

Lab – Configure Group Policy to Trust a Self-Signed Certificate

Overview

In this lab, students will learn how to configure the setting in group policy that will allow all workstations in the domain to trust a self-signed certificate bound in IIS automatically. Rolling this GPO out at the domain level will prevent users from having to accept the security risk each time they access any internal website using a browser.

Lab Requirements

- Server 2012 or 2016 running as a domain controller

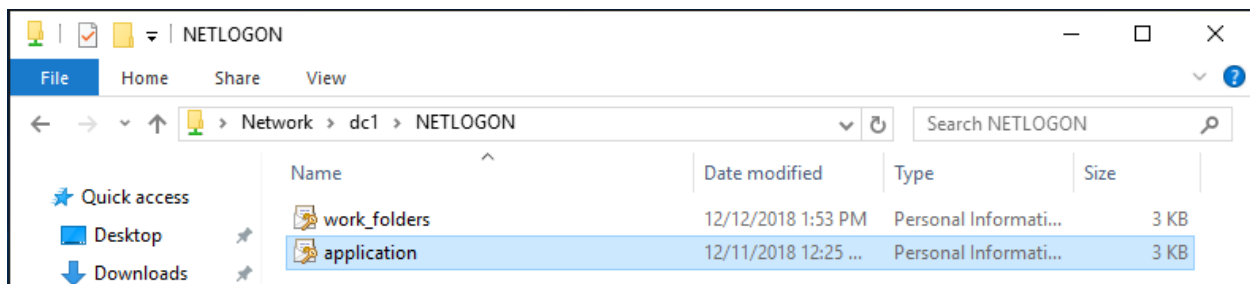
Begin the lab

Begin by relocating your **application_testing** cert from your desktop to your NETLOGON share. Anything that needs to be made accessible to everyone on the network can be placed in this folder to include certificates being rolled out using Group Policy.

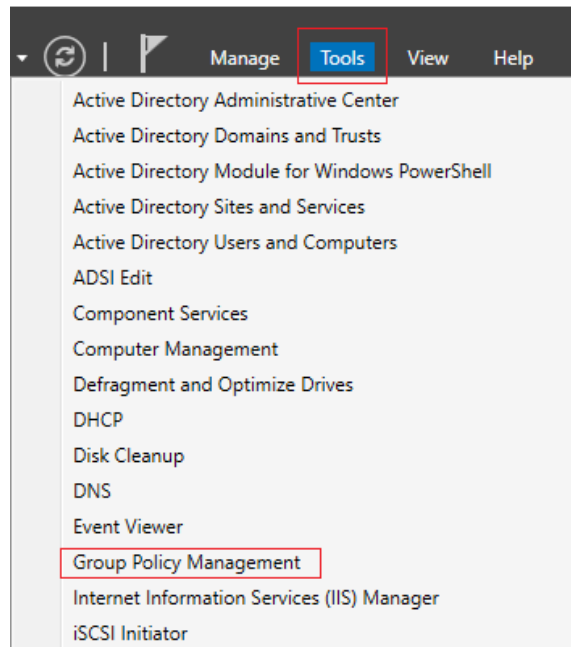
You can access the NETLOGON share by using the network path in either the Windows search bar, File Explorer or the run line, \\dc1



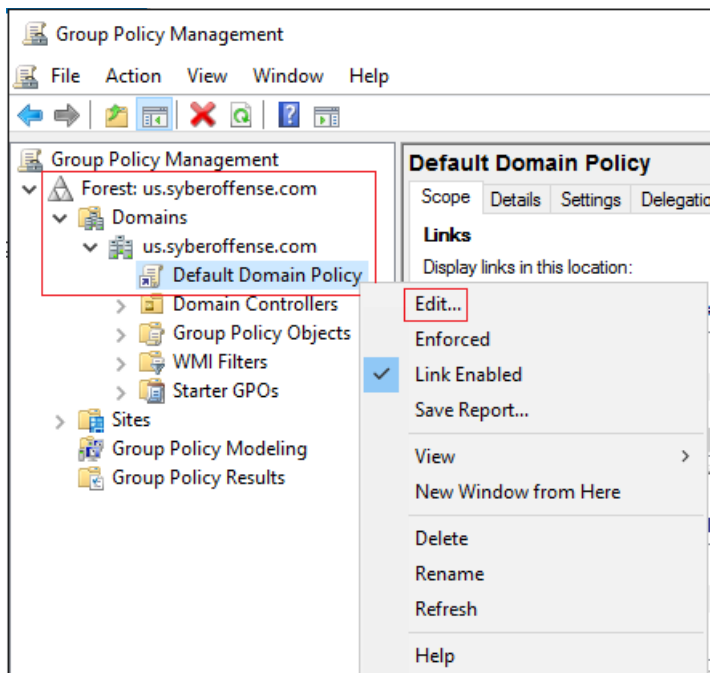
Hit enter, and you can now see and access your NETLOGON share. Either drag and drop the certificate into the folder or copy and paste the cert into the folder.



From the desktop of your domain controller, using Server Manager, go to Tools and from the context menu, select the Group Policy Management Snap-in.



In the left window pane, expand your forest details. Under domains, expand your domain details. From the available list of options, find the Default Domain Policy, right and from the context menu select edit.

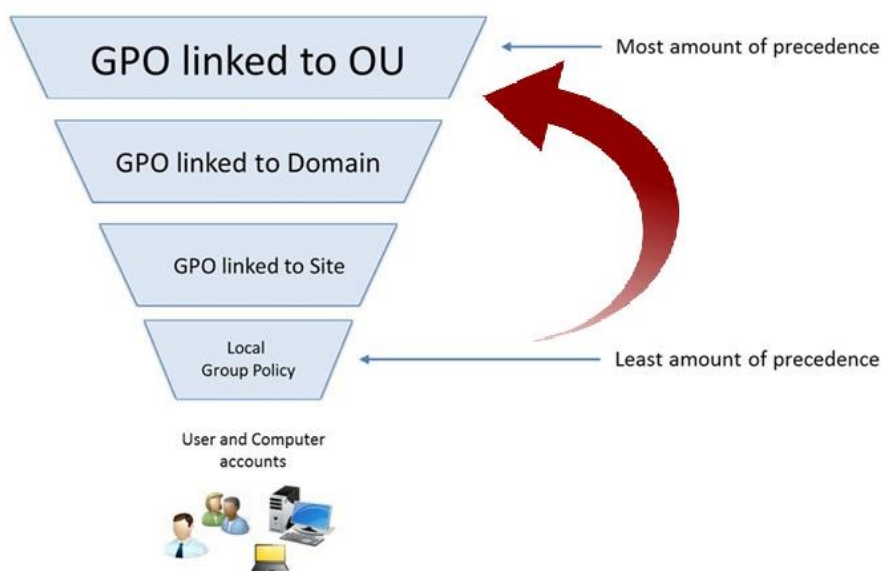


Why not just create a new GPO?

When a user logs onto the domain, their machine must load each GPO one at a time. Every setting in every GPO assigned must be examined to see if a setting needs to be Enabled, Disabled or Ignored. The settings inside a GPO are shortcuts to the machine's registry meaning that every registry setting must be examined and either ignored or modified. The same process is repeated but in reverse each time a user logs off their machine. Each setting must now be undone one GPO at a time.

For this reason, it is always best to use as few GPO's as possible and since the settings to trust our self-signed certificate must be rolled out to every machine in the domain, we can use the Default Domain Policy.

GPO's are loaded from the bottom up and the last one in wins. Therefore, GPO's assigned to OU's inside of Active Directory are loaded last. If there are any conflicts with the settings being applied from any two GPO's, precedence takes place in the following order.



Any computer or user setting that needs to affect every machine or user in the domain can use the Default Domain Policy. GPO's going across a slow site link must carefully be considered and kept to an absolute minimum.

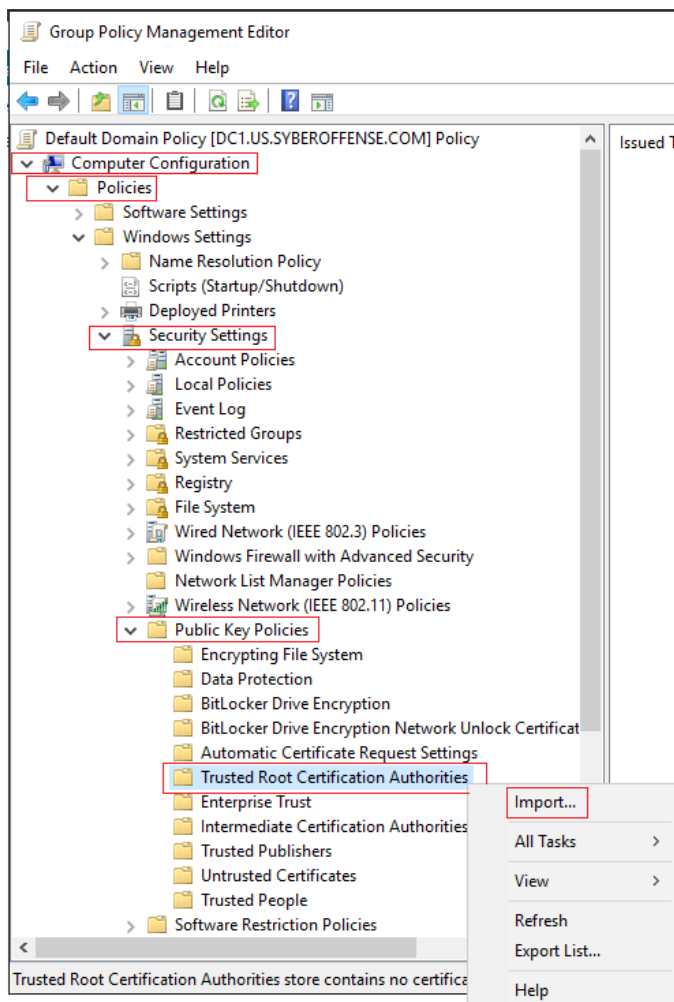
Some network admins will want to configure a new group policy for every computer or user setting for the domain, and that's fine, but when you start loading 10 or 15 GPO's, conflicts can start to happen, and users will have to wait just a little longer for their desktop to appear.

When a specific GPO is needed for a certain group of users or computers, either should be placed in a separate OU with the group policy linked directly to them. Remember, the last one in wins unless there is a conflict, but even that can be fixed using the Group Policy loopback feature. Enough about Group Policy.

On with the lab

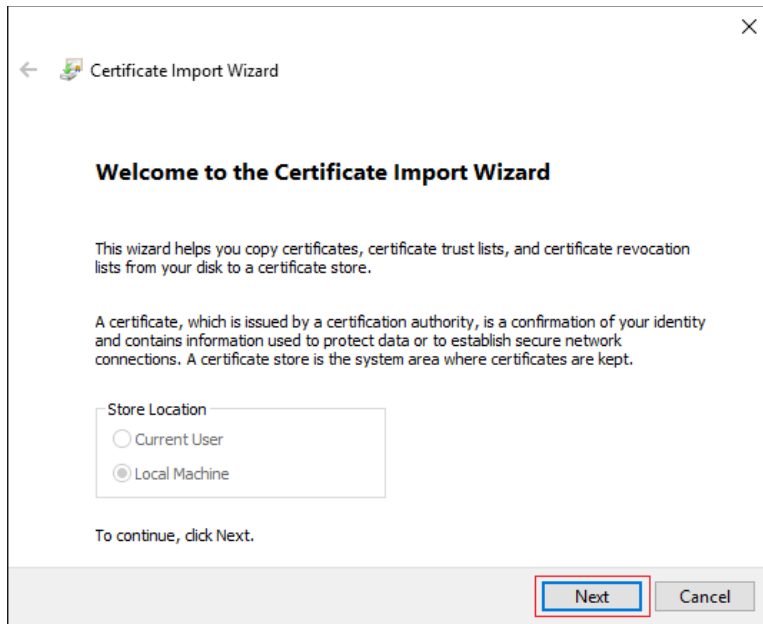
You now have the Group Policy Editor open for your Default Domain Policy. In the left window pane for the Default Domain Policy settings, click and expand Computer Configuration, expand Policies, expand Windows Settings, expand Security Settings, and finally, expand the Public Key Policies.

Inside the Public Key Policies container, find the Trusted Root Certification authority container, right click and from the context menu, select Import.

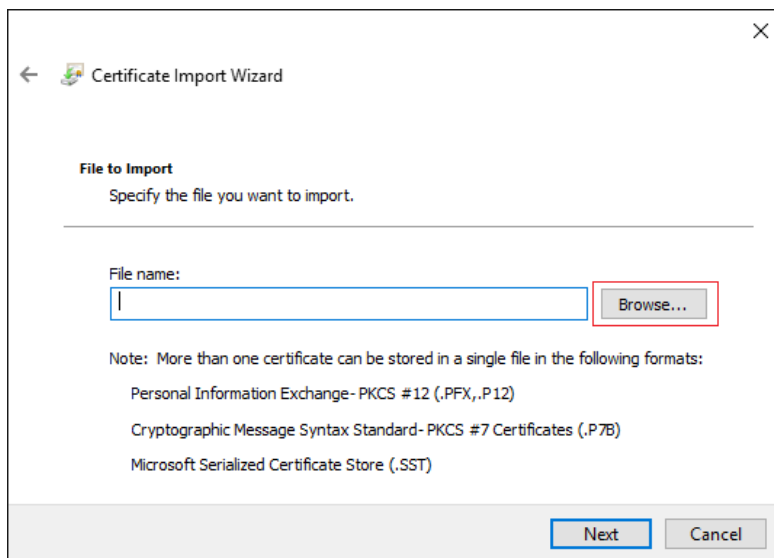




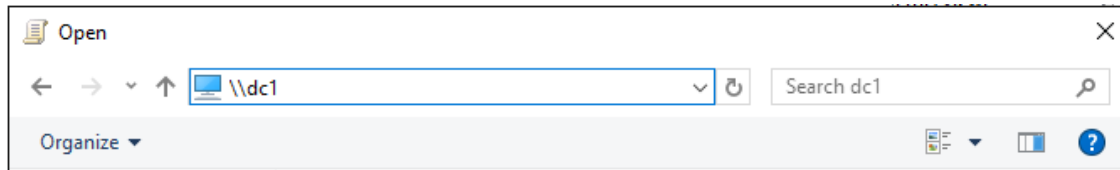
The Certificate Import Wizard opens up. Click next.



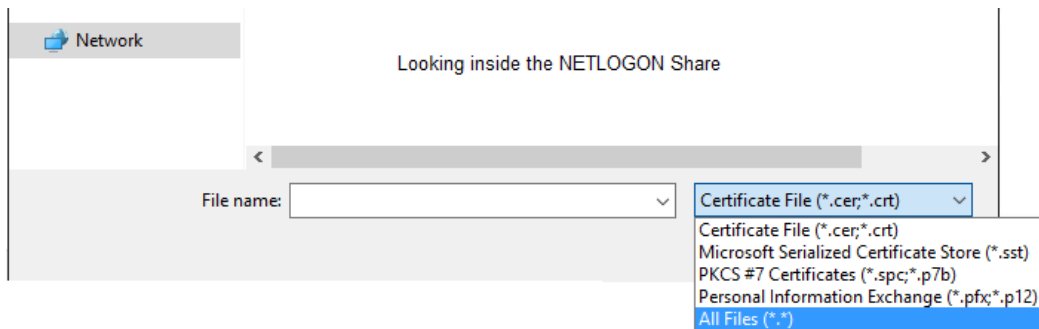
On the next window, click on the browse button.



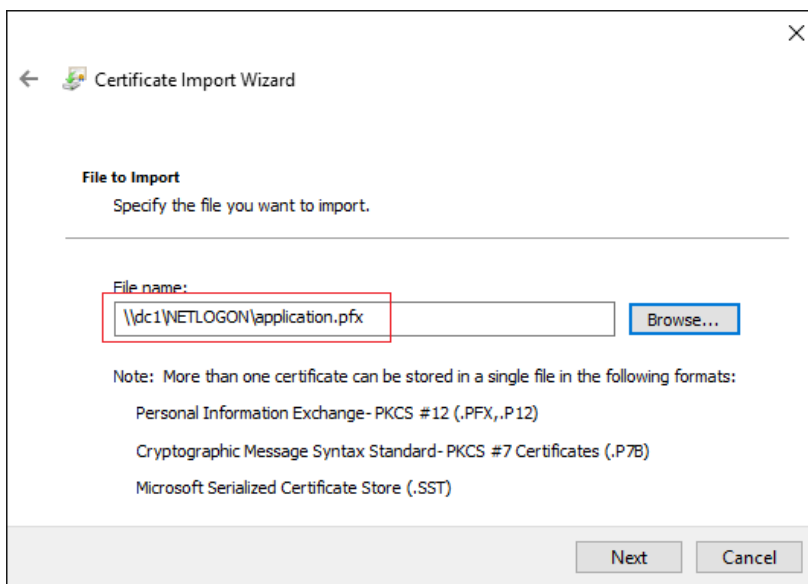
On the next window in the path box, in the location box, type the network path for your domain controller. In this example, my network path is [\\dc1](#).



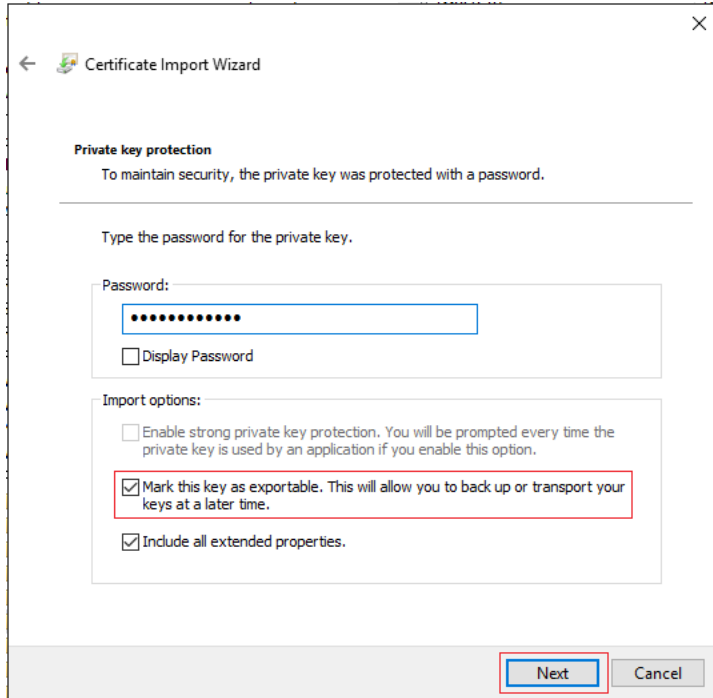
Open your NETLOGON folder and at the bottom right of the window where it shows Certificate Files as the file type, pull down the window and select All files to see the available certs available inside the NETLOGON share.



Using the All files as our file type, we see the application certificate. Select the Application certificate for the needed path. On the next window, Click Next.



On the next window, type in the password when you first created the cert. Check the box to mark the key exportable. Click Next.



← Certificate Import Wizard

Private key protection
To maintain security, the private key was protected with a password.

Type the password for the private key.

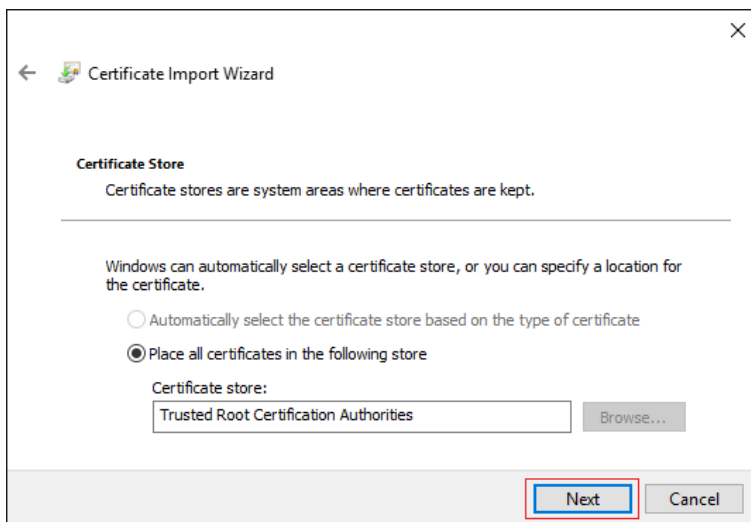
Password:
[Password field with 10 dots]
☐ Display Password

Import options:
☐ Enable strong private key protection. You will be prompted every time the private key is used by an application if you enable this option.
☒ Mark this key as exportable. This will allow you to back up or transport your keys at a later time.
☒ Include all extended properties.

Next Cancel

On the next window, make sure the Trusted Root Certification Authorities store is selected.

This cert will now be pushed out to every computer on the domain and placed in the users Trusted Root Certification Authority container inside their certificate store allowing then to automatically trust any internal IIS site that requires the self-signed cert be trusted to establish a secure SSL connection.



← Certificate Import Wizard

Certificate Store
Certificate stores are system areas where certificates are kept.

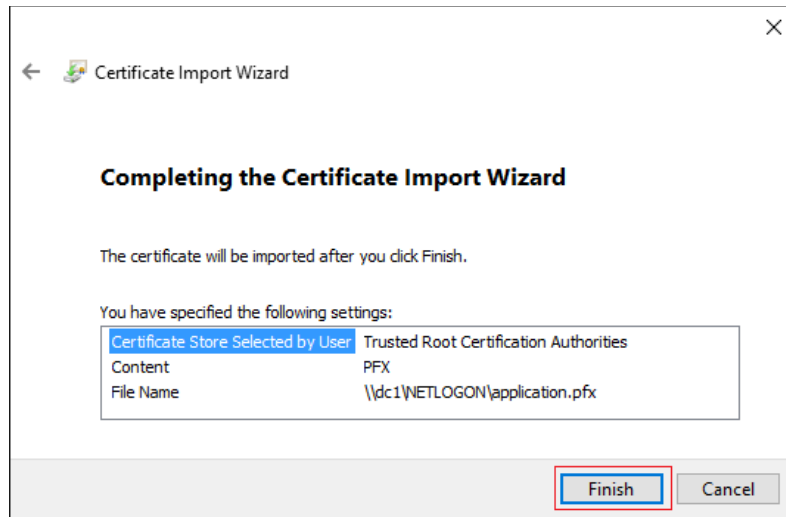
Windows can automatically select a certificate store, or you can specify a location for the certificate.

☐ Automatically select the certificate store based on the type of certificate
☒ Place all certificates in the following store

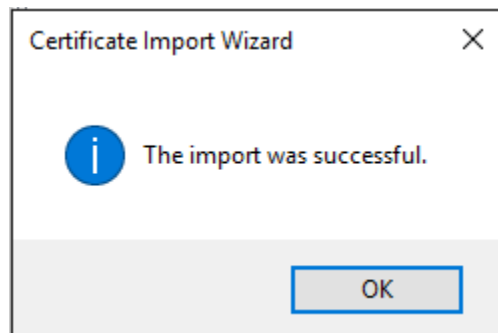
Certificate store:
Trusted Root Certification Authorities [Browse...]

Next Cancel

On the next window, verify everything is correct and click Finish.



Say OK to the confirmation screen.



How SSL certificates work

The client contacts the server. The server sends the client's browser a copy of its SSL certificate. **The client's browser checks whether it can trust the source of the certificate.** The browser will check for three things:

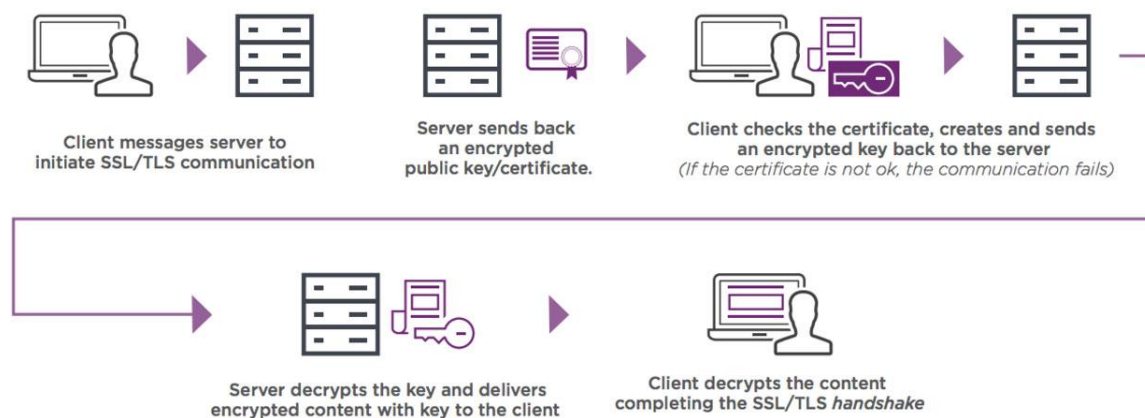
1. The certificate comes from a trusted party
2. The certificate is currently valid
3. The certificate has a relationship with the website.

If the certificate is accepted, the two machines initiate a secure session. The server sends the client a copy of its public key. Using the server's public key, the client generates and encrypts a

symmetric key and sends it back to the server. The two machines can now use symmetric-key encryption to encrypt and decrypt the traffic between them.

Once the session has ended, each machine will discard the symmetric key used for that session. Any additional sessions require a new symmetric key be created.

For this discussion, SSL and TLS are used to mean the same thing. SSL is the predecessor to TLS. Websites use either SSL, TLS or both but newer browsers such as Firefox will not always allow a secure connection unless the website is using a newer version of TLS.



Since our self-signed certificate was not present in our client's Trusted Root Certification Authority container, the client was prompted to either accept the risk or back out of the session.

Summary –

In the last three labs, we have learned how to create a self-signed certificate to be used internally in different ways. We also learned how to use Group Policy to place a self-signed certificate inside the correct certificate store of workstation so that the user's browser would see the that certificate did come from a trusted source.

Self-signed certs have many different uses **internally** — these include creating a self-signed SSL cert for an in-house Exchange or FTP server but there are any number of reasons a self-signed certificate makes more sense than purchasing one from a trusted source, especially if the need to strictly internal.

End of the lab!