IdPClusteringStateless

本ページの記述はIdPv2に対するものです。IdPv3ではここで説明されている機能が概ね組み込みで標準提供されています。

Shibboleth-IdP冗長化環境構築手順書（Stateless Clustering編）

2012年5月8日
国立情報学研究所

- 目次 -

1. はじめに

1.1. 本章の目的

本書は、Stateless Clustering方式によるShibboleth-IdPの冗長化手順書です。
本書にて、Stateless Clustering方式によるShibboleth-IdP の冗長化ができることを目的とします。

1.2. 前提条件

Stateless Clustering方式によるShibboleth-IdPの冗長化を構築するにあたり、下記を前提条件とします。
前提条件

- Shibboleth-IdPは、学認の技術ガイドに基づいてインストールされており、認証できる状態であるものとします。
- OSはCentOS(64bit)を前提とします。
- Stateless Clustering方式のShibboleth-IdPを構築するサーバは2台とします。
- 本書は下記のソフトウェアを使用して記述します。

ソフトウェア	バージョン	インストール先
osuidpext (Ohio State Custom Login Handler)	1.1	-
Shibboleth-IdP	2.3.5	/opt/shibboleth-idp
Apache Tomcat	6.0.35	/usr/java/tomcat

1.3. ソフトウェア配布元

Stateless Clusteringに必要なソフトウェアは以下で配布されています。その他Shibboleth IdPに取り込まれている機能も利用しています。

- 下記URLの"Ohio State Custom Login Handler"
  https://wiki.shibboleth.net/confluence/display/SHIB2/Contributions#Contributions-IdentityProviderExtensions

2. IdP Stateless Clustering方式の構築

IdP Stateless Clustering方式のShibboleth-IdPを構築する全てのサーバで実施します。

2.1. Apache Tomcatの停止

- Tomcatが起動している場合は停止します。

```
# /etc/init.d/tomcat stop
```

2.2. ライブラリの追加

- Stateless Clustering用ライブラリを追加します。

```
# wget https://webauth.service.ohio-state.edu/~shibboleth/downloads/osuidpext-1.1-src.tar.gz
# tar zxvf osuidpext-1.1-src.tar.gz
# cd osuidepext-1.1/lib
# cp osuidpext-1.1.jar /usr/java/tomcat/webapps/idp/WEB-INF/lib/
```

- StatelessClustering用ライブラリが利用しているSecurIDPrincipal.javaをソースからコンパイルしjarファイル化します。

```
# svn co https://svn.shibboleth.net/extensions/java-jaas-securid/trunk/
# cd trunk
# rm -f  src/edu/internet2/middleware/shibboleth/jaas/securid/SecurIDLoginModule.java
# ./ant.sh

jarファイルの作成場所　:　trunk/dist/jaas-securid-1.0.1-jdk-1.5.jar
```

※svnコマンドの実行には、subversionクライアントがインストールされている必要があります。

- jarファイルをサーバに追加します。

```
# cp dist/jaas-securid-1.0.1-jdk-1.5.jar /usr/java/tomcat/webapps/idp/WEB-INF/lib/
```

2.3. keyStoreの作成

- keytoolにてkeystoreを作成し、作成後、全てのShibboleth-IdPサーバにコピーします。

```
# keytool -genseckey -keystore secret.jks -storetype JCEKS -alias secret -keyalg AES -keysize 128
キーストアのパスワードを入力してください:　　←例)　secret　と入力
新規パスワードを再入力してください:　　←例)　secret　と入力
<secret> の鍵パスワードを入力してください。
　　(キーストアのパスワードと同じ場合は RETURN を押してください):　　←そのままenter

# cp secret.jks /opt/shibboleth-idp/credentials/
```

2.4. Shibboleth-IdPの設定ファイルの変更

- web.xmlにカスタムサーブレットをインストールして実行するサブモジュールを宣言します。

```
# vi /usr/java/tomcat/webapps/idp/WEB-INF/web.xml
```

web.xmlに下記を追加します。

```
<context-param>
    <param-name>contextConfigLocation</param-name>
    <param-value>file:///opt/shibboleth-idp/conf/internal.xml;
                 file:///opt/shibboleth-idp/conf/service.xml;
                 file:///opt/shibboleth-idp/conf/osuext.xml;</param-value>
</context-param>
```

web.xmlに下記を追加します。

```
<!-- Servlet for doing stateless cookie-based authentication -->
<servlet>
    <servlet-name>StatelessAuthHandler</servlet-name>
    <servlet-class>edu.osu.ocio.shibboleth.idp.authn.provider.StatelessLoginServlet</servlet-class>
    <load-on-startup>5</load-on-startup>
    <init-param>
        <param-name>dataSealerRef</param-name>
        <param-value>shibboleth.SSODataSealer</param-value>
    </init-param>
    <init-param>
        <param-name>submodules</param-name>
        <param-value>shibboleth.LDAPLoginSubmodule shibboleth.VelocityFormLoginSubmodule</param-value>
    </init-param>
    <init-param>
        <param-name>errorContext</param-name>
        <param-value>/</param-value>
    </init-param>
    <init-param>
        <param-name>errorPage</param-name>
        <param-value>/stale.html</param-value>
    </init-param>
</servlet>

<servlet-mapping>
    <servlet-name>StatelessAuthHandler</servlet-name>
    <url-pattern>/Authn/Stateless</url-pattern>
</servlet-mapping>
```

- 設定ファイル（osuext.xml）を新規に追加します。

```
# vi /opt/shibboleth-idp/conf/osuext.xml
```

以下の内容を追加します。

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:util="http://www.springframework.org/schema/util"
        xsi:schemaLocation="http://www.springframework.org/schema/beans
        http://www.springframework.org/schema/beans/spring-beans-2.0.xsd
        http://www.springframework.org/schema/util
        http://www.springframework.org/schema/util/spring-util-2.0.xsd" >
    <bean id="shibboleth.SSODataSealer"
            class="edu.internet2.middleware.shibboleth.common.util.DataSealer"
            depends-on="shibboleth.LogbackLogging" init-method="init">
        <!-- 2.3章で作成したkeyStoreを設定します。 -->
        <property name="keystorePath" value="/opt/shibboleth-idp/credentials/secret.jks" />
        <property name="keystorePassword" value="secret" />
        <property name="cipherKeyAlias" value="secret" />
        <property name="cipherKeyPassword" value="secret" />
    </bean>
    <bean id="shibboleth.SSOVelocityEngine"
            class="org.springframework.ui.velocity.VelocityEngineFactoryBean"
            depends-on="shibboleth.LogbackLogging">
        <property name="overrideLogging" value="true"/>
        <property name="velocityProperties">
            <props>
                <prop key="runtime.log.logsystem.class">
                    edu.internet2.middleware.shibboleth.common.util.Slf4JLogChute
                </prop>
                <prop key="resource.loader">file</prop>
                <prop key="file.resource.loader.class">
                    org.apache.velocity.runtime.resource.loader.FileResourceLoader
                </prop>
                <prop key="file.resource.loader.path">/opt/shibboleth-idp/conf</prop>
                <prop key="file.resource.loader.cache">false</prop>
            </props>
        </property>
```

```
    </bean>
    <bean id="shibboleth.LDAPLoginSubmodule"
            class="edu.osu.ocio.shibboleth.idp.authn.provider.JAASLoginSubmodule">
        <constructor-arg value="file:///opt/shibboleth-idp/conf/login.config "/>
        <property name="jaasConfigName"
                                    value="ShibUserPassAuth" /> ←login.configにかかれているものと同じにします。
        <property name="authnMethods">
            <set>
                <value>urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport</value>
            </set>
        </property>
        <property name="unknownUsernameErrors">
            <list>
                <value>Cannot authenticate dn, invalid dn</value>
            </list>
        </property>
        <property name="invalidPasswordErrors">
            <list>
                <value>AcceptSecurityContext error, data 52e</value>
                <value>[LDAP: error code 49 - Invalid Credentials]</value>
            </list>
        </property>
    </bean>

    <bean id="shibboleth.VelocityFormLoginSubmodule"
            class="edu.osu.ocio.shibboleth.idp.authn.provider.VelocityFormLoginSubmodule"
            depends-on="shibboleth.LogbackLogging">
        <property name="templateName" value="login.vt" />
        <property name="velocityEngine" ref="shibboleth.SSOVelocityEngine" />
    </bean>

    <bean id="shibboleth.OSUServletAttributeExporter"
            class="edu.internet2.middleware.shibboleth.common.config.service.ServletContextAttributeExporter"
            depends-on="shibboleth.SSODataSealer shibboleth.LogbackLogging shibboleth.LDAPLoginSubmodule shibboleth.
VelocityFormLoginSubmodule"
                        ↑半角の空白スペースで区切ります
            init-method="initialize" >
        <constructor-arg>
            <list>
                <value>shibboleth.SSODataSealer</value>
                <value>shibboleth.LDAPLoginSubmodule</value>
                <value>shibboleth.VelocityFormLoginSubmodule</value>
            </list>
        </constructor-arg>
    </bean>

</beans>
```

※端末のサイズによっては表記がずれる可能性がございます。画面を広くしてご覧ください。

- handler.xmlを変更します。

```
# vi /opt/shibboleth-idp/conf/handler.xml
```

3行目付近を編集します。

```
<ph:ProfileHandlerGroup
        xmlns:ph="urn:mace:shibboleth:2.0:idp:profile-handler"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:osu="urn:mace:osu.edu:shibboleth:idp-ext"
        xsi:schemaLocation="urn:mace:shibboleth:2.0:idp:profile-handler
            classpath:/schema/shibboleth-2.0-idp-profile-handler.xsd
            urn:mace:osu.edu:shibboleth:idp-ext classpath:/schema/idp-osu-ext.xsd">
```

※ダブルクォート(")の位置に注意

以下を追加します。

```
    <!-- 追加する部分　ここから -->
    <ph:LoginHandler xsi:type="osu:Stateless" authenticationServletURL="/Authn/Stateless">
        <ph:AuthenticationMethod>
            urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport
        </ph:AuthenticationMethod>
    </ph:LoginHandler>
    <!-- 追加する部分　ここまで -->
</ph:ProfileHandlerGroup>
```

以下をコメントアウトします。

```
<!--
<ph:LoginHandler xsi:type="ph:UsernamePassword"
        jaasConfigurationLocation="file:///opt/shibboleth-idp/conf/login.config">
    <ph:AuthenticationMethod>urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport
    </ph:AuthenticationMethod>
</ph:LoginHandler>
-->
```

- attribute-resolver.xmlを変更します。

```
# vi /opt/shibboleth-idp/conf/attribute-resolver.xml
```

<resolver:AttributeDefinition>タグの並びに以下を追加します。

```
<resolver:AttributeDefinition id="cryptoTransientId" xsi:type="ad:CryptoTransientId"
                              xmlns:ad="urn:mace:shibboleth:2.0:resolver:ad"
                              dataSealerRef="shibboleth.SSODataSealer"
                              lifetime="PT3M">

    <resolver:AttributeEncoder xsi:type="encoder:SAML1StringNameIdentifier"
                               xmlns:encoder="urn:mace:shibboleth:2.0:attribute:encoder"
                               nameFormat="urn:mace:shibboleth:1.0:nameIdentifier"/>

    <resolver:AttributeEncoder xsi:type="encoder:SAML2StringNameID"
                               xmlns:encoder="urn:mace:shibboleth:2.0:attribute:encoder"
                               nameFormat="urn:oasis:names:tc:SAML:2.0:nameid-format:transient"/>
</resolver:AttributeDefinition>
```

<resolver:PrincipalConnector>タグの並びに以下を追加します。

```
<resolver:PrincipalConnector id="shibCryptoTransient" xsi:type="pc:CryptoTransient"
                             xmlns:pc="urn:mace:shibboleth:2.0:resolver:pc"
                             dataSealerRef="shibboleth.SSODataSealer"
                             nameIDFormat="urn:mace:shibboleth:1.0:nameIdentifier"/>

<resolver:PrincipalConnector id="saml2CryptoTransient" xsi:type="pc:CryptoTransient"
                             xmlns:pc="urn:mace:shibboleth:2.0:resolver:pc"
                             dataSealerRef="shibboleth.SSODataSealer"
                             nameIDFormat="urn:oasis:names:tc:SAML:2.0:nameid-format:transient"/>
```

以下の resolver:PrincipalConnector はコメントアウトします。

```
<!--
<resolver:PrincipalConnector xsi:type="pc:Transient" id="shibTransient"
        nameIDFormat="urn:mace:shibboleth:1.0:nameIdentifier"/>
<resolver:PrincipalConnector xsi:type="pc:Transient" id="saml1Unspec"
        nameIDFormat="urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified"/>
<resolver:PrincipalConnector xsi:type="pc:Transient" id="saml2Transient"
        nameIDFormat="urn:oasis:names:tc:SAML:2.0:nameid-format:transient"/>
-->
```

- attribute-filter.xmlを変更します。

```
# vi /opt/shibboleth-idp/conf/attribute-filter.xml
```

<afp:AttributeRule>タグの並びに以下を追加します。また、attributeID="transientId" のタグをコメントアウトします。

```
<!--
<afp:AttributeRule attributeID="transientId">
    <afp:PermitValueRule xsi:type="basic:ANY"/>
</afp:AttributeRule>
-->
<!-- 追加する部分　ここから -->
<afp:AttributeRule attributeID="cryptoTransientId">
    <afp:PermitValueRule xsi:type="basic:ANY"/>
</afp:AttributeRule>
<!-- 追加する部分　ここまで -->
```

- ログインファイル（login.vt）を新規に追加します。

```
# vi /opt/shibboleth-idp/conf/login.vt
```

以下の内容を追加します。

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/REC-html401/loose.dtd">
<html lang="ja">
<head>
    <META http-equiv="Content-Type" content="text/html;charset=UTF-8">
</head>
<body>

<!-- excerpt of the interesting bits -->

<div id="loginForm">

#if ($authnInfo.isAccountLocked())
    <blockquote style="font-weight:bold; text-align:left">
    <p style="color:#cc0000">Login failed. Your Ohio State Username account is locked.</p>
    </blockquote>
#elseif ($authnInfo.isExpiredPassword())
    <blockquote style="font-weight:bold; text-align:left">
    <p style="color:#cc0000">Login failed. The password associated with your Ohio State Username account has expired.</p>
    </blockquote>
#elseif ($authnInfo.isInvalidPassword())
    <blockquote style="font-weight:bold; text-align:left">
    <p style="color:#cc0000">Login failed. The password you entered is incorrect.</p>
    </blockquote>
#elseif ($authnInfo.isUnknownUsername())
    <blockquote style="font-weight:bold; text-align:left">
    <p style="color:#cc0000">Login failed. The username you entered cannot be identified.</p>
    </blockquote>
#elseif ($authnInfo.getLoginException())
    <blockquote style="font-weight:bold; text-align:left">
    <p style="color:#cc0000">An error occurred during login: $encoder.encodeForHTML($authnInfo.getLoginException().
getMessage())</p>
    </blockquote>
#elseif ($authnInfo.getAuthnException())
    <blockquote style="font-weight:bold; text-align:left">
    <p style="color:#cc0000">An error occurred during login: $encoder.encodeForHTML($authnInfo.getAuthnException().
getMessage())</p>
    </blockquote>
#end
    <form name="login" method="POST" action="$servletPath">
        <fieldset>
            <legend>Identify Yourself</legend>
            <div>
                <label for="userid">
                    Enter your Username
                </label>
                <br/>
                <input type="text" class="text" id="userid" name="j_username" size="50"/>
            </div>
        </fieldset>
        <fieldset>
            <legend>Password <span class="smallOr">or</span> Passcode</legend>
            <div>
                <label for="password">
                    Enter your account password.<br/>
                </label>
                <br/>
                <input type="password" class="text" id="password"  name="j_password" size="50"/>
            </div>
        </fieldset>
        <input type="hidden" name="j_continue" value="1"/>
        <input id="submit" class="submit" type="submit" value="Login"/>
    </form>

</div>

</body>
</html>
```

2.5. httpdの設定変更

この設定は2.4節で作成したログインファイル（login.vt）がテキストで表示されてしまう場合のみ設定してください。

- velocityを利用できるようにhttpd.confを編集します。

```
# vi /etc/httpd/conf/httpd.conf
```

DefaultType text/plainを以下のように変更します。

```
DefaultType text/html
```

2.6. tomcatの起動

- tomcatの起動

```
# /etc/init.d/tomcat start
```

2.7. httpdの再起動

```
# /etc/init.d/httpd restart
```

以上。