IdP 2 Configuration Heaven

Work less, achieve more and get total flexibility: Some Examples

SWITCH

Serving Swiss Universities

Lukas Hämmerle
lukas.haemmerle@switch.ch

Metadata refresh: Now built-in in IdP

```
<!-- Fill in metadataURL and backingFile attributes with deployment
  specific information -->
<MetadataProvider   id="URLMD"
  xsi:type="FileBackedHTTPMetadataProvider"
  xmlns="urn:mace:shibboleth:2.0:metadata"
  metadataURL="http://www.switch.ch/[…]/metadata.signed.xml"
  backingFile="/etc/shibboleth/metadata.aaitest.xml"
  maintainExpiredMetadata="true”
  cacheDuration="3600">

        <MetadataFilter
          xmlns="urn:mace:shibboleth:2.0:metadata"
          xsi:type="SignatureValidation"
          requireSignedMetadata="true"
          trustEngineRef="shibboleth.MetadataTrustEngine" />
</MetadataProvider>
[..]
<security:Certificate>
/etc/ssl/certs/metadata.crt
</security:Certificate>
```

Static Attribute Resolution Example

- Example of a static attribute resolver
  - This way you don't need these attributes in your LDAP/AD anymore

```
<!-- Example Static Connector -->
<resolver:DataConnector id="myStaticAttributes"
                        xsi:type="Static"
                        xmlns="urn:mace:shibboleth:2.0:resolver:dc">

<!-- These values can be set for all users -->
  <Attribute id="swissEduPersonHomeOrganization">
    <Value>
      unixy.ch
    </Value>
  </Attribute>

  <Attribute id="swissEduPersonHomeOrganizationType">
    <Value>
      university
    </Value>
  </Attribute>

</resolver:DataConnector>
```

Scriptlet Attribute Example

- Can be used to add/modify almost any attribute

```
<resolver:AttributeDefinition xsi:type="Script"
              xmlns="urn:mace:shibboleth:2.0:resolver:ad"
              id="eduPersonEntitlement">

  <!-- Dependency information would go here -->
  <resolver:Dependency ref="UniversityLdapConnector" />
  <resolver:Dependency ref="affiliation" />
   <Script>
     <![CDATA[
    // Only add entitlement for student and staff members
     if (
       affiliation.getValues().contains("staff")
       || affiliation.getValues().contains("student"))
     {
       entitlement = new BasicAttribute("eduPersonEntitlement");
       entitlement.add("urn:mace:dir:entitlement:common-lib-terms");
     }
       ]]>
   </Script>

</resolver:AttributeDefinition>
```

Attribute Filter files in Shibboleth 2.0 IdP

- Successor of arp.site.xml (Shibboleth 1.3)
- Much more powerful and versatile
- IdP 2.0 can automatically download it from a URL
  - No metadatatool, Perl, curl, mailx, updateARP needed anymore
  - Much easier setup of Identity Provider
- Resource Registry generated attribute-filter.xml for you
  - You can define very specific exceptions rule
  - You can exclude certain resources from filter to create custom rules

Example of a rule generated by the RR

```
<!--RESOURCE_NAME#SWITCH, AAIportal DEMO-->
<AttributeFilterPolicy>
  <PolicyRequirementRule xsi:type="basic:AND">
    <basic:Rule xsi:type="basic:AttributeRequesterString"
                value="https://demo.aaiportal.switch.ch/shibboleth" />
    <basic:Rule xsi:type="saml:AttributeRequesterInEntityGroup"
                groupID="urn:mace:switch.ch:SWITCHaai" />
  </PolicyRequirementRule>

  <AttributeRule attributeID="affiliation">
    <PermitValueRule xsi:type="basic:ANY" />
  </AttributeRule>

  <AttributeRule attributeID="uniqueID">
    <PermitValueRule xsi:type="basic:ANY" />
  </AttributeRule>

</AttributeFilterPolicy>
```

- This will only release a user's affiliation and uniqueID
- Very similar to arp.site.xml rules from Shibboleth 1.3

Other things you could do

- Time-based attribute release
  - Only release affiliation=staff during office hours :-)
- Login name-dependent attribute release
  - Could be used e.g. if login name has a faculty prefix
- Value dependent attribute release
  - Only release needed entitlement values to a resource
- Authentication method dependent release
  - Don’t release SAP attribute unless X.509 authentication was used
- Scripted attribute release
  - Release certain attributes only if user is within intranet

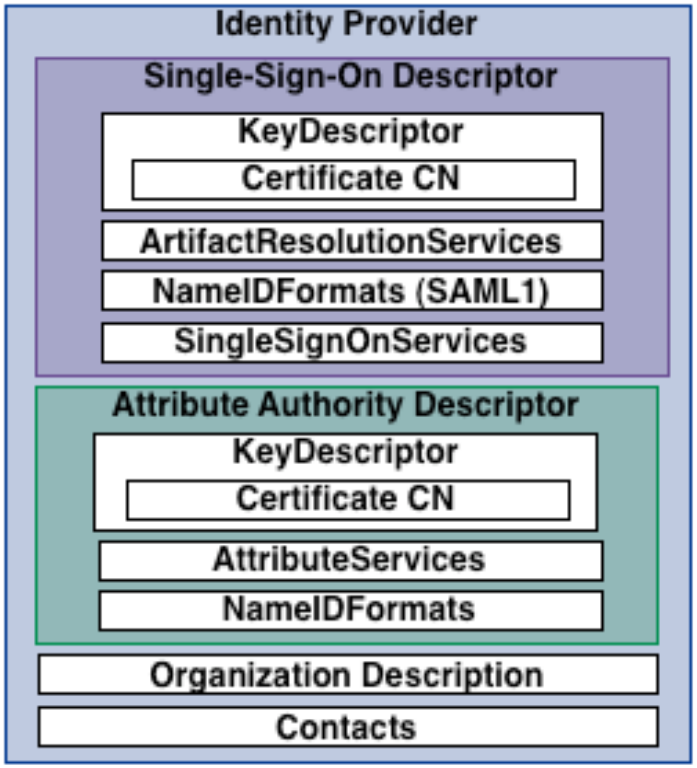
… and more. You are totally flexible with Shibboleth 2.0

Attribute-filter refresh: Also built-in

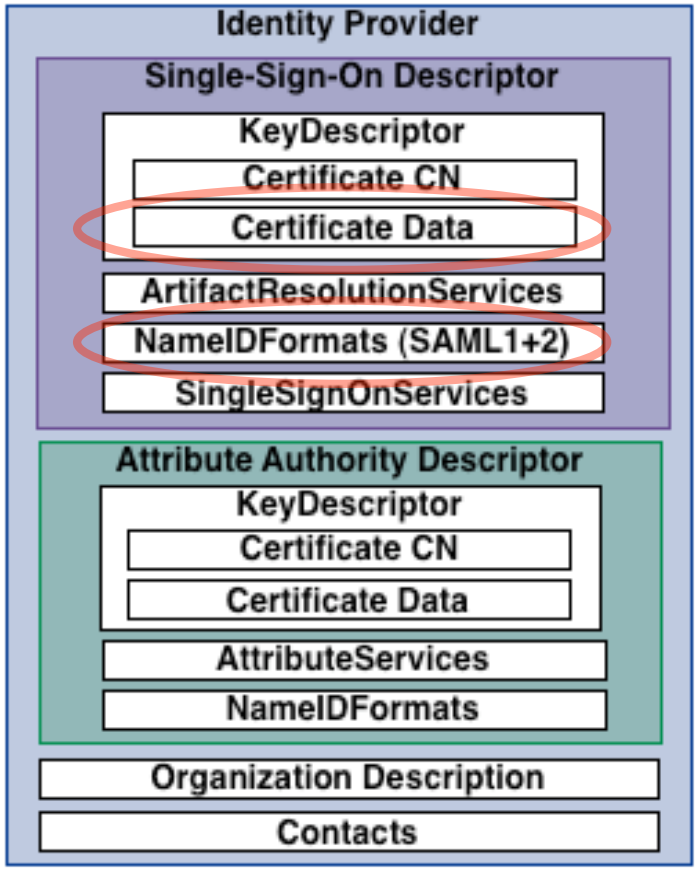
```
<Service
      id="shibboleth.AttributeFilterEngine"
      xsi:type="attribute-afp:ShibbolethAttributeFilteringEngine">
         <ConfigurationResource
         url="https://aai-rr.switch.ch/[…]/attribute-filter.xml"
         file="/opt/shibboleth-idp-trunk/conf/attribute-filter.xml"
         xsi:type="resource:FileBackedHttpResource" />
</Service>
```

- No need for updateARP script anymore
- Refresh time could be specified too
- Not shown above is signature validation

What's new for Shibboleth 2 IdP Metadata



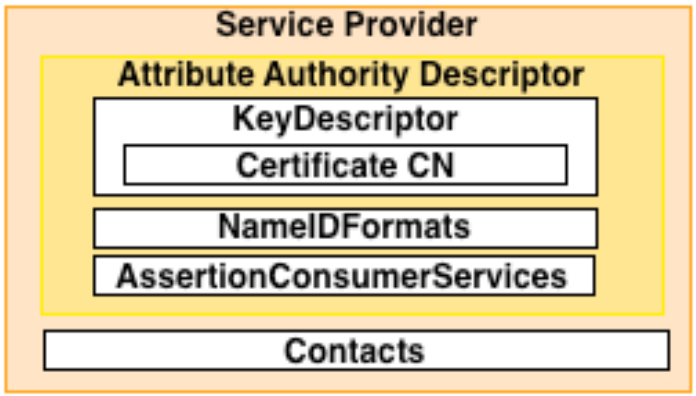
So far …



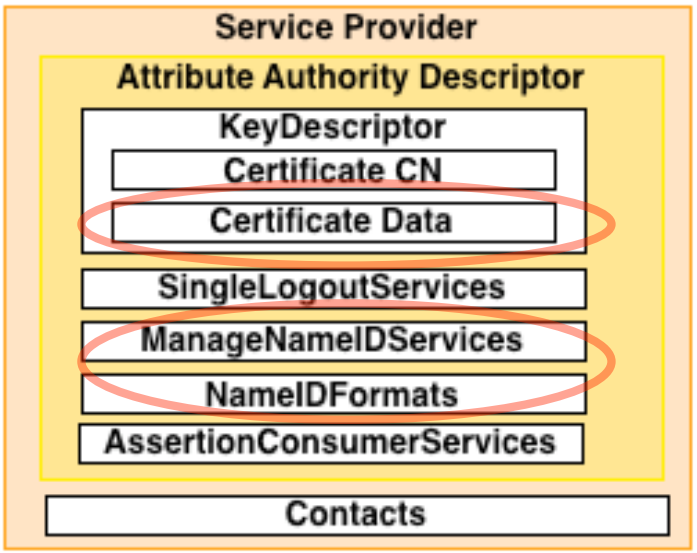
Now also possible

Certificate Data can also be Included for Shibboleth 1.3

What's new for Shibboleth 2 SP Metadata



So far …



Now also possible

Certificate Data can also be
Included for Shibboleth 1.3