Fetching Metadata and Attribute Filter and caching them locally

This documents an alternative solution to managing refreshes to the metadata and attribute filter files. The easier solution recommended in the installation manual is to let the IdP manage the refreshes.

This implementation is based on using an external script (`fetch-xml.sh`). This script loads the XML file (over an HTTPS connection), checks the XML document for well-formedness, optionally verifies the signature on the downloaded XML document - and if all tests are passed, replaces the original file with a single "mv".

The script takes three arguments: the remote URL, the local file name, and an email address to send any errors to (no email sent if everything goes well).

An extra optional step (documented below) is to install XmlSecTool for verifying the signature. Otherwise, downloading the file over HTTPS and checking the XML structure provides also reasonable guarantees. If using XmlSecTool, the script takes a fourth argument, the certificate to check the signature with. And in this case, XmlSecTool must be found either in the `PATH` or in the `XMLSECTOOL` environment variable.

To deploy this solution without XmlSecTool:

- Download the fetch-xml.sh script into `/opt/shibboleth-idp/bin`

```
wget -O /opt/shibboleth-idp/bin/fetch-xml.sh https://github.com/REANNZ/Tuakiri-public/raw/master/scripts
/fetch-xml.sh
chmod +x /opt/shibboleth-idp/bin/fetch-xml.sh
```

- Determine the URLs you will be loading the files (metadata and attribute filter) from and locations you will be putting them into - same as in the standard implementation above.
- Download the metadata signing certificate into `$IDP_HOME/credentials`:

```
wget http://directory.tuakiri.ac.nz/metadata/tuakiri-metadata-cert.pem -O $IDP_HOME/credentials/tuakiri-
metadata-cert.pem
```

- Invoke `fetch-xml.sh` once to download the metadata:

```
/opt/shibboleth-idp/bin/fetch-xml.sh https://directory.tuakiri.ac.nz/metadata/tuakiri-metadata-signed.
xml /opt/shibboleth-idp/metadata/tuakiri-metadata-signed.xml errors@institution.domain.ac.nz
```

- Invoke `fetch-xml.sh` once to download the attribute filter for your IdP (note that you have to request one to be published, same as in the standard implementation above):

```
/opt/shibboleth-idp/bin/fetch-xml.sh http://directory.tuakiri.ac.nz/attribute-filter/institution.domain.
ac.nz.xml /opt/shibboleth-idp/conf/tuakiri-attribute-filter.xml errors@institution.domain.ac.nz
```

- Configure a 3.x IdP to load the Tuakiri metadata and atribute filter files:
  - Configure the IdP to load the Federation Metadata in `/opt/shibboleth-idp/conf/metadata-providers.xml` by adding the following snippet into the `ChainingMetadataProvider`.

```
    <MetadataProvider id="TuakiriMetadata"
                      xsi:type="FilesystemMetadataProvider"
                refreshDelayFactor="0.125"
                maxRefreshDelay="PT2H"
                metadataFile="%{idp.home}/metadata/tuakiri-metadata.xml">

            <MetadataFilter xsi:type="SignatureValidation"
                    certificateFile="${idp.home}/credentials/tuakiri-metadata-cert.pem"
                    requireSignedRoot="false">
            </MetadataFilter>
            <MetadataFilter xsi:type="EntityRoleWhiteList">
                    <RetainedRole>md:SPSSODescriptor</RetainedRole>
            </MetadataFilter>

    </MetadataProvider>
```

        - Please see the notes in the main instructions for Configuring an IdP to load the Tuakiri metadata for additional information about the parameters in this snippet.

    - Edit `$IDP_HOME/conf/services.xml` and add the additional attribute filter as an additional value in the `shibboleth.AttributeFilterResources` `util:list` bean:

```
<value>%{idp.home}/conf/tuakiri-attribute-filter.xml</value>
```

- For archival purposes, we also keep the original instructions for configuring the Tuakiri metadata and attribute filter on a 2.x IdP - unfold the box below to see the IdP 2.x compatible syntax:

    - Load the metadata from the local file: add the following into `$IDP_HOME/conf/relying-party.xml` (the variation from the standard implementation above is using a `FilesystemResource` instead of a `FileBackedHttpResource`)
        - Add the following snippet into the `ChainingMetadataProvider`:

```
        <!-- Tuakiri -->
        <metadata:MetadataProvider id="Tuakiri" xsi:type="metadata:
ResourceBackedMetadataProvider">
          <metadata:MetadataFilter xsi:type="metadata:ChainingFilter" xmlns="urn:mace:
shibboleth:2.0:metadata">
            <metadata:MetadataFilter xsi:type="metadata:SignatureValidation" xmlns="urn:
mace:shibboleth:2.0:metadata"
                                trustEngineRef="shibboleth.MetadataTrustEngine"
                                requireSignedMetadata="true" />
          </metadata:MetadataFilter>
          <metadata:MetadataResource xsi:type="resource:FilesystemResource" file="/opt
/shibboleth-idp/metadata/tuakiri-metadata.xml" />
        </metadata:MetadataProvider>
```

        - Same as in the standard implementation, uncomment the `<security:TrustEngine id="shibboleth.MetadataTrustEngine" xsi:type="security:StaticExplicitKeySignature">` element if it is still commented out and add in this snippet to load the metadata signing certificate

```
        <security:Credential id="Tuakiri-FederationCredentials" xsi:type="security:
X509Filesystem">
            <security:Certificate>/opt/shibboleth-idp/credentials/tuakiri-metadata-cert.
pem</security:Certificate>
        </security:Credential>
```

    - Load the attribute filter from a local file: Add the following entry into `<srv:Service id="shibboleth.AttributeFilterEngine"` in `$IDP_HOME/conf/service.xml`:

```
        <srv:ConfigurationResource file="/opt/shibboleth-idp/conf/tuakiri-attribute-filter.
xml" xsi:type="resource:FilesystemResource" />
```

- Create a cron job to periodically (every 2 hours) download the metadata and the attribute filter: run `crontab -e` and add the following entry (matching the command you had run on the command line earlier):

```
02 */2 * * * /opt/shibboleth-idp/bin/fetch-xml.sh https://directory.tuakiri.ac.nz/metadata/tuakiri-
metadata-signed.xml /opt/shibboleth-idp/metadata/tuakiri-metadata.xml errors@institution.domain.ac.nz
02 */2 * * * /opt/shibboleth-idp/bin/fetch-xml.sh https://directory.tuakiri.ac.nz/attribute-filter
/institution.domain.ac.nz.xml /opt/shibboleth-idp/conf/tuakiri-attribute-filter.xml errors@institution.
domain.ac.nz
```

Optional: Installing XmlSecTool

- Download latest version (2.0.0 as of July 2016) from http://www.shibboleth.net/downloads/tools/xmlsectool/ into `~/inst`
    - Unzip into `/opt/xmlsectool-$XMLSECTOOL_VERSION`
    - Symlink as `/opt/xmlsectool`

```
export XMLSECTOOL_VERSION="2.0.0"
wget -P ~/inst/ http://www.shibboleth.net/downloads/tools/xmlsectool/$XMLSECTOOL_VERSION
/xmlsectool-$XMLSECTOOL_VERSION-bin.zip
cd /opt
unzip ~/inst/xmlsectool-$XMLSECTOOL_VERSION-bin.zip
ln -s xmlsectool-$XMLSECTOOL_VERSION xmlsectool
```

- Set JAVA_HOME to your Java installation:

```
export JAVA_HOME=/usr/lib/jvm/java
```

- Invoke as **`/opt/xmlsectool/xmlsectool.sh`**
- Modify `fetch-xml.sh`cron jobs to use XmlSecTool to verify signature:
  - Add `/opt/shibboleth-idp/credentials/tuakiri-metadata-cert.pem` as an additional argument (the certificate to verify signatures with)
  - Prefix the commands with environment variable settings to tell the script where to find XmlSecTool and tell XmlSecTool where to find Java: `JAVA_HOME=/usr/lib/jvm/java XMLSECTOOL=/opt/xmlsectool/xmlsectool.sh`
  - The final form of the cron jobs is:

```
02 */2 * * * JAVA_HOME=/usr/lib/jvm/java XMLSECTOOL=/opt/xmlsectool/xmlsectool.sh /opt/shibboleth-
idp/bin/fetch-xml.sh https://directory.tuakiri.ac.nz/metadata/tuakiri-metadata-signed.xml /opt
/shibboleth-idp/metadata/tuakiri-metadata.xml errors@institution.domain.ac.nz /opt/shibboleth-idp
/credentials/tuakiri-metadata-cert.pem
02 */2 * * * JAVA_HOME=/usr/lib/jvm/java XMLSECTOOL=/opt/xmlsectool/xmlsectool.sh /opt/shibboleth-
idp/bin/fetch-xml.sh http://directory.tuakiri.ac.nz/attribute-filter/institution.domain.ac.nz.xml
/opt/shibboleth-idp/conf/tuakiri-attribute-filter.xml errors@institution.domain.ac.nz /opt
/shibboleth-idp/credentials/tuakiri-metadata-cert.pem
```