# Shib2IdpTerracottaConfiguration

# Shibboleth 2 Terracotta konfiguráció

Shibboleth 2.1.2 IdP -hez

```xml
<?xml version="1.0" encoding="UTF-8"?>

<tc:tc-config xmlns:tc="http://www.terracotta.org/config"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.terracotta.org/config http://www.terracotta.org/schema/terracotta-4.xsd">

  <!--
      Terracotta configuration file for Shibboleth.

      Complete documentation on the contents of this file may be found here:
      http://terracotta.org/web/display/docs/Configuration+Guide+and+Reference
  -->

  <tc-properties>
    <!-- server-to-server reconnect -->
    <property name="l2.nha.tcgroupcomm.reconnect.enabled" value="true" />
    <property name="l2.nha.tcgroupcomm.reconnect.timeout" value="15000" />

    <!-- client-to-server reconnect -->
    <property name="l2.l1reconnect.enabled" value="true"/>
    <property name="l2.l1reconnect.timeout.millis" value="15000" />
  </tc-properties>
```

```xml
<servers>
   <!-- START Terracotta server definitions -->
      <server name="idp1" host="idp1.aai.niif.hu">
         <dso>
            <persistence>
                <mode>permanent-store</mode>
            </persistence>
         </dso>

         <logs>/var/log/terracotta/server/logs</logs>
         <data>/var/lib/terracotta/server/data</data>
         <statistics>/var/lib/terracotta/server/stats</statistics>
      </server>

      <server name="idp2" host="idp2.aai.niif.hu">
         <dso>
            <persistence>
                <mode>permanent-store</mode>
            </persistence>
         </dso>

         <logs>/var/log/terracotta/server/logs</logs>
         <data>/var/lib/terracotta/server/data</data>
         <statistics>/var/lib/terracotta/server/stats</statistics>
      </server>

   <!-- END Terracotta server definitions -->

   <ha>
      <mode>networked-active-passive</mode>
      <networked-active-passive>
         <election-time></election-time>
      </networked-active-passive>
   </ha>
</servers>

<system>
   <configuration-model>production</configuration-model>
</system>

<clients>
```

```xml
      <logs>/var/log/terracotta/client/logs-%i</logs>
      <statistics>/var/lib/terracotta/client/stats-%i</statistics>
      <modules>
        <module name="tim-vector" version="2.3.1" group-id="org.terracotta.modules"/>
      </modules>
    </clients>
    <application>
      <dso>
        <additional-boot-jar-classes>
          <include>javax.security.auth.Subject</include>
          <include>javax.security.auth.Subject$SecureSet</include>
          <include>javax.security.auth.x500.X500Principal</include>
          <include>javax.security.auth.kerberos.KerberosPrincipal</include>
        </additional-boot-jar-classes>
        <roots>
          <root>
            <root-name>storageService</root-name>
            <field-name>edu.internet2.middleware.shibboleth.common.util.EventingMapBasedStorageService.store</field-name>
          </root>
        </roots>
        <instrumented-classes>
          <include>
            <class-expression>edu.vt.middleware.ldap.jaas.LdapPrincipal</class-expression>
            <honor-transient>true</honor-transient>
          </include>
          <include>
            <class-expression>edu.internet2.middleware.shibboleth.idp.authn.UsernamePrincipal</class-expression>
            <honor-transient>true</honor-transient>
          </include>
          <include>
            <class-expression>edu.vt.middleware.ldap.jaas.LdapCredential</class-expression>
            <honor-transient>true</honor-transient>
          </include>
          <include>
            <class-expression>edu.internet2.middleware.shibboleth.idp.authn.AuthenticationException</class-expression>
            <honor-transient>true</honor-transient>
          </include>
          <include>
```

```xml
        <class-expression>org.opensaml.util.storage.AbstractExpiringObject</class-expression>
        <honor-transient>true</honor-transient>
      </include>
      <include>
        <class-expression>edu.internet2.middleware.shibboleth.common.attribute.resolver.provider.attributeDefinition.TransientIdEntry</class-expression>
        <honor-transient>true</honor-transient>
      </include>
      <include>
        <class-expression>edu.internet2.middleware.shibboleth.idp.authn.LoginContextEntry</class-expression>
        <honor-transient>true</honor-transient>
      </include>
      <include>
        <class-expression>edu.internet2.middleware.shibboleth.idp.authn.LoginContext</class-expression>
        <honor-transient>true</honor-transient>
      </include>
      <include>
        <class-expression>edu.internet2.middleware.shibboleth.idp.authn.ShibbolethSSOLoginContext</class-expression>
        <honor-transient>true</honor-transient>
      </include>
      <include>
        <class-expression>edu.internet2.middleware.shibboleth.idp.authn.Saml2LoginContext</class-expression>
        <honor-transient>true</honor-transient>
      </include>
      <include>
        <class-expression>edu.internet2.middleware.shibboleth.idp.session.impl.AuthenticationMethodInformationImpl</class-expression>
        <honor-transient>true</honor-transient>
      </include>
      <include>
        <class-expression>org.opensaml.util.storage.ReplayCacheEntry</class-expression>
        <honor-transient>true</honor-transient>
      </include>
      <include>
        <class-
```

```xml
expression>edu.internet2.middleware.shibboleth.idp.session.impl.SessionManagerEntry</class-expression>
          <honor-transient>true</honor-transient>
        </include>
        <include>
          <class-
expression>edu.internet2.middleware.shibboleth.common.session.impl.AbstractSession</class-expression>
          <honor-transient>true</honor-transient>
        </include>
        <include>
          <class-expression>edu.internet2.middleware.shibboleth.idp.session.impl.SessionImpl</class-
expression>
          <honor-transient>true</honor-transient>
        </include>
        <include>
          <class-
expression>edu.internet2.middleware.shibboleth.idp.session.impl.ServiceInformationImpl</class-expression>
          <honor-transient>true</honor-transient>
        </include>
        <include>
          <class-expression>org.opensaml.common.binding.artifact.BasicSAMLArtifactMapEntry</class-
expression>
          <honor-transient>true</honor-transient>
        </include>
        <include>
          <class-expression>org.opensaml.xml.util.LazyList</class-expression>
          <honor-transient>true</honor-transient>
        </include>
      </instrumented-classes>
      <locks>
        <autolock auto-synchronized="false">
          <method-expression>* edu.vt.middleware.ldap.jaas.LdapPrincipal.*(..)</method-expression>
          <lock-level>write</lock-level>
        </autolock>
        <autolock auto-synchronized="false">
          <method-expression>*
edu.internet2.middleware.shibboleth.idp.authn.LoginContext.set*(..)</method-expression>
          <lock-level>write</lock-level>
        </autolock>
        <autolock auto-synchronized="false">
          <method-expression>*
edu.internet2.middleware.shibboleth.idp.authn.LoginContext.get*(..)</method-expression>
```

```xml
      <lock-level>read</lock-level>
    </autolock>
    <autolock auto-synchronized="false">
      <method-expression>* edu.internet2.middleware.shibboleth.idp.authn.ShibbolethSSOLoginContext.set*(..)</method-expression>
      <lock-level>write</lock-level>
    </autolock>
    <autolock auto-synchronized="false">
      <method-expression>* edu.internet2.middleware.shibboleth.idp.authn.ShibbolethSSOLoginContext.get*(..)</method-expression>
      <lock-level>read</lock-level>
    </autolock>
    <autolock auto-synchronized="false">
      <method-expression>* edu.internet2.middleware.shibboleth.idp.authn.Saml2LoginContext.set*(..)</method-expression>
      <lock-level>write</lock-level>
    </autolock>
    <autolock auto-synchronized="false">
      <method-expression>* edu.internet2.middleware.shibboleth.idp.authn.Saml2LoginContext.get*(..)</method-expression>
      <lock-level>read</lock-level>
    </autolock>
    <autolock auto-synchronized="false">
      <method-expression>* edu.internet2.middleware.shibboleth.common.session.impl.AbstractSession.set*(..)</method-expression>
      <lock-level>write</lock-level>
    </autolock>
    <autolock auto-synchronized="false">
      <method-expression>* edu.internet2.middleware.shibboleth.common.session.impl.AbstractSession.get*(..)</method-expression>
      <lock-level>read</lock-level>
    </autolock>
    <autolock auto-synchronized="false">
      <method-expression>* edu.internet2.middleware.shibboleth.idp.session.impl.SessionImpl.set*(..)</method-expression>
      <lock-level>write</lock-level>
    </autolock>
    <autolock auto-synchronized="false">
      <method-expression>* edu.internet2.middleware.shibboleth.idp.session.impl.SessionImpl.get*(..)</method-expression>
      <lock-level>read</lock-level>
```

```
        </autolock>
        <autolock auto-synchronized="false">
          <method-expression>*
edu.internet2.middleware.shibboleth.idp.session.impl.AuthenticationMethodInformationImpl.set*(..)</method-
expression>
          <lock-level>write</lock-level>
        </autolock>
        <autolock auto-synchronized="false">
          <method-expression>*
edu.internet2.middleware.shibboleth.idp.session.impl.AuthenticationMethodInformationImpl.get*(..)</method-
expression>
          <lock-level>read</lock-level>
        </autolock>
        <autolock auto-synchronized="false">
          <method-expression>*
edu.internet2.middleware.shibboleth.idp.session.impl.ServiceInformationImpl.set*(..)</method-expression>
          <lock-level>write</lock-level>
        </autolock>
        <autolock auto-synchronized="false">
          <method-expression>*
edu.internet2.middleware.shibboleth.idp.session.impl.ServiceInformationImpl.get*(..)</method-expression>
          <lock-level>read</lock-level>
        </autolock>
      </locks>
    </dso>
  </application>
</tc:tc-config>
```