Configuración y resolución de problemas de MKA con Secure Client 5

Contenido

Introducción

Este documento describe cómo configurar el encripción MACsec en un extremo utilizando Secure Client 5 como suplicante.

Prerequisites

Cisco recomienda conocimientos sobre los siguientes temas:

- Identity Services Engine
- 802.1x y Radius
- Cifrado MACsec MKA
- Secure Client versión 5 (conocido anteriormente como Anyconnect)

Componentes Utilizados

La información que contiene este documento se basa en las siguientes versiones de software y hardware.

- Identity Services Engine (ISE) versión 3.3
- Catalyst 9300 versión 17.06.05
- Cisco Secure Client 5.0.4032

Antecedentes

MACSec (Media Access Control Security) es un estándar de seguridad de red que proporciona cifrado y protección para tramas Ethernet en la capa 2 del modelo OSI (Data Link), definido por el estándar IEEE 802.1AE.

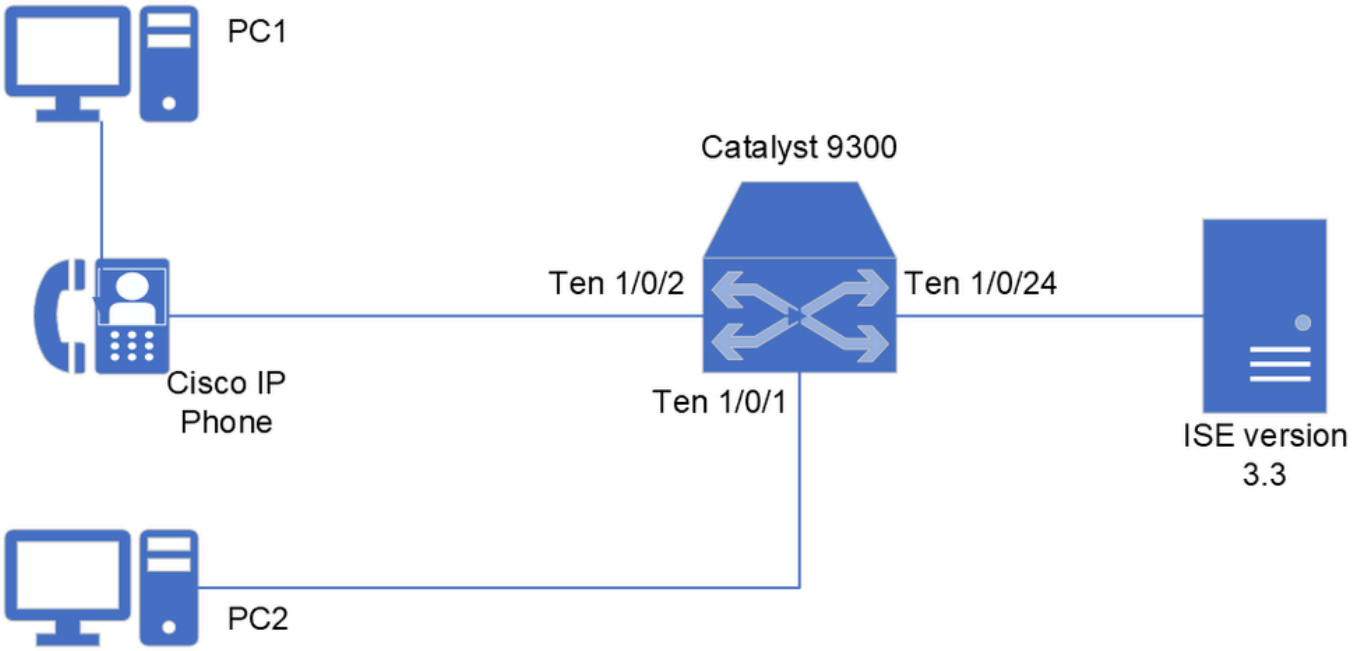
MACSec proporciona este cifrado en una conexión punto a punto que puede ser de switch a switch o de switch a host, por lo que la cobertura de este estándar se limita a las conexiones por cable.

Este estándar cifra todos los datos excepto la dirección MAC de origen y destino de las tramas que se transmiten en una conexión de capa 2.

El protocolo MACsec Key Agreement (MKA) es el mecanismo desde el que los pares MACsec negociarán las claves de seguridad necesarias para proteger el enlace.

Configurar

Diagrama de la red



Nota: Esta documentación considera que tiene reglas configuradas y en funcionamiento para la autenticación Radius para los dispositivos PC y el teléfono IP de Cisco. Para configurar una configuración desde cero, consulte la Guía de implementación prescriptiva de ISE Secure Wired Access para revisar la configuración en Identity Services Engine y el switch para el acceso a la red basado en identidad.

Configuración de políticas en ISE

La primera tarea consiste en configurar los perfiles de autorización correspondientes que se aplican a ambos PC (así como al teléfono IP de Cisco) que se muestran en el diagrama anterior.

En esta situación hipotética, los PC utilizarán el protocolo 802.1X como método de autenticación y el teléfono IP de Cisco utilizará la omisión de dirección MAC (MAB).

ISE se comunica con el switch a través del protocolo Radius acerca de los atributos que el switch debe aplicar en la interfaz desde la que se conecta el terminal a través de una sesión Radius.

Para el cifrado MACsec en hosts, el atributo requerido es cisco-av-pair = linksec-policy, que tiene estos 3 valores posibles:

- No debe ser seguro: El switch no realiza el encripción MKA en la interfaz donde se está

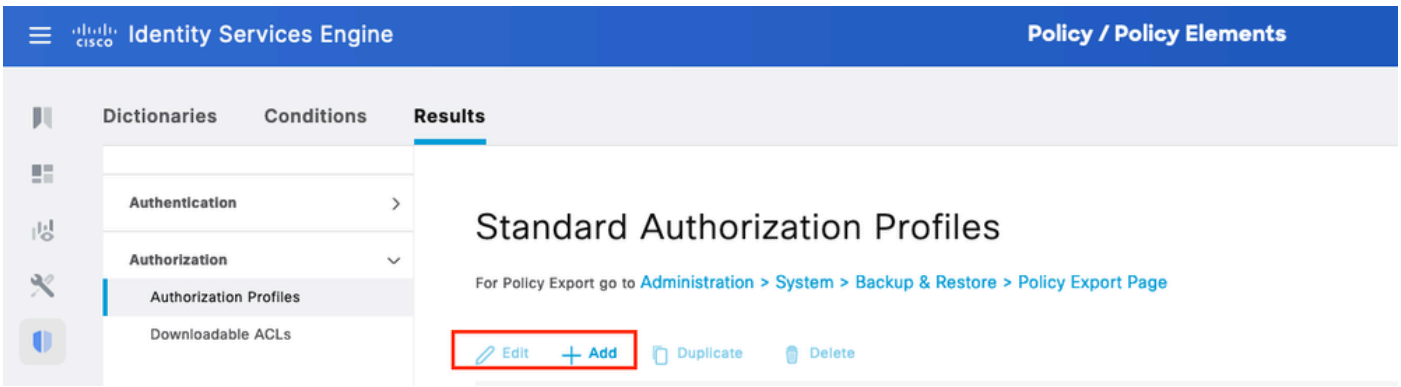
llevando a cabo la sesión Radius.

- Seguridad imprescindible: El switch debe aplicar el cifrado en el tráfico vinculado con la sesión Radius. Si se produce un error en la sesión MKA (o se agota el tiempo de espera), la conexión se considera un error de autorización y se realiza otro intento para establecer la sesión MKA.
- Seguros a prueba de fallos: El switch intenta realizar el cifrado MKA. Si la sesión MKA vinculada a la sesión Radius se realiza correctamente, el tráfico se cifra. Si el MKA falla o se agota el tiempo de espera, el switch permite ese tráfico sin cifrar vinculado a esa sesión Radius.

Paso 1. En ambos PC, aplique una política MKA segura para tener flexibilidad en el caso de que una máquina sin capacidades MKA se conecte a la interfaz Ten 1/0/1.

Opcionalmente, puede configurar una política para PC2 que aplique una política de seguridad obligatoria.

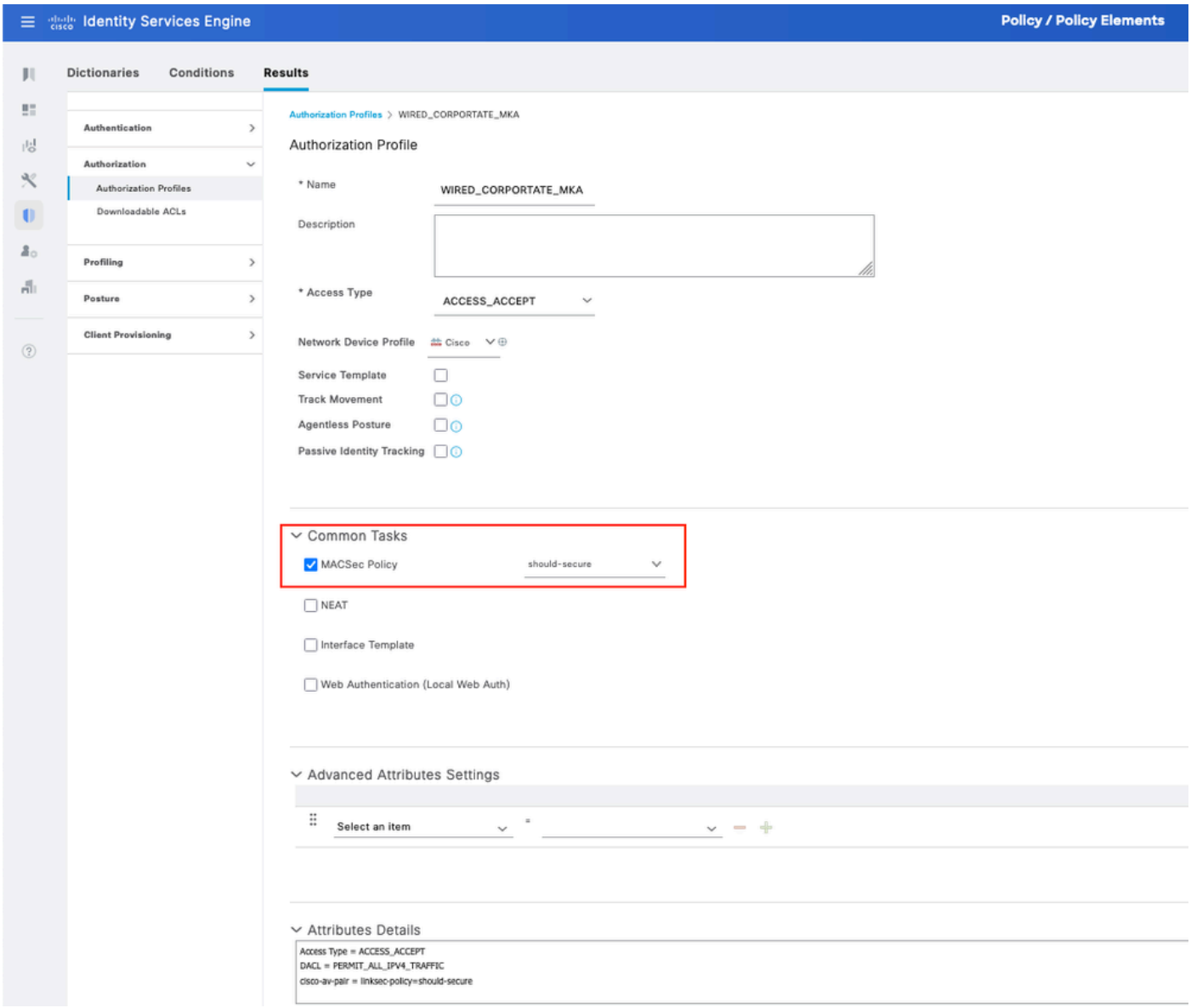
En este ejemplo, configure la política para los PC como en Política > Elementos de política > Resultados > Perfiles de autorización y luego +Agregar o Editar un perfil existente



Paso 2. Complete o personalice los campos requeridos para el perfil.

Asegúrese de que en Tareas comunes ha seleccionado Política MACSec y la política correspondiente para aplicar.

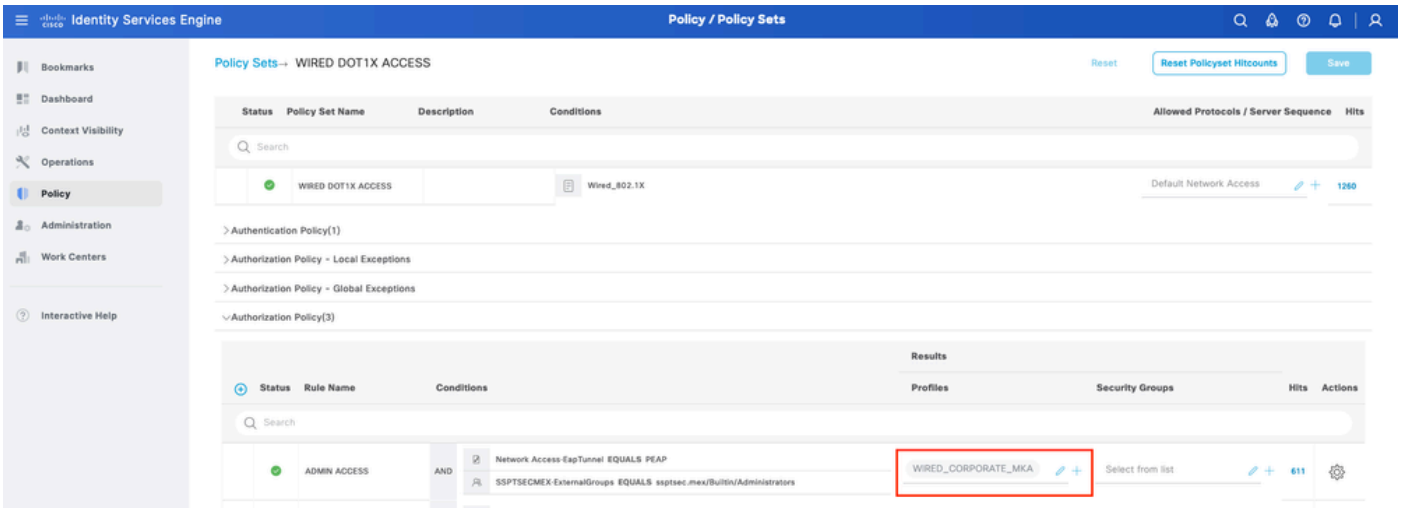
Desplácese hacia abajo y guarde la configuración.



Paso 3. Asigne el perfil de autorización correspondiente a las reglas de autorización que afectan a los dispositivos.

Navegue hasta Política > Conjuntos de políticas > (Seleccione Conjunto de políticas asignado) > Política de autorización.

Asocie la regla de autorización al perfil de autorización con la configuración de MACsec. Desplácese hacia abajo y guarde su configuración.



Configuración de MKA en Catalyst 9300

Paso 1. Configure una nueva política MKA como sugiere este ejemplo:

```
!
  mka policy MKA_PC
  key-server priority 0
  no delay-protection
  macsec-cipher-suite gcm-aes-128
  confidentiality-offset 0
  sak-rekey on-live-peer-loss
  sak-rekey interval 0
  include-icv-indicator
  no send-secure-announcements
  no use-updated-eth-header
  no ssci-based-on-sci
!
```

Paso 2. Habilite el cifrado MACsec en la interfaz donde están conectados los PC.

```
!
interface TenGigabitEthernet1/0/1
 macsec
 mka policy MKA_PC
!
```

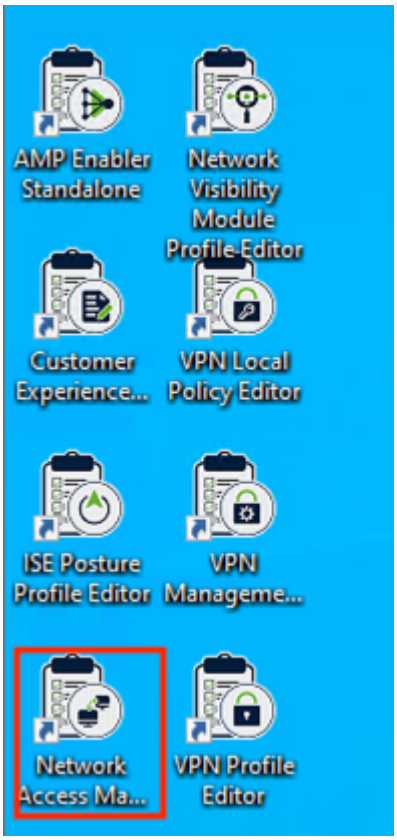
Nota: Para obtener más información relacionada con los comandos y las opciones de la configuración de MKA, revise la guía de configuración de seguridad correspondiente a la versión del switch que utilice. En este escenario para este ejemplo, Guía de configuración

de seguridad, Cisco IOS XE Bengaluru 17.6.x (switches Catalyst 9300)

Configuración de MKA mediante el Editor de perfiles del Administrador de acceso de red.

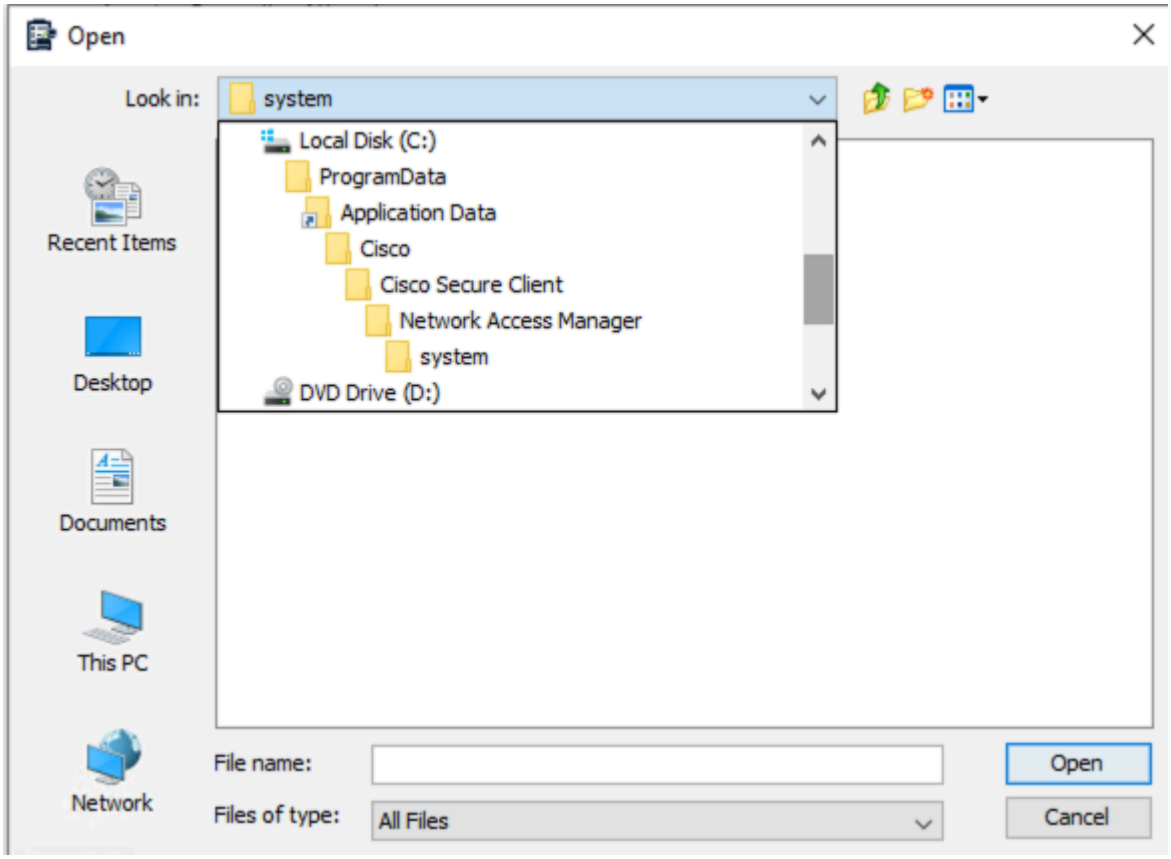
Paso 1. Descárguelo (desde el sitio web de descargas de Cisco) y abra el Editor de perfiles que coincida con la versión de Secure Client que se esté utilizando.

Una vez que haya instalado este programa en su computadora, proceda a abrir Cisco Secure Client Profile Editor - Network Access Manager.

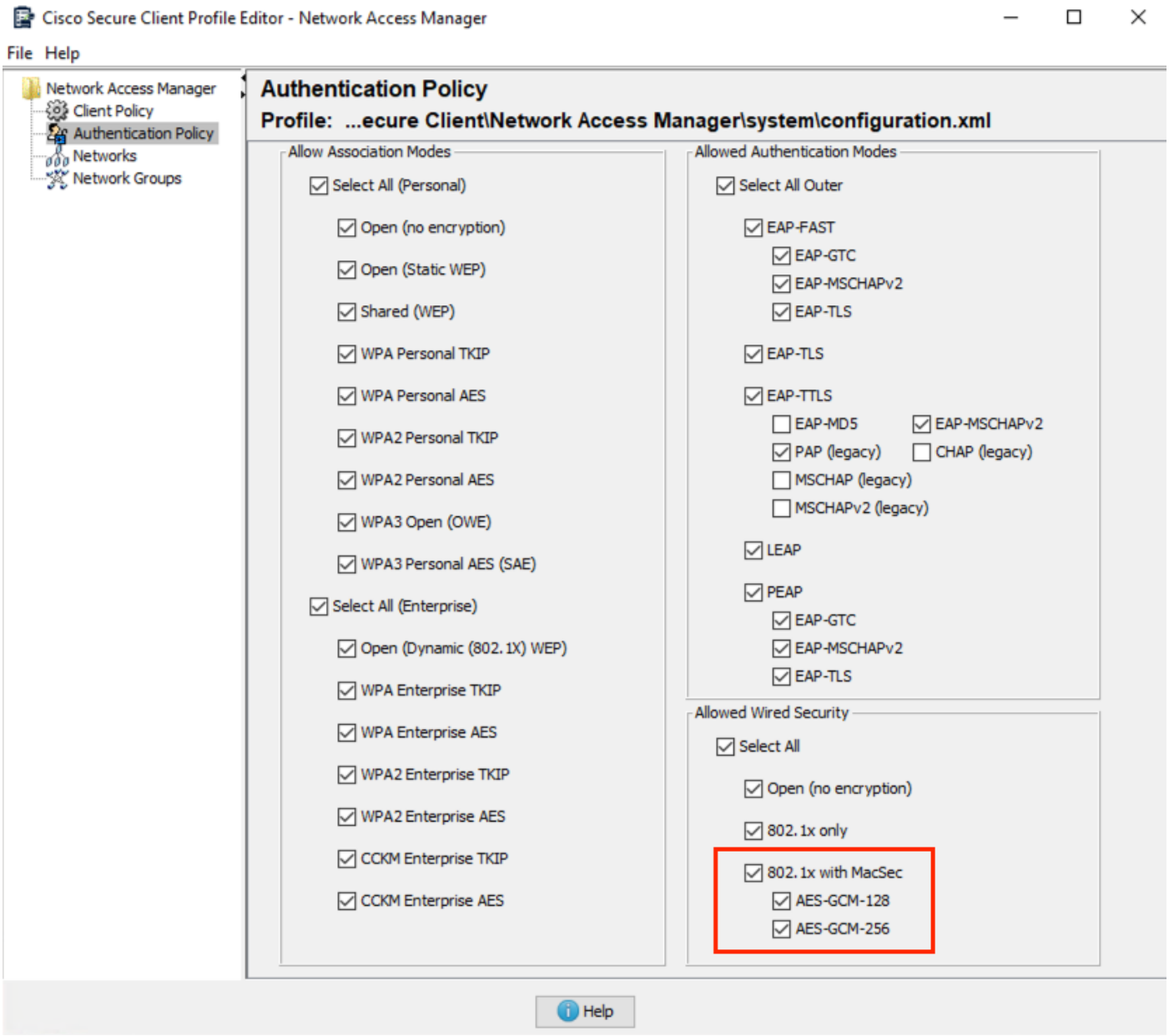


Paso 2. Seleccione Archivo > Abrir.

Paso 3. Seleccione el sistema de carpetas que se muestra en esta imagen. Dentro de esta carpeta, abra el archivo denominado configuration.xml.

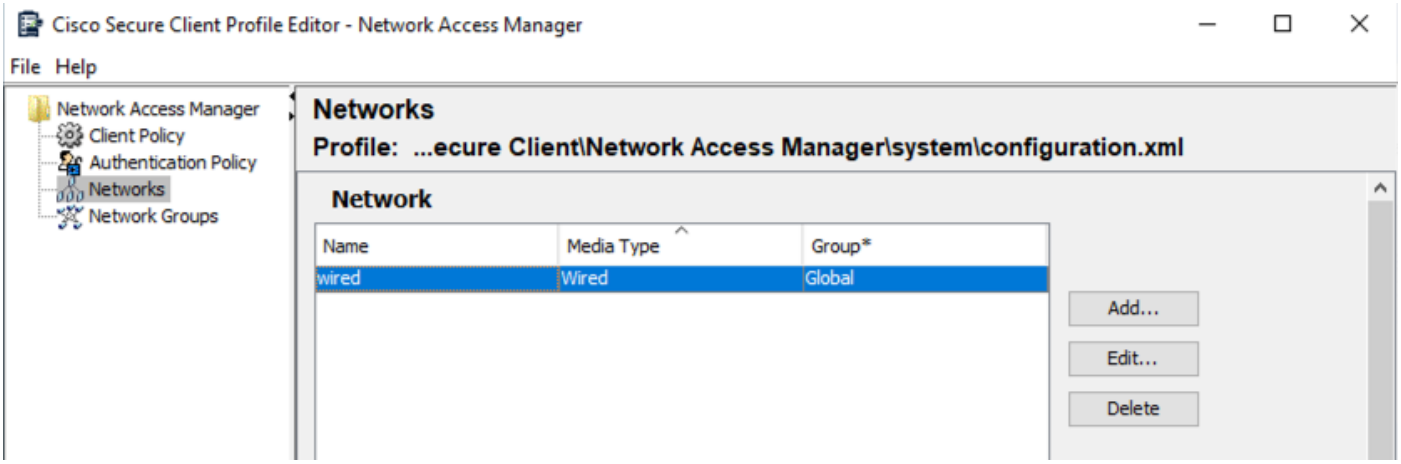


Paso 4. Una vez que el Editor de perfiles ha cargado el archivo, seleccione la opción Directiva de autenticación y asegúrese de que la opción relacionada con 802.1x con MACSec esté habilitada.



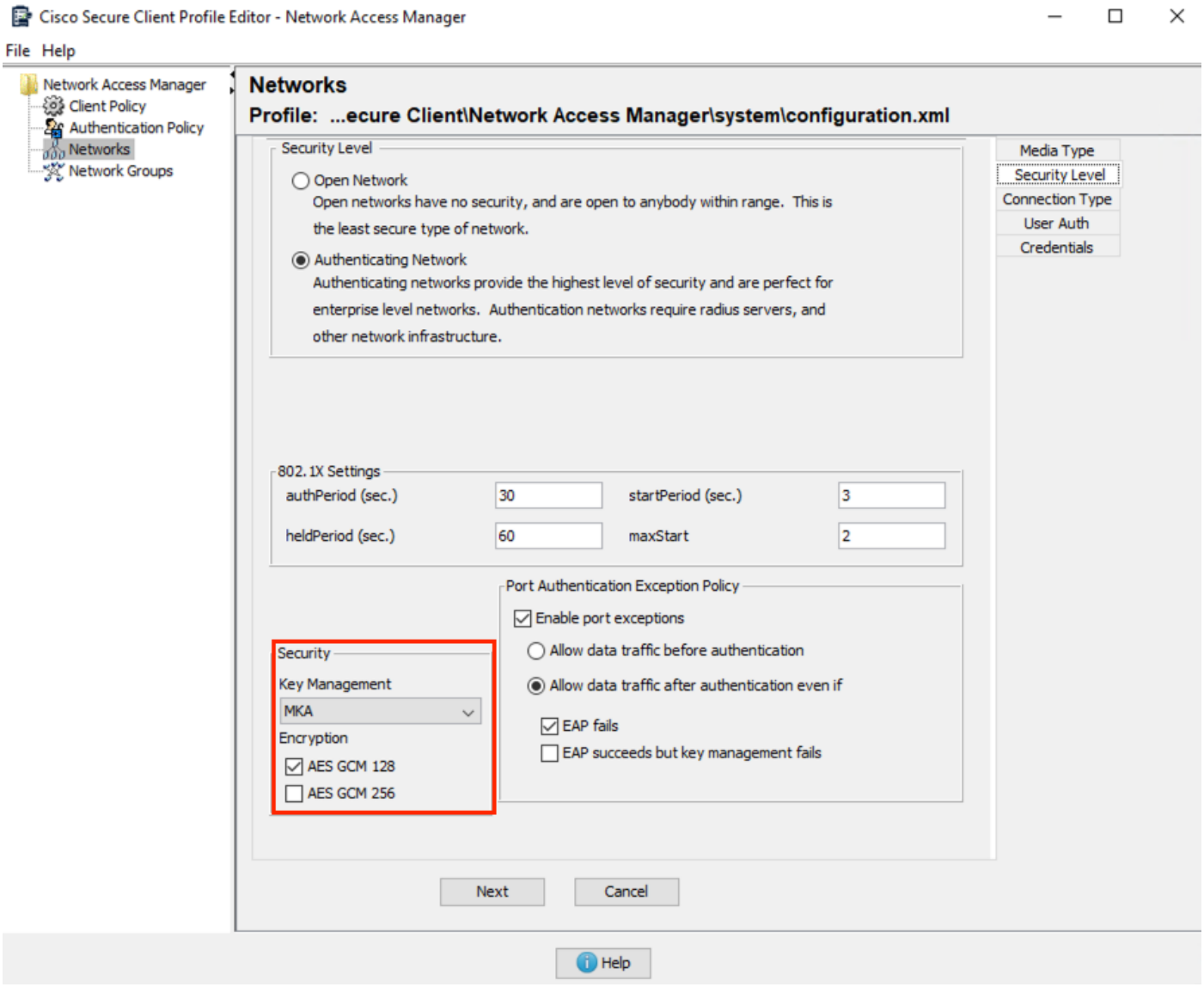
Paso 5. Vaya a la sección Redes. En esta parte, puede Agregar un nuevo perfil para una conexión cableada o Editar el perfil cableado predeterminado que se instala con Secure Client 5.0 .

En esta situación, vamos a Editar el perfil cableado existente.



Paso 6. Configure el perfil. En la sección Nivel de seguridad, ajuste la Administración de claves para utilizar MKA seguido de un cifrado AES GCM 128.

Ajuste también los otros parámetros para el dot1x de autenticación y las políticas.

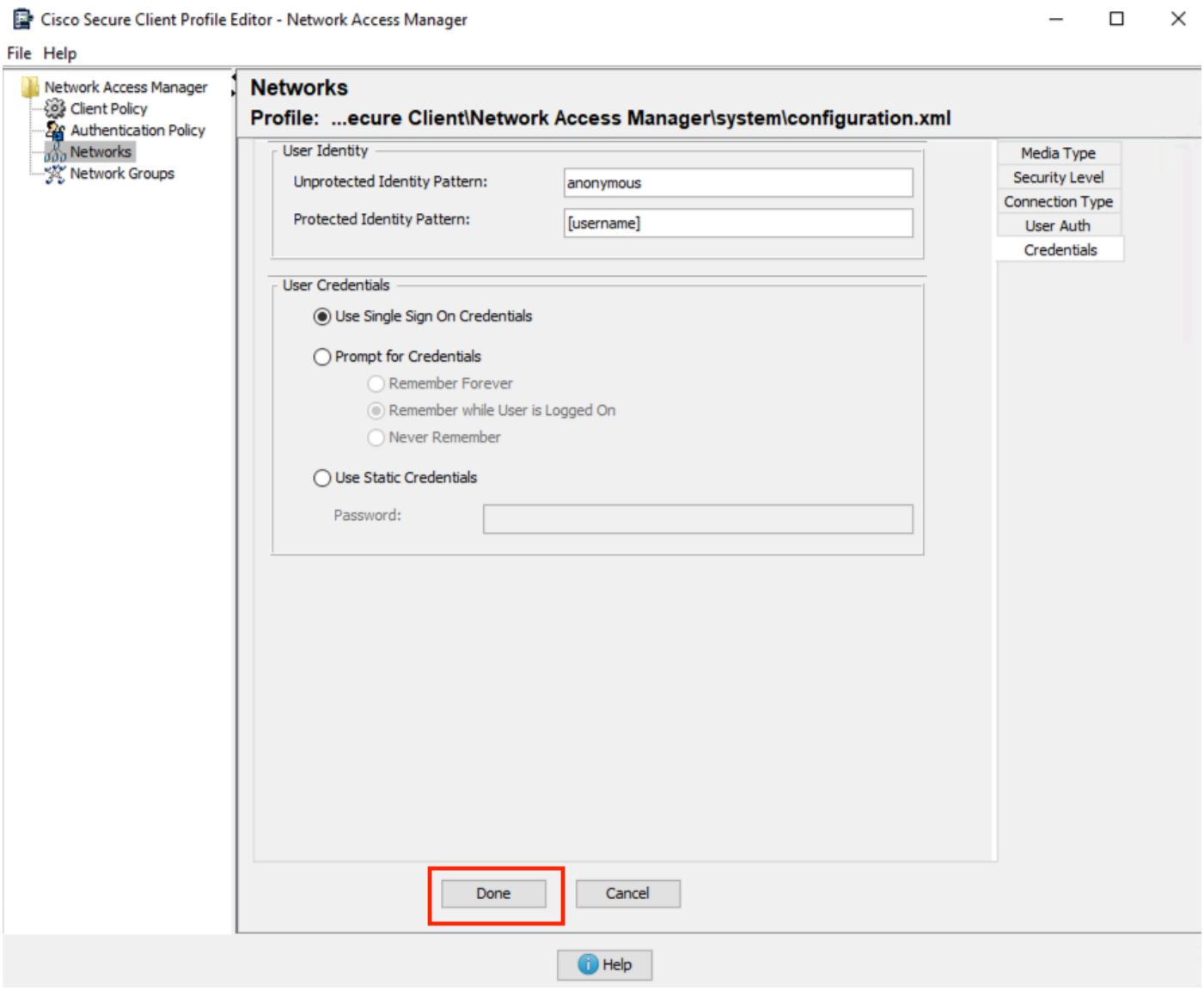


Paso 7. Configure las secciones restantes relativas a Connection Type, User Auth y Credentials.

Esas secciones varían según la configuración de autenticación seleccionada en la sección Nivel de seguridad.

Cuando termine con las configuraciones, haga clic en Finalizado.

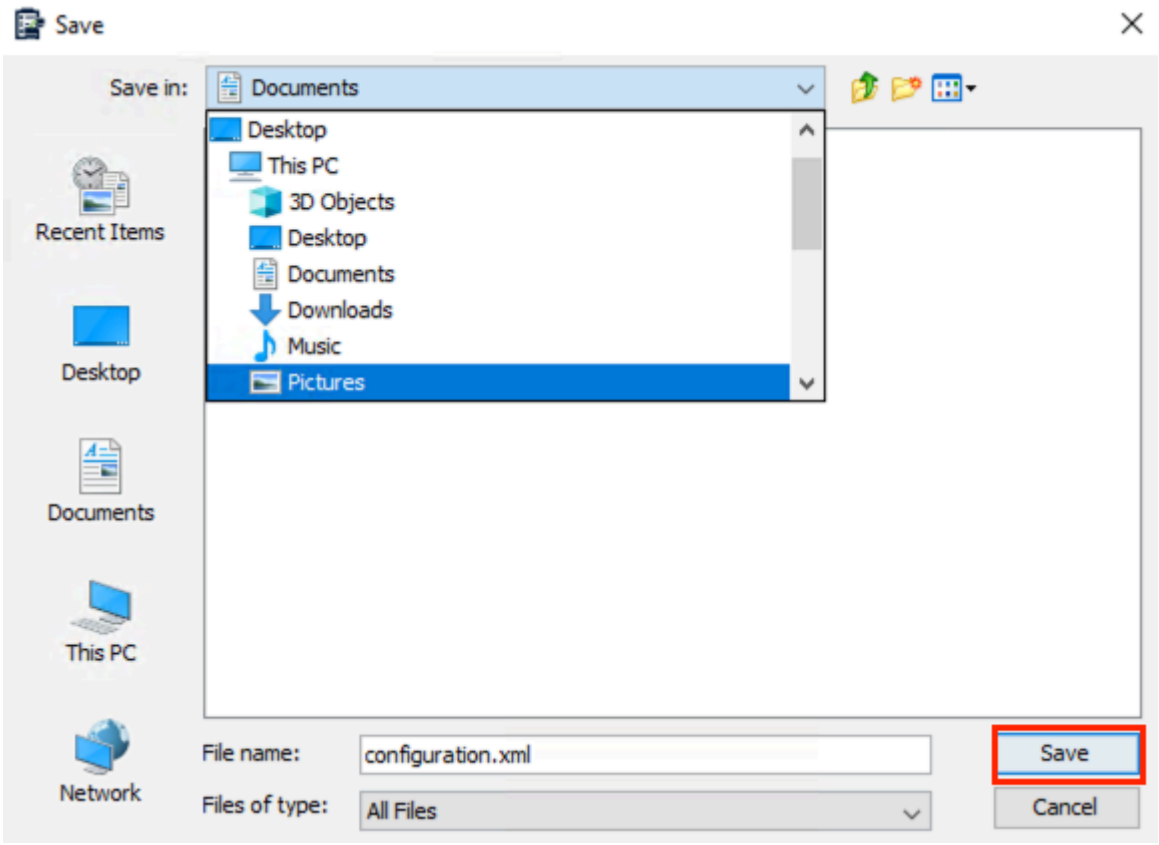
En esta situación, estamos utilizando el protocolo de autenticación extensible protegido (PEAP) con credenciales de usuario.



Paso 8. Navegue al menú Archivo. Continúe con la opción Guardar como.

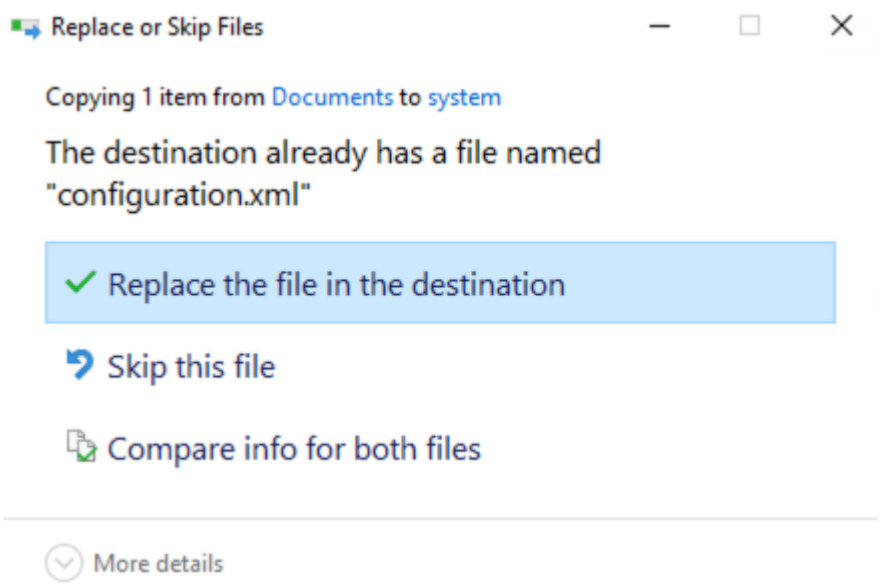
Asigne al archivo el nombre configuration.xml y guárdelo en una carpeta diferente de ProgramData\Cisco\Cisco Secure Client\Network Access Manager\system.

En este ejemplo, el archivo se guardó en la carpeta Documentos. Guarde el perfil.



Paso 8. Vaya a la ubicación del perfil, copie el archivo y reemplace el archivo que se encuentra en la carpeta ProgramData\Cisco\Cisco Secure Client\Network Access Manager\system.

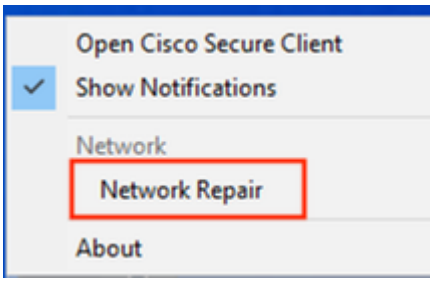
Seleccione la opción Reemplazar el archivo en el destino.



Paso 9. Para cargar el perfil modificado en Security Client 5.0, haga clic con el botón derecho del

ratón en el icono Secure Client situado en la barra de tareas inferior derecha de su equipo con Windows.

Realice una reparación de red.

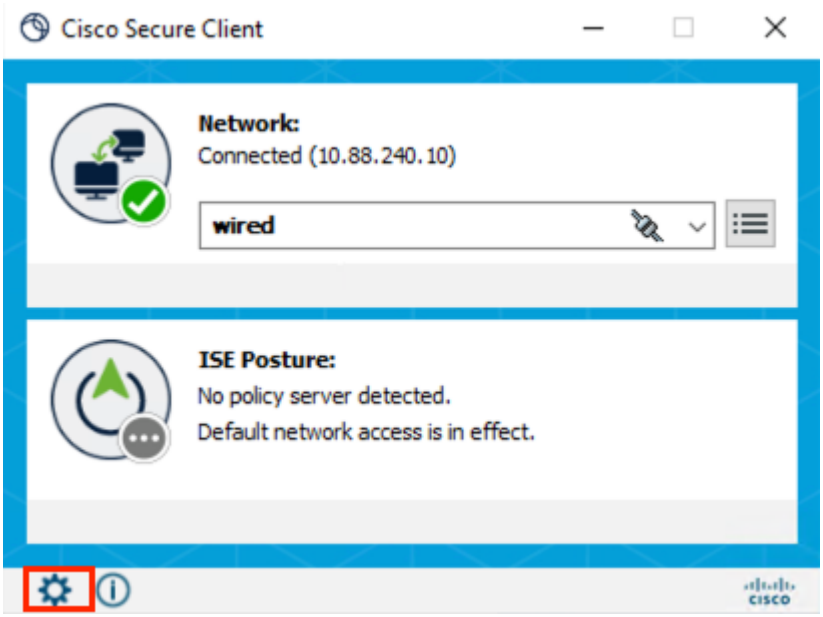


Nota: Todas las redes configuradas a través del editor de perfiles tienen privilegios de Administrator Network, por lo que los usuarios no pueden personalizar/cambiar el contenido configurado con esta herramienta.

Configuración de redes MKA mediante el Administrador de acceso de red (opcional).

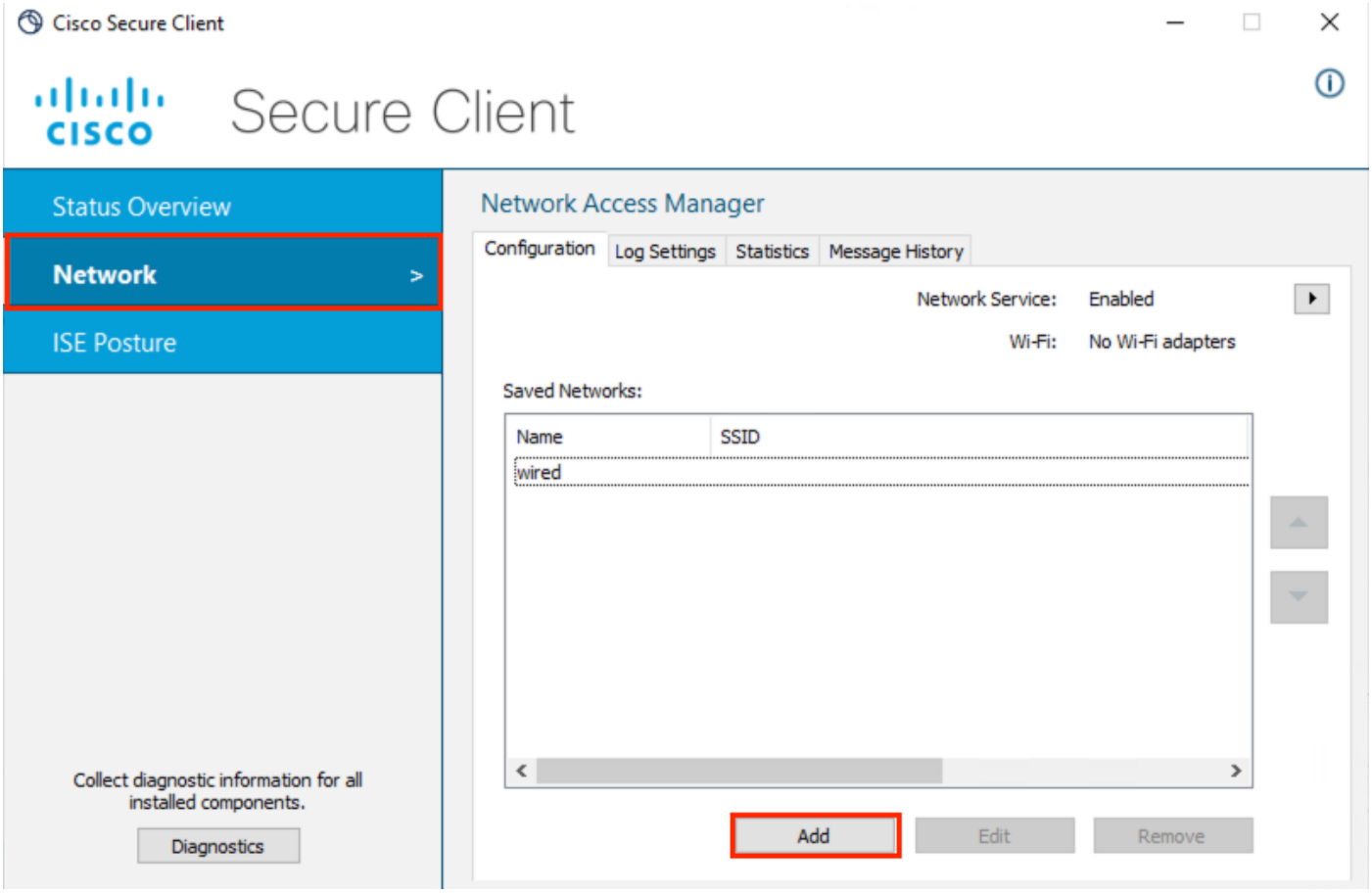
Paso 1. Como alternativa a la configuración de MKA mediante el Editor de perfiles, puede agregar redes sin utilizar esta herramienta.

En el conjunto de aplicaciones Secure Client, seleccione el icono de engranaje.



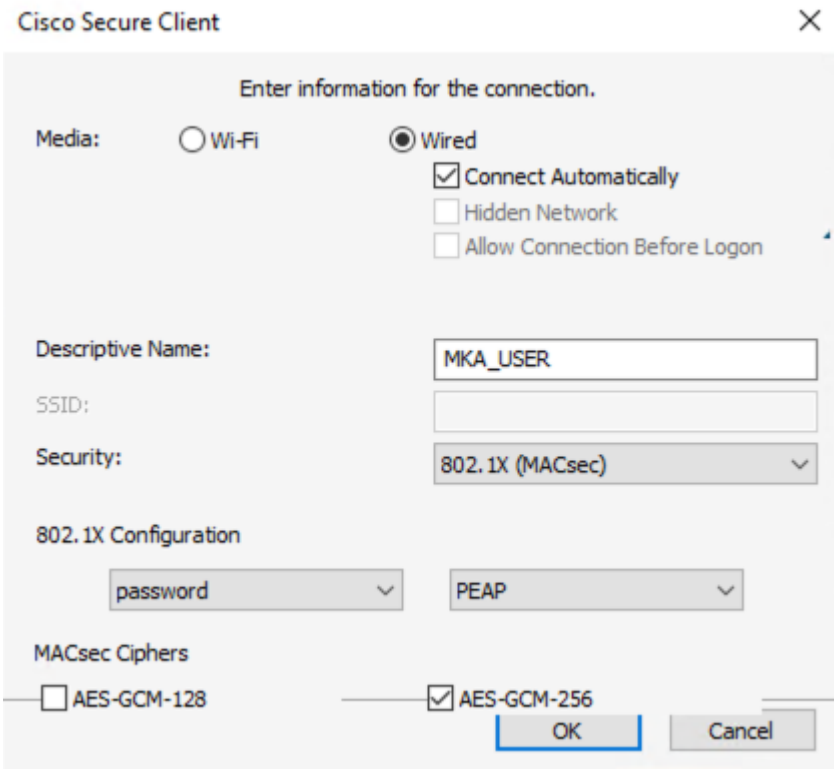
Paso 2. En la nueva ventana que se muestra, seleccione la opción Red.

En la sección Configuración, seleccione la opción Agregar para acceder a un MKA de red con privilegios de Red de Usuario.



Paso 3. En la nueva ventana de configuración, configure las características de su conexión y asigne un nombre a la red.

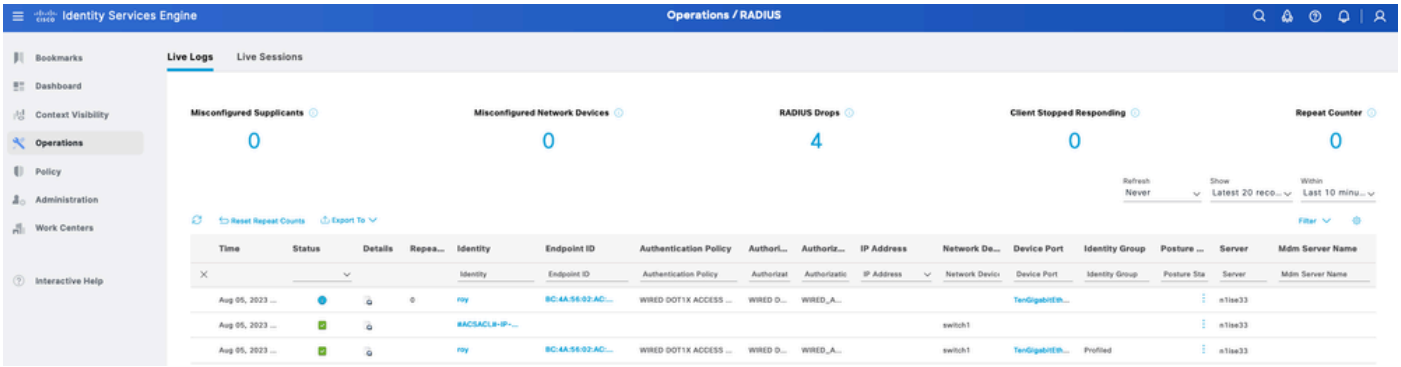
Cuando haya terminado, seleccione el botón OK.



Verificación

Validación en ISE

En ISE, una vez finalizada la configuración de este flujo, observe que el dispositivo está autenticado y autorizado en Livelogs.



Navegue en la sección Detalles de la autenticación y en la sección Resultado.

Los atributos establecidos en el perfil de autorización se envían al dispositivo de acceso a la red (NAD), así como al consumo de una licencia Essential.

cisco-av-pair	linksec-policy=should-secure
cisco-av-pair	ACS:CiscoSecure-Defined-ACL=#ACSACL#-IP-PERMIT_ALL_IPV4_TRAFFIC-57f6b0d3
MS-MPPE-Send-Key	****
MS-MPPE-Recv-Key	****
LicenseTypes	Essential license consumed.

Validación en el switch Catalyst.

Estos comandos se pueden utilizar para validar la funcionalidad adecuada de esta solución.

```
switch1#show mka policy
```

```
switch1#show mka policy

MKA Policy defaults :
         Send-Secure-Announcements: DISABLED

MKA Policy Summary...

Codes : CO - Confidentiality Offset, ICVIND - Include ICV-Indicator,
        SAKR OLPL - SAK-Rekey On-Live-Peer-Loss,
        DP - Delay Protect, KS Prio - Key Server Priority

Policy            KS   DP    CO SAKR  ICVIND Cipher          Interfaces
Name              Prio          OLPL         Suite(s)        Applied
================================================================================
*DEFAULT POLICY*  0    FALSE 0  FALSE TRUE   GCM-AES-128

MKA_PC            0    FALSE 0  FALSE FALSE  GCM-AES-128     Te1/0/1        Te1/0/2
```

```
switch1#show mka session
```

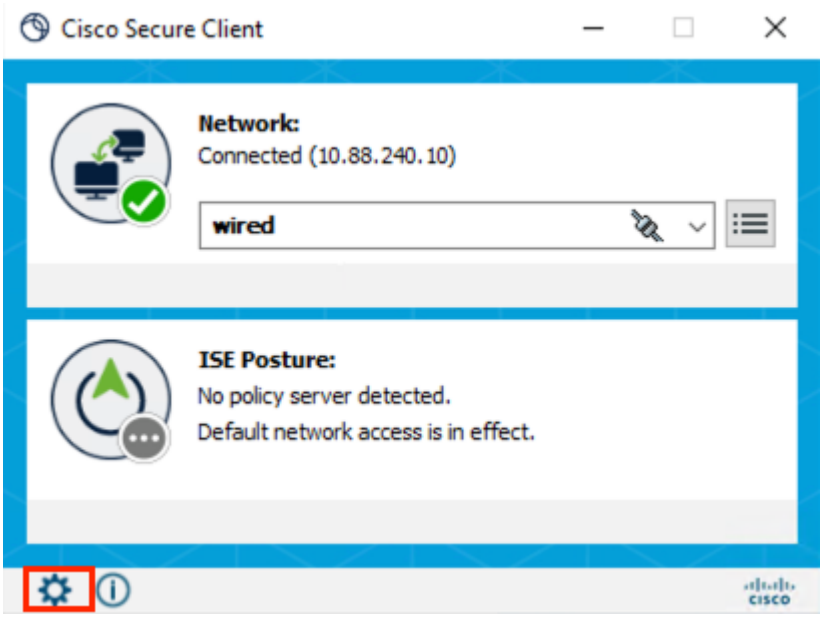
```
switch1#show mka session

Total MKA Sessions....... 2
      Secured Sessions... 2
      Pending Sessions... 0

===============================================================================================
Interface      Local-TxSCI         Policy-Name      Inherited        Key-Server
Port-ID        Peer-RxSCI          MACsec-Peers     Status           CKN
===============================================================================================
Te1/0/1        ac7a.5646.4d01/0002 MKA_PC           NO               YES
2              bc4a.5602.ac25/0000 1                Secured          60E8BC2A9EF5FE11532428862C747640

Te1/0/2        ac7a.5646.4d02/0002 MKA_PC           NO               YES
2              bc4a.5602.ac26/0000 1                Secured          C79300869F539BB534350133064744B6
```

switch1#show authentication session interface <interface_ID> detail

```
switch1#show authentication session interface ten 1/0/2 detail
            Interface:  TenGigabitEthernet1/0/2
                IIF-ID:  0x19FA2D8B
           MAC Address:  bc4a.5602.ac26
          IPv6 Address:
          IPv4 Address:
             User-Name:  alice
                Status:  Authorized
                Domain:  DATA
        Oper host mode:  multi-domain
      Oper control dir:  both
       Session timeout:  N/A
     Common Session ID:  C5AA580A00000030CD72CFE6
       Acct Session ID:  0x00000017
                Handle:  0x62000016
        Current Policy:  POLICY_Te1/0/2


Local Policies:
        Service Template: DEFAULT_LINKSEC_POLICY_SHOULD_SECURE (priority 150)

Server Policies:
      Security Policy:  Should Secure
              ACS ACL: #ACSACL#-IP-PERMIT_ALL_IPV4_TRAFFIC-57f6b0d3
      Security Status:  Link Secured

Method status list:
       Method           State
        dot1x           Authc Success

---------------------------------------

            Interface:  TenGigabitEthernet1/0/2
                IIF-ID:  0x101218F4
           MAC Address:  d4ad.bd2a.cbab
          IPv6 Address:
          IPv4 Address:
             User-Name:  D4-AD-BD-2A-CB-AB
                Status:  Authorized
                Domain:  VOICE
        Oper host mode:  multi-domain
      Oper control dir:  both
       Session timeout:  N/A
     Common Session ID:  C5AA580A00000040CD8001B3
       Acct Session ID:  0x0000001a
                Handle:  0xa1000018
        Current Policy:  POLICY_Te1/0/2

Local Policies:
        Service Template: DEFAULT_LINKSEC_POLICY_SHOULD_SECURE (priority 150)
      Security Policy:  Should Secure
      Security Status:  Link Unsecured

Server Policies:
              ACS ACL: xACSACLx-IP-DENY_ALL_IPV4_TRAFFIC-57f6b0d3

Method status list:
       Method           State
          mab           Authc Success
```

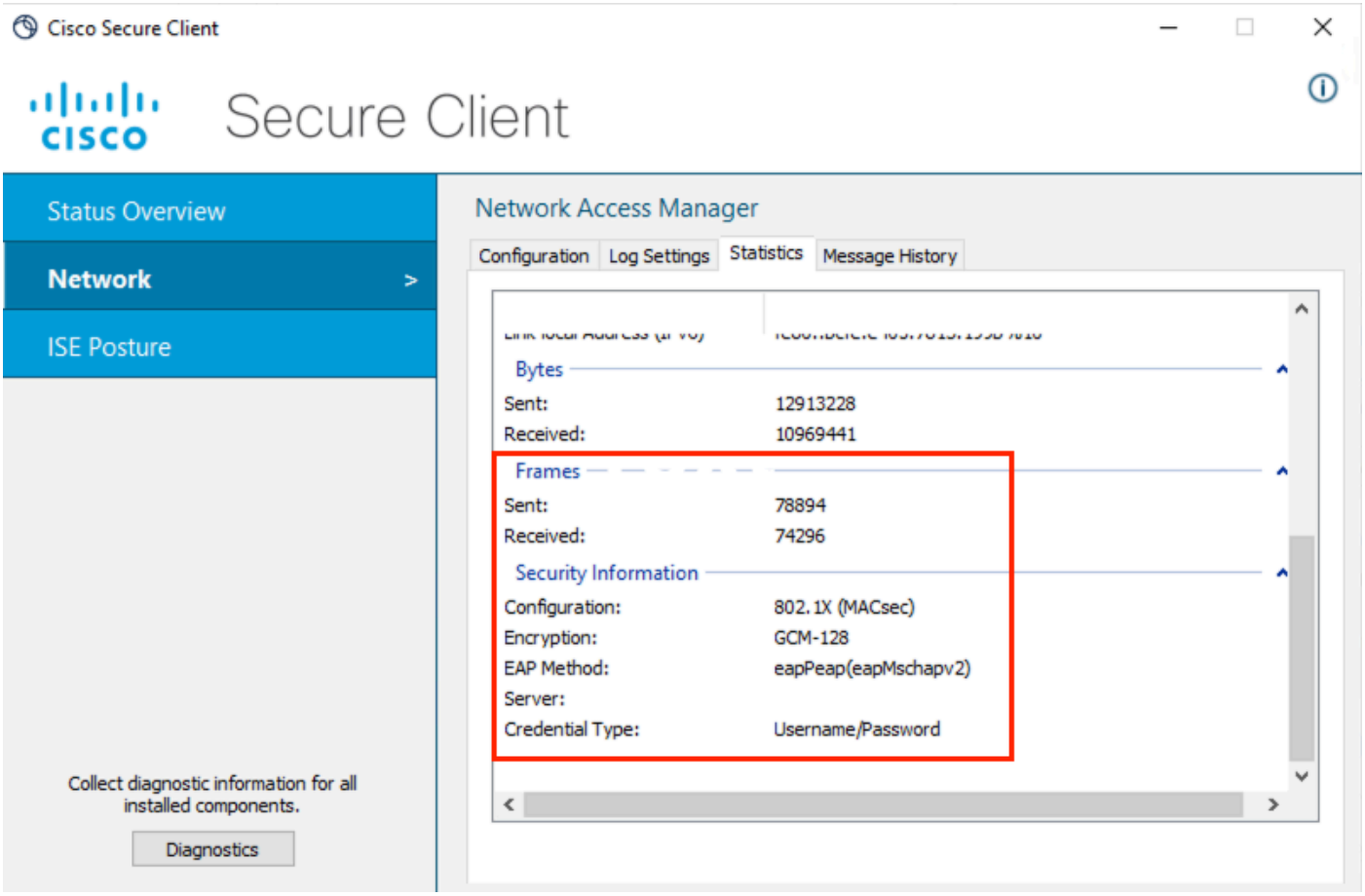
Validación en Secure Client.

La autenticación se realiza correctamente con el perfil creado con cifrado MACsec. Si hace clic en el icono del motor, se mostrará más información.



En el menú que se muestra aquí para Secure Client, en la sección Administrador de acceso de red > Estadísticas, observe el Cifrado y la configuración MACsec correspondiente.

Las tramas recibidas y enviadas aumentan a medida que se realiza el cifrado en la capa 2.

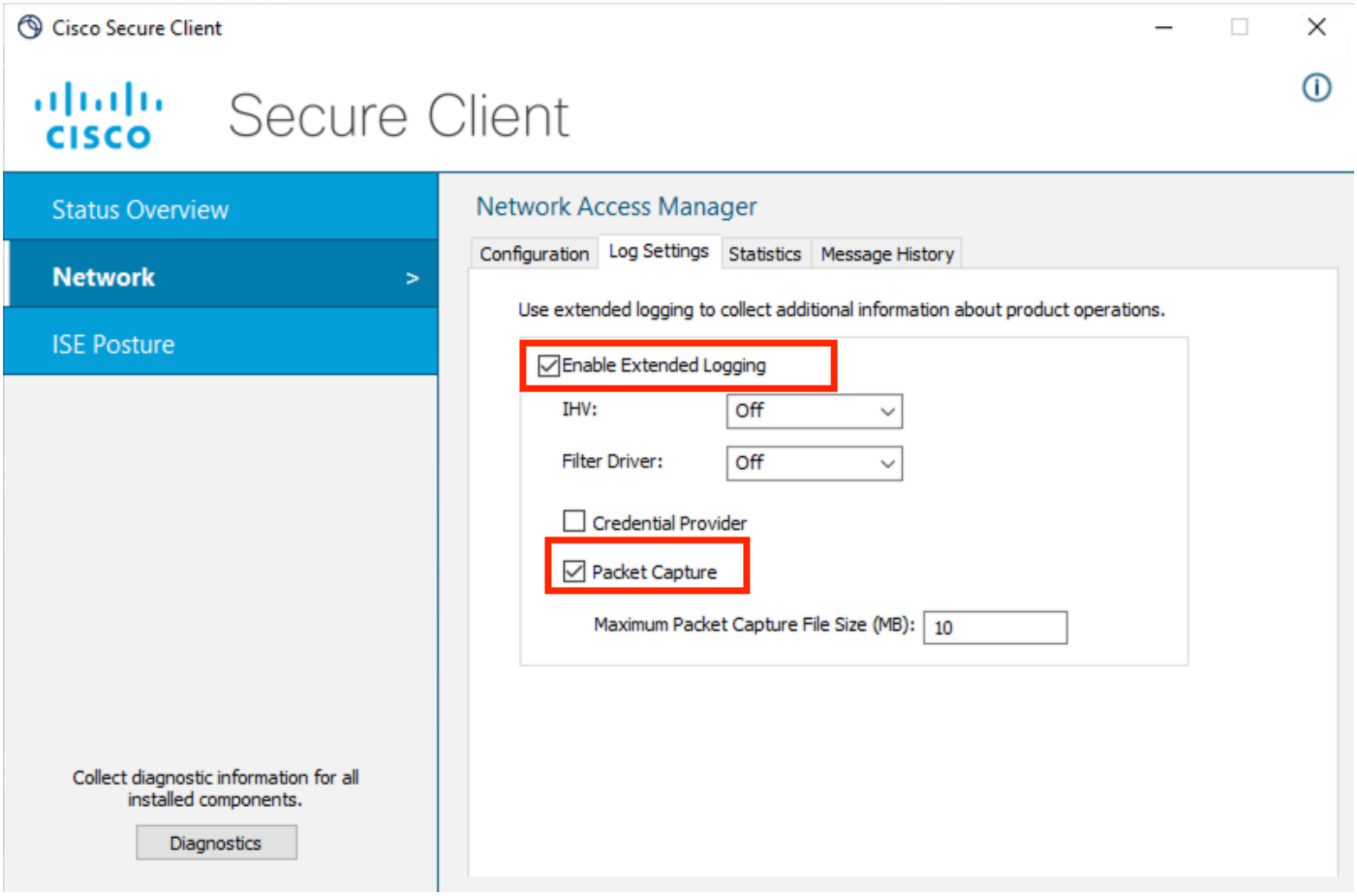


Troubleshoot

Nota: Esta sección cubre la parte de solución de problemas relacionados con los problemas MKA que pueden surgir. Si se enfrenta a un error de autenticación o autorización, consulte la Guía de implementación prescriptiva de acceso por cable seguro de ISE - Resolución de problemas para investigar. Esta guía asume que las autenticaciones funcionan correctamente sin el cifrado MACsec.

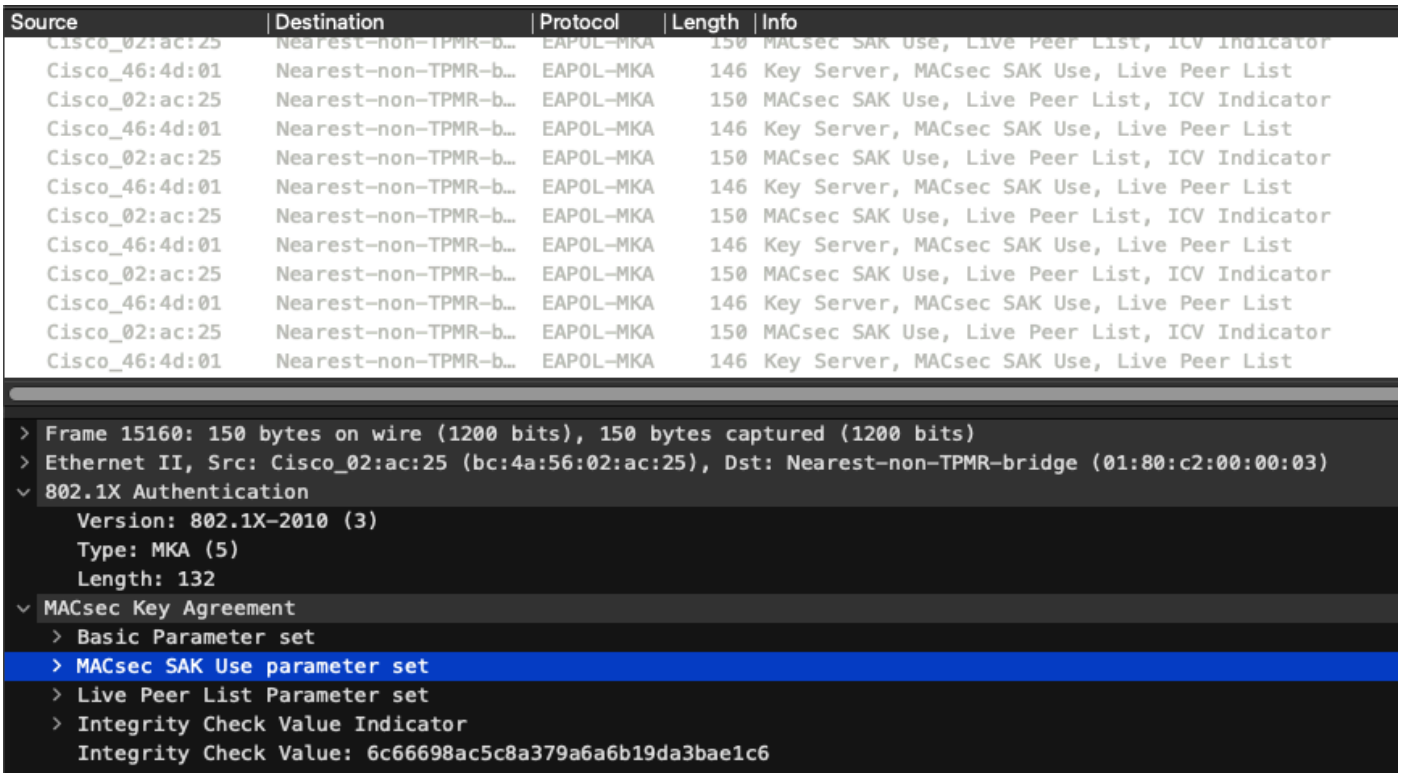
Cisco Secure Endpoint

- Habilite el módulo DART en la suite de terminales seguros.
- En este menú, habilite Registro extendido para recopilar más datos sobre la conexión MKA del usuario. También puede habilitar una captura de paquetes incluida en el paquete DART.



- Recopile el paquete DART para continuar con el análisis de configuration.xml y Network Access Manager. Consulte la documentación Ejecutar DART para recopilar datos para solucionar problemas

Este ejemplo muestra cómo los paquetes se ven como la información entre el host y el switch está encriptada :



En el paquete DART, podemos encontrar información útil para la autenticación 802.1X y la sesión MKA en el registro llamado NetworkAccessManager.txt.

Esta información se muestra en una autenticación correcta con cifrado MKA.

```
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: PORT (1) net: RECV (status: UP, AUTO) (portMsg.c 709)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: 8021X (2) RECEIVED SUCCESS (dot1x_util.c 326)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: STATE (2) current state = AUTHENTICATING (dot1x_sm.c 323)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: STATE (2) S_enterStateAux called with state = AUTHENTICATI
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: STATE (2) staying in 802.1x state: AUTHENTICATING (dot1x_s
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: 8021X (2) smTimer: sec=30 (dot1x_util.c 454)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: STATE (2) eap_type<0>, lengths<4,1496> (dot1x_proto.c 90)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: 8021X (2) smTimer: paused (dot1x_util.c 484)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: EAP (0) Received EAP-Success. (eap_auth_client.c 835)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: EAP (0) tlsAuthOnAuthEnd: clear TLS session (eap_auth_tls_
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: EAP (0) tlsAuthOnAuthEnd: successful authentication, save
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: EAP (3) new credential list saved (eapRequest.c 1485)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: EAP (3) EAP status: AC_EAP_STATUS_EAP_SUCCESS (eapMessage.
%csc_nam-7-DEBUG_MSG: %[tid=9028]: EAP-CB: EAP status notification: session-id=1, handle=04B2DD44, stat
%csc_nam-7-DEBUG_MSG: %[tid=9028]: EAP-CB: sending EapStatusEvent...
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: PORT (2) EAP response received. <len:400> <res:2> (dot1x_p
%csc_nam-7-DEBUG_MSG: %[tid=2716]: EAP: ...received EapStatusEvent: session-id=1, EAP handle=04B2DD44,
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: 8021X (2) smTimer: activated (dot1x_util.c 503)
%csc_nam-6-INFO_MSG: %[tid=2716]: EAP: Eap status AC_EAP_STATUS_EAP_SUCCESS.
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: STATE (2) current state = AUTHENTICATING (dot1x_sm.c 323)
%csc_nam-7-DEBUG_MSG: %[tid=2716]: EAP: processing EapStatusEvent in the subscriber
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: STATE (2) dot1x->eapSuccess is True (dot1x_sm.c 352)
%csc_nam-7-DEBUG_MSG: %[tid=2716]: Auth[wired:user-auth]: Enabling fast reauthentication
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: STATE (2) SUCCESS (dot1x_sm.c 358)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: STATE (2) S_enterStateAux called with state = AUTHENTICATE
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: STATE (2) S_enterStateAux calling sm8Event8021x due to aut
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: 8021X (2) smTimer: disabled (dot1x_util.c 460)
```

```
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: MKA (0) NASP: dot1xAuthSuccessEvt naspStopEapolAnnouncement
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: NASP (0) >> NASP: naspStopEapolAnnouncement (nasp.c 900)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: NASP (0) << NASP: naspStopEapolAnnouncement.  err = 0 (nasp
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: 8021X (2) dot1x->config.useMka = 1 (dot1x_main.c 829)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: INF (2) >> MKA: StartSession (mka.c 511)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: MKA (2) MKA: bUseMka = 1, bUseMacSec = 1706033334, MacsecSu
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: MKA (2) >> MKA: InitializeContext (mka.c 1247)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: INF (2) MKA: Changing state to Unconnected (mka.c 1867)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: INF (2) MKA: Changing Sak State to Idle (mka.c 1271)
%csc_nam-7-DEBUG_MSG: %[tid=2716]: Auth[wired:user-auth]: Fast reauthentication enabled on authenticati
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: MKA (2) << MKA: InitializeContext (mka.c 1293)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: MKA (2) MKA: Changing state to Need Server (mka.c 1871)
%csc_nam-7-DEBUG_MSG: %[tid=2716]: Auth[wired:user-auth]: Sending NOTIFICATION__SUCCESS to subscribers
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: INF (2) >> MKA: CreateKeySet (mka.c 924)
%csc_nam-7-DEBUG_MSG: %[tid=2716]: Network auth request NOTIFICATION__SUCCESS
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: NASP (0) >> NASP: naspGetNetCipherSuite (nasp.c 569)
%csc_nam-6-INFO_MSG: %[tid=9028][comp=SAE]: MKA (2) MKA: Key length is 16 bytes (mka.c 954)
%csc_nam-7-DEBUG_MSG: %[tid=2716]: Auth[wired:user-auth]: Finishing authentication
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: MKA (2) MKA: MyMac (mka.c 971)
%csc_nam-7-DEBUG_MSG: %[tid=2716]: Auth[wired:user-auth]: Authentication finished
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: API (1) event: STATUS - AC_PORT_STATUS_EAP_SUCCESS (portWo
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: API (1) event: complete (portWorkList.c 130)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: API (1) event: STATUS - AC_PORT_STATUS_MKA_UNCONNECTED (po
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: API (1) event: complete (portWorkList.c 130)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: API (1) event: STATUS - AC_PORT_STATUS_MKA_NEED_SERVER (po
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: API (1) event: complete (portWorkList.c 130)
%csc_nam-7-DEBUG_MSG: %[tid=8140][comp=SAE]: NET (0) SscfCallback(1): SSCF_NOTIFICATION_CODE_SEND_PACKE
%csc_nam-7-DEBUG_MSG: %[tid=8140][comp=SAE]: NET (0) CIMD Event: evtSeq#=0 msg=4 ifIndex=1 len=36 (cimd
%csc_nam-7-DEBUG_MSG: %[tid=8140][comp=SAE]: NET (1) cdiEvt:(3,0) dataLen=4 (cimdEvt.c 358)
%csc_nam-7-DEBUG_MSG: %[tid=8140][comp=SAE]: NET (1) cdiEvt:(3,1) dataLen=102 (cimdEvt.c 358)
%csc_nam-7-DEBUG_MSG: %[tid=8140][comp=SAE]: NET (1) netEvent(1): Recv queued (netEvents.c 91)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: PORT (1) net: RECV (status: UP, AUTO) (portMsg.c 709)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: MKA (2) >> MKA: EapolInput (mka.c 125)
%csc_nam-7-DEBUG_MSG: %[tid=9028][comp=SAE]: MKA (2) MKA: MKPDU In (mka.c 131)
```

Troubleshooting de Cisco IOS

Estos comandos se pueden implementar en el Dispositivo de acceso a la red (NAD) para revisar el cifrado MKA entre la plataforma y el solicitante.

Para obtener más información sobre los comandos, revise la guía de configuración correspondiente de la plataforma utilizada como NAD.

```
#show authentication session interface <interface_ID> detail
#show mka summary
#show mka policy
#show mka session interface <interface_ID> detail
#show macsec summary
#show macsec interface <interface_ID>
#debug mka events
#debug mka errors
#debug macsec event
#debug macsec error
```

Estas son depuraciones de una conexión MKA exitosa a un host. Puede utilizar esto como referencia cuando aparezca :

```
%LINK-3-UPDOWN: Interface TenGigabitEthernet1/0/1, changed state to down
Macsec interface TenGigabitEthernet1/0/1 is UP
MKA-EVENT: Create session event: derived CKN 9F0DC198A9728FB3DA198711B58570E4, len 16
MKA-EVENT EC000025: SESSION START request received...
NGWC-MACSec: pd get port capability is invoked
MKA-EVENT: New MKA Session on Interface TenGigabitEthernet1/0/1 with Physical Port Number 9 is using th
MKA-EVENT: New VP with SCI AC7A.5646.4D01/0002 on interface TenGigabitEthernet1/0/1
MKA-EVENT: Created New CA 0x80007F30A6B46F20 Participant on interface TenGigabitEthernet1/0/1 with SCI
%MKA-5-SESSION_START: (Te1/0/1 : 2) MKA Session started for RxSCI bc4a.5602.ac25/0000, AuditSessionID C
MKA-EVENT: Started a new MKA Session on interface TenGigabitEthernet1/0/1 for Peer MAC bc4a.5602.ac25 w
MKA-EVENT bc4a.5602.ac25/0000 EC000025: FSM (Init MKA Session) - Successfully derived CAK.
MKA-EVENT bc4a.5602.ac25/0000 EC000025: Successfully initialized a new MKA Session (i.e. CA entry) on i
MKA-EVENT bc4a.5602.ac25/0000 EC000025: FSM (Derive KEK/ICK) - Successfully derived KEK...
MKA-EVENT bc4a.5602.ac25/0000 EC000025: FSM (Derive KEK/ICK) - Successfully derived ICK...
MKA-EVENT bc4a.5602.ac25/0000 EC000025: New Live Peer detected, No potential peer so generate the first
MKA-EVENT bc4a.5602.ac25/0000 EC000025: >> FSM - Generate SAK for CA with CKN 9F0DC198 (Latest AN=0, Ol
MKA-EVENT bc4a.5602.ac25/0000 EC000025: Generation of new Latest SAK succeeded (Latest AN=0, KN=1)...
MKA-EVENT bc4a.5602.ac25/0000 EC000025: >> FSM - Install RxSA for CA with CKN 9F0DC198 on VP with SCI A
MKA-EVENT bc4a.5602.ac25/0000 EC000025: Clean up the Rx for dormant peers
MACSec-IPC: send_xable send msg success for switch=1
MACSec-IPC: blocking enable disable ipc req
MACSec-IPC: watched boolean waken up
MACSec-IPC: geting switch number
MACSec-IPC: switch number is 1
MACSec-IPC: create_tx_sc send msg success
Send create_tx_sc to IOMD successfully
alloc_cache called TxSCI: AC7A56464D010002 RxSCI: BC4A5602AC250000
Enabling replication for slot 1 vlan 330 and the ref count is 1
MACSec-IPC: vlan_replication send msg success
Added replication for data vlan 330
MACSec-IPC: geting switch number
MACSec-IPC: switch number is 1
MACSec-IPC: create_rx_sc send msg success
Sent RXSC request to FED/IOMD
MACSec-IPC: geting switch number
MACSec-IPC: switch number is 1
MACSec-IPC: install_rx_sa send msg success
Sent ins_rx_sa to FED and IOMD
MKA-EVENT bc4a.5602.ac25/0000 EC000025: Requested to install/enable new RxSA successfully (AN=0, KN=1 S
%LINEPROTO-5-UPDOWN: Line protocol on Interface TenGigabitEthernet1/0/1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan330, changed state to up
MKA-EVENT bc4a.5602.ac25/0000 EC000025: Sending SAK for AN 0 resp peers 0 cap peers 1
MKA-EVENT bc4a.5602.ac25/0000 EC000025: SAK Wait Timer started for 6 seconds.
MKA-EVENT bc4a.5602.ac25/0000 EC000025: (KS) Received new SAK-Use response to Distributed SAK for AN 0,
MKA-EVENT bc4a.5602.ac25/0000 EC000025: (KS) All 1 peers with the required MACsec Capability have indic
MKA-EVENT: Reqd to Install TX SA for CA 0x80007F30A6B46F20 AN 0 CKN 9F0DC198 - on int(TenGigabitEtherne
MKA-EVENT bc4a.5602.ac25/0000 EC000025: >> FSM - Install TxSA for CA with CKN 9F0DC198 on VP with SCI A
MACSec-IPC: geting switch number
MACSec-IPC: switch number is 1
MACSec-IPC: install_tx_sa send msg success
MKA-EVENT bc4a.5602.ac25/0000 EC000025: Before sending SESSION_SECURED status - SECURED=false, PREVIOUS
MKA-EVENT bc4a.5602.ac25/0000 EC000025: Successfully sent SECURED status for CA with CKN 9F0DC198.
MKA-EVENT: Successfully updated the CKN handle for interface: TenGigabitEthernet1/0/1 with 9F0DC198 (if
%MKA-5-SESSION_SECURED: (Te1/0/1 : 2) MKA Session was secured for RxSCI bc4a.5602.ac25/0000, AuditSessi
MKA-EVENT: MSK found to be same while updating the MSK and EAP Session ID in the subblock
```

```
MKA-EVENT bc4a.5602.ac25/0000 EC000025: After sending SESSION_SECURED status - SECURED=true, PREVIOUSLY_
```

Resolución de problemas de Identity Services Engine (ISE)

La resolución de problemas relacionada con esta función se limita a la entrega del atributo linksec-policy=should-secure de cisco-av-pair.

Asegúrese de que el resultado de la autorización esté enviando esa información a la sesión Radius vinculada a los puertos de switch donde se conectan los dispositivos.

Para obtener más análisis de autenticación sobre ISE, consulte Troubleshooting and Enable Debugs on ISE

Problemas Comunes

Discordancia de cifrado

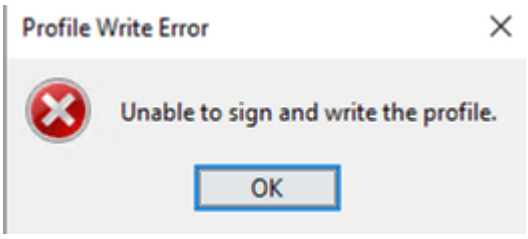
Este registro se puede ver en las depuraciones de MKA en NAD.

```
MKA-4-MKA_MACSEC_CIPHER_MISMATCH: (Te1/0/1 : 30) Lower strength MKA-cipher than macsec-cipher for RxSCI
```

Lo primero que se debe verificar en este escenario es que los cifrados configurados en la política MKA en el switch y en el perfil de Secure Client coinciden.

Para el caso del cifrado AES-GCM-256, estos requisitos deben cumplirse (según la documentación) Guía del administrador de Cisco Secure Client (incluido AnyConnect), versión 5

Incapacidad para guardar configuration.xml.



Este problema relacionado con Profile Write Error se resuelve guardando el archivo configuration.xml (como se describió anteriormente) denominado Setup of MKA mediante Network Access Manager Profile Editor.

El error está relacionado con que el archivo configuration.xml en uso no se puede modificar. Por lo tanto, guarde el archivo en otra ubicación para continuar con la sustitución de los perfiles.

Información Relacionada

- Configuración del cifrado MACsec
- Configuración de MACsec Switch a Host con Cat9k e ISE

Acerca de esta traducción
Cisco ha traducido este documento combinando la traducción automática y los recursos humanos a fin de ofrecer a nuestros usuarios en todo el mundo contenido en su propio idioma.

Tenga en cuenta que incluso la mejor traducción automática podría no ser tan precisa como la proporcionada por un traductor profesional.

Cisco Systems, Inc. no asume ninguna responsabilidad por la precisión de estas traducciones y recomienda remitirse siempre al documento original escrito en inglés (insertar vínculo URL).