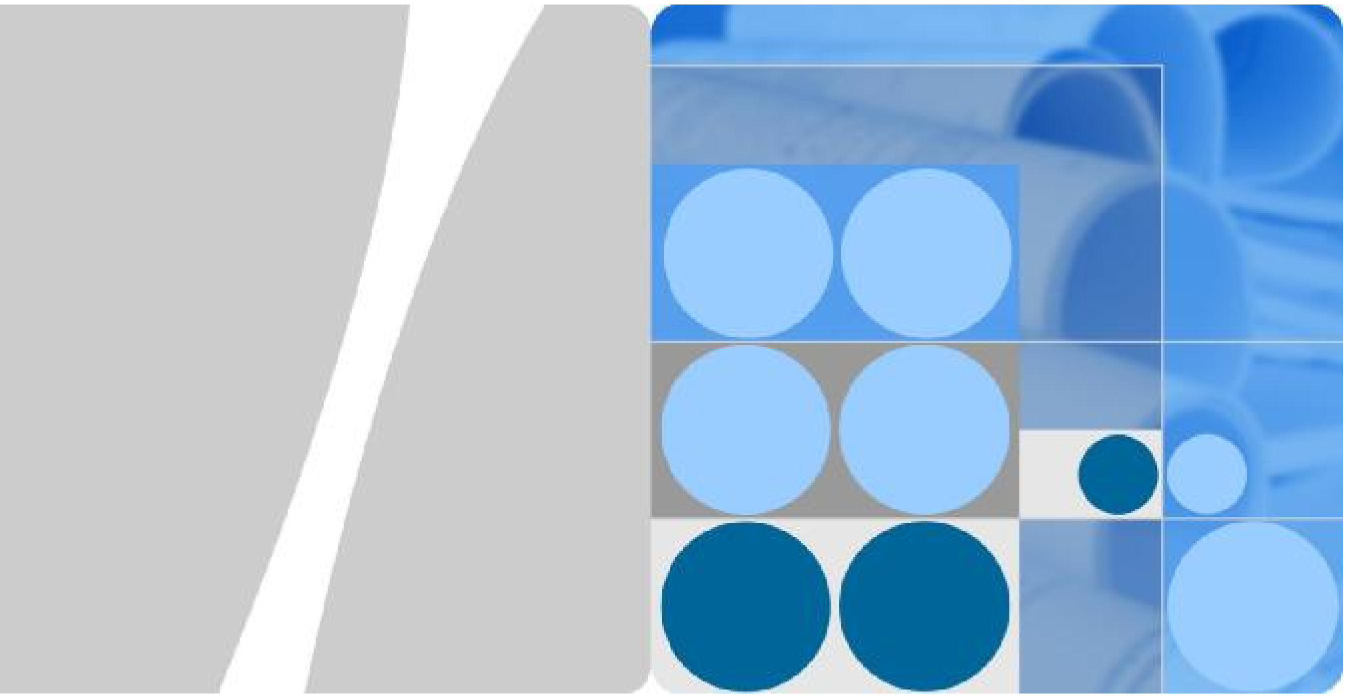


S Series Switches

MACsec Technology White Paper

Issue	**1.0**
Date	**2016-03-25**

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Huawei Technologies Co., Ltd.

Address: Huawei Industrial Base
Bantian, Longgang
Shenzhen 518129
People's Republic of China

Website: http://www.huawei.com

Email: support@huawei.com

Tel: 0755-28560000 4008302118

Fax: 0755-28560111

MACsec Technology White Paper

Keywords:

MACsec, 802.1AE

Abstract:

The MACsec (802.1AE) protocol secures communication at the Media Access Control (MAC) layer over a Local Area Network (LAN). The MACsec protocol provides data integrity, data origin authenticity, confidentiality, and replay protection.

Abbreviations:

Abbreviation	Full Name
AN	Association Number
CA	secure Connectivity Association
CAK	secure Connectivity Association Key
CKN	secure Connectivity Association Key Name
ICK	ICV Key
ICV	Integrity Check Value
KaY	MAC Security Key Agreement Entity
KDF	Key Derivation Function
KEK	Key Encrypting Key
KI	Key Identifier
KN	Key Number
MACsec	Media Access Control (MAC) Security
MI	Member Identifier
MKA	MACsec Key Agreement protocol
MKPDU	MACsec Key Agreement Protocol Data Unit
MN	Message Number
MPDU	MACsec or MAC Protocol Data Unit
MSDU	MAC Service Data Unit
PN	Packet Number
PSK	Pre-shared Key
SA	Secure Association

Abbreviation	Full Name
SAI	Secure Association Identifier
SAK	Secure Association Key
SC	Secure Channel
SCI	Secure Channel Identifier
SecY	MAC Security Entity
SecTAG	MAC Security TAG

Contents

Figures

1 MACsec Overview

Media Access Control Security (MACsec) is an IEEE 802 standard specifying how to secure data communication over a Local Area Network (LAN). By providing data integrity check, data origin authenticity, confidentiality, and replay protection services, MACsec ensures that data is securely sent and received at the MAC layer.

MACsec involves two standards: IEEE802.1AE and 802.1X. IEEE802.1AE-2006 defines the frame format for data encapsulation, encryption, and authentication. 802.1X-2010 defines MACsec Key Agreement (MKA), a key management protocol that provides a key generation mechanism in peer-to-peer or group mode. MKA packets use the same format as 802.1X packets, but MKA extends and optimizes the original 802.1X protocol. MACsec uses keys derived from MKA negotiation to encrypt data of authenticated users and perform integrity check on the data, preventing interfaces from processing packets of unauthenticated devices or packets modified by unauthenticated devices.

MACsec is a supplement, instead of a substitute, for existing end-to-end Layer 3 security technologies such as IPSec and Transport Layer Security (TLS). MACsec cryptographically protects frames on a hop-by-hop basis at Layer 2 on a LAN. It is applicable to scenarios requiring high data confidentiality, such as the networks of governments, militaries, and the financial industry. For example, when two switches on a LAN exchange data through an optical transmission device, MACsec ensures transmission security using cryptographic technology.

2 Implementation

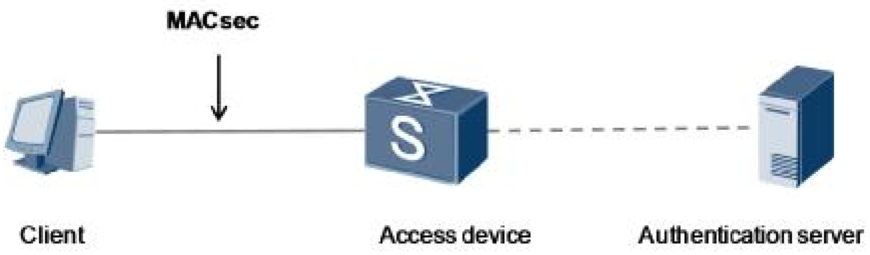
2.1 Typical MACsec Networking Modes

There are two commonly-used MACsec networking modes: point-to-point connection between a host and network device (point-to-point access LAN secured with MACsec) and point-to-point connection between network devices (point-to-point LAN secured with MACsec).

2.1.1 Point-to-Point Connection Between a Host and Network Device

This networking mode protects data frame transmission between a client (host) and the network device. Figure 2-1 shows the networking.

Figure 2-1 Point-to-point connection between a host and network device



There are three elements in the networking diagram:

- Client

  The client is a user terminal requesting access to the LAN. It is authenticated by the access device and the authentication server. The client negotiates the MACsec key with the access device and encrypts packets.

- Access device

  The access device controls client access. It communicates with the authentication server to implement 802.1X authentication on the client, negotiates the MACsec key with the client, and encrypts packets.

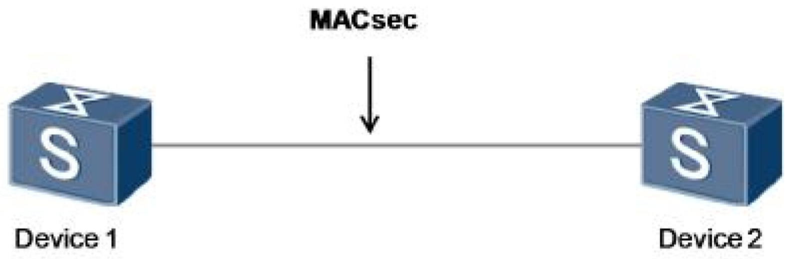
- Authentication server

  The authentication server is usually an AAA server that performs RADIUS authentication, authorization, and accounting on the client. After a client is authenticated, the authentication server distributes keys to the client and access device for subsequent MACsec key negotiation.

2.1.2 Point-to-Point Connection Between Two Network Devices

This networking mode protects data frame transmission between two devices. Figure 2-2 shows the networking. In this networking mode, devices are not differentiated as the client or access device, and no authentication server is needed. A Secure Connectivity Association Key (CAK) can be configured for two connected devices on the command line interface to implement MACsec key negotiation and packet encryption.

Figure 2-2 Point-to-point connection between two network devices



NOTE

Currently, Huawei switches support MACsec implementation on a point-to-point connection between two network devices, but not on point-to-point connection between a host and network device. This document describes how MACsec is implemented on a point-to-point connection between two network devices.

2.2 MACsec Concepts

- MKA

  The MACsec Key Agreement (MKA) protocol is used to negotiate keys used by MACsec.

- CA

  A secure Connectivity Association (CA) is a set of MACsec-enabled ports fully connected over a LAN. CAs are created and maintained by MKA. CA members are called CA participants.

- CAK

  A secure Connectivity Association Key (CAK) is not directly used to encrypt data packets, but is used in conjunction with other parameters to derive keys for encrypting data packets. CAKs can be delivered during 802.1X authentication or configured by users.

- CKN

  Secure Connectivity Association Key Name (CKN) is the name of a CAK.

- SC

  A Secure Channel (SC) is used to transmit data between CA participants at the MAC layer. Each SC provides one-to-one or one-to-many communication. The two links involved in one-to-one communication are considered $SC_{(a)}$ and $SC_{(b)}$, respectively.

- SCI

  A Secure Channel Identifier (SCI) consists of a 6-byte MAC address and a 2-byte port identifier. An SCI uniquely identifies a secure channel in the system.

- SA

  A Secure Association (SA) is a collection of security parameters of an SC, including a cipher suite and a key for data integrity check. An SC can have multiple SAs, and each SA has a unique key called Secure Association Key (SAK).

- SAK

  An SAK is derived from a CAK using a certain algorithm and encrypts data transmitted through an SC. The MKA protocol limits the number of frames that can be protected with a single SAK. When the Packet Numbers (PNs) in an SA are exhausted, the corresponding SAK will be refreshed. For example, the SAK on a 10-Gbps link is refreshed at a minimum interval of 4.8 minutes.

- ICV

  An Integrity Check Value (ICV) is generated by a packet sender based on a certain algorithm and placed at the end of the packet. The packet receiver uses the same algorithm to calculate an ICV and compares that ICV with the ICV in the packet. If the two ICVs are identical, the packet is not tampered with. If the two ICVs are different, the receiver discards the packet. This check method ensures data integrity.

- ICK

  An ICV key (ICK) is derived from a CAK using a certain algorithm. It is used only to calculate the ICV of the MKA packet.

- KEK

  A (Key Encrypting Key) is derived from a CAK based on a certain algorithm and used to encrypt an SAK. An SAK is sent in an MKA packet to participants of a CA, preventing the SAK from being intercepted during transmission.

- PN

  A Packet Number (PN) corresponds to a field in the Sec TAG. The sender increases the PN by one for replay protection. When the PN in an SA reaches 0xFFFFFFFF, the corresponding SAK will be refreshed. For example, the PN on a 10GE link is exhausted at a minimum interval of 4.8 minutes. When the PN is about to be exhausted, that is, when the PN reaches 0xC0000000, the SAK is refreshed.

- Key Server

  The Key Server is an MKA entity that determines the cipher suite and distributes keys.

- Supplicant

  The Supplicant is an MKA entity different from the Key Server.

- Confidentiality Offset

  The confidentiality offset for MKA is 0, 30, or 50 bytes. To allow certain applications to identify IPv4 or IPv6 packet header (for example, Eth-trunk uses the hash algorithm to load-balance traffic among member interfaces based on the IP address in an IP packet header), the number of bytes specified by the confidentiality offset after the Sec TAG is not encrypted.

- MACsec mode

  MACsec mode indicates the encryption mode, which can be **None** (default), **Normal**, or **Integrity-only**.

  - **None**: no encryption, integrity check, or encapsulation
  - **Normal**: encryption and integrity check
  - **Integrity-only**: integrity check only and no encryption

2.3 Implementation

MACsec implementation is comprised of three stages: CAK configuration, MKA key negotiation, and MACsec encryption and decryption.

Figure 2-3 shows how MACsec is implemented on a point-to-point connection between two network devices. The process is as follows: The network administrator configures the same CAK on the command-line interface (CLI) on both devices. The two devices elect a Key Server using the MKA protocol. The Key Server determines the cipher suite, generates an SAK based on the CAK and other parameters using a certain encryption algorithm, and sends the SAK to the peer device. The two devices then use the SAK to encrypt and decrypt MACsec packets.

Figure 2-3 MACsec implementation on a point-to-point connection between two network devices

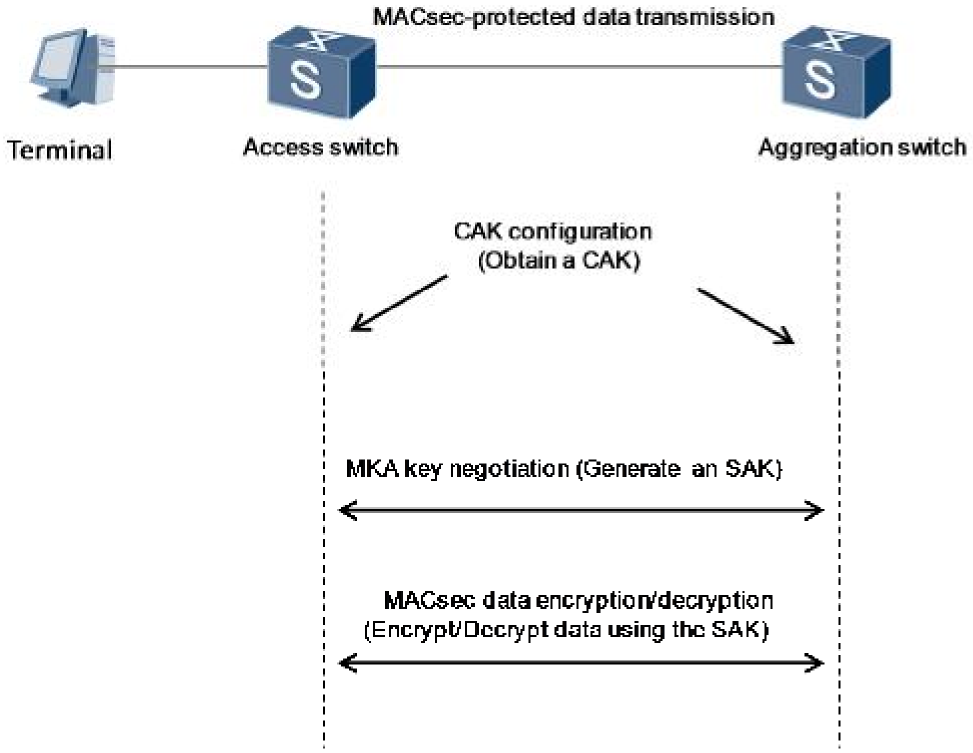
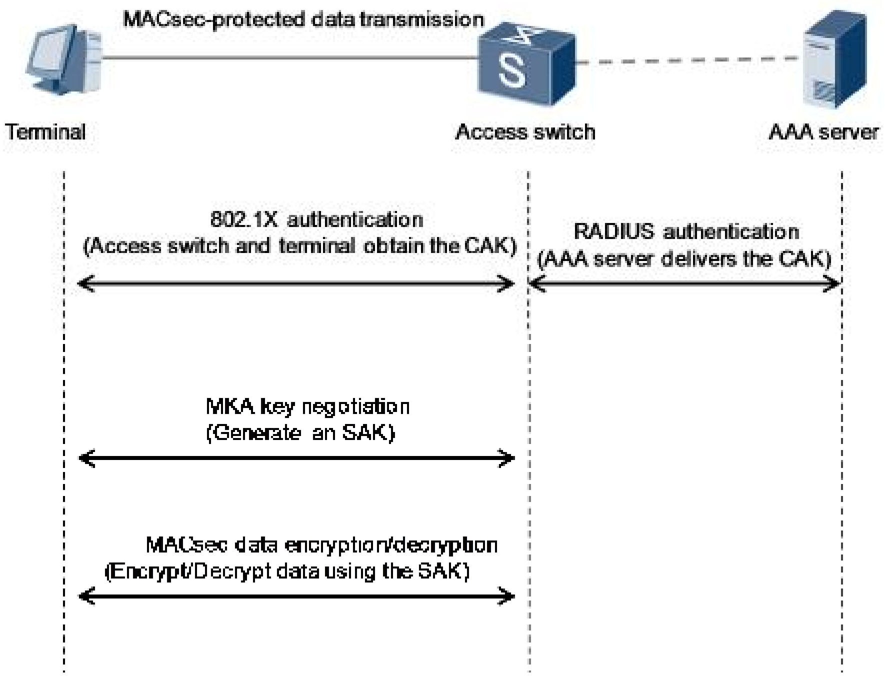

MACsec-protected data transmission
S
S
Terminal
Access switch
Aggregation switch
CAK configuration
(Obtain a CAK)
MKA key negotiation (Generate an SAK)
MACsec data encryption/decryption
(Encrypt/Decrypt data using the SAK)

Figure 2-4 shows how MACsec is implemented on a point-to-point connection between a host and network device. The process is as follows: The access switch performs 802.1X authentication on the terminal. After the terminal is authenticated, the AAA server delivers the CAK to the access switch and terminal. The access switch and terminal then use the CAK to encrypt and decrypt MACsec packets.

Currently, Huawei switches do not support MACsec implementation on a point-to-point connection between a host and network device.

Figure 2-4 MACsec implementation on a point-to-point connection between a host and network device

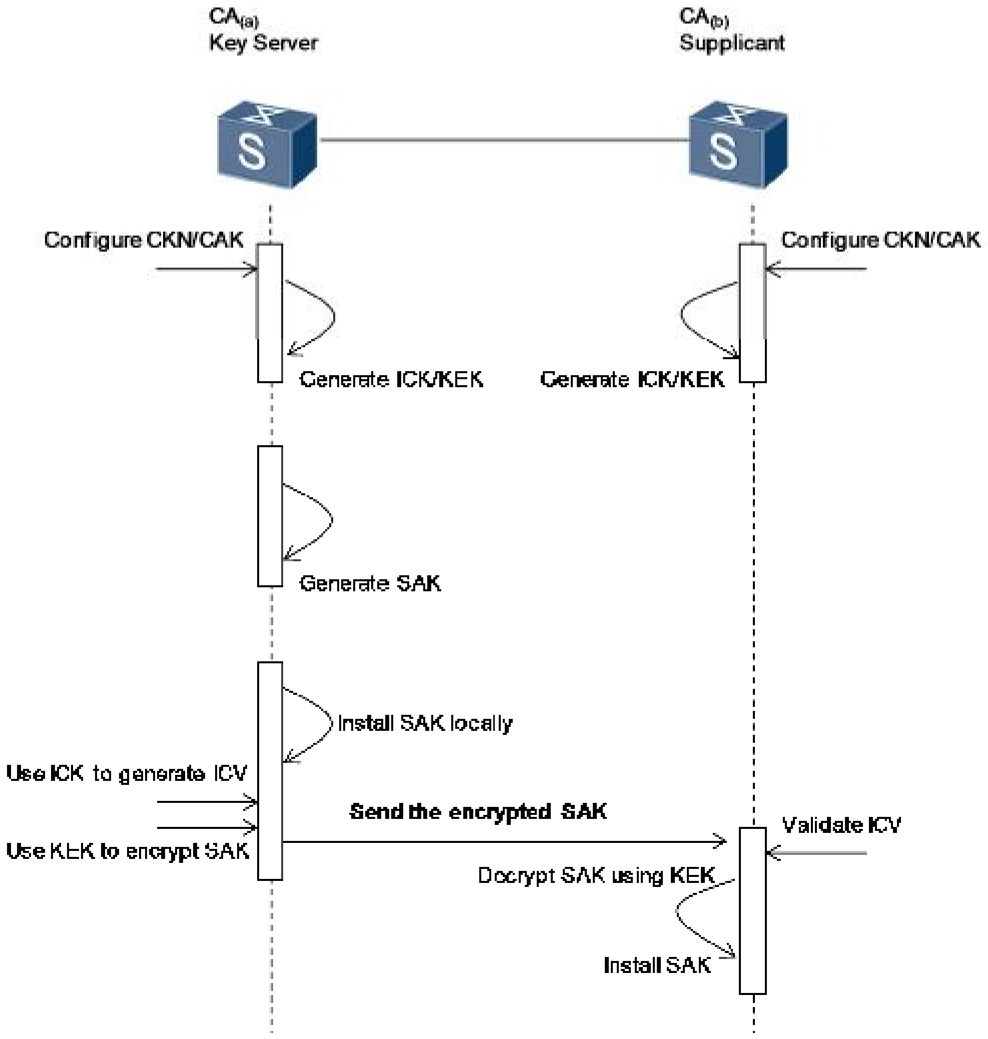

MACsec-protected data transmission
Terminal
Access switch
AAA server
802.1X authentication
(Access switch and terminal obtain the CAK)
RADIUS authentication
(AAA server delivers the CAK)
MKA key negotiation
(Generate an SAK)
MACsec data encryption/decryption
(Encrypt/Decrypt data using the SAK)

2.4 MACsec Key System

2.4.1 Key System Architecture

Figure 2-5 shows the process for the MACsec key system to generate an SAK based on a static CAK.

Figure 2-5 MACsec key system using a static CAK



The process is as follows:

2. The CA members ($CA_{(a)}$ and $CA_{(b)}$) generate the same Integrity Check Value Key (ICK) and Key Encrypting Key (KEK) based on the same CKN and CAK configured by a user. (The ICK and KEK are used to encrypt the SAK, preventing the SAK from being intercepted during transmission.)
3. The Key Server ($CA_{(a)}$) generates an SAK based on the CKN and CAK. Then the Key Server installs the SAK locally, so that the SAK can be used to encrypt and decrypt the packets sent and received by the Key Server.

4. The Key Server encrypts the SAK using the KEK, uses the ICK to generate ICVs (used to check packet integrity) based on a certain algorithm, and suffixes ICVs to an MKA packet. The Key Server then sends the encrypted SAK in the MKA packet to the Supplicant ($CA_{(b)}$).
5. After the Supplicant receives the MKA packet, it searches for the CAK and ICK by the CKN. If the CAK and ICK are not found, the Supplicant considers that the packet is not from a member of the same CA and discards the packet accordingly. If the CAK and ICK are found, the Supplicant performs ICV calculation on the MKA packet body to obtain ICVc. The Supplicant then compares ICVc and ICVs in the packet. If they are different, the packet has been tampered with (the CKN and ICVs in the packet are not encrypted using KEK).
6. After ICV validation, the Supplicant decrypts the packet using the KEK and obtains the SAK. The Supplicant then installs the SAK, so that the SAK can be used to encrypt and decrypt the packets sent and received by the Supplicant.

2.4.2 Key Algorithms

- KDF

  A Key Derivation Function (KDF) uses a Pseudorandom Function (PRF) to generate keys.

  To derive a 128-bit key, the KDF uses AES-CMAC-128; to derive a 256-bit key, the KDF uses AES-CMAC-256. Currently, Huawei switches support only 128-bit keys, so PRF AES-CMAC-128 is used.

  The KDF is described as follows:

  Output ← KDF (Key, Label, Context, Length) where

  Input:

  - **Key**: indicates a 128-bit or 256-bit key used to derive another key. Choose **128bit**.
  - **Label**: indicates a character string that specifies the name and purpose of the key derived by the KDF.
  - **Context**: indicates a bit string that provides context to identify the derived key.
  - **Length**: indicates a two-byte integer specifying the length (in bits) of the key derived by the KDF. The length of the derived key may be 128 bits or 256 bits. Choose 128 bits.
  - **Output**: indicates the key derived by the KDF, the length of which is specified by the **Length** parameter.

  The KDF generates the SAK, KEK, ICK, and ICV based on the CKN and CAK configured by a user.

- KEK derivation algorithm

  A KEK is generated based on the CKN and CAK configured by a user and used to encrypt or decrypt the SAK sent from the Key Server to the Supplicant.

  Algorithm to derive a KEK:

  KEK = KDF(Key, Label, Keyid, KEKLength)

  Where

  - **Key** = CAK
  - **Label** = "IEEE8021 KEK"
  - **Keyid** = the first 16 octets of the CKN, with null octets appended to pad to 16 octets if necessary
  - **KEKLength** = 128

- ICK derivation algorithm

  An ICK is generated based on the CKN and CAK configured by a user and used to check the integrity of MKA packets. ICKs are not transmitted via MKA packets, as the same CKN and CAK have been configured for the MKA packet sender and receiver.

  Algorithm to derive an ICK:

ICK = KDF(Key, Label, Keyid, ICKLength)

Where

- **Key** = CAK
- **Label** = "IEEE8021 ICK"
- **Keyid** = the first 16 octets of the CKN, with null octets appended to pad to 16 octets if necessary
- **ICKLength** = 128

- ICV derivation algorithm

Each transmitted MKA packet carries a 128-bit (16-byte) ICV to ensure data integrity.

The ICV is generated by the AES-CMAC algorithm based on an ICK as follows:

ICV = AES-CMAC(ICK, M, 128)

M = DA + SA + (MSDU – ICV)

Where

- **DA**: indicates the destination MAC address in an MKA packet.
- **SA**: indicates the source MAC address in an MKA packet.
- **MSDU**: indicates the MAC service data unit, that is, the service data unit in the original packet.

- SAK derivation algorithm

An SAK is generated by the Key Server based on the CKN and CAK configured by a user. An SAK is used to encrypt and decrypt data packets, and needs to be encrypted in an MKA packet using a KEK.

Algorithm to derive an SAK:

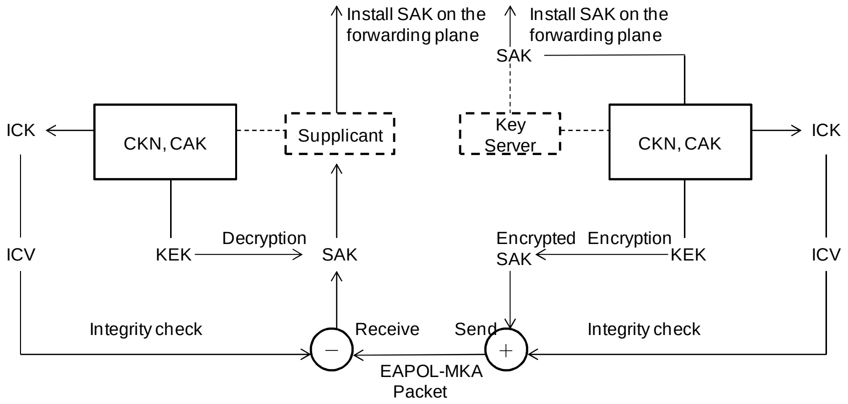
SAK = KDF(Key, Label, KS-nonce | MI-value list | KN, SAKlength)

Where

- **Key** = CAK
- **Label** = "IEEE8021 SAK"
- **KS-nonce** = a random number, the length of which (in bits) is specified by **SAKlength**. The **KS-nonce** value is obtained from the Key Server RNG each time an SAK is derived.
- **MI-value list** = a concatenation of MI values (in no particular order) from all live participants.
- **KN** = four-octet Key Number assigned by the Key Server as part of the KI
- **SAKlength** = 128

2.4.3 Key derivation relationship

Figure 2-6 Key derivation relationship

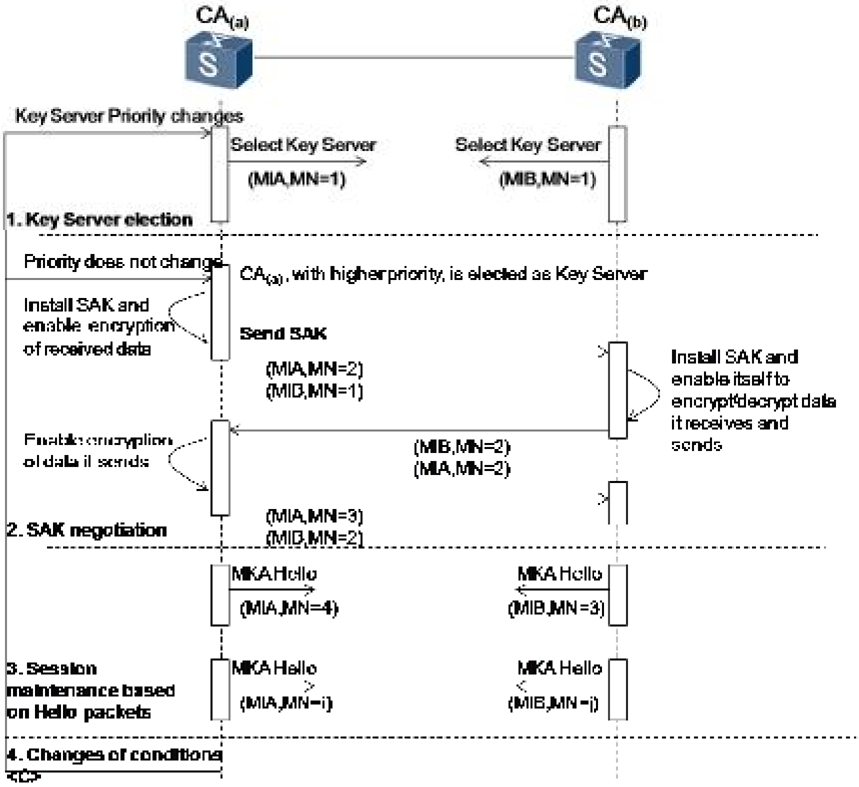


The KDF generates the SAK, KEK, ICK, and ICV based on the CKN and CAK configured by a user. Figure 2-6 shows the derivation relationship between them.

The CKN and CAK are configured on both the Key Server and Supplicant to generate the KEK, ICK, and ICV. The Key Server generates the SAK and installs it locally. The Key Server then encrypts the SAK using the KEK, suffixes the ICV to an MKA packet, and sends the encrypted SAK in the MKA packet to the Supplicant. When the Supplicant receives the MKA packet, it verifies the ICV. After the verification, the Supplicant decrypts and installs the SAK. The Key Server and Supplicant can then use the same SAK to encrypt and decrypt packets exchanged between them.

2.5 MKA Key Negotiation Process

Figure 2-7 MKA key negotiation process



The SAK negotiation process is as follows:

1. Key Server election

   CA members elect a Key Server among them by exchanging EAPOL-MKA packets. The Key Server selects the cipher suite, as well as manages and distributes keys. Each CA member can specify **Key Server Priority** for Key Server negotiation. A smaller **Key Server Priority** value indicates a higher priority. If two members have the same **Key Server Priority** value, the member with a smaller Secure Channel Identifier (SCI) will be the Key Server. (An SCI is comprised of a MAC address and a port ID.)

2. SAK negotiation

   The Key Server generates an SAK based on the CKN and CAK, and installs the SAK on the local forwarding plane. The Key Server then enables encryption of data it receives, and sends the SAK in an EAPOL-MKA packet to the Supplicant.

   The Supplicant decrypts the SAK, installs the SAK locally, and enables encryption and decryption of data it sends and receives. The Supplicant then sends a message to the Key Server, instructing it to encrypt data that it sends to the Supplicant.

   The Key Server sends a response message to the Supplicant to complete SAK negotiation, and the two members can normally send data to each other.

   Parameters used in the SAK negation process:

- **MI**: member identifier

- **MN**: message number
- (MIA, MN=i): The MN of the packet sent by the device with MI **A** is **i**. For example, (MIA,MN=2)(MIB,MN=1) indicates that when the device with MI **A** sends a packet whose MN is **2** to the device with MI **B**, it has received a negotiation packet whose MN is **1** from the device with MI **B**.

3. Session maintenance based on Hello packets

   After CA members negotiate an SAK, they can send data to each other. They send Hello packets to maintain the session. If one member cannot receive a Hello packet from the other member within the time specified by **MKA Life Time**, the two members will negotiate a new SAK. **MKA Hello Time** is usually 2s; **MKA Life Time** is usually 6s, but is configurable.

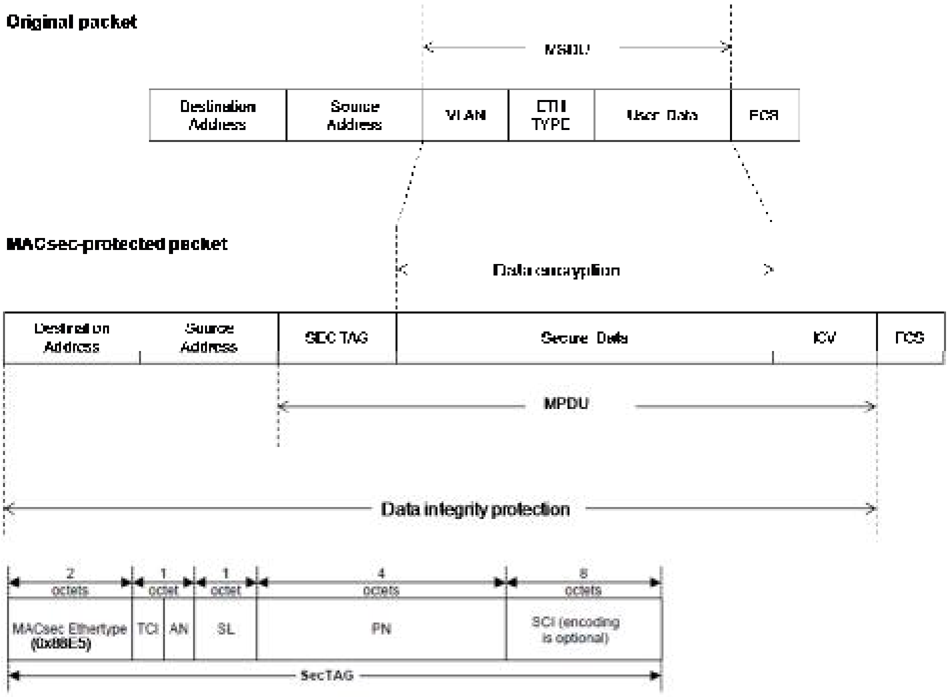
4. SAK switch or renegotiation due to changes in external conditions

   Changes in the external conditions (labeled as <C>) will result in SAK switch or renegotiation. For example:

- <C.1> If one member fails to receive a Hello packet from the other member within the time specified by **MKA Life Time**, SAK switch occurs (on only the Key Server).
- <C.2> When the PN on the forwarding plane of the Key Server is about to or have been exhausted, SAK switch occurs.
- <C.3> If the MI changes due to reasons other than MN rollover, SAK renegotiation occurs.
- <C.4> The PN of packets received by the Key Server from the Supplicant is about to or have been exhausted, SAK switch occurs.
- <C.5> If a user configures the CAK, Confidentiality Offset (configured only on the Key Server, which notifies other CA participants of it), and MACsec Mode, SAK switch occurs.
- <C.6> When the **Key Server Priority** setting is changed, a new Key Server needs to be elected, resulting in a new round of SAK negotiation.

2.6 MACsec Data Encryption and Decryption During Transmission

Figure 2-8 MACsec data encryption process



CA members start sending data to each other once the SAK is negotiated and installed. The sender of a packet uses the GCM-AES-128 cipher suite to encrypt the MSDU of the packet and SAK into Secure Data, and then calculates the source and destination MAC addresses, SEC TAG, Secure Data, and SAK to obtain the ICV. The sender then places the ICV at the end of the encrypted packet, and sends the encrypted packet to the peer end. The receiver of the packet uses the GCM-AES-128 cipher suite to decrypt the MACsec packet, uses an encryption algorithm to calculate the ICV, and compares that ICV with the ICV in the packet. If the two ICVs are identical, the packet has not been tampered with. The receiver then restores the original packet and forwards it over the LAN. If the two ICVs are different, the receiver discards the packet.

MACsec provides the replay protection function. When MACsec-encapsulated data packets are transmitted on the network, orders of the packets may be changed. MACsec replay protection mechanism allows a certain level of packet disorder, and the disordered packets can be accepted if their length is within the window size agreed upon by the two CA members. Packets will be discarded if their length exceeds the window size. The PN in the SEC TAG can be used for replay protection. The PN increases by one each time a packet is sent, and the receiver records the PN of the packet it receives. All packets will be accepted if their PNs are within the replay window size.

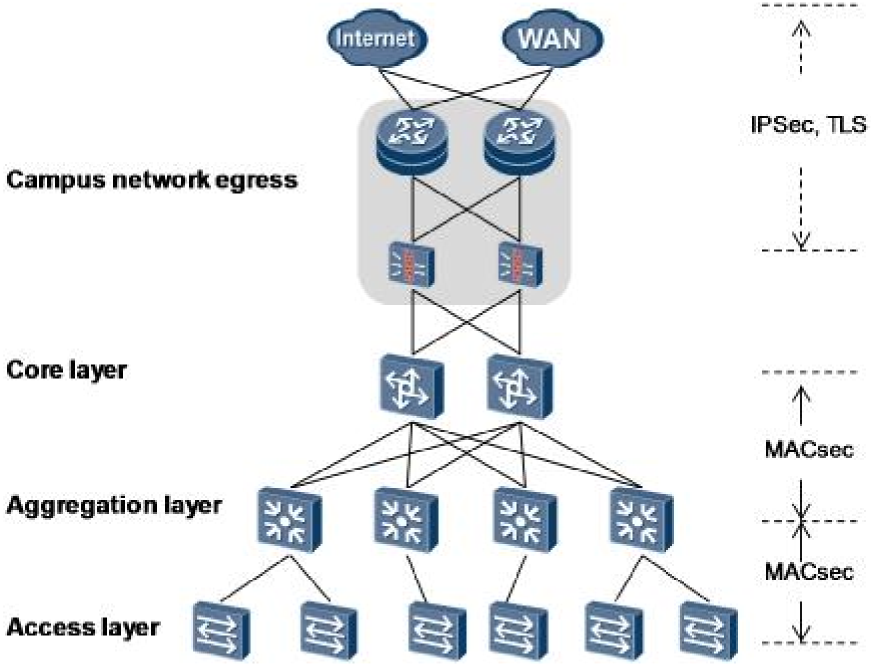
3 Typical Networking

3.1 Typical Networking of MACsec on a LAN

Networking Requirements

On networks that have high security requirements, such as networks of financial organizations, militaries, and governments, end-to-end security protection is required. When a large number of access devices are distributed in different places, management of these devices can be difficult and the entire network may face high security risks. In this scenario, MACsec can be configured on these devices to protect data during transmission, and authentication methods (such as 802.1X authentication) can be configured to control access of terminals. In addition, port isolation technology can be used on switches to block communication between unauthenticated terminals, and IPSec and TLS can be deployed at the campus network egress to ensure secure data transmission over the Internet or WAN. Figure 3-1 shows the typical networking for MACsec on a LAN.

Figure 3-1 Typical networking of MACsec on a LAN

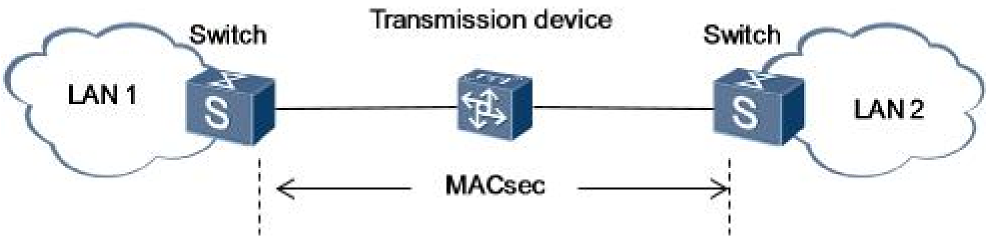


3.2 Typical Networking for MACsec When Transmission Devices Are Deployed

Networking Requirements

When transmission devices are deployed between two LANs to transparently transmit packets, MACsec can be configured on the outbound interfaces of the edge devices on both LANs to encrypt the packets, preventing the packets from being intercepted or tampered with. Figure 3-2 shows the typical network for MACsec when a transmission device is deployed.

Figure 3-2 Typical network for MACsec when a transmission device is deployed



4 Appendixes

4.1 MACsec Packet Format

Figure 4-1 shows the MACsec packet format.

Figure 4-1 MACsec packet format

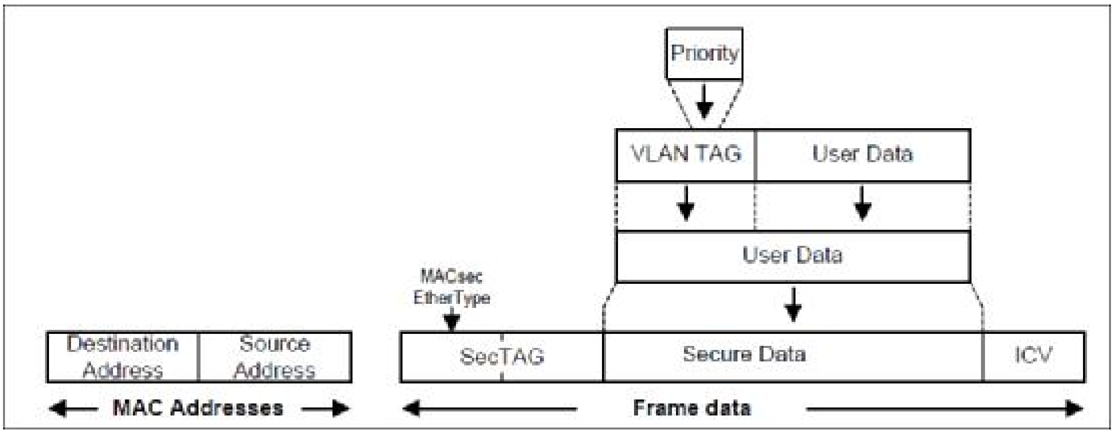


Figure 4-2 ICV format

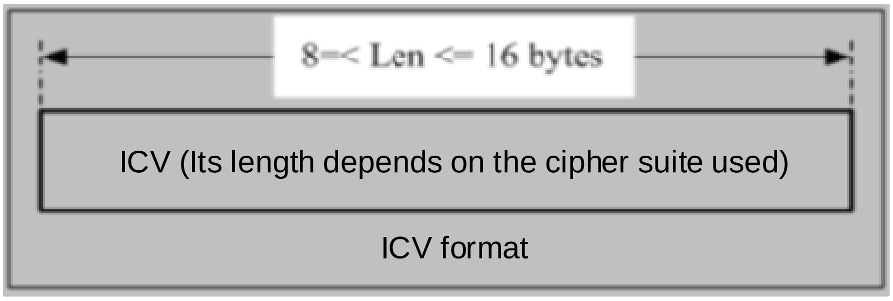
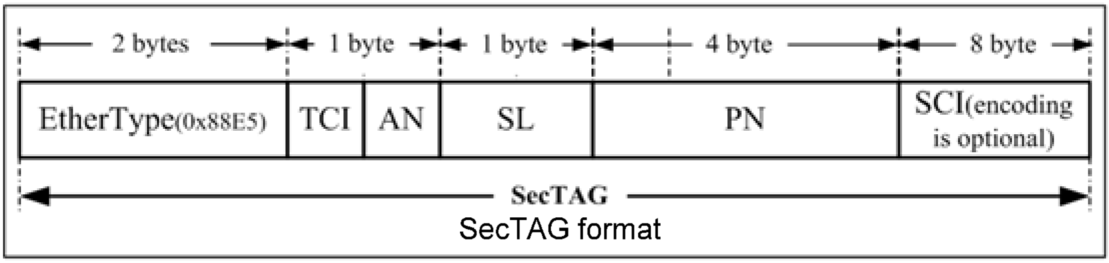


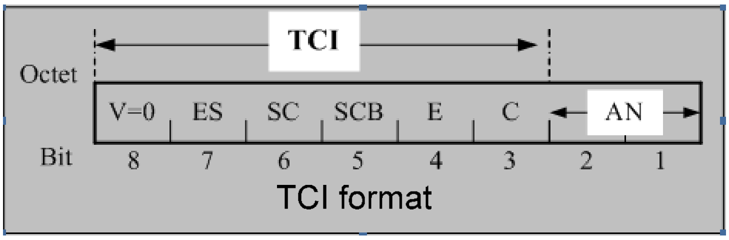
Figure 4-3 SEC TAG format



Field	Description
EtherType	Indicates the protocol encapsulated in the payload of a MACSec (802.1AE) Ethernet frame. The value is **0x88E5**.
TCI	Indicates tag control information about SEC TAG, which is 6-bit long.
AN	Indicates the association number, which identifies an SA in an SC. The AN is 2-bit long and an SC can have a maximum of four SAs.
SL	Indicates the short frame length, which is 6-bit long. When the SL is less than 48, it

Field	Description
	identifies the number of bytes in Secure Data, that is, the length between the last byte in the SEC TAG and the first byte of ICV. When the SL is 0, the frame is not a short one.
PN	Indicates the packet number, which identifies packets transmitted in an SA and prevents replay attacks.
SCI	Indicates the secure channel identifier. If the SC is 1 in the TCI, the SCI identifies the secure channel of a CA when there are two or more CA members. That is, the SCI is the network management identifier of the SecY that sends the frame. (In a point-to-point SC, the SC in the TCI does not need to be set to 1.)

Figure 4-4 TCI format



Field	Description
V	Indicates the version number. The value is 0.
ES	Indicates the end station. If MPDUs are sent by an end station and the first six bytes in the SCI are equal to the source MAC address, the ES can be set to 1. If the source MAC address is not used to identify the SCI, the ES is set to 0. If the ES is set to 1, the SC cannot be set to 1.
SC	Indicates the secure channel. If the SCI is explicitly encapsulated in the SEC TAG, the SC must be set to 1. If the SC is not set to 1, the SEC TAG must not contain the SCI.
SCB	Indicates the Ethernet passive optical network (EPON) broadcast identifier. The SCB is set to 1 in an MPDU only when the SC supports EPON Single Copy Broadcast (SCB) capability. If the SCB is set to 1, the SC cannot be set to 1. If the ES is 1 and SCB is 0, the Port Identifier in the SCI must be 00-01. If the SCB is 1, the Port Identifier in the SCI is 00-00, a value reserved for SCB capability.
E	Indicates whether User Data is encrypted. The value 1 indicates that User Data is encrypted, and the value 0 indicates that User Data is not encrypted. (If E is set to 1, User Data must have been modified, so C must also be set to 1.)
C	Stands for Changed Text. If C is set to 1, User Data has been modified by an encryption or verification algorithm. Otherwise, User Data has not been modified. That is, C indicates whether User Data is the same as Secure Data. When the default cipher suite (GCM-AES-128) is used to check data integrity, it does not modify User Data. That is, User Data is the same as Secure Data, and the ICV is 16 bytes long. Other cipher suites may modify User Data even they are supposed to provide only integrity check, and the ICV is not 16 bytes long when these cipher suites are used. C needs to be set to 1 when a cipher suite other than GCM-AES-128 is used.

Field	Description
	● E=1,C=1: indicates that a packet is encrypted, and User Data must have been modified. ● E=1,C=0: invalid value. ● E=0,C=0: indicates that a packet is not encrypted, and only integrity protection is implemented. The integrity check algorithm does not modify User Data. ● E=0,C=1: indicates that a packet is not encrypted, and only integrity protection is implemented, but the integrity check algorithm has modified User Data.

4.2 FAQs

1. Q: What networking modes does MACsec support?

   A: MACsec can be implemented only on a point-to-point connection between network devices, and cannot be implemented on a point-to-point connection between a host and network device.

2. Q: Will SAK switch cause packet loss?

   A: No.

3. Q: Will forwarding performance by affected if MACsec is enabled?

   A: After MACsec is enabled, packets can be forwarded at the line speed. After MACsec is enabled on a port, the port forwards packets that have the MACsec SEC TAG added. SEC TAG occupies bandwidth, so the bandwidth for transmitting user data decreases slightly when compared with the bandwidth required before MACsec is enabled. When sending data from a MACsec-incapable port to a MACsec-capable port, you need to allocate an appropriate bandwidth.

4. Q: Can QoS be implemented after MACsec is enabled?

   A: Yes. Perform QoS for traffic before the traffic is encrypted on an outbound interface, but after the traffic is decrypted on an inbound interface. Traffic shaping is performed based on the length of packets after they are encrypted.

5. Q: Can I enable MACsec on an aggregated port?

   A: MACsec is enabled on physical ports, so you can enable MACsec on each member port of an aggregation group.

4.3 Reference Standards

1. IEEE Std 802.1AE-2006, IEEE Standard for Local and metropolitan area networks Media Access Control (MAC) Security
2. IEEE Std 802.1X-2010, IEEE Standard for Local and metropolitan area networks Port-Based Network Access Control