Configurazione e risoluzione dei problemi di MKA con Secure Client 5

Sommario

Introduzione

In questo documento viene descritto come configurare la crittografia MACsec in un endpoint utilizzando Secure Client 5 come supplicant.

Prerequisiti

Cisco raccomanda la conoscenza dei seguenti argomenti:

- Identity Services Engine
- 802.1x e Radius
- Crittografia MACsec MKA
- Secure Client versione 5 (in precedenza Anyconnect)

Componenti usati

Le informazioni fornite in questo documento si basano sulle seguenti versioni software e hardware:

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

Premesse

MACSec (Media Access Control Security) è uno standard di sicurezza di rete che fornisce crittografia e protezione per i frame Ethernet al livello 2 del modello OSI (Data Link), definito dallo standard 802.1AE IEEE.

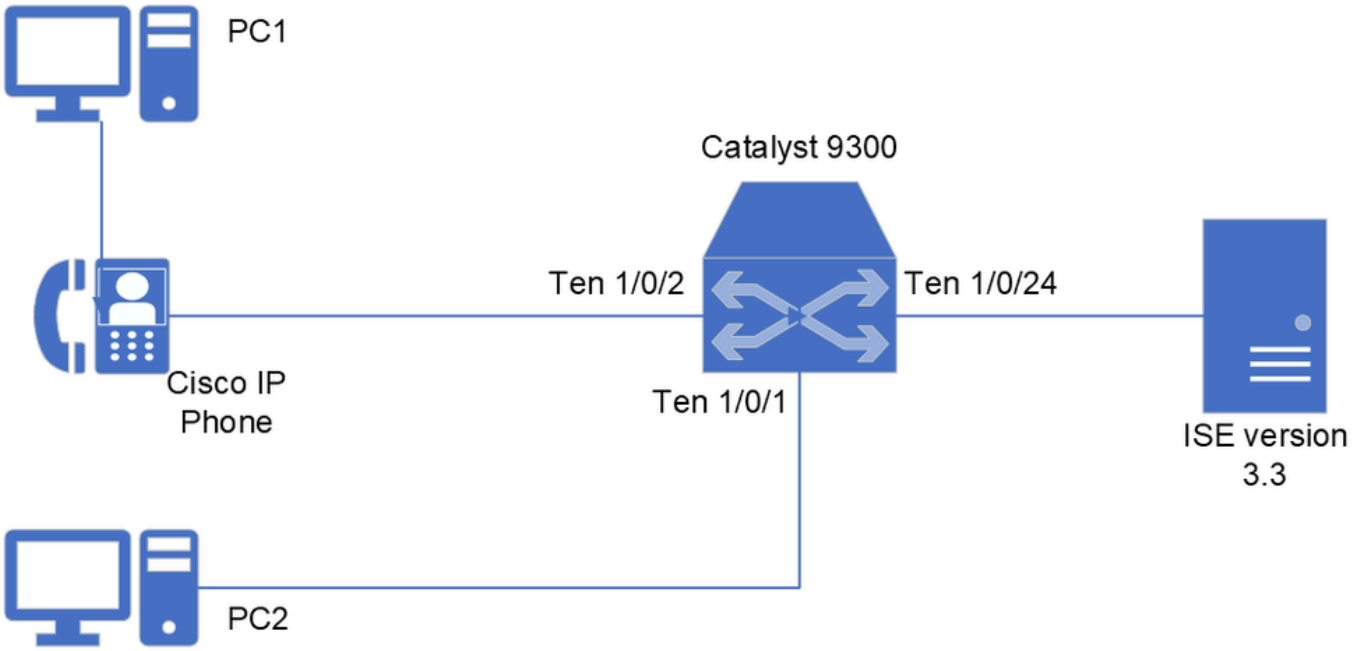
MACSec fornisce questa crittografia in una connessione point-to-point che può essere di tipo switch-to-switch o switch-to-host, pertanto la copertura di questo standard è limitata alle connessioni cablate.

Questo standard cripta tutti i dati ad eccezione dell'indirizzo MAC di origine e di destinazione dei frame trasmessi in una connessione di layer 2.

Il protocollo MKA (MACsec Key Agreement) è il meccanismo da cui i peer MACsec negoziano le chiavi di sicurezza necessarie per proteggere il collegamento.

Configurazione

Esempio di rete



Nota: Nella presente documentazione si presume che le regole siano state configurate e funzionino per l'autenticazione Radius per i dispositivi PC e i telefoni IP Cisco. Per configurare una configurazione da zero, consultare la guida alla distribuzione prescrittiva dell'accesso cablato sicuro ISE per esaminare la configurazione in Identity Services Engine and Switch for Identity-Based Network Access.

Configurazione delle policy su ISE

La prima operazione da eseguire è configurare i profili di autorizzazione corrispondenti che vengono applicati a entrambi i PC (nonché al telefono IP Cisco) visualizzati nel diagramma precedente.

In questo scenario ipotetico, i PC utilizzeranno il protocollo 802.1X come metodo di autenticazione e il Cisco IP Phone utilizzerà il Mac Address Bypass (MAB).

ISE comunica con lo switch tramite il protocollo Radius sugli attributi che lo switch deve applicare nell'interfaccia da cui l'endpoint è connesso tramite una sessione Radius.

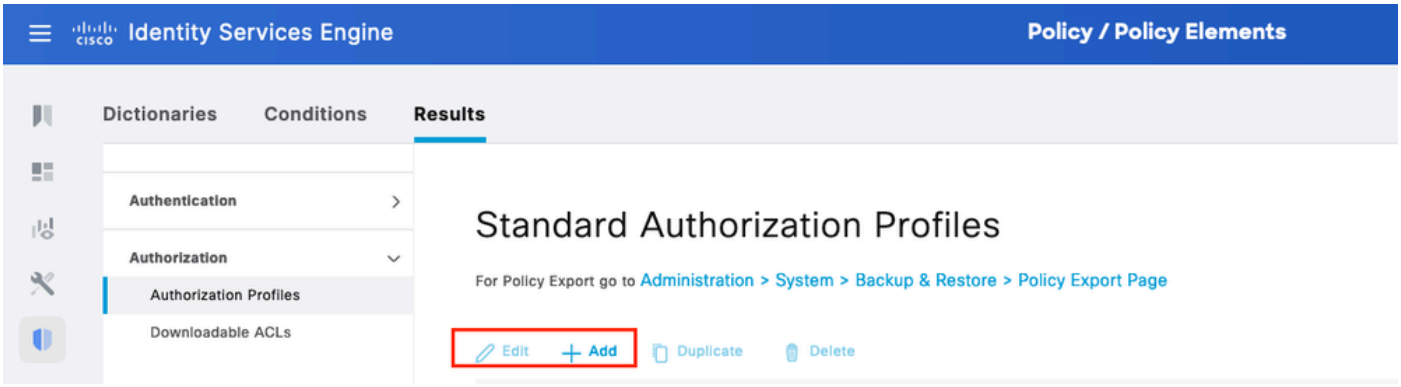
Per la crittografia MACsec negli host, l'attributo richiesto è cisco-av-pair = linksec-policy, che ha i seguenti 3 valori possibili:

- Should-not-Secure: Lo switch non esegue la crittografia MKA nell'interfaccia in cui è in corso la sessione Radius.
- Must-Secure: Lo switch deve applicare la crittografia nel traffico collegato alla sessione Radius. Se la sessione MKA ha esito negativo o ha un timeout, la connessione viene considerata come un errore di autorizzazione e viene eseguito un altro tentativo di stabilire la sessione MKA.
- Should-Secure: Lo switch cerca di eseguire la crittografia MKA. Se la sessione MKA collegata alla sessione Radius ha esito positivo, il traffico viene crittografato. In caso di errore o di timeout della MKA, lo switch consente il traffico non crittografato collegato alla sessione Radius.

Passaggio 1. In entrambi i PC, applicare una policy MKA che deve essere protetta per avere flessibilità nel caso in cui un computer senza funzionalità MKA si connetta all'interfaccia Ten 1/0/1.

In alternativa, è possibile configurare un criterio per PC2 che applichi un criterio che richiede protezione obbligatoria.

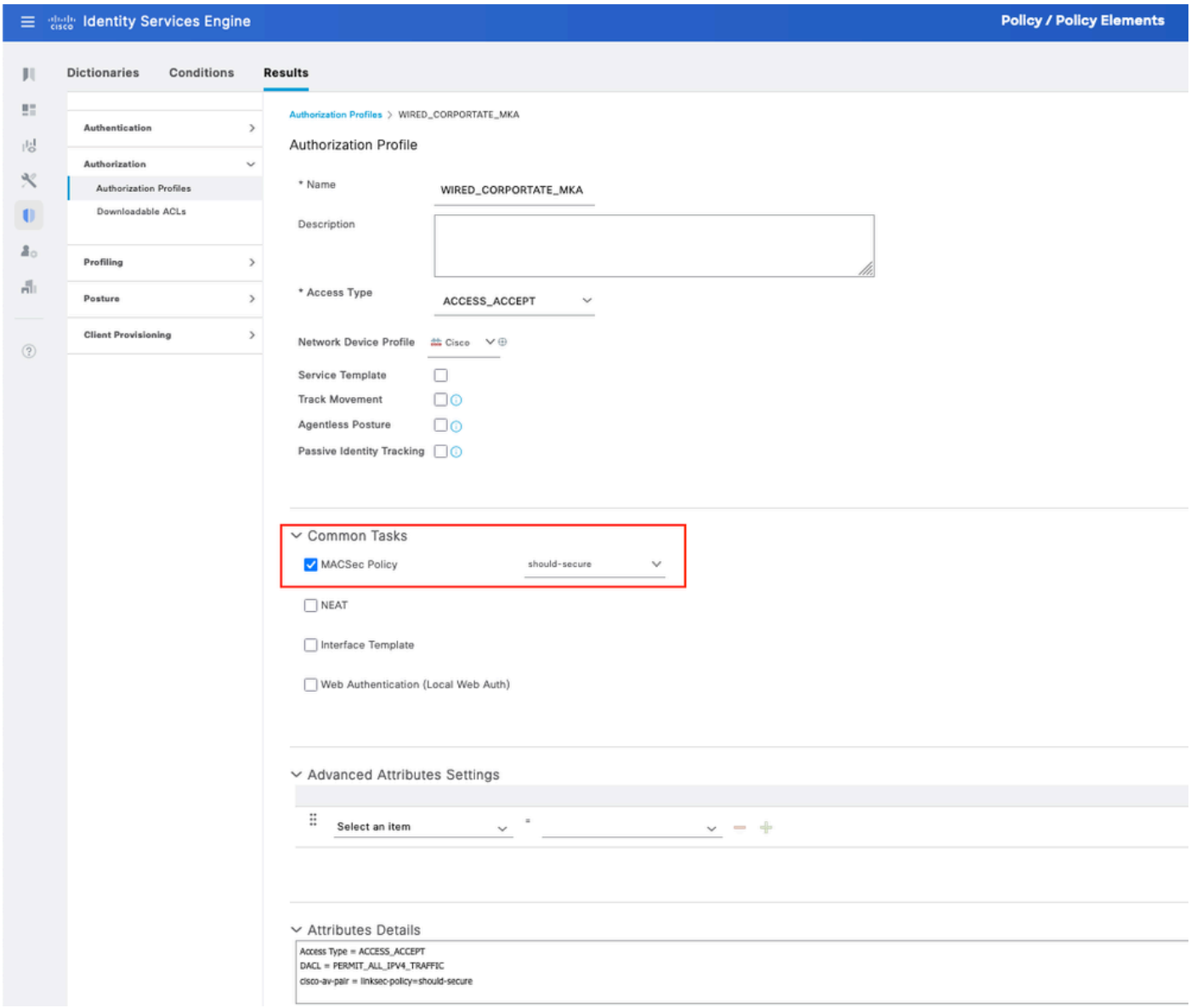
In questo esempio, configurare il criterio per i PC come in Criterio > Elementi criterio > Risultati > Profili di autorizzazione, quindi +Aggiungere o modificare un profilo esistente



Passaggio 2. Completare o personalizzare i campi obbligatori per il profilo.

Assicurarsi che in Operazioni comuni sia stata selezionata l'opzione Criterio MACSec e il criterio corrispondente da applicare.

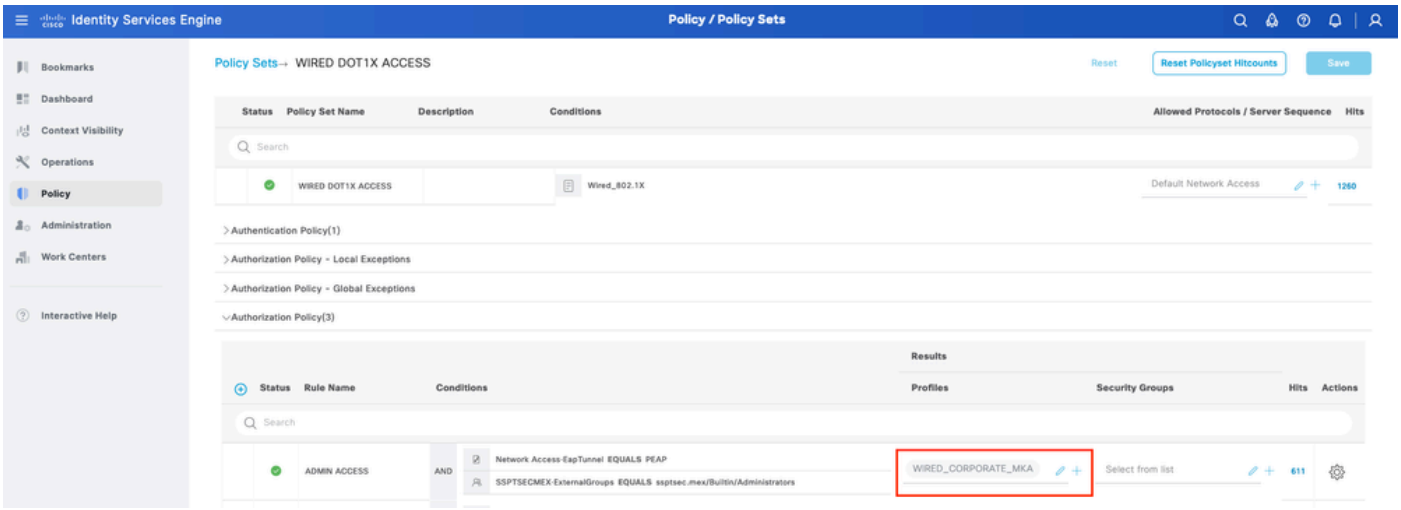
Scorrere verso il basso e salvare la configurazione.



Passaggio 3. Assegnare il profilo di autorizzazione corrispondente alle regole di autorizzazione interessate dai dispositivi.

Passare a Criterio > Set di criteri > (Selezionare Set di criteri assegnato) > Criterio di autorizzazione.

Associare la regola di autorizzazione al profilo di autorizzazione con le impostazioni MACsec. Scorrere verso il basso e salvare la configurazione.



Configurazione di MKA in Catalyst 9300

Passaggio 1. Configurare un nuovo criterio MKA come suggerito nell'esempio:

```
!
  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
!
```

Passaggio 2. Abilitare la crittografia MACsec nell'interfaccia a cui sono connessi i PC.

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

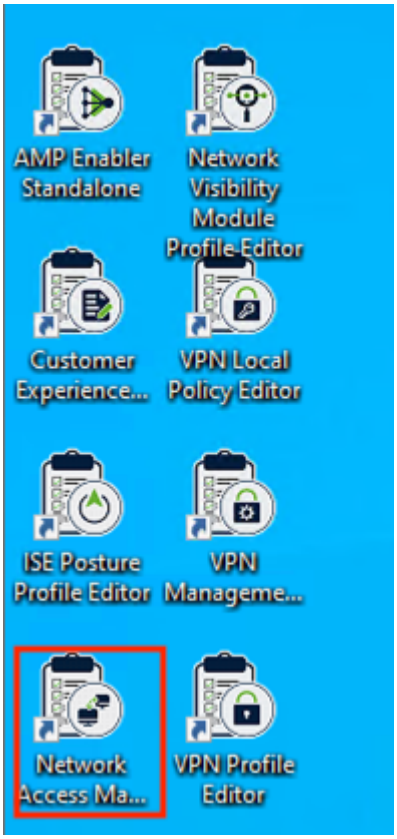
Nota: Per ulteriori informazioni relative ai comandi e alle opzioni nella configurazione MKA, consultare la guida alla configurazione della sicurezza corrispondente alla versione dello switch in uso. In questo scenario, ad esempio, Guida alla configurazione della

sicurezza, Cisco IOS XE Bengaluru 17.6.x (switch Catalyst 9300)

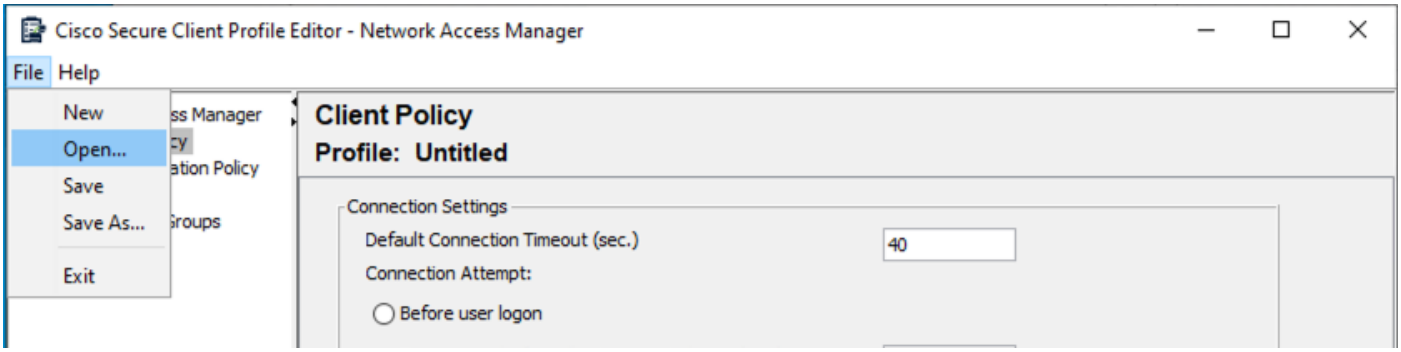
Installazione di MKA mediante l'Editor profili di Network Access Manager.

Passaggio 1. Scaricare il file (dal sito di download di Cisco) e aprire l'Editor di profili corrispondente alla versione di Secure Client in uso.

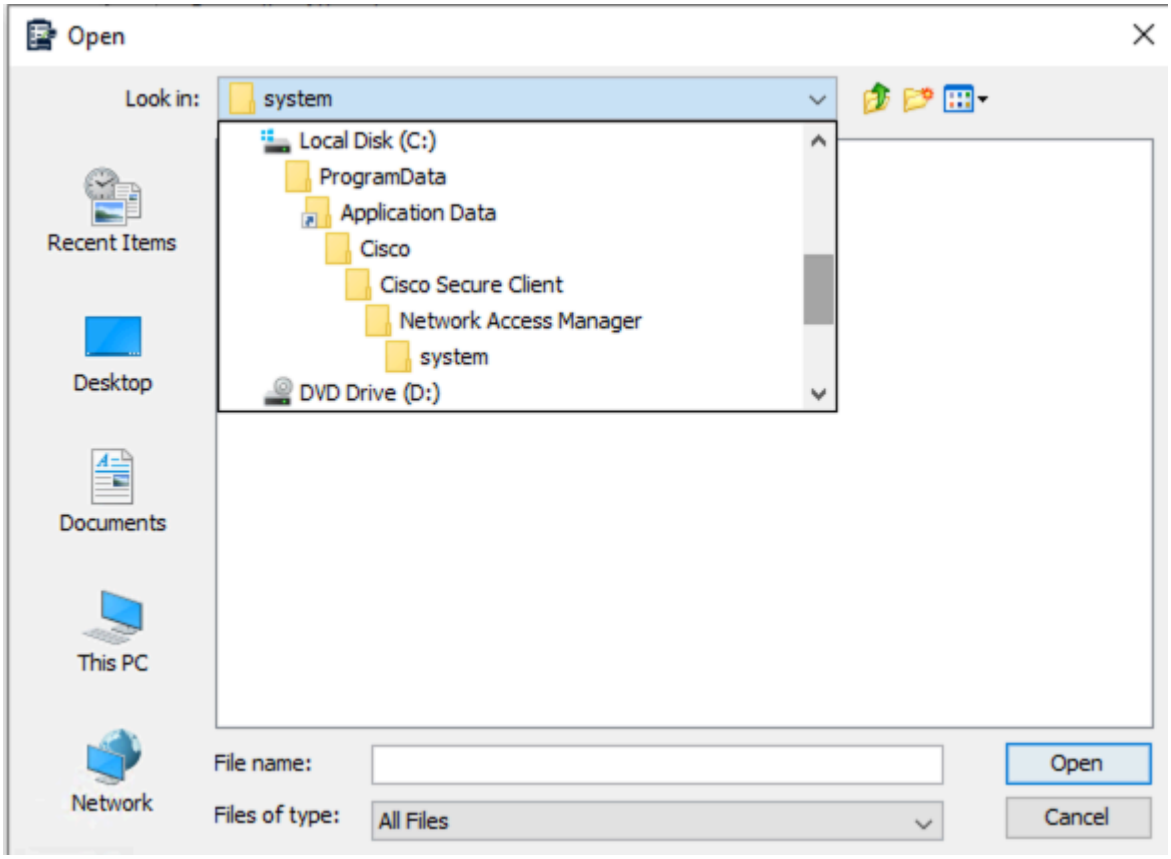
Dopo aver installato il programma nel computer, aprire Cisco Secure Client Profile Editor - Network Access Manager.



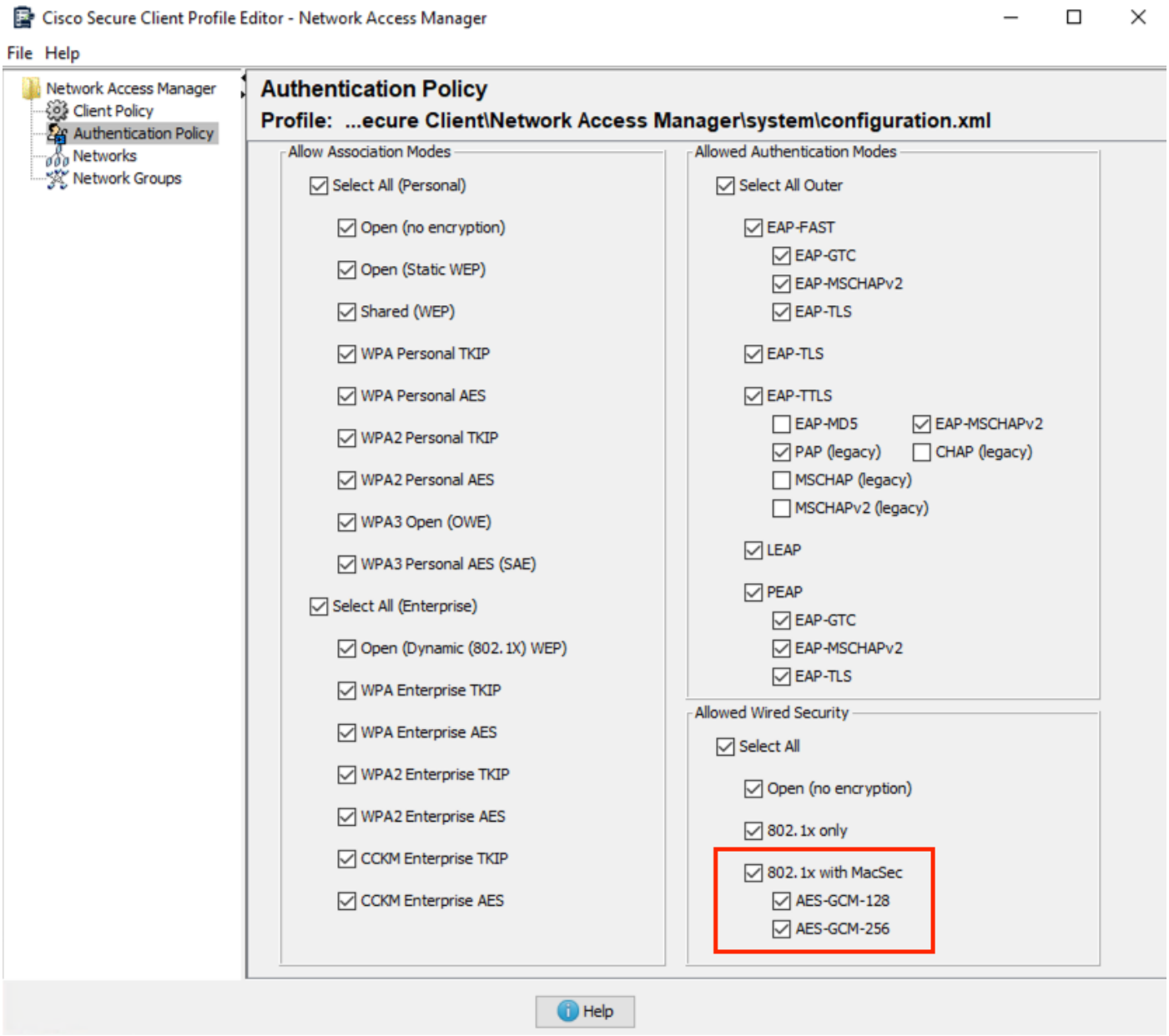
Passaggio 2. Selezionare File > Apri.



Passaggio 3. Selezionare il sistema di cartelle visualizzato nell'immagine. All'interno di questa cartella aprire il file configuration.xml.

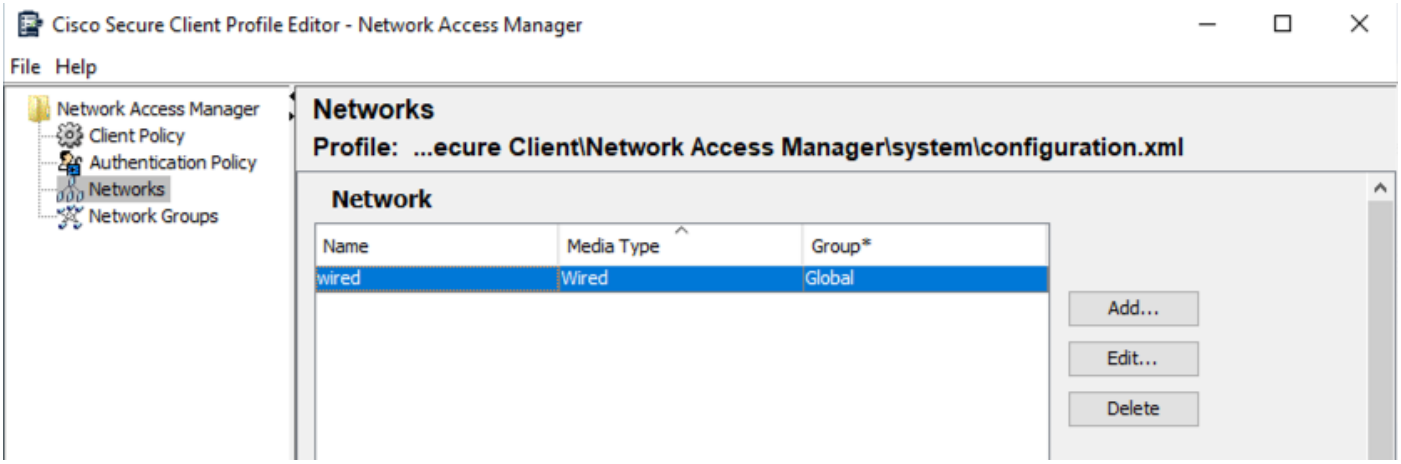


Passaggio 4. Una volta caricato il file dall'Editor di profili, selezionare l'opzione Criteri di autenticazione e verificare che l'opzione relativa a 802.1x con MACSec sia attivata.



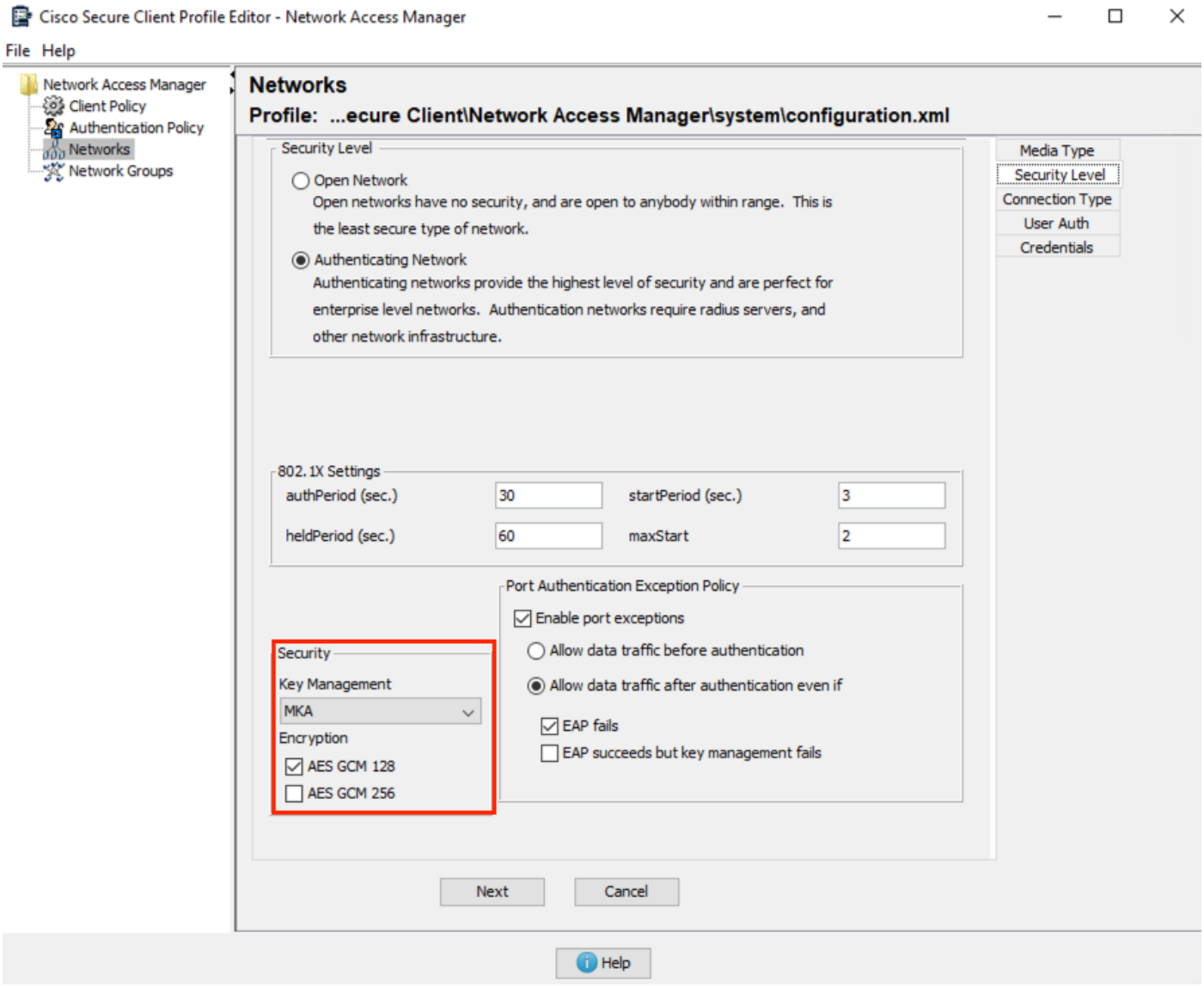
Passaggio 5. Procedere alla sezione Reti. In questa parte è possibile aggiungere un nuovo profilo per una connessione cablata o modificare il profilo predefinito installato con Secure Client 5.0.

In questo scenario verrà modificato il profilo cablato esistente.



Passaggio 6. Configurare il profilo. Nella sezione Livello di protezione, regolare Gestione chiavi in modo da utilizzare MKA seguito da una crittografia AES GCM 128.

Regolare anche gli altri parametri per il dot1x di autenticazione e i criteri.

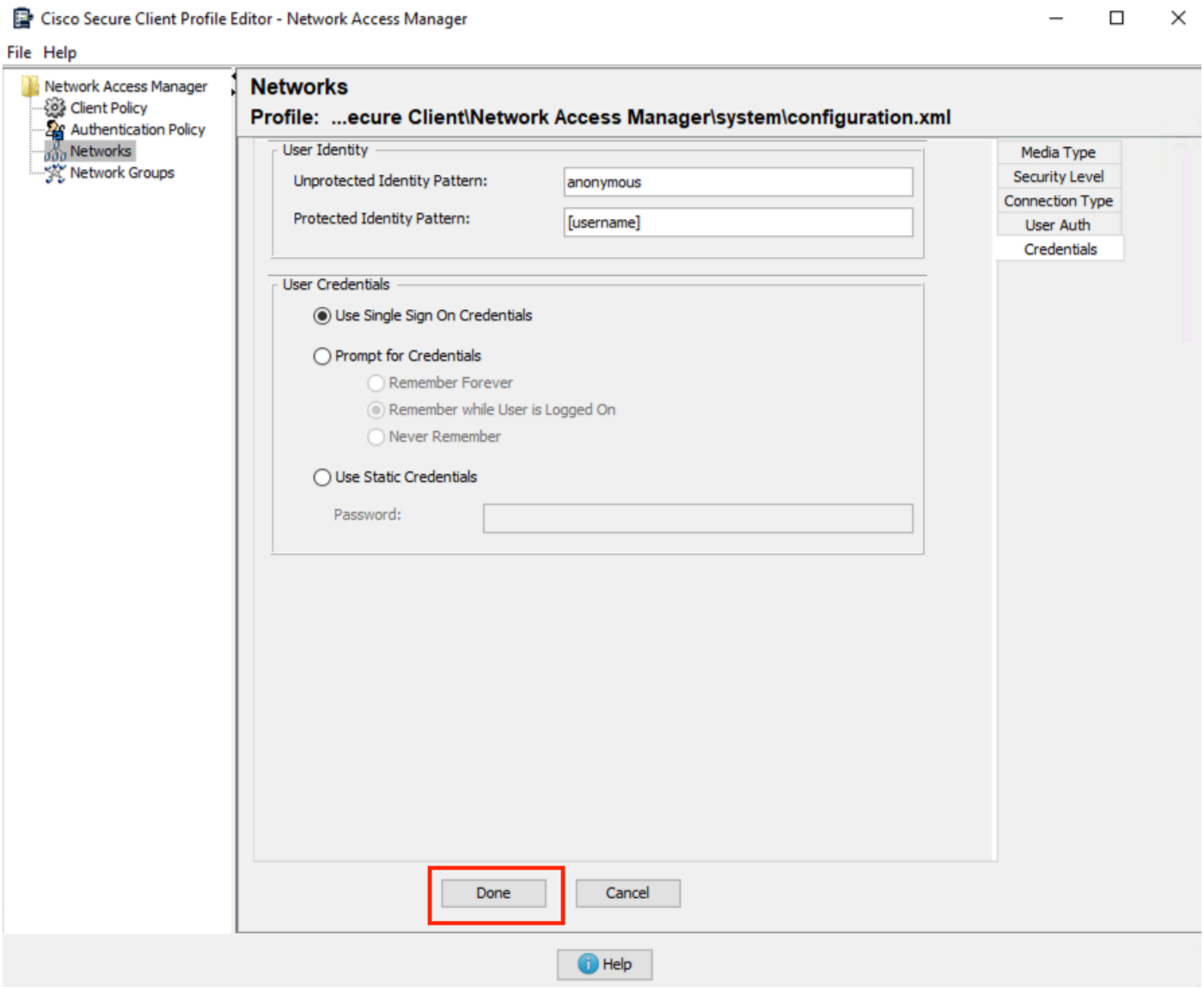


Passaggio 7. Configurare le sezioni rimanenti relative a Tipo di connessione, Autenticazione utente e Credenziali.

Queste sezioni variano a seconda delle impostazioni di autenticazione selezionate nella sezione Livello di protezione.

Al termine, fate clic su Fine (Done).

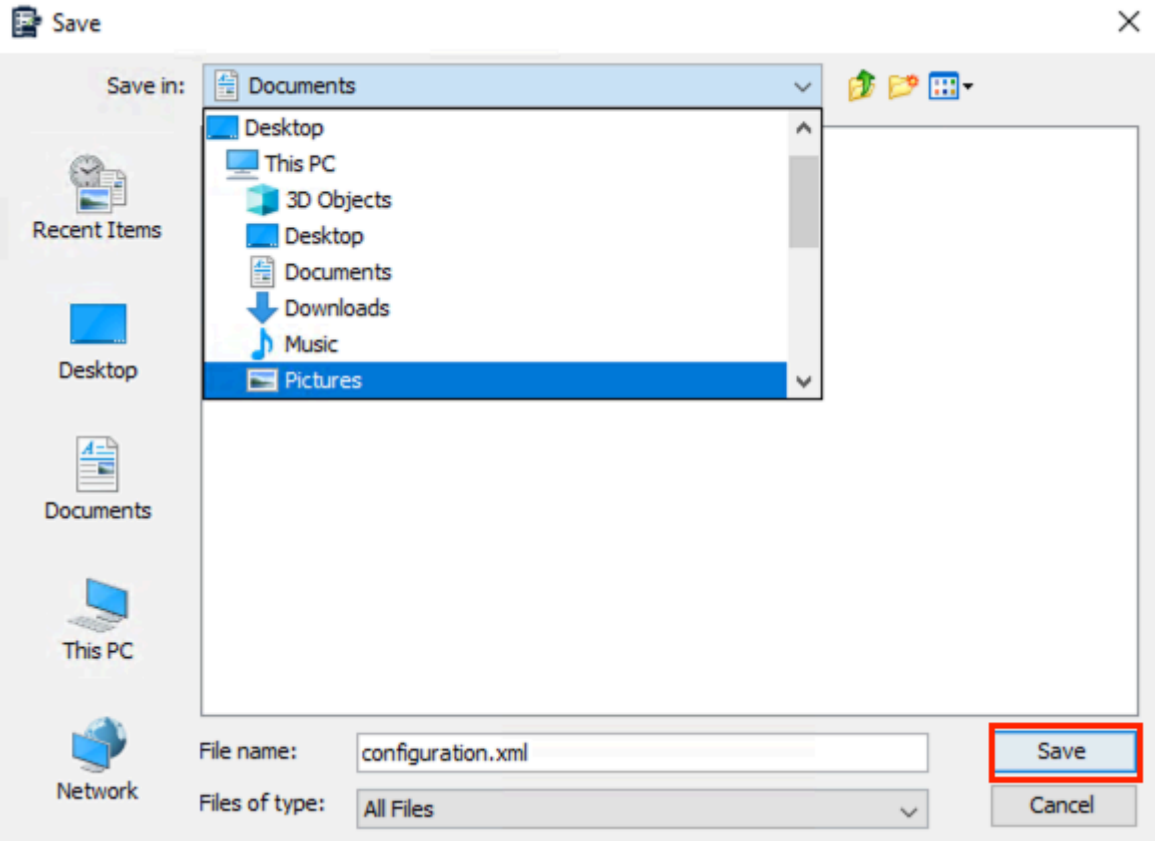
Per questo scenario viene utilizzato il protocollo PEAP (Protected Extensible Authentication Protocol) con le credenziali utente.



Passaggio 8. Passare al menu File. Continuare con l'opzione Salva con nome.

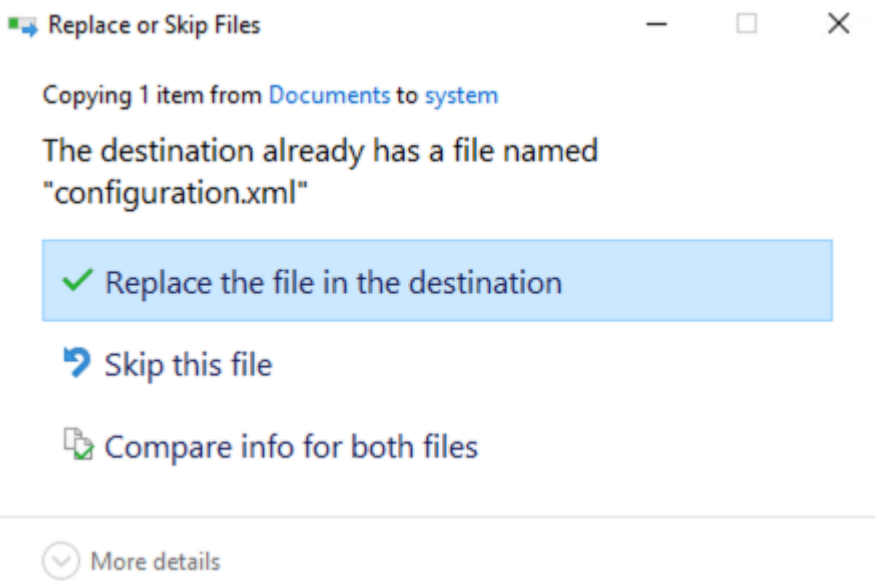
Assegnare al file il nome configuration.xml e salvarlo in una cartella diversa da ProgramData\Cisco\Cisco Secure Client\Network Access Manager\system.

In questo esempio, il file è stato salvato nella cartella Documenti. Salvare il profilo.



Passaggio 8. Passare al percorso del profilo, copiare il file e sostituire il file contenuto nella cartella ProgramData\Cisco\Cisco Secure Client\Network Access Manager\system.

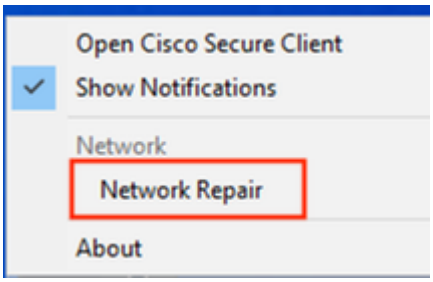
Selezionare l'opzione Sostituisci il file nella destinazione.



Passaggio 9. Per caricare il profilo modificato in Security Client 5.0, fare clic con il pulsante destro

del mouse sull'icona Secure Client nella barra delle applicazioni inferiore destra del computer Windows.

Eseguire un ripristino della rete.

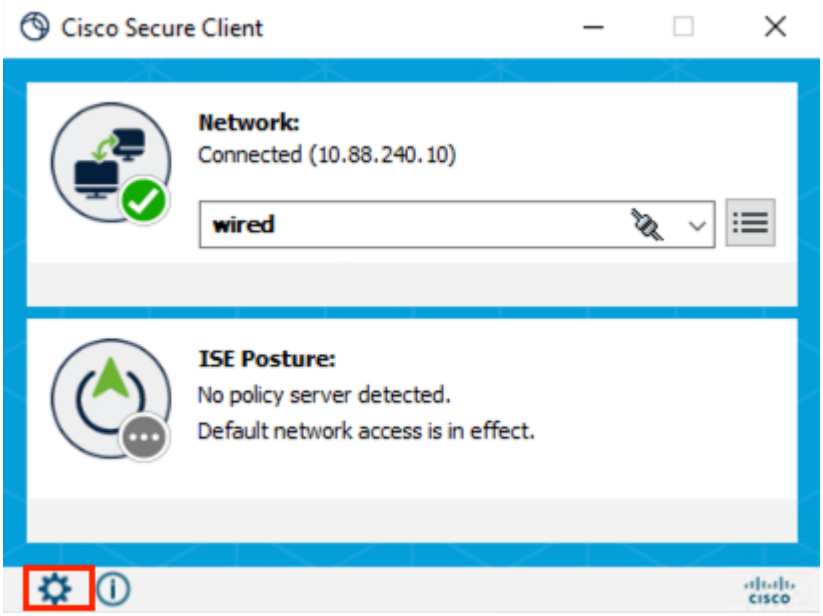


Nota: Tutte le reti configurate tramite l'editor dei profili dispongono dei privilegi di rete dell'amministratore, pertanto gli utenti non sono in grado di personalizzare/modificare il contenuto configurato utilizzando questo strumento.

Configurazione di reti MKA mediante Network Access Manager (opzionale).

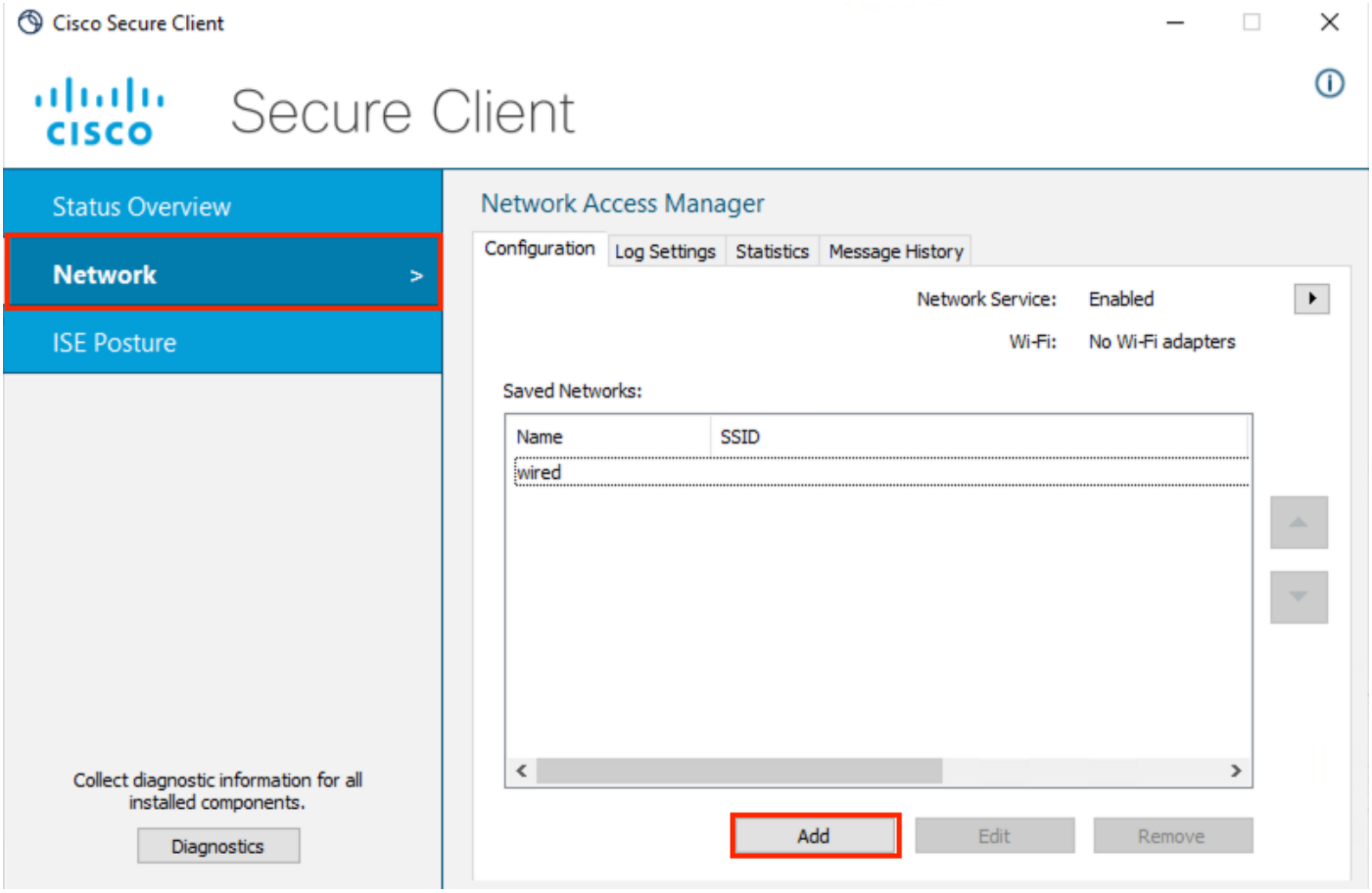
Passaggio 1. In alternativa all'installazione di MKA con l'Editor di profili, è possibile aggiungere reti senza utilizzare questo strumento.

Dalla suite Secure Client, selezionare l'icona gear.



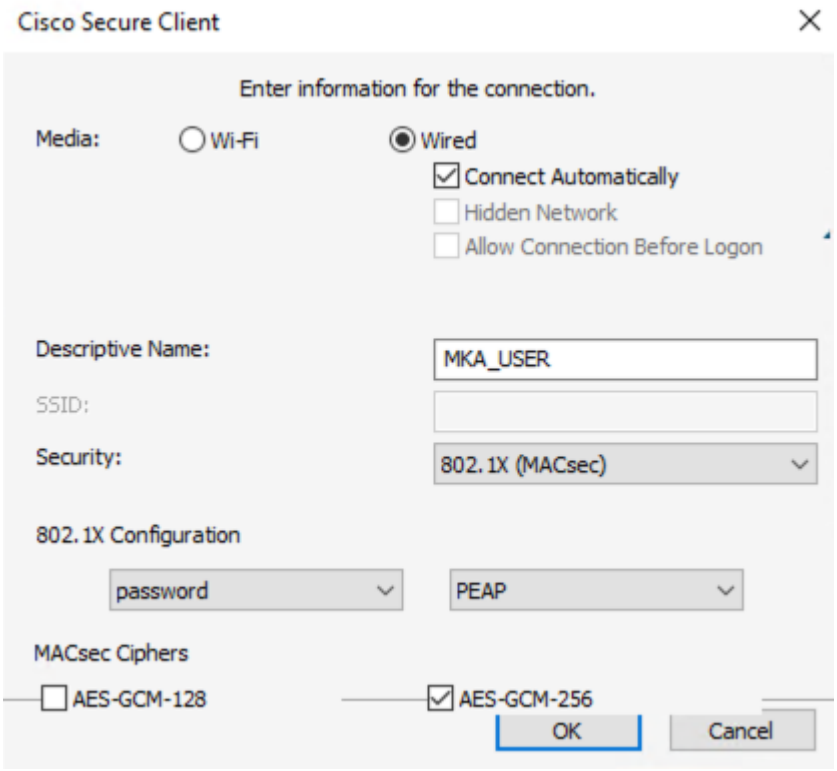
Passaggio 2. Nella nuova finestra visualizzata, selezionare l'opzione Rete.

Nella sezione Configurazione, selezionare l'opzione Add per accedere a una rete con capacità MKA con privilegi User Network.



Passaggio 3. Nella nuova finestra di configurazione, impostare le caratteristiche della connessione e assegnare un nome alla rete.

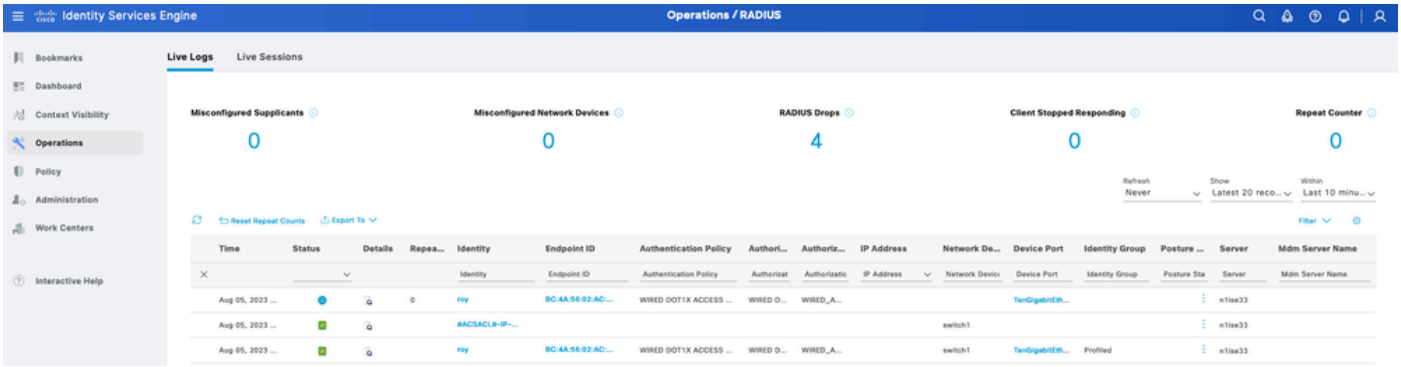
Al termine, selezionare il pulsante OK.



Verifica

Convalida su ISE

In ISE, una volta completata la configurazione del flusso, notare che il dispositivo è stato autenticato e autorizzato in Livelogs.



Spostarsi nella sezione Dettagli dell'autenticazione e Risultato.

Gli attributi impostati nel profilo di autorizzazione vengono inviati al dispositivo NAD (Network Access Device) e viene utilizzata una licenza 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.

Convalida sullo switch Catalyst.

Questi comandi possono essere utilizzati per convalidare la corretta funzionalità di questa soluzione.

```
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
```

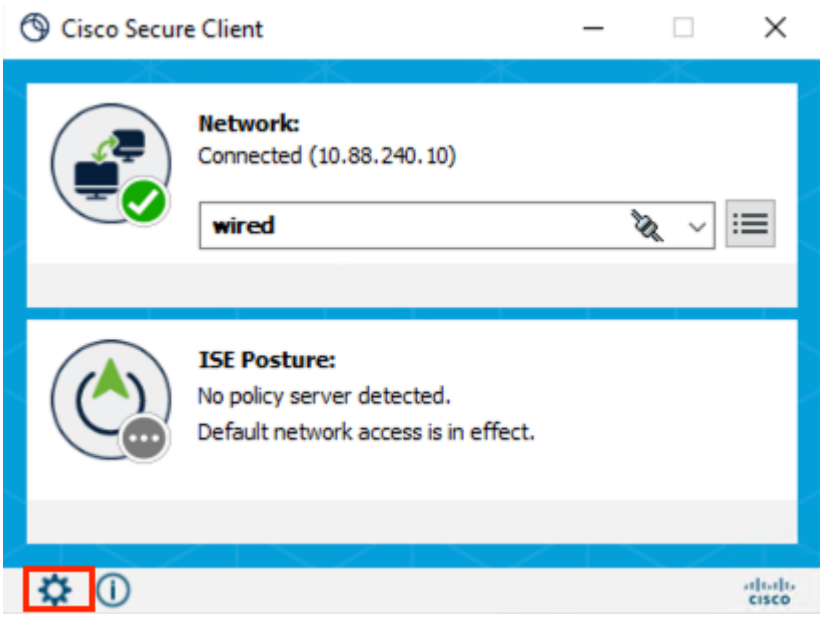
```
switch1#show mka session

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
```

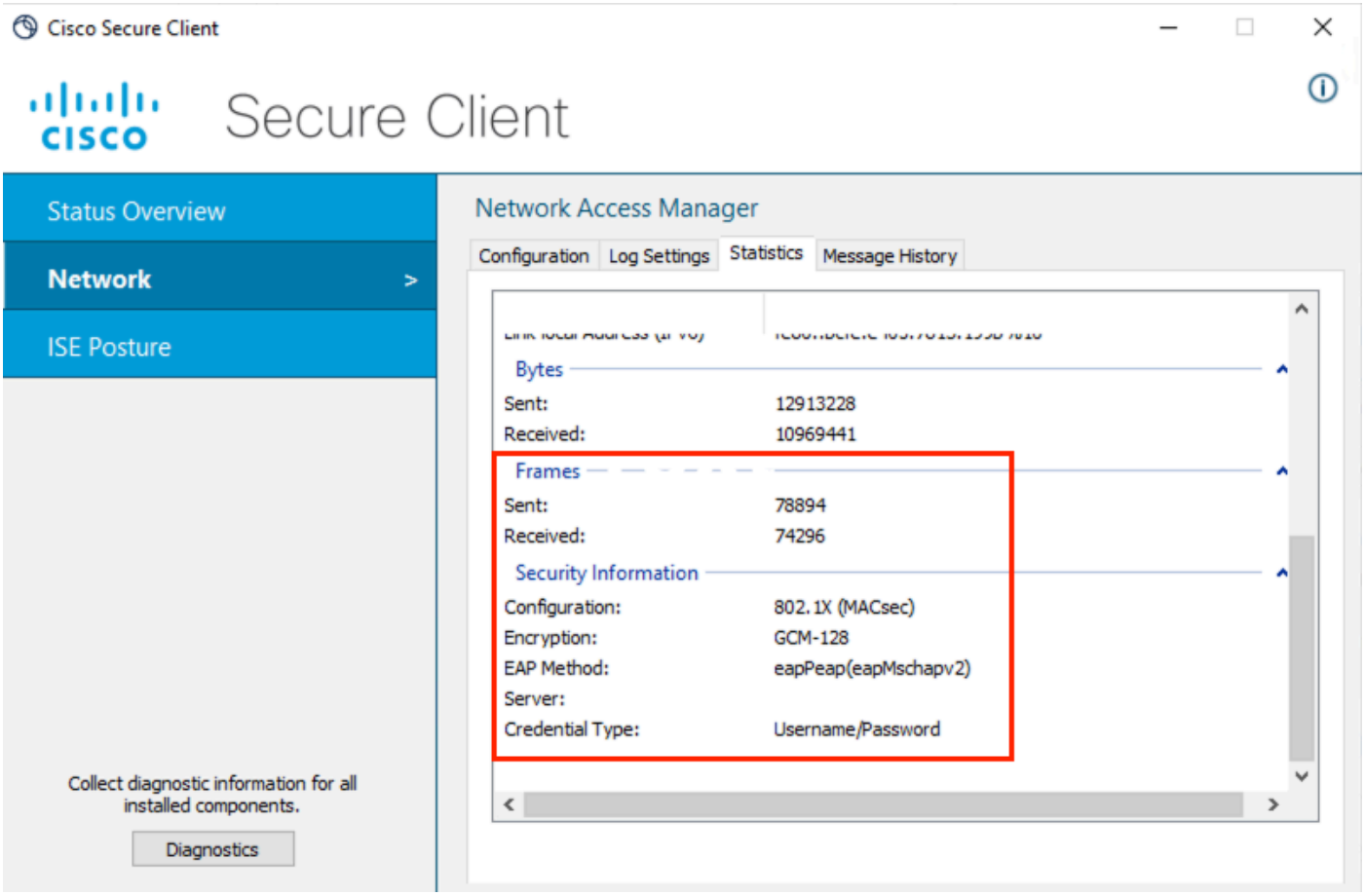
Convalida su client protetto.

Autenticazione completata con il profilo creato con la crittografia MACsec. Se si fa clic sull'icona del modulo, è possibile visualizzare ulteriori informazioni.



Nel menu visualizzato per Secure Client, nella sezione Network Access Manager > Statistics, notare la Crittografia e la configurazione MACsec corrispondente.

I frame ricevuti e inviati aumentano man mano che la crittografia viene eseguita sul layer 2.

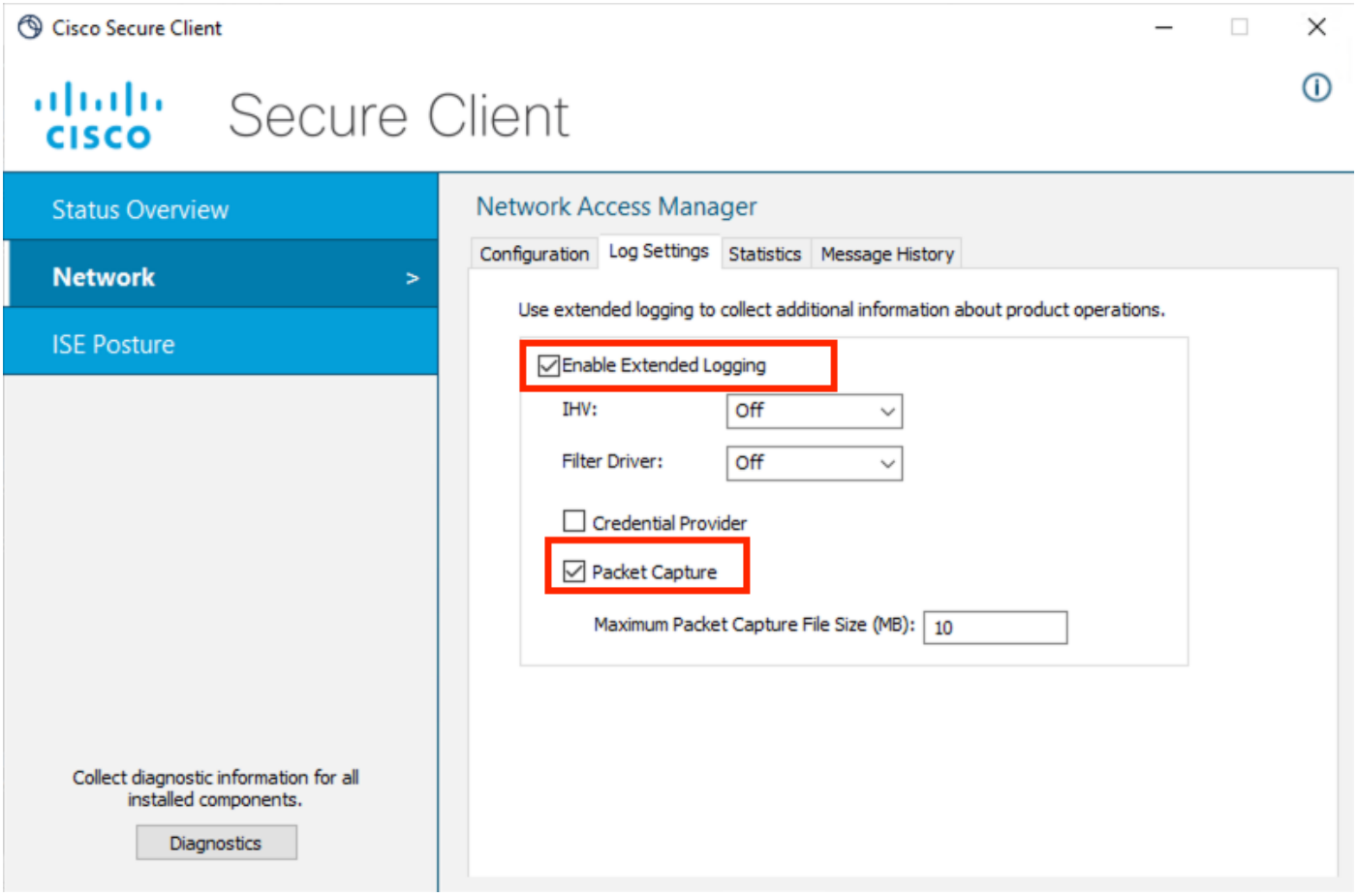


Risoluzione dei problemi

Nota: In questa sezione viene descritta la parte relativa alla risoluzione dei problemi relativi ai problemi MKA che possono emergere. In caso di problemi di autenticazione o autorizzazione, consultare la Guida all'installazione prescrittiva per l'accesso cablato sicuro ISE - Risoluzione dei problemi da esaminare. In questa guida si presume che le autenticazioni funzionino correttamente senza la crittografia MACsec.

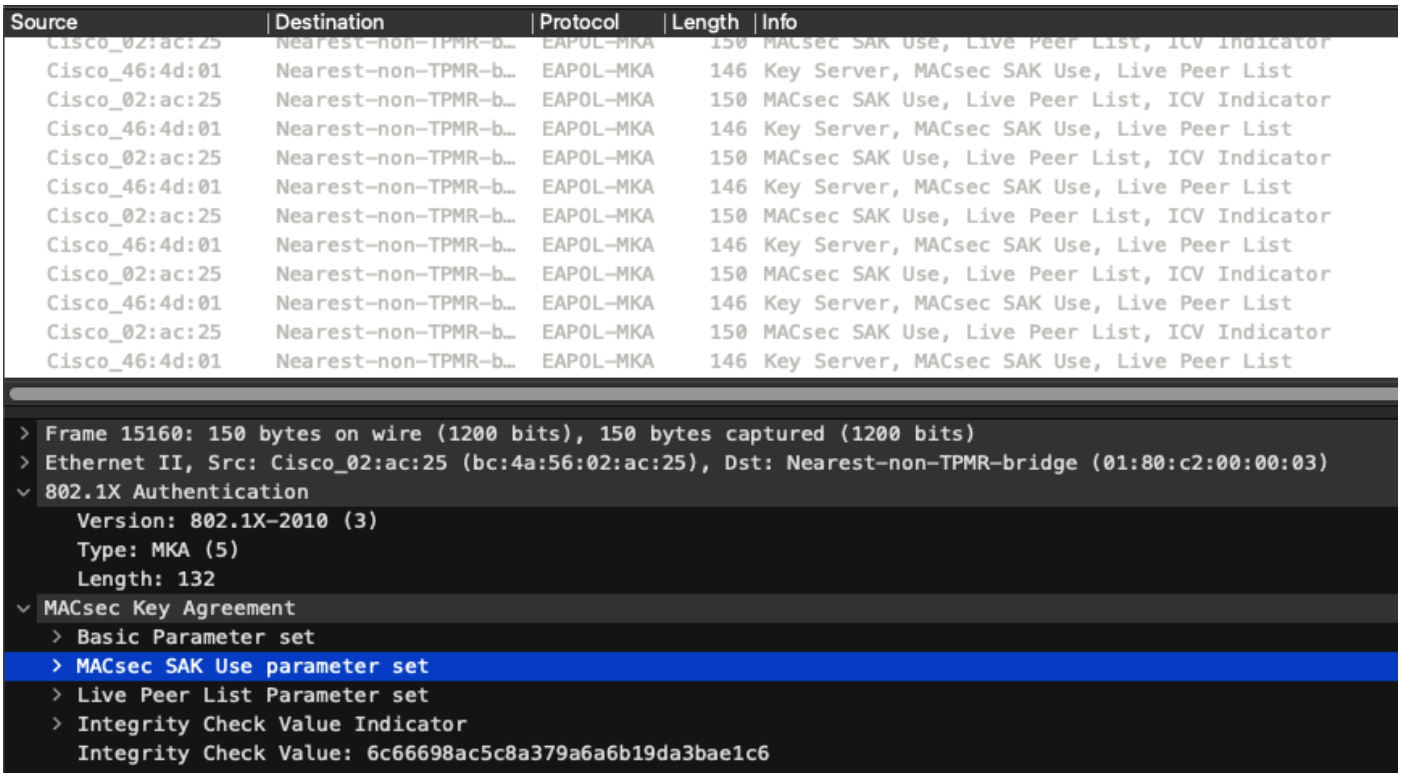
Cisco Secure Endpoint

- Abilitare il modulo DART nella suite Secure Endpoint.
- In questo menu, abilitare la funzione Registrazione estesa per raccogliere ulteriori dati sulla connessione MKA dell'utente. È inoltre possibile abilitare l'acquisizione di un pacchetto contenuto nel bundle DART.



- Raccogliere il bundle DART per procedere all'analisi di configuration.xml e Network Access Manager. Consultare la documentazione Eseguire DART per raccogliere dati per la risoluzione dei problemi

In questo esempio viene mostrato come i pacchetti vengono visti come informazioni crittografate tra l'host e lo switch:



Dal bundle DART, è possibile trovare informazioni utili per l'autenticazione 802.1X e la sessione MKA nel log denominato NetworkAccessManager.txt.

Queste informazioni vengono visualizzate se l'autenticazione con crittografia MKA ha esito positivo.

```
%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 auth
%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)
```

Risoluzione dei problemi di Cisco IOS

Questi comandi possono essere implementati nel Network Access Device (NAD) per rivedere la crittografia MKA tra la piattaforma e il richiedente.

Per ulteriori informazioni sui comandi, consultare la guida alla configurazione corrispondente della piattaforma utilizzata come 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
```

Si tratta dei debug di una connessione MKA riuscita a un host. È possibile utilizzare questo come riferimento:

```
%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
```

Risoluzione dei problemi di Identity Services Engine (ISE)

La risoluzione dei problemi relativi a questa funzionalità è limitata alla consegna dell'attributo cisco-av-pair linksec-policy=should-secure.

Accertarsi che il risultato dell'autorizzazione invii tali informazioni alla sessione Radius collegata alle porte switch a cui sono collegate le periferiche.

Per ulteriori analisi dell'autenticazione su ISE, consultare il documento sulla risoluzione dei problemi e l'abilitazione dei debug su ISE

Problemi comuni

Cifratura non corrispondente

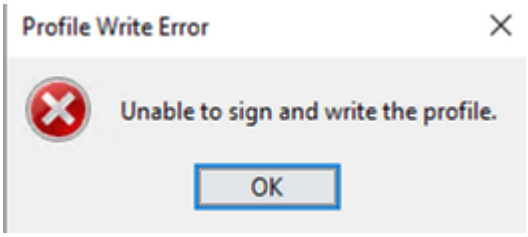
Questo log può essere visto nei debug MKA nel NAD.

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

La prima cosa da verificare in questo scenario è che le cifrature configurate nella policy MKA nello switch e nel profilo Secure Client corrispondono.

Nel caso della crittografia AES-GCM-256, questi requisiti devono essere soddisfatti (secondo la documentazione) (Cisco Secure Client (incluso AnyConnect) Administrator Guide, Release 5

Impossibile salvare il file configuration.xml.



Il problema relativo all'errore di scrittura del profilo viene risolto salvando il file configuration.xml (come descritto in precedenza) denominato Setup of MKA utilizzando l'Editor profili di Network Access Manager.

Poiché l'errore è correlato al fatto che il file configuration.xml in uso non può essere modificato, salvare il file in un'altra posizione per procedere con la sostituzione dei profili.

Informazioni correlate

- Configurazione della crittografia MACsec
- Configurazione dello switch MACsec su host con Cat9k e ISE

Informazioni su questa traduzione

Cisco ha tradotto questo documento utilizzando una combinazione di tecnologie automatiche e umane per offrire ai nostri utenti in tutto il mondo contenuti di supporto nella propria lingua. Si noti che anche la migliore traduzione automatica non sarà mai accurata come quella fornita da un traduttore professionista. Cisco Systems, Inc. non si assume alcuna responsabilità per l’accuratezza di queste traduzioni e consiglia di consultare sempre il documento originale in inglese (disponibile al link fornito).