Use of Shibboleth UI with Shibboleth IdP 3.4.x

One assumption is that you will put in place a way to transfer files produced by the Shibboleth UI into your Shibboleth IdP instance(s). beyond that, this page discusses the variety of configuration changes one needs to make to one's Shibboleth IdP 3.4.x to take advantage of the metadata-related files produced by the Shibboleth UI.

Deployment and Assumptions

The Shibboleth UI has been built to be deployed as its own web app. It would run in the same servlet engine as the IdP, or it could run in a separate one. The key link is that files generated by the UI need to end up making it into the IdP's configuration (conf/) directory, however you want to make that happen. That could be a NFS/SMB file sharing approach, it could be rsync or something similar, or you could put the files into a repo and then pull out of the repo into the IdP. Details on the deployment of the ShibUI can be found in the Bitbucket repo README.

There are some basic assumptions that the ShibUI makes about your Shibboleth IdP deployment, that you must do in order for the "full magic" of this ShibUI to have an impact. Some of those assumptions to date (thru MVP2) are:

- You will be running Shibboleth IdP v3.4 or later, because all the relying party overrides take advantage of the MetadataDrivenConfiguration that version 3.4 will support.
- You will have turned on support for the MetadataDrivenConfiguration by modifying your IdP's *conf/relying-party.xml* file to enable "tag-driven configuration" for your relying party defaults, overrides, and profile configurations as per the Relying Party Configuration section of that MetadataDrivenConfiguration wiki page.
- That you will create a sub-directory "generated" in your IdP's metadata/ directory (i.e. *metadata/generated)* into which all the "individual SP metadata files managed thru the ShibUI" will be written into, and that you will modify your *conf/metadata-providers.xml* file to define that directory as a LocalDynamicMetadataProvider directory. That would mean adding config such as the following to your *conf/metadata-providers.xml* file. We still need to consider recommended settings around the caching of that metadata, so that changes you make to files in that directory become effective as "quickly as you would like". At a minimum, one might want to adjust the **minCacheDuration** (defaults to 10 minutes) and the **maxCacheDuration** (defaults to 8 hours) to ensure changes are checked for more quickly. Note, though, that Scott Cantor indicates he is looking into adjusting the "smarts" of this connector in picking up on changes automatically, and changing how the caching works a bit, and that those changes might also come out in version 3.4.
    - ```
      <MetadataProvider id="localDynamicMetadata" xsi:type="LocalDynamicMetadataProvider"
      sourceDirectory="%{idp.home}/metadata/generated" />
      ```
- The latest additions to the ShibUI allow one to create entity attribute metadata filters that can target specific metadata entries in the InCommon aggregate. As part of supporting that, the ShibUI will output its own special *metadata-providers.xml* file that includes the config for reading in the InCommon aggregate, along with the filters that one has created thru the Shibboleth UI. That "UI-controlled *metadata-providers.xml* file" will need to make it into your IdP, and the Shibboleth IdP will need to be configured to use both its standard *conf/metadata-providers.xml* file and this 2nd one (by modifying its *conf/services.xml* file.)
- You will have added into your primary *conf/attribute-filter.xml* file, or have configured in a 2nd such file, the following release rules that can be activated by the appropriate entity attribute rules:
    - ```
      <!-- Attribute release rules activated by Entity Attribute markup in metadata (filter) -->

      <AttributeFilterPolicy id="releaseEntityAttributeMarkedEduPersonPrincipalName">
          <PolicyRequirementRule xsi:type="EntityAttributeExactMatch"
                                 attributeName="http://shibboleth.net/ns/attributes
      /releaseAllValues"
                                 attributeValue="eduPersonPrincipalName" />
          <AttributeRule attributeID="eduPersonPrincipalName">
              <PermitValueRule xsi:type="ANY" />
          </AttributeRule>
      </AttributeFilterPolicy>
      <AttributeFilterPolicy id="releaseEntityAttributeMarkedUid">
          <PolicyRequirementRule xsi:type="EntityAttributeExactMatch"
                                 attributeName="http://shibboleth.net/ns/attributes
      /releaseAllValues"
                                 attributeValue="uid" />
          <AttributeRule attributeID="uid">
              <PermitValueRule xsi:type="ANY" />
          </AttributeRule>
      </AttributeFilterPolicy>
      <AttributeFilterPolicy id="releaseEntityAttributeMarkedMail">
          <PolicyRequirementRule xsi:type="EntityAttributeExactMatch"
                                 attributeName="http://shibboleth.net/ns/attributes
      /releaseAllValues"
                                 attributeValue="mail" />
          <AttributeRule attributeID="mail">
              <PermitValueRule xsi:type="ANY" />
          </AttributeRule>
      </AttributeFilterPolicy>
      <AttributeFilterPolicy id="releaseEntityAttributeMarkedSurname">
          <PolicyRequirementRule xsi:type="EntityAttributeExactMatch"
      ```

```
                                attributeName="http://shibboleth.net/ns/attributes
/releaseAllValues"
                                attributeValue="surname" />
        <AttributeRule attributeID="surname">
            <PermitValueRule xsi:type="ANY" />
        </AttributeRule>
    </AttributeFilterPolicy>
    <AttributeFilterPolicy id="releaseEntityAttributeMarkedGivenName">
        <PolicyRequirementRule xsi:type="EntityAttributeExactMatch"
                                attributeName="http://shibboleth.net/ns/attributes
/releaseAllValues"
                                attributeValue="givenName" />
        <AttributeRule attributeID="givenName">
            <PermitValueRule xsi:type="ANY" />
        </AttributeRule>
    </AttributeFilterPolicy>
    <AttributeFilterPolicy id="releaseEntityAttributeMarkedDisplayName">
        <PolicyRequirementRule xsi:type="EntityAttributeExactMatch"
                                attributeName="http://shibboleth.net/ns/attributes
/releaseAllValues"
                                attributeValue="displayName" />
        <AttributeRule attributeID="displayName">
            <PermitValueRule xsi:type="ANY" />
        </AttributeRule>
    </AttributeFilterPolicy>
    <AttributeFilterPolicy id="releaseEntityAttributeMarkedEduPersonAffiliation">
        <PolicyRequirementRule xsi:type="EntityAttributeExactMatch"
                                attributeName="http://shibboleth.net/ns/attributes
/releaseAllValues"
                                attributeValue="eduPersonAffiliation" />
        <AttributeRule attributeID="eduPersonAffiliation">
            <PermitValueRule xsi:type="ANY" />
        </AttributeRule>
    </AttributeFilterPolicy>
    <AttributeFilterPolicy id="releaseEntityAttributeMarkedEduPersonScopedAffiliation">
        <PolicyRequirementRule xsi:type="EntityAttributeExactMatch"
                                attributeName="http://shibboleth.net/ns/attributes
/releaseAllValues"
                                attributeValue="eduPersonScopedAffiliation" />
        <AttributeRule attributeID="eduPersonScopedAffiliation">
            <PermitValueRule xsi:type="ANY" />
        </AttributeRule>
    </AttributeFilterPolicy>
    <AttributeFilterPolicy id="releaseEntityAttributeMarkedEduPersonPrimaryAffiliation">
        <PolicyRequirementRule xsi:type="EntityAttributeExactMatch"
                                attributeName="http://shibboleth.net/ns/attributes
/releaseAllValues"
                                attributeValue="eduPersonPrimaryAffiliation" />
        <AttributeRule attributeID="eduPersonPrimaryAffiliation">
            <PermitValueRule xsi:type="ANY" />
        </AttributeRule>
    </AttributeFilterPolicy>
    <AttributeFilterPolicy id="releaseEntityAttributeMarkedEduPersonEntitlement">
        <PolicyRequirementRule xsi:type="EntityAttributeExactMatch"
                                attributeName="http://shibboleth.net/ns/attributes
/releaseAllValues"
                                attributeValue="eduPersonEntitlement" />
        <AttributeRule attributeID="eduPersonEntitlement">
            <PermitValueRule xsi:type="ANY" />
        </AttributeRule>
    </AttributeFilterPolicy>
    <AttributeFilterPolicy id="releaseEntityAttributeMarkedEduPersonAssurance">
        <PolicyRequirementRule xsi:type="EntityAttributeExactMatch"
                                attributeName="http://shibboleth.net/ns/attributes
/releaseAllValues"
                                attributeValue="eduPersonAssurance" />
        <AttributeRule attributeID="eduPersonAssurance">
            <PermitValueRule xsi:type="ANY" />
        </AttributeRule>
    </AttributeFilterPolicy>
    <AttributeFilterPolicy id="releaseEntityAttributeMarkedEduPersonUniqueId">
```

```
        <PolicyRequirementRule xsi:type="EntityAttributeExactMatch"
                               attributeName="http://shibboleth.net/ns/attributes
/releaseAllValues"
                               attributeValue="eduPersonUniqueId" />
        <AttributeRule attributeID="eduPersonUniqueId">
            <PermitValueRule xsi:type="ANY" />
        </AttributeRule>
    </AttributeFilterPolicy>
    <AttributeFilterPolicy id="releaseEntityAttributeMarkedEmployeeNumber">
        <PolicyRequirementRule xsi:type="EntityAttributeExactMatch"
                               attributeName="http://shibboleth.net/ns/attributes
/releaseAllValues"
                               attributeValue="employeeNumber" />
        <AttributeRule attributeID="employeeNumber">
            <PermitValueRule xsi:type="ANY" />
        </AttributeRule>
    </AttributeFilterPolicy>
```

    - The above XML is also attached as a file to this wiki page.
    - That your *conf/attribute-resolver.xml* file actually defines attributes with those IDs.
    - Note that you can add the above attribute release rules without any negative impact on your IdP. Particularly once we add the setting to disable any "incoming metadata" from being allowed to already include any entity attribute config for the entity attribute "**http://shibboleth.net/ns/attributes/releaseAllValues**", ensuring that no SP could trigger any such rules by adding those entity attribute definitions into their metadata.
    - Instead of needing to define an entry in your *conf/attribute-fitler.xml* file for a given SP, and adding attribute release rules into it, you can instead add the needed entity attribute/values into that SP's metadata (using the UI) that will trigger each of the above per-attribute release rules.
    - The attributes chosen to date were based on ones that are commonly used. One could add as many more as one wanted, but one would also need to modify the supported list in the ShibUI to include those also.
- The expectation is that TIER will put together a Shibboleth IdP pre-built package that includes the needed IdP config above in the "out of the box IdP", so that the Shibboleth UI can be fully effective with the TIER release. That would mean all of the above config file changes would have already been done for you.
- Summary of IdP config files that would need an update in order to take full advantage of this ShibUI's functionality to date:
    - First, you'll need Shibboleth IdP version 3.4 (not out yet)
    - conf/relying-party.xml (turn on support for metadata-driven "tag" overrides)
    - conf/attributes-filter.xml (add entity attribute-driven release rules)
    - conf/services.xml (to read in 2nd metadata-providers.xml file that the UI manages)
    - conf/metadata-providers.xml (to define a LocalDynamicMetadataProvider provider)
    - metadata/generated (created a "generated" sub-directory in the metadata/ directory)