: 1868 ARP Extension - UNARP.

Trusted Computing Group. (2003). Retrieved August 13, 2009, from http://en.wikipedia.org/wiki/Trusted_ Computing_Group

PBC Library. (n.d.). Retrieved March 29, 2009, from http:// crypto.stanford.edu/pbc/

Krovetz, T. (n.d.). Fast Cryptography. Retrieved December 16, 2009, from http://fastcrypto.org/

Huang, J., Lee, J. and Li, H. (2008). A Fast FPGA Implementation of Tate Pairing in Cryptography over Binary Field. In *The 2008 International Conference on Security and Management, pp. 3–9*. Las Vegas, USA.

NetFPGA. (n.d.). Retrieved January 29, 2010, from http://www.netfpga.org/

Junjie, J. (2007). *Bilinear Pairing (Eta_T Pairing) IP Core*. Retrieved December 08, 2009, from http://www.cs. cityu.edu.hk/∼ecc/doc/etat_datasheet_v2.pdf

Yang, B., Mishra, S. and Karri, R. (2005). A High Speed Architecture for Galois/Counter Mode of Operation (GCM). In *Cryptology ePrint Archive: Report 2005/146 (2005)*.

Friedman, D. and Nagle, D. (2001). Building Firewalls with Intelligent Network Interface Cards. Technical report CMUCS00173, Carnegie Mellon University.

Otey, M., Parthasarathy, S., Ghoting, A., Li, G., Narravula, S. and Panda D. (2003). Towards NIC-based Intrusion Detection. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 723 – 728*. Washington, USA.

Altunbasak, H., Krasser, S., Owen, H. L., Grimminger, J., Huth, H-P. and Sokol, J. (2004). Addressing the Weak Link Between Layer 2 and Layer 3 in the Internet Architecture. In *The 29th Annual IEEE Conference on Local Computer Networks (LCN), pp. 417–418*. Florida, USA.

Altunbasak, H. and Owen, H. L. (2007). Security Interlayering for Evolving and Future Network Architectures. In *Proceedings of IEEE SoutheastCon 2007, pp. 615–620*. Virginia, USA.

Understanding and Configuring DHCP Snooping. *Cisco IOS Software Configuration GuideRelease 12.1 (12c) EW*. (n.d.). Retrieved January 06, 2010, from http: //www.cisco.com/en/US/docs/switches/lan/catalyst45 00/12.1/12ew/configuration/guide/dhcp.pdf

Jerschow, Y. I., Lochert, C., Scheuermann, B. and Mauve, M. (2008). CLL: A Cryptographic Link Layer for Local Area Networks. In *R. Ostrovsky, R. De Prisco, and I. Visconti (eds.) SCN 2008, LNCS, vol. 5229, pp. 21–38*. Springer, Heidelberg.

Lootah, W., Enck, W. and McDaniel, P. (2005). TARP: Ticket-based Address Resolution Protocol. In *21st Annual Computer Security Applications Conference, pp.106–116*.

Park, Y. and Rhee, K. H. (2009). An Authenticated On-Demand Routing Protocol with Key Exchange for Secure MANET. In *IEICE Trans. Inf. & Syst., vol. E92.D, no. 5, pp. 810–817*.

Zúquete, A. and Marques, H. (2006). A Security Architecture for Protecting LAN Interactions. In *S.K. Katsikas et al. (eds.) ISC 2006, LNCS. vol. 4176, pp. 311–326*. Springer, Heidelberg.

Wahid, K. F. (2010a). Maximizing Ethernet security by switch-based single secure domain. In *Proceedings of the 7th International Conference on Information Technology : New Generations*. Las Vegas, USA.

Wahid, K. F. (2010b). Rethinking the link security approach to manage large scale Ethernet network. In *Proceedings of the 17th IEEE Workshop on Local and Metropolitan Area Networks*. New Jersey, USA.

Federal Information Processing Standards Publication 197: Advanced Encryption Standard (AES). (2001). http:// www.csrc.nist.gov/publications/fips/fips197/fips-197. pdf