使用安全客户端5配置MKA并对其进行故障排除

目录

简介

本文档介绍如何使用安全客户端5作为请求方在终端中配置MACsec加密。

先决条件

思科建议了解以下主题：

- 身份服务引擎
- 802.1x和Radius

- MACsec MKA加密
- 安全客户端版本5（以前称为Anyconnect）

使用的组件

本文档中的信息基于以下软件和硬件版本：

- 身份服务引擎(ISE)版本3.3
- Catalyst 9300版本17.06.05
- 思科安全客户端5.0.4032

背景信息

MACSec（介质访问控制安全）是网络安全标准，它为802.1AE IEEE标准定义的OSI模型（数据链路）第2层的以太网帧提供加密和保护。

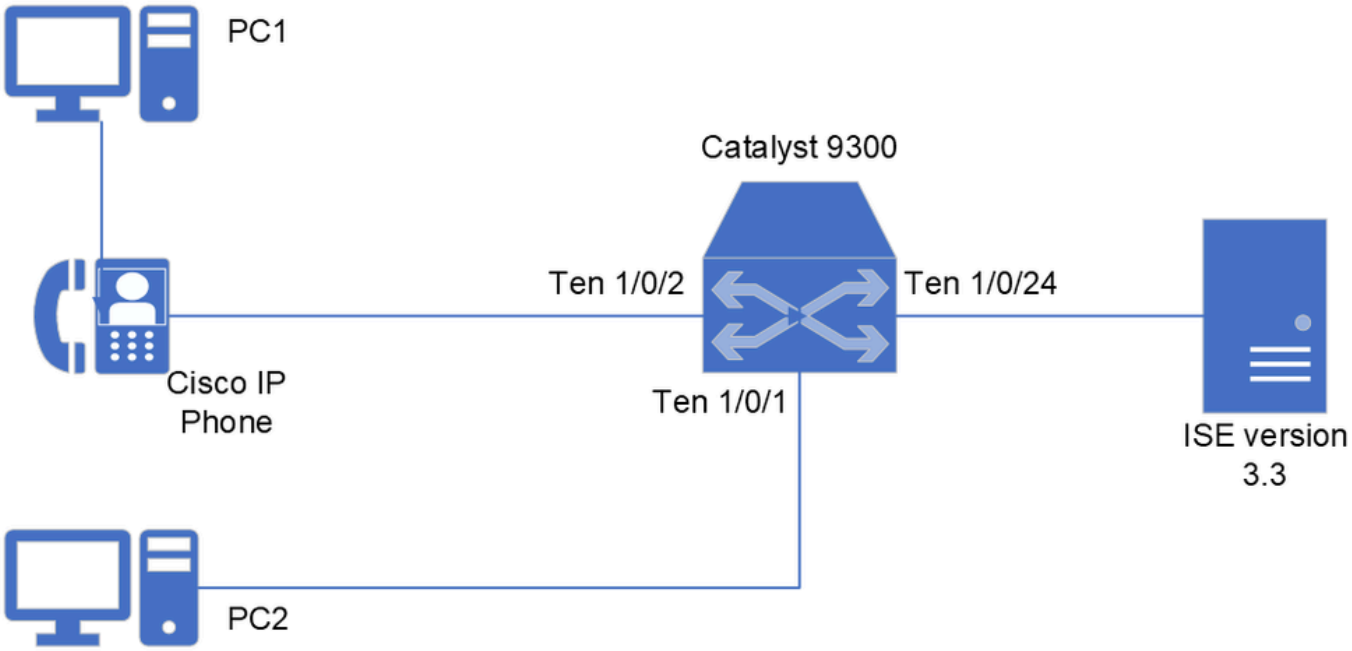
MACSec在点对点连接中提供这种加密，这种连接可以是交换机到交换机或交换机到主机的连接，因此此标准的覆盖范围仅限于有线连接。

此标准加密整个数据，第2层连接中传输的帧的源和目的MAC地址除外。

MACsec密钥协议(MKA)协议是MACsec对等体用来协商安全链路所需的安全密钥的机制。

配置

网络图



注意：本文档认为您已为PC设备和Cisco IP电话配置并使用Radius身份验证规则。要从头开始设置配置，请参阅ISE安全有线访问规范部署指南以检查身份服务引擎和交换机中的配置，以便进行基于身份的网络访问。

在ISE上设置策略

第一项任务是配置应用于上图中所示的两台PC（以及Cisco IP电话）的相应授权配置文件。

在此假设方案中，PC将使用802.1X协议作为身份验证方法，而Cisco IP电话使用Mac地址旁路(MAB)。

ISE通过RADIUS协议与交换机通信，了解交换机需要从通过RADIUS会话连接终端的接口实施的属性。

对于主机中的MACsec加密，所需的属性为cisco-av-pair = linksec-policy，它有以下3个可能的值：

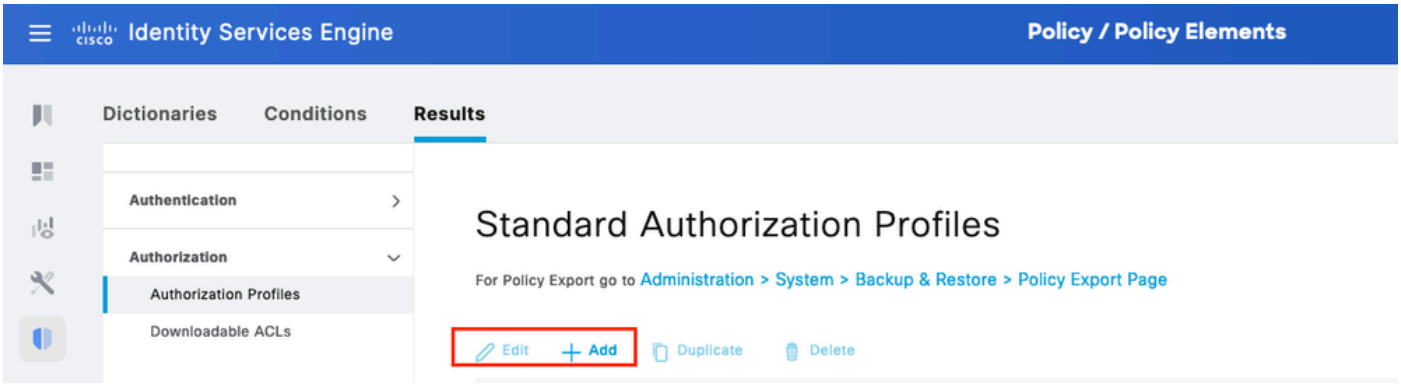
- Should-not-Secure:交换机不会在发生Radius会话的接口上执行MKA加密。
- Must-Secure:交换机必须在与Radius会话关联的流量中执行加密。如果MKA会话失败（或超时），连接将被视为授权失败，并会再次尝试建立MKA会话。
- 应该保护：交换机尝试执行MKA加密。如果链接到Radius会话的MKA会话成功，流量将被加

密。如果MKA发生故障或超时，交换机将允许该未加密流量链接到该Radius会话。

步骤1.在两台PC中，实施should-secure MKA策略，以便在没有MKA功能的计算机连接到接口Ten 1/0/1时具有灵活性。

作为一个选项，您可以为PC2配置实施必须安全的策略的策略。

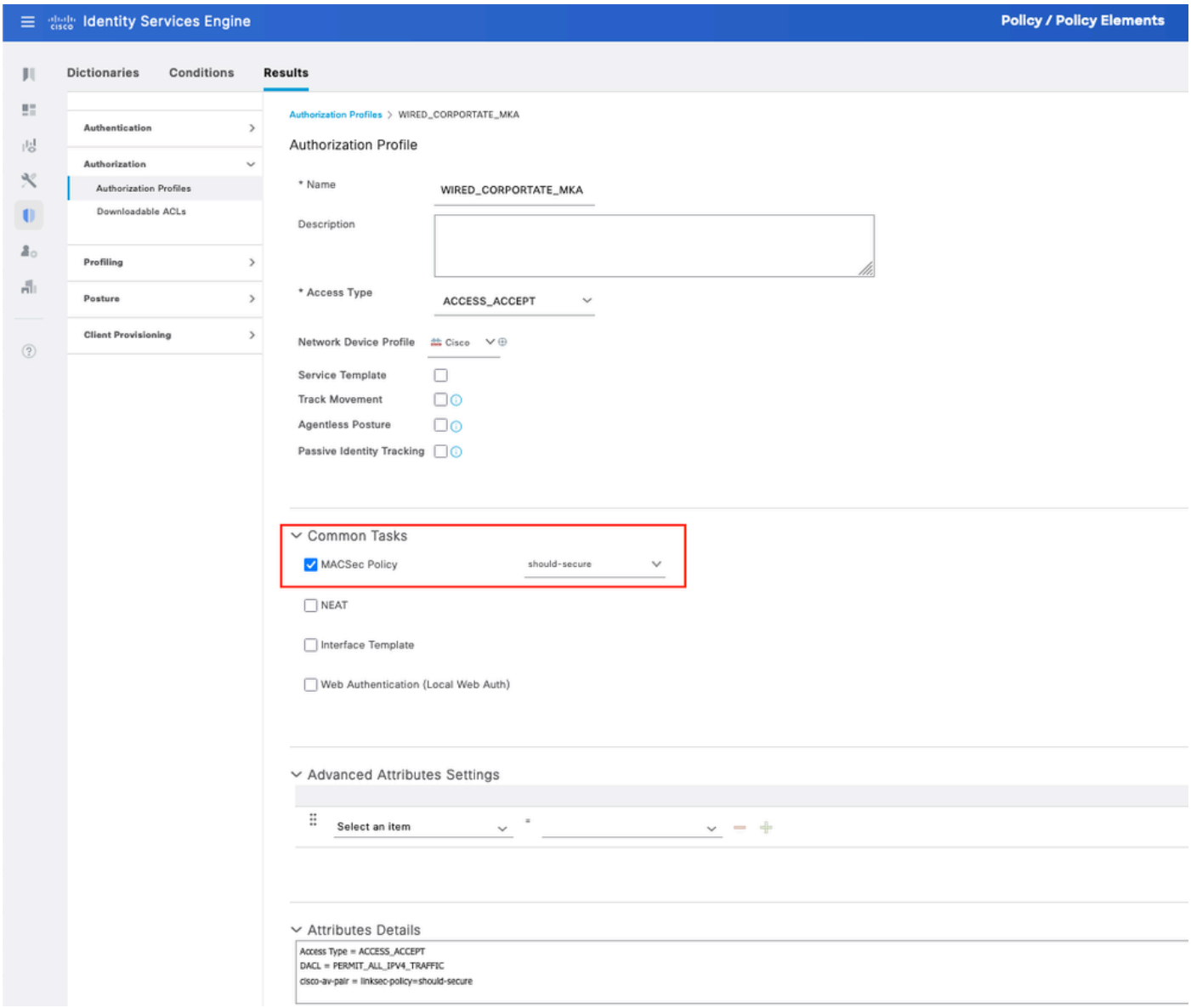
在本示例中，按照Policy > Policy Elements > Results > Authorization Profiles和+Add或Edit中的说明配置PC的策略



第2步。填写或自定义配置文件所需的字段。

确保在Common Tasks中选择了MACSec Policy和要应用的相应策略。

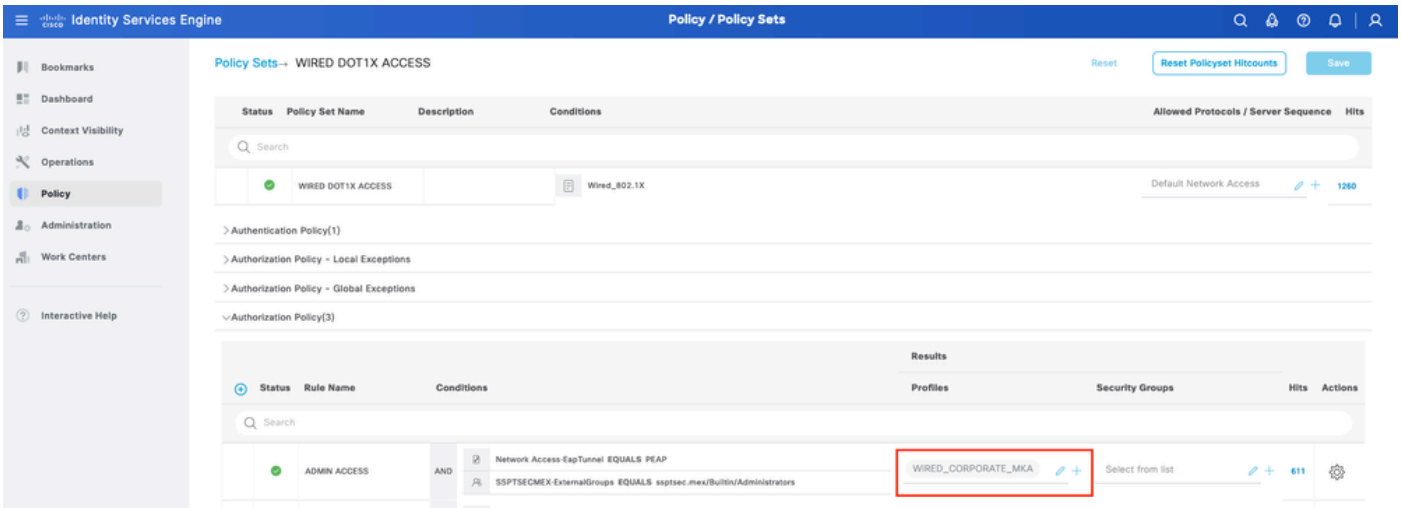
向下滚动并保存配置。



第3步。将相应的授权配置文件分配到设备所命中的授权规则。

导航到Policy > Policy Sets >(Select Policy Set assigned)> Authorization Policy。

将授权规则与具有MACsec设置的授权配置文件相关联。向下滚动并保存您的配置。



Catalyst 9300中MKA的设置

第1步。如以下示例所示，配置新的MKA策略：

```
!
  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
!
```

步骤2.在连接PC的接口中启用MACsec加密。

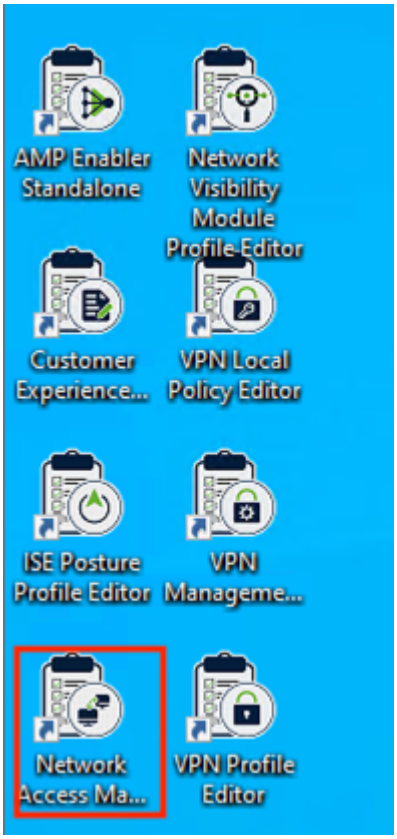
```
!
interface TenGigabitEthernet1/0/1
 macsec
 mka policy MKA_PC
!
```

注意：有关MKA配置中的命令和选项的更多信息，请查看与您使用的交换机版本对应的安全配置指南。在本示例中，安全配置指南Cisco IOS XE Bengaluru 17.6.x（Catalyst 9300交换机）

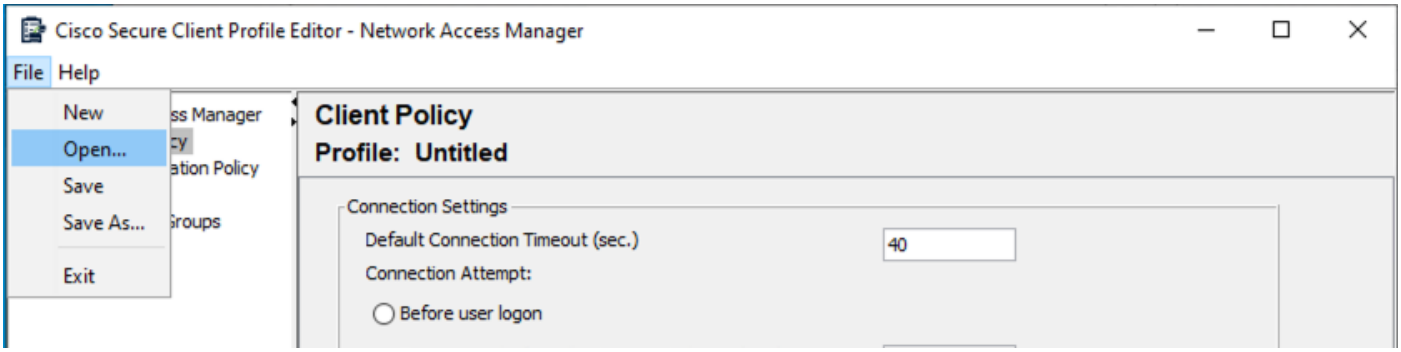
使用网络访问管理器配置文件编辑器设置MKA。

第1步：下载（从思科的下载网站）并打开与正在使用的安全客户端版本相匹配的配置文件编辑器。

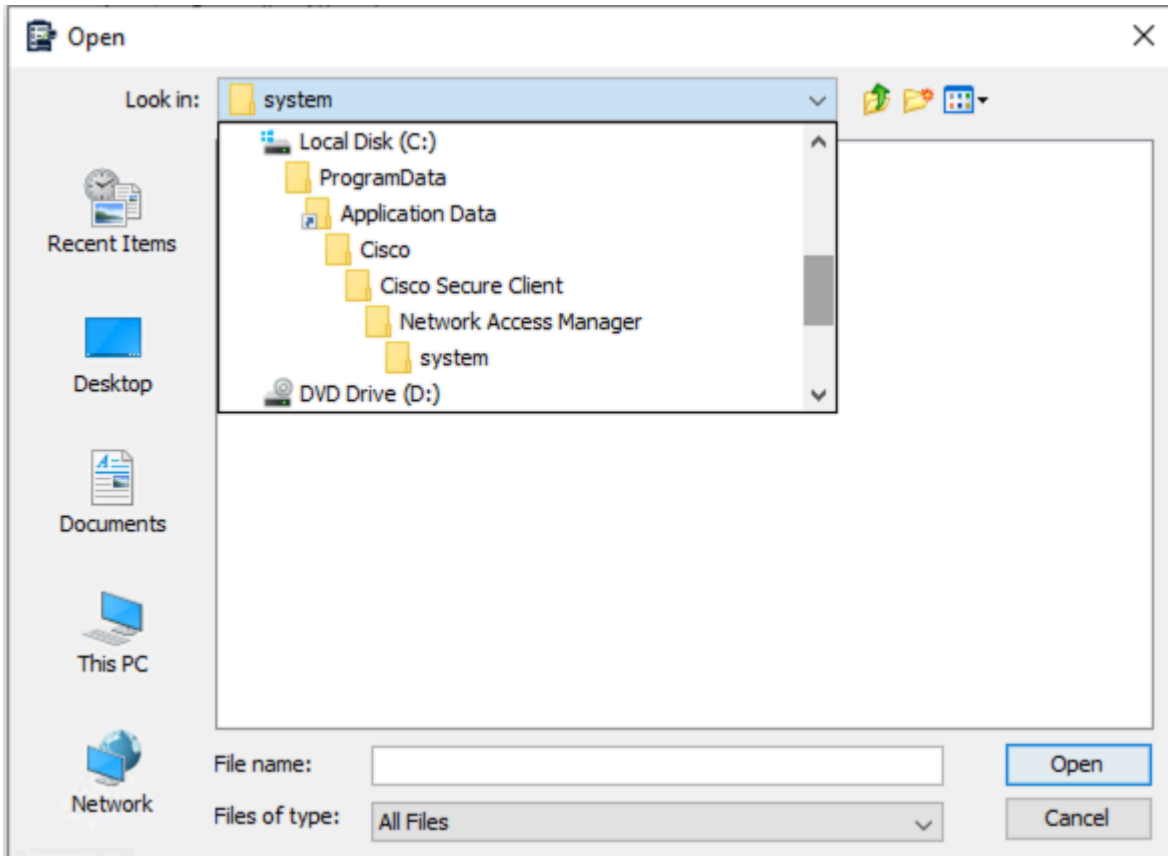
在计算机上安装此程序后，请继续打开Cisco Secure Client Profile Editor - Network Access Manager。



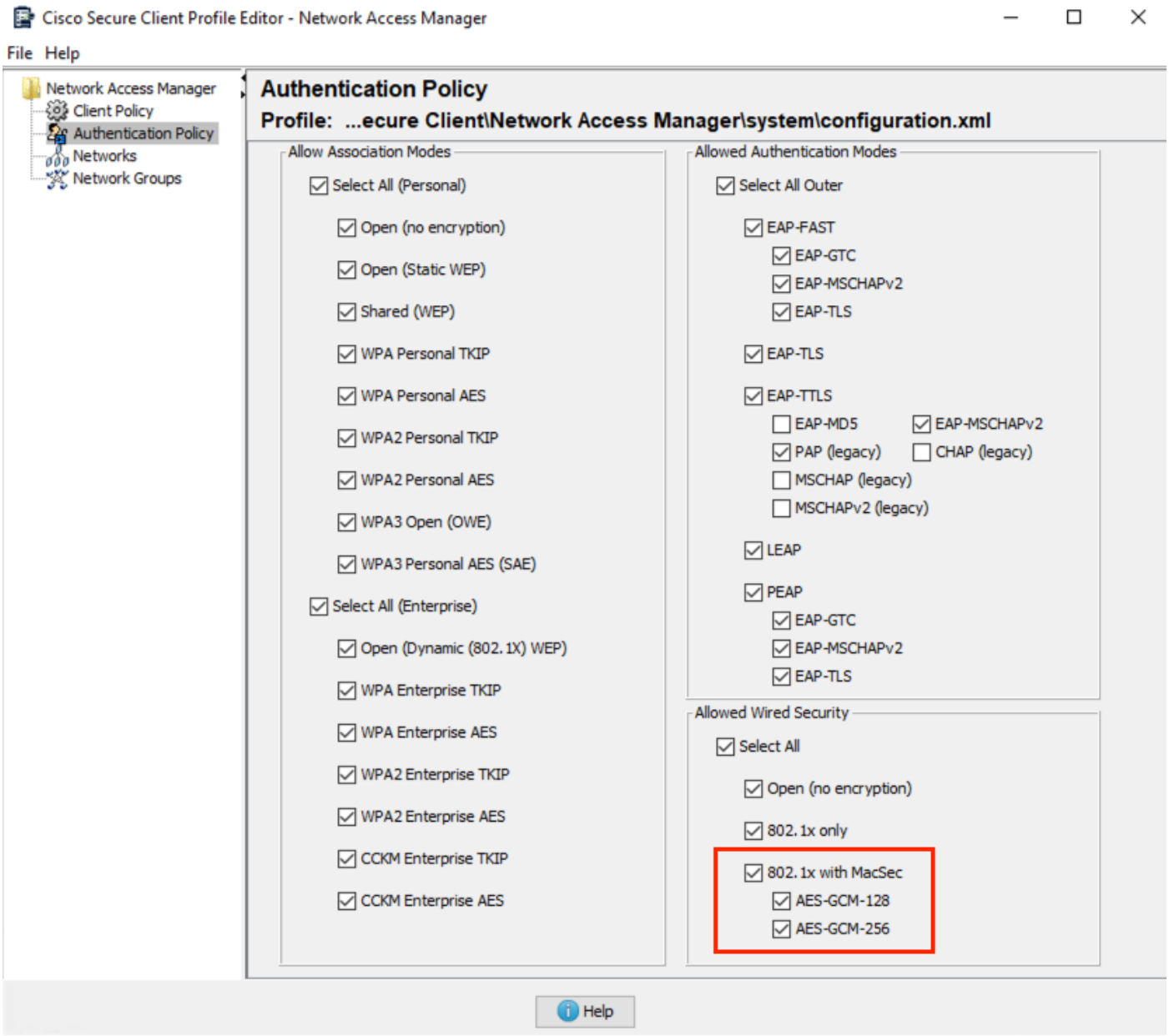
步骤2.选择File > Open。



第3步。选择在此映像中显示的文件夹系统。在此文件夹中打开名为configuration.xml的文件。

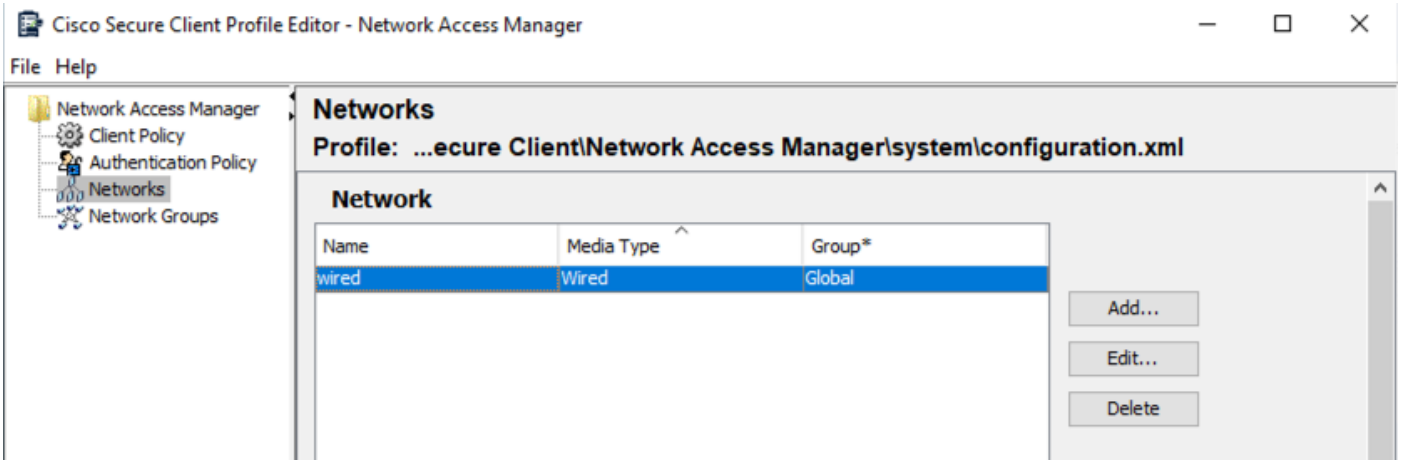


第4步。文件编辑器加载文件后，选择Authentication Policy选项，并确保启用与带MACSec的802.1x相关的选项。



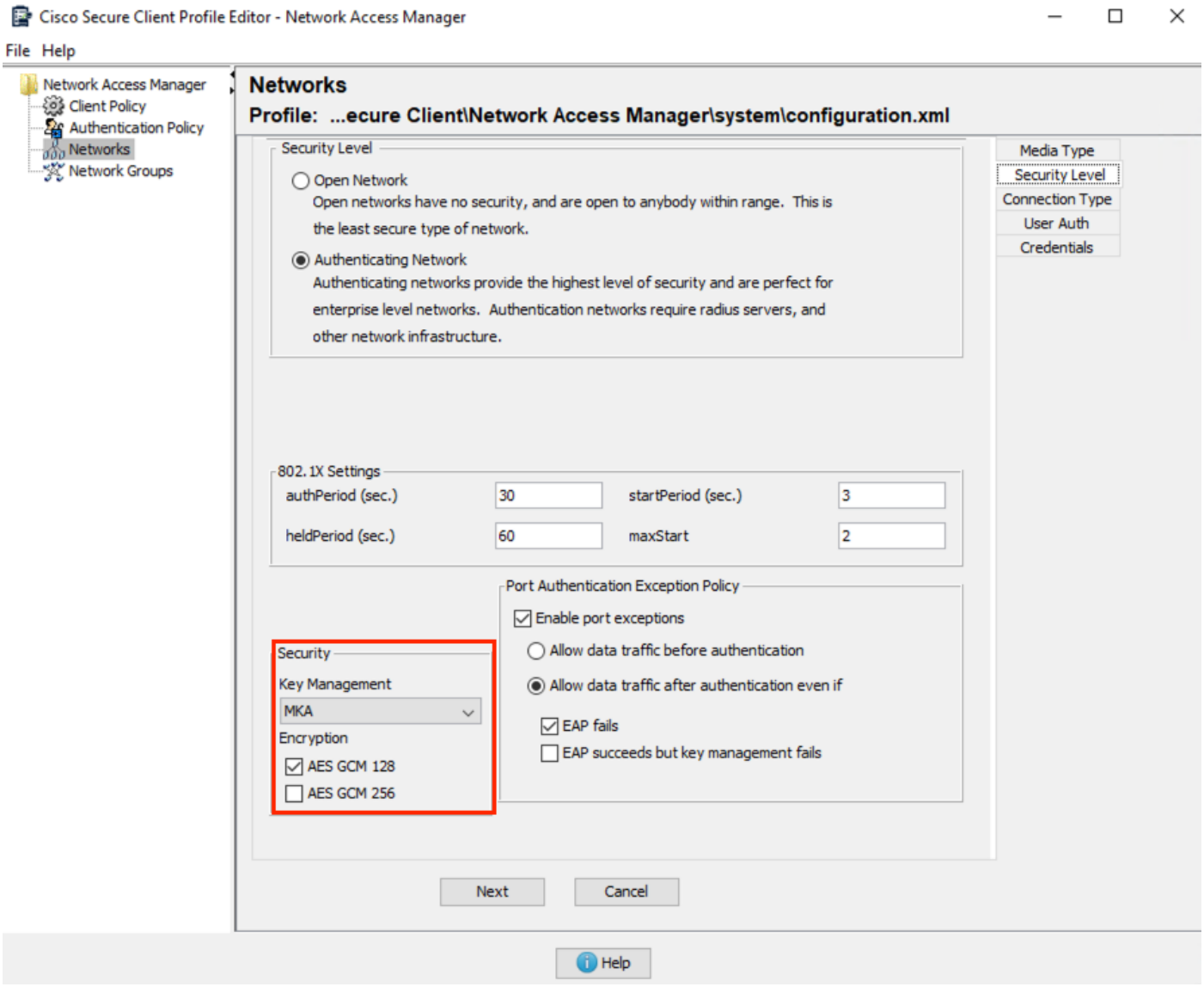
步骤5.继续进入Networks部分。在此部分中，您可以为有线连接添加新配置文件或编辑随Secure Client 5.0一起安装的默认有线配置文件。

在此场景中，我们将编辑现有的有线配置文件。



步骤6。配置配置文件。在Security Level部分，调整Key Management以使用MKA，然后是加密AES GCM 128。

调整身份验证dot1x和策略的其他参数。

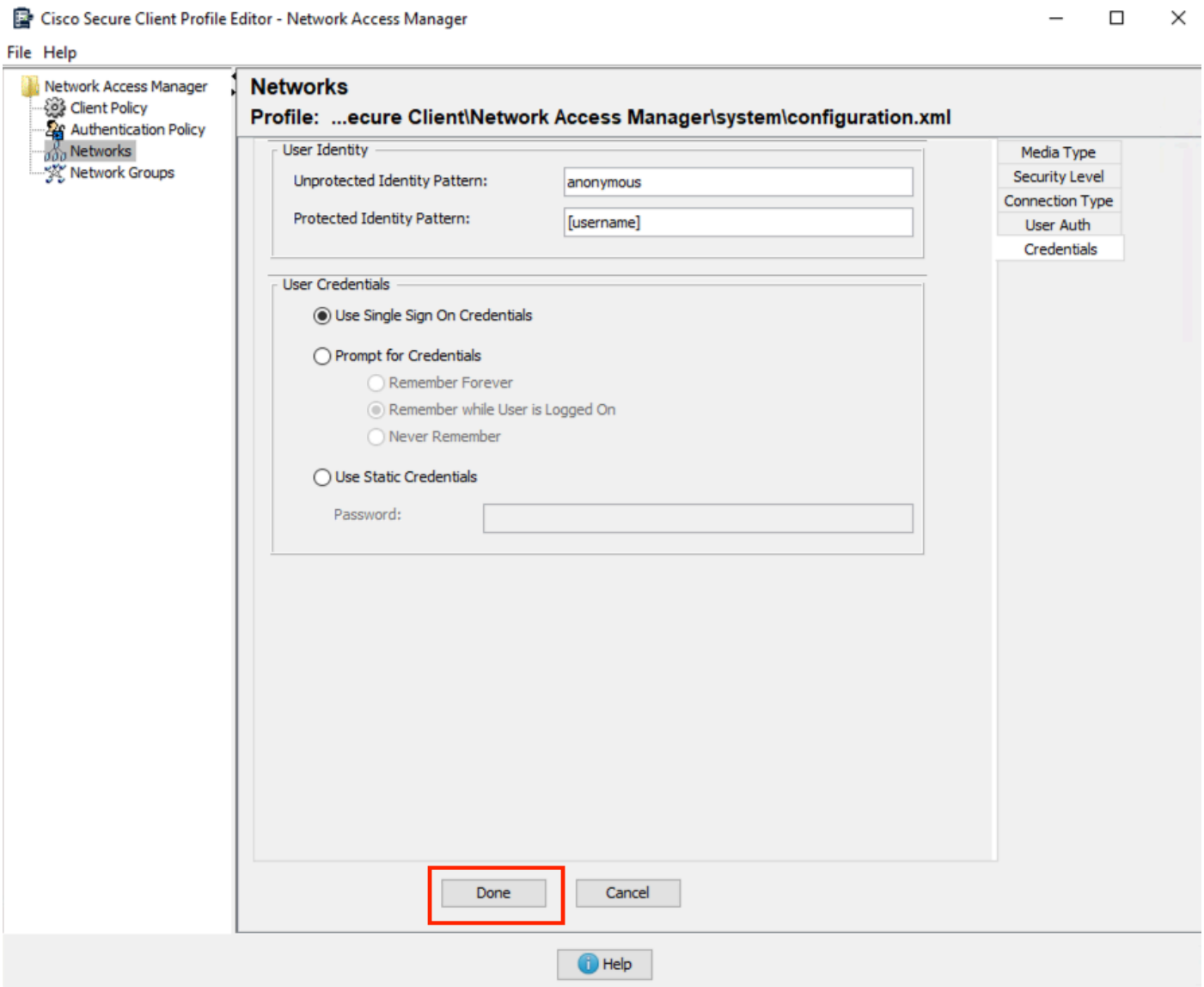


第7步配置其余有关Connection Type、User Auth和Credentials的部分。

这些部分因Security Level部分中所选的身份验证设置而异。

完成配置后，点击完成。

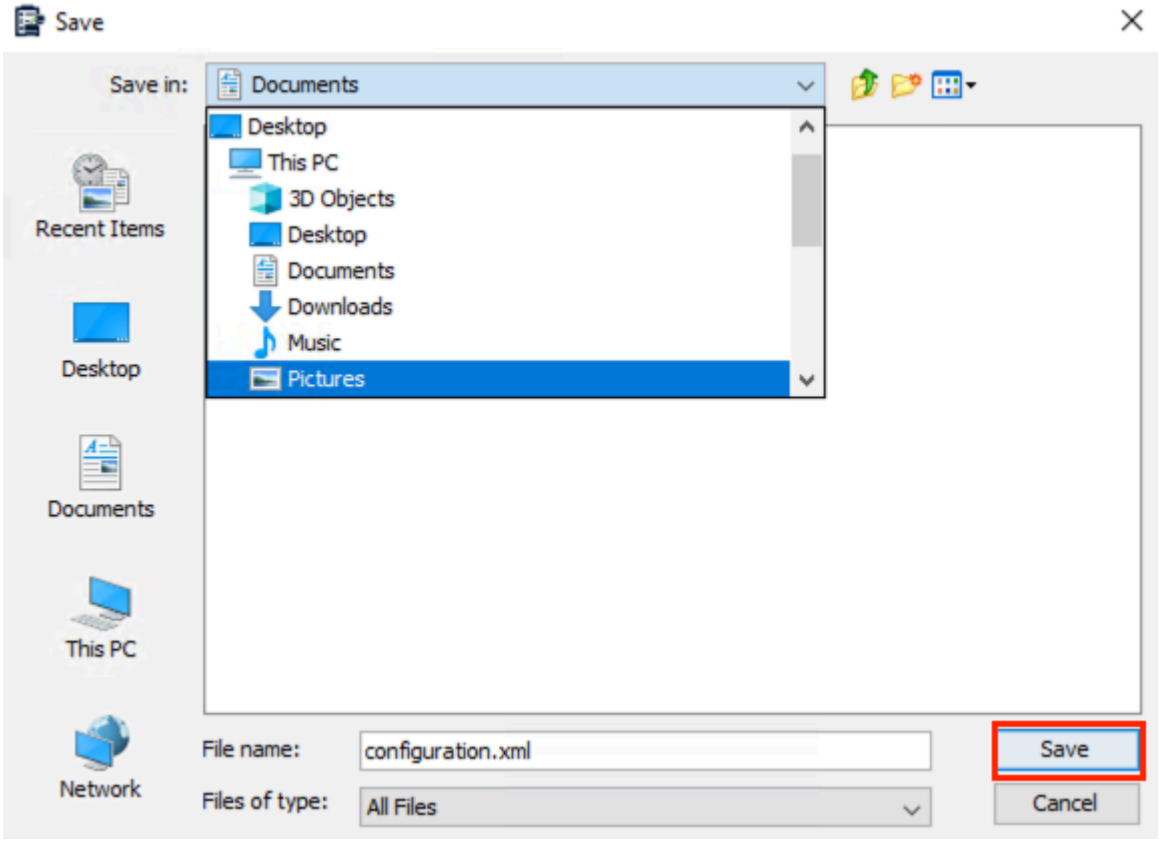
对于此方案，我们将受保护的可扩展身份验证协议(PEAP)与用户凭证。



步骤8.导航至菜单File.使用Save as选项。

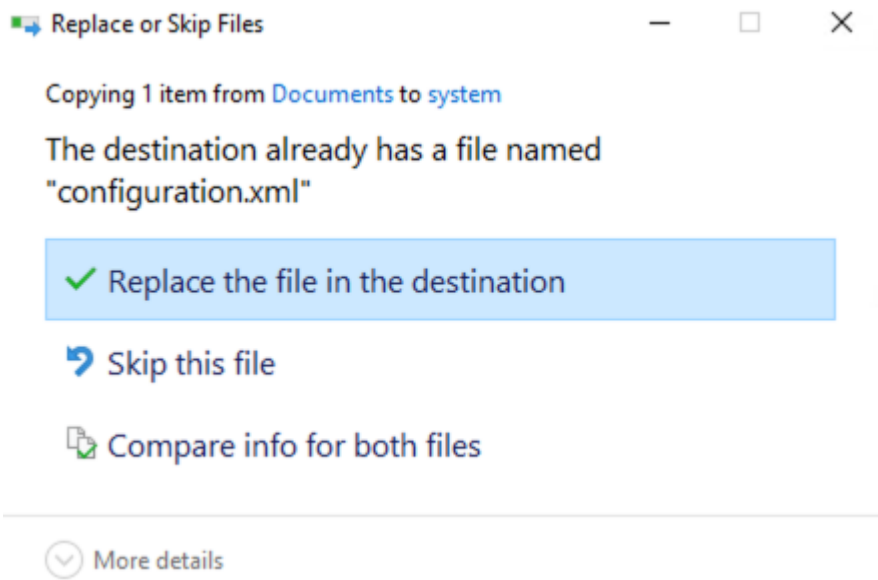
将文件命名为configuration.xml，并保存在ProgramData\Cisco\Cisco Secure Client\Network Access Manager\system以外的文件夹中。

在本示例中，文件保存在Documents文件夹中。保存配置文件。



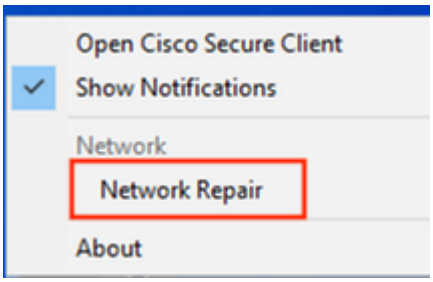
第8步。继续配置文件位置，复制文件，并替换文件夹ProgramData\Cisco\Cisco Secure Client\Network Access Manager\system中包含的文件。

选择替换目标中的文件选项。



第9步。要加载在安全客户端5.0中修改的配置文件，请右键单击Windows计算机右下任务栏中的安全客户端图标。

执行网络修复。

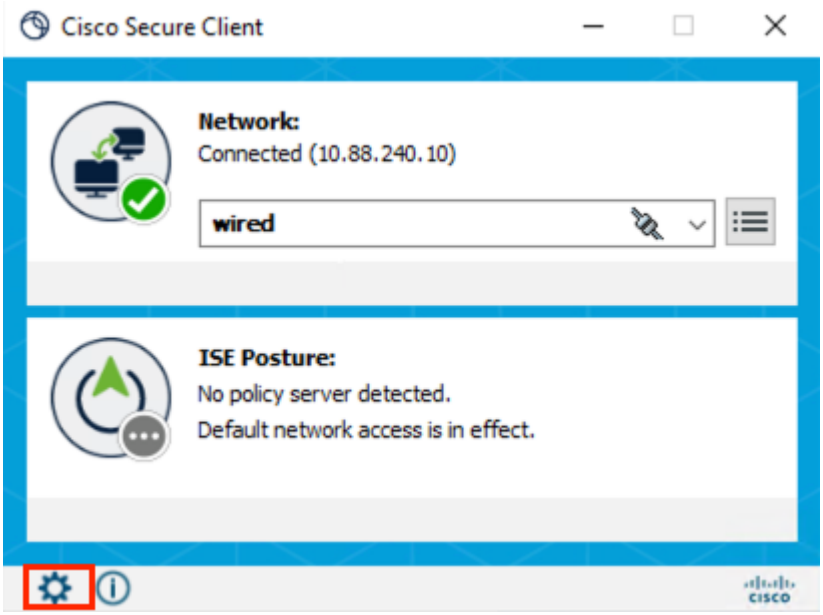


注意：通过配置文件编辑器配置的所有网络都具有Administrator Network的权限，因此用户无法定制/更改使用此工具配置的内容。

使用网络访问管理器设置MKA网络（可选）。

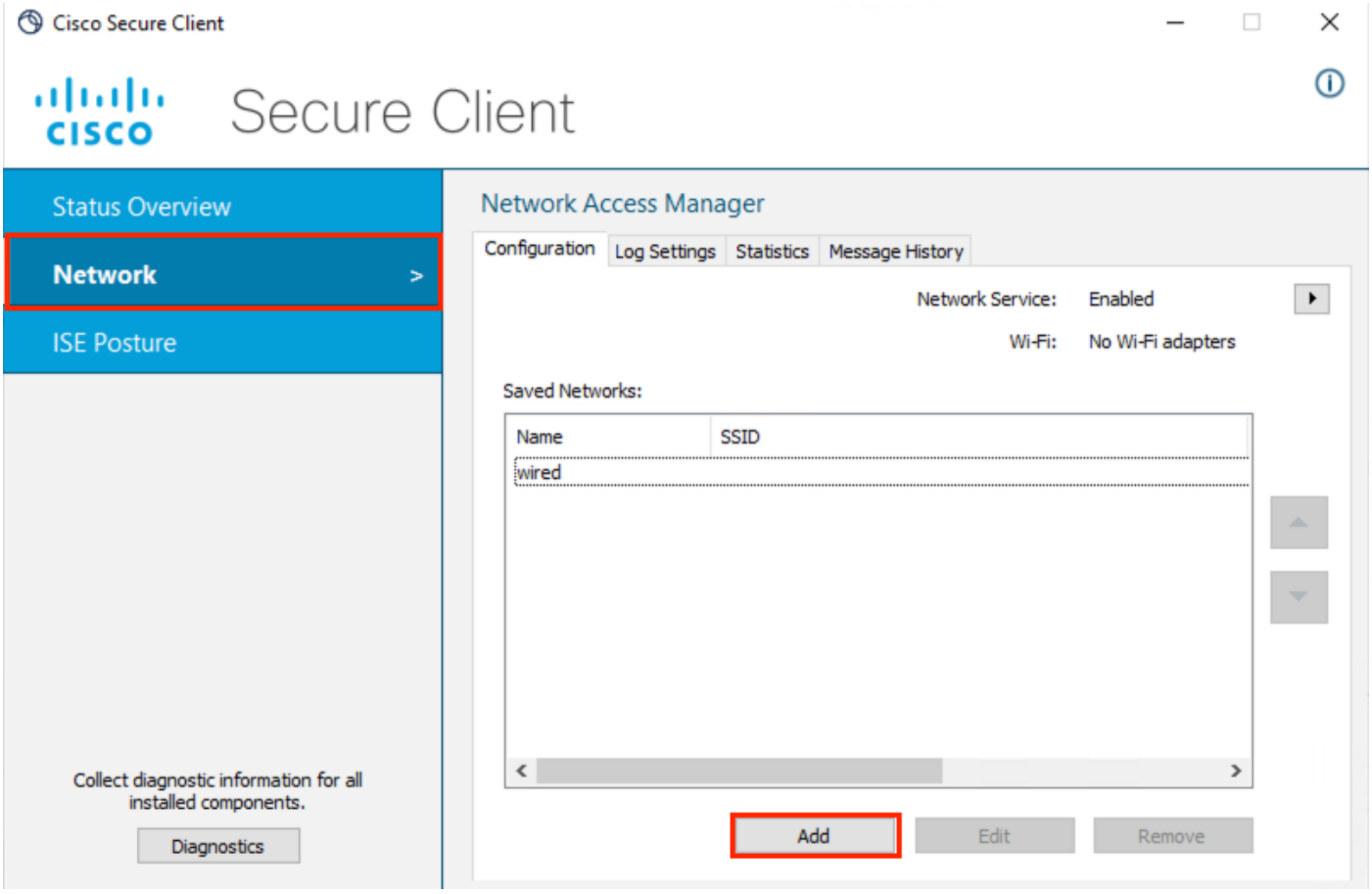
步骤1.作为使用配置文件编辑器进行MKA设置的替代方法，您可以添加网络，而无需使用此工具。

在Secure Client套件中选择齿轮图标。



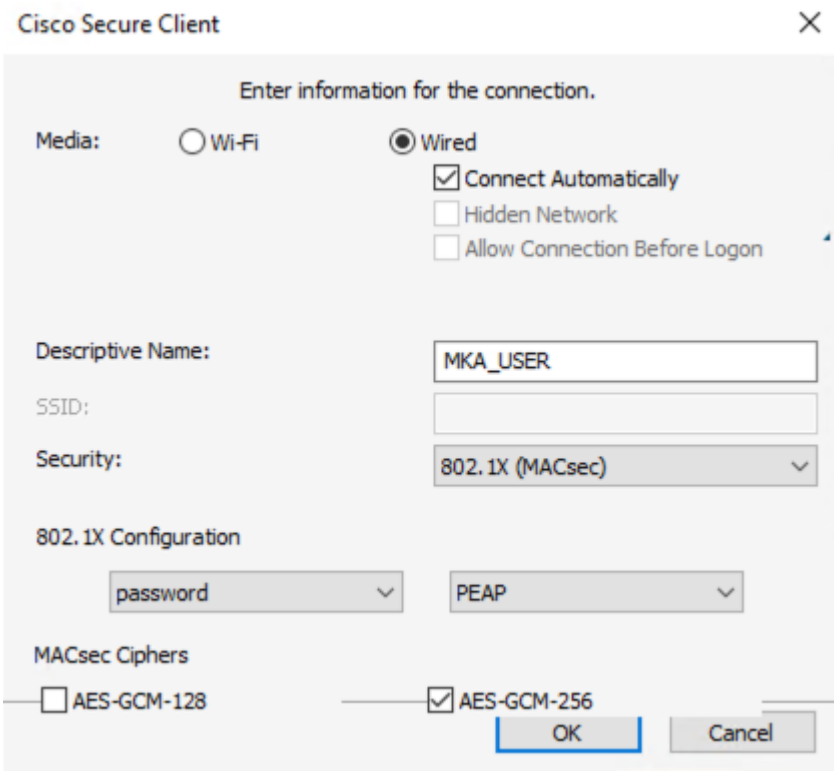
第2步。在显示的新窗口中，选择选项Network。

在Configuration部分中，选择Add选项以入口具有用户网络权限的网络MKA。



步骤3.在新配置窗口中，设置连接特征并命名网络。

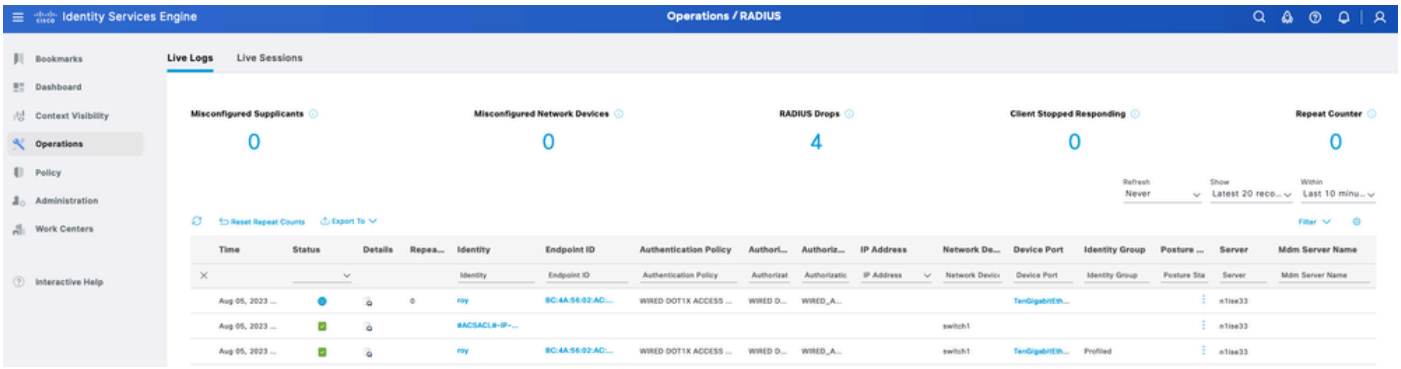
完成后，选择OK按钮。



验证

在ISE上验证

在ISE中，完成此流的配置后，请注意设备在Livelogs中进行身份验证和授权。



导航到Details of the authentication和Result部分。

在授权配置文件中设置的属性将发送到网络接入设备(NAD)并消耗一个基本许可证。

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.

在Catalyst交换机上进行验证。

这些命令可用于验证此解决方案的正确功能。

```
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

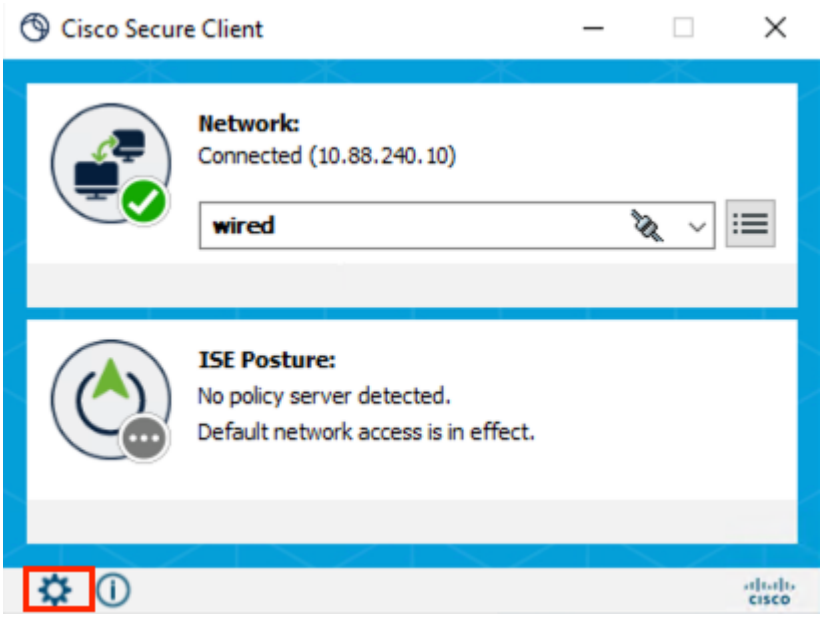
```
switch1#show mka session

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
```

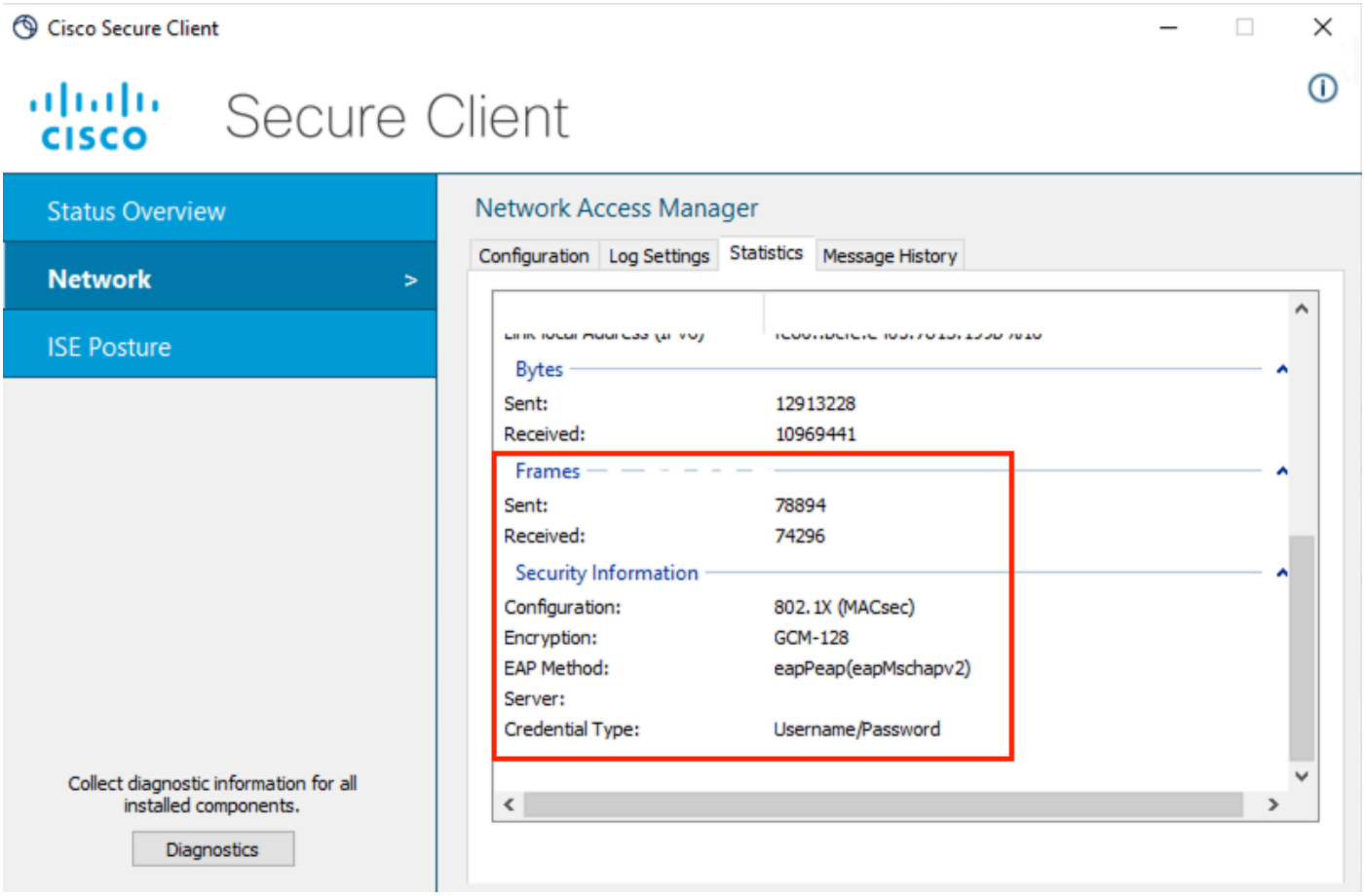
在安全客户端上验证。

使用通过MACsec加密创建的配置文件成功进行身份验证。如果单击引擎图标，则会显示更多信息。



在此安全客户端显示的菜单中，在Network Access Manager > Statistics部分中，注意Encryption和相应的MACsec配置。

在第2层执行加密时，接收和发送的帧会增加。

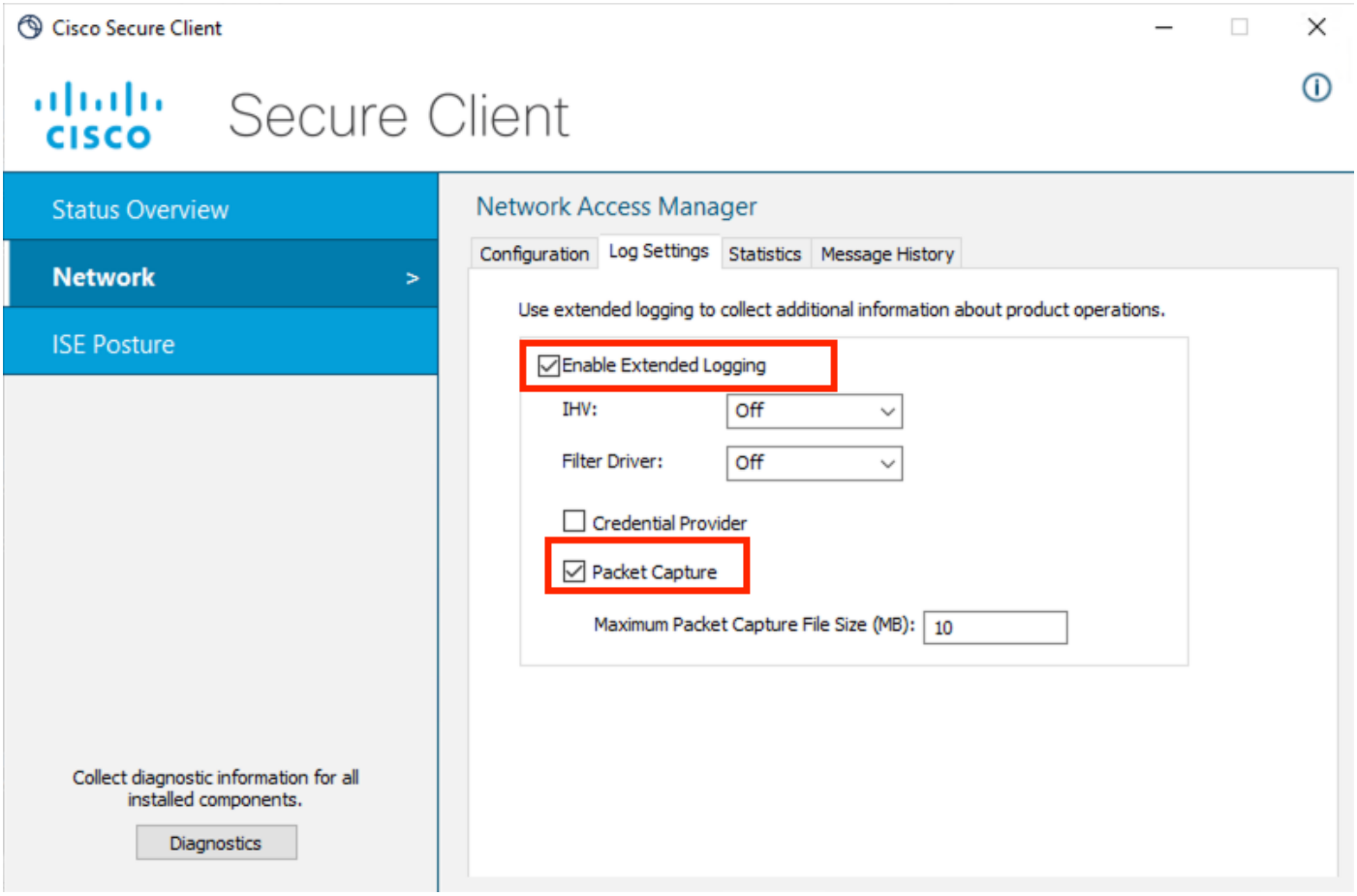


故障排除

注意：本节介绍与可能出现的MKA问题相关的故障排除部分。如果遇到身份验证或授权故障，请参阅ISE安全有线接入规范部署指南 — 故障排除以进行调查。本指南假定身份验证在没有MACsec加密的情况下运行正常。

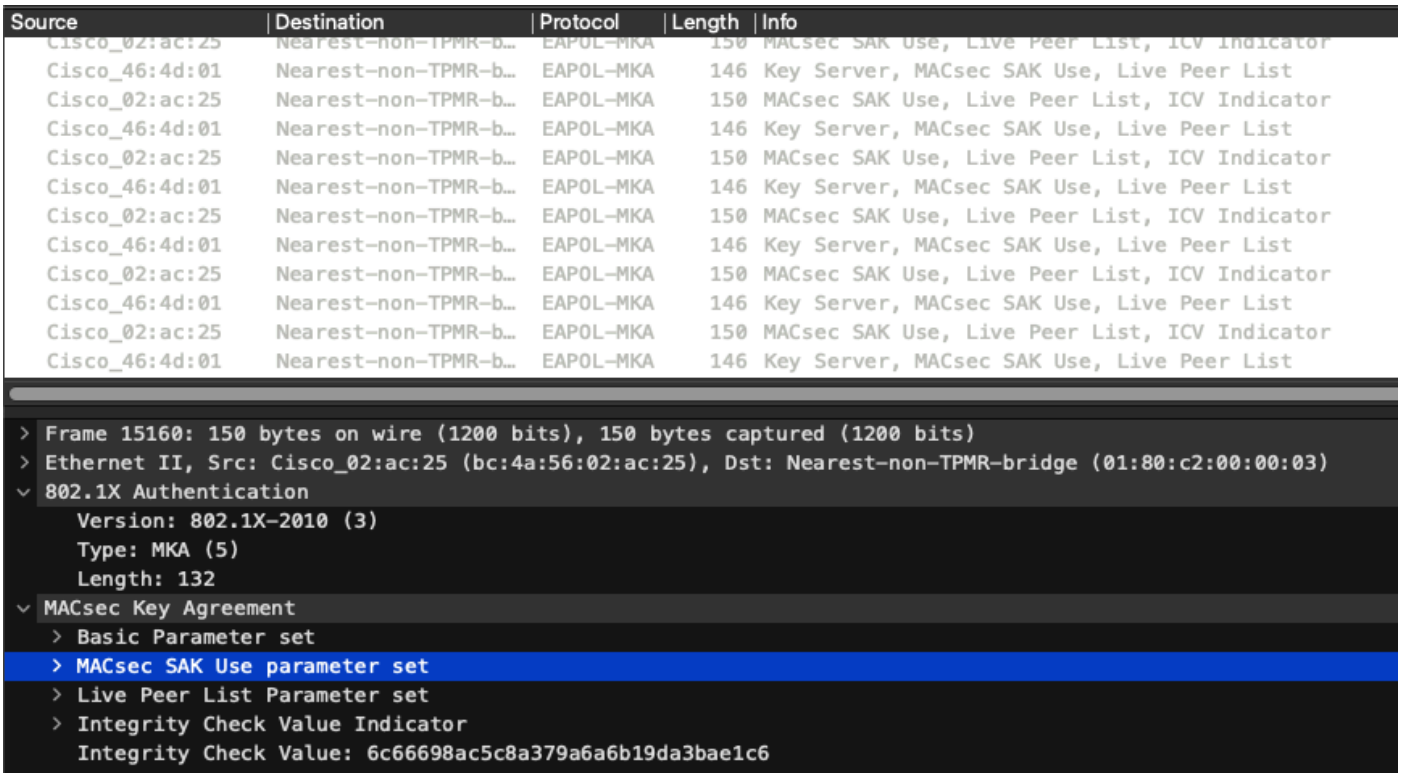
Cisco Secure Endpoint

- 在安全终端套件中启用DART模块。
- 在此菜单中，启用扩展日志记录以收集有关用户MKA连接的更多数据。您还可以启用DART捆绑包中包含的数据包捕获。



- 收集DART捆绑包，以继续分析configuration.xml和网络访问管理器。请参阅文档运行DART以收集数据以进行故障排除

此示例显示如何将数据包视为主机和交换机之间的信息加密（例如IPv6和IPv6）：



从DART捆绑包中，我们可以在名为NetworkAccessManager.txt的日志中找到有关身份验证802.1X和MKA会话的有用信息。

此信息显示在使用MKA加密的身份验证成功中。

```
%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)
```

Cisco IOS故障排除

这些命令可在网络访问设备(NAD)中实施，以检查平台和请求方之间的MKA加密。

有关命令的详细信息，请查看用作NAD的平台的相应配置指南。

```
#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
```

这些是到主机的一个成功MKA连接的调试。您可以将此作为引用出现：

```
%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 A
%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
```

身份服务引擎(ISE)故障排除

与此功能相关的故障排除仅限于提供cisco-av-pair属性linksec-policy=should-secure。

确保授权结果会将该信息发送到链接到设备所连接的交换机端口的Radius会话。

有关ISE的进一步身份验证分析，请参阅在ISE上排除故障并启用调试

常见问题

密码不匹配

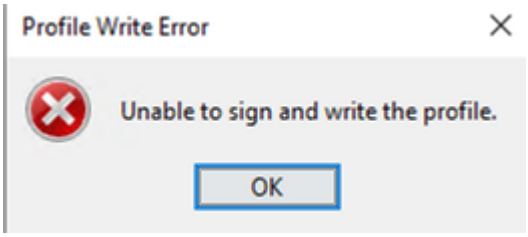
此日志可在NAD的MKA调试中看到。

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

在此场景中，首先要验证的是交换机的MKA策略和安全客户端配置文件中配置的密码是否匹配。

对于AES-GCM-256加密，需要满足这些要求（根据文档）《Cisco Secure Client（包括AnyConnect）管理员指南》，版本5

无法保存configuration.xml。



通过使用网络访问管理器配置文件编辑器保存名为MKA的设置的configuration.xml（如前所述），可以解决与配置文件写入错误相关的问题。

该错误与无法修改使用的文件configuration.xml有关。因此，请将文件保存在其他位置以继续替换配置文件。

相关信息

- 配置MACsec加密
- 使用Cat9k和ISE配置MACsec交换机到主机

关于此翻译
思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。