Shibboleth Installation notes

Soffid Federation is based on shibboleth open source project. Actually the installation is a mixed procedure between Shibboleth installation and Soffid configuration. In the future Shibboleth installation will be integrated on Soffid installation in order to assume better integration level.

This guides help administrators to streamline shibboleth installation process, but it does not replace the oficial shibboleth documentation in any way.

Install shibboleth

On ubuntu

```
sudo apt-get install shibboleth-sp2-schemas libshibsp-dev
sudo apt-get install libshibsp-doc libapache2-mod-shib2 opensaml2-tools
sudo apt-get install libapr-memcache-dev libapr-memcache0 policycoreutils
```

On RedHat

Follow Installing via Yum instructions on shibboleth wiki:

https://wiki.shibboleth.net/confluence/display/SHIB2/NativeSPLinuxRPMInstall

On Windows Server

Follow installing via Windows Server instructions on Shibboleth wiki:

https://wiki.shibboleth.net/confluence/display/SHIB2/NativeSPWindowsInstall

Configure SELinux (if needed)

create shibd.te file with this content:

```
module httpd_shibd 1.0;
require {
```

```
        type tmp_t;
        type var_run_t;
        type httpd_t;
        type initrc_t;
        class sock_file write;
        class unix_stream_socket connectto;
}
#============= httpd_t ==============
allow httpd_t initrc_t:unix_stream_socket connectto;
allow httpd_t var_run_t:sock_file write;
```

Execute

```
sudo checkmodule -M -m -o shibd.mod shibd.te
sudo semodule_package -o shibd.pp -m shibd.mod
sudo semodule -i shibd.pp
sudo setsebool -P httpd_can_network_connect 1
```

Create service provider Shibboleth keys & metadata

Execute

```
sudo shib-keygen -h HOSTNAME -e https://HOSTNAME/shibboleth
```

Verify the permissions of the generated key.

At this point, verify the hostname specified matches the ServerName directive at Apache config file, including scheme and port.

Edit configuration file

Update shibboleth2.xml in order to download the federation data from Soffid master or backup Synchronization Server. You will need to specify the Identity Provider public ID, as it is created on Soffid SAML Federation

```
<ApplicationDefaults entityID="https://HOSTNAME/shibboleth"
                     REMOTE_USER="eppn persistent-id targeted-id">
...
```

```
    <Sessions>
        <SSO entityID="SOFFID-IDP-public ID">
          SAML2 SAML1
        </SSO>
        <Logout>SAML2 Local</Logout>
...
    </Sessions>
...
    <MetadataProvider type="XML" uri="https://SYNCSERVER:760/SAML/metadata.xml" handlerSSL="true"
            backingFilePath="federation-metadata.xml" reloadInterval="7200">
    </MetadataProvider>
...
</ApplicationDefaults>
```

Finally, uncomment the required attributes on attribute-map.xml. You must also add the following ones:

```
    <Attribute name="urn:oid:0.9.2342.19200300.100.1.1" id="uid"/>
    <Attribute name="urn:oid:1.3.6.1.4.1.22896.3.1.1" id="sessionId"/>
    <Attribute name="urn:oid:1.3.6.1.4.1.22896.3.1.2" id="soffidSecrets"/>
    <Attribute name="urn:oid:1.3.6.1.4.1.22896.3.1.4" id="userType"/>
    <Attribute name="urn:oid:1.3.6.1.4.1.22896.3.1.5" id="givenNames"/>
```

Enable Single Logout back-channel

It's advisable to use single logout back-channel while using non SAML-aware applications.

To do this, add a new Logout intitator configuration at shibboleth2.xml file:

```
 <!-- LogoutInitiators enable SP-initiated local or global/single logout of sessions. -->
          <LogoutInitiator type="Chaining" Location="/Logout">
            <LogoutInitiator type="SAML2"
                  template="bindingTemplate.html"/>
            <LogoutInitiator type="Local"/>
          </LogoutInitiator>
          <!-- Logout inititator to be used by WebSSO -->
          <LogoutInitiator type="Chaining" Location="/SOAPLogout">
            <LogoutInitiator type="SAML2"
                  outgoingBindings="urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
                  template="bindingTemplate.html"/>
```

```
        <LogoutInitiator type="Local"/>
      </LogoutInitiator>


      <!-- md:SingleLogoutService locations handle single logout (SLO) protocol messages. -->
      <md:SingleLogoutService Location="/SLO/SOAP"
        Binding="urn:oasis:names:tc:SAML:2.0:bindings:SOAP"/>
      <md:SingleLogoutService Location="/SLO/Redirect" conf:template="bindingTemplate.html"
        Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect"/>
      <md:SingleLogoutService Location="/SLO/POST" conf:template="bindingTemplate.html"
        Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"/>
      <md:SingleLogoutService Location="/SLO/Artifact" conf:template="bindingTemplate.html"
        Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact"/>
```

For security reasons, you should at the signing parameter at the application defaults tag in order to enable logout message signature:

```
<ApplicationDefaults entityID="..."
                     signing="true"
                     REMOTE_USER="eppn persistent-id targeted-id">
..
```

Finally

Restart services:

```
sudo service apache2 start
sudo service shibd start
```

Revision #5
Created 7 June 2022 13:26:56
Updated 17 October 2024 09:26:52