SSL Solutions

Contents

SSL Certificate Solutions

This document discusses problems people have when using HTTPS and SSL certificates within Swivel:

- Handling invalid certificates
- Importing certificates from another machine

Further information on SSL certificates with appliances can be found in the SSL Certificate PINsafe Appliance How to Guide, and for non appliance installations see Tomcat 6 SSL.

Prerequisites

- Some of the solutions below require a tool to manipulate Java keystores. The one we recommend is Keystore Explorer.
- DNS name for the Swivel instance, usually the public IP address

Exporting the Certificate from the Appliance

This may be required where the certificate is site certificate and is to be used elsewhere. It is generally simpler to take a copy of the Appliance certificate store to your local machine and work on it there. For this, you can either use Webmin, or an application such as WinSCP. If you use WinSCP, use the same credentials as you normally do to connect to the Appliance console.

The file you need to retrieve is /home/swivel/.keystore. Make a copy of it on your local PC.

To extract the certificate, you will need Keystore Explorer, mentioned above. Install it, if you have not already done so, and run it. Click on "Open an existing KeyStore", and locate the keystore you have just downloaded. When asked for a password, it is "lockbox".

Locate the appliance certificate, which should be called "swivel" if you have created it using the CMI. Right-click on in and select Export, then "Export Certificate Chain". Accept the default settings on the next page, and Save the file with an extension of .cer.

Importing Server Certificates from Another Machine

This section refers to installing private/public key pair server certificates within Swivel. For details on trusting public key certificates, see above.

The Problem

Many customers already have a commercial certificate (e.g. a wildcard certificate) installed on a Windows server, and want to use the same certificate on a Swivel appliance.

Note: we recommend that you generate the Appliance SSL certificate using the CMI menus. You should only use this technique if you already have a suitable certificate, and do not want to spend more money on another one.

Prerequisites

It is essential for this solution that you have both the public and private keys for the certificate. The public key certificate returned by the certificate authority in response to a certificate signing request is not enough. Also, the certificate needs to be marked as exportable.

Solution - The Simpler Way

Export the certificate as a .pfx file

The certificate must be exported and the step below explains how to do this. If this step is carried out by another party then the certificate will need to be supplied, usually as a pfx file with a password.

Open the Certificate management plug-in as described earlier: Select Start, Run and enter "MMC". Click File -> "Add/Remove Snap-in". In the pop-up dialog, click Add, then select Certificates from the list. Click Add, and make sure you select Computer Account, then Local computer.

Navigate to Personal, then Certificates, and locate the certificate you wish to export. Right-click on it, then select "All Tasks", "Export". Make sure you select Yes to importing the private key. It is also recommended that you choose to include the certificates in the certification path.

You will be required to enter a password for this file. You will need to enter this when you import the certificate to the appliance later on. Note that older appliances had to use "lockbox" as a password, and entering anything else would cause problems. This is no longer an issue.

Transfer the PFX file to the appliance

You need to copy the PFX to your appliance. You can do this using WinSCP or any SCP/SFTP file transfer tool.

The credentials to connect to the appliance are the same as your login credentials for the CMI: username admin and your password: you should have been given an initial password, but it is recommended you change that.

The file must be copied to the appliance in the folder /backups/upload in order for it to be visible to the CMI.

Install the PFX file as the new certificate

- Log into the appliance CMI using your credentials.
- Select the Tomcat menu.
- Select the Certificates menu.
- Select the option "Import / Roll Back to Previous Keystore"
- Select the option "Import Keystore"
- Choose your certificate from the list shown
- Enter the password when prompted

You will be prompted to restart Tomcat before the new certificate is active. Select Yes if you are confident the certificate is correct. If you want to check it has been successfully installed first, select No and then go back to the Certificate menu and select View Keystore. There should be an entry corresponding to your certificate. Select that and review it, checking the owner, issuer, validity and entry type: this last should be PrivateKeyEntry.

Once you are happy the certificate is correct, you must restart Tomcat to make it active, assuming you didn't do this in the step above. Do this from the Tomcat menu.

Solution - The Old Way

This documentation is provided for backward compatibility. It won't normally be required for new appliances, unless the method above fails.

Export the certificate as a .pfx file

This is the same first step as above.

Import the PFX into KeyStore Explorer

Open Keystore Explorer and select "Open an existing Keystore". Select the PFX file you just exported.

Convert the keystore into a JKS keystore

NOTE: this is not necessary on a version 4.x appliance. It can import PFX keystores directly.

From the Tools menu, select "Change Type", then "JKS". Accept any warnings displayed.

Warning: despite exporting the certificate with a password of "lockbox" in the above steps, this password will be lost when you do the conversion. So you need to change the password again within Keystore Explorer, in two places - the keypair and the keystore. First, right click the keypair entry in the Keystore Explorer window and set the keypair password to be lockbox. Then go to Tools and Set the keystore password to be lockbox.

Warning: if you are converting a wildcard certificate, the certificate alias (the name displayed) will probably contain a "*" character. This doesn't affect Tomcat's ability to use the certificate, but it does cause problems if you want to view the certificate using the appliance CMI later. Since the alias is just a name used to identify the certificate, it doesn't have to be the same as the certificate subject. You can rename the certificate by right-clicking on it and selecting "Rename". You will be asked for the password again before entering the new name.

After you have made any necessary changes, save the file with a different name. It is recommended you use an extension of either .jks or .keystore.

An alternative method of importing PFX files

If the above method fails to import the private key, which has been observed on occasions, try the following, slightly modified method instead:

- Create a new keystore of type "JKS".
- From the Tools menu, select "Import Key Pair".
- Select type "PKCS #12".
- Enter the password and PFX file name.

- Enter a name for the certificate (see the notes above).
- Enter a new password

Continue as above.

Copy the keystore on to the Swivel Appliance

You can do this with Webmin, but we recommend using WinSCP or similar, as it has more file management options.

NOTE: any changes you make will not take effect until you restart Tomcat.

Firstly, make a backup copy of the current Tomcat keystore, which is /home/swivel/.keystore. We recommend that you keep a copy of this file off the Appliance anyway, for recovery purposes, but in this particular instance, we also recommend that you rename .keystore to something else so that you can restore it quickly if anything goes wrong.

Now copy the new keystore you have just created into the /home/swivel folder, and rename it to .keystore.

Check the owner of the file, (using right-click, Properties in WinSCP). If the owner is not swivel, then change both owner and group to swivel.

Restart Tomcat and check that you can access the Swivel admin console. If you can't, check the file catalina.out for errors.

Managing Certificate Errors

Symptoms

At best, certificate errors can cause warning messages to be displayed in your browser. At worst, they will stop https requests from working at all. This is particularly a problem with embedded TURing images where the hostname is not the same as the main page, or where a server needs to make requests to Swivel behind the scenes, and it is not possible to configure the server to ignore these errors.

There are essentially three ways in which SSL certificates can be invalid:

- The certificate has expired
- The certificate hostname does not match the request hostname
- The certificate is not issued by a trusted authority

Expired Certificates

There is nothing that can be done about expired certificates, except to renew the certificate. See the Swivel Appliance guide for details on renewing a Swivel certificate.

Incorrect Hostname

There is little that can be done about this: the hostname within the request **must** match the hostname in the certificate. However, there are some techniques you should be aware of that can help to ensure this.

First of all, always use hostnames when specifying HTTPS requests, never IP addresses. You can get away with IP addresses when using certain Swivel tools that allow you to ignore certificate requests, but when accessing direct from the browser, or when making requests with built-in tools, you should always use the hostname specified by the SSL certificate.

One problem that can arise is that the certificate hostname cannot be resolved, or resolves to an external IP address when an internal one is required. This can be overcome by editing the hosts file on the Swivel Agent. On Windows machines, this file is located in C:\Windows\System32\drivers\etc. It is a simple text file, and all you need to do add a new line consisting of the target IP address, followed by one or more spaces or tabs, followed by the hostname. Note that this solution will affect **all** references to this hostname from this machine, so make sure it is not required to access the external IP address by the same hostname, if relevant.

Untrusted Signing Certificate

This is a problem that can be resolved relatively easily in a number of ways.

This simplest but most expensive solution is to buy a commercial certificate from a recognised authority. This authority should be included in your list of trusted root certificates already.

You can also sign your own certificates. The Swivel appliance has an option to generate a self-signed certificate, which is fine for a one-off solution. You can also set up your own certificate signing authority. Windows Server operating systems include a Certificate Services option, which allows you to accept and sign certificates.

The problem with signing your own certificates is that they are not trusted by default. However, you can add any certificate to the trusted root certificates store on your computer. All you need is a copy of the signing authority public key certificate. For a Swivel appliance, getting the certificate is described below. For Windows Certificate Services, you can download a copy of the root CA certificate.

To install certificates on a Windows machine, select Start, Run and enter "MMC". Click File -> "Add/Remove Snap-in". In the pop-up dialog, click Add, then select Certificates from the list. Click Add, and make sure you select Computer Account, then Local computer. Close all dialogs, and expand the Certificates tree to "Trusted Root Certification Authorities", then "Certificates". Right-click and select All Tasks -> Import. Select the root certificate you have already retrieved, and install it.

Troubleshooting

Wildcard certificate import problem checks

Does the certificate have both a private and public key?

Are the keystore and certificate passwords the same? If the password is not the Swivel default then you need to change that in the Tomcat server.xml file.

Are the intermediate and root certificates included in the keystore file.

Is the file named ".keystore" when you copied it to the appliance?

Check the permissions and that the owner is "swivel"?