Configuring SimpleSAMLphp as an Identity Provider

Guy Halse https://orcid.org/0000-0002-9388-8592

Prerequisites

This workshop assumes a reasonable understanding of Linux, Apache & PHP (the traditional LAMP server). You should be comfortable working at a Linux command prompt, and should be able to edit files with one of the built-in command-line text editors.

It may also help to have some basic understanding of SAML. The South African Identity Federation (SAFIRE) has made a primer available at https://safire.ac.za/safire/publications/saml-primer/.

Lab environment

The lab for this workshop is a series of pre-configured virtual machines that will be made available to you. Each participant has their own virtual machine, with a hostname matching the form:

***yourfirstname*.lab.safire.ac.za**

Where *yourfirstname* is the first given name that you supplied during registration. If you need to check what yours may be called, you can look at https://lab.safire.ac.za/.

You can log into this machine using SSH:

Username:	*yourfirstname*
Password:	55555

Once logged in, you can set your preferred editor as follows:

```
sudo update-alternatives --config editor
```

In addition to a user account, you also have a user object in a shared LDAP directory server:

LDAP Server:	ldap.lab.safire.ac.za
Base DN:	dc=lab,dc=safire,dc=ac,dc=za
Username:	cn=*yourfirstname*,ou=people,dc=lab,dc=safire,dc=ac,dc=za
Password:	88888

Further details are shown in the message of the day (MOTD).

Screen

We strongly recommend that you use screen(1) when working in the lab. If you've not used screen before, you can start a new screen by typing:

```
screen
```

And you can recover an existing screen (for example, after you've lost a connection and reconnected) by typing:

```
screen -dr
```

Installing SimpleSAMLphp

Instructions for installing SimpleSAMLphp can be found at https://simplesamlphp.org/docs/stable/simplesamlphp-install

On the lab machines, all the pre-requisites have been preinstalled for you and the latest stable version of SimpleSAMLphp has been extracted into `/var/www`

There is one small difference from SimpleSAMLphp's documentation: rather than copying the directory in place, we've symlinked it as `/var/www/simplesamlphp` as we have found this makes future upgrades much simpler.

Configuring Apache

Apache has been pre-configured for you in accordance with SimpleSAMLphp's instructions. You will find the configuration at `/etc/apache2/conf-enabled/simplesamlphp.conf`

The lab environment uses LetsEncrypt to ensure all websites are served over HTTPS.

Configuring SimpleSAMLphp

We're going to be configuring SimpleSAMLphp as an identity provider, loosely following the documentation at https://simplesamlphp.org/docs/stable/simplesamlphp-idp.

Enabling the Identity Provider functionality

The first thing we're going to do is enable the identity provider functionality in SimpleSAMLphp and do some basic configuration:

```
sudo editor /var/www/simplesamlphp/config/config.php
```

In `config`.php, make the following changes:

1. Find the `baseurlpath` and make sure it matches your Apache configuration.
2. Find `technicalcontact_`name and `technicalcontact_email` and set them to match your organisation's name & email address.
3. Find `timezone` and set it to your home timezone (e.g. `Africa/Kampala` – see https://www.php.net/manual/en/timezones.africa.php)
4. Find `secretsalt` and set it to a secure value (see comments above)
5. Find `auth.adminpassword` and change it to a strong password
6. Find `enable.saml20-idp` and set it to `true`
7. Find `theme.header` and set it to your organisation's name. You need to uncomment it.
8. To facilitate easier testing, set `production` to `false`.

You should test that the resulting file has no syntax errors in it:

```
php -l /var/www/simplesamlphp/config/config.php
```

Authentication module

For an identity provider to be useful, it needs to be able to authenticate against some form of database holding information about your users. This could be your campus Active Directory, a Google GSuite domain, a SQL database, or an LDAP server.

The lab environment has a central LDAP server available, and we're going to be using this as our source of truth for user information.

```
sudo editor /var/www/simplesamlphp/config/authsources.php
```

In authsources.php, make the following changes:

1. Find the example-ldap configuration block and make sure it is enabled (uncommented).
2. Set the LDAP hostname to `ldap.lab.safire.ac.za`
3. Ensure `enable_tls` is set to `true`
4. Set `search.enable` to `true`
5. Set `search.base` to the base DN (i.e `dc=lab,dc=safire,dc=ac,dc=za here`)
6. Set `search.attributes` to `['eduPersonPrincipalName', 'uid', 'SAMAccountName', 'userPrincipalName', 'mail', 'mailNickname'],`
7. Set `search.filter` (uncommented) to `'(|(objectClass=inetorgperson)(objectClass=user))',`
8. Set `search.username` to a user that can search (e.g. `cn=manager,dc=lab,dc=safire,dc=ac,dc=za`)
9. Set `search.password` to the account's password (e.g. 99999)

You should test that the resulting file has no syntax errors in it:

```
php -l /var/www/simplesamlphp/config/authsources.php
```

At this stage you should be able to test you have authentication working correctly by visiting https://*yourfirstname*.lab.safire.ac.za/simplesaml/module.php/admin/test

Creating a self signed certificate

This has already been done for you in `/var/www/simplesamlphp/cert/server.crt`.

Note that for simplicity's sake, the private key for this certificate is not encrypted – in a production environment you should encrypt your private key and set the password in the appropriate place in metadata.

Configuring the IdP

There is full documentation of the saml20-idp-hosted.php file at https://simplesamlphp.org/docs/stable/simplesamlphp-reference-idp-hosted.

To simplify the configuration of an identity provider, we've use the South African Identity Federation's template, available from https://safire.ac.za/technical/resources/configuring-simplesamlphp-for-safire/#configure-a-hosted-idp. This is a more complete version than the minimal example provided in the documentation that should satisfy most SAML2Int needs. However, you may need to edit some values to comply with your federation policy.

```
sudo editor /var/www/simplesamlphp/metadata/saml20-idp-hosted.php
```

While the template we're using is reasonably complete, there are a number of changes you will need to make in saml20-idp-hosted.php:

1. Change `auth` to the authsource you configured previously (i.e. `example-ldap`)
2. Change the `scope` array to match our LDAP directory. All values in the directory are scoped as `lab.safire.ac.za`.
3. Update the `OrganizationName`, `OrganizationDisplayName`, `OrganizationURL`, and `UIInfo` to reflect your own organisation. Feel free to remove the Afrikaans (af) translations.
4. Update the `contacts` to reflect accurate contact information for your organisation. Remember we already defined a technical contact in config.php.

You should test that the resulting file has no syntax errors in it:

```
php -l /var/www/simplesamlphp/metadata/saml20-idp-hosted.php
```

At this stage you should be able to view the resulting identity provider metadata at https://*yourfirstname*.lab.safire.ac.za/simplesaml/module.php/admin/federation.

Using the uri NameFormat on attributes

`attributes.NameFormat` was already set in the template we used in the previous step. However, the authentication processing filter (authproc) rules still need to be configured.

> You can find more information on authentication processing filters at https://simplesamlphp.org/docs/stable/simplesamlphp-authproc

There are two places you can configure authproc filter rules. Either in individual metadata, or more generically in config.php. While SimpleSAMLphp's documentation suggests doing this in the IdP metadata, we recommend centralising your authproc filters in config.php.

```
sudo editor /var/www/simplesamlphp/config/config.php
```

Find the `authproc.idp` config stanza. This controls the authproc filters for all configured identity providers (just as `authproc.sp` controls them for all service providers). Then do the following:

1. Below the `core:LanguageAdaptor` entry, add:

   ```
   100 => ['class' => 'core:AttributeMap', 'name2oid'],
   ```

2. Below the `core:TargetedID` entry add:

   ```
   20 => ['class' => 'saml:TransientNameID',],
   21 => ['class' => 'saml:PersistentNameID', 'attribute' => '
   eduPersonPrincipalName',],
   22 => ['class' => 'saml:PersistentNameID2TargetedID', 'attribute' =>
   'eduPersonTargetedID', 'nameId' => true,],
   ```

3. You can add any other attribute transforms you might need at an appropriate place. For instance, to generated eduPersonScopedAffiliation, you could do:

   ```
   40 => [
     'class' => 'core:ScopeAttribute',
     'scopeAttribute' => 'eduPersonPrincipalName',
     'sourceAttribute' => 'eduPersonAffiliation',
     'targetAttribute' => 'eduPersonScopedAffiliation',
   ],
   ```

   Similarly to add a static attribute such as schacHomeOrganization you could do:

   ```
   41 => [
     'class' => 'core:AttributeAdd',
     'schacHomeOrganization' => 'lab.safire.ac.za',
     'schacHomeOrganizationType' =>
   'urn:schac:homeOrganizationType:int:university',
   ],
   ```

4. If you use a service that needs an email address as a NameID (e.g. Google GSuite), then you could do:

   ```
   23 => [
     'class' => 'saml:AttributeNameID',
     'attribute' => 'mail',
     'Format' => 'urn:oasis:names:tc:SAML:1.1:nameid-
   format:emailAddress',
   ],
   ```

You should test that the resulting file still has no syntax errors in it:

```
php -l /var/www/simplesamlphp/config/authsources.php
```

Adding SPs to the IdP

By far the most common way to add service providers to an identity provider is to get dynamic metadata from your home federation. This requires the metarefresh module to automatically fetch and maintain metadata:

There's detailed documentation on automated metadata management at https://simplesamlphp.org/docs/1.17/simplesamlphp-automated_metadata

First enable the cron and metarefresh modules:

```
sudo touch /var/www/simplesamlphp/modules/cron/enable
sudo cp /var/www/simplesamlphp/modules/cron/config-templates/*.php
/var/www/simplesamlphp/config/
sudo touch /var/www/simplesamlphp/modules/metarefresh/enable
sudo cp /var/www/simplesamlphp/modules/metarefresh/config-
templates/*.php /var/www/simplesamlphp/config/
```

Then configure metarefresh:

```
sudo editor /var/www/simplesamlphp/config/config-metarefresh.php
```

This should then be configured in accordance with your federation's guidelines. For instance, SAFIRE publishes instructions at https://safire.ac.za/technical/resources/configuring-simplesamlphp-for-safire/#configure-metarefresh.

In this lab environment, there's some preconfigured metadata for you in `/var/www/simplesamlphp/metadata/saml20-sp-remote.php`.

Adding this IdP to other SPs

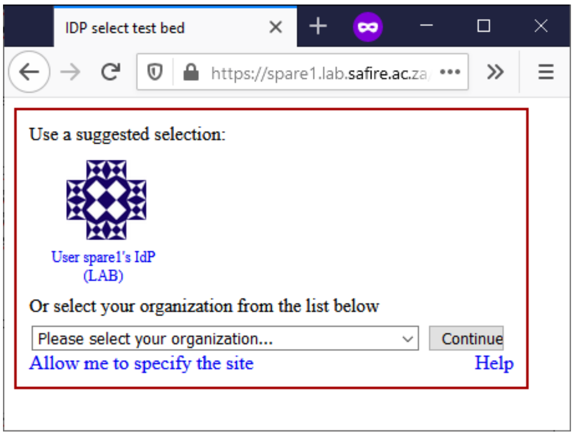
To add your identity provider to your federation or to other SPs you need to exchange metadata. You can find the metadata for your lab identity provider at https://*yourfirstname*.lab.safire.ac.za/simplesaml/saml2/idp/metadata.php?output=xhtml

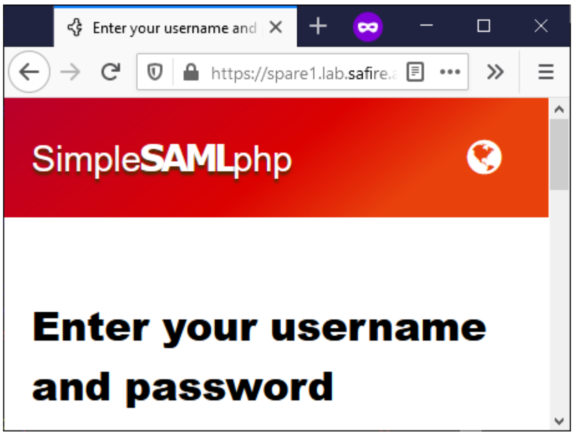
Testing the IdP

To allow you to test your (and other people's) identity providers, the lab includes a simple test service provider. This is configured to use Shibboleth and the Shibboleth embedded discovery service, and is prepopulated with the lab's metadata. You can test this at:

https://*yourfirstname*.lab.safire.ac.za/secure/

If everything is working, you should be presented with a discovery interface where you can select your and your fellow participant's identity providers.


IDP select test bed
https://spare1.lab.safire.ac.za
Use a suggested selection:
User spare1's IdP (LAB)
Or select your organization from the list below
Please select your organization...
Continue
Allow me to specify the site
Help



Once logged in, you should be presented with a page that shows the CGI environment of your web server. This will include any attributes received from the identity provider (look, for example, for `affiliation`).

Incognito / private window

Both SimpleSAMLphp and Shibboleth native SP make use of cookies to track sessions. This is important for making single-sign-on (SSO) work properly, but makes testing harder.

For this reason, it is recommended that you yours your browser's private browsing mode to control what cookies are sent. This is variously called incognito window (Chrome) or private window (Firefox, Microsoft Edge).