IdP User Consent

Transparency for attribute release

SWITCHaai Team
aai@switch.ch

Part 1: Overview of user consent in IdP version 3

Part 2: Technical bits

User consent

Two pieces

1. Attribute release consent [enabled]
2. Terms of use consent [disabled]

Both prompt user on first access to every SP and again when attributes or terms change.

What's in version 3

- Attribute release and terms of use consent now built in
- Inspired by uApprove and uApproveJP plugins for v2
- No consent data migration, storage implementations are *not compatible*
- May be enabled or disabled per relying party and per profile
- Decisions logged

Differences with uApprove

- User can select attributes to release [disabled]
- Consent duration choices
  1. "Ask me again if information changes"
     = if *set* of attributes changes
  2. "Ask me again at next login" [enabled]
  3. "Do not ask me again", ever, for any SP [enabled]
- No regular expression for SP white/black lists
- No translations provided

Why enable user consent?

- Easier to have now with v3 than with v2
- Required by SWITCHaai Interfederation Access Declaration
- Inform users about what personal data is transmitted in a more real-time fashion
- Recommended to comply with data protection laws

Why (not) enable user consent?

- One more page to read and click through upon login, but only the first time for each SP
- Global consent disabled: users cannot choose "I don't care about my privacy"
- Decide for all your users or let them decide?

When should consent be sought?

- All SPs: the best option ← recommended
- Outside your organisation: good, less clicks
- Outside CH: currently no technical means to distinguish "Swiss" SPs and the data might not even be stored in Switzerland

Part 2: Technical bits

Global configuration options

Configured by Spring beans in `conf/relying-party.xml`, see comments inside for overrides

Post-authentication flows

- Attribute consent [enabled]
- Terms of use consent [disabled]

Example conf/relying-party.xml

Both post-authentication flows enabled for SAML2 SSO
[only attribute-release]

```
<bean id="shibboleth.DefaultRelyingParty" parent="RelyingParty">
  <property name="profileConfigurations">
    <list>
      <bean parent="Shibboleth.SSO"
        p:postAuthenticationFlows="attribute-release" />
      <ref bean="SAML1.AttributeQuery" />
      <ref bean="SAML1.ArtifactResolution" />
      <bean parent="SAML2.SSO"
        p:postAuthenticationFlows="#{{'terms-of-use','attribute-release'}}"/>
      <ref bean="SAML2.ECP" />
      <ref bean="SAML2.Logout" />
      <ref bean="SAML2.AttributeQuery" />
      <ref bean="SAML2.ArtifactResolution" />
    </list>
  </property>
</bean>
```


Attribute consent configuration

Configured by Java properties in
`conf/idp.properties`

Consent duration options

- "Ask me again if information changes"
  Always available to users
- "Ask me again at next login"
  `idp.consent.allowDoNotRemember = true` [true]
- "Do not ask me again"
  `idp.consent.allowGlobal = false` [true]

Attribute consent configuration

Per attribute behaviour

- Allow selection of attributes to release, may break applications if required attributes are withheld
  `idp.consent.allowPerAttribute = false` [false]
- Ask again if attribute *values* change
  `idp.consent.compareValues = true` [false]

Intercept flow configuration

Configured by Spring beans in
`conf/intercept/consent-intercept-config.xml`

White & black lists

Which attribute to prompt for [all except black list]

- White list [empty]
  When filled: any attribute not mentioned in a list is released *without asking*
- Black list [`transientId`, `persistentId`, `eduPersonTargetedID`]
- Pattern match [not defined]

Intercept flow configuration

Attribute display order (coming in version 3.2.0)

- Alphabetical order by default
- Except attributes in white list show up first
- Set pattern to `^.*$` to catch all other attributes
- Fully customised order ⇒ implement `java.util.Comparator<String>`

Terms of use consent configuration

Configured by Java properties in `messages/consent-messages.properties`

Terms for each SP

- Default mapping using the `entityID`

```
https\://sp.example.org = example-tou-1
example-tou-1.title = Example Terms of Use
example-tou-1.text = <em>This is an example ToU</em> [...]
```

- Other mapping configurable but the key is still `entityID` (default value available)

Custom terms of use mapping

Configured by Spring beans in `conf/intercept/consent-intercept-config.xml`

Provided bean mapping `entityID`s to values [disabled]

```
<bean id="shibboleth.consent.terms-of-use.Key"
      class="com.google.common.base.Functions" factory-method="compose">
  <constructor-arg name="g">
    <bean class="com.google.common.base.Functions" factory-method="forMap"
          c:defaultValue="terms-of-use">
      <constructor-arg name="map">
        <map>
          <entry key="https://sp.example.org/shibboleth" value="example-terms"
        </map>
      </constructor-arg>
    </bean>
  </constructor-arg>
  <constructor-arg name="f">
    <ref bean="shibboleth.RelyingPartyIdLookup.Simple" />
  </constructor-arg>
</bean>
```


Hands-on: use only one ToU

We want to always display the same terms of use regardless of the SP.

Hands-on solution

- Enable `terms-of-use` flow in `conf/relying-party.xml`
- Change key bean in `conf/intercept/consent-intercept-config.xml` to

```
<bean id="shibboleth.consent.terms-of-use.Key"
      class="com.google.common.base.Functions"
      factory-method="constant">
  <constructor-arg value="my-terms" />
</bean>
```


Hands-on solution

- Add text in `messages/consent-messages.properties`

```
my-terms = bogus-tou
bogus-tou.title = Bogus Terms of Use
bogus-tou.text = You can do anything you want!
```

References

- Shibboleth wiki: ConsentConfiguration
  (https://wiki.shibboleth.net/confluence/display/IDP30/ConsentConfiguration)
- Shibboleth wiki: RelyingPartyConfiguration
  (https://wiki.shibboleth.net/confluence/display/IDP30/RelyingPartyConfiguration)
- Google Guava Functions class Javadoc
  (http://docs.guava-libraries.googlecode.com/git/javadoc/com/google/common/base/Functions.html)

Appendix: Disabling attribute consent prompt for particular SPs

Disabling prompt for particular SPs

Relying party overrides

- Template beans in `conf/relying-party.xml` to match SPs by:
  - name: `entityID`
  - group: `<EntitiesDescriptor>` in metadata
  - tag: `<EntityAttributes>` metadata extension
- **First match wins** ⇒ order in `conf/relying-party.xml` is significant

Disabling prompt for particular SPs

Entity attributes in metadata

- Entity categories
  - GÉANT Data Protection Code of Conduct (CoCo)
  - REFEDS Research & Scholarship
- New attributes available!
  - `swissEduPersonHomeOrganization`
  - `swissEduPersonHomeOrganizationType`

Example metadata with attributes

```
<EntityDescriptor
  entityID="https://attribute-viewer.aai.switch.ch/interfederation-test/shibbol
  <Extensions>
    <mdattr:EntityAttributes>
      <saml:Attribute Name="http://macedir.org/entity-category">
        <saml:AttributeValue>
          http://www.geant.net/uri/dataprotection-code-of-conduct/v1
        </saml:AttributeValue>
      </saml:Attribute>
      <saml:Attribute FriendlyName="swissEduPersonHomeOrganization"
                      Name="urn:oid:2.16.756.1.2.5.1.1.4">
        <saml:AttributeValue>switch.ch</saml:AttributeValue>
      </saml:Attribute>
      <saml:Attribute FriendlyName="swissEduPersonHomeOrganizationType"
                      Name="urn:oid:2.16.756.1.2.5.1.1.5">
        <saml:AttributeValue>others</saml:AttributeValue>
      </saml:Attribute>
    </mdattr:EntityAttributes>
  </Extensions>
  <!-- ... rest of metadata for entity -->
</EntityDescriptor>
```


Example relying party override

Disables flows for SPs belonging to a home organisation

```
<util:list id="shibboleth.RelyingPartyOverrides">
  <!-- ... more beans -->
  <bean id="shibboleth.NoUserConsentRelyingParty" parent="RelyingPartyByTag">
    <constructor-arg name="candidates">
      <list>
        <bean id="disableForSingleHomeOrganization" parent="TagCandidate"
              c:name="urn:oid:2.16.756.1.2.5.1.1.4"
              p:values="example.org" />
        <!-- ... more beans -->
      </list>
    </constructor-arg>
    <property name="profileConfigurations">
      <list>
        <ref bean="Shibboleth.SSO" />
        <ref bean="SAML2.SSO" />
        <!-- ... other profiles -->
      </list>
    </property>
  </bean>
</util:list>
```