PoC – Keycloak & SimpleSAMLphp

Installing Keycloak

1. Download [Keycloak](https://www.keycloak.org/downloads.html) (<https://www.keycloak.org/downloads.html>) (keycloak-X.X.X.Final.[zip|tar.gz]) and unpack the file in the desired directory.
2. Edit the file *standalone/configuration/standalone.xml* and change the following line to the port: 8180, to avoid conflicts with the running tomcat (*Activiti*).

<socket-binding name="http" port="${jboss.http.port:8180}"/>

1. Run *Keycloak* in "Standalone" mode:

sh /bin/standalone.sh

1. Create an admin account:  
   <https://www.keycloak.org/docs/3.2/getting_started/topics/first-boot/initial-user.html>
2. Verify you can access *Keycloak* in the browser by using localhost and the port you configured above.

**Note:** Details about more installation options can be found [here](https://www.keycloak.org/docs/2.5/server_installation/topics/installation.html) (<https://www.keycloak.org/docs/2.5/server_installation/topics/installation.html>).

Configuring Activiti to use Keycloak for Authentication

Configuring Activiti

1. Edit the following file, located within the *Activiti* installation dir: *tomcat/webapps/activiti-app/WEB-INF/classes/META-INF/activiti-app/activiti-identity-service.properties*
2. Make sure the content of the file is the following:

# --------------------------

# IDENTITY SERVICE

# --------------------------

keycloak.enabled=true

keycloak.realm=alfresco

keycloak.auth-server-url=http://localhost:8180/auth

keycloak.ssl-required=none

keycloak.resource=alfresco

keycloak.principal-attribute=email

# set to true if access type is public for this client in keycloak

keycloak.public-client=true

# set secret key if access type is not public for this client in keycloak

#keycloak.credentials.secret=

keycloak.always-refresh-token=true

keycloak.autodetect-bearer-only=true

keycloak.token-store=cookie

keycloak.enable-basic-auth=true

1. Restart Activiti

Configuring Keycloak

1. Create a realm called "alfresco"
2. Under the new realm, create a "Client" with the following settings: *Client ID:*alfresco *Enabled:*On *Client Protocol:*openid-connect *Access Type:*public  
   *Valid Redirect URIs:* \*
3. Create a user (using an email that matches an already existing user within*Activiti*).
4. Try going to*Activiti* using the browser (you should be redirected to the*Keycloak* authentication interface).
5. Sign in using the credentials for the user you just configured (the one that has an email of an already existing *Activiti* user).
6. Confirm that you can successfully access *Activiti* using *Keycloak* for authentication.

Installing SimpleSAMLphp

1. Download it [here](https://simplesamlphp.org/download) (<https://simplesamlphp.org/download>) (latest stable version).
2. Follow the configuration steps [here](https://simplesamlphp.org/docs/stable/simplesamlphp-install#section_8) (<https://simplesamlphp.org/docs/stable/simplesamlphp-install#section_8>).
3. Configure *Apache*  
   \*Add the virtual host:

<VirtualHost \*:80>

        DocumentRoot "/PATH-TO-LOCAL-INSTALLATION/simplesamlphp/www"

        ServerName simplesaml-test.local

        ErrorLog "/private/var/log/apache2/simplesaml-test.local-error\_log"

        CustomLog "/private/var/log/apache2/simplesaml-test.local-access\_log" common

        SetEnv SIMPLESAMLPHP\_CONFIG\_DIR /PATH-TO-LOCAL-INSTALLATION/simplesamlphp/config

        Alias /simplesaml /PATH-TO-LOCAL-INSTALLATION/simplesamlphp/www

        <Directory "/PATH-TO-LOCAL-INSTALLATION/simplesamlphp/www">

            <IfModule !mod\_authz\_core.c>

            # For Apache 2.2:

            Order allow,deny

            Allow from all

            </IfModule>

            <IfModule mod\_authz\_core.c>

            # For Apache 2.4:

            Require all granted

            </IfModule>

        </Directory>

</VirtualHost>

Note: Replace the"**/**PATH-TO-LOCAL-INSTALLATION/" with the path to the local installation directory.

\*Enable php in Apache:  
Edit the file: ***/etc/apache2/httpd.conf*** and make sure the following lines are uncommented:

LoadModule php7\_module libexec/apache2/libphp7.so

LoadModule proxy\_html\_module libexec/apache2/mod\_proxy\_html.so

LoadModule proxy\_module libexec/apache2/mod\_proxy.so

LoadModule proxy\_http\_module libexec/apache2/mod\_proxy\_http.so

LoadModule proxy\_ajp\_module libexec/apache2/mod\_proxy\_ajp.so

1. Restart Apache
2. Make sure you can access *SimpleSAMLphp* by going to [http://simplesaml-test.local](http://simplesaml-test.local/) and using the credentials (admin / pwd-specified-in-the-config-steps)

# Configuring Keycloak as SP and SimpleSAMLphp as IDP

## Configuring Keycloak as SP

1. Inside the Keycloak admin console, go to: Configure → Identity Providers → Add Provider → SAML v2.0
2. Use the following properties when creating the new provider:

| **Property** | **Value** |
| --- | --- |
| Alias | simple-saml |
| Display Name | Simple SAML |
| Enabled | ON |
| First Login Flow | first broker login |
| Single Sign-On Service URL | <http://simplesaml-test.local/simplesaml/saml2/idp/SSOService.php> |
| Single Logout Service URL | <http://simplesaml-test.local/simplesaml/saml2/idp/SingleLogoutService.php> |
| NameID Policy Format | Email |

1. Go to Mappers and create the following three mappers:

| **Field Name** | **Value** |
| --- | --- |
| Name | First Name Mapper |
| Mapper Type | Attribute Importer |
| Attribute Name | firstNN |
| Friendly Name | First Name |
| User Attribute Name | firstName |
| **Field Name** | **Value** |
| Name | Last Name Mapper |
| Mapper Type | Attribute Importer |
| Attribute Name | lastNN |
| Friendly Name | Last Name |
| User Attribute Name | lastName |

| **Field Name** | **Value** |
| --- | --- |
| Name | Email Mapper |
| Mapper Type | Attribute Importer |
| Attribute Name | email |
| Friendly Name | Email |
| User Attribute Name | email |

## Configuring SimpleSAMLphp as IDP

### Configuring the auth source

We need some users for testing:

1. Inside the SimpleSAMLphp installation dir, open the following file: config/authsources.php
2. Uncomment the section of the code that contains: "example-userpass", and add some users inside it using the following format:

'isai:isaipass' => array(

     'uid' => array('isai'),

     'email' => array('isai.ulate@app.activiti.com'),

     'firstNN' => array('Isai from SimpleSAMLphp'),

     'lastNN' => array('Ulate from SimpleSAMLphp'),

 ),

|  |
| --- |
|  |

1. The **email**, **firstNN**, **lastNN** are optional and were set to test the "mapping attributes" functinality of Keycloak. These attributes simulate attributes that can be specified in a real SAML Identity provider.  
     
   **Important Note:**  
   If Keycloak cannot map **all** the attributes it is expecting: **firstName**, **lastName**, **email**from an user ... then it will present an "After First Login" screen to that user (only the first time that users signs in), in order to collect those attributes. After those attributes are collected, Keycloak creates the user account and redirects the user to Activiti.   
   This happens when the mapping configuration is wrong or the information of those attributes are not set for an user.  
     
   If Keycloak gets to map all attributes for an user, then the first login of that user will be completely transparent (no "After First Login" screen will be shown to that user; Keycloak will automatically create the user account and the user will be redirected directly to Activiti).  
     
   As an example, if the following "users" are set inside the auth source configuration of SimpleSAMLphp then:

'example-userpass' => array(

    'exampleauth:UserPass',

    'student:studentpass' => array(

        'uid' => array('student'),

        'email' => array('student@app.activiti.com'),

        'firstNN' => array('Student First Name'),

        'lastNN' => array('Student Last Name'),

    ),

    'employee:employeepass' => array(

        'uid' => array('employee'),

        'email' => array('employee@app.activiti.com'),

    ),

    'isai:isaipass' => array(

        'uid' => array('isai'),

        'email' => array('isai.ulate@app.activiti.com'),

        'firstNN' => array('Isai from SimpleSAMLphp'),

        'lastNN' => array('Ulate from SimpleSAMLphp'),

    ),

),

|  |
| --- |
|  |

The users isai and student will not be presented with the "After First Login" screen.  
The user employee will be presented with the "After First Login" screen.

### Configuring the Remote SP

Add the following content to the metadata/saml20-sp-remote.php file:

$metadata['<http://localhost:8180/auth/realms/alfresco'>] = array(

    'AssertionConsumerService' => '<http://localhost:8180/auth/realms/alfresco/broker/saml/endpoint'>,

    'NameIDFormat' => 'urn:oasis:names:tc:SAML:2.0:nameid-format:emailAddress',

    'simplesaml.nameidattribute' => 'uid',

    'simplesaml.attributes' => TRUE,

);

|  |
| --- |
|  |

### Configuring the Hosted IDP

1. Generate the server.pem and server.crt files inside the **cert** directory if the SimpleSAMLphp installation.
2. Make sure to uncomment and set the following properties to the metadata/saml20-idp-hosted.php file:

'host' => '\_\_DEFAULT\_\_',

.

.

'privatekey' => 'server.pem',

'certificate' => 'server.crt',

.

.

'auth' => 'example-userpass',

|  |
| --- |
|  |

## Configuring Keycloak to automatically redirect to SimpleSAMLphp

Keycloak does not automaticaly redirect to the Identity provider, even if it is the only one configured and enabled.  
What we want is to be presented with the SimpleSAMLphp authentication interface once we try to access .. to accomplish this:

1. Go to the Keycloak admin console.
2. Inside the "realm" previously configured, go to Authentication.
3. In the table, go to the "Identity Provider Redirector" row and click on Actions → Config
4. In the "Default Identity Provider" field, set the alias of the Identity Provider for SimpleSAMLphp.
5. The "Alias" field is just a name and it can have any text value. We will call it: Simple SAML Auth
6. Save the changes

Check again, when trying to access Activiti, the user should be presented with the SimpleSAMLphp authentication interface.

# Virtual Hosts Config

## Apache Virtual Hosts

1. Create the following files inside the /etc/apache2/vhosts dir:  
     
   gds-chocolate-test.local.conf

<VirtualHost \*:80>

    ServerAdmin root@localhost

    ServerName gds-chocolate-test.local

    ServerAlias gds-chocolate-test1.com

    ServerAlias gds-chocolate-test2.com

    DefaultType text/html

    ProxyRequests off

    ProxyPreserveHost On

    ProxyPass / http://localhost:8080/

    ProxyPassReverse / http://localhost:8080/

</VirtualHost>

|  |
| --- |
|  |

keycloak-test.local.conf

<VirtualHost \*:80>

    ServerAdmin root@localhost

    ServerName keycloak-test.local

    ServerAlias keycloak-test1.com

    ServerAlias keycloak-test2.com

    DefaultType text/html

    ProxyRequests off

    ProxyPreserveHost On

    ProxyPass / http://localhost:8180/

    ProxyPassReverse / http://localhost:8180/

</VirtualHost>

|  |
| --- |
|  |

simplesaml-test.local.conf

|  |
| --- |
| <VirtualHost \*:80>          DocumentRoot "/Users/isaiulate/simplesamlphp/www"          ServerName simplesaml-test.local          ErrorLog "/private/var/log/apache2/simplesaml-test.local-error\_log"          CustomLog "/private/var/log/apache2/simplesaml-test.local-access\_log" common            SetEnv SIMPLESAMLPHP\_CONFIG\_DIR /Users/isaiulate/simplesamlphp/config            Alias /simplesaml /Users/isaiulate/simplesamlphp/www            <Directory "/Users/isaiulate/simplesamlphp/www">              <IfModule !mod\_authz\_core.c>              # For Apache 2.2:              Order allow,deny              Allow from all              </IfModule>              <IfModule mod\_authz\_core.c>              # For Apache 2.4:              Require all granted              </IfModule>          </Directory>  </VirtualHost> |

## "/etc/hosts" Config

|  |
| --- |
| 127.0.0.1       gds-chocolate-test.local keycloak-test.local simplesaml-test.local |

# ADF with Keycloak

1. In Keycloak, make sure the "Implicit Flow Enabled" setting is enabled.
2. In the ADF app, add:

"authType": "OAUTH",

"oauth2":{

  "host": "<http://localhost:8180/auth/realms/alfresco>",

  "clientId": "alfresco",

  "scope": "openid",

  "secret": "",

  "implicitFlow": true,

  "silentLogin": true,

  "redirectUri": "/",

  "redirectSilentIframeUri": "/#",

  "redirectUriLogout": "/logout"

},

|  |
| --- |
|  |

**Note:**The clientId depends on the id you set in keycloak. This, for the realm specified in the host as well.