Lab 14 – Configuring SSL for WebSphere

At the end of this exercise, you should be able to:

- Define the certificate life span of a profile
- Use the administrative console to find and view certificates within the cell
- Configure and run the certificate expiration service
- Propagate the generated plug-in keystore out to the plug-in
- Create a keystore for a web server
- Generate a self-signed key
- Configure IBM HTTP Server to load and use HTTPS

Section 1: Create a backup

Since this exercise changes the existing environment, which, if done incorrectly, can cause problems for the rest of the exercises, creating a backup is a good idea.

Section 2: Create a profile

To better understand the various pieces of SSL within the WebSphere Application Server environment, a new custom profile is created.

Section 3: Examine the node certificates

This new node has a couple of certificates that are associated with it. This section of the exercise uses the administrative console to examine them.

Section 4: Examine certificate expiration and updating

Since the personal certificates have a life span of only one year, administrators must be aware that these certificates expire. Fortunately, WebSphere has a built-in mechanism to automatically renew these certificates when they are about to expire. And, since the signer certificates remain the same, it is not necessary to propagate anything new to the remote nodes or plug-in.

Section 5: Plug-in key ring propagation

Not only are the processes within cells (deployment managers, node agents, and application servers) required to have certificates and know about other signer certificates, so are the web server plug-ins. To secure the communication between the web server plug-ins and the application servers, the plug-ins and application servers must be able to negotiate an SSL session. They must have personal certificates (by default the application servers use the node personal certificate) and have access to the other signer certificates.

WebSphere is able to make sure that all of the required certificates are available to the web server plug-in by creating the plug-in keystores from within WebSphere. By doing so, WebSphere can make sure that not only does the plug-in have a valid personal certificate, but it also has the necessary cell root signer certificate. At the same time, WebSphere can ensure that the plug-in signer certificate is also available in the cell truststore.

The real problem with this approach is that after WebSphere generates the plug-in keystore, it still must be propagated to the host that is running the web server. The propagation process of plug-in keystores is similar to the propagation of the `plugin-cfg.xml` file. It is usually done manually, but in some cases can be configured to be done automatically (usually not desirable).

Section 6: Configuring SSL for IBM HTTP Server (optional)

This part of the exercise examines the process of creating a certificate and a key ring for the web server. The steps are used to configure SSL on the connection between the client browser and the web server.

Section 7: Testing the SSL connection