Shibboleth IdPによるアクセス制限

本メニューはIdPv2のFPSPプラグイン相当のことをIdPv4の組み込み機能で実現することが目的です。

1. はじめに

本メニューでは、IdPをカスタマイズします。
送信属性の値を使って、IdP側でSPへのアクセス制限を行います。

---

2. 実習セミナーでは

以下の手順で作業を進めてください。

・ContextCheckを有効にする

4.1.0以降では同意機能はモジュール化されており、利用するには有効化操作が必要です。以下のコマンドを実行してください。（当該モジュールがすでに有効化されているかを確認し、有効化されていない場合に有効化するものです）

```
# /opt/shibboleth-idp/bin/module.sh -t idp.intercept.ContextCheck || /opt/shibboleth-idp/bin/module.sh -e idp.intercept.ContextCheck
```

・relying-party.xmlの修正

/opt/shibboleth-idp/conf/relying-party.xmlにアクセス制限が行えるように設定します。
本メニューでは、送信属性同意機能を有効にした状態でアクセス制限を行う設定とします。

```
<bean id="shibboleth.DefaultRelyingParty" parent="RelyingParty">
  <property name="profileConfigurations">
    <list>

 （省略）
<!--
      <bean parent="SAML2.SSO" p:postAuthenticationFlows="attribute-release" />
-->
      <bean parent="SAML2.SSO" p:postAuthenticationFlows="#{ {'context-check', 'attribute-release'} }" />
      <ref bean="SAML2.ECP" />
      <ref bean="SAML2.Logout" />
<!--
      <ref bean="SAML2.AttributeQuery" />
-->
      <ref bean="SAML2.ArtifactResolution" />
      <ref bean="Liberty.SSOS" />
   </list>
 </property>
</bean>
```

もしくはSPを限定して適用するには以下のRelyingPartyOverridesの部分に追加します。
※SPのentityIDの部分は**各自に割り振られたもの**を用いてください。

```
            </property>
        </bean>
        -->
        <bean parent="RelyingPartyByName" c:relyingPartyIds="#{{'https://ex-sp-test01.gakunin.nii.ac.jp/shibboleth-sp'}}">
            <property name="profileConfigurations">
                <list>
                    <!-- SAML 1.1 and SAML 2.0 AttributeQuery are disabled by default. -->
                    <!--
                    <bean parent="Shibboleth.SSO" p:postAuthenticationFlows="#{{'context-check', 'attribute-release'}}" />
                    <ref bean="SAML1.AttributeQuery" />
                    <ref bean="SAML1.ArtifactResolution" />
                    -->
                    <bean parent="SAML2.SSO" p:postAuthenticationFlows="#{{'context-check', 'attribute-release'}}" />
                    <ref bean="SAML2.ECP" />
                    <ref bean="SAML2.Logout" />
                    <!--
                    <ref bean="SAML2.AttributeQuery" />
                    -->
                    <ref bean="SAML2.ArtifactResolution" />
                </list>
            </property>
        </bean>

    </util:list>

</beans>
```

複数のSPで制限を行う場合は c:relyingPartyIds="#{{'https://ex-sp-test01.gakunin.nii.ac.jp/shibboleth-sp', 'https://ex-sp-test02.gakunin.nii.ac.jp/shibboleth-sp'}}" のように書くことができます。

relying-party.xmlの前者の設定では、後述のcontext-check-intercept-config.xmlで記述していない他のSPについては全ユーザがアクセスできなくなりますのでご注意ください。SPを限定する記述を取り除くか、アクセスを許容する全てのSPを列挙するようにしてください。
もしくは、上記後者の記述で特定のSPのみに本ContextCheck機能を適用してください。

・context-check-intercept-config.xmlの修正

/opt/shibboleth-idp/conf/intercept/context-check-intercept-config.xmlにアクセス制限の条件を設定します。
本メニューでは、構築SPについてログイン時のUsernameが「test002」の場合、アクセスできるように設定します。
※SPのentityIDの部分は**各自に割り振られたもの**を用いてください。

```
<bean id="shibboleth.context-check.Condition" parent="shibboleth.Conditions.AND">
 <constructor-arg>
   <list>
<!--
     <bean parent="shibboleth.Conditions.RelyingPartyId" c:candidates="#{ 'https://sp.example.org' }" />
-->
     <bean parent="shibboleth.Conditions.RelyingPartyId" c:candidates="#{{ 'https://ex-sp-test01.gakunin.nii.ac.jp/shibboleth-sp' }}"
/>
     <bean class="net.shibboleth.idp.profile.logic.SimpleAttributePredicate">
       <property name="attributeValueMap">
         <map>
<!--
           <entry key="eppn">
-->
           <entry key="eduPersonPrincipalName">
             <list>
<!--
               <value>*</value>
-->
               <value>test002</value>
             </list>
           </entry>
         </map>
       </property>
     </bean>
   </list>
 </constructor-arg>
</bean>
```

複数のSPにマッチさせる場合は c:candidates="#{{'https://ex-sp-test01.gakunin.nii.ac.jp/shibboleth-sp', 'https://ex-sp-test02.gakunin.nii.ac.jp/shibboleth-sp'}}" のように書くことができます。

別の例: https://shibboleth.atlassian.net/wiki/spaces/IDP4/pages/1265631713/ActivationConditions#Examples

・Jettyの再起動

Jettyを再起動して、修正した設定ファイルを読み込ませます。

```
# systemctl restart jetty
```

今後も、context-check-intercept-config.xml を直接編集した場合はreload-service.shによる再読み込みができませんので、更新を反映するためにはJettyを再起動する必要があります。

これを避けるために判定のロジックをattribute-resolver.xmlで実装する、すなわちtrue/falseを表す専用の属性を生成し、context-check-intercept-config.xml では当該属性のみを参照するようにすることをお勧めします。attribute-resolver.xmlの更新であれば

```
# /opt/shibboleth-idp/bin/reload-service.sh -id shibboleth.AttributeResolverService
```

により再読み込みすることが可能です。

---

3. 手順書

以下は、英語での情報が記載されたwiki.shibboleth.netのURLです。手順の詳細にご興味がある方はご参照ください。
※送信属性によるアクセス制限の設定については記載されていませんが、 RelyingPartyについてのページのリンクとなります。

参考: RelyingPartyConfiguration

---

4. 動作確認

① 各自が使用するSPの接続確認用ページにアクセスします。

```
例）1番を割り振られた場合 https://ex-sp-test01.gakunin.nii.ac.jp/
```

② ログインボタンをクリックします。

③ DSの所属機関の選択画面が表示されるので、各自が使用するIdPを選択します。

④ IdPのログイン画面が表示されます。

⑤ Username/Passwordを入力して認証を行います。
※test001、test003でログイン：アクセスが拒否されることを確認します。
（「ウェブログインサービス - アクセス拒否」と表示されたShibbolethIdPの画面が表示）
※ test002でログイン：⑥以降に進みます。

⑥ 送信属性同意画面が表示されます。

⑦ 正しく属性受信の確認ページに表示される事を確認してください。