

Install a Third-Party CA-signed Certificate in ISE

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Background Information](#)

[Configure](#)

[Step 1. Generate Certificate Signing Request \(CSR\).](#)

[Step 2. Import a New Certificate Chain.](#)

[Verify](#)

[Troubleshoot](#)

[Supplicant Does not Trust the ISE Local Server Certificate during a dot1x Authentication](#)

[ISE Certificate Chain is Correct but Endpoint Rejects ISEServer Certificate during Authentication](#)

[Related Information](#)

Introduction

This document describes how to install a certificate signed by a third-party Certificate Authority (CA) in Cisco Identity Services Engine (ISE).

Prerequisites

Requirements

Cisco recommends that you have knowledge of Basic Public Key Infrastructure.

Components Used

The information in this document is based on Cisco Identity Services Engine (ISE) Release 3.0. The same configuration applies to releases 2.X

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.

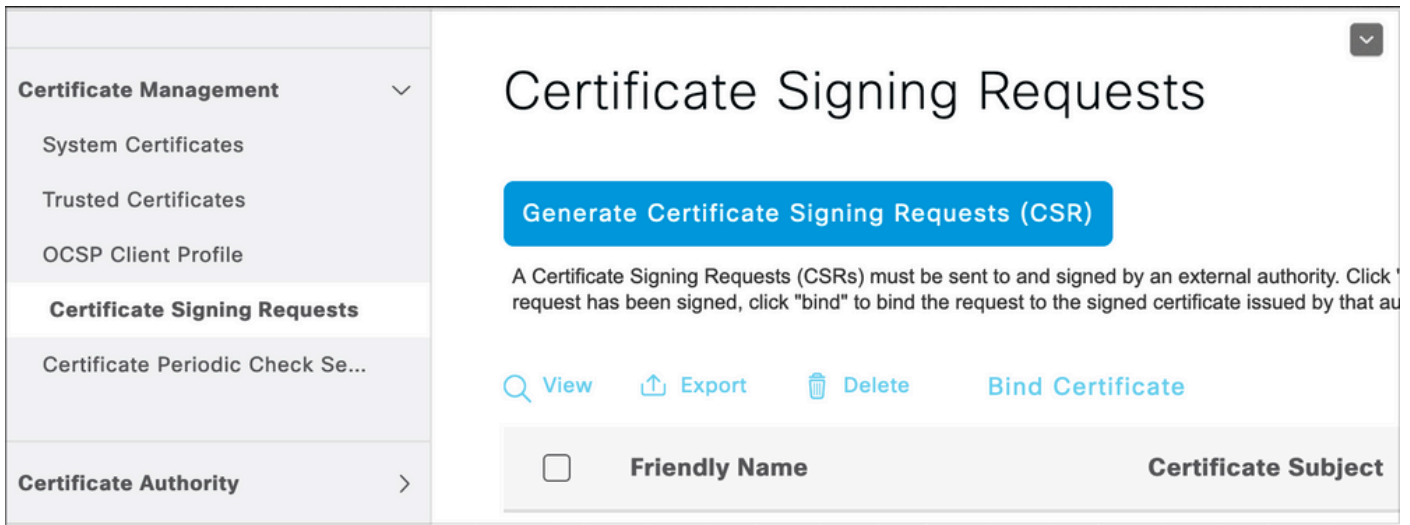
Background Information

This process is the same regardless of the final certificate role (EAP authentication, Portal, Admin, and pxGrid).

Configure


Step 1. Generate Certificate Signing Request (CSR).

In order to generate the CSR, navigate to **Administration > Certificates > Certificate Signing Requests** and click on **Generate Certificate Signing Requests (CSR)**.




The screenshot shows the 'Certificate Signing Requests' management page. On the left is a navigation sidebar with categories: 'Certificate Management' (containing System Certificates, Trusted Certificates, and OCSP Client Profile), 'Certificate Signing Requests' (containing Certificate Periodic Check Se...), and 'Certificate Authority'. The main content area has a title 'Certificate Signing Requests' and a prominent blue button labeled 'Generate Certificate Signing Requests (CSR)'. Below the button is a note: 'A Certificate Signing Requests (CSRs) must be sent to and signed by an external authority. Click 'request has been signed, click "bind" to bind the request to the signed certificate issued by that au'. At the bottom of the main area are action buttons: 'View', 'Export', 'Delete', and 'Bind Certificate'. Below these is a table header with columns 'Friendly Name' and 'Certificate Subject'.

1. Under the Usage section, select the role to be used from the drop-down menu. If the certificate is used for multiple roles you can select Multi-use. Once the certificate is generated the roles can be changed if necessary.
2. Select the node for which the certificate can be generated.
3. Fill out the information as needed (Organizational Unit, Organization, City, State and Country).

 **Note:** Under Common Name (CN) field ISE auto-populates the node Fully Qualified Domain Name (FQDN).

Wildcards:


- If the goal is to generate a wildcard certificate check the **Allow Wildcard Certificates** box.
- If the certificate is used for EAP authentications the * symbol must not be in the Subject CN field as Windows supplicants reject the server certificate.
- Even when **Validate Server Identity** is disabled on the supplicant, the SSL handshake can fail when the * is in the CN field.
- Instead, a generic FQDN can be used in the CN field, and then the *.domain.com can be used on the Subject Alternative Name (SAN) DNS Name field.

 **Note:** Some Certificate Authorities (CA) can add the wildcard (*) in the CN of the certificate automatically even if it not present in the CSR. In this scenario, a special request is required to be raised to prevent this action.

Individual server certificate CSR example:

Usage

Certificate(s) will be used for Multi-Use 

 You can use a single certificate for multiple services, but doing so is not a recommended practice. Rather, you should obtain individual certificates specifically for each service (for example, one certificate each for Guest Portals, EAP, and pxGrid).

Allow Wildcard Certificates 

Node(s)

Generate CSR's for these Nodes:

Node	CSR Friendly Name
<input checked="" type="checkbox"/> abtomar30	abtomar30#Multi-Use

Subject

Common Name (CN)
\$FQDN\$ 

Organizational Unit (OU)
Cisco TAC 

Organization (O)
Cisco 

City (L)
Bangalore



State (ST)
Karnataka

Country (C)
IN

Subject Alternative Name (SAN)

 IP Address  10.106.120.87   


* Key type

RSA  

Wildcard CSR example:

Usage


Certificate(s) will be used for Multi-Use

 You can use a single certificate for multiple services, but doing so is not a recommended practice. Rather, you should obtain individual certificates specifically for each service (for example, one certificate each for Guest Portals, EAP, and pxGrid).

Allow Wildcard Certificates 

Subject

Common Name (CN)

Mycluster.mydomain.com 

Organizational Unit (OU)

Cisco TAC 

Organization (O)

Cisco 

City (L)

Bangalore

State (ST)

Karnataka

Country (C)

IN

Subject Alternative Name (SAN)



IP Address



10.106.120.87



DNS Name




*.mydomain.com



* Key type


RSA



 **Note:** Each deployment node(s) IP address can be added to the SAN field to avoid a certificate warning when you access the server via the IP address.

Once the CSR is created, ISE displays a pop-up window with the option to export it. Once exported, this file must be sent to the CA for signing.



Successfully generated CSR(s) 

Certificate Signing request(s) generated:

abtomar30.abtomar.local#Multi-Use

Click Export to download CSR(s) or OK to return to list of CSR(s) screen


OK


Export

Step 2. Import a New Certificate Chain.

The Certificate Authority returns the signed server certificate along with the full certificate chain (Root/Intermediate). Once received, do the steps here to import the certificates into your ISE server:

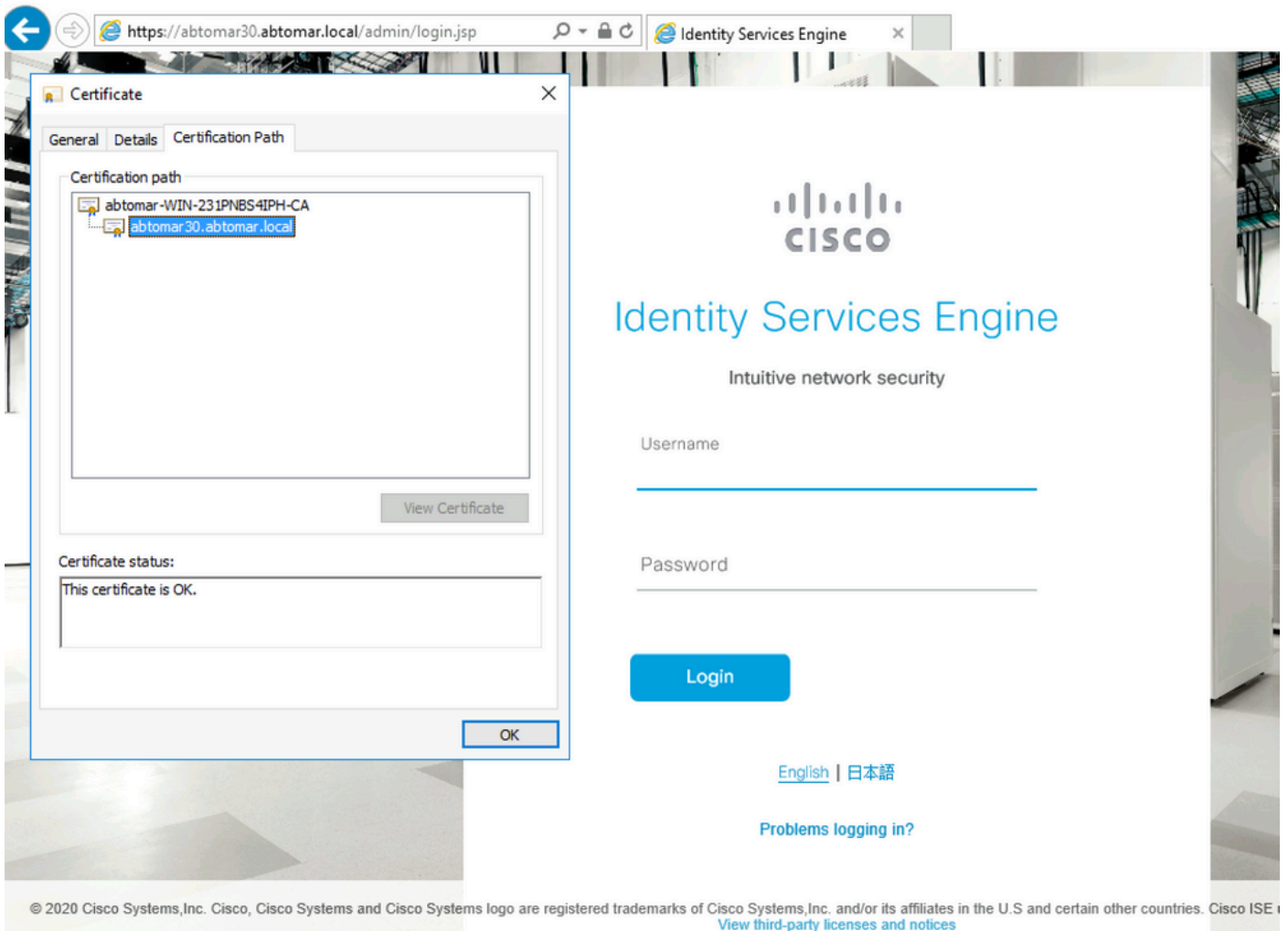
1. In order to import any Root and (or) Intermediate certificates provided by the CA, navigate to **Administration > Certificates > Trusted Certificates**.
2. Click **Import** and then choose the Root and/or Intermediate certificate and choose the relevant check boxes as they applied to submit.
3. In order to import the Server certificate, navigate to **Administration > Certificates > Certificate Signing Requests**.
4. Select the CSR previously created and click on **Bind Certificate**.
5. Select the new certificate location and ISE binds the certificate to the private key created and stored in the database.

 **Note:** If the Admin Role has been selected for this certificate, the specific ISE server services restart.

 **Caution:** If the certificate imported is for the Primary Administration Node of the deployment and if the Admin role is selected, then the services on all nodes restart one after the other. This is expected and a downtime is recommended to perform this activity.

Verify

If the admin role was selected during the certificate import you can verify the new certificate is in place by loading the admin page in the browser. The browser must trust the new admin certificate as long as the chain was built correctly and if the certificate chain is trusted by the browser.



For additional verification select the lock symbol in the browser and under the certificate path verify the full chain is present and trusted by the machine. This is not a direct indicator that the full chain was passed down correctly by the server but an indicator of the browser able to trust the server certificate based on its local trust store.

Troubleshoot

Supplicant Does not Trust the ISE Local Server Certificate during a dot1x Authentication

Verify ISE is passing the full certificate chain during the SSL handshake process.

When using EAP methods that require a server certificate (that is, PEAP) and **Validate Server Identity** is selected, the supplicant validates the certificate chain using the certificates it has in its local trust store as part of the authentication process. As part of the SSL handshake process, ISE presents its certificate and also any Root and (or) intermediate certificates present in its chain. The supplicant won't be able to validate the server identity if the chain is incomplete. To verify the certificate chain is passed back to your client, you can perform the next steps:

1. In order to take a capture from ISE (TCPDump) during the authentication, navigate to **Operations > Diagnostic Tools > General Tools > TCP Dump**.
2. Download/Open the capture and apply the filter **ssl.handshake.certificates** in Wireshark and find an access-challenge.

- Once Selected, navigate to **Expand Radius Protocol > Attribute Value Pairs > EAP-Message Last segment > Extensible Authentication Protocol > Secure Sockets Layer > Certificate > Certificates.**

Certificate chain in the capture.

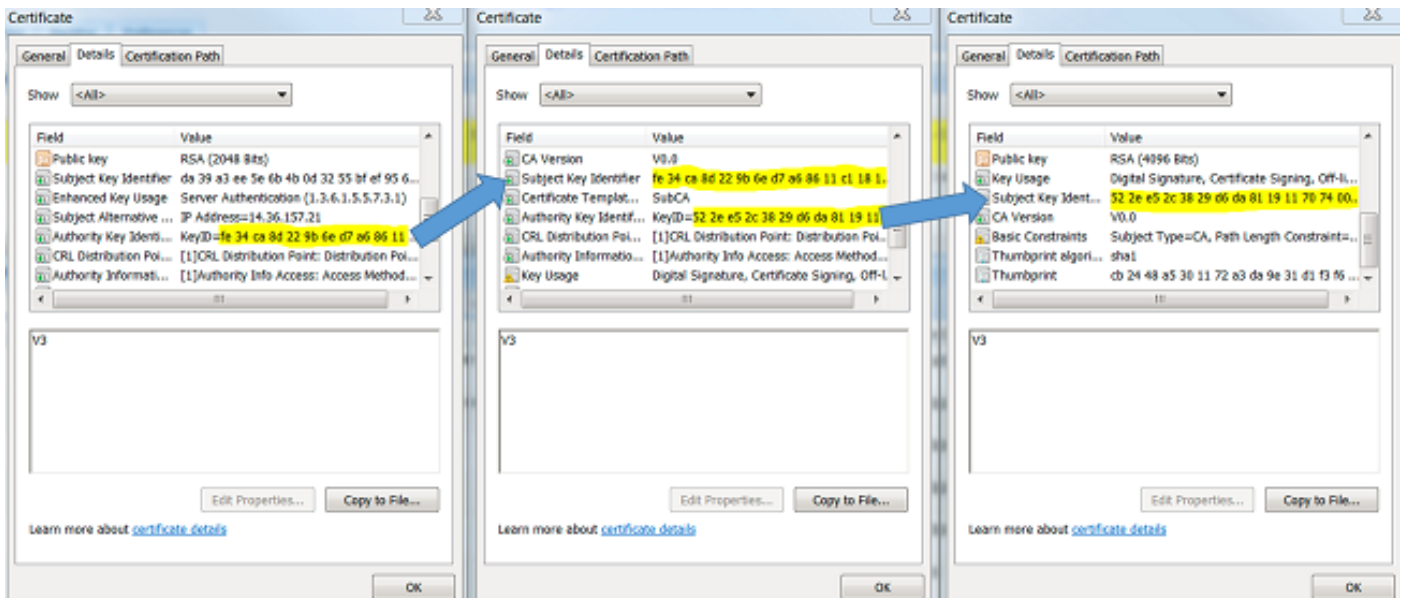
The image shows a Wireshark capture of network traffic. The top pane displays a list of packets with columns for No., Time, Source, Destination, Protocol, Length, and Info. The bottom pane shows the detailed view of a selected packet, specifically an EAP-Message (79) Last Segment. The structure is as follows:

- AVP: l=255 t=EAP-Message(79) Segment[1]
- AVP: l=255 t=EAP-Message(79) Segment[2]
- AVP: l=255 t=EAP-Message(79) Segment[3]
- AVP: l=255 t=EAP-Message(79) Last Segment[4]
 - EAP fragment
 - Extensible Authentication Protocol
 - Code: Request (1)
 - Id: 41
 - Length: 1012
 - Type: Protected EAP (EAP-PEAP) (25)
 - EAP-TLS Flags: 0xc0
 - EAP-TLS Length: 3141
 - [4 EAP-TLS Fragments (3141 bytes): #857(1002), #860(1002), #862(1002), #864(135)]
 - Secure Sockets Layer
 - TLSv1 Record Layer: Handshake Protocol: Server Hello
 - TLSv1 Record Layer: Handshake Protocol: Certificate
 - Content Type: Handshake (22)
 - Version: TLS 1.0 (0x0301)
 - Length: 3048
 - Handshake Protocol: Certificate
 - Handshake Type: Certificate (11)
 - Length: 3044
 - Certificates Length: 3041
 - Certificates (3041 bytes)
 - Certificate Length: 1656
 - Certificate (id-at-commonName=TORISE20A.rtpaaa.net,id-at-organizationalUnitName=RTPAAA,id-at-organizationName=CISCO,id-at-localityName=RT) Certificate Length: 1379
 - Certificate (id-at-commonName=rtpaaa-ca,dc=rtpaaa,dc=net)

If the chain is incomplete, navigate to **ISE Administration > Certificates > Trusted Certificates** and verify that the Root and (or) Intermediate certificates are present. If the certificate chain is passed successfully, the chain itself must be verified as valid by using the method outlined here.

Open each certificate (server, intermediate and root) and verify the chain of trust by matching the Subject Key Identifier (SKI) of each certificate to the Authority Key Identifier (AKI) of the next certificate in the chain.

Example of a certificate chain.

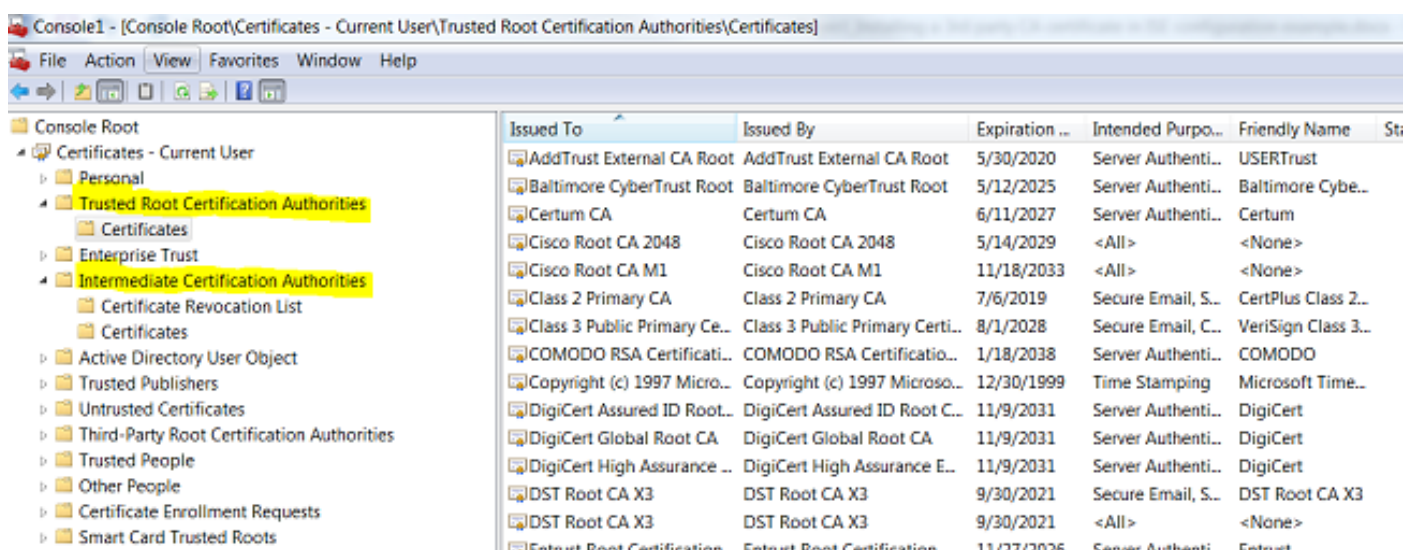


ISE Certificate Chain is Correct but Endpoint Rejects ISE Server Certificate during Authentication

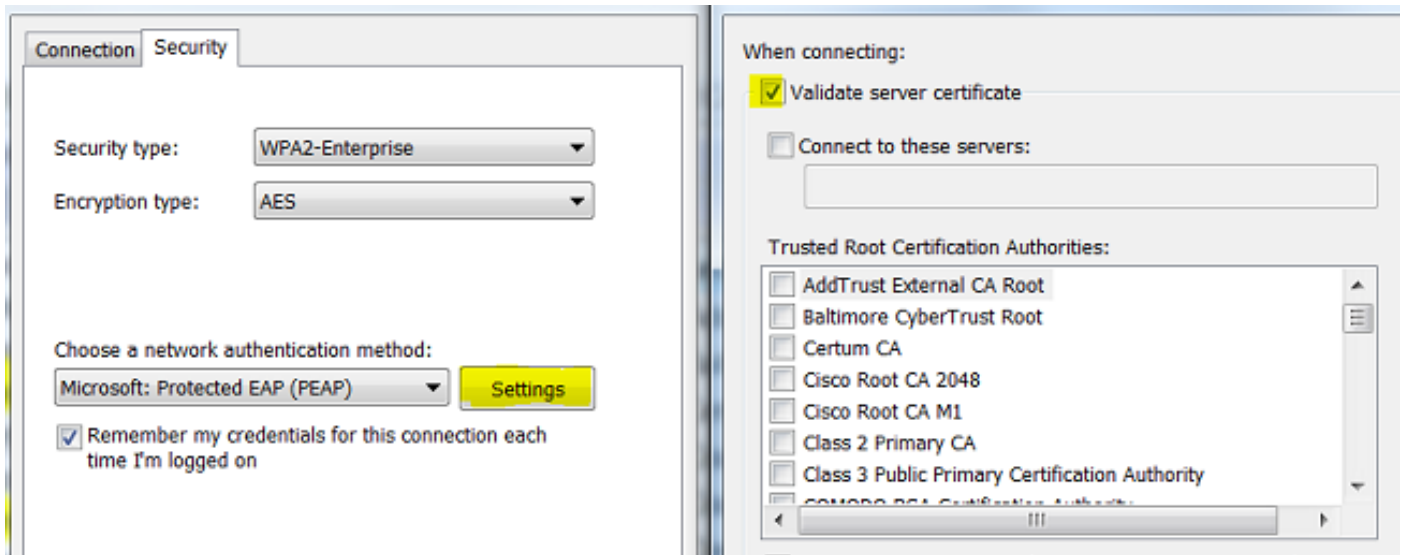
If ISE is presenting its full certificate chain during the SSL handshake and the supplicant is still rejecting the certificate chain; the next step is to verify that the Root and(or) Intermediate certificates are in the client Local Trust Store.

In order to verify this from a Windows device, navigate to **mmc.exe File > Add-Remove Snap-in**. From Available snap-ins column select **Certificates** and click on **Add**. Select either **My user account** or **computer account** depending upon the authentication type in use (User or Machine) and then click on **OK**.

Under the console view, select **Trusted Root Certification Authorities** and **Intermediate Certification Authorities** to verify the presence of Root and Intermediate certificate in the local trust store.



An easy way to verify that this is a Server Identity Check issue, uncheck **Validate Server Certificate** under the supplicant profile configuration and test it again.



Related Information

- [Cisco Identity Services Engine Administrator Guide, Release 3.0](#)
- [Cisco Technical Support & Downloads](#)