Shib2IdpRHEL

Előkészületek

entityID

Tanúsítvány

JDK

https://wiki.shibboleth.net/confluence/display/SHIB2/JVMTuning

```
[idp2:~/java]$ sudo_ssh rpm -Uvh jdk-6u7-linux-i586.rpm
Preparing...                ########################################### [100%]
   1:jdk                    ########################################### [100%]
Unpacking JAR files...
        rt.jar...
        jsse.jar...
        charsets.jar...
        tools.jar...
        localedata.jar...
        plugin.jar...
        javaws.jar...
        deploy.jar...
```

A többi rpm-mel nem törődünk.

Shibboleth Security Provider

Be kell másolni a `lib/shib-jce-1.0.jar` állományt a `$JAVA_HOME/jre/lib/ext` könyvtárba. Ha az `ext/` könyvtár nem létezik, akkor hozzuk létre.

```
cp lib/shib-jce-1.0.jar $JAVA_HOME/jre/lib/ext
```

Ezek után be kell állítani, hogy a JRE használni is tudja ezt a providert. Ehhez a `$JAVA_HOME/jre/lib/security/java.security` fájlban keressük meg az ún. "security provider"-eket, és írjuk hozzá a következő sort:

```
security.provider.7=edu.internet2.middleware.shibboleth.DelegateToApplicationProvider
```

- **Megj.:** a "security.provider." után következő szám mindig a megelőzőnél legyen eggyel nagyobb!

Bouncy Castle JCE

A JVM-mel jövő Java Cryptography Engine (JCE) nem támogatja az összes kriptográfiai algoritmust, amelyre az Identity Providernek szüksége lehet (pl. XML Digital Signature, XML Encryption). A Bouncy Castle JCE ezek mellett még olyan algoritmusokat is tartalmaz (általában hatékonyabb és szabványosabb formában), amelyek benne vannak a Java JCE-ben.

Ehhez először le kell tölteni a Bouncy Castle JCE-t. A JCE állományok a Provider oszlopban, a "Signed Jar Files" részben találhatók. (A nevük `bcprov-jdk-VERZIO.jar`.) Letöltés után a .jar fájlokat a `$JAVA_HOME/jre/lib/ext` könyvtárba kell tenni.

```
wget http://www.bouncycastle.org/download/bcprov-jdk15-141.jar
cp bcprov-jdk15-141.jar $JAVA_HOME/jre/lib/ext
```

Ezek után be kell állítani, hogy a JRE használni is tudja ezt a providert. Ehhez a `$JAVA_HOME/jre/lib/security/java.security` fájlban keressük meg az ún. "security provider"-eket, és írjuk hozzá a következő sort:

```
security.provider.8=org.bouncycastle.jce.provider.BouncyCastleProvider
```

- **Megj.:** a "security.provider." után következő szám mindig a megelőzőnél legyen eggyel nagyobb!

Tomcat

Tomcat 6-ot fogunk használni, AJP connectorral

https://wiki.shibboleth.net/confluence/display/SHIB2/IdPApacheTomcatPrepare

```
#XXX-TODO useradd tomcat
cd /usr/local
tar xzf ~/apache-tomcat-6.0.18.tar.gz
ln -s apache-tomcat-6.0.18 tomcat
cd tomcat
mv conf /etc/tomcat ; ln -s /etc/tomcat conf
mv logs /var/log/tomcat; ln -s /var/log/tomcat logs
mkdir /var/lib/tomcat
mv temp /var/lib/tomcat; ln -s /var/lib/tomcat/temp .
mv webapps /var/lib/tomcat; ln -s /var/lib/tomcat/webapps .
mv work /var/lib/tomcat; ln -s /var/lib/tomcat/work .
chown -R tomcat:tomcat /etc/tomcat /var/log/tomcat /var/lib/tomcat
```

Konfiguráció

A `/etc/tomcat/server.xml` -ben módosítani kell a 8009-es porton figyelő Connectort az alábbira:

```
<!-- Define an AJP 1.3 Connector on port 8009 -->
<Connector  port="8009"
            protocol="AJP/1.3" redirectPort="8443"
            enableLookups="false" tomcatAuthentication="false"
            address="127.0.0.1"/>
```

Init script

Az alábbi házi készítésű init script használható a tomcat elindításához:

```
#!/bin/bash
#
# chkconfig: - 85 15
# processname: tomcat
# description:  Start up the Tomcat servlet engine.

# Source function library.
. /etc/init.d/functions
```

```
PATH=/bin:/usr/bin:/sbin:/usr/sbin
export JAVA_HOME=/usr/java/default
CATALINA_HOME="/usr/local/tomcat"
TOMCAT_USER=tomcat

# Max. heap size: 512 MB
# Memory allowed for the permanent generation object space: 256 MB
export JAVA_OPTS="-Xmx512m -XX:MaxPermSize=256m"

if [`id -u` -ne 0 ](); then
   echo "You need root privileges to run this script"
   exit 1
fi

case "$1" in
 start)
       if [-f $CATALINA_HOME/bin/startup.sh ]();
         then
           echo $"Starting Tomcat"
           su -c "$CATALINA_HOME/bin/startup.sh" -s /bin/bash $TOMCAT_USER
       fi
       ;;
 stop)
       if [-f $CATALINA_HOME/bin/shutdown.sh ]();
         then
           echo $"Stopping Tomcat"
           su -c "$CATALINA_HOME/bin/shutdown.sh" -s /bin/bash $TOMCAT_USER
       fi
       ;;
 restart)
       $0 stop
       sleep 2;
       $0 start
       ;;
 *)
       echo $"Usage: $0 {start|stop|restart}"
       exit 1
       ;;
esac
```

```
exit $?
```

A következő parancsok hatására a Tomcat automatikusan elindul bootoláskor:

```
chkconfig --add tomcat
chkconfig tomcat on
```

Változat #1
cziernorbert hozta létre 2026-04-14 13:22:42 CEST
cziernorbert frissítette 2026-04-17 10:55:53 CEST