Pseudonym identifierare (EPTID)

Detta är en guide för hur man sätter upp sk pseudonyma identifierare för Shibboleth IdP:n. Instruktionerna är baserade på en Ubuntu eller debian-baserad Linux men motsvarande bör funka även på andra unix-varianter och Windows.

En pseudonym identifierare är en permanent, anonym identifierare som är unik för en kombination av IdP, SP och användare. En sådan identfierare kan inte användas för att korrelera information mellan SPer och innehåller heller inte någon persondata. En pseudonym identifierare är oftast lämplig att lämna ut till alla SPer.

> SWAMID rekommenderar att alla IdPer lämnar ut pseudonymer som SAML 2.0 NameID samt som attribut av typen eduPersonTargetedID till alla SPer.
> Instruktionerna nedan åstadkommer precis detta.

Installera mysql

```
# apt-get install mysql-server

.. under installationen sätts ett root-lösenord ..
```

Skapa databas och tabell

Skapa en databas...

```
# mysql -p
... använd lösenordet från installationen
mysql> SET NAMES 'utf8';
SET CHARACTER SET utf8;
CHARSET utf8;
CREATE DATABASE IF NOT EXISTS shibboleth CHARACTER SET=utf8;
USE shibboleth;
Query OK, 0 rows affected (0.00 sec)
```

Skapa en tabell (för versioner från och med Shibboleth 3.2.0)...

```
mysql> CREATE TABLE IF NOT EXISTS shibpid (
  localEntity VARCHAR(255) NOT NULL,
  peerEntity VARCHAR(255) NOT NULL,
  principalName VARCHAR(255) NOT NULL default '',
  localId VARCHAR(255) NOT NULL,
  persistentId VARCHAR(50) NOT NULL,
  peerProvidedId VARCHAR(255) default NULL,
  creationDate timestamp NOT NULL default CURRENT_TIMESTAMP
  on update CURRENT_TIMESTAMP,
  deactivationDate timestamp NULL default NULL,
  KEY persistentId (persistentId),
  KEY persistentId_2 (persistentId, deactivationDate),
  KEY localEntity (localEntity, peerEntity,localId),
  KEY localEntity_2 (localEntity, peerEntity, localId, deactivationDate)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
Query OK, 0 rows affected (0.00 sec)
```

skapa slutligen en user och ge rättigheter på tabellen. Denna user bör ha ett annat lösenord än `hemligt123`.

```
mysql> create user shibboleth identified by 'hemligt123';
Query OK, 0 rows affected (0.00 sec)
mysql> grant ALL on shibboleth.shibpid to 'shibboleth'@'localhost';
Query OK, 0 rows affected (0.00 sec)
```

Installera JDBC-connector

Hämta en JDBC-connector för mysql från http://dev.mysql.com/downloads/connector/j/ (tex mysql-connector-java-5.1.35.tar.gz). Packa upp i lämplig katalog och kopiera jar-filen (tex mysql-connector-java-5.1.35-bin.jar) till `lib`-katalogen för binär-paketet till shibboleth. Detta är katalogen med en `install.sh` och `install.bat`. Kör sedan `install.sh} (eller {{instal.bat` om du använder Windows) för att skapa en ny version av `idp.war` med mysql-connectorn instoppad. Starta sedan om din servlet-motor.

```
# cp mysql-connector-java-5.1.13-bin.jar /opt/shibboleth-idp/edit-webapp
/WEB-INF/lib/
```

> När du kör install.sh så får du frågan om du vill skriva över den existerande installationen. Svara 'N' (nej) på den frågan - annars försvinner dina inställningar. En ny war-fil skapas alltid.

Studera loggarna för din servlet-motor samt idp (idp-process.log). Om du får felmeddelanden om att det inte går att hitta mysql-connectorn i classpath så har du misslyckats med att installera connectorn - gå tillbaka och kontrollera att en ny `idp.war` faktiskt skapades.

Skapa DataConnector

Skapa följande `DataConnector` i conf/`attribute-resolver.xml`:

```
<resolver:DataConnector id="StoredId"
            xsi:type="StoredId"
            xmlns="urn:mace:shibboleth:2.0:resolver:dc"
            generatedAttributeID="persistentId"
            sourceAttributeID="uid"
            salt="large random salt value">
               <resolver:Dependency ref="uid" />
        <dc:BeanManagedConnection>MyGlobalDataSource</dc:
BeanManagedConnection>
</resolver:DataConnector>
```

Ersätt "large random salt value" med ett stort (mellan 16 och 48 tecken) långt slumpmässigt lösenord. Ett sätt att generera ett sådant är programmet `apg` eller följande kommando:

```
# openssl rand -base64 36 2>/dev/null
```

Detta lösenord är mycket viktigt att spara - om det går förlorat eller behöver ändras kommer alla pseudonymer att ändras vilket betyder att alla SPer kommer att uppfatta inloggningar som "nya".

Skapa följande bean i conf/global.xml

```
<!-- A Global DataSource for use in the attribute-resolver.xml for DB
connectivity -->
<bean id="MyGlobalDataSource" class="org.apache.commons.dbcp2.
BasicDataSource"
    p:driverClassName="com.mysql.jdbc.Driver"
    p:url="jdbc:mysql://127.0.0.1:3306/shibboleth?autoReconnect=true&amp;
localSocketAddress=127.0.0.1&amp;connectTimeout=1800&amp;
initialTimeout=2&amp;logSlowQueries=true&amp;autoReconnectForPools=true"
    p:username="shibboleth"
    p:password="hemligt123"
    p:maxIdle="5"
    p:maxWaitMillis="15000"
    p:testOnBorrow="true"
    p:validationQuery="select 1"
    p:validationQueryTimeout="5" />
</beans>
```

Attribut-definitioner

Skapa nu följande attribut-definitioner i `attribute-resolver.xml`. Det första är legacy-attributet `eduPersonTargetedID`. SWAMID rekommenderar att detta görs tillgängliga till alla SPer.

```
<resolver:AttributeDefinition xsi:type="ad:SAML2NameID" id="
eduPersonTargetedID"                                    nameIdFormat="urn:
oasis:names:tc:SAML:2.0:nameid-format:persistent" sourceAttributeID="
persistentId">        <resolver:Dependency ref="StoredId"
/>        <resolver:AttributeEncoder xsi:type="enc:SAML1XMLObject" name="
urn:oid:1.3.6.1.4.1.5923.1.1.1.10" />        <resolver:AttributeEncoder
xsi:type="enc:SAML2XMLObject" name="urn:oid:
1.3.6.1.4.1.5923.1.1.1.10" friendlyName="eduPersonTargetedID" /></resolver:
AttributeDefinition>
```

Attribute-release

> Detta kommer att göra pseudonymer tillgängliga för alla SP:er vilket är SWAMIDs rekommendation. Om du inte vill lämna ut pseudonyer till alla så måste du ändra `PolicyRequirementRule` nedan

SWAMID rekommenderar att detta attribut releasas till alla SP:er. Detta gör man enklast genom följande entry i `attribute-filter.xml`:

```
<AttributeFilterPolicy id="releasePermanentIdToAnyone">
        <PolicyRequirementRule xsi:type="ANY" />
        <!-- Jan 2016: as of Shib3.2.1 persistentId is automatically
released and processed in saml-nameid.xml -->

        <AttributeRule attributeID="eduPersonTargetedID">
                <PermitValueRule xsi:type="ANY" />
        </AttributeRule>
</AttributeFilterPolicy>
```

Test

Så här ska en persistent_id ser ut:

```
https://idp-test.kau.se/idp/shibboleth!https://sp.swamid.se/shibboleth!
mbz5i2+tqo7PT4hlmeNrHpYCBdo=
```