# DevNote #106 – Configuring HTTPS in a Windows Service with the SIFWorks® ADK for .Net

The Schools Interoperability Framework 1.1 and higher specifications require that all agents be able to communicate with zone integration servers over the HTTPS transport protocol. The SIFWorks® Agent Developer Kit (ADK™) for .Net offers both HTTP and HTTPS implementations that are easily selected by changing an agent's properties at runtime. Configuring an agent based on the .Net ADK to use HTTPS is described in *DevNote #105 –Testing HTTPS with the SIFWorks® ADK for .Net*.

This technote presumes the information provided in that document. This technote also presumes use of the configuration functionality provided by the `AgentConfig` class in the ADK. If you are configuring an agent that does not make use of `AgentConfig`, the configuration should be done using the appropriate properties available in the `HttpsProperties` class.

When the Windows certificate store is accessed by an ADK gent that is running as a service, it is no longer connecting to the certificate store owned by the interactive user. The certificate store that it connects to is based upon the user credentials that the service is running under. This devnote will describe a number of different ways that an agent based on the ADK can be configured to use HTTPS while running as a service.

## 1. Differences from DevNote #105

### 1.1. Configuring the agent to trust the ZIS's certificate

When configuring the agent to trust the ZIS certificate, you need to ensure that the ZIS's certificate gets installed into the "Certificates (Local Computer)\Trusted Root Certification Authorities" store. In DevNote #105, the instructions say to let Windows automatically choose the certificate store, but if the agent is running as a service, the certificate needs to go into the local computer's certificate store. You can override the default location by clicking the "Show Physical Stores" checkbox after clicking "Install Certificate" and choosing "Trusted Root Certification Authorities\Local Computer" as the destination. You can also do this manually by copying and pasting it in Windows Certificate manager, or exporting the ZIS's certificate as a .CER file and importing it into the Certificates (Local Computer)\Trusted Root Certification Authorities store in the Windows certificate manager.

### 1.2. Configuring the ZIS to trust the agent's certificate

When configuring the agent to use client certificates, you need to configure the ZIS to trust the agent's root signing certificate, not the certificate the agent uses for SSL. In DevNote #105, this is the certificate called "MyCA", which is stored in the Trusted Root Certification Authorities node.

## 2. Configuring HTTPS using a certificate file

The easiest way to configure an agent running as a service to use HTTPS is to use a certificate file. When using a certificate file, the agent will be able to easily find and use its HTTPS certificate, no matter which credentials it is running under. The certicate that it uses must contain a private key and stored in the PFX/P12 format. Certificates of this type have a ".pfx" extension. Certificate with a ".cer" extension do not contain a private key and cannot be used for server authentication.

Once a certificate has been created for HTTPS, such as the one created for the GetZoneStatus agent as described in DevNote #105, it can be exported in the PFX/P12 format. To export the certificate, follow the steps below.

Copyright ©2005 Edustructures LLC. All Rights Reserved.

1. Locate it in the Windows certificate store using the certificate manager.

2. Right-click on the certificate, choose "All Tasks -> Export".

3. You must export the private key. Do not choose "Enable Strong Key Protection". Choose the password for the private key and export the certificate file to the directory that the agent is running from.

4. In the agent's configuration file, modify the HTTPS properties as follows.

5. Locate the <transport> node with the protocol attribute set to "https".

6. Add a property named "sslCertFile". Set the value to the filename you exported in step 3.

7. Add a property named "sslCertFilePassword". Set the value to the password you choose for the private key in step 3.

8. Ensure that the "enabled" property for the https transport node is set to "true" and the corresponding "enabled" property for the http transport is set to false.

When you are done, you should have a set of properties that look like this.

```
  <transport enabled="true" protocol="https">
    <property name="port" value="8443" />
    <property name="pushHost" value="localhost" />
    <property name="sslCertFile" value="localhost.pfx" />
    <property name="sslCertFilePassword" value="changeit" />
  </transport>
  <transport enabled="false" protocol="http">
    <property name="port" value="8080" />
</transport>
```

## 3. Configuring HTTPS in a Windows service running as SYSTEM

If the service that the agent is running in is configured to run under the credentials of the built-in System account, the agent can be configured to read the certificate from the LocalMachine certificate store. In order for this to work, the certificate must be imported into the certificate store representing the local computer. This can be done by using the certificate manager. Create a certificate manager console, as described in DevNote #105. Add certificate snap-ins for both the current user account and the local computer account.

After you have certificate manager open, locate the certicate in the Certificates – Current User\Personal\Certificates node. Right-Click it and choose "Copy". You can then locate the Certificates (Local Computer)\Personal\Certificates node and "Paste" it there.

Once the certificates have been added to the local computer's certificate store, you must tell the agent to look there for the certificates. To do so, add a property to the https transport node with a name of "certStoreLocation" and a value of "LocalMachine". This will work for any service that is running under the SYSTEM account. If the account is running under a different account, the account will not have permissions to read certificate entries from the "LocalMachine" certificate store.

## 4. Configuring HTTPS in a Windows service running under a specific user account

If the service that the agent is running in is configured to run under another user account, it can be configured to use certificates from that user account's Windows certificate store. The easiest way to do this is to log on to the system using the same user account that the service is running under. Once you are logged on, you can create a self-signed certificate using the instructions in DevNote #105, or you can import a certificate from a 3rd party certificate authority. Once you have done so, the agent service should be able to find and use the certificate.

Copyright ©2005 Edustructures LLC. All Rights Reserved.

## 5. Troubleshooting

If you have problems configuring the agent to connect using https, make sure you have logging turned all the way up and examine the agent's log after you try to connect to a zone. Most common problems with HTTPS configuration are explicitly logged to the agent's log file.

Copyright ©2005 Edustructures LLC. All Rights Reserved.