# Configurar e solucionar problemas do MKA usando Secure Client 5

## Contents



## Introdução

Este documento descreve como configurar a criptografia MACsec em um endpoint usando o Secure Client 5 como solicitante.

### Pré-requisitos

A Cisco recomenda conhecimento sobre estes tópicos:

- Identity Services Engine
- 802.1x e Radius
- Criptografia MACsec MKA
- Secure Client versão 5 (anteriormente conhecido como Anyconnect)

### Componentes Utilizados

As informações neste documento são baseadas nestas versões de software e hardware:

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

### Informações de Apoio

MACSec (Media Access Control Security) é um padrão de segurança de rede que fornece criptografia e proteção para quadros Ethernet na camada 2 do modelo OSI (Enlace de Dados), definido pelo padrão IEEE 802.1AE.

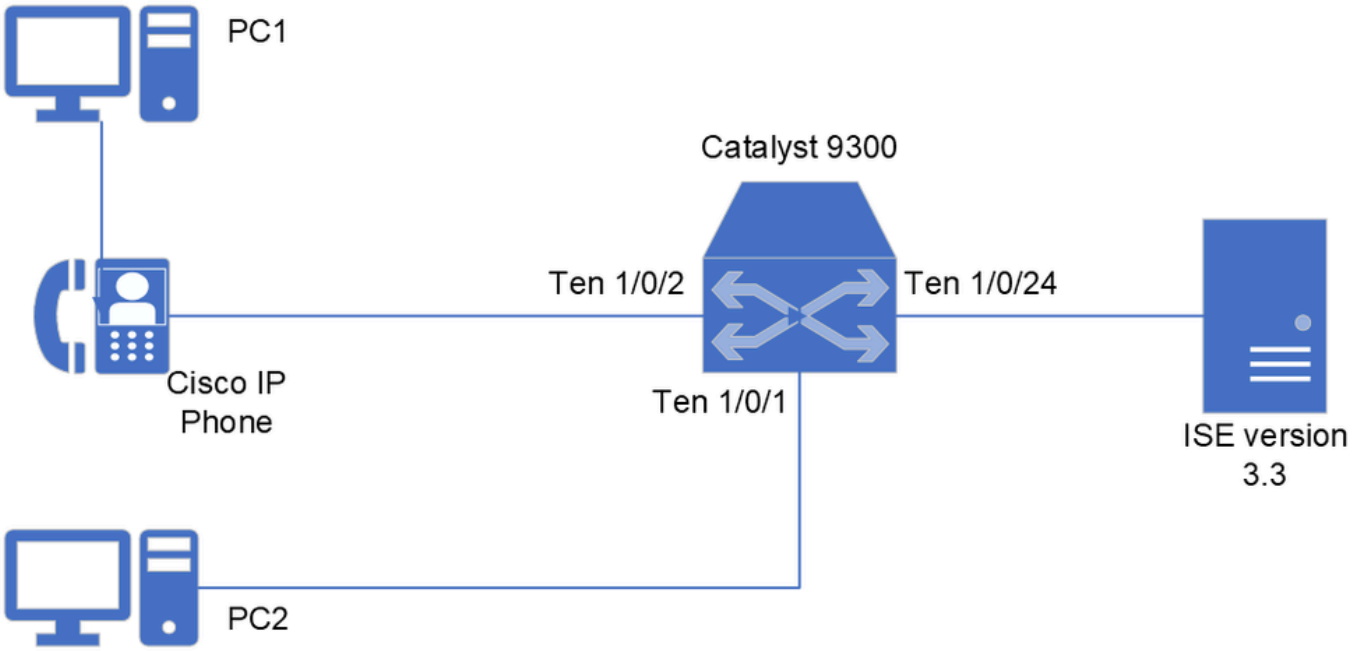
A MACSec fornece essa criptografia em uma conexão ponto-a-ponto que pode ser de switch para switch ou de switch para host, portanto, a cobertura deste padrão é limitada a conexões com fio.

Esse padrão criptografa todos os dados, exceto o endereço MAC origem e destino dos quadros que são transmitidos em uma conexão da camada 2.

O protocolo MACsec Key Agreement (MKA) é o mecanismo a partir do qual os correspondentes MACsec negociarão as chaves de segurança necessárias para proteger o link.

## Configurar

### Diagrama de Rede




Note: Esta documentação considera que você tem regras configuradas e funcionando para a Autenticação Radius para os dispositivos do PC e o telefone IP da Cisco. Para definir uma configuração do zero, consulte o Guia de implantação prescritiva do acesso com fio seguro do ISE para revisar a configuração no Identity Services Engine e no Switch para acesso de rede baseado em identidade.

## Configuração de políticas no ISE

A primeira tarefa é configurar os perfis de autorização correspondentes que são aplicados a ambos os PCs (assim como ao Telefone IP da Cisco) exibidos no diagrama anterior.

Neste cenário hipotético, os PCs usarão o protocolo 802.1X como o método de autenticação e o telefone IP da Cisco usará Mac Address Bypass (MAB).

O ISE se comunica com o switch através do protocolo Radius sobre os atributos que o switch precisa aplicar na interface de onde o endpoint está conectado através de uma sessão Radius.

Para a criptografia MACsec em hosts, o atributo necessário é cisco-av-pair = linksec-policy, que tem estes três valores possíveis:

- Não deve ser seguro: O switch não executa a criptografia MKA na interface em que a

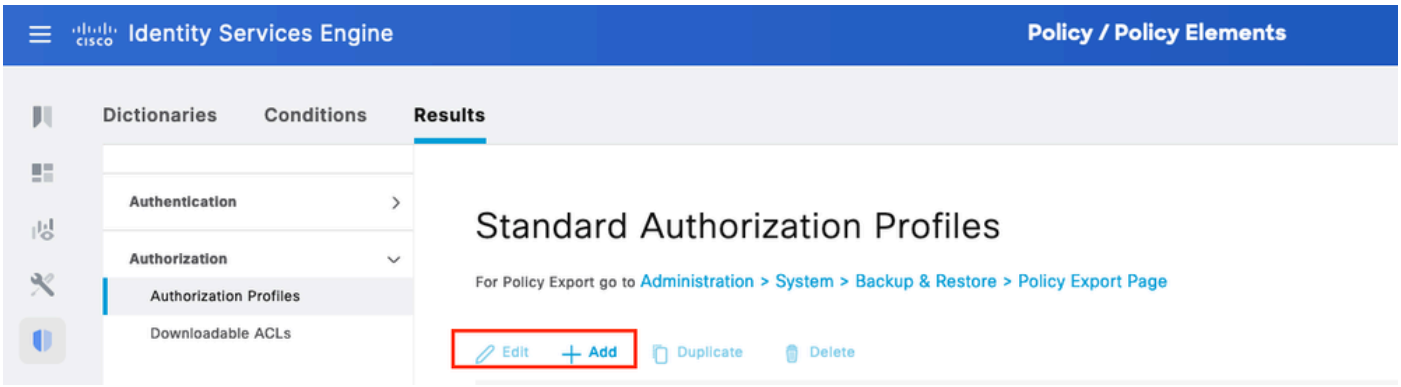
sessão Radius está ocorrendo.

- Deve-Proteger: O switch deve impor a criptografia no tráfego vinculado à sessão Radius. Se a sessão MKA falhar (ou tiver um tempo limite), a conexão será considerada uma falha de autorização e outra tentativa será feita para estabelecer a sessão MKA.
- Deve-Proteger: O switch tenta executar a criptografia MKA. Se a sessão MKA vinculada à sessão Radius for bem-sucedida, o tráfego será criptografado. Se o MKA falhar ou expirar, o switch permitirá o tráfego não criptografado vinculado a essa sessão Radius.

Etapa 1. Em ambos os PCs, aplique uma política de MKA deve-proteger para ter flexibilidade no caso de uma máquina sem recursos MKA se conectar à interface Ten 1/0/1.

Como opção, você pode configurar uma política para PC2 que aplique uma política de segurança obrigatória.

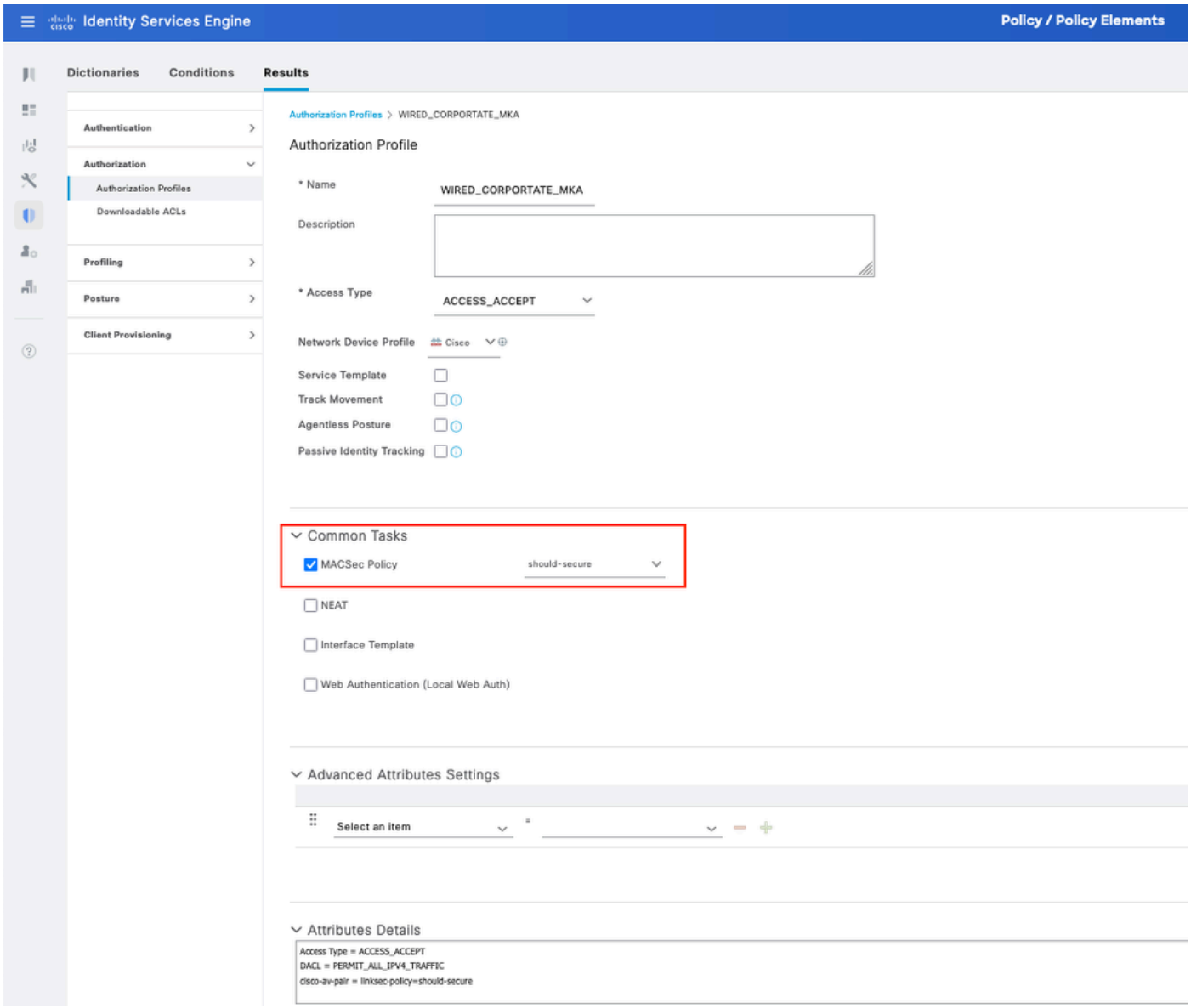
Neste exemplo, configure a política para os PCs como em Policy > Policy Elements > Results > Authorization Profiles e depois +Add ou Edit em um perfil existente




Etapa 2. Preencha ou personalize os campos necessários para o perfil.

Certifique-se de que em Common Tasks você tenha selecionado MACSec Policy e a política correspondente para aplicar.

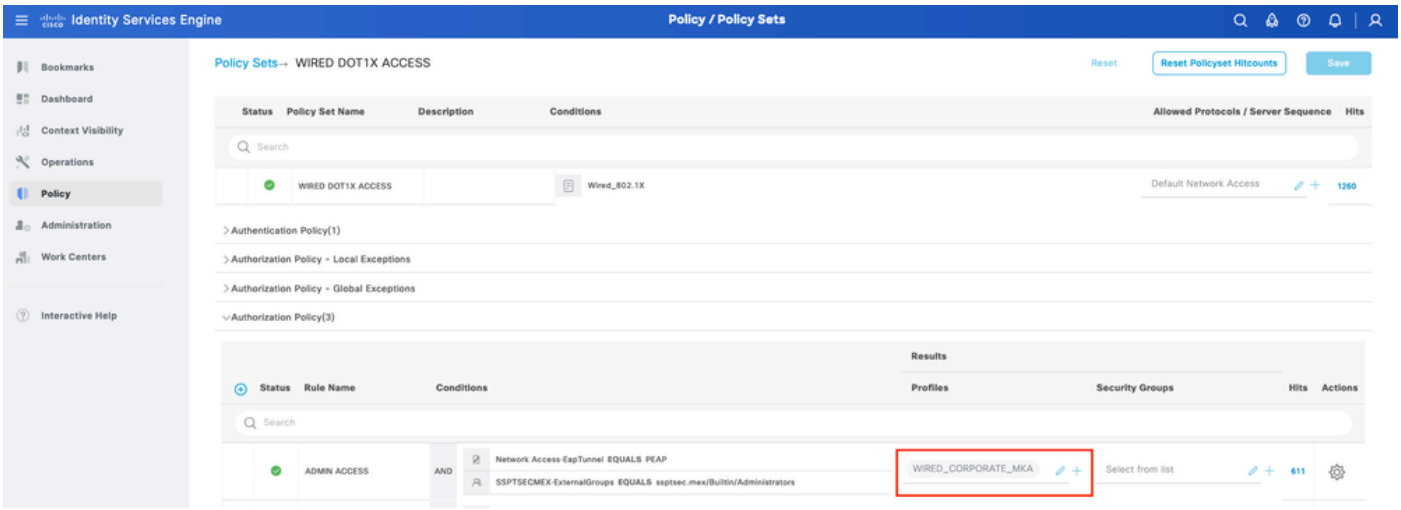
Role para baixo e Salve a configuração.




Etapa 3. Atribua o perfil de autorização correspondente às regras de autorização atingidas pelos dispositivos.

Navegue até Política > Conjuntos de políticas > (Selecione Conjunto de políticas atribuído) > Política de autorização.

Associe a regra de autorização ao perfil de autorização com as Configurações do MACsec. Role para baixo e Salve sua configuração.




## Configuração do MKA no Catalyst 9300

Etapa 1. Configure uma nova política MKA como este exemplo sugere:

```
!
  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
!
```

Etapa 2. Ative a criptografia MACsec na interface na qual os computadores estão conectados.

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

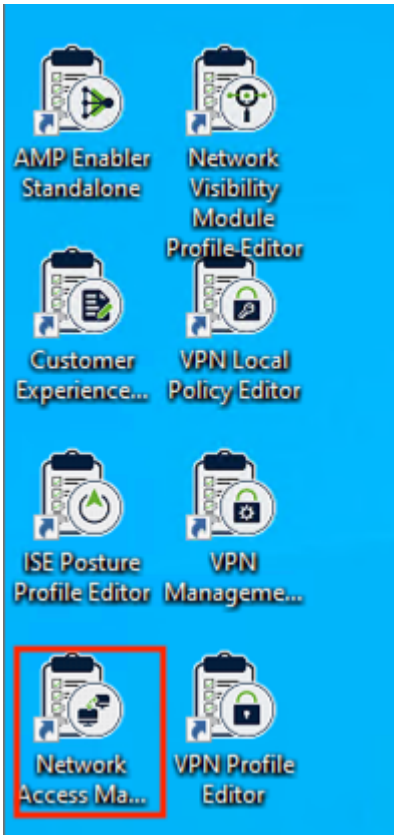
Note: Para obter mais informações relacionadas aos comandos e opções na configuração MKA, consulte o Guia de configuração de segurança correspondente à versão do switch que você usa. Neste cenário para este exemplo, o Guia de Configuração de Segurança,

Cisco IOS XE Bengaluru 17.6.x (Switches Catalyst 9300)

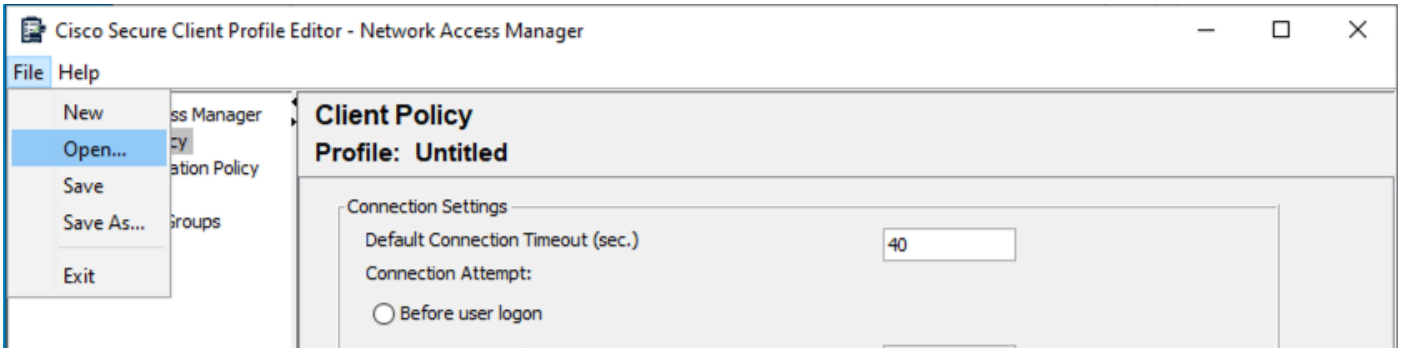
## Configuração do MKA usando o Editor de perfis do Network Access Manager.

Etapa 1. Faça o download (no site de download da Cisco) e abra o Editor de perfis que corresponde à versão do Secure Client que está sendo usada.

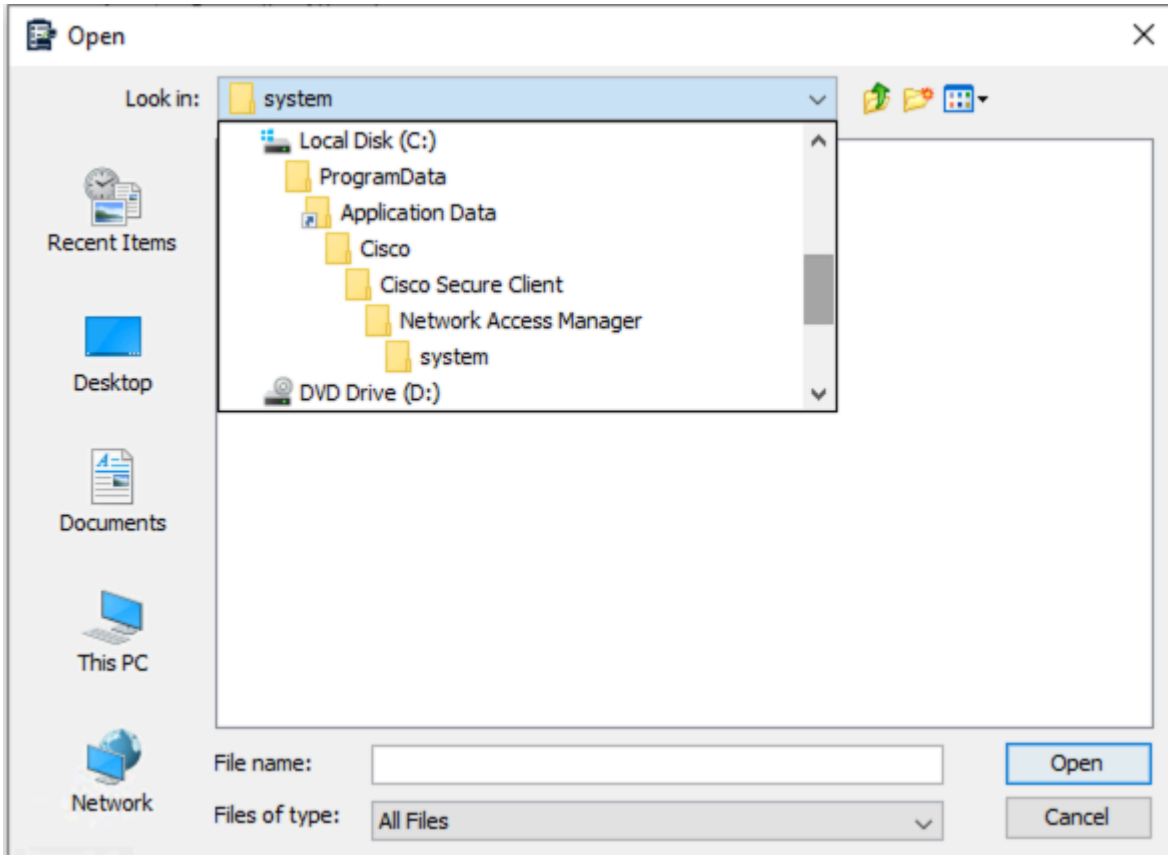
Depois de instalar este programa no computador, continue para abrir o Cisco Secure Client Profile Editor - Network Access Manager.



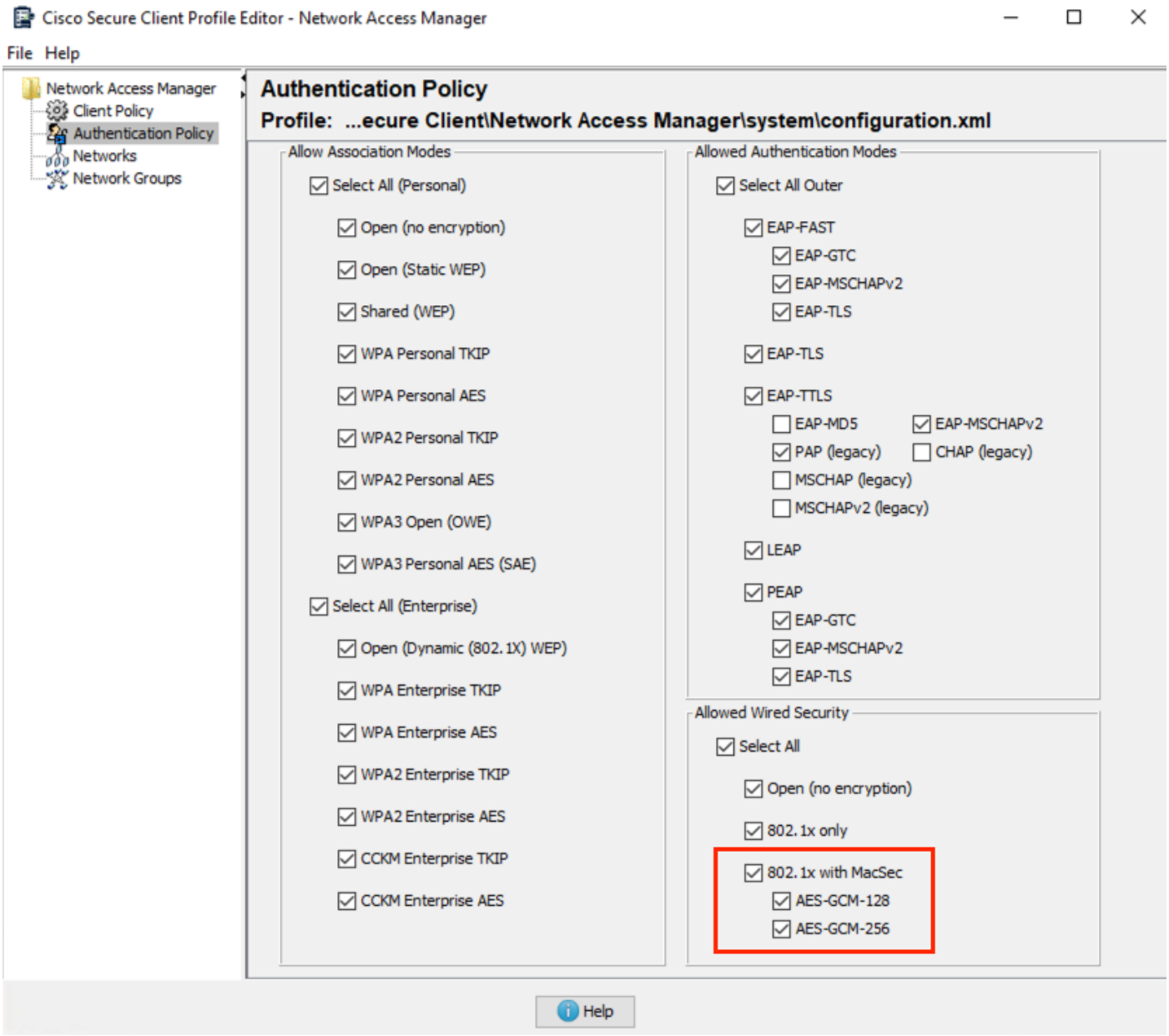

Etapa 2. Selecione Arquivo > Abrir.



Etapa 3. Selecione o sistema de pastas que está sendo exibido nesta imagem. Dentro dessa pasta, abra o arquivo chamado configuration.xml.

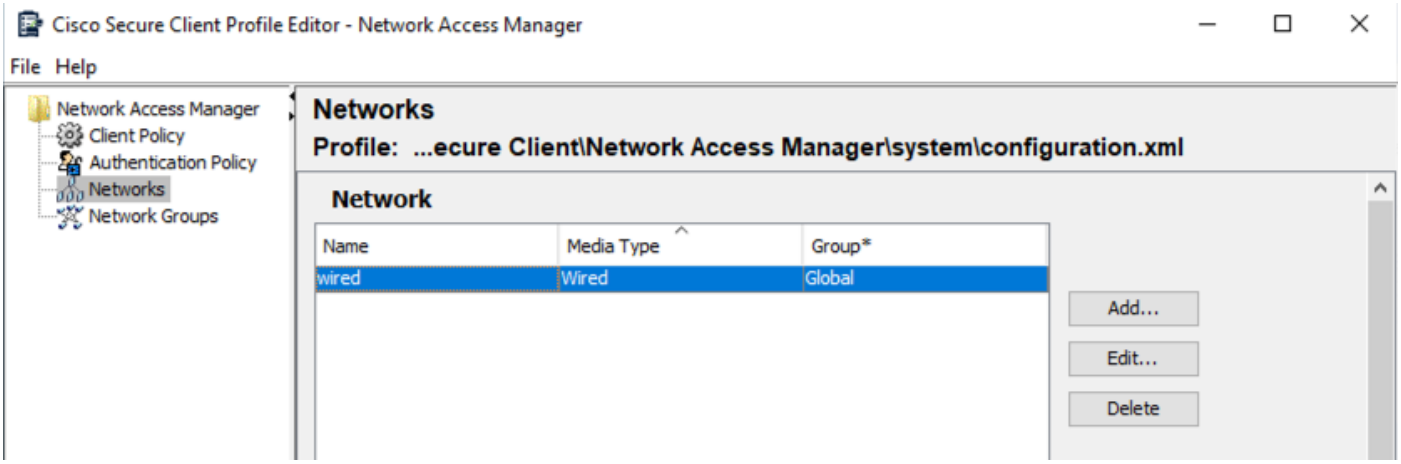



Etapa 4. Depois que o arquivo tiver sido carregado pelo Editor de perfis, selecione a opção Authentication Policy e verifique se a opção relacionada ao 802.1x com MACSec está habilitada.



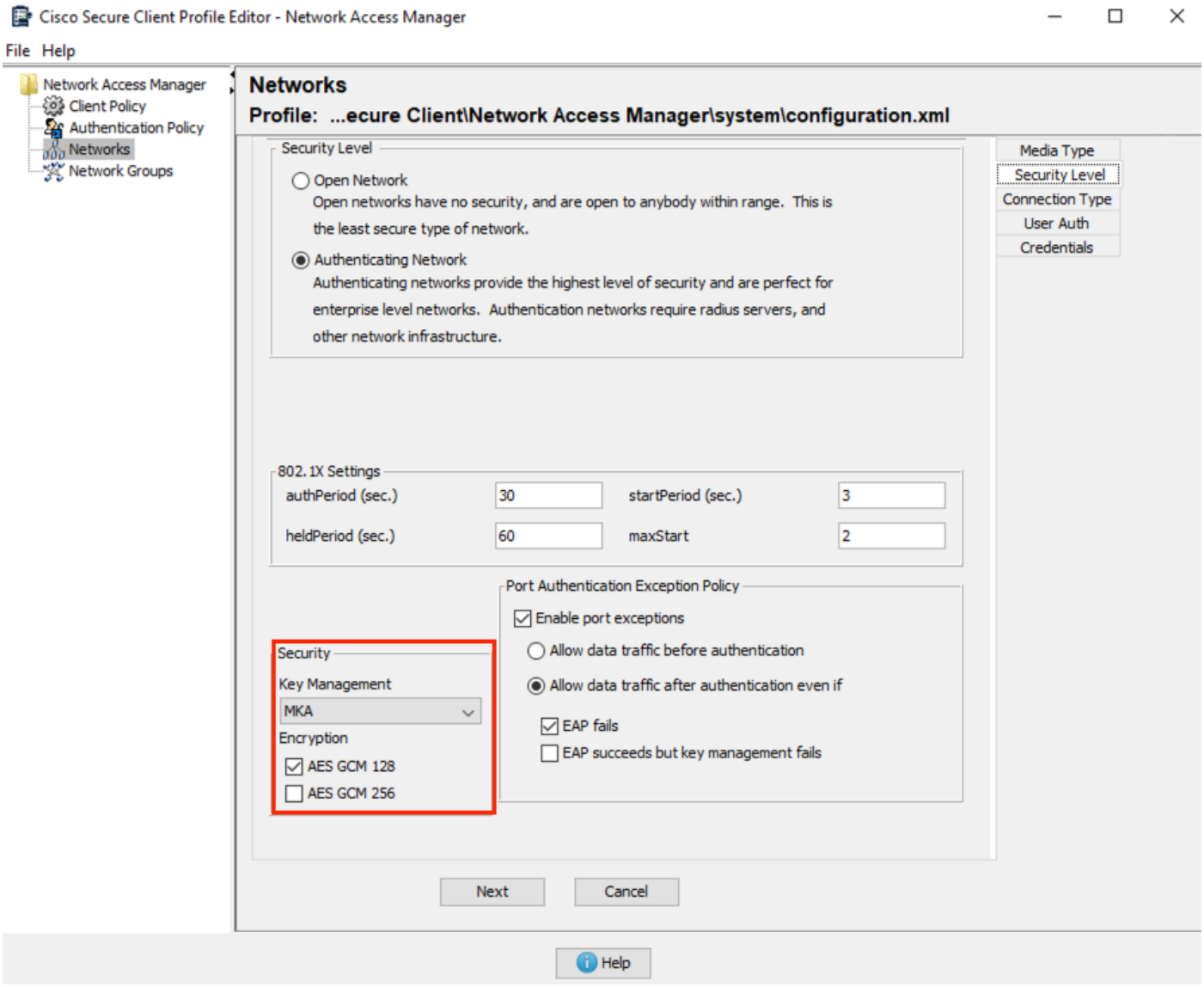

Etapa 5. Prossiga para a seção Redes. Nesta parte, você pode Adicionar um novo perfil para uma conexão com fio ou Editar o perfil com fio padrão instalado com o Secure Client 5.0 .

Neste cenário, vamos Editar o perfil com fio existente.




Etapa 6. Configure o perfil. Na seção Security Level, ajuste o Key Management para usar o MKA seguido por uma criptografia AES GCM 128.

Ajuste também os outros parâmetros para a autenticação dot1x e políticas.

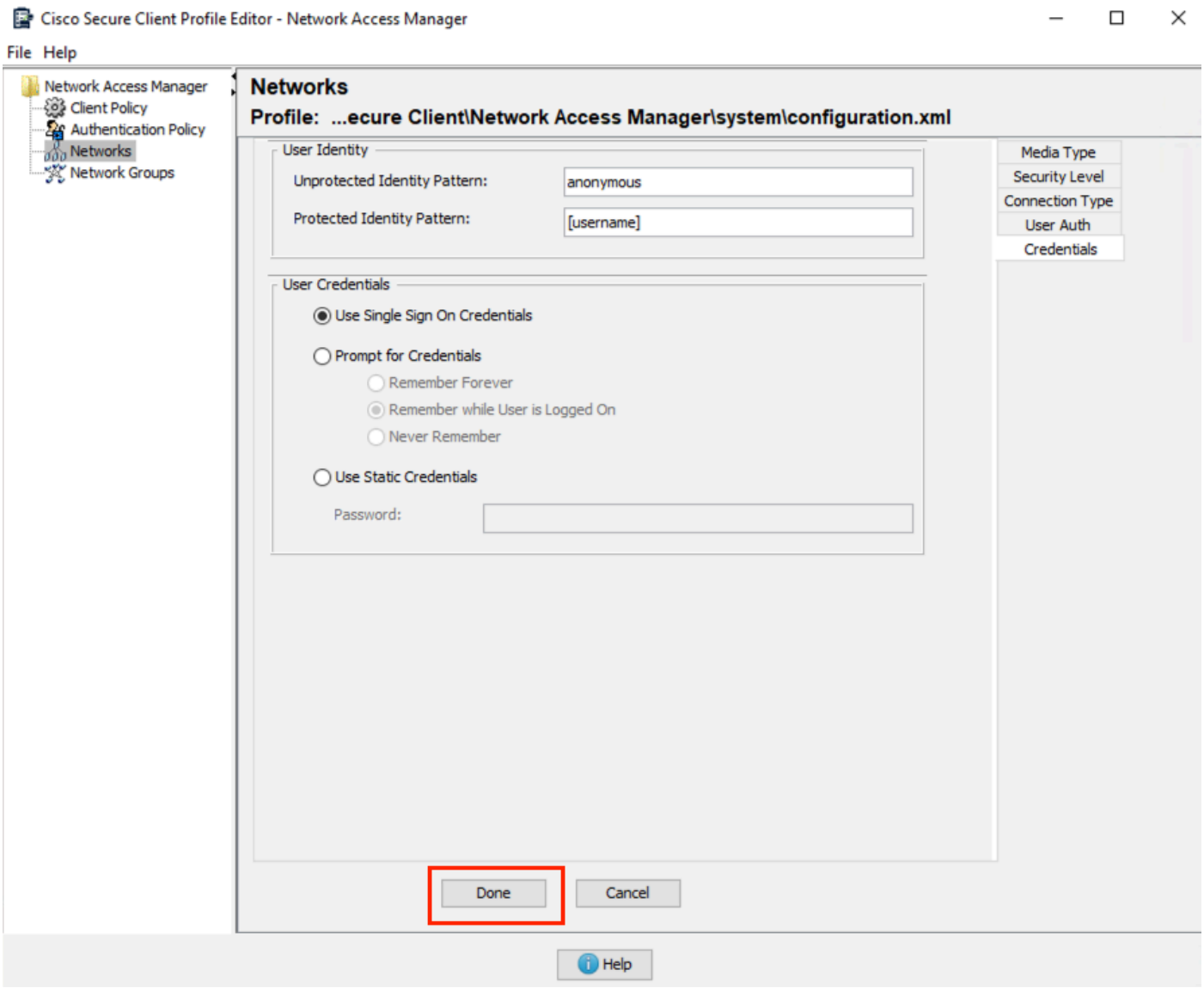


Etapa 7. Configure as seções restantes referentes a Tipo de conexão, Autenticação de usuário e Credenciais.

Essas seções variam de acordo com as configurações de autenticação selecionadas na seção Nível de segurança.

Quando concluir as configurações, clique em Done (Concluído).

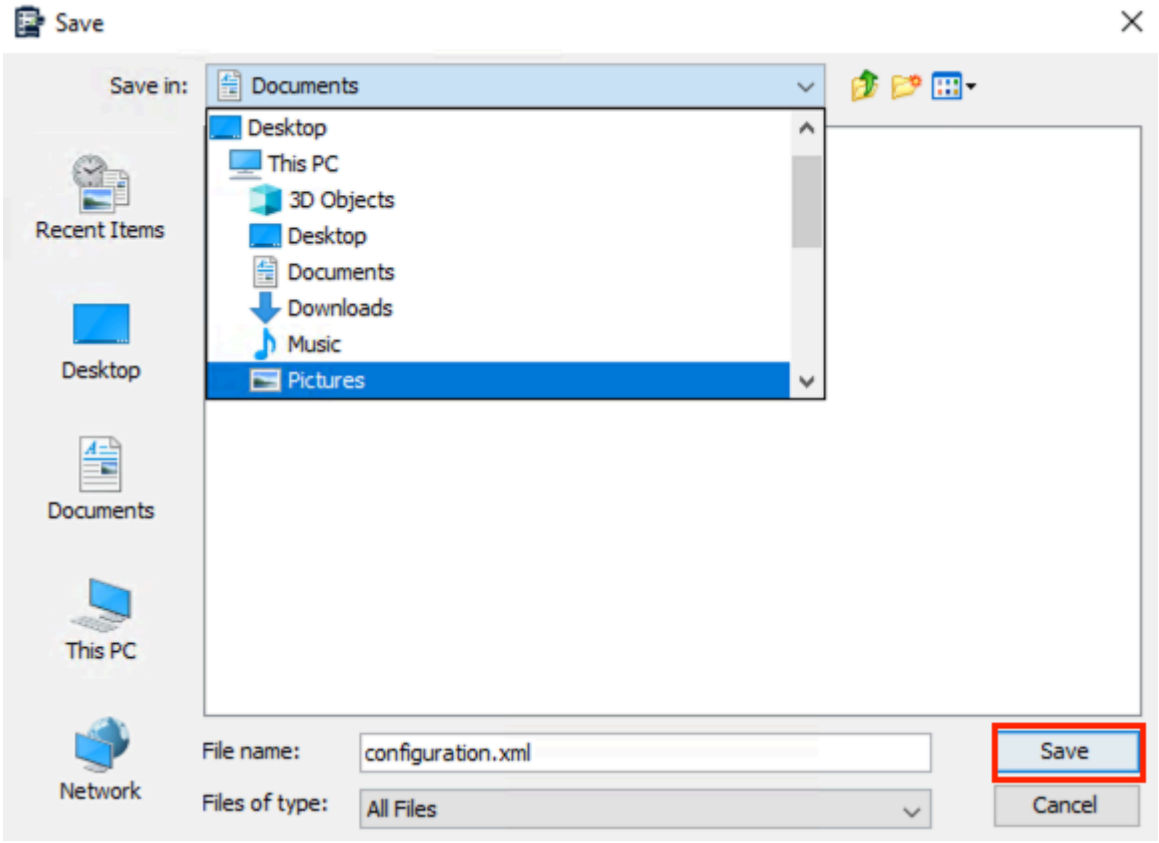
Para este cenário, estamos usando o Protected Extensible Authentication Protocol (PEAP) com credenciais de usuário.




Etapa 8. Navegue até o menu Arquivo. Continue com a opção Salvar como.

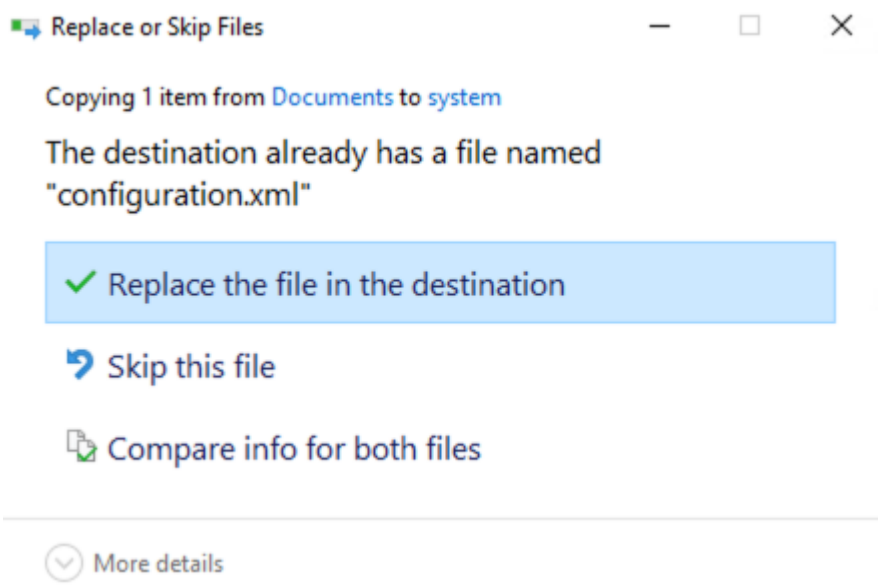
Nomeie o arquivo como configuration.xml e salve-o em uma pasta diferente de ProgramData\Cisco\Cisco Secure Client\Network Access Manager\system.

Neste exemplo, o arquivo foi salvo na pasta Documentos. Salve o perfil.




Etapa 8. Continue com o local do perfil, copie o arquivo e substitua o arquivo contido na pasta ProgramData\Cisco\Cisco Secure Client\Network Access Manager\system.

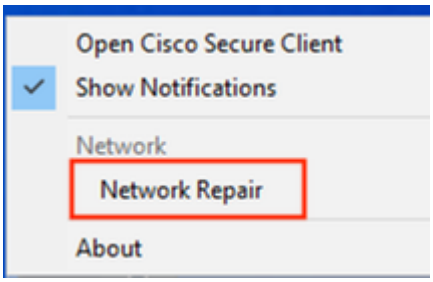
Selecione a opção Substituir o arquivo no destino.




Etapa 9. Para carregar o perfil modificado no Security Client 5.0, clique com o botão direito do

mouse no ícone do Secure Client localizado na barra de tarefas inferior direita da máquina Windows.

Execute um reparo de rede.

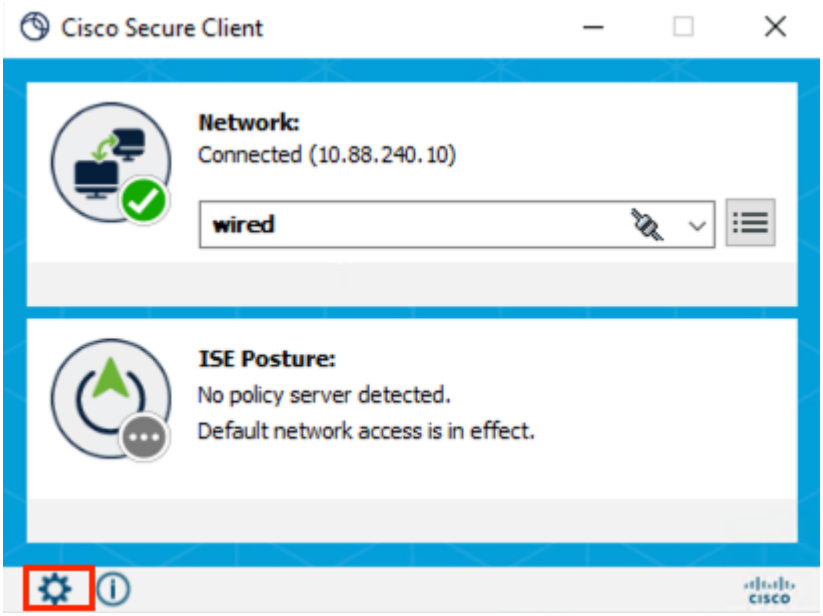



Note: Todas as redes configuradas por meio do editor de perfis têm privilégios de Administrador de rede, portanto, os usuários não podem personalizar/alterar o conteúdo que você configurou usando essa ferramenta.

## Configuração de redes MKA usando o Network Access Manager (opcional).

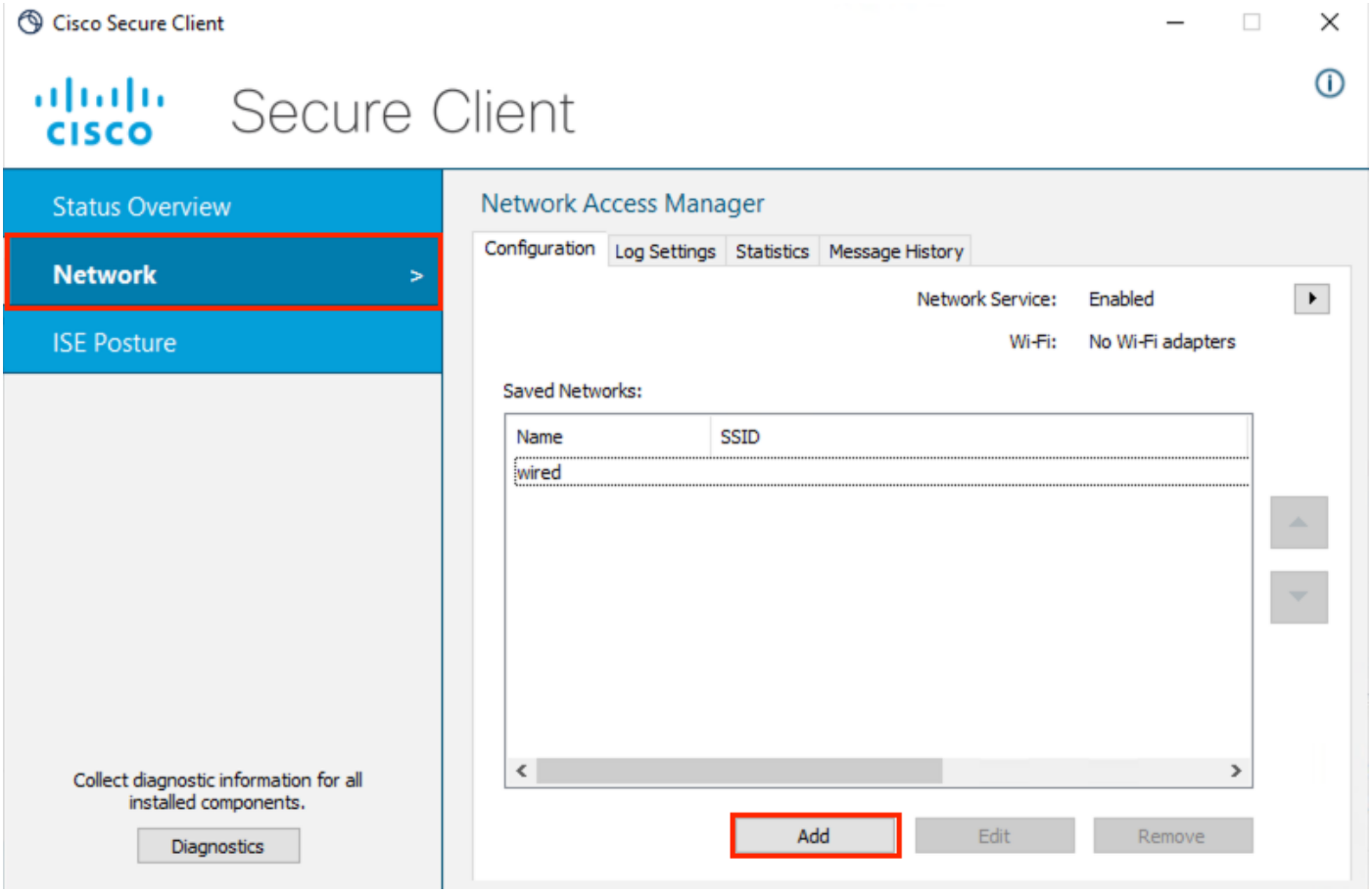
Etapa 1. Como uma alternativa para a configuração do MKA usando o Editor de perfis, você pode adicionar redes sem o uso desta ferramenta.

Na suíte Secure Client, selecione o ícone de engrenagem.



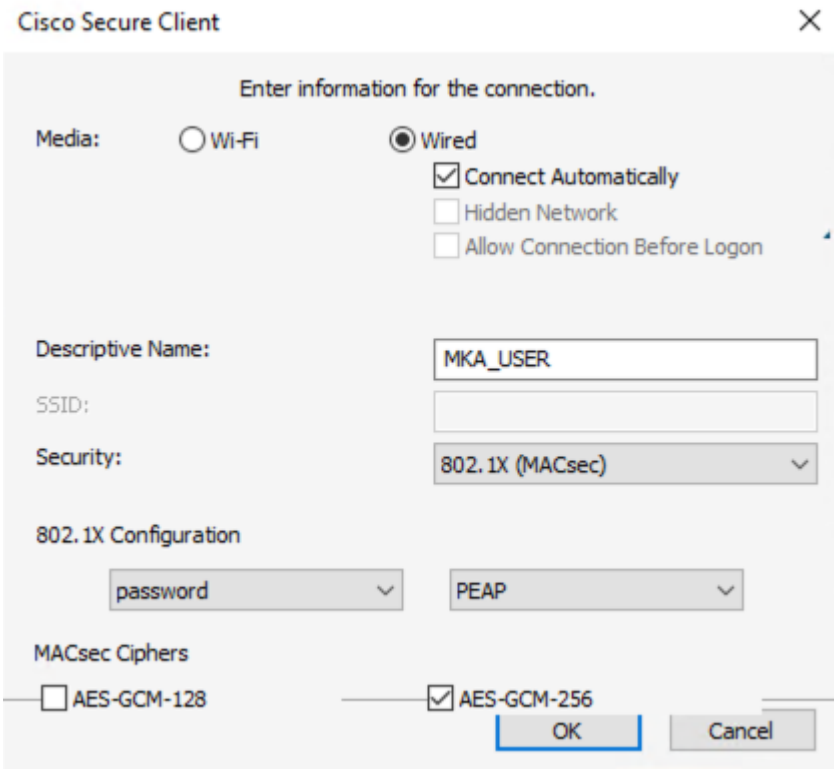

Etapa 2. Na nova janela exibida, selecione a opção Rede.

Na seção Configuração, selecione a opção Adicionar para entrar em uma rede MKA capaz com privilégios User Network.




Etapa 3. Na nova janela de configuração, defina as características da sua conexão e nomeie a rede.

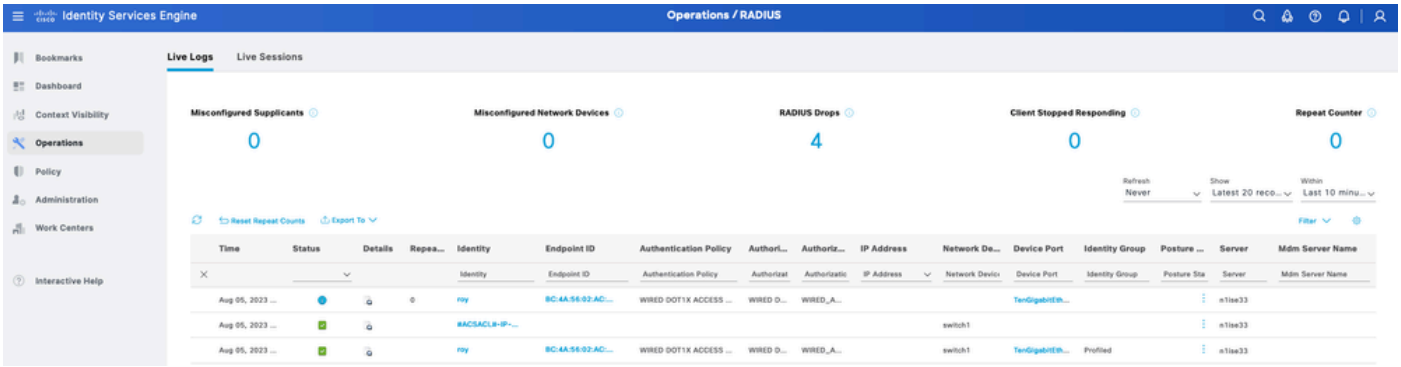
Quando terminar, selecione o botão OK.




# Verificar

## Validação no ISE

No ISE, após concluir a configuração desse fluxo, observe o dispositivo sendo autenticado e autorizado nos Livelogs.




Navegue na seção Detalhes da autenticação e na seção Resultado.

Os atributos definidos no perfil de autorização são enviados ao dispositivo de acesso à rede (NAD), bem como o consumo de uma licença 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. |

## Validação no switch Catalyst.

Esses comandos podem ser usados para validar a funcionalidade adequada dessa solução.

```
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
```

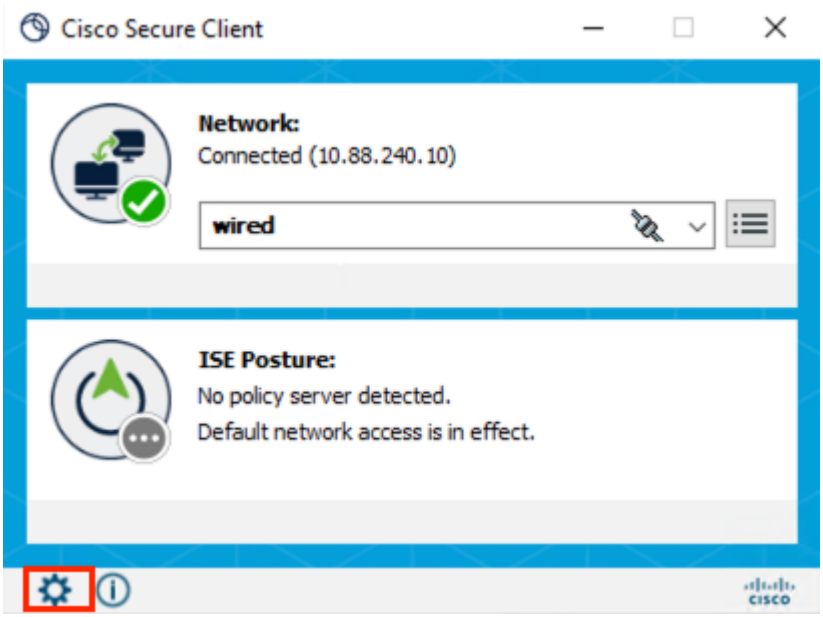
```
switch1#show mka session

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
```

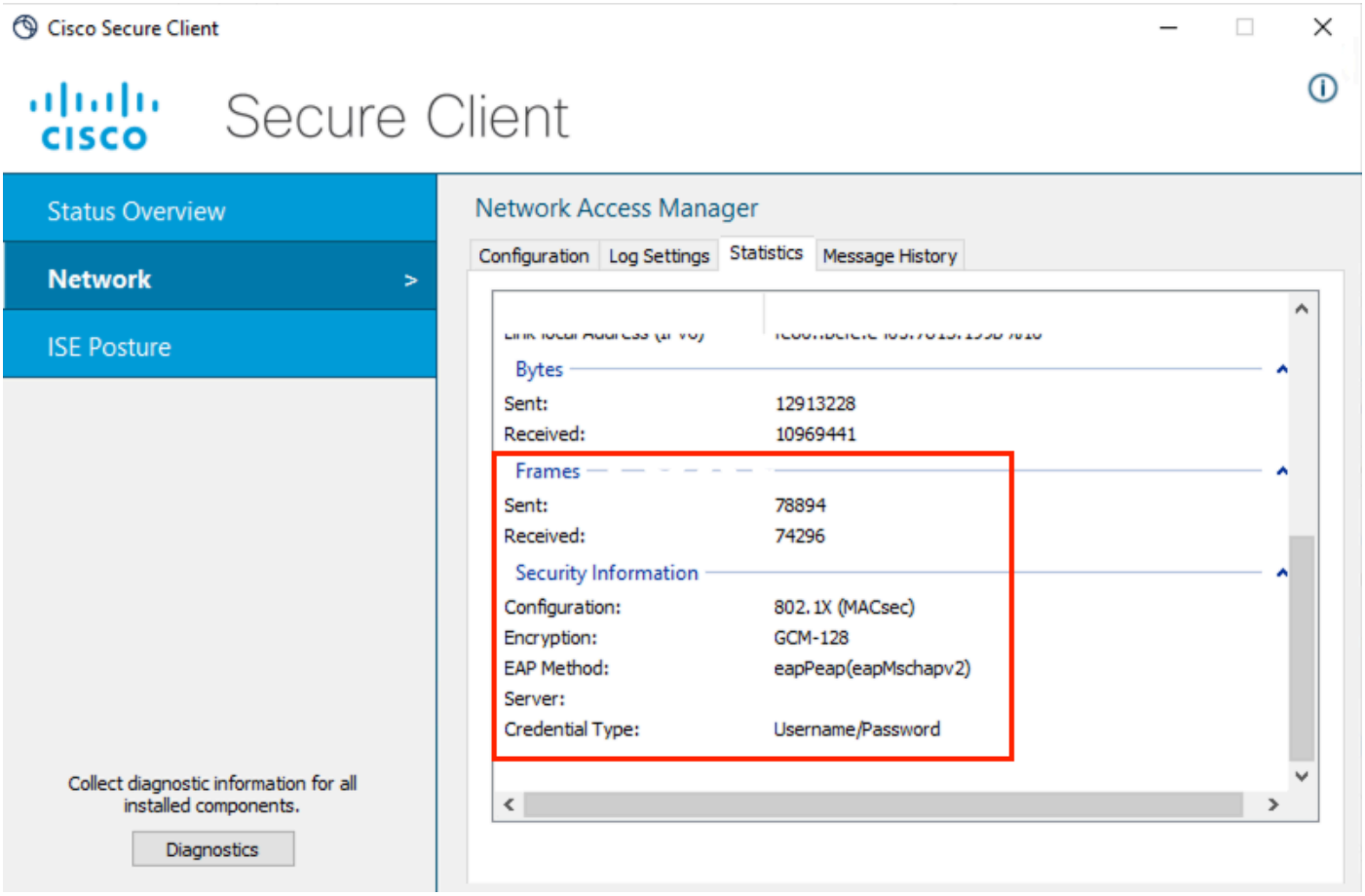
Validação no Cliente Seguro.

A autenticação é bem-sucedida com o perfil que foi criado com a criptografia MACsec. Se você clicar no ícone do mecanismo, mais informações poderão ser exibidas.




No menu exibido aqui para o Secure Client, na seção Network Access Manager > Statistics, observe a Criptografia e a configuração MACsec correspondente.

Os quadros recebidos e enviados aumentam conforme a criptografia é executada na camada 2.

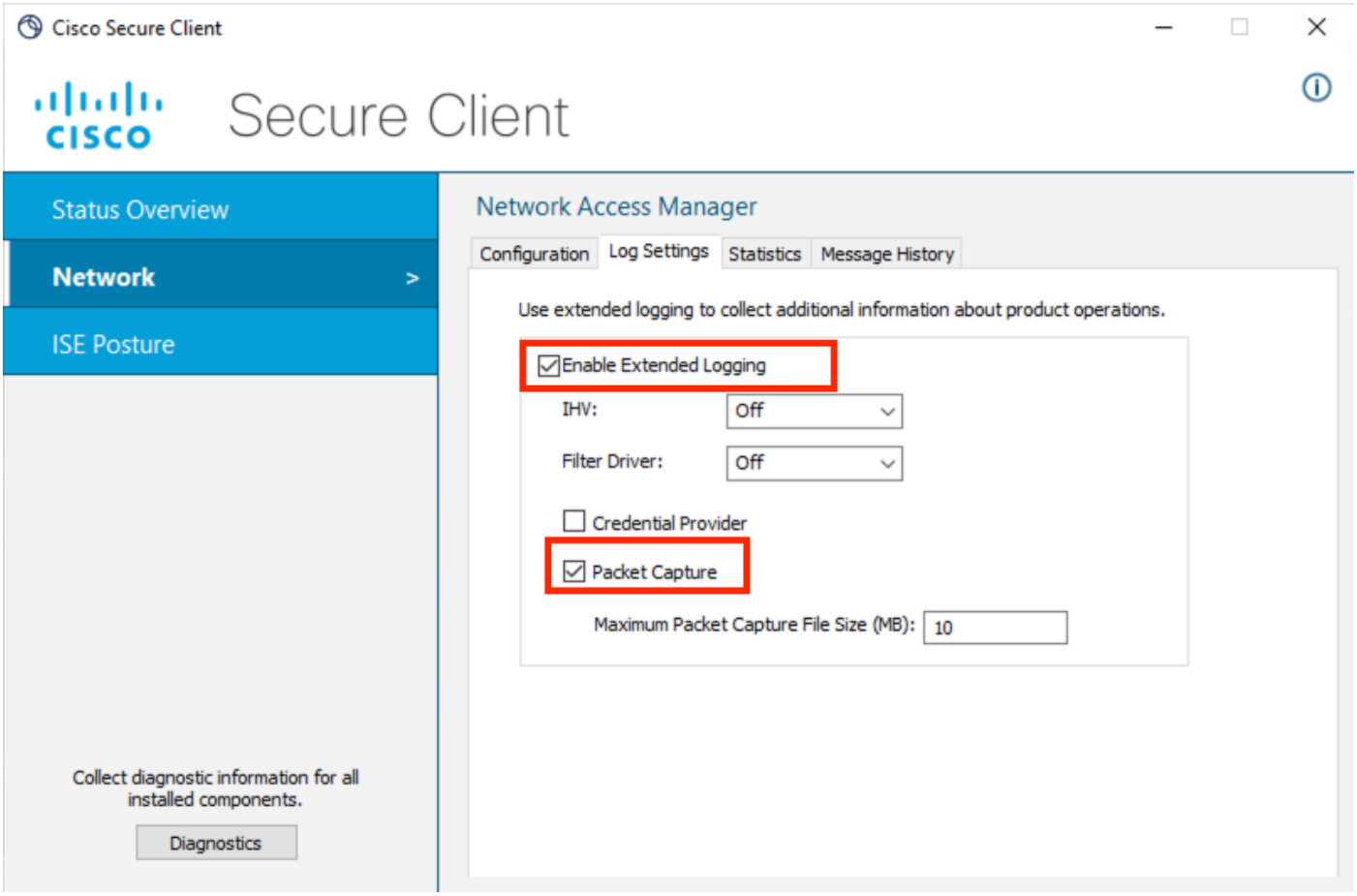



# Troubleshooting

Note: Esta seção aborda a parte de Troubleshooting relacionada aos problemas de MKA que podem surgir. Se você enfrentar uma falha de autenticação ou autorização, consulte o Guia de implantação prescritiva do acesso com fio seguro do ISE - Solução de problemas para investigar. Este guia supõe que as autenticações estão funcionando bem sem a criptografia MACsec.

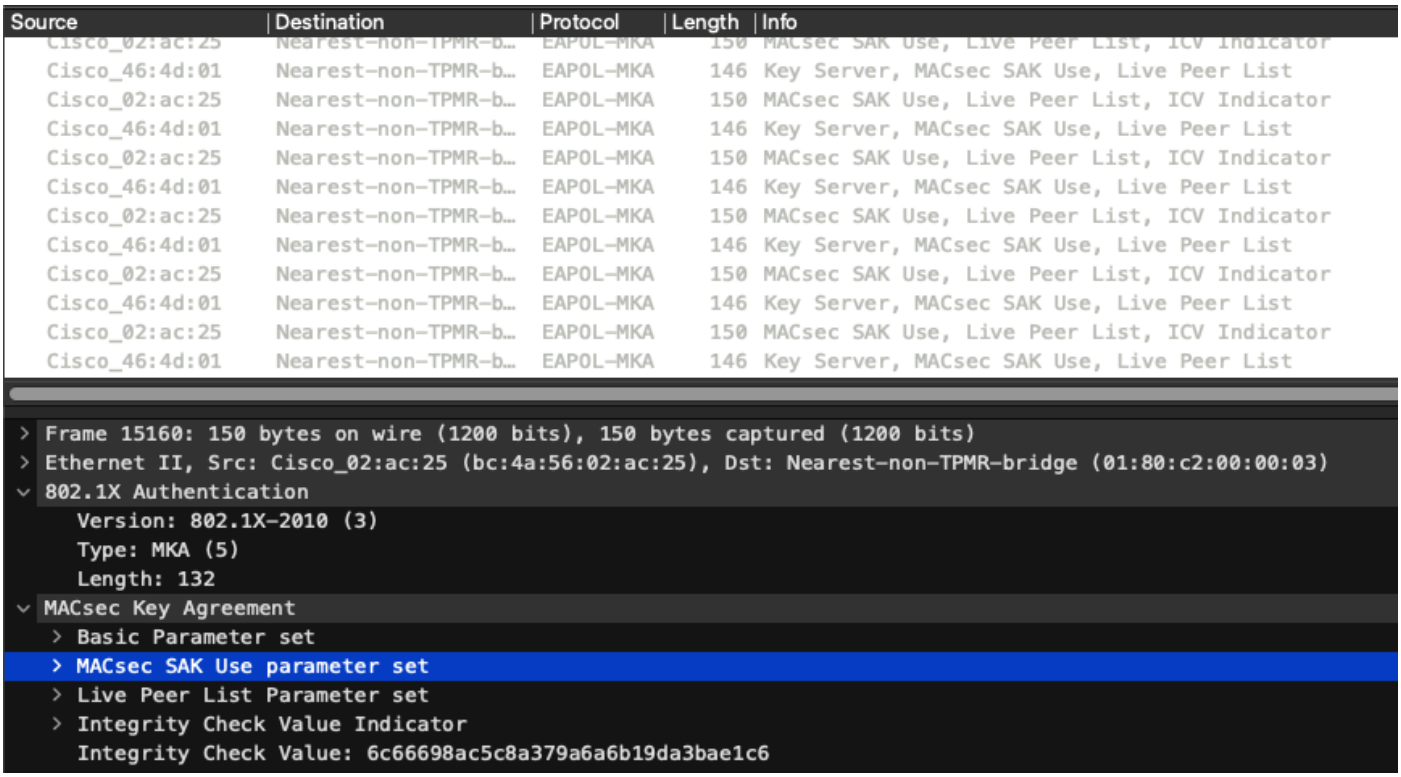
## Endpoint seguro da Cisco

- Ative o módulo DART no pacote Secure Endpoint.
- Neste menu, ative o Registro estendido para coletar mais dados sobre a conexão MKA do usuário. Você também pode ativar uma captura de pacote contida no pacote DART.




- Colete o pacote DART para prosseguir com a análise do configuration.xml e do gerenciador de acesso à rede. Consulte a documentação Run DART to Gather Data for Troubleshooting

Este exemplo mostra como os pacotes são vistos quando as informações entre o host e o switch são criptografadas :




No pacote DART, podemos encontrar informações úteis para a autenticação 802.1X e a sessão MKA no registro chamado NetworkAccessManager.txt.

Essas informações são exibidas em uma autenticação bem-sucedida com criptografia 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)
```

## Solução de problemas do Cisco IOS

Esses comandos podem ser implementados no Network Access Device (NAD) para revisar a criptografia MKA entre a plataforma e o solicitante.

Para obter mais informações sobre os comandos, consulte o guia de configuração correspondente da plataforma usada 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 são depurações de uma conexão MKA bem-sucedida com um host. Você pode usá-lo como uma referência que surge :

```
%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
```

### Solução de problemas do Identity Services Engine (ISE)

A solução de problemas relacionada a esse recurso é limitada à entrega do atributo cisco-av-pair linksec-policy=should-secure.

Certifique-se de que o resultado da autorização esteja enviando essas informações para a sessão Radius vinculada às portas do switch onde os dispositivos estão sendo conectados.

Para obter mais análise de autenticação no ISE, consulte Troubleshooting e Habilitação de Depurações no ISE

## Problemas comuns

### Incompatibilidade de codificação

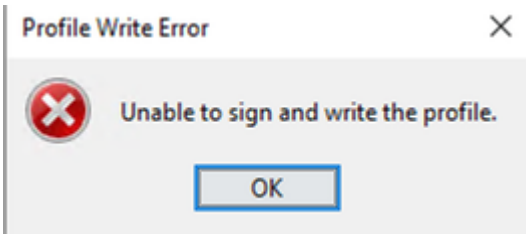
Esse log pode ser visto nas depurações MKA no NAD.

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

A primeira coisa a ser verificada neste cenário é que as cifras configuradas na política MKA no switch e no perfil do Secure Client são correspondentes.

No caso da criptografia AES-GCM-256, esses requisitos precisam ser atendidos (por documentação) Guia do Administrador do Cisco Secure Client (incluindo AnyConnect), Versão 5

### Impossibilidade de salvar o configuration.xml.



Este problema relacionado ao erro de gravação de perfil é resolvido salvando o configuration.xml (como descrito anteriormente) chamado Setup of MKA usando o Editor de perfis do Network Access Manager.

O erro está relacionado ao arquivo configuration.xml em uso não pode ser modificado. Portanto, salve o arquivo em outro local para continuar com a substituição dos perfis.

## Informações Relacionadas

- Configurando a criptografia MACsec
- Configurando o switch MACsec para host com Cat9k e ISE

Sobre esta tradução
A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês (link fornecido) seja sempre consultado.