

Configure 802.1X Authentication with PEAP, ISE 2.1 and WLC 8.3

Contents

[Introduction](#)
[Prerequisites](#)
[Requirements](#)
[Components Used](#)
[Background Information](#)
[Configure](#)
[Network Diagram](#)
[Configuration](#)
[Declare RADIUS Server on WLC](#)
[Create SSID](#)
[Declare WLC on ISE](#)
[Create New User on ISE](#)
[Create Authentication Rule](#)
[Create Authorization Profile](#)
[Create Authorization Rule](#)
[Configuration of End Device](#)
[End Device Configuration - Install ISE Self-Signed Certificate](#)
[End Device Configuration - Create the WLAN Profile](#)
[Verify](#)
[Authentication Process on WLC](#)
[Authentication Process on ISE](#)
[Troubleshoot](#)

Introduction

This document describes how to set up a Wireless Local Area Network (WLAN) with 802.1x security and Virtual Local Area Network (VLAN) override.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- 802.1x
- Protected Extensible Authentication Protocol (PEAP)
- Certification Authority (CA)
- Certificates

Components Used

The information in this document is based on these software and hardware versions:

- WLC v8.3.102.0

- Identity Service Engine (ISE) v2.1
- Windows 10 Laptop

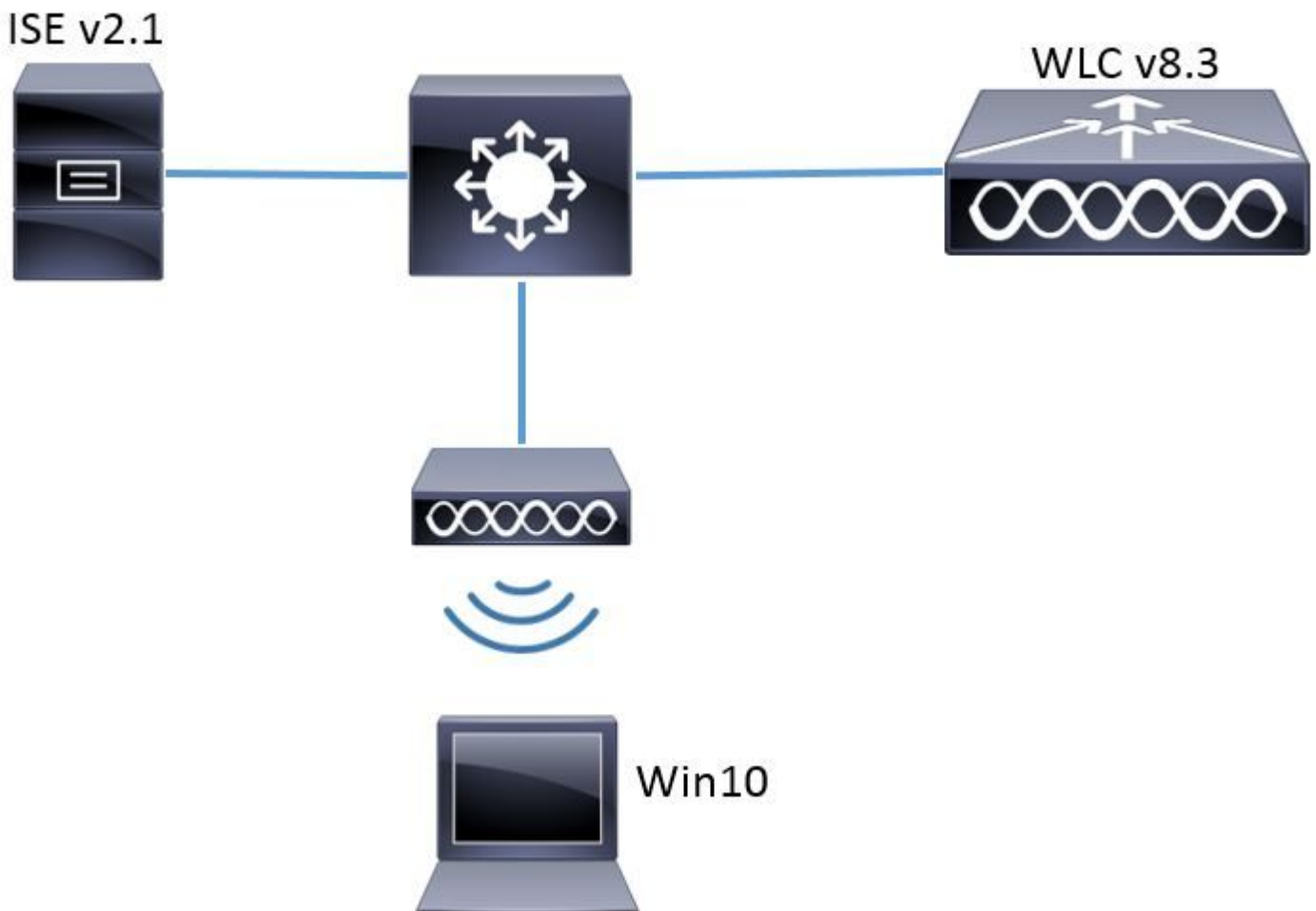
The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Background Information

When you set up a WLAN with 802.1x security and VLAN, you can override with Protected Extensible Authentication Protocol as Extensible Authentication Protocol (EAP).

Configure

Network Diagram



Configuration

The general steps are:

1. Declare RADIUS Server on WLC and vice versa to allow communication with each other.
2. Create the Service Set Identifier (SSID) in the WLC.
3. Create the authentication rule on ISE.
4. Create the authorization profile on ISE.

5. Create the authorization rule on ISE.
6. Configure the endpoint.

Declare RADIUS Server on WLC

In order to allow communication between RADIUS server and WLC, you need to register RADIUS server on WLC and vice versa.

GUI:

Step 1. Open the GUI of the WLC and navigate to **SECURITY > RADIUS > Authentication > New** as shown in the image.



Step 2. Enter the RADIUS server information as shown in the image.

RADIUS Authentication Servers > New

Server Index (Priority)	2	
Server IP Address(Ipv4/Ipv6)	a.b.c.d	
Shared Secret Format	ASCII	
Shared Secret	••••••••	
Confirm Shared Secret	••••••••	
Key Wrap	<input type="checkbox"/>	(Designed for FIPS customers and requires a key wrap compliant RADIUS server)
Port Number	1812	
Server Status	Enabled	
Support for CoA	Disabled	
Server Timeout	10	seconds
Network User	<input checked="" type="checkbox"/>	Enable
Management	<input checked="" type="checkbox"/>	Enable
Management Retransmit Timeout	2	seconds
IPSec	<input type="checkbox"/>	Enable

CLI:

- ```
> config radius auth add <index> <a.b.c.d> 1812 ascii <shared-key>
> config radius auth disable <index>
> config radius auth retransmit-timeout <index> <timeout-seconds>
```

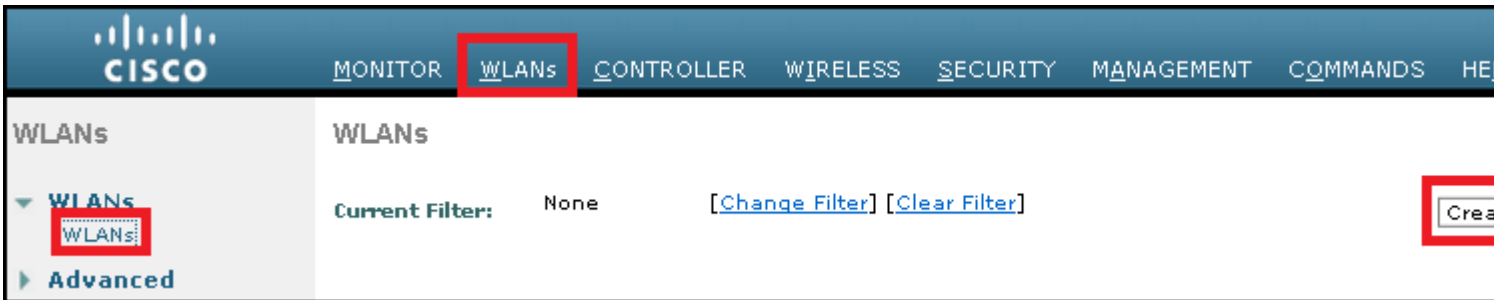
```
> config radius auth enable <index>
```

<a.b.c.d> corresponds to the RADIUS server.

## Create SSID

GUI:

Step 1. Open the GUI of the WLC and navigate to **WLANS > Create New > Go** as shown in the image.



Step 2. Choose a name for the SSID and profile, then click **Apply** as shown in the image.



CLI:

```
> config wlan create <id> <profile-name> <ssid-name>
```

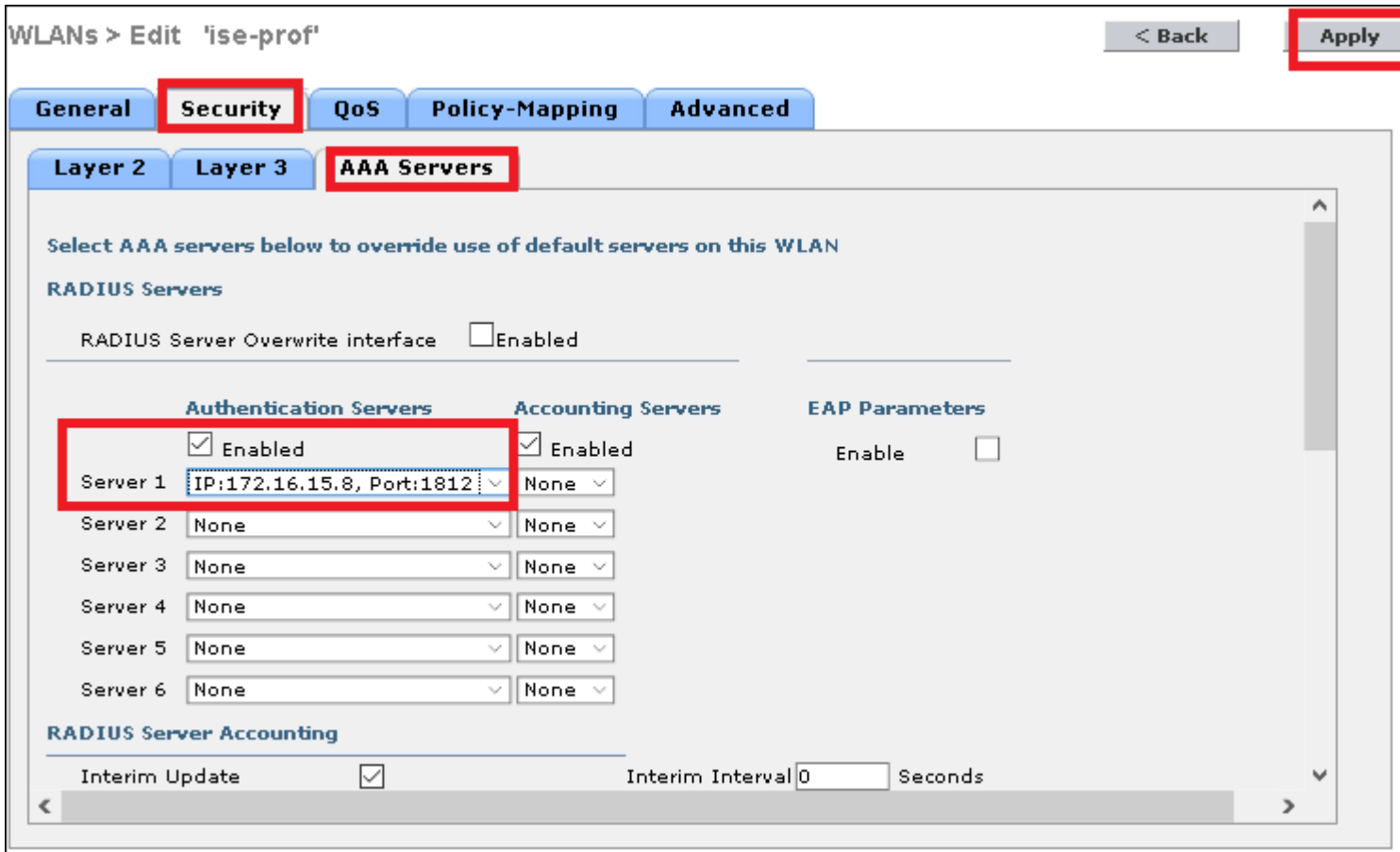
Step 3. Assign the RADIUS server to the WLAN.

CLI:

```
> config wlan radius_server auth add <wlan-id> <radius-index>
```

GUI:

Navigate to **Security > AAA Servers** and choose the desired RADIUS server, then hit **Apply** as shown in the image.



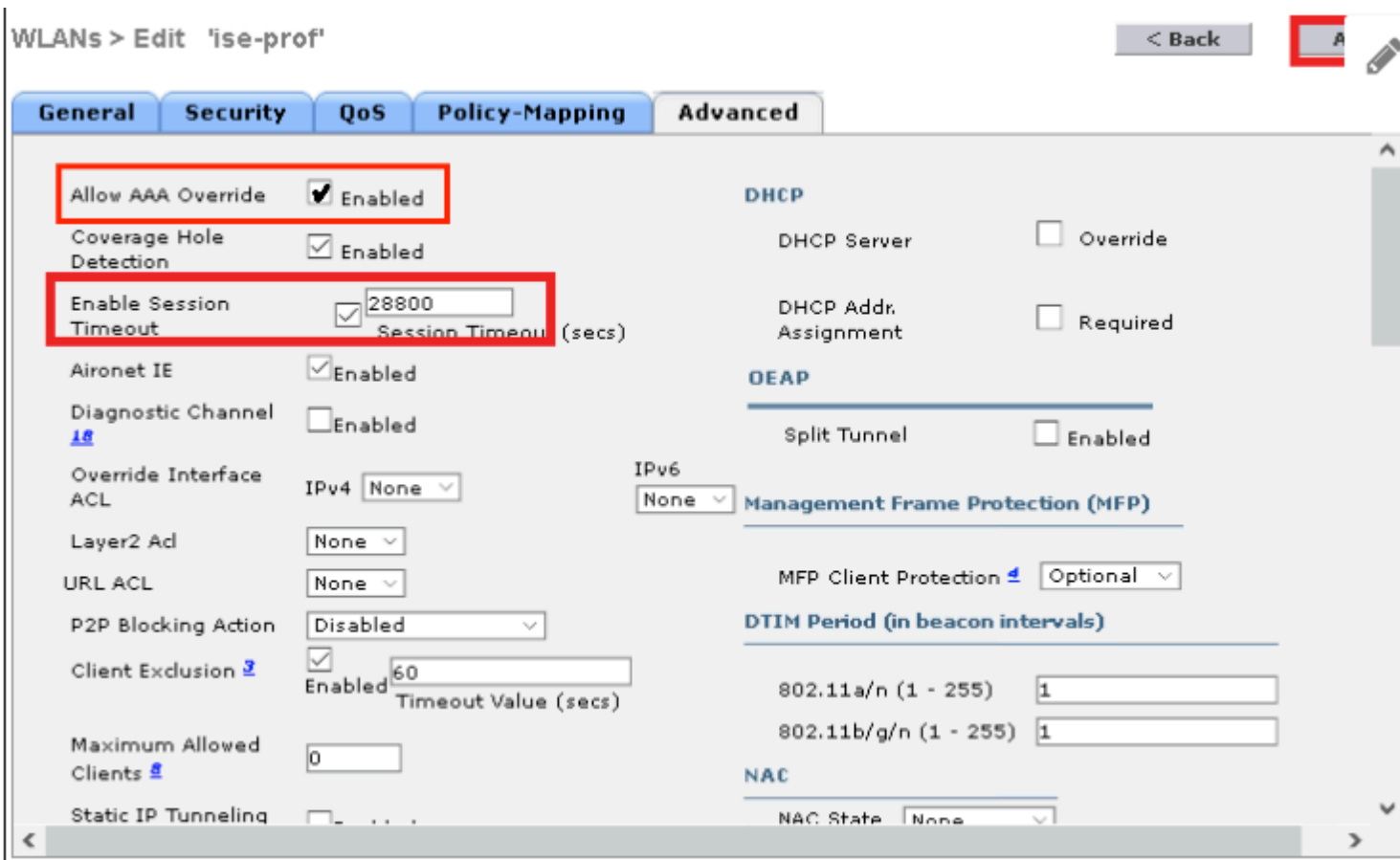
Step 4. Enable **Allow AAA Override** and optionally increase the session timeout

CLI:

```
> config wlan aaa-override enable <wlan-id>
> config wlan session-timeout <wlan-id> <session-timeout-seconds>
```

GUI:

Navigate to **WLANs > WLAN ID > Advanced** and enable **Allow AAA Override**. Optionally specify the Session Timeout as shown in the image.



Step 5. Enable the WLAN.

CLI:

```
> config wlan enable <wlan-id>
```

GUI:

Navigate to **WLANs > WLAN ID > General** and enable the SSID as shown in the image.

WLANs > Edit 'ise-prof' [< Back](#) [Apply](#)

**General** Security QoS Policy-Mapping Advanced

Profile Name: ise-prof

Type: WLAN

SSID: ise-ssid

Status:  Enabled

Security Policies: [WPA2][Auth(802.1X)]  
(Modifications done under security tab will appear after applying the changes.)

Radio Policy: All

Interface/Interface Group(G): management

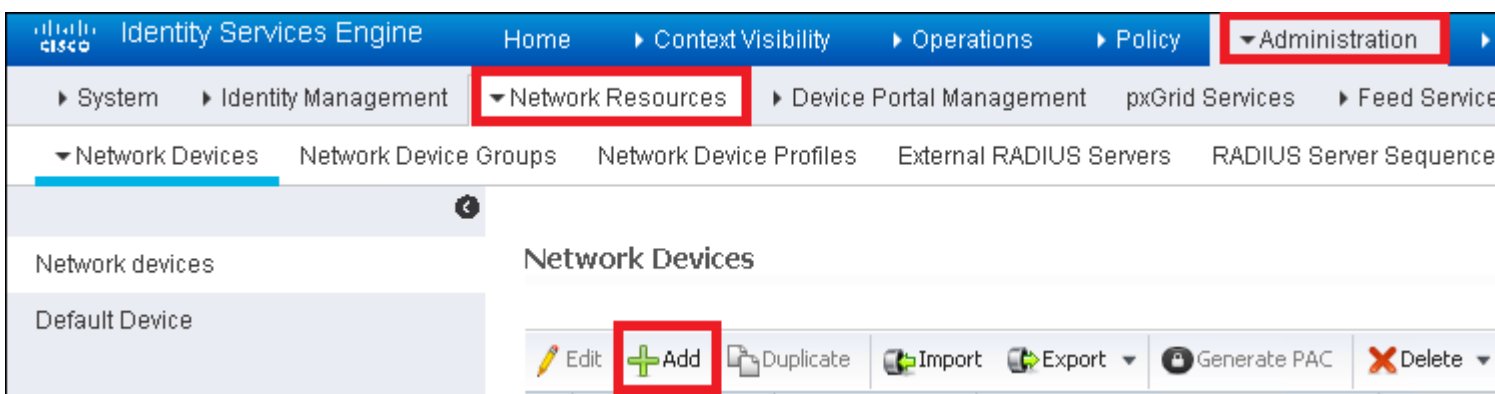
Multicast Vlan Feature:  Enabled

Broadcast SSID:  Enabled

NAS-ID: none

## Declare WLC on ISE

Step 1. Open ISE console and navigate to **Administration > Network Resources > Network Devices > Add** as shown in the image.



Step 2. Enter the values.

Optionally, it can be a specified Model name, software version, description and assign Network Device groups based on device types, location or WLCs.

a.b.c.d correspond to the WLC interface that sends the authentication requested. By default, it is the management interface as shown in the image.

Network Devices List > **New Network Device**

### Network Devices

\* Name

Description

\* IP Address:  /

\* Device Profile

Model Name

Software Version

\* Network Device Group

Device Type

Location

WLCs

**RADIUS Authentication Settings**

Enable Authentication Settings

Protocol **RADIUS**

\* Shared Secret

Enable KeyWrap

\* Key Encryption Key

\* Message Authenticator Code Key

Key Input Format  ASCII  HEXADECIMAL

CoA Port

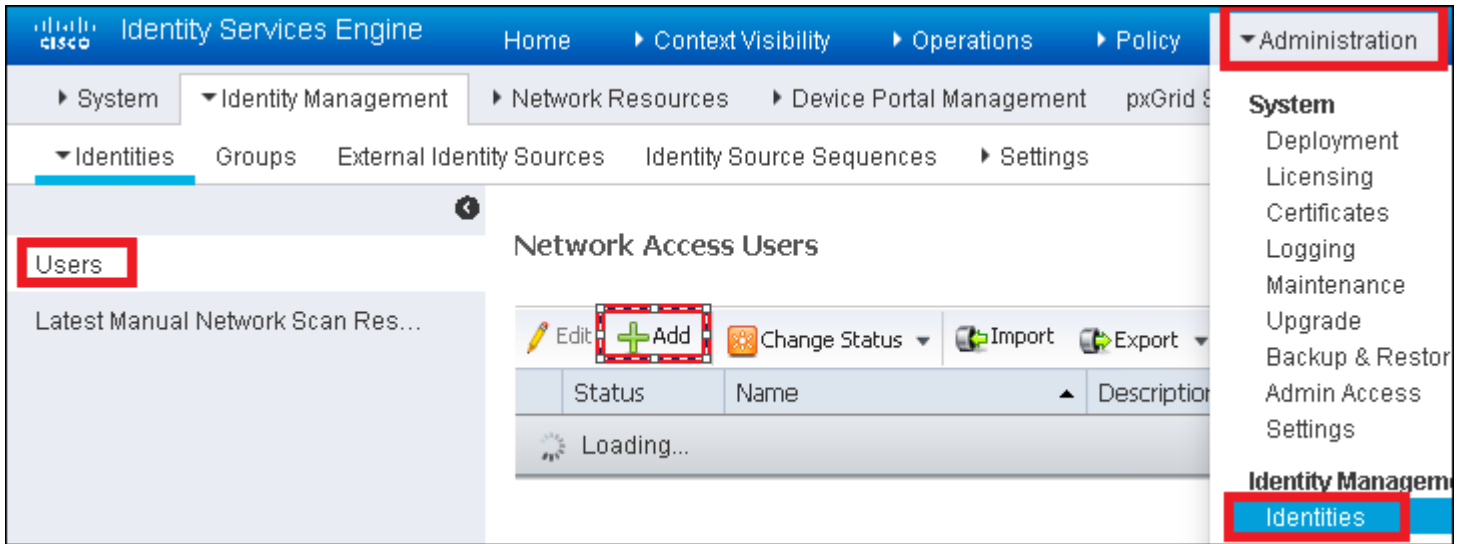
For more information about Network Device Groups:

[ISE - Network Device Groups](#)

### Create New User on ISE

Step 1. Navigate to **Administration > Identity Management > Identities > Users > Add** as shown in the image.





Step 2. Enter the information.

In this example, this user belongs to a group called ALL\_ACCOUNTS, but it can be adjusted as needed, as shown in the image.

Network Access Users List > **New Network Access User**

**Network Access User**

\* Name

Status  Enabled

Email

**Passwords**

Password Type:

Password  Re-Enter Password

\* Login Password

Enable Password

**User Information**

First Name

Last Name

**Account Options**

Description

Change password on next login

**Account Disable Policy**

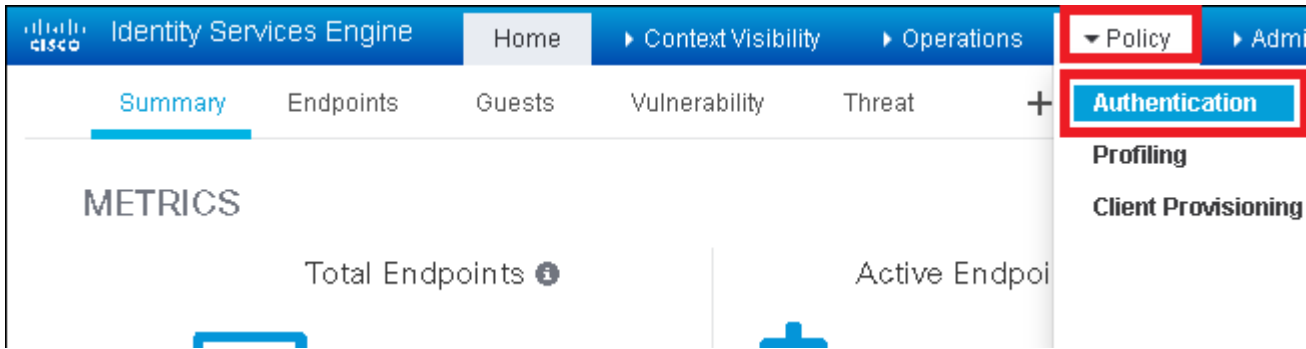
Disable account if date exceeds

**User Groups**

## Create Authentication Rule

Authentication rules are used to verify if the credentials of the users are correct (verify if the user really is who they say they are), and limit the authentication methods that are allowed to be used by it.

Step 1. Navigate to **Policy > Authentication** as shown in the image.

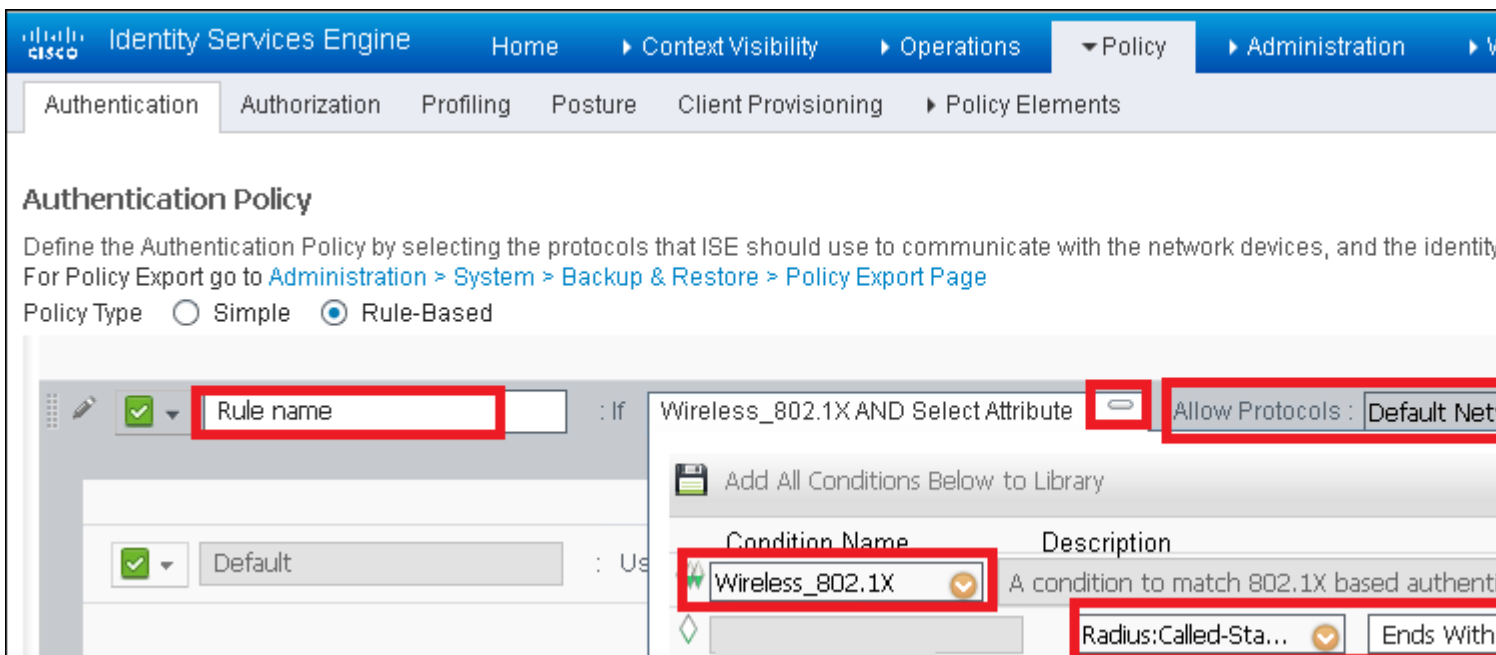


Step 2. Insert a new authentication rule as shown in the image.

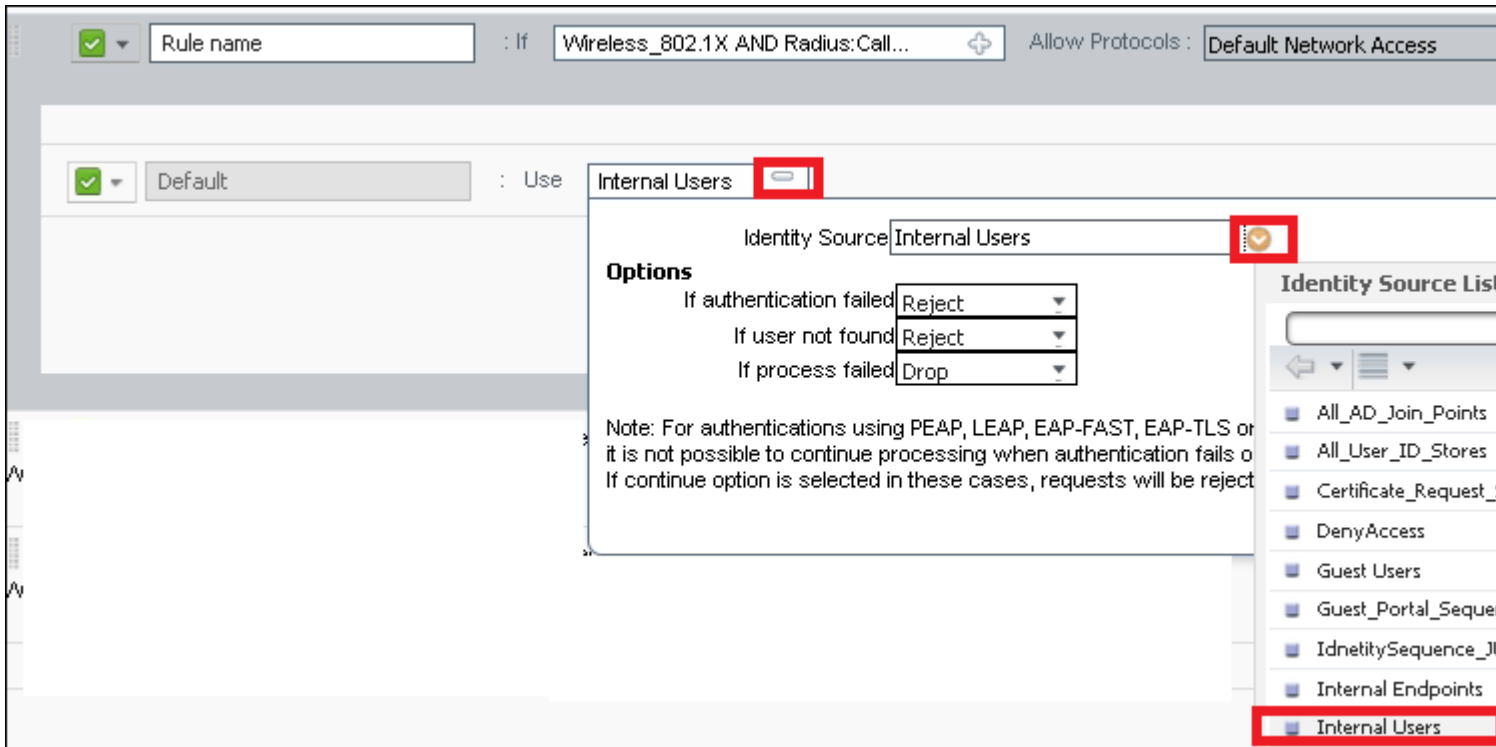


Step 3. Enter the values.

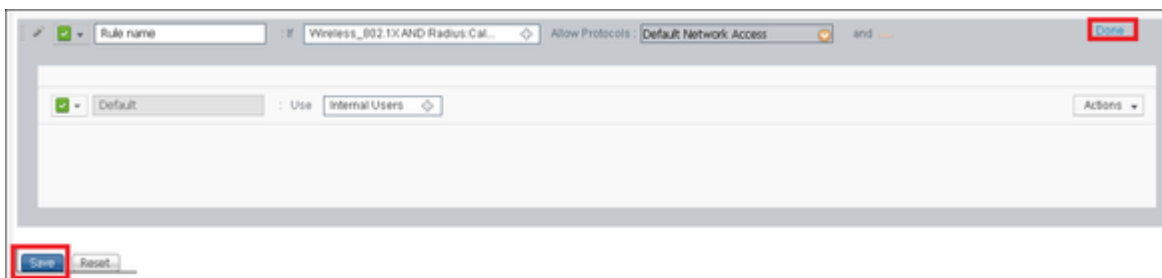
This authentication rule allows all the protocols listed under the Default Network Access list. This applies to the authentication request for Wireless 802.1x clients, and with Called-Station-ID, and ends with ise-ssid as shown in the image.



Also, choose the Identity source for the clients that match this authentication rule. This example uses Internal users identity source list as shown in the image.



Once finished, click **Done** and **Save** as shown in the image.



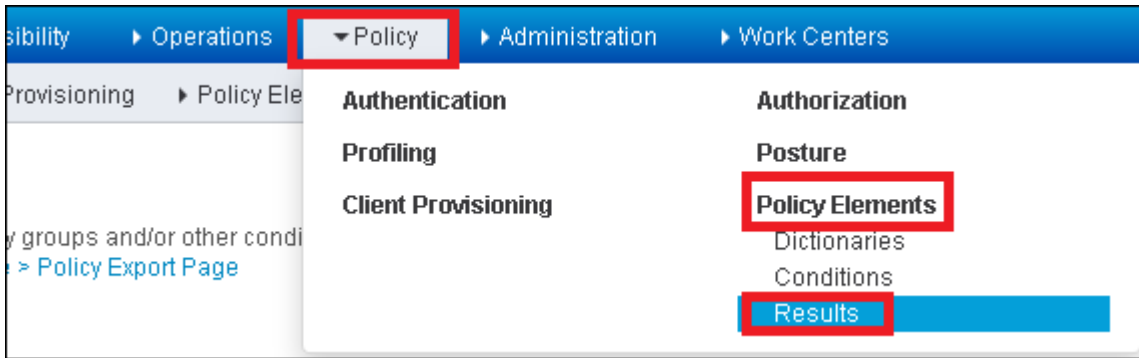
For more information about Identity sources consult this link:

[Create a User Identity Group](#)

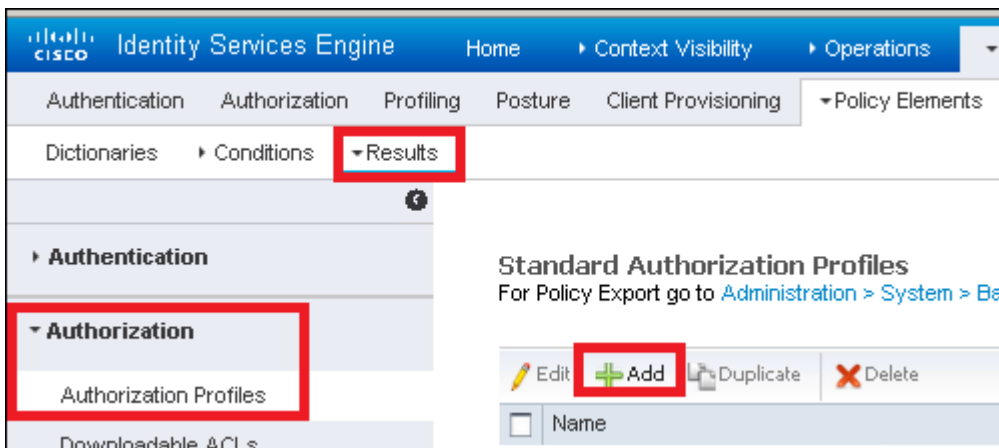
## Create Authorization Profile

The authorization profile determines if you have access or not to the network. Push Access Control Lists (ACLs), VLAN override or any other parameter. The authorization profile shown in this example sends an access accept to you, and assigns VLAN 2404.

Step 1. Navigate to **Policy > Policy Elements > Results** as shown in the image.



Step 2. Add a new Authorization Profile. Navigate to **Authorization > Authorization Profiles > Add** as shown in the image.



Step 3. Enter the values as shown in the image.

Authorization Profiles > **New Authorization Profile**

### Authorization Profile

\* Name

Description

\* Access Type

Network Device Profile

Service Template

Track Movement

Passive Identity Tracking

#### Common Tasks

ACL (Filter-ID)

VLAN Tag ID   ID/Name

Voice Domain Permission

Web Redirection (CWA, MDM, NSP, CDD)

#### Advanced Attributes Settings

Select an item  =

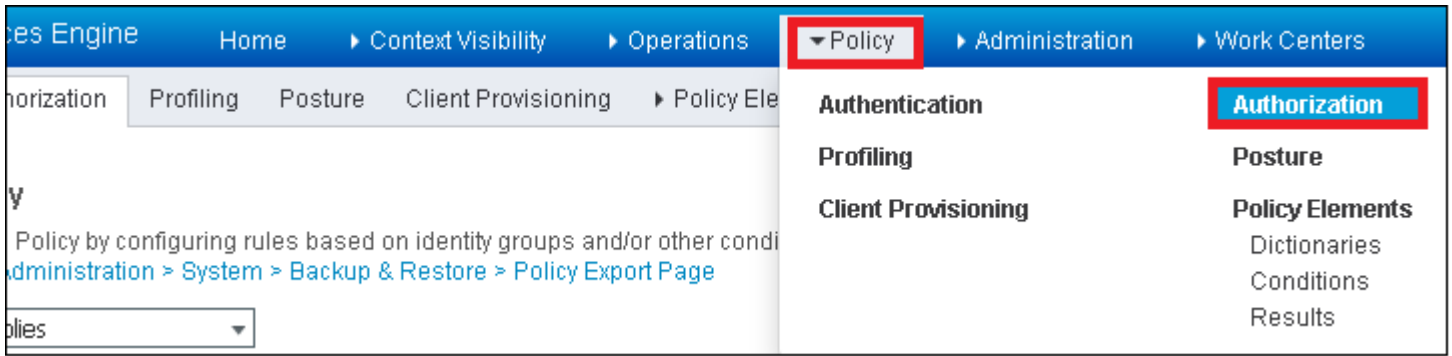
#### Attributes Details

Access Type = ACCESS\_ACCEPT  
Tunnel-Private-Group-ID = NaN:2404  
Tunnel-Type = NaN:13  
Tunnel-Medium-Type = NaN:6

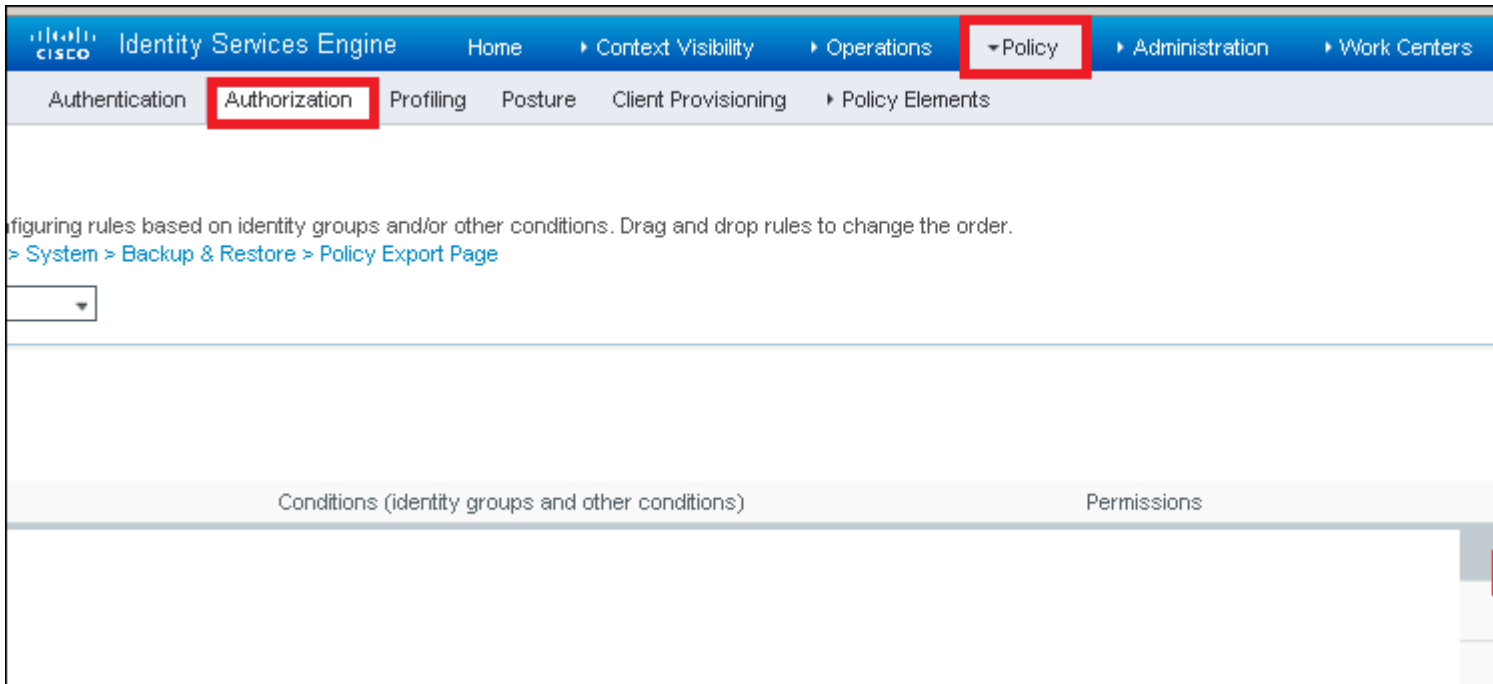
## Create Authorization Rule

The authorization rule is the one in charge to determine which permissions (which authorization profile) result is applied to you.

Step 1. Navigate to **Policy > Authorization** as shown in the image.

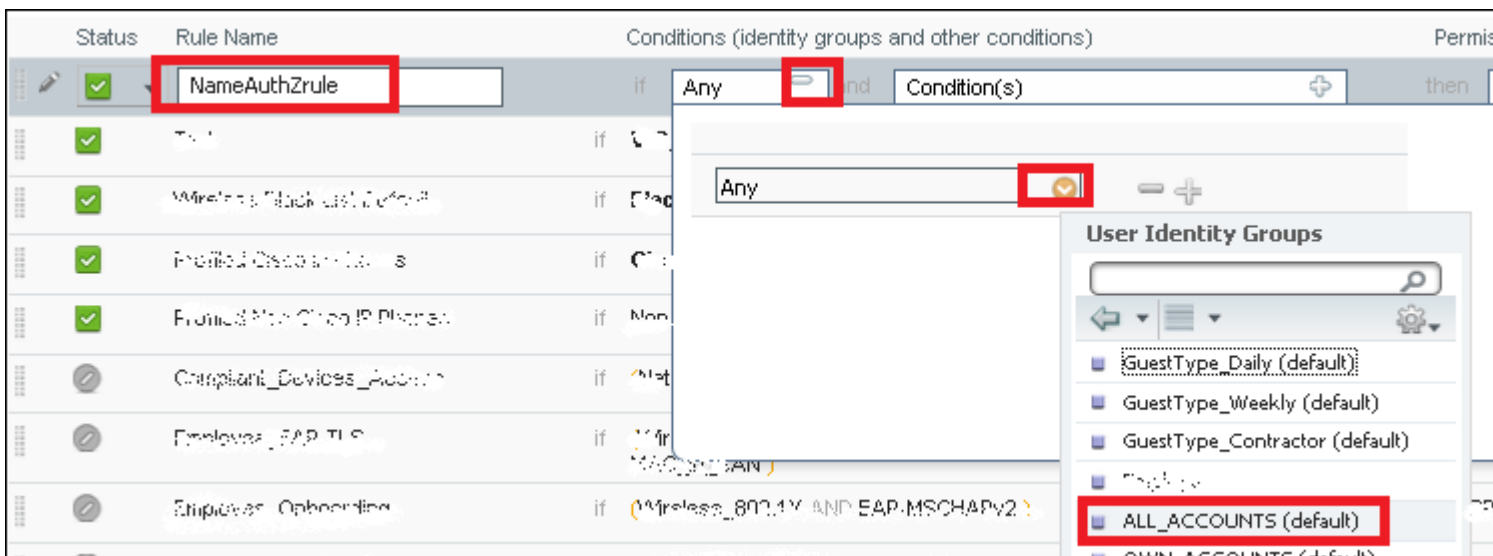


Step 2. Insert a new rule as shown in the image.

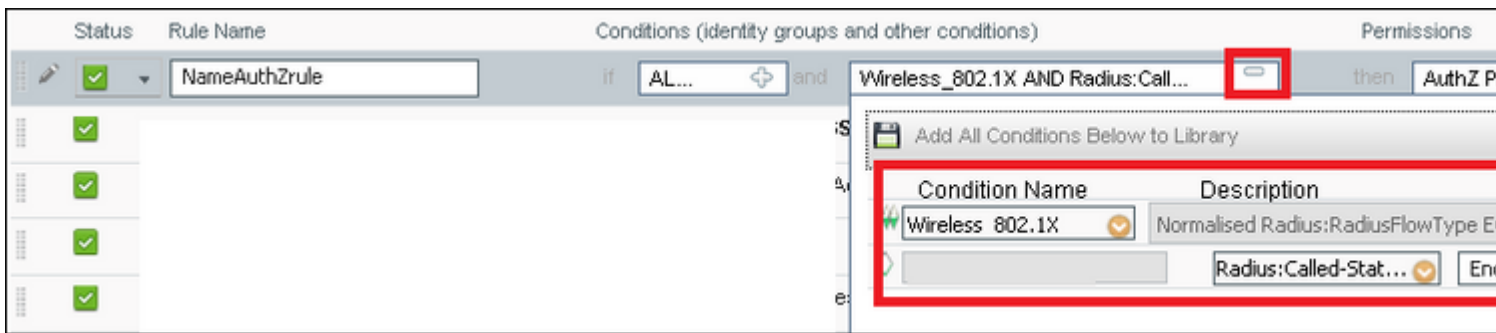


Step 3. Enter the values.

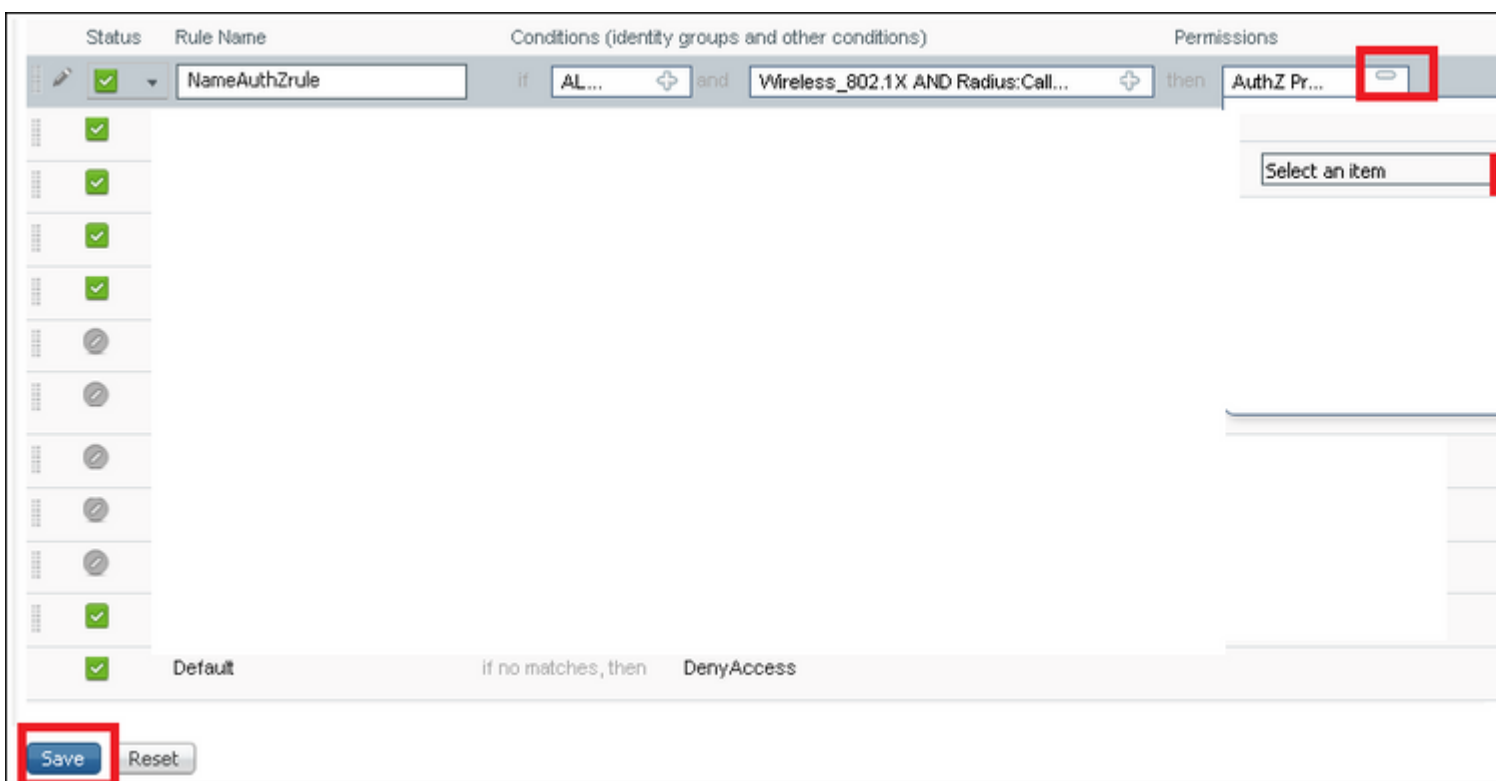
First, select a name for the rule, and the identity group where the user is stored (ALL\_ACCOUNTS) as shown in the image.



After that, select other conditions that cause the authorization process to fall into this rule. In this example, the authorization process hits this rule if it uses 802.1x Wireless and its called station ID ends with ise-ssid as shown in the image.



Finally, select the Authorization profile that is assigned to you that hits that rule. Click **Done** and **Save** as shown in the image.



## Configuration of End Device

Configure a laptop Windows 10 machine to connect to an SSID with 802.1x Authentication and PEAP/MS-CHAPv2 (Microsoft version of the Challenge-Handshake Authentication Protocol) Version 2.

In this configuration example, ISE uses its self-signed certificate to perform the authentication.

In order to create the WLAN profile on the windows machine, there are two options:

1. Install the self-signed certificate on the machine to validate, and trust ISE server in order to complete the authentication.
2. Bypass the validation of the RADIUS server, and trust any RADIUS server used to perform the authentication (not recommended, as it can become a security issue).

The configuration for these options are explained on End device configuration - Create the WLAN Profile -



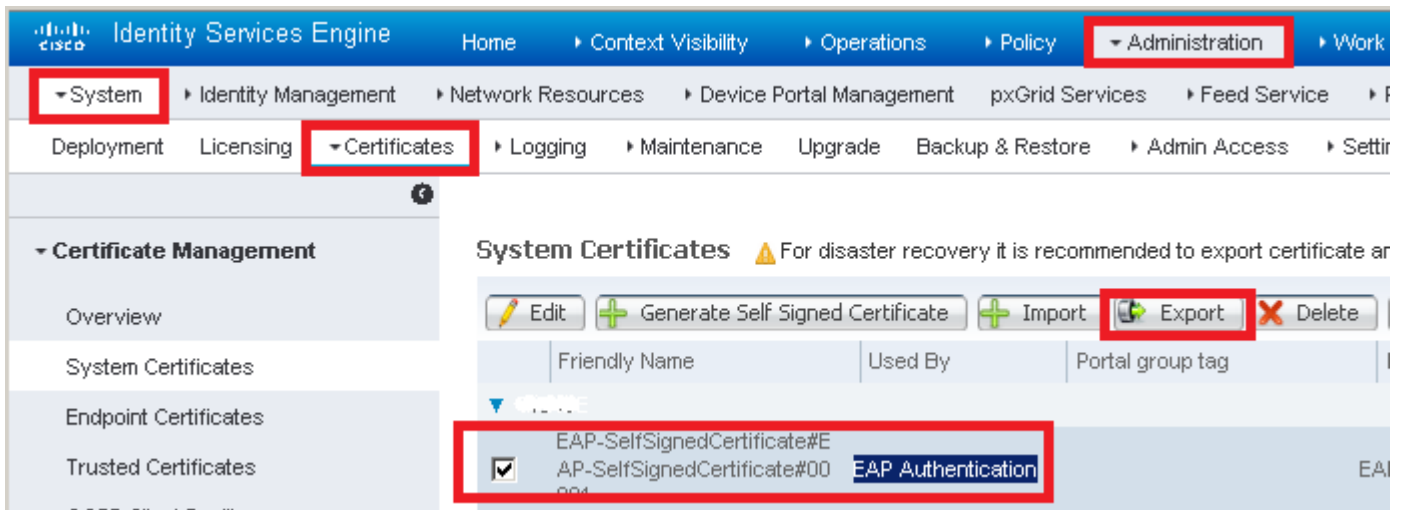
Step 7.

## End Device Configuration - Install ISE Self-Signed Certificate

Step 1. Export self-signed certificate.

Log in to ISE and navigate to **Administration > System > Certificates > System Certificates**.

Then choose the certificate used for **EAP Authentication** and click **Export** as shown in the image.

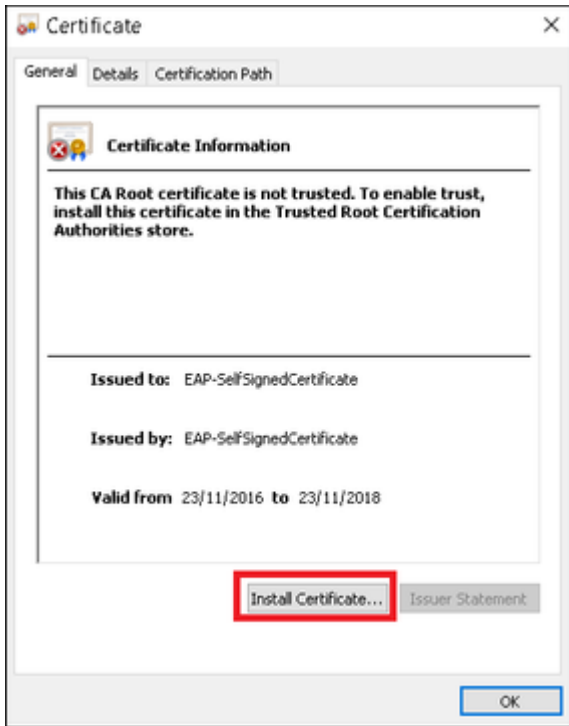


Save the certificate in the needed location. That certificate must be installed on the windows machine as shown in the image.

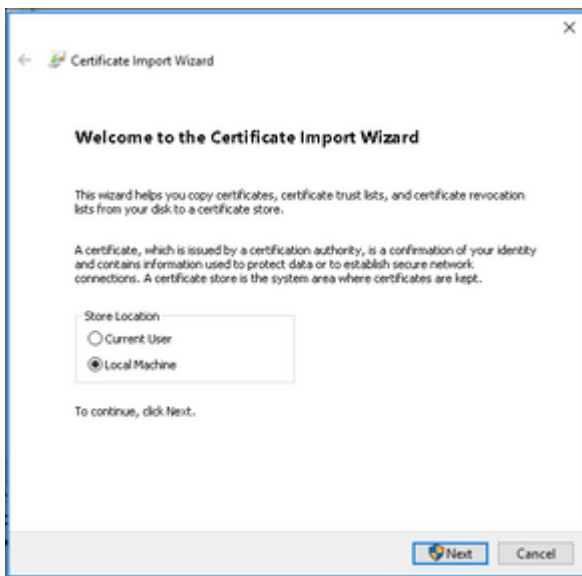


Step 2. Install the certificate in the windows machine.

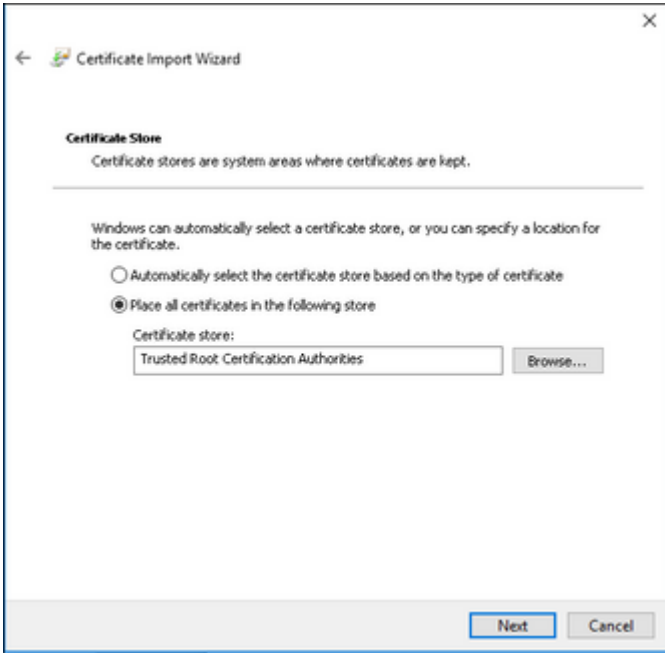
Copy the certificate exported from ISE into the windows machine, change the extension of the file from .pem to .crt, and after that double click in order to install it as shown in the image.



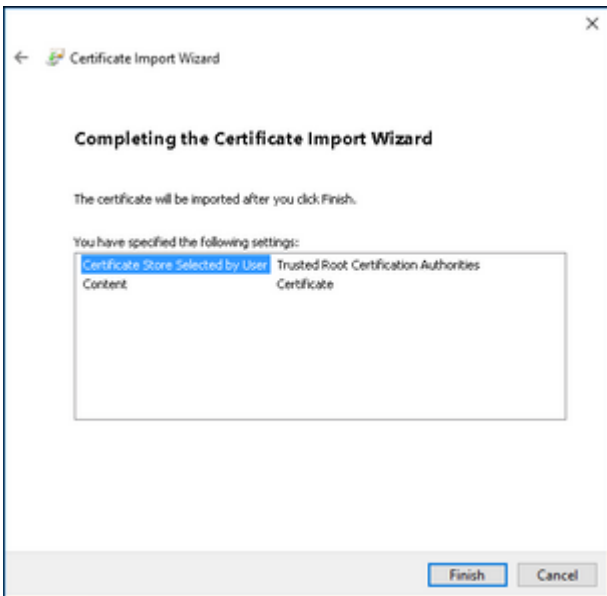
Step 3. Select install it in **Local Machine** and click **Next** as shown in the image.



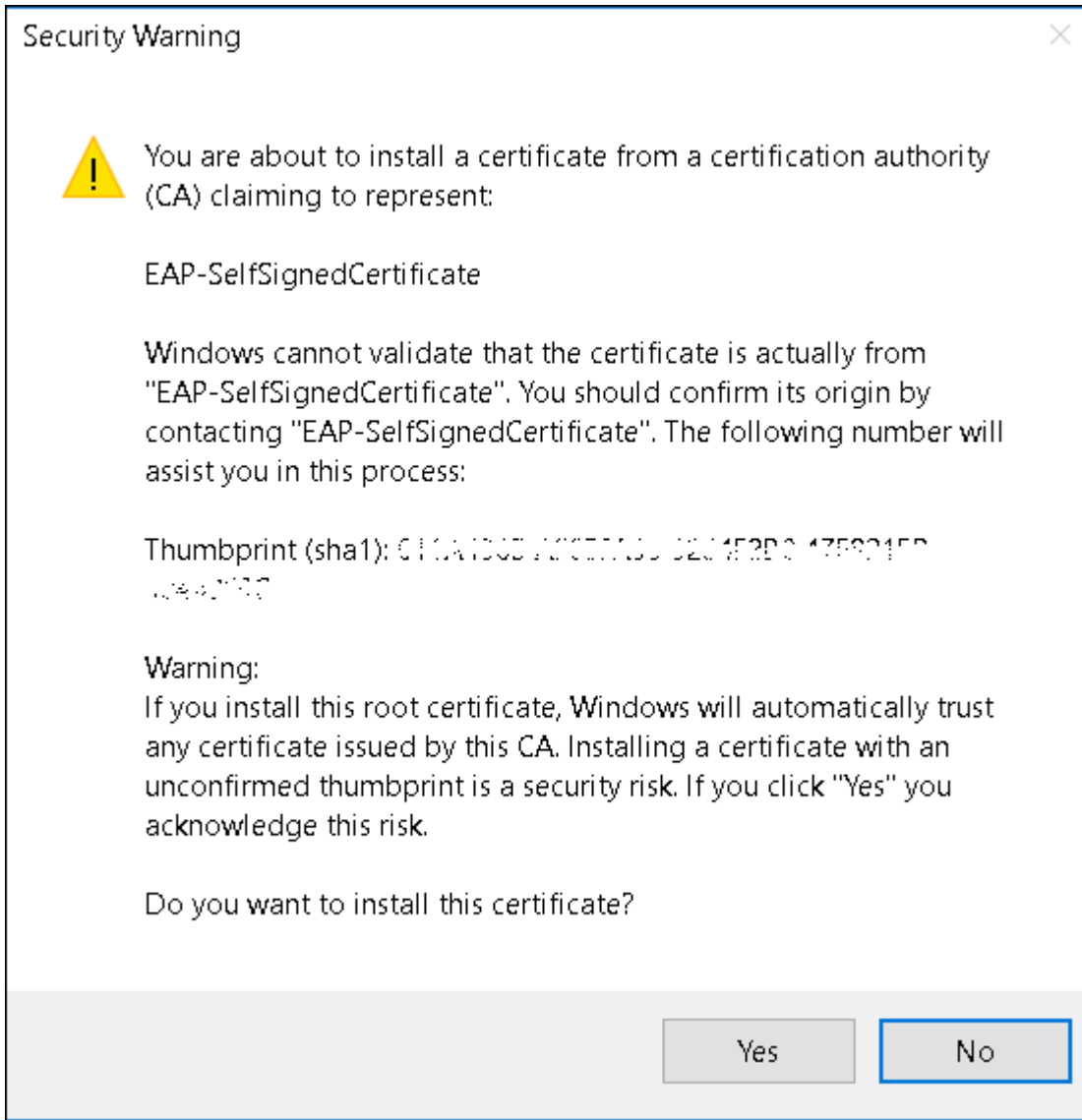
Step 4. Select **Place all certificates in this store**, then browse and select **Trusted Root Certification Authorities**. After that, click **Next** as shown in the image.



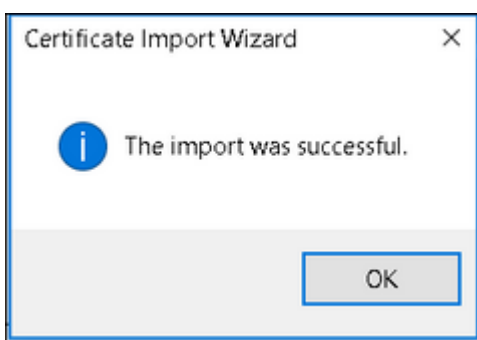
Step 5. Then, click **Finish** as shown in the image.



Step 6. Confirm the installation of the certificate. Click **Yes** as shown in the image.

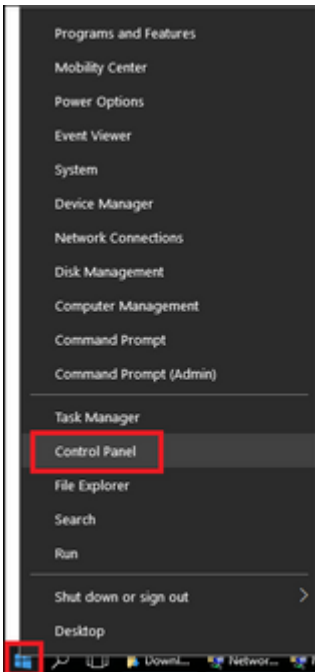


Step 7. Finally, click **OK** as shown in the image.

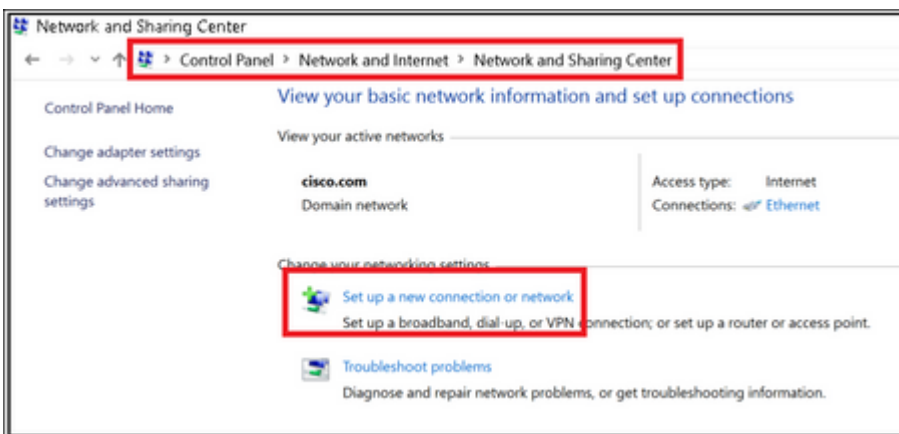


## End Device Configuration - Create the WLAN Profile

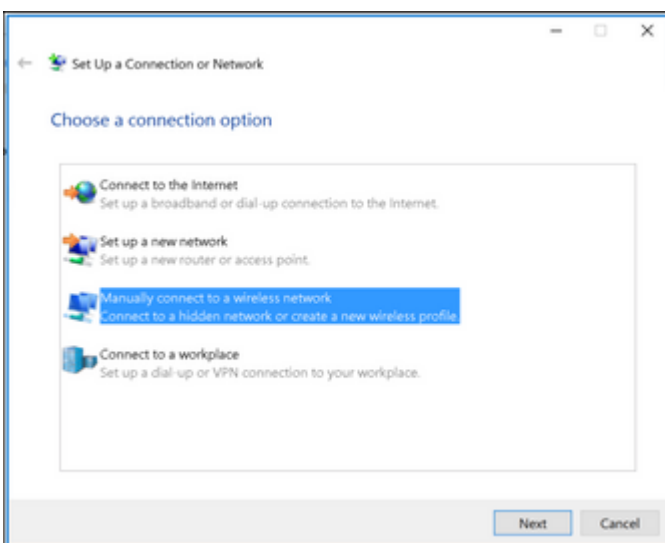
Step 1. Right click on **Start** icon and select **Control Panel** as shown in the image.



Step 2. Navigate to **Network and Internet**, and after that navigate to **Network and Sharing Center**, and click **Set up a new connection or network** as shown in the image.

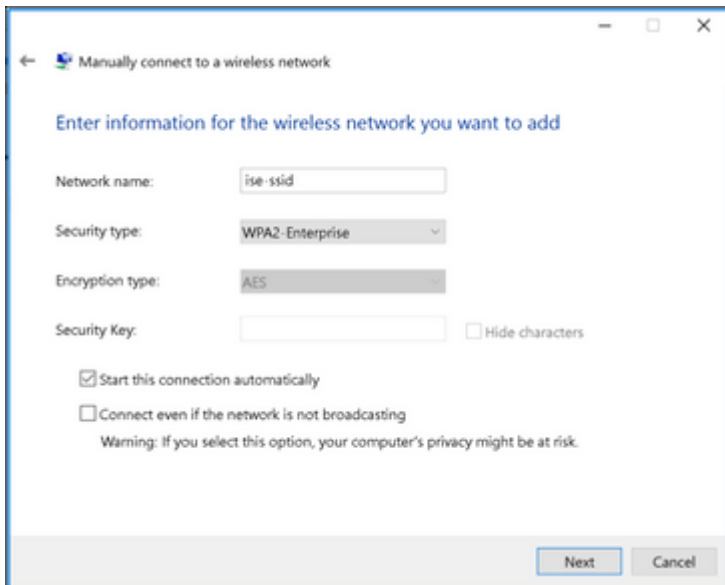


Step 3. Select **Manually connect to a wireless network**, and click **Next** as shown in the image.

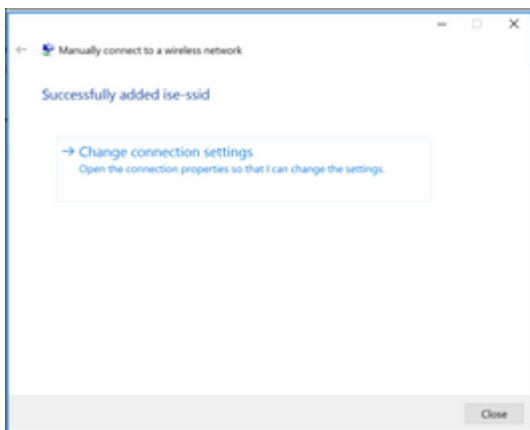


Step 4. Enter the information with the name of the SSID and security type WPA2-Enterprise and click

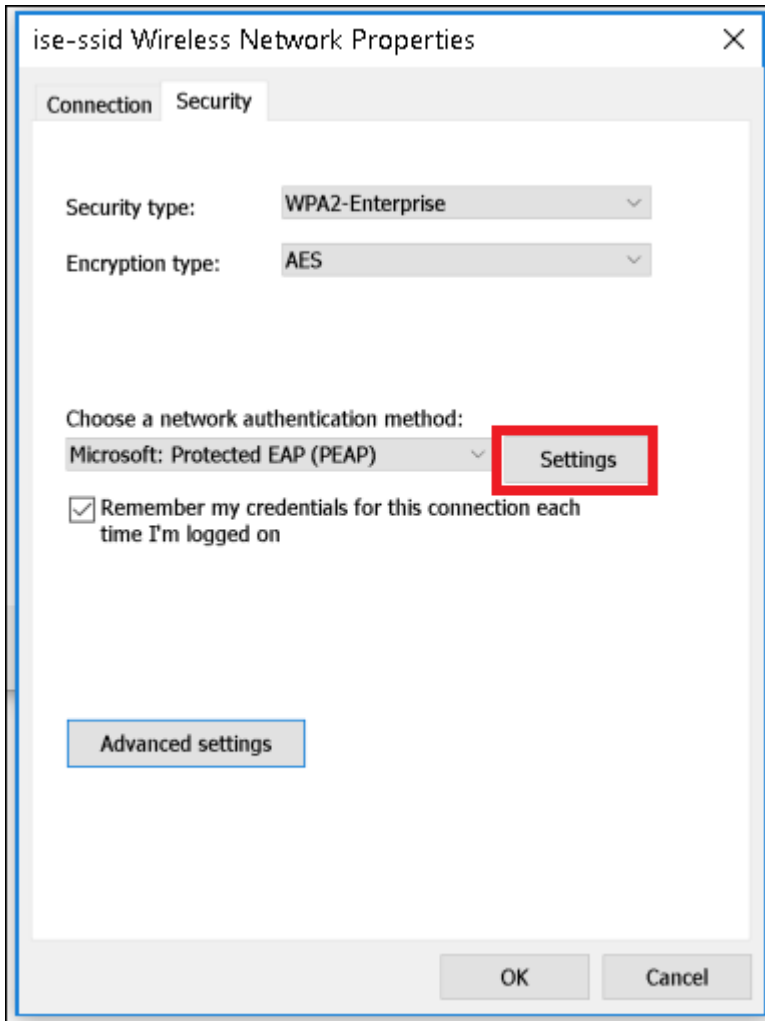
**Next** as shown in the image.



Step 5. Select **Change connection settings** in order to customize the configuration of the WLAN profile as shown in the image.



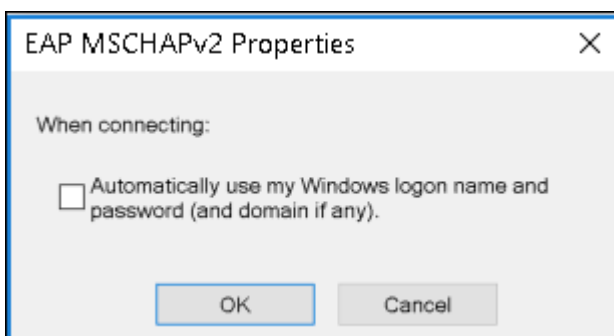
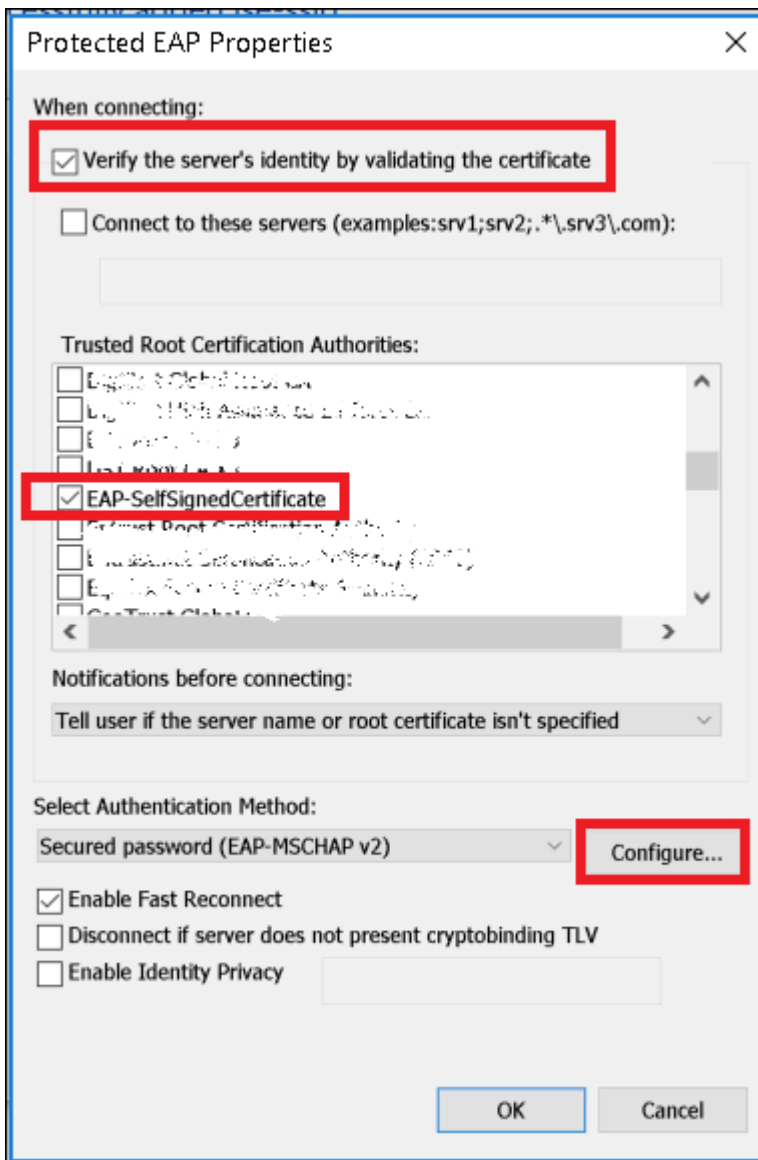
Step 6. Navigate to **Security** tab and click **Settings** as shown in the image.



Step 7. Select if RADIUS server is validated or not.

If yes, enable **Verify the server identity by validating the certificate** and from **Trusted Root Certification Authorities:** list select the self-signed certificate of ISE.

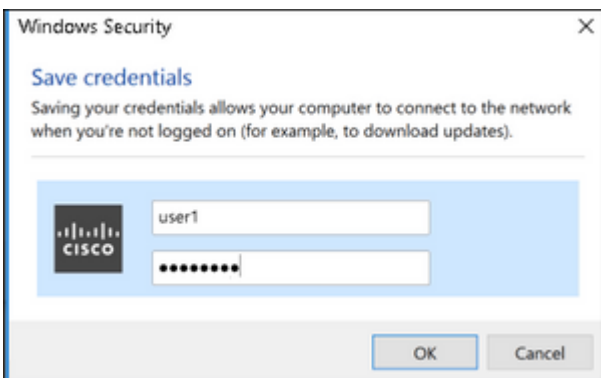
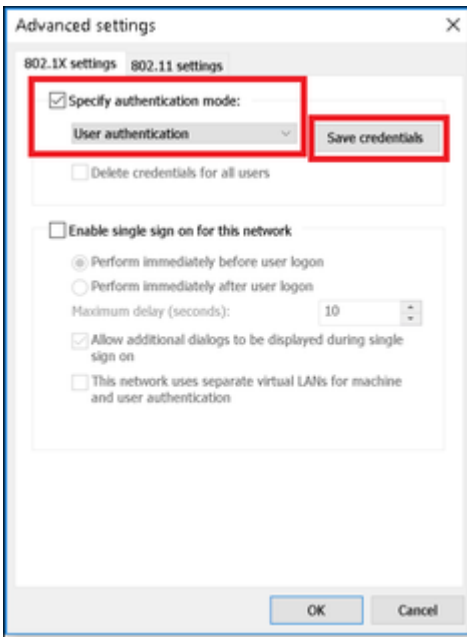
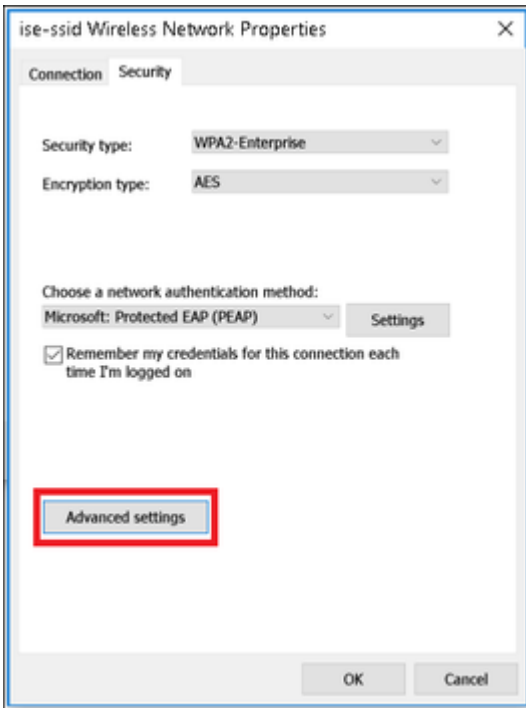
After that select **Configure** and disable **Automatically use my Windows logon name and password...**, then click **OK** as shown in the images.



Step 8. Configure the user credentials.

Once back to **Security** tab, select **Advanced settings**, specify authentication mode as User authentication, and **save** the credentials that were configured on ISE in order to authenticate the user as shown in the images.





**Verify**

Use this section in order to confirm that your configuration works properly.

The authentication flow can be verified from WLC or from ISE perspective.

## Authentication Process on WLC

Run the next commands in order to monitor the authentication process for a specific user:

```
> debug client <mac-add-client>
> debug dot1x event enable
> debug dot1x aaa enable
```

Example of a successful authentication (some output has been omitted):

```
<#root>
```

```
*apfMsConnTask_1: Nov 24 04:30:44.317:
```

```
e4:b3:18:7c:30:58 Processing assoc-req station:e4:b3:18:7c:30:58 AP:00:c8:8b:26:2c:d0-00
```

```
thread:1a5cc288
```

```
*apfMsConnTask_1: Nov 24 04:30:44.317: e4:b3:18:7c:30:58 Reassociation received from mobile on BSSID 00:c8:8b:26:2c:d0-00
```

```
*apfMsConnTask_1: Nov 24 04:30:44.318: e4:b3:18:7c:30:58 Applying Interface(management) policy on Mobile
```

```
*apfMsConnTask_1: Nov 24 04:30:44.318: e4:b3:18:7c:30:58 Applying site-specific Local Bridging override
```

```
*apfMsConnTask_1: Nov 24 04:30:44.318: e4:b3:18:7c:30:58 Applying Local Bridging Interface Policy for st
```

```
*apfMsConnTask_1: Nov 24 04:30:44.318: e4:b3:18:7c:30:58 RSN Capabilities: 60
```

```
*apfMsConnTask_1: Nov 24 04:30:44.318: e4:b3:18:7c:30:58 Marking Mobile as non-
```

```
e4:b3:18:7c:30:58 Received 802.11i 802.1X key management suite, enabling dot1x Authentication
```

```
11w Capable
```

```
*apfMsConnTask_1: Nov 24 04:30:44.318: e4:b3:18:7c:30:58 Received RSN IE with 1 PMKIDs from mobile e4:b3:18:7c:30:58
```

```
*apfMsConnTask_1: Nov 24 04:30:44.319: Received PMKID: (16)
```

```
*apfMsConnTask_1: Nov 24 04:30:44.319: e4:b3:18:7c:30:58 Searching for PMKID in MSCB PMKID cache for mob
```

```
*apfMsConnTask_1: Nov 24 04:30:44.319: e4:b3:18:7c:30:58 No valid PMKID found in the MSCB PMKID cache fo
```

```
*apfMsConnTask_1: Nov 24 04:30:44.319: e4:b3:18:7c:30:58 0.0.0.0 START (0) Initializing policy
```

```
*apfMsConnTask_1: Nov 24 04:30:44.319:
```

```
e4:b3:18:7c:30:58 0.0.0.0 START (0) Change state to AUTHCHECK (2) last state START (0)
```

```
*apfMsConnTask_1: Nov 24 04:30:44.319:
```

```
e4:b3:18:7c:30:58 0.0.0.0 AUTHCHECK (2) Change state to 8021X_REQD (3) last state AUTHCHECK (2)
```

```
*apfMsConnTask_1: Nov 24 04:30:44.319: e4:b3:18:7c:30:58 0.0.0.0 8021X_REQD (3) Plumbed mobile LWAPP ru
```

```
*apfMsConnTask_1: Nov 24 04:30:44.319: e4:b3:18:7c:30:58 apfMsAssoStateInc
```

```
*apfMsConnTask_1: Nov 24 04:30:44.319: e4:b3:18:7c:30:58 apfPemAddUser2 (apf_policy.c:437) Changing stat
```

```
*apfMsConnTask_1: Nov 24 04:30:44.319: e4:b3:18:7c:30:58 apfPemAddUser2:session timeout forstation e4:b3:18:7c:30:58
```

```
*apfMsConnTask_1: Nov 24 04:30:44.319: e4:b3:18:7c:30:58 Stopping deletion of Mobile Station: (callerId:
```

```
*apfMsConnTask_1: Nov 24 04:30:44.319: e4:b3:18:7c:30:58 Func: apfPemAddUser2, Ms Timeout = 0, Session T
```

```
*apfMsConnTask_1: Nov 24 04:30:44.320: e4:b3:18:7c:30:58 Sending Assoc Response to station on BSSID 00:c8:8b:26:2c:d0-00
```

```
*spamApTask2: Nov 24 04:30:44.323: e4:b3:18:7c:30:58 Successful transmission of LWAPP Add-Mobile to AP 00:c8:8b:26:2c:d0-00
```

```
*spamApTask2: Nov 24 04:30:44.325: e4:b3:18:7c:30:58 Received ADD_MOBILE ack - Initiating 1x to STA e4:b3:18:7c:30:58
```

```
*spamApTask2: Nov 24 04:30:44.325: e4:b3:18:7c:30:58
```

```
Sent dot1x auth initiate message for mobile e4:b3:18:7c:30:58
```

```
*Dot1x_NW_MsgTask_0: Nov 24 04:30:44.326: e4:b3:18:7c:30:58 reauth_sm state transition 0 ---> 1 for mobi
```

\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.326: e4:b3:18:7c:30:58 EAP-PARAM Debug - eap-params for Wlan-Id :2  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.326: e4:b3:18:7c:30:58 Disable re-auth, use PMK lifetime.  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.326: e4:b3:18:7c:30:58 Station e4:b3:18:7c:30:58 setting dot1x reauth timeout to 0 seconds, got from  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.326: e4:b3:18:7c:30:58 Stopping reauth timeout for e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.326: e4:b3:18:7c:30:58 dot1x - moving mobile e4:b3:18:7c:30:58 into EAP State  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.326:

**e4:b3:18:7c:30:58 Sending EAP-Request/Identity to mobile e4:b3:18:7c:30:58 (EAP Id 1)**

\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.380: e4:b3:18:7c:30:58 Received EAPOL EAPPKT from mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.380: e4:b3:18:7c:30:58 Received Identity Response (count=1) from mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.380: e4:b3:18:7c:30:58 Resetting reauth count 1 to 0 for mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.380: e4:b3:18:7c:30:58 EAP State update from Connecting to Authenticating  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.380: e4:b3:18:7c:30:58 dot1x - moving mobile e4:b3:18:7c:30:58 into EAP State  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.380: e4:b3:18:7c:30:58 Entering Backend Auth Response state for mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.380: e4:b3:18:7c:30:58 Created Acct-Session-ID (58366cf4/e4:b3:18:7c:30:58) for mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.386: e4:b3:18:7c:30:58 Processing Access-Challenge for mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.387: e4:b3:18:7c:30:58 Entering Backend Auth Req state (id=215) for mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.387: e4:b3:18:7c:30:58 WARNING: updated EAP-Identifier 1 ==> 215 for mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.387: e4:b3:18:7c:30:58 Sending EAP Request from AAA to mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.387: e4:b3:18:7c:30:58 Allocating EAP Pkt for retransmission to mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.390: e4:b3:18:7c:30:58 Received EAPOL EAPPKT from mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.390: e4:b3:18:7c:30:58 Received EAP Response from mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.390: e4:b3:18:7c:30:58 Resetting reauth count 0 to 0 for mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.390: e4:b3:18:7c:30:58 Entering Backend Auth Response state for mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.393: e4:b3:18:7c:30:58 Processing Access-Challenge for mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.393: e4:b3:18:7c:30:58 Entering Backend Auth Req state (id=216) for mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.393: e4:b3:18:7c:30:58 Sending EAP Request from AAA to mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.393: e4:b3:18:7c:30:58 Reusing allocated memory for EAP Pkt for retransmission to mobile e4:b3:18:7c:30:58  
. . .  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.530:

**e4:b3:18:7c:30:58 Processing Access-Accept for mobile e4:b3:18:7c:30:58**

\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.530: e4:b3:18:7c:30:58 Resetting web IPv4 acl from 255 to 255  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.530: e4:b3:18:7c:30:58 Resetting web IPv4 Flex acl from 65535 to 65535  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.530:

**e4:b3:18:7c:30:58 Username entry (user1) created for mobile, length = 253**

\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.530:

**e4:b3:18:7c:30:58 Found an interface name:'vlan2404' corresponds to interface name received: vlan2404**

\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.530: e4:b3:18:7c:30:58 override for default ap group, marking intgrp as default  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.530: e4:b3:18:7c:30:58 Applying Interface(management) policy on Mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.530: e4:b3:18:7c:30:58 Re-applying interface policy for client e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531: e4:b3:18:7c:30:58 apfApplyWlanPolicy: Apply WLAN Policy over Pkts from e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531:

**e4:b3:18:7c:30:58 Inserting AAA Override struct for mobile**

MAC: e4:b3:18:7c:30:58, source 4

\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531: e4:b3:18:7c:30:58 Applying override policy from source Override struct for mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24

**04:30:44.531: e4:b3:18:7c:30:58 Found an interface name:'vlan2404' corresponds to interface name received: vlan2404**

\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531: e4:b3:18:7c:30:58 Applying Interface(vlan2404) policy on Mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531: e4:b3:18:7c:30:58 Re-applying interface policy for client e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531: e4:b3:18:7c:30:58 Setting re-auth timeout to 0 seconds, got from e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531: e4:b3:18:7c:30:58 Station e4:b3:18:7c:30:58 setting dot1x reauth timeout to 0 seconds, got from e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531: e4:b3:18:7c:30:58 Stopping reauth timeout for e4:b3:18:7c:30:58

\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531: e4:b3:18:7c:30:58 Creating a PKC PMKID Cache entry for station  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531: e4:b3:18:7c:30:58 Resetting MSCB PMK Cache Entry 0 for station  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531: e4:b3:18:7c:30:58 Adding BSSID 00:c8:8b:26:2c:d1 to PMKID cache  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531: New PMKID: (16)  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531: [0000] cc 3a 3d 26 80 17 8b f1 2d c5 cd fd a0 8a c4 39  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531: e4:b3:18:7c:30:58 unsetting PmkIdValidatedByAp  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531: e4:b3:18:7c:30:58 Updating AAA Overrides from local for station  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531: e4:b3:18:7c:30:58 Adding Audit session ID payload in Mobility  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531: e4:b3:18:7c:30:58 0 PMK-update groupcast messages sent  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531: e4:b3:18:7c:30:58 PMK sent to mobility group  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531: e4:b3:18:7c:30:58 Disabling re-auth since PMK lifetime can take  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.531: e4:b3:18:7c:30:58 Sending EAP-Success to mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.532: e4:b3:18:7c:30:58 Freeing AAACB from Dot1xCB as AAA auth is done  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.532: e4:b3:18:7c:30:58 key Desc Version FT - 0  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.532: e4:b3:18:7c:30:58 Found an cache entry for BSSID 00:c8:8b:26:2c:d1  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.532: Including PMKID in M1 (16)  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.532: [0000] cc 3a 3d 26 80 17 8b f1 2d c5 cd fd a0 8a c4 39  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.532: M1 - Key Data: (22)  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.532: [0000] dd 14 00 0f ac 04 cc 3a 3d 26 80 17 8b f1 2d c5  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.532: [0016] cd fd a0 8a c4 39  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.532:

**e4:b3:18:7c:30:58 Starting key exchange to mobile e4:b3:18:7c:30:58, data packets will be dropped**

\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.532:

**e4:b3:18:7c:30:58 Sending EAPOL-Key Message to mobile e4:b3:18:7c:30:58**

state INITPMK (message 1), replay counter 00.00.00.00.00.00.00.00

\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.532: e4:b3:18:7c:30:58 Reusing allocated memory for EAP Pkt for retransmission  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.532: e4:b3:18:7c:30:58 Entering Backend Auth Success state (id=223)  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.532: e4:b3:18:7c:30:58 Received Auth Success while in Authenticating state  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.532: e4:b3:18:7c:30:58 dot1x - moving mobile e4:b3:18:7c:30:58 into PTK state  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.547: e4:b3:18:7c:30:58 Received EAPOL-Key from mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.547: e4:b3:18:7c:30:58 Ignoring invalid EAPOL version (1) in EAPOL-Key  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.547: e4:b3:18:7c:30:58 key Desc Version FT - 0  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.547:

**e4:b3:18:7c:30:58 Received EAPOL-key in PTK\_START state (message 2) from mobile**

e4:b3:18:7c:30:58

\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.548: e4:b3:18:7c:30:58 Successfully computed PTK from PMK!!!  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.548: e4:b3:18:7c:30:58 Received valid MIC in EAPOL Key Message M2!  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.548: e4:b3:18:7c:30:58 Not Flex client. Do not distribute PMK Key Data  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.548: e4:b3:18:7c:30:58 Stopping retransmission timer for mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.548: e4:b3:18:7c:30:58 key Desc Version FT - 0  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.548: e4:b3:18:7c:30:58 Sending EAPOL-Key Message to mobile e4:b3:18:7c:30:58  
state PTKINITNEGOTIATING (message 3), replay counter 00.00.00.00.00.00.00.01  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.548: e4:b3:18:7c:30:58 Reusing allocated memory for EAP Pkt for retransmission  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.555: e4:b3:18:7c:30:58 Received EAPOL-Key from mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.555: e4:b3:18:7c:30:58 Ignoring invalid EAPOL version (1) in EAPOL-Key  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.555: e4:b3:18:7c:30:58 key Desc Version FT - 0  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.555:

**e4:b3:18:7c:30:58 Received EAPOL-key in PTKINITNEGOTIATING state (message 4)**

from mobile e4:b3:18:7c:30:58

\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.555: e4:b3:18:7c:30:58 Stopping retransmission timer for mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.555: e4:b3:18:7c:30:58 Freeing EAP Retransmit Buffer for mobile e4:b3:18:7c:30:58  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.555: e4:b3:18:7c:30:58 apfMs1xStateInc  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.555: e4:b3:18:7c:30:58 apfMsPeapSimReqCntInc  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.555: e4:b3:18:7c:30:58 apfMsPeapSimReqSuccessCntInc  
\*Dot1x\_NW\_MsgTask\_0: Nov 24 04:30:44.555:

**e4:b3:18:7c:30:58 0.0.0.0 8021X\_REQD (3) Change state to L2AUTHCOMPLETE (4) last state 8021X\_REQD (3)**

```

*Dot1x_NW_MsgTask_0: Nov 24 04:30:44.555: e4:b3:18:7c:30:58 Mobility query, PEM State: L2AUTHCOMPLETE
*Dot1x_NW_MsgTask_0: Nov 24 04:30:44.555: e4:b3:18:7c:30:58 Building Mobile Announce :
*Dot1x_NW_MsgTask_0: Nov 24 04:30:44.556: e4:b3:18:7c:30:58 Building Client Payload:
*Dot1x_NW_MsgTask_0: Nov 24 04:30:44.556: e4:b3:18:7c:30:58 Client Ip: 0.0.0.0
*Dot1x_NW_MsgTask_0: Nov 24 04:30:44.556: e4:b3:18:7c:30:58 Client Vlan Ip: 172.16.0.134, Vlan mask
*Dot1x_NW_MsgTask_0: Nov 24 04:30:44.556: e4:b3:18:7c:30:58 Client Vap Security: 16384
*Dot1x_NW_MsgTask_0: Nov 24 04:30:44.556: e4:b3:18:7c:30:58 Virtual Ip: 10.10.10.10
*Dot1x_NW_MsgTask_0: Nov 24 04:30:44.556: e4:b3:18:7c:30:58 ssid: ise-ssid
*Dot1x_NW_MsgTask_0: Nov 24 04:30:44.556: e4:b3:18:7c:30:58 Building VlanIpPayload.
*Dot1x_NW_MsgTask_0: Nov 24 04:30:44.556: e4:b3:18:7c:30:58 Not Using WMM Compliance code qosCap 00
*Dot1x_NW_MsgTask_0: Nov 24 04:30:44.556: e4:b3:18:7c:30:58 0.0.0.0 L2AUTHCOMPLETE (4) Plumbed mobile L
*Dot1x_NW_MsgTask_0: Nov 24 04:30:44.556:

```

```

e4:b3:18:7c:30:58 0.0.0.0 L2AUTHCOMPLETE (4) Change state to DHCP_REQD (7) last state L2AUTHCOMPLETE (4)

```

```

*Dot1x_NW_MsgTask_0: Nov 24 04:30:44.556: e4:b3:18:7c:30:58 0.0.0.0 DHCP_REQD (7) pemAdvanceState2 6677,
*Dot1x_NW_MsgTask_0: Nov 24 04:30:44.556: e4:b3:18:7c:30:58 0.0.0.0 DHCP_REQD (7) Adding Fast Path rule
type = Airespace AP - Learn IP address
on AP 00:c8:8b:26:2c:d0, slot 0, interface = 1, QOS = 0
IPv4 ACL ID = 255, IPv
*Dot1x_NW_MsgTask_0: Nov 24 04:30:44.556: e4:b3:18:7c:30:58 0.0.0.0 DHCP_REQD (7) Fast Path rule (contd...)
*Dot1x_NW_MsgTask_0: Nov 24 04:30:44.556: e4:b3:18:7c:30:58 0.0.0.0 DHCP_REQD (7) Fast Path rule (contd...)
*Dot1x_NW_MsgTask_0: Nov 24 04:30:44.556: e4:b3:18:7c:30:58 0.0.0.0 DHCP_REQD (7) Successfully plumbed m
*Dot1x_NW_MsgTask_0: Nov 24 04:30:44.556: e4:b3:18:7c:30:58 Successfully Plumbed PTK session Keysfor mob
*spamApTask2: Nov 24 04:30:44.556: e4:b3:18:7c:30:58 Successful transmission of LWAPP Add-Mobile to AP 0
*pemReceiveTask: Nov 24 04:30:44.557: e4:b3:18:7c:30:58 0.0.0.0 Added NPU entry of type 9, dtlFlags 0x0
*apfReceiveTask: Nov 24 04:30:44.557: e4:b3:18:7c:30:58 0.0.0.0 DHCP_REQD (7) mobility role update requ
Peer = 0.0.0.0, Old Anchor = 0.0.0.0, New Anchor = 172.16.0.3
*apfReceiveTask: Nov 24 04:30:44.557: e4:b3:18:7c:30:58 0.0.0.0 DHCP_REQD (7) State Update from Mobility
*apfReceiveTask: Nov 24 04:30:44.557: e4:b3:18:7c:30:58 0.0.0.0 DHCP_REQD (7) pemAdvanceState2 6315, Ad
*apfReceiveTask: Nov 24 04:30:44.557: e4:b3:18:7c:30:58 0.0.0.0 DHCP_REQD (7) Replacing Fast Path rule
IPv4 ACL ID = 255,
*apfReceiveTask: Nov 24 04:30:44.557: e4:b3:18:7c:30:58 0.0.0.0 DHCP_REQD (7) Fast Path rule (contd...)
*apfReceiveTask: Nov 24 04:30:44.557: e4:b3:18:7c:30:58 0.0.0.0 DHCP_REQD (7) Fast Path rule (contd...)
*apfReceiveTask: Nov 24 04:30:44.557: e4:b3:18:7c:30:58 0.0.0.0 DHCP_REQD (7) Successfully plumbed mobil
*pemReceiveTask: Nov 24 04:30:44.557: e4:b3:18:7c:30:58 Sent an XID frame
*dtlArpTask: Nov 24 04:30:47.932: e4:b3:18:7c:30:58 Static IP client associated to interface vlan2404 wh
*dtlArpTask: Nov 24 04:30:47.933: e4:b3:18:7c:30:58 apfMsRunStateInc
*dtlArpTask: Nov 24 04:30:47.933:

```

```

e4:b3:18:7c:30:58 172.16.0.151 DHCP_REQD (7) Change state to RUN (20)

```

```

last state DHCP_REQD (7)

```

For an easy way to read debug client outputs, use the [Wireless debug analyzer tool](#):

[Wireless Debug Analyzer](#)

## Authentication Process on ISE

Navigate to **Operations > RADIUS > Live Logs** in order to see which authentication policy, authorization policy, and authorization profile was assigned to the user.

For more information, click **Details** in order to see a more detailed authentication process as shown in the image.

Identity Services Engine

Home > Context Visibility > **Operations** > Policy > Administration > Work Centers

▼ RADIUS TC-NAC Live Logs > TACACS Reports > Troubleshoot > Adaptive Network Control

**Live Logs** Live Sessions

Misconfigured Supplicants 0 Misconfigured Network Devices 0 RADIUS Drops 0 Client Stopp

Refresh Never

Refresh Reset Repeat Counts Export To

| Time  | Sta... | Details | Ide... | Endpoint ID       | Endpoint ... | Authentication Policy           | Authorization Policy  |
|-------|--------|---------|--------|-------------------|--------------|---------------------------------|-----------------------|
| No... |        |         | user1  | 08:74:02:77:13:45 | Apple-Device | Default >> Rule name >> Default | Default >> NameAuthZr |

## Troubleshoot

There is currently no specific information available to troubleshoot this configuration.