Upgrading Older ISAPI Configuration

- Installation
- Upgrading from ISAPI to the New Module
- Deconfiguring the Old Plugin
- Configuring the New Plugin
- Configuration and Behavioral Changes

**Installation**

The V3 SP installer no longer provides the option to install the old-style ISAPI filter, but on a machine with V2 previously installed, the old `isapi_shib.dll` plugin is upgraded and left in place.

**No 32-bit shibd needed on 64-bit OS**

The 32-bit web server modules can function with a 64-bit shibd service, so there is no need to install a 32-bit shibd to handle 32-bit app pools.

**Upgrading from ISAPI to the New Module**

In order to configure the new IIS7 plugin, you must "deconfigure" the ISAPI filter and extension and configure the new module.

**Deconfiguring the Old Plugin**

Before making any configuration changes to IIS, backup up your system appropriately. Although the overlay installer undoes its own configuration it will not revert any configuration that you change.

- In the GUI, select the server in the left-hand pane and click on the "ISAPI Filters" in the right.
- Now remove any Shibboleth-related filters. Depending on the IIS version, for a default install this will be called Shibboleth or ShibFilter and be for executable: `C:\opt\shibboleth-sp\lib64\shibboleth\isapi_shib.dll`
- In the GUI, select the server in the left-hand pane and click on the "Handler Mappings" in the Right.
- Now remove any Shibboleth-related ISAPI handler mapping (generally for path `*.sso`). Depending on the IIS version, for a default install this will be called something like AboMapperCustom-8689 and be for executable: `C:\opt\shibboleth-sp\lib64\shibboleth\isapi_shib.dll`
- In the GUI, select the server in the left-hand pane and click on the "ISAPI and CGI Restrictions" in the Right.
- Now remove any Shibboleth-related permission grant. For a default install this will be called something like Shibboleth Web Service Extension and be for executable: `C:\opt\shibboleth-sp\lib64\shibboleth\isapi_shib.dll`
- Edit `%SYSTEMROOT%\System32\InetSrv\config\applicationHost.config` and look for the words "shib" or "shib_isapi" and remove the elements containing these words.
- As well as the `applicationHost.config` file, you may need to inspect the `web.config` files for the sites and their subfolders.

**Configuring the New Plugin**

This is best done from the command line (you will also need Admin privileges). From the `C:\Windows\System32\InetSrv` directory:

```
appcmd install module /name:ShibNative32 /image:"c:\opt\shibboleth-sp\lib\shibboleth\iis7_shib.dll" /precondition:bitness32
appcmd install module /name:ShibNative /image:"c:\opt\shibboleth-sp\lib64\shibboleth\iis7_shib.dll" /precondition:bitness64
```

(Obviously on a 32-bit machine only the first line is required).

There should be no immediate need to make any changes to the `shibboleth2.xml` file, nor even to restart shibd, but it is best to reboot at this stage (or if the configuration ever seems to "stick"). Retest, bearing in mind that an unchanged configuration will not be setting headers but will be setting variables.

If the ISAPI module is still configured the following tell-tales will indicate this:

1. The event viewer ("Windows Logs\Application") may show a warning.
2. An attempt to access a protected resource will return a failure (status 500) and the native log output will have the following line:

```
ERROR Shibboleth.NATIVE [<pid>] native_shib: Shibboleth handler invoked at an unconfigured location.
```
This indicates that configuration for the ISAPI filter is still active somewhere.

**Configuration and Behavioral Changes**

The IIS7 plugin is fully defined in the `<ISAPI>` documentation; the following changes should be noted:

- If a `<Site>` element is not specified for a site which uses the plugin, then the module will ignore requests to that site, as with the old plugin.
- When the `<Site>` element is specified, the following new settings can be provided:
  - `useVariables=boolean` (default `true`) controls whether attributes are passed to the application as *Server Variables*.
  - `useHeaders=boolean` (default `false`) controls whether attributes are passed to the application as *HTTP Headers*. This setting should be avoided but is present to provide a level of compatibility with applications developed against the old isapi_shib plugin. You should move aggressively to fix this.
- `REMOTE_USER` will (usually) be populated in the manner one expects and that is familiar to the use of Shibboleth on Apache. This was not possible with the old plugin, which populated a weird and dangerous header (HTTP_REMOTE_USER) as a workaround. The new module does not populate that header regardless of the settings used, so this may impact applications immediately.
- Additionally, a new element `<Roles>` may be specified. This configures the roles that can be used in native *Roles Based Authorization*.

<table>
<thead>
<tr>
<th>Example</th>
</tr>
</thead>
</table>
|```xml
<ISAPI normalizeRequest="true" safeHeaderNames="true">
  <Roles roleAttributes="ePa ePsa" />
  <Site id="1" name="iis.example.org"/>
</ISAPI>
``` |

The first site will be given the host name `iis.example.org`