Attribute Conversion for simpleSAMLphp

This page describes the features of Attribute Conversion and Filtering library for simpleSAMLphp

Introduction

eduGAIN uses Bridging Elements for interconnecting federations. To provide attribute translation and filtering services, an attribute 'mangling' library was developed for the Java-based bridging elements. As simpleSAMLphp can also be used as an eduGAIN bridging element, the conversion and filtering library was ported to PHP.

Beyond eduGAIN, you can use this module for every IdP or SP operating mode (shib13 SP/IdP, saml2 SP/IdP) of simpleSAMLphp in order to provide more powerful attribute conversion and filtering capabilities.

Download and support

You can download the module from here. The module is in beta stage, it needs broader community review. It is not yet recommended for production environments.

If you have any questions regarding the module, please write to '*aai aT niif _dOt hu*.

For changelogs please visit the project repository.

Compatibility

eduGAIN

This library is intended to be configuration-compatible with the eduGAIN Attribute_Conversion_for_eduGAIN Java library. The module can read the eduGAIN converter and filter engine XML configuration files and should operate the same way as the Java one.

Configuration files

The eduGAIN attribute converter and filter module defines its own XML schema for attribute conversion and attribute filtering purposes. See the Attribute_Conversion_for_eduGAIN page for more information on attribute rules.

Using the module

This module has a working name `edugain`. As this module only addresses the attribute translation part of the 'eduGAIN-problem', it might be renamed later.

Enabling the simpleSAMLphp module

This module depends on the **xsl** php extensions (more specifically, the *XSLTProcessor* class), so make sure it is properly configured.

The module can be enabled by creating an empty file named `modules/edugain/default-enable`.

simpleSAMLphp module configuration

EduGAIN is available for simpleSAMLphp as an authentication processing filter: *edugain:Attributes*. The Attributes processing filter takes the following configuration properties:

```
'authproc' => array(
  50 => array(
   'class' => 'edugain:Attributes',
   'mode' => 'idp',
   'converterconfig' => '/path/to/AttributeConverter.xml',
   'filterconfig' => '/path/to/AttributeFilter.xml',
   'cache' => true
  )
)
```

Configuration parameters for the module

- **class** (required): defines the eduGAIN filter for simpleSAMLphp.
- **mode** (required): configures the way this module operates (`idp` or `sp`). See below for more information on operating modes
- **converterconfig** (optional): configures the path of the attribute converter configuration xml file.
- **filterconfig** (optional): configures the path of the attribute filter configuration xml file.
- **cache** (optional, default: true): enables or disables the internal configuration cache. See the Configuration_cache section below for more.

> **Info**
>
> If either `converterconfig` or `filterconfig` is omitted, then the relevant part of the module (conversion or filtering respectively) is disabled. Note that **disabling filter means you let all the attributes through**.

Operating modes

EduGAIN module can operate in two modes, **idp** or **sp**. This mode affects two behaviors: the conversion-filtering order, and the provider matching.

- in **idp** mode, attribute filter is run **after** conversion, and the RemoteProvider match is done against the SP (or R-BE in eduGAIN bridged environment) which initiated the SSO session .
- in **sp** mode, attribute filter is run **before** conversion, and the RemoteProvider match is done against the IdP (or H-BE in eduGAIN bridged environment) which released the attributes to our simpleSAMLphp SP.

Configuration file

The simpleSAMLphp eduGAIN module reads the eduGAIN XML configuration format and transforms it into php arrays using XSL transformation. The submodules (*edugain:SplitMerge* and *edugain:Filter*) are configured automatically by the edugain:Attributes class.

PHP configuration interface for these filters are not supported at the moment and may be subject to change, so please use the XML configuration.

Configuration cache

The XML reading is very time-consuming but conversion and filtering rules should be evaluated on every request. Because of that, the eduGAIN module can cache the XML configuration into a serialized PHP array, which is stored locally in a directory named `cache`. If the XML file is not updated since the last cache file generation then the cache is used and the XML parsing part is skipped. Cache file name is computed according to the following:

```
md5(full_configuration_file_path).cache.php
```

> **Info**
>
> Enabling the cache is strongly recommended in production environments.

Differences between the Java and the PHP implementations

- There is no **CustomRule** for attribute conversion. One can use simpleSAMLphp authentication processing filter API to implement arbitrary conversion rules.
- **LocalProvider** matching is unsupported in simpleSAMLphp. Unfortunately when simpleSAMLphp is in bridging mode (using the SP module to protect an IdP), the IdP processing filters do not see the peer entity of the SP module. However, you can achieve the correct behavior by putting one *edugain:Attributes* processing filter in the SP configuration and use **RemoteProvider** matches to filter and convert attributes there.
- You don't need to use a separate attribute name mapper, because simpleSAMLphp contains built-in **name2oid**,**oid2name**, **name2urn** and **urn2name** methods, which provide the same functionality.
- Regular expressions are somewhat different in PHP. The eduGAIN module uses perl-compatible regular expressions (see preg_match documentation for details).

Változat #3
cziernorbert hozta létre 2026-04-14 13:22:23 CEST
cziernorbert frissítette 2026-04-17 13:04:28 CEST