

My First SP - PHP



Please [start here](#) if you want to connect your service to the SURFconext platform

Creating a simple SP from scratch

In this tutorial, we will set up a simple Service Provider, written in PHP, and connect it to the SURFconext infrastructure, using the SimpleSAMLphp product.

The tutorial assumes you are using a Linux environment; however, there is no reason to assume that this approach won't work on different operating systems that support PHP.



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>

Set up the SP

We start with a Hello World php app:

```
<?php

# do nothing for now...
print "Hello world";
```

For this guide, we will assume that the SP will be available from <https://mfsp.example.org/mfsp/test.php>.

In the examples below, you need to replace this with your own location.

Enable federated login

The goal of this tutorial is to enable federated logins via SURFconext for this simple php test file.

SimpleSAMLphp

We are going to use SimpleSAMLphp for this. SimpleSAMLphp is a very powerful product, which can be used in a number of roles in an identity federation (e.g., as IdP, SP, etc). This manual assumes you're using SimpleSAMLphp 2.0 or higher.

To set up SimpleSAMLphp to allow SURFconext users to log in, we need to set up SimpleSAMLphp as a Service Provider. On [the SimpleSAMLphp homepage](#), the configuration process is very well documented.

Below, we will go through the process step by step.

First download and set up SimpleSAMLphp as described on [this page](#). In particular, you need to

- Download and unpack SimpleSAMLphp into `/var/simplesamlphp` (step 4)
- Configure Apache (step 6)
- Configure SimpleSAMLphp (step 7)



Install Apache and PHP as follows:

- `sudo apt update`
- `sudo apt install apache2`
- `sudo apt install php libapache2-mod-php php-mysql php-curl php-xml`

Restart your server after changing stuff:

- `sudo systemctl restart apache2`

Find more information how to setup [Apache in Linux here](#).

Once you've set up SimpleSAMLphp, check that it works by visiting <https://mfsp.example.org/simplesaml/admin>. You will be asked for the admin password that is specified in config.php.


SimpleSAMLphp










English

Configuration Test Federation Log out

SimpleSAMLphp is installed in: /srv/demosp/simplesamlphp/
You are running version 2.0.0.

Modules

You have the following modules installed ( means the module is not enabled):

 SAML 2.0 IdP	 cron	 metarefresh
 admin	 discopower	 multiauth
 core	 exampleauth	 saml

Details

[Diagnostics on hostname, port and protocol](#)
[Information on your PHP installation](#)
[Cron module information page](#)
[Metarefresh](#)

As a simple first check to see if everything is configured correctly so far, check the below half of this page. This should tell you that everything was fine:

[Cron module information page](#)

[Metarefresh](#)

Your PHP installation

✓	required	PHP 7.4 or newer is needed. You are running: 7.4.33
✓	required	Date/Time Extension
✓	required	Hashing function
✓	required	ZLib
✓	required	OpenSSL
✓	required	XML DOM
✓	required	Regular expression support
✓	required	PHP intl extension
✓	required	JSON support
✓	required	Standard PHP library (SPL)
✓	required	Multibyte String extension
✓	optional	cURL (might be required by some modules)
✓	required	Session extension
✓	optional	PDO Extension (required if a database backend is used)
✗	optional	LDAP extension (required if an LDAP backend is used)
✗	optional	redis/predis (required if the redis data store is used)
✗	optional	Memcache or Memcached extension (required if the memcache backend is used)
✓	optional	The technicalcontact_email configuration option should be set
✓	required	The auth.adminpassword configuration option must be set

© 2007-2023 SimpleSAMLphp



Next, we need to configure SimpleSAMLphp as a Service Provider for SURFconext

Setting up an SP

The procedure to set up SimpleSAMLphp as a Service Provider is explained in the [SimpleSAMLphp documentation](#).

We will follow the procedure as laid out there:

- Copy `config/authsources.php.dist` to `config/authsources.php` in the SimpleSAMLphp root (`/var/simplesamlphp`).
- The default config file has a lot of examples. We just want to have a username/password based admin login, and a SAML authentication source (which will be connected to SURFconext).

The authsources config thus becomes:

```
<?php
$config = [
    // This is a authentication source which handles SimpleSAMLphp admin authentication.
    'admin' => [
        'core:AdminPassword',
    ],

    // An authentication source which can authenticate against SURFconext
    'default-sp' => [
        'saml:SP',

        // The entityID is a globally unique identifier for your service and should not
        // change. We recommend to set it to the base URL of your application.
        'entityID' => 'https://mfsp.example.org',

        'idp' => 'https://engine.test.surfconext.nl/authentication/idp/metadata',

        // The entries below are all OPTIONAL but RECOMMENDED to tell SURFconext
        // some details about your service.
        'name' => [
            'en' => 'Name of the Service',
            'nl' => 'Naam van de Dienst',
        ],

        'description' => [
            'en' => 'Description of the Service',
            'nl' => 'Omschrijving van de dienst',
        ],
    ],
];
```



Be aware of the environment your IdP lives in. In the example above the test environment of SURFconext is used. A common mistake is to try to connect to the test environment of SURFconext, and make use of the production environments link. You can find the right entityID inside the metadata for each environment:

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

Next, we need to configure SURFconext as an Identity Provider for your service. To configure this, replace the file `metadata/saml20-idp-remote.php`, which identifies the Identity Providers that your service can use, with the following (for SURFconext TEST):

```
<?php
$metadata['https://engine.test.surfconext.nl/authentication/idp/metadata'] = [
    'name' => [
        'en' => 'SURFconext test',
        'nl' => 'SURFconext test',
    ],
    'SingleSignOnService' => 'https://engine.test.surfconext.nl/authentication/idp/single-sign-on/key:20190208',
    'certificate' => 'engineblock.test.surfconext.nl.20190208.pem',

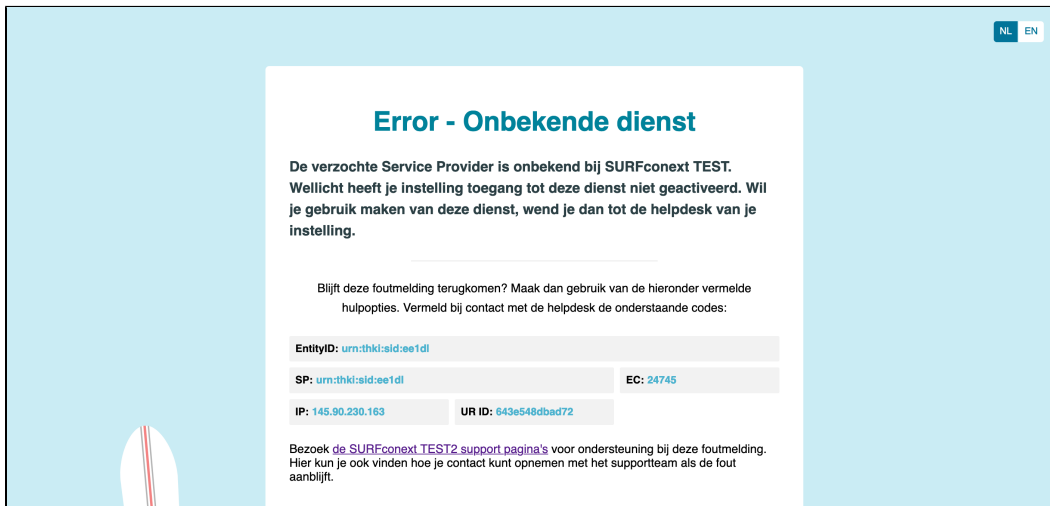
    // convert OID formatted attributes from SAML assertion to 'name' similar to LDAP
    // so they will be available as 'mail' and 'displayName'
    'authproc' => [
        50 => [
            'class' => 'core:AttributeMap', 'oid2name',
        ],
    ],
];
```

Where the `.pem` file is a file placed under the `cert/` directory containing the SAML signing certificate you can find this file at <https://metadata.test.surfconext.nl/> under "Assertion signing certificate".

Testing connection

Now we can test our side of the setup: in the SimpleSAMLphp interface of your SP, go to *Authentication*, *Test configured authentication sources*, and choose *default SP*.

You should then be redirected to SURFconext. If everything is configured correctly, you should get the following error message:



The error occurs because SURFconext does not know your SP yet, and therefore will not allow its users to log onto your SP. However, the fact that you are redirected to SURFconext signifies that the configuration of the SP is correct.

Configuring SURFconext

Next, we need to add the SP to SURFconext. SURFnet doesn't allow just anyone to connect directly to its production platform. To register, put the metadata URL (<https://mfsp.example.org/simplesaml/module.php/saml/sp/metadata.php/default-sp>) in the SP Dashboard form, fill it out and publish the entity to TEST.

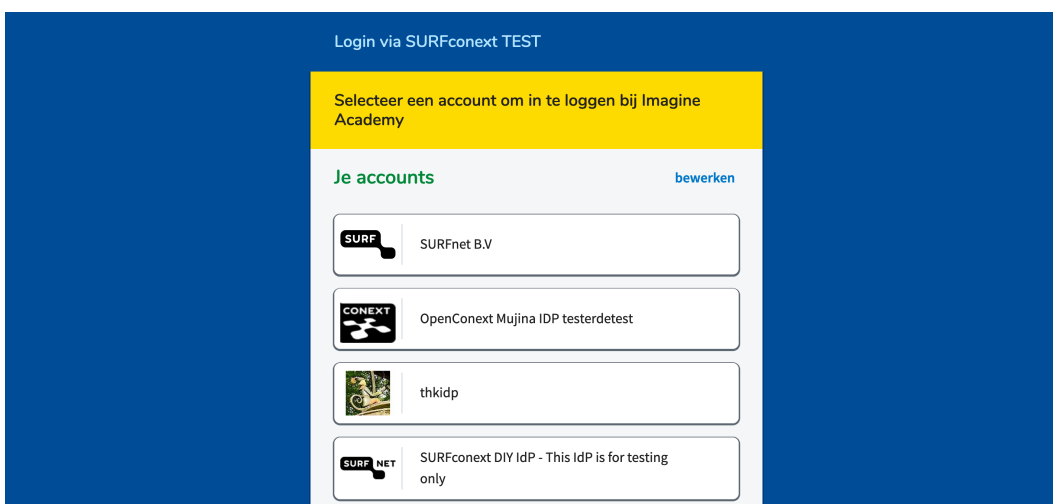
If not using the SP dashboard, send an email to support@surfconext.nl.

Include at least the following information:

- location of your metadata (see above)
- description of your service
- name, email address, and phone number of a technical and an administrative contact at your organization
- contact information for you end user support
- a list of identity providers that need access to your service. For testing purposes, it is common to request access for the [SURFconext DIY IdP](#) only.

The SURFconext administrators will get back to you quickly.

After your SP has been connected, visiting the authentication test (like above), you should get the WAYF (Where Are You From) screen:



After logging in at your IdP, and giving consent for SURFconext to hand your personal information to SURFconext, you will be redirected back to your local SimpleSAMLphp installation, and you should get a page like this, displaying your identity information:

SimpleSAMLphp

English

Configuration

Test

Federation

Log out

Hi, this is the status page of SimpleSAMLphp. Here you can see if your session is timed out, how long it lasts until it times out and all the attributes that are attached to your session.

Your session is valid for 28786 seconds from now.

Your attributes

User ID urn:mace:dir:attribute-def:uid	student1
User ID urn:oid:0.9.2342.19200300.100.1.1	student1
Given name urn:mace:dir:attribute-def:givenName	Student
Given name urn:oid:2.5.4.42	Student
Surname urn:mace:dir:attribute-def:surname	One

Congratulations, you are now connected to SURFconext!

Protecting your PHP application

To protect a PHP application, we simply need to add a call to SimpleSAMLphp in the code. Change the `test.php` which we installed before to the following:

```
<?php
require_once('/var/simplesamlphp/lib/_autoload.php');

$as = new SimpleSAML_Auth_Simple('default-sp');
$as->requireAuth();
$attributes = $as->getAttributes();
?>
<html>
<head><title>My First Service Provider in PHP</title></head>
<body>
<h1>My First SP</h1>
<p>Hello world!</p>

<h2>Your attributes:</h2>
<pre><?php print_r($attributes); ?></pre>

</body></html>
```

When visiting this script, you will now be required to log on via SURFconext.

Further configuration

We recommend you consider [Securing your simpleSAMLphp setup](#).

That's all folks!

Please direct any questions or comments about this document to support@surfconext.nl.