Recomendaciones de Seguridad para MACsec

1. INTRODUCCIÓN A MACSEC

Actualmente, la mayor parte de las redes de área local (LAN) cableadas, tanto en el ámbito corporativo como en el doméstico, hacen uso de tecnología Ethernet. Esta tecnología ha ido evolucionando a lo largo del tiempo, proporcionando mayor velocidad de transmisión y permitiendo aumentar las distancias de los enlaces punto a punto. Sin embargo, las consideraciones de seguridad en esta tecnología se han mantenido en segundo plano.

MACsec, Media Access Control (MAC) Security, es un estándar definido en IEEE 802.1AE, que especifica un conjunto de protocolos que proporcionan medidas de seguridad para la protección de los datos en la capa de enlace (L2 OSI) de redes de área local (LAN) tipo Ethernet. La primera versión fue publicada en 2006. La versión actual es la **IEEE 802.1AE-2018**.

MACsec proporciona, principalmente, los siguientes servicios de seguridad (se recoge más detalle en el apartado 6):

- **Confidencialidad de datos**, mediante el uso del algoritmo de cifrado simétrico AES en modos AEAD (*Authenticated Encryption with Additional Data*), mayormente AES-GCM.
- **Protección de la integridad de los datos sin conexión,** gracias al uso de un código de verificación de integridad (ICV) exclusivo y único en cada paquete MACsec, y no relacionado con el resto de paquetes (*sin conexión*). Esto permite que cualquier cambio no autorizado sea detectado.
- **Autenticidad de datos en origen,** ya que con el uso del código de verificación ICV, se garantiza que el paquete MAC recibido ha sido enviado por quien se indica en el campo *Dirección MAC Origen* del paquete.
- **Protección anti-reenvíos (*anti-replay*)**, mediante números de secuencia añadidos al mensaje (campo PN, *Packet Number*).
- **Arquitectura segura *hop-by-hop***, gracias al cifrado y descifrado de la información en cada nodo MACsec por el que pasa el paquete.
- **Límite de retraso de recepción**, mediante el uso del parámetro *bounded time*, el cual suele tener, normalmente, un valor menor a 2 segundos. Este parámetro indica el tiempo límite para la recepción de un paquete desde su momento de envío. Sirve para descartar todos aquellos paquetes que tardan más del tiempo habitual en ser recibidos, indicando que han podido ser interceptados y modificados entre el origen y el destino.

La seguridad de las redes está enfocada, principalmente, en neutralizar ataques **externos.** Sin embargo, debido al cambio constante del ámbito empresarial y de la tecnología, una gran cantidad de amenazas han cambiado su punto de mira hacia la privacidad y los datos de la red **interna**. La seguridad de las redes detrás de los firewalls, se ha convertido en una prioridad.

MACsec permite proteger gran parte del tráfico interno, como LLDP, LACP, DHCP y ARP, junto con otros protocolos que no suelen protegerse. Tecnologías de capas superiores, como son IPsec y TLS, no pueden detectar ni prevenir ataques de capa de enlace, pero pueden complementarse con MACsec para conseguir una seguridad de red extremo a extremo.

2. FUNCIONAMIENTO DEL PROTOCOLO MACESC

2.1. DEFINICIONES Y CONCEPTOS

Para entender el desarrollo del protocolo, es fundamental la definición de una serie de conceptos que se utilizarán a lo largo de todo el documento.

a) Conceptos relacionados con protocolos:

MACsec: Protocolo de seguridad de la capa de enlace (L2 OSI), definido en la IEEE 802.1AE.

IEEE 802.1X: Estándar para el control de acceso a red, basado en puertos. Permite que solo los dispositivos autenticados y autorizados puedan utilizar los puntos de acceso a servicios LAN (como, por ejemplo, puertos de un *switch*).

MKA: *MACsec Key Agreement Protocol*. Protocolo de generación y distribución de claves definido en la revisión 2010 de IEEE 802.1X.

EAP: *Extensible Authentication Protocol*. Es un marco de autenticación (*authentication framework*) para redes LAN cableadas o inalámbricas. Se especifica en la RFC 3748.

EAP-TLS: *EAP-Transport Layer Security*. Método EAP especificado en la RFC 5216, que incluye soporte para derivación de claves y autenticación mutua basada en certificados, utilizando las capacidades de negociación protegida de *cipher suites*, la autenticación mutua y la gestión de claves del protocolo TLS.

EAPoL: *EAP over LANs*. Protocolo especificado en IEEE 802.1X, consistente en la encapsulación de EAP sobre redes LAN.

MPDU: *MACsec Protocol Data Units.*

MSDU: *MAC Service Data Units.*

MKPDU: *MACsec Key Agreement Protocol Data Units.*

b) Conceptos MACsec/MKA:

CA: *Secure Connectivity Association*. Es una asociación de seguridad establecida entre dos (2) o más dispositivos MACsec dentro de una red LAN o WAN. Esta asociación se establece y se mantiene a través de protocolos de acuerdo de claves (MKA).

SA: *Secure Association*. Es una relación de seguridad que proporciona garantías de seguridad a las tramas transmitidas desde un miembro de una CA, a otros miembros de la misma CA. Cada SA utiliza una clave secreta de cifrado (SAK) o un conjunto de ellas. El identificador de una SA se denomina **SAI** (*Secure Association Identifier*) y se compone del SCI (ver más abajo) concatenado con un número asignado a la SA (AN, *Association Number*).

SC: *Secure Channel.* Es una relación de seguridad que proporciona garantías de seguridad a las tramas transmitidas desde un miembro de una CA a otros miembros. Un canal seguro va estableciendo varias SAs para poder ir usando claves secretas (SAKs) recientes. El identificador

de un SC se denomina **SCI** (*Secure Channel Identifier*) y se compone de la dirección MAC del dispositivo junto con el identificador del puerto.

CAK: *Connectivity Association Key*. Es una clave de larga vida que se utilizará para derivar el material de claves necesario para MKA y MACsec (claves ICK, KEK y SAK). Una CAK se identifica con su identificador **CKN** (*Connectivity Association Key Name*). Todos los nodos de una CA (*Connectivity Association*) utilizan la misma CAK.

SAK: *Secure Association Key*. Es la clave que se deriva de la CAK utilizada para cifrar los datos (MSDU) de una sesión en una dirección entre dos dispositivos MACsec. Cada SAK está asociada a una SA.

ICK: *Integrity Check Key*. Es la clave derivada de la CAK utilizada para proteger la integridad y autenticidad de los paquetes MKA (MKPDUs).

KEK: *Key Encrypting Key*. Clave derivada de la CAK utilizada para el cifrado de las claves SAK para su distribución por parte del Key Server.

KS: *Key Server*. Estación MACsec, dentro de la CA, encargada de la selección de la *cipher suite* y de la generación y distribución de las claves SAK.

2.2. FUNCIONAMIENTO GLOBAL DEL PROTOCOLO

Dentro del funcionamiento global de MACsec, intervienen los protocolos IEEE 802.1X/EAP, EAPOL y MKA.

a) **IEEE802.1X/EAP** se utiliza para la **autenticación**, la **generación y distribución de una clave maestra** MSK (*Master Session Key*). De esta clave maestra posteriormente se derivará la **CAK** (*Connectivity Association Key*). Las tramas que se intercambian para estas negociaciones, se encapsulan con el protocolo EAPOL.

b) **MKA** (*MACsec Key Agreement Protocol*) se utiliza, principalmente, para:

- Descubrir miembros de una CA.
- Confirmar la posesión mutua de una CAK y, por tanto, probar una autenticación anterior.
- Derivar otras claves a partir de la CAK: la clave ICK que será usada para la protección de la integridad y autenticidad de los paquetes MKA, y la clave KEK que será usada para el cifrado de la clave SAK (clave final de cifrado de datos).
- Acordar la cipher suite y derivar y distribuir de forma segura las claves SAK utilizadas por MACsec para el cifrado de los datos (MSDU).
- Asegurar que las tramas enviadas no tienen retardo, mediante el uso conjunto de los parámetros MI (*Member Identifier*) y MN (*Message Number*).

El funcionamiento tanto de MACsec como de 802.1X y MKA es complejo. Se recomienda consultar las especificaciones IEEE correspondientes para mayor detalle. En este documento se hace un breve resumen de los aspectos más importantes para entender el funcionamiento general de los protocolos, que se resumen en el siguiente diagrama.

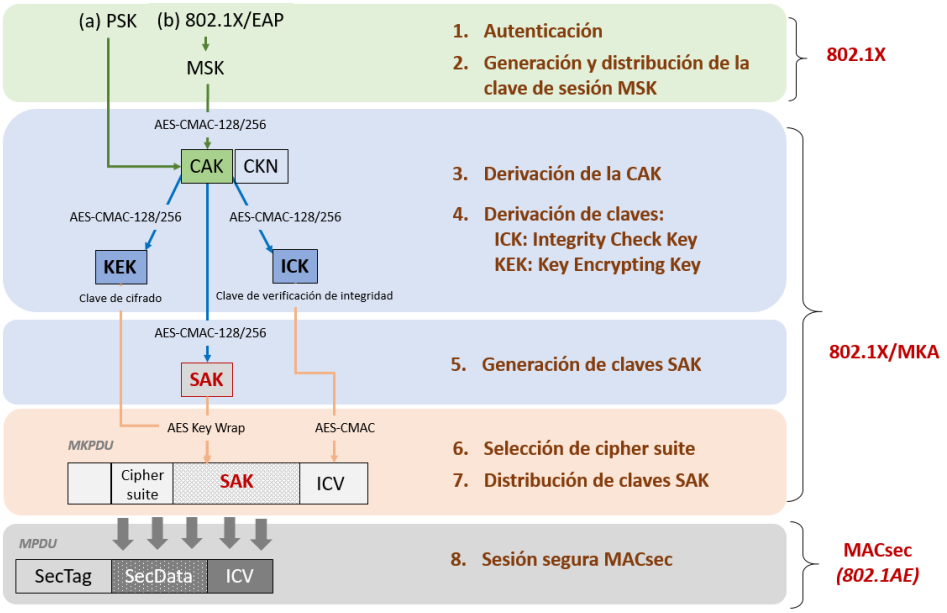

Figura 1. Diagrama de fases generales de MACsec

2.3. AUTENTICACIÓN Y DERIVACIÓN DE CLAVE CAK

El primer paso para el establecimiento de una sesión MACsec es la autenticación de los dispositivos que formarán parte de la CA, y la obtención de la **clave CAK** *(Secure Connectivity Association Key)*. Todos los dispositivos que forman parte de una *Asociación de Conectividad Segura* (**CA**), comparten la clave CAK. De esta clave se derivará el material de claves.

Para ello se pueden utilizar, principalmente, dos (2) métodos: "Claves pre-compartidas" o IEEE 802.1X/EAP.

a) "Claves Pre-compartidas"

Este método es típicamente usado en enlaces *switch-to-switch*. El usuario debe introducir de forma manual en cada dispositivo lo que se asemeja a una clave pre-compartida (PSK), que en realidad se trata de la clave CAK junto con su identificador CKN.

La verificación de que los dispositivos disponen de la misma CAK representa la autenticación mutua.

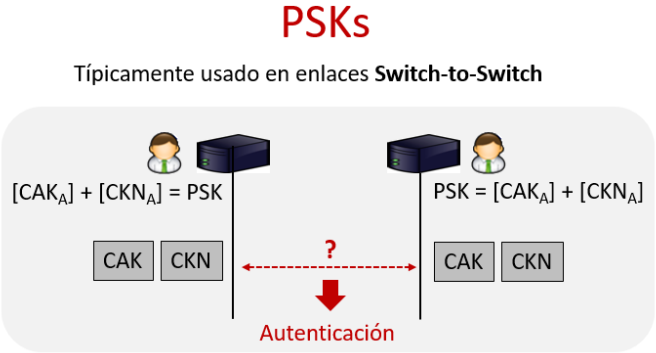

Figura 2. Autenticación y generación de CAK mediante claves pre-compartidas

Recomendación 1:

No se recomienda la autenticación con **claves pre-compartidas**, por dos motivos principales: cualquier parte que sepa la clave podría autenticarse y, además, estas claves son vulnerables a ataques de diccionario. Además, añade complejidad a la instalación y al mantenimiento cuando el número de estaciones es elevado.

Únicamente deberían utilizarse cuando sea posible asegurar que han sido generadas con la entropía suficiente para aportar la fortaleza deseada (Ej.: para sistemas del ENS categoría Alta se exigen 128 bits) y es posible renovarlas en un período inferior a su "cripto período".

b) 802.1X/EAP

IEEE 802.1X define varios términos relacionados con la autenticación:

- **Suplicante**: dispositivo cliente que solicita el acceso a la red.
- **Autenticador**: dispositivo que permite o bloquea el acceso a la red (por ejemplo, un *switch*).
- **Servidor de Autenticación (AS)**: servidor que determina, a partir de las credenciales proporcionadas por el suplicante, si este está autorizado a acceder a los servicios proporcionados por el autenticador. Otra de las funciones del AS es la de generar e intercambiar las claves criptográficas con el suplicante, y distribuirlas de forma segura al autenticador.

Para el proceso de autenticación y autorización de acceso a la red, el estándar IEEE 802.1X define el protocolo **EAPoL** (*EAP over LAN*), que es la encapsulación de EAP (*Extensible Authentication Protocol*) sobre redes LAN.

El proceso de autenticación y autorización, se desarrolla de la siguiente forma:

1. El suplicante inicia el proceso de autenticación enviando un *EAPoL Start* al Autenticador. Este le responde solicitándole la identidad, a través de una trama *EAP Request / Identity*. El suplicante responde con *EAP Response*, indicando cuál es su identidad y ratificando con ello su solicitud de autenticación.
2. A partir de entonces, el Autenticador hace de intermediario entre el Suplicante, con el que se comunica con mensajes EAP, y el Servidor de Autenticación (AS), con el que se comunica con mensajes en función del protocolo de autenticación, por ejemplo, RADIUS. El AS y el suplicante, en primer lugar, acordarán cuál es el mecanismo de autenticación (método EAP). En función del mecanismo empleado, el suplicante proporcionará las credenciales correspondientes al AS y este las verificará (intercambio de mensajes *EAP Request* y *Response*, trasladadas por el Autenticador). Si todo el proceso es correcto, el Autenticador finalizará con un mensaje *EAP Success* y permitirá al suplicante el acceso a la red.
3. Una vez autenticado el solicitante y permitido su acceso a la red, el servidor de autenticación y el solicitante acuerdan una clave maestra MSK (*Master Session Key*). Para ello, se utiliza también EAP. Este material criptográfico puede ser derivado de mutuo acuerdo entre el solicitante y el AS, o puede ser generado por el AS y distribuido al solicitante. El AS posteriormente deberá distribuirlo por un canal seguro al autenticador.

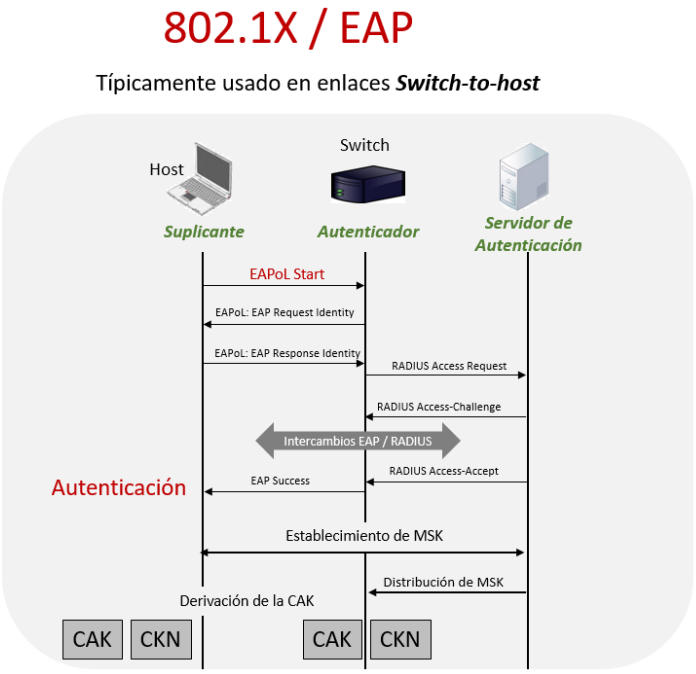

Figura 3. Autenticación y generación de CAK mediante 802.1X/EAP

A partir de la MSK se deriva la clave **CAK** (*Connectivity Association Key*). Esta es una **clave de larga vida** a partir de la cual se deriva el material criptográfico para MKA y MACsec. Todos los dispositivos que comparten dicha clave CAK forman parte de una *Asociación de Conectividad Segura* (**CA**).

La función de derivación KDF (*Key Derivation Function*), utilizada para generar la clave CAK, hace uso de la función pseudoaleatoria **AES-CMAC-128 o AES-CMAC-256,** dependiendo de si la clave que se desea obtener es de longitud 128 o 256 bits.

Recomendación 2:

En caso de que se utilice la autenticación con **claves pre-compartidas**, se recomienda utilizar un identificador de la clave CAK (CKN) de longitud, al menos, 32 bytes.

En caso de que se utilice la autenticación 802.1X se deberá utilizar una implementación que cumpla el estándar IEEE802.1X (sección 6.2.2 *Using EAP for CAK key derivation*).

Dado que los métodos EAP deben ser capaces de generar y distribuir el material de claves criptográficas y esto representa un proceso complejo y especialmente crítico desde el punto de vista de la seguridad, deberán utilizarse tecnologías maduras. Esto limita mucho las posibilidades de elección entre los métodos EAP[1]. Actualmente, solo se recomiendan los métodos basados en TLS.

[1] Los métodos EAP actualmente existentes, salvo aquellos que son propiedad de fabricantes, se encuentran en el Registro de IANA: *Extensible Authentication Protocol Registry: http://www.iana.org/assignments/eap-numbers/eap-numbers.xhtml#eap-numbers-4.*

Recomendación 3:

Se recomienda la autenticación **IEEE 802.1X/EAP** con el método de autenticación **EAP – TLS** (RFC 5216), empleando una versión de **TLS 1.2 o superior**.

2.4. DERIVACIÓN DE CLAVES ICK Y KEK

Dentro de la jerarquía de claves MKA, la clave raíz de la jerarquía es CAK.

MKA no utiliza la clave CAK directamente, sino que deriva de ella otras claves que irán ligadas al identificador de la CAK (CKN) y cuyo uso, por lo tanto, estará restringido a las entidades que utilizan esa CAK (participantes de la misma CA):

- **Clave ICK**: *Integrity Check Key*. Clave usada para el cálculo del ICV (*Integrity Check Value*) empleado para verificar la integridad de los paquetes MKA (MKPDUs, *MKA Data Units*) y para verificar la autenticación previa realizada entre los pares. Esto último es posible porque el ICV demuestra que el nodo que envía el mensaje cuenta con la clave CAK.

- **Clave KEK**: *Key Encrypting Key*. A través de paquetes MKA (MKPDUs) se enviará la clave final de cifrado (SAK). El envío de esta clave está a su vez protegido con un cifrado *AES Key Wrap* usando la clave KEK.

La función de derivación KDF (*Key Derivation Function*), utilizada para la generación de la clave KEK, hace uso de la función pseudoaleatoria **AES-CMAC-128 o AES-CMAC-256** dependiendo de si la clave que se desea obtener es de longitud 128 o 256 bits.

Recomendación 4:

Se recomienda que la implementación del protocolo MKA genere las claves **ICK y KEK,** mediante la derivación de la CAK haciendo uso de la función KDF. Para ello, la implementación deberá cumplir el estándar IEEE 802.1X-2010/2020 (sección 9.3.3 *Derived keys*).

2.5. CLAVE SAK

SAK es la *Clave de Asociación Segura*, que será una clave asociada a cada asociación de seguridad (SA) y empleada para el cifrado de todos los paquetes que se transmitan en esa SA.

En el proceso de generación y distribución de la clave de cifrado SAK interviene un ***Key Server***, que es uno de los participantes de la CA que desempeñará este rol.

2.5.1 *ELECCIÓN DEL KEY SERVER*

El ***Key Server***, o servidor de claves, es elegido de manera dinámica entre todos los participantes de la CA. Su principal función es la generación, a partir de la clave CAK, de las claves SAKs asociadas a las SAs, junto con la distribución de manera segura de estas a los miembros correspondientes.

La elección del servidor se realiza de la siguiente manera:

1. Cada estación o nodo MACsec envía un mensaje *heartbeat* en broadcast, el cual contiene:
    a. Prioridad del *Key Server*.

b. Información anti-reenvío (lista de nodos *vivos*[2]).

2. Una vez que las estaciones estén de acuerdo en la lista de nodos *vivos*, el nodo que tenga mayor prioridad será el elegido *Key Server*.
3. Si el nodo abandona la lista de nodos *vivos*, se deberá escoger un nuevo *Key Server*.

2.5.2 *DERIVACIÓN DE LAS CLAVES SAK*

La función de derivación KDF (*Key Derivation Function*), utilizada para la generación de la clave SAK en el Key Server, hace uso de la función pseudoaleatoria **AES-CMAC-128** en caso de que la clave CAK sea de 128 bits, o **AES-CMAC-256**, si la clave CAK es de 256 bits.

Recomendación 5:

Se recomienda que la implementación del protocolo MKA genere las claves SAKs mediante derivación de la CAK. Para ello, la implementación deberá cumplir el estándar IEEE 802.1X-2010/2020 (sección 9.8.1 *SAK generation*).

A cada clave SAK se le asigna un identificador (**KI**) de 128 bits, compuesto por el identificador del *key server* (**MI,** *Member Identifier*) de 96 bits, junto con un número de 32 bits (**KN,** *Key Number*) asignado a la clave SAK por el Key Server. Este KN se asigna de forma secuencial empezando en 1.

2.5.3 *SELECCIÓN DE CIPHER SUITE Y DISTRIBUCIÓN DE LAS CLAVES SAK*

El *Key Server* selecciona la *cipher suite* y la anuncia con cada clave SAK. Se distribuye el código identificativo de la *cipher suite* (8 octetos). También se distribuye un valor que es el "*Confidentiality Offset*", que puede ser 0, 30 o 50. Este valor significa que los primeros 0, 30 o 50 octetos del paquete MAC (MSDU) no van cifrados, solo llevan protección de integridad y aparecerán en el paquete MACsec (MPDU) inmediatamente después de campo SecTAG en el orden y composición que iban en MSDU. Los restantes octetos sí irán cifrados[3].

Las cipher suites posibles son cuatro: *GCM-AES-128, GCM-AES-256, GCM-AES-XPN-128 y GCM-AES-XPN-256* y se detallan en el apartado 3.

El *Key Server* distribuye la clave SAK cifrada con ***AES Key Wrap*[4] usando la clave KEK.** Todos los participantes que dispongan de la CAK de la que se ha derivado la KEK, podrán descifrar el paquete. En el paquete MKA (MKPDU), el *Key Server* también incluye la *cipher sui*te seleccionada para esa SAK y un número que el Key Server asignará a esa SAK (KN, *Key Number*). Este KN se utilizará, junto con otros parámetros, para componer el identificador de la clave SAK (KI, *Key Identifier*).

[2] La expresión *nodos vivos* hace referencia a la lista de nodos que se encuentran activos y conectados con dicha estación en ese instante.

[3] La opción de *Confidentiality offset* se especificó en la 802.1AE-2006 para facilitar la implementación temprana de MACsec en sistemas que necesitaban examinar los octetos iniciales de las tramas de IPv4 o IPv6 y que, por lo tanto, debían ir sin cifrar.

[4] AES Key Wrap según la definición de IETF RFC 3394

Recomendación 6:

Se recomienda que la implementación MKA utilice la función **AES Key Wrap** para la distribución de claves SAK entre nodos MACsec. Para ello, la implementación deberá cumplir el estándar IEEE 802.1X-2010/2020 (sección 9.8.2 *Use of AES Key Wrap*).

El Key Server distribuirá las claves SAK en los paquetes MKPDU a todos los nodos de la red, hasta que dichos nodos respondan al mensaje confirmando la instalación de la clave SAK.

Las claves SAK se regeneran si se da alguno de los siguientes casos:

- Se supera el límite de usos de la clave. Dependiendo de la *cipher suite* usada, se permite un mayor o menor número de usos con una misma clave. Esto se desarrolla con mayor detalle en el apartado 3.
- Se añade un nuevo nodo o se elimina un nodo de la CA.
- Se modifica la *cipher suite* a usar.
- Se ha añadido un tiempo límite para la regeneración de las claves SAK.

En cualquiera de estos casos, el *Key Server* es el encargado de volver a generar una nueva clave SAK, tal y como se indica en el apartado 2.5.2, y de distribuirla a todos los miembros de la CA.

Recomendación 7:

Se recomienda que la implementación del protocolo MKA especifique que el *Key Server* regenere la clave SAK cada vez que se añada o elimine un nuevo miembro a la CA. Para ello, la implementación deberá cumplir el estándar IEEE 802.1X-2010/2020 (sección 9.8 *SAK generation, distribution, and selection*).

2.6. TRANSPORTE MKA

Como se ha indicado con anterioridad, MKA proporciona un transporte seguro de información multipunto a multipunto entre los miembros de una misma CA. La clave CAK, que debe ser conocida por todos los miembros de la CA, se utiliza para autenticar cada paquete trasmitido (MKPDU), ya que la posesión de esa CAK demuestra que el transmisor es un miembro autenticado de la CA.

Cada miembro emisor incluye en el paquete MKPDU:

- El nombre de la CAK (CKN).
- Su SCI, *Secure Channel Identifier*, compuesto por la dirección MAC y el puerto.
- Su MI: *Member Identifier*, número aleatorio de 96 bit que escoge cada miembro cuando comienza su participación en el protocolo.
- Su MN: *Message Number*, número de 32 bits secuencial (iniciándose en 1) asignado a los paquetes MKPDU emitidos por un transmisor. Se incrementa con cada MKPDU transmitido.

- Un valor ICV (*Integrity Check Value*) calculado a partir de la clave ICK (derivada, a su vez, de la clave CAK tal y como se indica en el apartado 2.4).

Al incluir el identificador de la CAK usada (CKN), el receptor podrá verificar el ICV usando la CAK correcta. Esta verificación demuestra que el emisor poseía la CAK y, por lo tanto, se puede autenticar el paquete MKPDU.

Por otro lado, el uso de los valores MI y MN en conjunto, permite proteger las transmisiones de retardos o de ataques *replay*.

2.7. SESIÓN SEGURA MACsec

Una vez los nodos o estaciones MACsec han instalado correctamente la clave SAK, todo el tráfico transmitido estará cifrado con las *cipher suites* elegidas por el *Key Server*, utilizando las claves SAK correspondientes a cada SA. Estas claves SAK se van renovando, según se vaya superando el límite de paquetes enviados con dicha clave.

En la sesión segura, solo se permitirá el paso del tráfico cifrado MACsec, con la excepción del tráfico de control, como las tramas EAPoL.

La sesión segura utiliza un identificador único SCI (*Secure Channel Identifier*). Durante la sesión establecida solo se aceptan aquellos paquetes que contengan el identificador SCI de esa sesión.

La sesión segura también proporciona **protección anti-reenvíos (*anti-replay*)** utilizando el PN (*packet number*) y dos parámetros: *replayProtection* y *replayWindow*. El primero indica si está o no activa la protección anti-reenvío. El segundo indica la ventana de paquetes que podrán llegar desordenados y ser admitidos. Por ejemplo, si el *replayWindow* es 300 quiere decir que si recibe un paquete con PN=5, el siguiente paquete que admitirá podrá tener un PN entre 6 y 306. La protección anti-reenvío, consiste en un *replayWindow* cero, que es el valor por defecto indicado en el estándar IEEE 802.1AE.

Recomendación 8:

Se recomienda que la implementación del protocolo MACsec disponga de la **protección anti-reenvío**. Para ello, la implementación deberá cumplir el estándar IEEE 802.1X-2010/2020 (sección 10.7.8 *Frame verification controls*).

3. CIPHER SUITES Y PARÁMETROS CRIPTOGRÁFICOS

El protocolo MACsec utiliza un algoritmo para el cifrado de los paquetes y para la protección de integridad, a través de lo que se denomina la suite criptográfica o ***cipher suite***. Esta *cipher suite* está compuesta por los algoritmos y funciones criptográficas, junto con sus parámetros (longitud de clave, modo de operación, etc.) que se van a utilizar para proteger la conexión.

De forma general, la *cipher suite* se representa de la siguiente manera:

Mode-Cipher-KeyLength

Los diferentes valores de la *cipher suite* corresponden al algoritmo de cifrado (*Cipher*), su longitud de clave (*KeyLength*) y su modo de operación (*Mode*).

En la versión actual del estándar MACsec[5], se encuentran aceptadas las siguientes cuatro (4) *cipher suites*, las cuales se emplean tanto para el cifrado como para la protección de la integridad.

Cipher Suite	Obligatoria / Opcional
GCM-AES-128	Obligatoria
GCM-AES-256	Opcional
GCM-AES-XPN-128	Opcional
GCM-AES-XPN-256	Opcional

Tabla 1. Cipher suite aceptadas

Como se puede observar en la tabla, la *cipher suite GCM-AES-128* es **obligatoria** y el resto son opcionales. En caso de emplear una *cipher suite* distinta a las indicadas en la tabla, deberán cumplirse una serie de requisitos indicados en el apartado 14.4.1 del estándar IEEE 802.1AE-2018.

Recomendación 9:

Se recomienda que la implementación del protocolo MACsec utilice las *cipher suites* indicadas en la tabla anterior (Tabla 1). Para ello, la implementación deberá cumplir el estándar IEEE 802.1AE-2018 (sección 14.4 *Cipher Suite conformance*).

Indicar que, como se puede observar en las *cipher suites* de MACsec, la mínima longitud de clave AES es **128 bits.** Esto representa una **fortaleza suficiente para categoría ALTA del ENS**.

Respecto al modo de operación de AES, actualmente MACsec solo soporta el modo **GCM**. *Galois Counter Mode*[6] es un modo de operación para cifradores de bloque. La operación que realiza es lo que se llama un AE "***Autheticated Encryption***" ya que proporciona autenticidad y confidencialidad.

Las *cipher suites* indicadas en la tabla como *GCM-AES-XPN-128/256* hacen uso de la característica ***MACsec Extended Packet Numbering (XPN)***, que se describe a continuación.

[5] Estándar IEEE 802.1AE-2018.

[6] Se puede consultar más información sobre el funcionamiento del modo GCM en la ISO 17792 (Authenticated Encryption Modes), o en la publicación especial NIST SP 800-38D: *Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC.*

Como en otros modos de operación de cifradores de bloque, se requiere el uso de un Vector de Inicialización (IV) que deberá ser único para cada paquete cifrado con una clave (SAK, en este caso). Para componer el IV, se utiliza el parámetro **PN**[7] (*Packet Number*) de **32 bits**. Esto indica que el límite de usos de una misma clave SAK será de 2^{32}, lo cual puede ser un problema en enlaces de alta velocidad (40 Gb/s), ya que el PN se agota a los pocos segundos y es necesario volver a generar las claves SAK.

Con el uso de **XPN (*Extended* PN)**, se extiende la longitud del PN de cada paquete MACsec hasta 64 bits, lo cual requeriría varios años para que se agote el PN, asegurando que la regeneración de claves SAK no se produzca con tanta frecuencia en enlaces de alta velocidad.

Es importante indicar que el uso de XPN está diseñado solo para dispositivos que cuenten con un rendimiento igual o superior a 40 Gb/s.

La *cipher suite* y la clave SAK, se utilizan para el cifrado y para la protección de integridad.

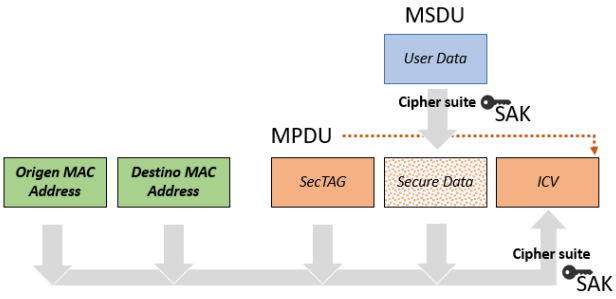

Figura 4. Cifrado y cálculo del ICV MACsec

MACsec utiliza dos (2) cifrados:

(a) El cifrado de la clave SAK para su envío seguro. Esto lo hace el *Key Server* con el protocolo MKA, y utiliza para ello la función *AES Key Wrap* (ver apartado 2.5.3).

(b) El cifrado de los datos (MSDU) utilizando la clave de cifrado SAK. Para este cifrado se utiliza una de las *cipher suites* indicadas.

Es importante indicar que, tal y como se especifica en el estándar IEEE 802.1AE-2018 (sección 14.2 *Cipher Suite capabilities*) **en MACsec el uso del cifrado es opcional**.

Recomendación 10:

Se recomienda que se configure el uso del cifrado en MACsec.

Para proporcionar integridad al paquete MACsec, se realiza el cálculo del valor **ICV** (*Integrity Check Value*) el cual es posteriormente añadido al paquete MACsec. **Esta función del cálculo del ICV sí es obligatoria.**

El cálculo del ICV protege la integridad de las direcciones MAC origen y destino, los datos (cifrados, en caso de que se esté aplicando la protección de confidencialidad) y la etiqueta SecTAG.

[7] PN (*Packet Number*): Es un número de 32 o 64 bits (en este último caso es *Extended Packet Number, XPN*) que se va incrementando y que es único para cada trama MACsec transmitida con una misma clave SAK.

Recomendación 11:

Se recomienda que la implementación del protocolo MACsec proporcione integridad al paquete transmitido (MPDU) mediante el cálculo del parámetro **ICV** (*Integrity Check Value*) derivado de la clave SAK. Para ello, la implementación deberá cumplir el estándar IEEE 802.1AE-2018 (sección 14.1 *Cipher Suite use*).

El cifrado y el cálculo del ICV utilizan la función ***Protect***, mientras que su descifrado y verificación utilizan la función ***Validate***. Se puede consultar más información sobre la operación de estas funciones en la sección 14. *Cipher Suites*, del estándar IEEE 802.1AE.

4. MACSEC PROTOCOL DATA UNITS (MPDUS)

En la capa de enlace (L2 del modelo OSI), durante una comunicación sin MACsec, el encapsulado de los paquetes por la red LAN es el siguiente:

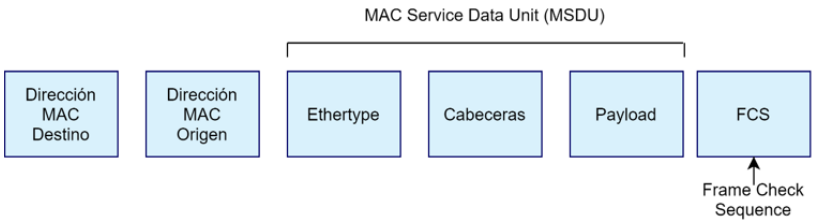

Figura 5. Paquete encapsulado L2 normal

MACsec, para componer el MPDU añade una cabecera adicional (**SecTAG**) insertada tras la dirección MAC de origen y antes del campo *Ethertype*, añade un valor ICV al final de la trama para la verificación de integridad y, en el caso de proporcionar confidencialidad, cifra los datos recibidos (MSDU).

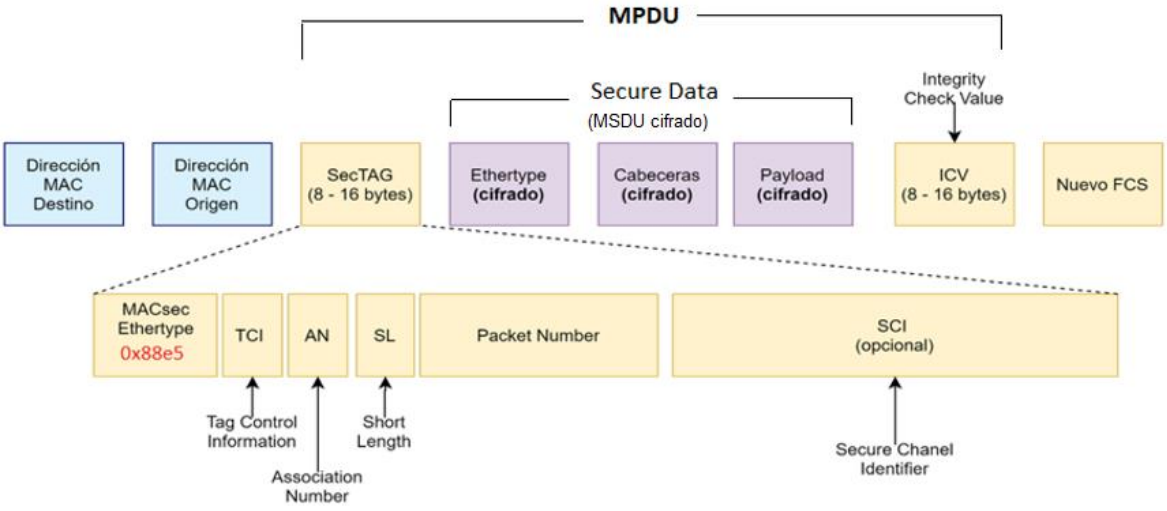

Figura 6. Paquete encapsulado con MACsec

Dentro de la cabecera **SecTAG (16 octetos)**, se añaden varios parámetros, como se puede observar en la figura anterior:

1. **MACsec Ethertype** (2 octetos): Se cambia el valor de *Ethertype* al correspondiente del protocolo MACsec (0x88e5) para indicar que la trama es MACsec.
2. **TCI** (6 bits, del 8 al 3 del tercer octeto): *TAG Control Information*. Contiene varios parámetros, entre ellos, la versión de protocolo MACsec y si los modos *solo integridad* o *solo cifrado* están en uso.
3. **AN** (2 bit, el 1 y 2 del tercer octeto): *Association Number*. Identifica hasta cuatro (4) asociaciones de Seguridad (SA) diferentes dentro de un mismo *Secure Channel (SC)*.
4. **SL** (bits 1 a 6 del cuarto octeto. Los bits 7 y 8 van a cero): *Short Length*. Indica el número de octetos de los datos del usuario (*User Data*), es decir, de los datos que van después de SecTAG y antes de ICV.

5. **PN** (32 bits, octetos 5 a 8): Son los 32 bits menos significativos del número de paquete (*Packet Number*, PN). Junto con el SCI (*Secure Channel Identifier*), estos bits los utilizan las *cipher suites* (AES-GCM-128/256) para construir el vector de inicialización. También se utilizan para protección de ataques *replay*.
6. **SCI** (8 octetos, del 9 al 16): Identificador del canal seguro (*Secure Channel Identifier*) al que corresponde la trama. Está compuesto de la dirección MAC (48 bits) junto con el identificador del puerto (16 bits).

El campo **ICV (*Integrity Check Value*)** es un valor calculado utilizando la *cipher suite* negociada, para garantizar la integridad tanto de todos los campos del MPDU como los campos de dirección MAC origen y destino. Su longitud depende de la *cipher suite* empleada, pero nunca será menor de 8 octetos, ni mayor de 16.

5. USOS DE MACSEC

5.1. MACSEC SOBRE LAN

El principal caso de uso de MACsec es securizar las redes LAN. Para hacerlo hay dos (2) arquitecturas diferentes dependiendo del enlace a proteger, tal y como se muestra en la siguiente imagen.

Figura 7. MACsec sobre LAN

Arquitectura Switch-to-Host

Los enlaces MACsec de tipo ***switch-to-host***, o también denominados *downlink*, establecen una conexión protegida entre un dispositivo de red (*switch*) y un dispositivo final (*endpoint*).

En este tipo de enlace se hace uso de **EAP/802.1X**, tal y como se explica en el apartado 2.3 b), para realizar la autenticación de los dispositivos, y posteriormente hacen uso de MKA para la creación y distribución de claves. El paquete es cifrado en el envío, y descifrado en la recepción.

Arquitectura *Switch-to-Switch*

Los enlaces de tipo ***switch-to-switch*** de MACsec, o también conocidos como *uplink*, establecen una conexión segura entre dos dispositivos de red (*switch*).

La conexión *uplink* puede ser establecida de manera **manual** o **dinámica**. Para la realización **manual** se requiere una configuración previa por parte de un administrador en cada dispositivo de red, indicando los parámetros necesarios a usar durante la conexión. En este caso, se haría uso de la opción de claves pre-compartidas para el establecimiento de la clave CAK.

En el modo **dinámico** se hace uso de 802.1X/EAP para la autenticación, de manera que un *switch* se convierte en el suplicante NDAC [8]y el otro en el servidor de autenticación, encargado de autenticar y de distribuir las claves al suplicante, para que este, posteriormente, se convierta en el autenticador frente a los dispositivos finales.

Recomendación 12:

En arquitecturas ***switch-to-switch***, se recomienda hacer uso del modo **dinámico** para el establecimiento de la conexión.

5.2. MACSEC SOBRE VXLAN

VXLAN (*Virtual Extensible LAN*) es un protocolo que se basa principalmente en encapsular el tráfico de capa de enlace (concretamente, tráfico Ethernet) de una red de área local, y transportarlo sobre

[8] Un suplicante NDAC es un *switch* que actúa como suplicante y se autentica frente al servidor de autenticación, de manera previa a convertirse en un autenticador de usuarios, *hosts*.

una red IP (encapsulado MAC-in-UDP) hasta otra LAN física diferente, consiguiendo así que los *hosts* de ambas redes se puedan comunicar de igual manera que si se encontrasen en la misma red de área local[9].

MACsec es compatible con VXLAN y otras tecnologías de encapsulación similares.

Un ejemplo de aplicación sería un usuario que cuenta con una red LAN privada virtual en la nube. En este caso, se puede hacer uso de MACsec para cifrar todo el tráfico interno entre sus máquinas virtuales. Esto consigue que el proveedor de servicios de nube sea incapaz de ver las comunicaciones entre las máquinas virtuales asegurando así, la confidencialidad de la información del usuario.

El túnel MACsec se establece entre las máquinas virtuales, en la VXLAN proporcionada por el proveedor de servicios en la nube, a través de las interfaces MACsec. En la siguiente imagen se observa como hay dos (2) VXLANs diferentes y solo se hace uso de MACsec sobre VXLAN1, que conecta las máquinas virtuales HA1, HA2 y HA3.

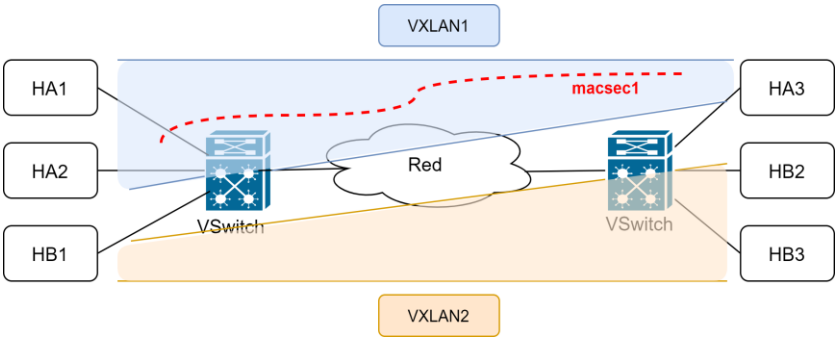

Figura 8. MACsec sobre VXLAN

5.3. MACSEC SOBRE WAN

Otro gran uso de MACsec se da sobre redes WAN. Actualmente, una organización no está compuesta de una sola red LAN, sino que cuenta con diversos proveedores de servicios, infraestructuras *cloud*, etc. El uso de WAN MACsec ofrece una conexión segura entre los diferentes sitios remotos de la organización.

WAN MACsec se basa en los estándares MACsec y utiliza características útiles para acoplarlo al uso en una WAN, como son:

- Configuración del parámetro *Anti-Replay Window*, que indica el tiempo, en segundos, en el cual está permitido el reenvío de paquetes.
- Configuración del parámetro *EAPoL Destination Address*, que permite modificar la dirección MAC destino del paquete MKA. Normalmente, MKA usa la dirección *multicast* EAPoL para que los paquetes lleguen a varios destinos. Como EAPoL es un protocolo usado por varios mecanismos de autenticación, pueden darse situaciones en las que dispositivos de red, cojan el paquete, lo intenten procesar y lo descarten, provocando un fallo en el

[9] VXLAN inicialmente surgió para resolver el problema generado en centros de proceso de datos con miles de máquinas virtuales ya que, en caso necesidad de agrupar estas máquinas en VLANs, el número de VLANs que se pueden generar es limitado. Con VXLAN se proveen los mismos servicios que una red VLAN convencional, pero aumentando la extensibilidad y la flexibilidad limitadas de este tipo de redes.

establecimiento de la sesión MKA. Para evitar estas situaciones se permite configurar el parámetro *EAPoL Destiantion Address*.

El uso de MACsec en redes WAN ha ido creciendo durante los últimos años, debido a la seguridad que proporciona y a la velocidad que presenta frente a la tecnología IPsec. Esto se encuentra explicado más en detalle en el apartado 7.

En la siguiente imagen se muestra un ejemplo del funcionamiento de WAN MACsec *site-to-site*:

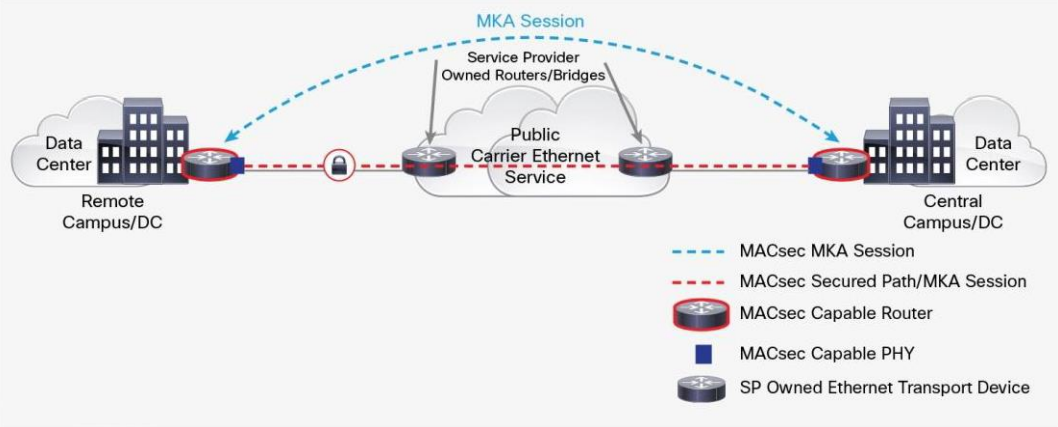

Figura 9.WAN MACsec ejemplo site-to-site.10

[10] Imagen obtenida de: https://www.cisco.com/c/dam/en/us/td/docs/solutions/Enterprise/Security/MACsec/WP-High-Speed-WAN-Encrypt-MACsec.pdf

6. SERVICIOS DE SEGURIDAD MACSEC

Una vez se establece la sesión segura, MACsec proporciona las siguientes características de seguridad:

Característica	Descripción
Confidencialidad	El MSDU puede ser cifrado completa o parcialmente (*confidentiality offset*) o no cifrarse. En caso de cifrado, este proporciona **confidencialidad** en las comunicaciones de la capa de enlace. Los algoritmos de cifrado usados son de tipo AEAD, principalmente AES-GCM.
Integridad	El uso de **ICV** asegura la integridad de las direcciones MAC, del *SecTAG* y del *MSDU (User Data)*.
Autenticidad de Origen	La clave única **SAK**, distribuida mediante el protocolo MKA, es usada para autenticar (cálculo del ICV) y cifrar cada paquete. Esta clave solo está en posesión de las entidades que pertenecen a dicha SA, por lo que solo dichas entidades podrán descifran y autenticar los paquetes protegidos con esa clave SAK.
Filtrado de tráfico solo MACsec	Permite distinguir el paquete MACsec de cualquier otro no protegido, gracias a los dos primeros bytes de *SecTAG*, en los cuales se indica el *ethertype* de MACsec (**0x88e5**).
Protección anti-reenvío	Cada paquete cuenta con un *Packet Number* (PN), único en cada SAK, que es usado para la detección y el descarte de paquetes reenviados utilizando los parámetros *replayProtection* y *replayWindow*.
Protección anti-retardo	El uso del parámetro *bounded time* permite a MACsec establecer un tiempo límite entre el envío de un paquete por el origen y la recepción del paquete en el destino.
Arquitectura *hop-by-hop*	MACsec ayuda a la protección de la red interna mediante la protección de los datos intercambiados con una técnica ***hop-by-hop***. Es decir, los paquetes son descifrados y, seguidamente, cifrados por cada nodo MACsec por el que pasan.
Seguridad Punto a Multipunto	La operación MACsec se basa en SAs (Asociaciones de Seguridad) unidireccionales punto-a-multipunto. Cada estación MACsec cuenta con una SA para el tráfico de salida y una SA para el tráfico entrante procedente de cada estación MACsec.

Tabla 2. Características de sesión segura MACsec

7. MACSEC & IPSEC

MACsec puede usarse en combinación con otros protocolos de diferentes capas, como puede ser IPsec (capa L3 de OSI), para proporcionar seguridad extremo a extremo.

IPsec es un protocolo que proporciona servicios de seguridad a nivel de red, frente a MACsec que trabaja a nivel de capa de enlace. Ambos protocolos pueden usarse conjuntamente proporcionando una seguridad completa a la arquitectura de red, tal y como se muestra en el siguiente ejemplo:

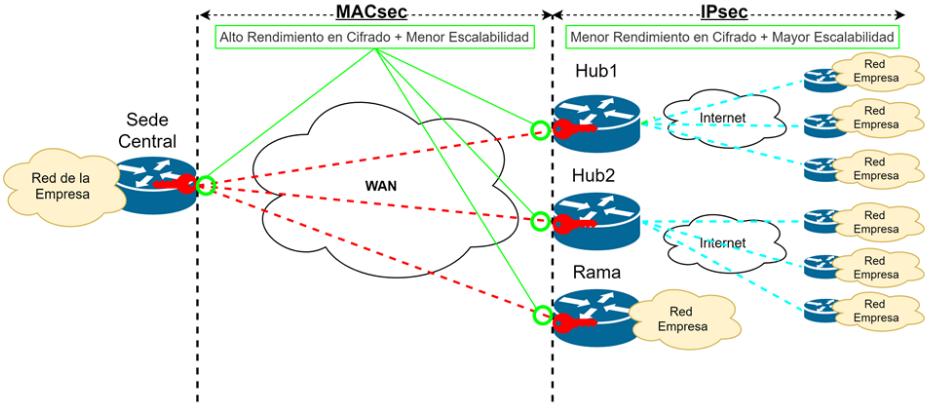

Figura 10. Arquitectura MACsec + IPsec

El enfoque del diseño híbrido MACsec / IPsec mostrado en la imagen anterior es óptimo para mezclar tecnologías con distinto rendimiento en cuanto al cifrado.

Uno de los problemas de IPsec es el bajo rendimiento de cifrado en altas velocidades. En la imagen mostrada a continuación se observa como a partir de velocidades superiores a los 40 GB/s, la velocidad de cifrado es menor que la velocidad del enlace, suponiendo una infra utilización de la red. MACsec, a diferencia de IPsec, proporciona una velocidad de cifrado acorde a la velocidad del enlace, permitiendo un cifrado de alto rendimiento en enlaces de altas velocidades. Por esta razón, es recomendable el uso de MACsec frente a IPsec en enlaces de alta velocidad.

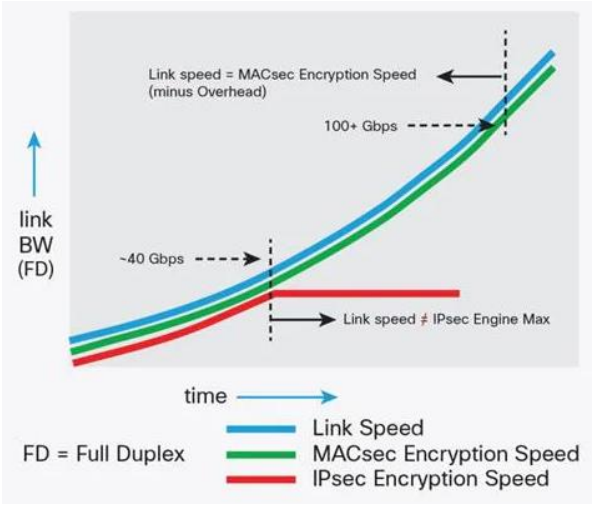

Figura 11. Velocidad de cifrado de IPsec, MACsec frente a la velocidad del enlace[11]

[11] Imagen obtenida de: *https://www.cisco.com/c/en/us/products/collateral/ios-nx-os-software/identity-based-networking-services/white-paper-c11-737544.html*

MACsec proporciona un alto rendimiento de cifrado, pero solo puede usarse en arquitecturas de red tipo punto a punto, o punto a multipunto, teniendo así una menor escalabilidad. Por esta razón, WAN MACsec se usa entre los *routers* regionales o enlaces troncales, los cuales hacen uso de enlaces de altas velocidades.

En las ramas de las diferentes localizaciones, que cuentan con un mayor número de dispositivos, se hace uso de IPsec para proporcionar una mayor escalabilidad. Esto es debido a que IPsec permite una enrutación más simple sobre cualquier topología de red y sobre miles de dispositivos finales. La combinación de ambas tecnologías ofrece una seguridad extremo a extremo, proporcionado una escalabilidad y velocidad de cifrado adecuadas a cada tipo de enlace.

8. RESUMEN

A continuación, se incluye una tabla con las recomendaciones realizadas a lo largo de este documento. **Se ha indicado con un asterisco (*) las que son de obligado cumplimiento para aquellos sistemas bajo el alcance del Esquema Nacional de Seguridad (ENS).**

Nº	Ámbito	Resumen de la Recomendación	Tipo de Recomendación[12]
1	Autenticación	Únicamente deberían utilizarse **claves pre-compartidas** (**PSK**) cuando sea posible asegurar que han sido generadas con la entropía suficiente para aportar la fortaleza deseada	Configuración
2	Autenticación	Asociación de cada identificador CKN con su respectiva CAK. Usar un CKM de, al menos, 32 bytes	Configuración Implementación
3	Autenticación	Usar autenticación **IEEE 802.1X/EAP** con método EAP, usar **EAP - TLS** (**TLS 1.2 o superior**).	Configuración
4	MKA	Generación de las claves ICK y KEK mediante la derivación de la CAK haciendo uso de la función KDF.	Implementación
5	MKA	Generación de las claves SAKs mediante derivación de la CAK.	Implementación
6 (*)	MKA	Distribución de claves SAK entre nodos MACsec, mediante **AES Key Wrap**.	Implementación
7	MKA	Regeneración de la clave SAK al añadir o eliminar un nuevo miembro a la CA.	Implementación
8	MACsec	Protección anti-reenvío activada	Implementación
9	*Cipher suites*	Utilizar las cipher suites indicadas en el estándar: ▪ GCM-AES-128 ▪ GCM-AES-256 ▪ GCM-AES-XPN-128 ▪ GCM-AES-XPN-256	Implementación
10	MACsec	Activar el uso del cifrado	Configuración
11	Integridad	Proporcionar **integridad** al paquete MACsec mediante el uso del parámetro **ICV**.	Implementación
12	Arquitectura *Switch-to-switch*	Hacer uso del modo **dinámico** para el establecimiento de la conexión.	Configuración

Tabla 4. Resumen de Recomendaciones MACsec

[12] El tipo de recomendación se refiere a si se trata de algo que el administrador debe configurar de forma manual (Configuración) o bien se trata de algo que la implementación del protocolo debe cumplir (Implementación).

10. ANEXO. EJEMPLO DE CONFIGURACIÓN DE MACSEC CON MKA EN LINUX

Como se ha mencionado en el apartado 8, Linux cuenta con una implementación de *software* para MACsec. Esta se encuentra en el conjunto de herramientas *iproute2* y se ejecuta mediante el comando *ip macsec.* Esta herramienta no hace uso del protocolo MKA, simplemente se limita a establecer una comunicación MACsec entre dos (2) hosts, con unas claves SAK predefinidas por el usuario. Un ejemplo de configuración básica se encuentra en la [REF3**¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.**].

Además de dicha herramienta, Linux cuenta con otra más específica, ***wpa_supplicant***, que permite hacer uso del protocolo **MKA** y establecer una conexión MACsec. Esta viene integrada en las últimas versiones de Linux. En caso de no venir integrada por defecto, es necesario instalar el paquete *wpasupplicant* de la siguiente forma:

$ sudo apt-get install wpasupplicant

Mediante la herramienta *wpa_supplicant* se pueden realizar diferentes configuraciones. Entre ellas, activar MACsec mediante el uso de **IEEE 802.1X/EAP** o claves **pre-compartidas** para la autenticación.

A continuación, se expone un ejemplo de configuración del protocolo MACsec entre dos (2) hosts diferentes, alojados en máquinas virtuales diferentes. La arquitectura planteada es la mostrada en la siguiente imagen, haciendo uso de la herramienta *wpa_supplicant* con **claves pre-compartidas**.

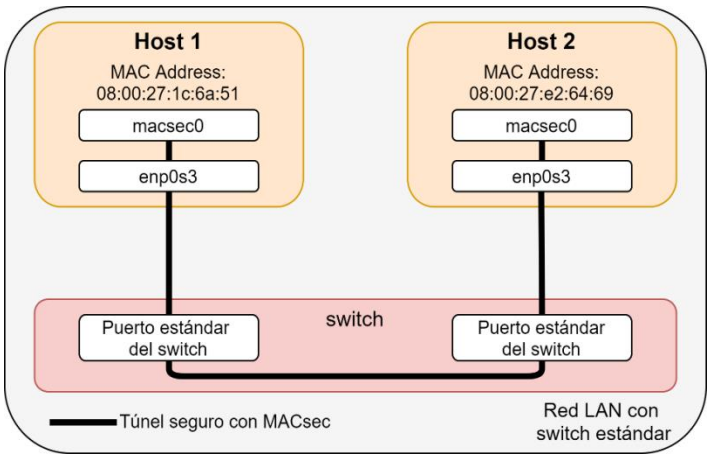

Figura 12. Arquitectura a configurar

Los pasos a seguir son los siguientes:

1. El primer paso será la creación de la clave **CAK** (16 bytes, en hexadecimal) y del valor del parámetro **CKN** (32 bytes, en hexadecimal). Esto puede realizarse con un generador aleatorio de claves. Para el ejemplo se usarán unos valores simples:

   CAK: 12345678901234567890123456789012

   CKN: 1234567890123456789012345678901234567890123456789012345678901234

2. Una vez definidos los valores de CAK y CKN, se ha de generar el fichero de **configuración** que se le pasa a la herramienta y en el cual se indica la configuración a utilizar. El nombre del fichero será el elegido por el usuario. En este ejemplo será *wpa_supplicant.conf*. Para generar y modificar el fichero se usa el comando:

$ sudo nano wpa_supplicant.conf

La estructura del fichero es la siguiente:

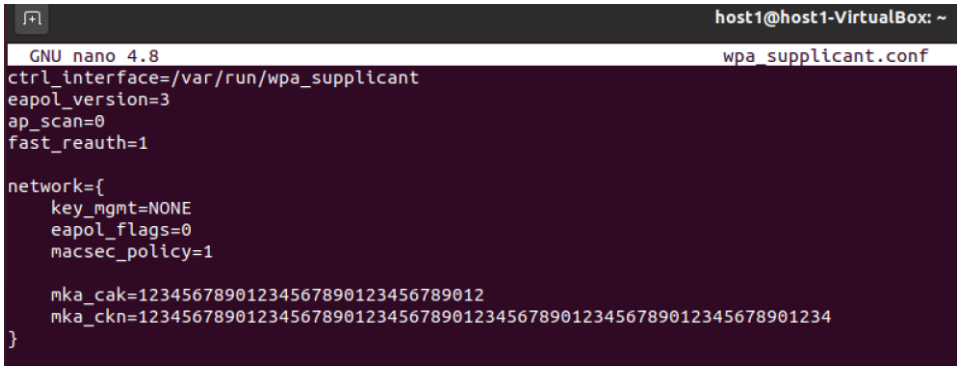

Figura 13. Fichero de configuración

Los parámetros que se encuentran indicados son:

- ***ctrl_interface***: Ruta de la localización de la interfaz de control. Por defecto: */var/run/wpa_supplicant*.
- ***eapol_version***: Indica la versión del estándar IEEE802.1X que se va a usar. Como **MKA** se añadió en la última versión, esta se hace referencia con el valor **3**.
- ***ap_scan***: En el caso de las conexiones Ethernet cableadas, el valor ha de ser **0**.
- ***fast_reauth***: Variable para desactivar la rápida reautenticación de EAP.
- ***key_mgmt***: Indica qué protocolo se usa para el establecimiento y manejo de las claves. En este caso se indica el parámetro *NONE* ya que se va a hacer uso de **claves pre-compartidas**. En caso de querer hacer uso de IEEE 802.1X se debería indicar el parámetro *IEEE8021X*.
- ***eapol_flags***: Cuando se hace uso de autenticación por cable, se ha de poner con valor **0** para una autenticación correcta.
- ***macsec_policy*:** Se pone a **1** para indicar que se quiere activar el protocolo MACsec. Por defecto, no se encuentra activado.
- ***mka_cak***: Clave CAK pre-compartida creada anteriormente.
- ***mka_ckn***: Valor CKN pre-compartido creado anteriormente.

3. Una vez se ha creado el archivo de configuración, solo quedaría **activar la conexión**. Esto se realiza con el siguiente comando:

$ sudo wpa_supplicant -i enp0s3 -Dmacsec_linux -c wpa_supplicant.conf

Los parámetros del comando son:

- *-i*: indica la interfaz de red sobre la que se creará el dispositivo MACsec.
- *-Dmacsec_linux*: indica que se hace uso del driver de MACsec para Linux.
- *-c*: indica el fichero de configuración a usar.
- *-B*: Este parámetro se puede añadir para ejecutar el protocolo como un demonio en segundo plano.

```
host1@host1-VirtualBox:~$ sudo wpa_supplicant -i enp0s3 -Dmacsec_linux -c wpa_supplicant.conf
Successfully initialized wpa_supplicant
enp0s3: Associated with 01:80:c2:00:00:03
enp0s3: CTRL-EVENT-CONNECTED - Connection to 01:80:c2:00:00:03 completed [id=0 id_str=]
enp0s3: CTRL-EVENT-SUBNET-STATUS-UPDATE status=0
```

Figura 14. Resultado del comando de activación de la conexión

4. Una vez ejecutado el comando anterior se observa como se ha iniciado la conexión, y como se ha creado la nueva interfaz MACsec con el nombre macsec0.

```
host1@host1-VirtualBox:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.68  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 fe80::6e7a:56d7:8a24:4f35  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:1c:6a:51  txqueuelen 1000  (Ethernet)
        RX packets 71316  bytes 70760490 (70.7 MB)
        RX errors 0  dropped 312  overruns 0  frame 0
        TX packets 14027  bytes 1449827 (1.4 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Bucle local)
        RX packets 704  bytes 69686 (69.6 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 704  bytes 69686 (69.6 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

macsec0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1468
        inet6 fe80::a00:27ff:fe1c:6a51  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:1c:6a:51  txqueuelen 1000  (Ethernet)
        RX packets 19  bytes 2655 (2.6 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 37  bytes 6466 (6.4 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Figura 15. Visualización de la nueva interfaz

5. Además, se puede ver que el protocolo MACsec está activado con el comando:

$ ip macsec show.

```
host1@host1-VirtualBox:~$ ip macsec show
4: macsec0: protect on validate strict sc off sa off encrypt on send_sci on end_station off scb off replay off
    cipher suite: GCM-AES-128, using ICV length 16
    TXSC: 0800271c6a510001 on SA 0
        0: PN 20, state on, key 829d15ea65a884c00bd83b7301000000
    RXSC: 080027e264690001, state on
        0: PN 24, state on, key 829d15ea65a884c00bd83b7301000000
```

Figura 16. Información sobre el estado del protocolo MACsec

Con este comando se puede observar qué **cipher suite** se está usando, si se encuentra activo el cifrado, el **PN** de los canales de transmisión y de recepción, si está activada la protección anti-replay, etc. Se puede observar también cómo hay dos (2) canales seguros diferentes, cada uno con su clave SAK y su valor PN: el **TXSC**, canal de transmisión de datos, y el **RXSC**, canal de recepción de datos.

Además, se puede ver qué dirección mandará los paquetes por cada canal seguro. En el de transmisión, la dirección es la dirección MAC del *host1*, y en el canal de recepción es la dirección MAC de la máquina *host2*.

6. Para comprobar que los canales MACsec establecidos funcionan, se va a realizar la misma configuración en la máquina host2:

   a) Se hace uso de la **misma clave CAK** y el **mismo CKN**, para estar en la misma CA.

   b) El fichero de configuración *wpa_supplicant.conf* es el mismo que el que se ha configurado para el *host1*.

c) Se ejecuta el comando para activar *wpa_supplicant* y se observa que ha sido activado en la máquina host2.

En este caso, vamos a observar con mayor detalle la información de la configuración, mediante el uso del comando:

$ ip -s macsec show:

```
host2@host2-VirtualBox:~$ ip -s macsec show
5: macsec0: protect on validate strict sc off sa off encrypt on send_sci on end_station off scb off replay off
    cipher suite: GCM-AES-128, using ICV length 16
    TXSC: 080027e264690001 on SA 0
    stats: OutPktsUntagged InPktsUntagged OutPktsTooLong InPktsNoTag InPktsBadTag InPktsUnknownSCI InPktsNoSCI
InPktsOverrun
                       0             0             0         253           0                0           0
           0
    stats: OutPktsProtected OutPktsEncrypted OutOctetsProtected OutOctetsEncrypted
                           0            25                   0              3967
        0: PN 26, state on, key 829d15ea65a884c00bd83b7301000000
    stats: OutPktsProtected OutPktsEncrypted
                           0            25
    RXSC: 0800271c6a510001, state on
    stats: InOctetsValidated InOctetsDecrypted InPktsUnchecked InPktsDelayed InPktsOK InPktsInvalid InPktsLate
InPktsNotValid InPktsNotUsingSA InPktsUnusedSA
                           0          3146             0             0       20             0           0
           0             0             0
        0: PN 21, state on, key 829d15ea65a884c00bd83b7301000000
    stats: InPktsOK InPktsInvalid InPktsNotValid InPktsNotUsingSA InPktsUnusedSA
                  20            0             0                0             0
```

Figura 17. Información sobre el estado del protocolo MACsec

7. A continuación, se va a realizar una prueba sencilla para verificar el funcionamiento del protocolo. Para ello, se van a asignar unas direcciones IP a cada interfaz *macsec0* en los dispositivos, para realizar un ping y capturar la información con un analizador de protocolos.

   a) En la máquina *host1*:

   $ sudo ifconfig macsec0 10.10.10.1

   b) En la máquina *host2*:

   $ sudo ifconfig macsec0 10.10.10.2

   c) Se realiza el *ping* desde la máquina *host2* a la máquina *host1*:

   $ ping 10.10.10.1

   d) Desde el analizador, se realiza la captura de los paquetes MACsec:

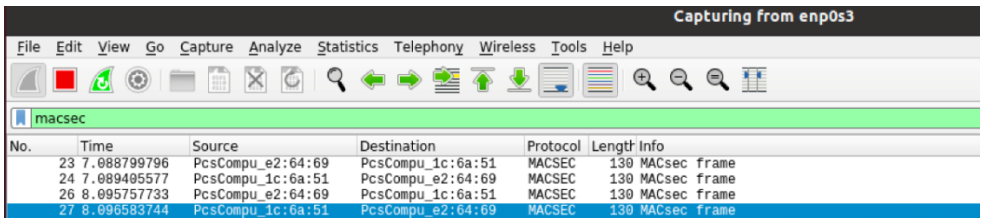

Figura 18. Captura de tráfico

   e) Se obtiene más información sobre el paquete señalado en la Figura 18.

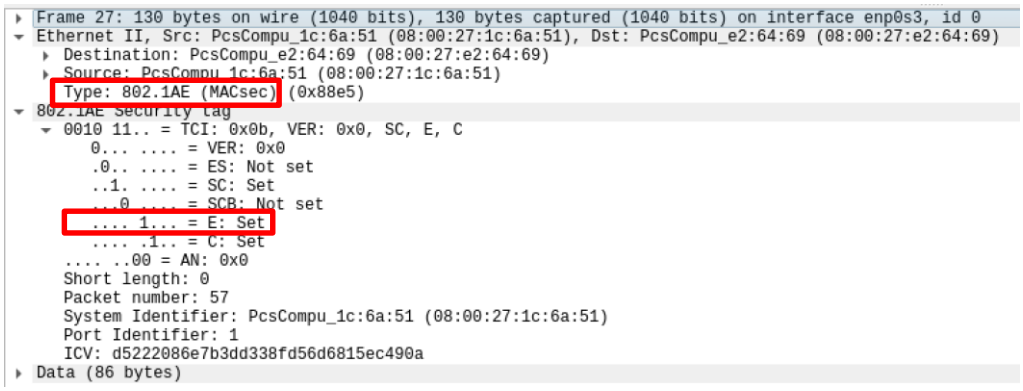

Figura 19. Información de paquete capturado

En la captura se puede observar que el paquete interceptado es de tipo MACsec (802.1AE). Además, dentro de la cabecera **SecTag** del paquete, se ven los diferentes parámetros del protocolo, y entre ellos se puede comprobar como el parámetro **E** (*Encrypted*) se encuentra a **1**, lo que indica que el cifrado está activo, tal y como se había observado anteriormente desde la consola.

8. Por último, se puede observar cómo ha aumentado el valor del parámetro PN, debido al envío de los paquetes:

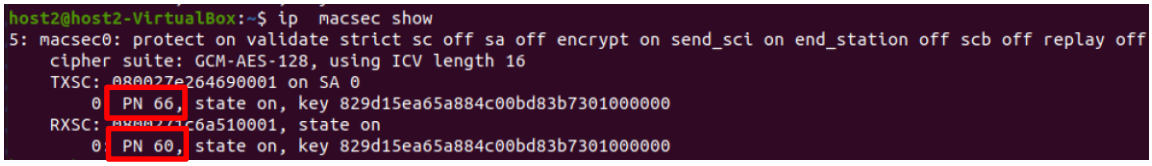

Figura 19. Valores PN actualizados

9. En caso de que en lugar de usar las claves pre-compartidas CAK, se quiera utilizar IEEE 802.1X/EAP, el fichero de configuración sería similar al siguiente:

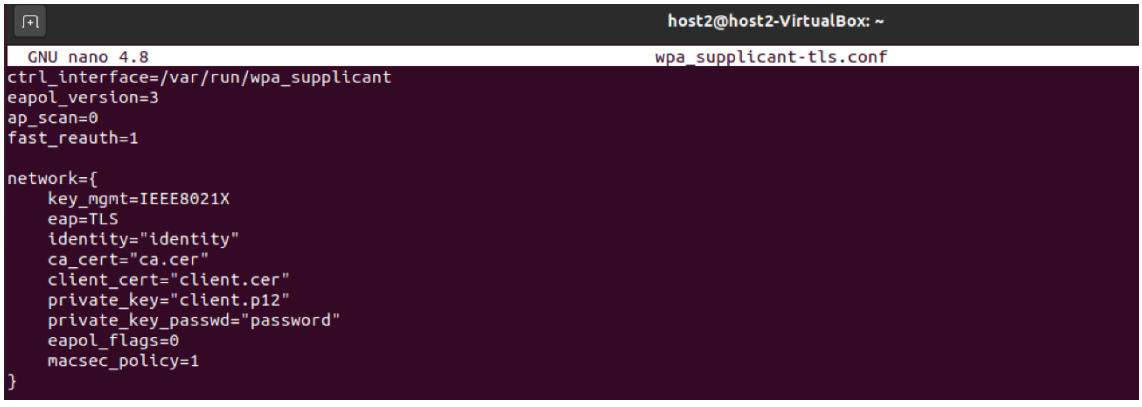

Figura 20. Fichero de configuración con IEEE802.1X/EAP

Los parámetros adicionales que se han de añadir para este caso son los siguientes:

- ***key_mgmt***: Al hacer uso del estándar IEEE 802.1X, el valor que ha de tomar este parámetro es *IEEE8021X*.
- ***eap****:* En el caso de hacer uso de IEEE 802.1X se ha de indicar que método EAP se va a utilizar. Las diferentes opciones son: MD5, MSCHAPV2, OTP, GTC, TLS, PEAP o TTLS. EAP-TLS es la opción recomendada.
- ***identity***: Valor usado para la identidad EAP, como un nombre de usuario.
- ***ca_cert***: Ruta del certificado de la Autoridad de Certificación (CA).

- ***client_cert***: Ruta del certificado del cliente.
- ***private_key***: Ruta de la clave privada del cliente.
- ***private_key_passwd***: Contraseña para la clave privada del cliente.

9. REFERENCIAS

1. "IEEE Standard for Local and metropolitan area networks - Media Access Control (MAC) Security", IEEE Std 802.1AE-2018, 2018. Disponible Online: https://ieeexplore.ieee.org/document/8585421
2. "IEEE Standard for Local and Metropolitan Area Networks--Port-Based Network Access Control", IEEE Std 802.1X-2020, 2020. Disponible Online: https://ieeexplore.ieee.org/document/9018454
3. "MACsec: a different solution to encrypt network traffic", 14 Octubre 2016. Disponible Online: https://developers.redhat.com/blog/2016/10/14/macsec-a-different-solution-to-encrypt-network-traffic/
4. "How MACsec works", 10 Mayo 2019. Disponible Online: https://docs.arris.com/bundle/fastiron-08090-licenseguide/page/GUID-EBB8AA84-C558-4A12-82F5-3A947FD66CBE.html
5. "GCM Cipher Suites with Extended Packet Numbering", 18 Julio 2011. Disponible Online: https://www.ieee802.org/1/files/public/docs2011/new-seaman-macsec-xpn-0711-v1.pdf
6. "Linux Based Implementation of MACsec Key Agreement (MKA)", Septiembre 2012. Disponible Online: https://pdfs.semanticscholar.org/3c14/f70c95eb722454dd1d0d0765c5fb194c24b3.pdf
7. "Network Device Collaborative Protection Profile (NDcPP) Extended Package MACsec Ethernet Encryption", 10 Mayo 2016. Disponible Online: https://www.niap-ccevs.org/MMO/PP/pp_ndcpp_macsec_ep_v1.2.pdf
8. "Identity-Based Networking Services: MAC Security. Deployment Guide", Cisco. Mayo 2011. Disponible Online: https://www.cisco.com/c/en/us/products/collateral/ios-nx-os-software/identity-based-networking-services/deploy_guide_c17-663760.pdf
9. "MACsec: Encryption for wired LAN", Sabrina Dubroca. Febrero 2016. Disponible Online: https://netdevconf.info/1.1/proceedings/papers/MACsec-Encryption-for-the-wired-LAN.pdf
10. "Innovations in Ethernet Encryption (802.1AE - MACsec) for Securing High Speed (1-100GE) WAN Deployments", Craig Hill y Stephen Orr, Cisco. 2016. Disponible Online: https://www.cisco.com/c/dam/en/us/td/docs/solutions/Enterprise/Security/MACsec/WP-High-Speed-WAN-Encrypt-MACsec.pdf
11. "MACsec Switch-host Encryption with Cisco AnyConnect and ISE Configuration Example", Cisco 2014. Disponible Online: https://www.cisco.com/c/en/us/support/docs/lan-switching/8021x/117277-config-anyconnect-00.html#anc2

Contacto

Correo electrónico CCN-PYTEC	**ccn-pytec@cni.es**
Twitter	**@CCNPYTEC**
LinkedIn	**https://www.linkedin.com/company/CCN-PYTEC**
Catálogo CPSTIC	**Enlace web**

El Departamento de Productos y Tecnologías de Seguridad TIC del Centro Criptológico Nacional (CCN-PyTec) promueve el desarrollo, la evaluación, la certificación y el uso de productos para garantizar la seguridad de los sistemas de tecnologías de la información y la comunicación.

Praeventio sit vincere