My First SP - Shibboleth

Shibboleth is a free, open-source web single sign-on system with rich attribute-exchange based on open standards, principally SAML. It supports both Apache (on several platforms, notably Linux, OSX, Solaris, and Windows), and several versions of Microsoft IIS (5, 6, 7).

Because Shibboleth functions at the level of the http server, no language-specific configuration is required. Therefore, any web application can use Shibboleth for authentication, regardless whether it is written in PHP, Perl, Java, VB.NET, or C#.NET, etc. In some cases, however, it might be required to use Apache (or IIS) as a reverse proxy to shield the actual web application (which might for example be running on Apache Tomcat). An example setup for this scenario is described elsewhere.

In this document, we will set up Shibboleth for use with Apache and connect it to SURFconext. We use a recent Debian GNU/Linux system, but instructions should carry over to other UNIX-like system pretty straightforwardly. The Shibboleth Wiki describes installation of Shibboleth on other systems and platforms more extensively.

In this document, we will show you step by step how to set up an SP. As an example, we will use the test-SP My First SP at https://mfsp.gadgets.surfconext.nl/ . **For your own SP, you will have to change the configuration examples to match you local setup!**

If you encounter any problems while following these instructions, please feel free to contact us at support@surfconext.nl.

A security checklist for Shibboleth can be found here.

SURFconext Metadata

Take note that the metadata and the metadata locations used for the test and production environments of SURFconext differ. Use them accordingly:

- **Test:** https://metadata.test.surfconext.nl/idp-metadata.xml
- **Production:** https://metadata.surfconext.nl/idp-metadata.xml

Setting up Shibboleth

Start by setting up Apache as you normally would. The SP to connect to SURFconext should be using HTTPS with valid certificates (self-signed certificates do not suffice). An example configuration file for the SP could look like this:

```
<VirtualHost _default_:443>
        Servername mfsp.gadgets.surfconext.nl
        ServerAdmin bas.zoetekouw@surfnet.nl

        DocumentRoot /var/www/mfsp
        <Directory />
                Options FollowSymLinks
                AllowOverride None
        </Directory>
        <Directory /var/www/>
                Options Indexes FollowSymLinks MultiViews
                AllowOverride None
        </Directory>

        SSLEngine on
        SSLCertificateFile    /etc/ssl/certs/yourhost.crt
        SSLCertificateKeyFile /etc/ssl/private/yourhost.key
</VirtualHost>
```

Then, install Shibboleth. In Debian and Ubuntu, the package is called libapache2-mod-shib (or libapache2-mod-shib2 in older releases), and simply apt-getting will work fine.

Shibboleth consists of two parts: a daemon (shibd) that handles communication with the SP and IdPs, and an Apache module that handles the authentication in the web server. Make sure that the daemon is running, and that the Apache module is loaded (a2enmod shib; apachectl -k graceful).

If everything is set up correctly, you should be able to reach https://mfsp.gadgets.surfconext.nl/Shibboleth.sso/Status (substitute your local host name, obviously). This should show Shibboleth status information in XML form. Note that this link will only work from a remote machine if you modify the access control list (acl) attribute of the <Handler type="Status"> entry in the `/etc/shibboleth/shibboleth2.xml` file (the file is named `shibboleth2.xml` also in version 3 and later).
The file should look like this:

```
<StatusHandler time="2011-10-14T14:06:55Z">
  <Version Xerces-C="3.2.1" XML-Tooling-C="3.0.3" XML-Security-C="2.0.2" OpenSAML-C="3.0.0" Shibboleth="
3.0.3"/>
  <NonWindows sysname="Linux" nodename="mfsp" release="3.10.0-862.14.4.el7.x86_64" version="#1 SMP Wed Sep
26 15:12:11 UTC 2018" machine="i686"/>
  <SessionCache>
    <OK/>
  </SessionCache>
  <Application id="default" entityID="https://sp.example.org/shibboleth"/>
  <Handlers>
    <Handler type="ArtifactResolutionService" Location="/Artifact/SOAP" Binding="urn:oasis:names:tc:SAML:2.0:
bindings:SOAP"/>
    <Handler type="AssertionConsumerService" Location="/SAML2/POST" Binding="urn:oasis:names:tc:SAML:2.0:
bindings:HTTP-POST"/>
    <Handler type="AssertionConsumerService" Location="/SAML2/POST-SimpleSign" Binding="urn:oasis:names:tc:
SAML:2.0:bindings:HTTP-POST-SimpleSign"/>
    <Handler type="AssertionConsumerService" Location="/SAML2/Artifact" Binding="urn:oasis:names:tc:SAML:2.0:
bindings:HTTP-Artifact"/>
    <Handler type="AssertionConsumerService" Location="/SAML2/ECP" Binding="urn:oasis:names:tc:SAML:2.0:
bindings:PAOS"/>
    <Handler type="AssertionConsumerService" Location="/SAML/POST" Binding="urn:oasis:names:tc:SAML:1.0:
profiles:browser-post"/>
    <Handler type="AssertionConsumerService" Location="/SAML/Artifact" Binding="urn:oasis:names:tc:SAML:1.0:
profiles:artifact-01"/>
    <Handler type="SessionInitiator" Location="/Login"/>
    <Handler type="SingleLogoutService" Location="/SLO/SOAP" Binding="urn:oasis:names:tc:SAML:2.0:bindings:
SOAP"/>
    <Handler type="SingleLogoutService" Location="/SLO/Redirect" Binding="urn:oasis:names:tc:SAML:2.0:
bindings:HTTP-Redirect"/>
    <Handler type="SingleLogoutService" Location="/SLO/POST" Binding="urn:oasis:names:tc:SAML:2.0:bindings:
HTTP-POST"/>
    <Handler type="SingleLogoutService" Location="/SLO/Artifact" Binding="urn:oasis:names:tc:SAML:2.0:
bindings:HTTP-Artifact"/>
    <Handler type="LogoutInitiator" Location="/Logout"/>
    <Handler type="MetadataGenerator" Location="/Metadata"/>
    <Handler type="Status" Location="/Status"/>
    <Handler type="Session" Location="/Session"/>
    <Handler type="DiscoveryFeed" Location="/DiscoFeed"/>
  </Handlers>
  <Status>
    <OK/>
  </Status>
</StatusHandler>
```

Setting up your SP in Shibboleth

Next, Shibboleth needs to be set up as an SP. The configuration process is described more elaborately at https://wiki.shibboleth.net/confluence/display/SP3/GettingStarted, but the instructions below should get you up to speed quickly.

Start by generating a new SSL key pair. This key pair will be used to sign metadata and SAML messages that are exchanged between SURFconext and your SP.

```
openssl req -newkey rsa:4096 -new -x509 -days 3652 -nodes -text -out shib.crt -keyout shib.key
```

This will give you a private key (`shib.key`) and a public certificate (`shib.crt`) that are valid for one year. Note that the public certificate does not need to be signed by a CA; the public keys will be exchanged with SURFconext over a secure HTTPS connection, which will need to have a properly signed key.

The public and private keys should be placed in /etc/shibboleth.

Change owner of the private key so shibd can read it:

```
chown _shibd /etc/shibboleth/shib.key
```

Then download the SURFconext metadata signing certificate. The wget command downloads the certificate and writes is to a filename of your choice, e.g:

```
 wget https://metadata.surfconext.nl/SURFconext-metadata-signer.pem --output-document=/etc/shibboleth
/surfconext.pem
```

(Location for the test environment differs.)

Edit `/etc/shibboleth/shibboleth2.xml` and make the following changes:

- Change the `entityID` in the `<ApplicationDefaults>` section to the URI of your SP. This defines the name by which SURFconext will refer to your SP. The value should be a proper URL, for example

```
<ApplicationDefaults entityID="https://mfsp.gadgets.surfconext.nl/shibboleth"
                     REMOTE_USER="eppn persistent-id targeted-id">
```

- In the `<ApplicationDefaults>` section, add the names of the key and certificate file that you have just created.

```
<CredentialResolver type="File" key="shib.key" certificate="shib.crt"/>
```

- Inside the `<ApplicationDefaults>` section, add a MetadataProvider for SURFconext. This tells Shibboleth where to find SURFconext's SAML metadata:

```
      <MetadataProvider type="XML"
                        url="https://metadata.surfconext.nl/idp-metadata.xml"
                        backingFilePath="metadata-surfconext.xml"
                        reloadInterval="7200">
        <MetadataFilter type="Signature" certificate="surfconext.pem"/>
      </MetadataProvider>
```

- Inside the `<ApplicationDefaults>` section should be a `<Sessions>` section. In there, add a Single Sign-On entry for SURFconext. This tells Shibboleth that SURFconext users can use Single Sign-On and that authentication information with SURFconext should be exchanged using SAML2.

```
<SSO entityID="https://engine.surfconext.nl/authentication/idp/metadata">SAML2</SSO>
```

- Inside the `<ApplicationDefaults>` section should be a `<Sessions>` section. In that section. a `MetadataGenerator` handler should be defined. Here, you need to add additional information about your service and your organization. Edit the section to look like this:

```
 <Handler type="MetadataGenerator" Location="/Metadata" signing="true">
    <mdui:UIInfo>
        <mdui:DisplayName xml:lang="nl">Voorbeelddienst</mdui:DisplayName>
        <mdui:DisplayName xml:lang="en">Example Service</mdui:DisplayName>
        <mdui:Description xml:lang="nl">Een mooie voorbeelddienst om te laten zien hoe Shibboleth
werkt</mdui:Description>
        <mdui:Description xml:lang="en">A nice example Service to show how to work with Shibboleth
and SURFconext</mdui:Description>
        <mdui:Logo height="300" width="500">https://plaatjes.example.com/media/plaatje.png</mdui:Logo>
    </mdui:UIInfo>
    <md:Organization>
        <md:OrganizationName xml:lang="nl">Voorbeeld BV</md:OrganizationName>
        <md:OrganizationName xml:lang="en">Example BV</md:OrganizationName>
        <md:OrganizationDisplayName xml:lang="nl">Voorbeeld</md:OrganizationDisplayName>
        <md:OrganizationDisplayName xml:lang="en">Example</md:OrganizationDisplayName>
        <md:OrganizationURL xml:lang="nl">http://www.example.org</md:OrganizationURL>
        <md:OrganizationURL xml:lang="en">http://www.exampler.org/en</md:OrganizationURL>
    </md:Organization>
    <md:ContactPerson contactType="support">
        <md:GivenName>Piet</md:GivenName>
        <md:SurName>Jansen</md:SurName>
        <md:EmailAddress>piet.Jansen@example.org</md:EmailAddress>
    </md:ContactPerson>
    <md:ContactPerson contactType="technical">
        <md:GivenName>Klaas</md:GivenName>
        <md:SurName>Jansen</md:SurName>
        <md:EmailAddress>klaas.jansen@example.org</md:EmailAddress>
    </md:ContactPerson>
    <md:ContactPerson contactType="administrative">
        <md:GivenName>Jans</md:GivenName>
        <md:SurName>Jansen</md:SurName>
        <md:EmailAddress>jans.jansen@example.org</md:EmailAddress>
    </md:ContactPerson>
</Handler>
```

Additionally, make sure the `md` and `mdui` xml namespaces are defined in the `<SPConfig>` tag on the top of `shibboleth.xml`:

```
<SPConfig xmlns="urn:mace:shibboleth:2.0:native:sp:config"
    xmlns:conf="urn:mace:shibboleth:2.0:native:sp:config"
    xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
    xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
    xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata"
    xmlns:mdui="urn:oasis:names:tc:SAML:metadata:ui"
    clockSkew="180">
```

Result

After these changes, the `shibboleth2.xml` file should look like this:

```
<SPConfig xmlns="urn:mace:shibboleth:2.0:native:sp:config"
  xmlns:conf="urn:mace:shibboleth:2.0:native:sp:config"
  xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
  xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
  xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata"
  clockSkew="180">

  <ApplicationDefaults entityID="https://mfsp.gadgets.surfconext.nl/shibboleth"
            REMOTE_USER="eppn persistent-id targeted-id">
    <Sessions lifetime="28800" timeout="3600" checkAddress="false" relayState="ss:mem" handlerSSL="false">
      <SSO entityID="https://engine.surfconext.nl/authentication/idp/metadata">SAML2</SSO>
      <Logout>SAML2 Local</Logout>
      <Handler type="MetadataGenerator" Location="/Metadata" signing="false"/>
      <Handler type="Status"            Location="/Status"/>
      <Handler type="Session"           Location="/Session" showAttributeValues="false"/>
      <Handler type="DiscoveryFeed"     Location="/DiscoFeed"/>
    </Sessions>
    <Errors supportContact="root@localhost" logoLocation="/shibboleth-sp/logo.jpg" styleSheet="/shibboleth-sp
/main.css"/>
    <MetadataProvider type="XML" uri="https://metadata.surfconext.nl/idp-metadata.xml" backingFilePath="
metadata-surfconext.xml" reloadInterval="7200">
      <MetadataFilter type="RequireValidUntil" maxValidityInterval="172800"/>
      <MetadataFilter type="Signature" certificate="surfconext.pem"/>
    </MetadataProvider>
    <AttributeExtractor type="XML" validate="true" path="attribute-map.xml"/>
    <AttributeResolver type="Query" subjectMatch="true"/>
    <AttributeFilter type="XML" validate="true" path="attribute-policy.xml"/>
    <CredentialResolver type="File" key="shib.key" certificate="shib.crt"/>
  </ApplicationDefaults>
  <SecurityPolicyProvider type="XML" validate="true" path="security-policy.xml"/>
  <ProtocolProvider type="XML" validate="true" reloadChanges="false" path="protocols.xml"/>
</SPConfig>
```

Now, restart Shibd. It should tell you (in the logs in /var/log/shibboleth/) that it is fetching metadata from SURFconext. The message you're looking fore will look like this:

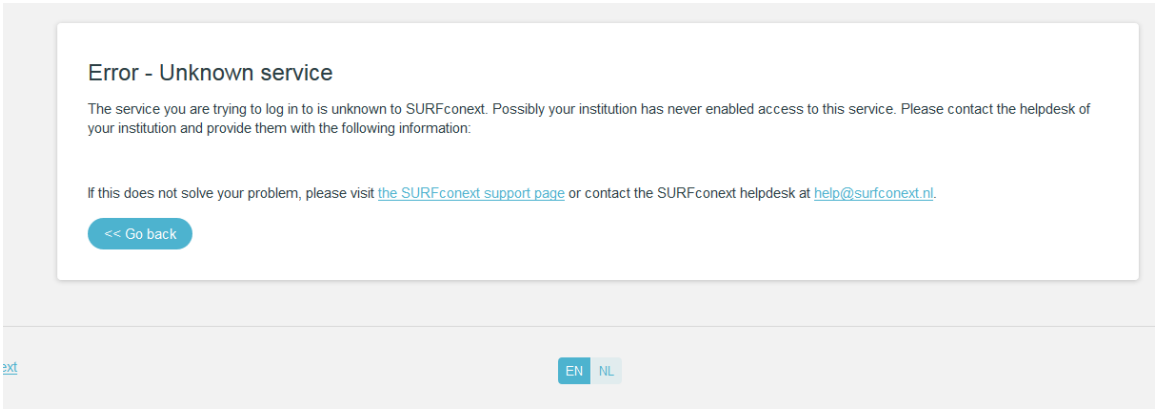
```
2011-10-17 13:49:15 INFO OpenSAML.MetadataProvider.XML : loaded XML resource (https://metadata.surfconext.nl
/idp-metadata.xml)
```

Setting up Apache

Now we need to setup Apache to require Shibboleth-based authentication. Add the following entry to your apache vhost configuration:

```
<Location /secure>
  AuthType shibboleth
  ShibRequestSetting requireSession 1
  require valid-user
</Location>
```

to enable Shibboleth-authentication for all files under https://mfsp.gadgets.surfconext.nl/secure. See the Shibboleth wiki for more information about the available configuration options.
If you visit anything under https://mfsp.gadgets.surfconext.nl/secure., Apache and Shibboleth should now try to authenticate using the SURFconext IdP. As your SP is not yet registered with SURFconext, you should get the following error message:



Connecting your SP

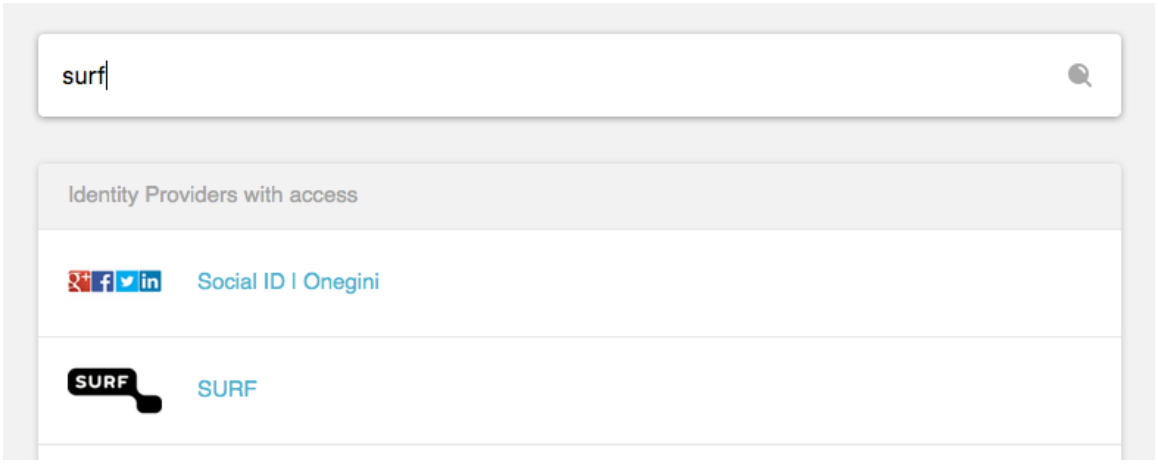
At this point you should contact support@surfconext.nl and request a login for the SP Dashboard. With this Self Service dashboard you will be able to configure a SP on the test environment of SURFconext.

Details which have to be filled out comprise

- the name of your service as will be presented to endusers;
- a description of the functionality your SP offers;
- a link to your SP's metadata (e.g, https://mfsp.gadgets.surfconext.nl/Shibboleth.sso/Metadata);
- A name and email address of a technical contact person responsible for the service;
- the attrubtes you wish to use for this service.

Testing

When your SP has been added to SURFconext, and you visit the secure part of your site (e.g., https://mfsp.gadgets.surfconext.nl/secure), you should be presented with a so-called WAYF (Where Are You From) screen. There, the user can select which IdP he would like to use for authentication. This could look like this:



Which IdPs are shown depends on the configuration of SURFconext. By default only the IdPs that have explicitly allowed access to your service are shown here. In addition, the Guest IdP can be used by other people to login.

After selection, the user is asked to log in at his local institute, and is redirected back to your website. To make the web application aware of how the authentication was handled, Shibboleth sets a number of server variables that can be accessed formt he web application.

To test if everything works correctly, put the following simple PHP script in https://mfsp.gadgets.surfconext.nl/secure/test.php:

```
<html>
<head><title>Shibboleth test</title></head>
<body><pre><?php print_r($_SERVER); ?></pre></body>
</html>
```

The output from this script (after successful authentication) should contain (apart from regular Apache server variables) a number of Shibboleth-specific variables:

```
    [Shib-Application-ID] => default
    [Shib-Session-ID] => _1597c53856f95eb328b232e0dc74702d
    [Shib-Identity-Provider] => https://engine.surfconext.nl/authentication/idp/metadata
    [Shib-Authentication-Instant] => 2011-11-02T10:55:35Z
    [Shib-Authentication-Method] => urn:oasis:names:tc:SAML:2.0:ac:classes:Password
    [Shib-AuthnContext-Class] => urn:oasis:names:tc:SAML:2.0:ac:classes:Password
    [Shib-Session-Index] => 82a98744094c34a839f91800239d7a10
    [persistent-id] => https://engine.surfconext.nl/authentication/idp/metadata!https://mfsp.gadgets.
surfconext.nl/shibboleth! 8e01d4e3965255f4e3beeeae42e84f357fa87a84
```

The meaning of these variables is described at the Shibboleth wiki.

Asking for the right attributes

As you can see in the example above, only a limited number of attributes are passed from Shibboleth to the web server. In order to get additional attributes, you need to edit the file `/etc/shibboleth/attribute-map.xml`. This file defines the mapping between SAML attributes and Apache server variables.

In order to get the default SURFconext attributes, the file `/etc/shibboleth/attribute-map.xml` needs to be replaced with the following:

```
<?xml version="1.0"?>
  <Attributes xmlns="urn:mace:shibboleth:2.0:attribute-map" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <Attribute name="urn:oasis:names:tc:SAML:2.0:nameid-format:persistent" id="persistent-id">
    <AttributeDecoder xsi:type="NameIDAttributeDecoder" formatter="$Name" defaultQualifiers="true"/>
  </Attribute>
  <Attribute name="urn:oid:1.3.6.1.4.1.5923.1.1.1.10"     nameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri" id="Shib-user"/>
  <Attribute name="urn:oid:2.5.4.4"                       nameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri" id="Shib-surName"/>
  <Attribute name="urn:oid:2.5.4.42"                      nameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri" id="Shib-givenName"/>
  <Attribute name="urn:oid:2.5.4.3"                       nameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri" id="Shib-commonName"/>
  <Attribute name="urn:oid:2.16.840.1.113730.3.1.241"     nameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri" id="Shib-displayName"/>
  <Attribute name="urn:oid:0.9.2342.19200300.100.1.3"     nameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri" id="Shib-email"/>
  <Attribute name="urn:oid:1.3.6.1.4.1.25178.1.2.9"       nameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri" id="Shib-HomeOrg"/>
  <Attribute name="urn:oid:1.3.6.1.4.1.25178.1.2.10"      nameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri" id="Shib-HomeOrgType"/>
  <Attribute name="urn:oid:1.3.6.1.4.1.25178.1.2.14"      nameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri" id="Shib-PersonalUnqiueCode"/>
  <Attribute name="urn:oid:1.3.6.1.4.1.5923.1.1.1.1"      nameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri" id="Shib-Affiliation"/>
  <Attribute name="urn:oid:1.3.6.1.4.1.1466.115.121.1.15" nameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri" id="Shib-ScopedAffiliation"/>
  <Attribute name="urn:oid:1.3.6.1.4.1.5923.1.1.1.7"      nameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri" id="Shib-Entitlement"/>
  <Attribute name="urn:oid:1.3.6.1.4.1.5923.1.1.1.6"      nameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri" id="Shib-eduPersonPN"/>
  <Attribute name="urn:oid:1.3.6.1.4.1.5923.1.5.1.1"      nameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri" id="Shib-memberOf"/>
  <Attribute name="urn:oid:0.9.2342.19200300.100.1.1"     nameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri" id="Shib-uid"/>
  <Attribute name="urn:oid:2.16.840.1.113730.3.1.39"      nameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri" id="Shib-language"/>
</Attributes>
```

The meaning of these attributes, as well as examples of what their values look like, can be found here.

With this new attribute mapping in place (and after restarting `shibd`), the server variables displayed by https://mfsp.gadgets.surfconext.nl/secure/test.php should look like:

```
[Shib-Application-ID] => default
[Shib-Session-ID] => _51601ddffbb5537cc24295a8f5804d11
[Shib-Identity-Provider] => https://engine.surfconext.nl/authentication/idp/metadata
[Shib-Authentication-Instant] => 2011-11-02T12:36:03Z
[Shib-Authentication-Method] => urn:oasis:names:tc:SAML:2.0:ac:classes:Password
[Shib-AuthnContext-Class] => urn:oasis:names:tc:SAML:2.0:ac:classes:Password
[Shib-Session-Index] => 82a98744094c34a839f91800239d7a10
[Shib-HomeOrg] => surfnet.nl
[Shib-commonName] => Bas Zoetekouw
[Shib-displayName] => Bas Zoetekouw
[Shib-eduPersonPN] => bas@surfnet.nl
[Shib-email] => Bas.Zoetekouw@surfnet.nl
[Shib-givenName] => Bas
[Shib-surName] => Zoetekouw
[Shib-uid] => bas
[Shib-userStatus] => member
[Shib-memberOf] => urn:collab:org:surf.nl
[Shib-user] => urn:collab:person:surfnet.nl:bas
[persistent-id] => 8e01d4e3965255f4e3beeeae42e84f357fa87a84
```

That's all folks!

At this point, your SP is successfully coupled to SURFconext. Congratulations!