SPiPlanetConfiguration

The NSAPI module for Netscape/iPlanet/Sun servers will only work on platforms where the web server has been linked to the same C++ library as the Shibboleth code. Currently this includes Windows and possibly Solaris, but not Linux.

The examples below are for Solaris. Paths and filenames are different on Windows (e.g. the module filename is `nsapi_shib.dll`).

First install the module with the following lines in `magnus.conf`:

```
Init fn="load-modules" funcs="nsapi_shib_init,nsapi_shib,shib_handler" shlib="/opt/shibboleth-sp/libexec
/nsapi_shib.so"
Init fn="nsapi_shib_init" server-name="sp.example.org"
```

If `server-name` is not set, it may be possible to guess the right value but only if the box knows the correct name, so setting it is best.

You may want to try restarting the server to make sure that it still loads properly, in case any library path or SP configuration problems occur. Once that works, the filter is loaded and running, just not active yet.

The next step may not be necessary, but for consistency, it's suggested. Modify `mime.types` and add this mapping for the Shibboleth handler inside the filter:

```
type=magnus-internal/shibboleth   exts=sso
```

The next step is to install the handler in `obj.conf`. There are multiple ways to do it, but one standard configuration looks like:

```
<Object name="default">
...
NameTrans fn="assign-name" from="/Shibboleth.sso/*" name="shib-handler"
...
</Object>

<Object name="shib-handler">
ObjectType fn=force-type type=magnus-internal/shibboleth
</Object>
```

At that point, the server should respond to requests to /Shibboleth.sso/Foo with Shibboleth error messages, indicating the handler is answering.

Finally, you can deal with the content. This can be done using ShibbolethXml in the RequestMap element, as with Apache and IIS, or you can use commands inside `obj.conf`. Many different approaches can be used there. One way is to use logical object naming:

```
<Object name="default">
...
NameTrans fn="assign-name" name="secure" from="/secure/*"
...
</Object>

<Object name="secure">
AuthTrans fn="nsapi_shib" requireSession="1"
</Object>
```

Another way is with physical path matching:

```
<Object ppath="$docroot/secure/*">
AuthTrans fn="nsapi_shib" requireSession="1"
</Object>
```

While any RequestMap attribute can be set on the AuthTrans command line, generally you'll find using the RequestMap simpler and more dynamic, because `obj.conf` is static and doesn't give you the same features an .htaccess file would.

A note about vhosting: some code is in the filter that should properly derive the vhost name from the virtual server configuration, but it's untested. If you're vhosting, you'll want to do some experimenting to make sure it's working correctly.