Kiegészít? tudnivalók

- Integrációk
  - EntraID in eduID.hu
  - AAI AzureADasAuthsource
  - MediaWikiShibboleth
  - ElsevierSP
  - GoogleAuth
- Eszközök
  - Eszközök
  - SamlSign
  - Xmltooling Log4cpp patch
- Intézmény átnevezés
- Erasmusplus ESI
- Tanúsítványok a föderációban
- Interoperabilitás mátrix
- AA Testing
- EduIDTest
- Publikációk
- SSP EntityCategories
- NIIFSchema
- Szövetségi alapon történő felhasználó azonosítás Huntéka könyvtári alkalmazásokban
- IdP whichone
- VO

Integrációk

Integrációk

EntraID in eduID.hu

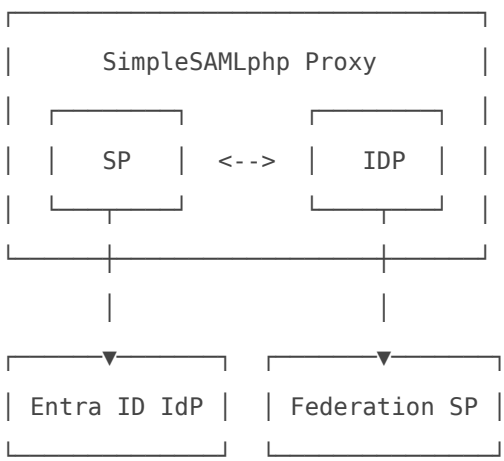
Overview

This document explains how to configure SimpleSAMLphp so that it uses Microsoft Entra ID as the authentication authority (Identity Provider) and then acts as a SAML Identity Provider (IdP) to external federated Service Providers (SPs). This pattern is commonly known as an **authentication proxy** or **IdP proxy**.

In plain terms:

- End users authenticate against **Entra ID**.
- SimpleSAMLphp receives this authentication and optionally enriches or transforms attributes.
- SimpleSAMLphp then issues SAML assertions to federation partners or internal applications.

The proxy setup looks like this:



Configure a SimpleSAMLphp SAML 2.0 Service Provider

To configure SimpleSAMLphp as a SAML 2.0 Service Provider, a new authentication source must be defined in the file `config/authsources.php`. This authentication source represents SimpleSAMLphp in its SP role towards Entra ID and is used to publish SP metadata.

The following example shows a minimal configuration suitable for use with Microsoft Entra ID:

```
$config = [
    /* ... */
    /* An authentication source that can authenticate against SAML 2.0 IdPs. */
    'entraid-sp' => [
        'saml:SP',
        // The entity ID of this SP.
        'entityID' => 'https://proxy.example.org/simplesaml',
        // The entity ID of the IdP this SP should contact.
        'idp' => 'https://sts.windows.net/<your-entra-tenant-id>/'
        'name' => ['en' => 'Microsoft Entra ID'],
        // certificates
        'certificate' => 'server.crt',
        'privatekey' => 'server.key',
        'privatekey_pass' => 'YourPrivateKeyPassphrase', /* you encrypt your private key,
right? */
        'authproc' => [
            /* authproc rules*/
        ],
        // fine tuning the auth source for Entra ID
        'sign.authnrequest' => true,
        'sign.logout' => true,
        'proxymode.passAuthnContextClassRef' => true,
        'disable_scoping' => true,
        'signature.algorithm' => 'http://www.w3.org/2001/04/xmldsig-more#rsa-sha256',
    ],
],
```

Configure SimpleSAMLphp SAML 2.0 Identity Provider

In order for SimpleSAMLphp to issue SAML assertions to downstream Service Providers, it must be configured as a SAML 2.0 Identity Provider. This configuration is defined in the file `metadata/saml20-idp-hosted.php`.

The IdP configuration references the previously defined authentication source, effectively chaining authentication to Entra ID.

```
$metadata['http://proxy.example.org/idp'] = [
   /*
    * The hostname of the server (VHOST) that will use this SAML entity.
    *
    * Can be '__DEFAULT__', to use this entry by default.
    */
   'host' => '__DEFAULT__',
   // X.509 key and certificate. Relative to the cert directory.
   'privatekey' => 'server.pem',
   'privatekey_pass' => 'YourPrivateKeyPassphrase',
   'certificate' => 'server.crt',
   /*
    * Authentication source to use. Must be one that is configured in
    * 'config/authsources.php'.
    */
   'auth' => 'entraid-sp', // proxy to Microsoft Entra ID
   'attributes.NameFormat' => 'urn:oasis:names:tc:SAML:2.0:attrname-format:uri',
   'authproc' => [
   ],
];
```

Create a new Enterprise Application in Entra ID

1. Create a new Enterprise Application

A new **Enterprise Application** must be created in the Entra ID portal to represent SimpleSAMLphp in its role as a SAML Service Provider. This can be done by navigating to the **Enterprise applications** section of the Entra ID portal and creating a new application. During creation, the option to create a custom application that is not found in the gallery should be selected. A descriptive name and also select *Integrate any other application you don't find in the gallery*.

2. Configure SAML-based Single Sign-On

After the application has been created, SAML-based single sign-on must be enabled. This is done by opening the application, navigating to the **Single sign-on** section, and selecting **SAML** as the sign-on method. The trust relationship between Entra ID and SimpleSAMLphp is established by

uploading the SAML 2.0 SP metadata generated by SimpleSAMLphp. The metadata upload automatically populates the basic SAML configuration, including the entity ID and assertion consumer service URL.

3. Download Entra ID Federation Metadata

To finalise the SimpleSAMLphp side of the bilateral trust relationship between your Entra ID tenant and SimpleSAMLphp, copy your Enterprise Application's *App Federation Metadata*. Using SimpleSAMLphp's Metadata Converter (found on the *Federation* tab of SimpleSAMLphp's admin portal), convert your App Federation Metadata to SimpleSAMLphp's native PHP format. Once you have the converted metadata, paste it into the `metadata/saml20-idp-remote.php` file.

4. Configure Attribute Claims Rules

Attribute and claim mappings can be adjusted in the Entra ID application to ensure that the required user attributes are released to SimpleSAMLphp. These attributes will later be available for transformation, filtering, or enrichment before being sent to downstream Service Providers.

Attribute Mapping and Transformation

When authenticating against Microsoft Entra ID, user attributes are returned as SAML claims using Microsoft-specific or WS-Federation-style claim URIs. In most federation environments, these claims must be mapped to standard SAML or eduPerson attribute names before they are released to downstream Service Providers.

SimpleSAMLphp performs attribute mapping through authentication processing filters. Mapping rules are applied in the `authproc` section of the authentication source that represents Entra ID, ensuring that attributes are normalized as soon as they enter SimpleSAMLphp. These mappings can either reuse
[https://github.com/simplesamlphp/simplesamlphp/blob/master/attributemap/entra2name.php built-in attribute maps provided] by SimpleSAMLphp or be defined explicitly using custom rules.

Here is an example of using `core:AttributeMap` processing filter:

```
'authproc' => [
    /* ... */
    60 => [
        'class' => 'core:AttributeMap',
        /* there are several versions of the userprincipalname claim, you only need the one
you use */
        'http://schemas.xmlsoap.org/claims/UPN' => 'eduPersonPrincipalName',
        'http://schemas.xmlsoap.org/ws/2005/05/identity/claims/upn' =>
```

```
'eduPersonPrincipalName',
        'http://schemas.xmlsoap.org/ws/2005/05/identity/claims/name' =>
'eduPersonPrincipalName',
        /* other possible attributes */
        'http://schemas.xmlsoap.org/claims/CommonName' => 'displayName',
        'http://schemas.xmlsoap.org/ws/2005/05/identity/claims/givenname' => 'givenName',
        'http://schemas.xmlsoap.org/ws/2005/05/identity/claims/surname' => 'sn',
        'http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress' => 'mail',
        'http://schemas.microsoft.com/ws/2008/06/identity/claims/groups' => 'memberOf',
    ],
    /* ... */
],
```

or

```
'authproc' => [
    60 => [
'class' => 'core:AttributeMap',
'attributemap' => 'entra2name',
    ],
],
```

Once mapped, attributes can be further filtered, enriched, or selectively released by additional authentication processing filters before being issued by the proxy IdP.

Configure SimpleSAMLphp to Use Entra ID as an Authentication Source

With the Enterprise Application configured, SimpleSAMLphp must be instructed to use Entra ID as its authentication source. This is done by setting the IdP entity ID in the entraid-sp authentication source to the Entra ID tenant identifier.

```
'idp' => 'https://sts.windows.net/<your-entra-tenant-id>/',
```

This configuration causes SimpleSAMLphp, acting as a Service Provider, to redirect authentication requests to Entra ID. After importing the Entra ID metadata, the corresponding entity ID should be visible under SAML 2.0 IdP metadata on the Federation tab of the SimpleSAMLphp admin interface.

Testing

You should now be able to go to the **Test** tab in the admin portal, log in to your `entraid-sp` authentication source, and be redirected to your Entra ID application's login page. Once logged in, it is worth verifying that SimpleSAMLphp is correctly receiving the attributes from Entra ID.

Sources

- https://nathansenblog.wordpress.com/2021/02/23/azure-ad-single-sign-on-with-simplesamlphp
- https://safire.ac.za/technical/resources/configuring-simplesamlphp-to-use-entra-id

AAI AzureADasAuthsource

Amennyiben Azure AD-ban tároljuk a felhasználói adatokat, úgy lehetőség van azt azonosítási forrásként is használni. A SimpleSAMLphp oldalon leírt telepítés után az alábbiak elvégzésére van szükség:

(ez csak egy példakonfiguráció, természetesen el lehet ettől térni)

Teend?k SimpleSAMLPHP (SSP) oldalon

Keressük ki az Azure AD-ból a Tenant ID-t. A beállítás során erre *TenantID*-val fogunk hivatkozni, oda minden esetben ezt az azonosítót kell behelyettesíteni. Az azonosítót jelenleg a https://portal.azure.com/#blade/Microsoft_AAD_IAM/ActiveDirectoryMenuBlade/Overview oldalon keresztül lehet beszerezni (Forrás: https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-how-to-find-tenant)

A *DOMAIN* helyére a használni kívánt scope-ot szükséges behelyettesíteni (pl intezmeny.hu)

config/authsources.php fájlba

```
    'default-sp' => [
        'saml:SP',

        // The entity ID of this SP.
        // Can be NULL/unset, in which case an entity ID is generated based on the metadata
URL.
        'entityID' => null,

        // The entity ID of the IdP this SP should contact.
        // Can be NULL/unset, in which case the user will be shown a list of available IdPs.
        'idp' => 'https://sts.windows.net/_TenantID_/',

        // The URL to the discovery service.
```

```
        // Can be NULL/unset, in which case a builtin discovery service will be used.
        'discoURL' => null,
        'NameIDFormat' => 'urn:oasis:names:tc:SAML:2.0:nameid-format:persistent',
        'simplesaml.nameidattribute' => 'eduPersonTargetedID',

        /*
         * The attributes parameter must contain an array of desired attributes by the SP.
         * The attributes can be expressed as an array of names or as an associative array
         * in the form of 'friendlyName' => 'name'. This feature requires 'name' to be set.
         * The metadata will then be created as follows:
         * <md:RequestedAttribute FriendlyName="friendlyName" Name="name" />
         */
        /*
        'name' => [
            'en' => 'A service',
            'no' => 'En tjeneste',
        ],

        'attributes' => [
            'attrname' => 'urn:oid:x.x.x.x',
        ],
        'attributes.required' => [
            'urn:oid:x.x.x.x',
        ],
        */
    ],
```

config/config-metarefresh.php fájlba

```
$config = [

    'sets' => [
        'azure' => [
            'cron' => ['hourly'],
            'sources' => [
                [
                    'src' =>
'https://login.microsoftonline.com/_TenantID_/federationmetadata/2007-
06/federationmetadata.xml',
```

```
                ],
            ],
            'expireAfter' => 34560060, // Maximum 4 days cache time (3600*24*4)
            'outputDir' => 'metadata/metarefresh-azure',
            'outputFormat' => 'flatfile',
        ],
    ],
];
```

metadata/saml20-idp-hosted.php

A

```
    'authproc' => [

        10 => [
              'class' => 'core:AttributeMap',
              'azure2name'
        ],

        15 => [
            'class' => 'core:AttributeCopy',
            'eduPersonPrincipalName' => 'schacPersonalUniqueCode',
        ],

        16 => ['class' => 'core:PHP',            'code' => '                $attributes[=
"urn:schac:personalUniqueCode:int:esi:_DOMAIN_:" .
$attributes["schacPersonalUniqueCode"]("schacPersonalUniqueCode"][0])[0];
            ',
        ],

        18 => [
            'class' => 'core:AttributeAlter',
            'subject' => 'eduPersonPrincipalName',
            'pattern' => '/^.*$/',
            'replacement' => '${0}@_DOMAIN_',
            'target' => 'eduPersonPrincipalName'
        ],
```

```
        20 => [
            'class' => 'core:AttributeAdd',
            'eduPersonEntitlement' => array('urn:mace:dir:entitlement:common-lib-terms')
        ],

        22 => [
            'class' => 'core:AttributeAdd',
            'schacHomeOrganization' => array('_DOMAIN_')
        ],

        23 => [
            'class' => 'core:AttributeAdd',
            'schacHomeOrganizationType' =>
array('urn:schac:homeOrganizationType:eu:higherEducationalInstitution')
        ],

// Azure AD-ban célszerű az affilation-t (intézményhez való viszonyt, pl hallgató, oktató,
dolgozó) security group alapján meghatározni. Ezeknek az objektum azonosítóját át fogja adni
az Azure AD, amit könnyen kicserélhetünk a kívánt affilation-re:

        31 => [
            'class' => 'core:AttributeAlter',
            'subject' => 'eduPersonAffiliation',
            'pattern' => '/^_eduID_Dolgozó_GroupID_$/', // _eduID_Dolgozó_GroupID_ értéket
cseréljük a megfelelő Objektum ID-ra
            'replacement' => 'faculty',
        ],

        32 => [
            'class' => 'core:AttributeAlter',
            'subject' => 'eduPersonAffiliation',
            'pattern' => '/^_eduID_Hallgató_GroupID_$/', // _eduID_Hallgató_GroupID_ értéket
cseréljük a megfelelő Objektum ID-ra
            'replacement' => 'student',
        ],

        33 => [
            'class' => 'core:AttributeAlter',
            'subject' => 'eduPersonAffiliation',
            'pattern' => '/^_eduID_Admin_GroupID_$/', // _eduID_Admin_GroupID_ értéket
```

```
cseréljük a megfelelő Objektum ID-ra
            'replacement' => 'staff',
        ],

        34 => [
            'class' => 'core:AttributeAdd',
            'eduPersonAffiliation' => array('member'),
        ],

        35 => [
            'class' => 'core:AttributeCopy',
            'eduPersonAffiliation' => 'eduPersonScopedAffiliation'
        ],

        36 => [
            'class' => 'core:AttributeAlter',
            'subject' => 'eduPersonScopedAffiliation',
            'pattern' => '/^.*$/',
            'replacement' => '${0}@$_DOMAIN_',
        ],

        50 => [
            'class' => 'core:TargetedID',
            'identifyingAttribute' => 'eduPersonPrincipalName',
            'nameId' => TRUE,
        ],

        60 => [
            'class' => 'core:AttributeMap',
            'name2oid'
        ],
        75  => [
            'class' => 'entitycategories:EntityCategory',
            'default' => true,
            'strict' => false,
            'allowRequestedAttributes' => true,
            'http://eduid.hu/category/registered-by-eduidhu' => [],
            'http://www.geant.net/uri/dataprotection-code-of-conduct/v1' => [
                'urn:oid:2.16.840.1.113730.3.1.241', # displayName
                'urn:oid:2.5.4.4', # sn
```

```
                'urn:oid:2.5.4.42', # givenName
                'urn:oid:0.9.2342.19200300.100.1.3', # mail
                'urn:oid:1.3.6.1.4.1.5923.1.1.1.6', # eduPersonPrincipalName
                'urn:oid:1.3.6.1.4.1.5923.1.1.1.9', # eduPersonScopedAffiliation
                'urn:oid:1.3.6.1.4.1.5923.1.1.1.1', # eduPersonAffiliation
            ],
            'http://refeds.org/category/research-and-scholarship' => [
                'urn:oid:2.16.840.1.113730.3.1.241', # displayName
                'urn:oid:2.5.4.4', # sn
                'urn:oid:2.5.4.42', # givenName
                'urn:oid:0.9.2342.19200300.100.1.3', # mail
                'urn:oid:1.3.6.1.4.1.5923.1.1.1.6', # eduPersonPrincipalName
                'urn:oid:1.3.6.1.4.1.5923.1.1.1.9', # eduPersonScopedAffiliation
                'urn:oid:1.3.6.1.4.1.5923.1.1.1.1', # eduPersonAffiliation
            ],
        ],

        90 => 'core:AttributeLimit',
    ],

    'simplesaml.nameidattribute' => 'urn:oid:1.3.6.1.4.1.5923.1.1.1.6',

    'attributeencodings' => array(
        'urn:oid:1.3.6.1.4.1.5923.1.1.1.10' => 'raw', /* eduPersonTargetedID with oid
NameFormat. */
    ),

    'sign.logout' => true
];
```

attributemap/azure2oid.php

```
<?php
$attributemap = [
    // displayName
    'http://schemas.microsoft.com/identity/claims/displayname' =>
'urn:oid:2.16.840.1.113730.3.1.241',
    // eppn
```

```
    'http://schemas.xmlsoap.org/ws/2005/05/identity/claims/name' =>
'urn:oid:1.3.6.1.4.1.5923.1.1.1.6',
    // givenName
    'http://schemas.xmlsoap.org/ws/2005/05/identity/claims/givenname' => 'urn:oid:2.5.4.42',
    // cn
    ':://schemas.xmlsoap.org/ws/2005/05/identity/claims/surname' => 'urn:oid:2.5.4.3',
    // surname
    'http://schemas.xmlsoap.org/ws/2005/05/identity/claims/surname' => 'urn:oid:2.5.4.4',
    // mail
    'http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress' =>
'urn:oid:0.9.2342.19200300.100.1.3',
    // o & organisation
    'http://schemas.microsoft.com/identity/claims/tenantid' => 'urn:oid:2.5.4.10',
];
```

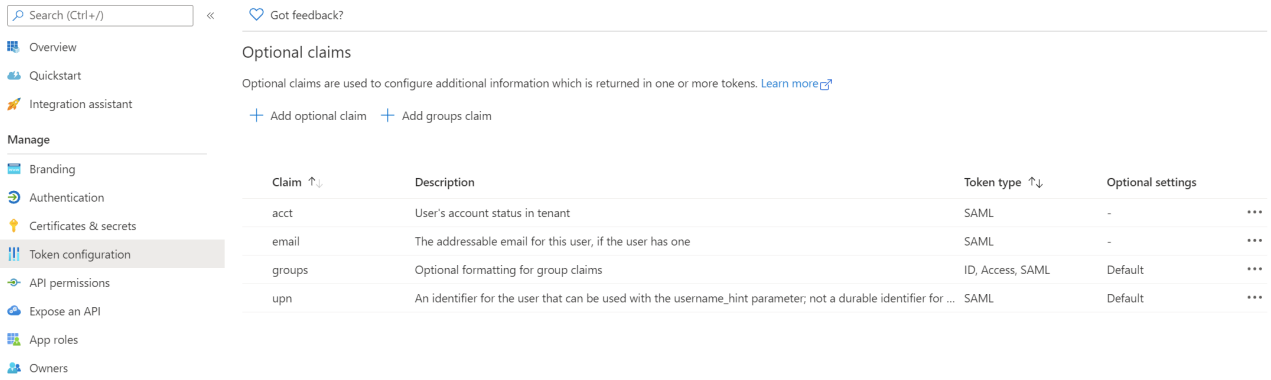
attributemap/azure2name.php

```
<?php
$attributemap = [
    // eppn
    'http://schemas.microsoft.com/identity/claims/objectidentifier' =>
'eduPersonPrincipalName',
    // mail
    'http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress' => 'mail',
    // displayName
    'http://schemas.microsoft.com/identity/claims/displayname' => 'displayName',
    // givenName
    'http://schemas.xmlsoap.org/ws/2005/05/identity/claims/givenname' => 'givenName',
    // cn
    'http://schemas.xmlsoap.org/ws/2005/05/identity/claims/surname' => 'sn',
    // affiliation
    'http://schemas.microsoft.com/ws/2008/06/identity/claims/groups' =>
'eduPersonAffiliation',
];
```

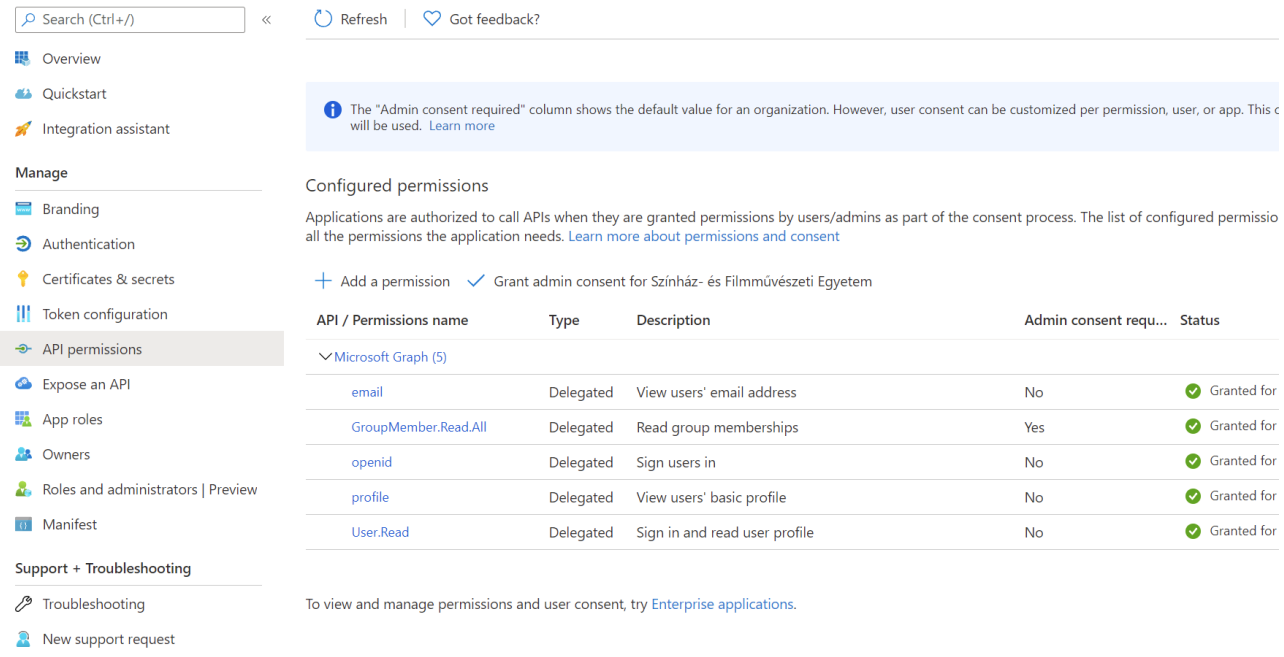
Teend?k Azure AD oldalon

1. A https://portal.azure.com/ oldalon jelentkezzünk be egy adminisztrátori fiókkal

2. Válasszuk az "App registrations"-t
3. "New registration"
4. "Redirect URI (optional)" -nál adjuk meg a telepített SSP default SP címét. Pl: https://idp.*DOMAIN*/simplesaml/module.php/saml/sp/metadata.php/default-sp
5. "Token configuration" # > "Add optional claim"> "Token type"-nál válasszuk a "SAML"-t és jelöljük ki az összes attribútumot, majd, "Add"
6. "Add groups claim", majd mentsük el



7. Állítsuk be az API persmissions-t az alábbi alapján:



Teszt

MediaWikiShibboleth

- Shibboleth Authentication Extension: http://www.mediawiki.org/wiki/Extension:Shibboleth_Authentication
- Shibboleth Authentication Plus Extension: http://www.mediawiki.org/wiki/Extension:Shibboleth_Authentication_Plus

ElsevierSP

Speciális eduPersonTargetedID kiadásának beállítása

Shibboleth IdP alatt

```
vim [/path/to]/shibboleth-idp/conf/attribute-resolver.xml
```

Szúrjuk be az alábbi attribútumdefiníciót, ahol

- a `scope` értéke az intézményi scope (ugyanaz, amit pl. az eduPersonPrincipalName attribútum előállításakor is használ)
- a `sourceAttributeID` értéke a persistent nameID-t előállító attribútum id-je

```
    <!-- Buggy edupersonTargetedId required for Elsevier: -->
    <resolver:AttributeDefinition id="elsevierId" xsi:type="Scoped" scope="niif.hu"
sourceAttributeID="persistentId"
                                  xmlns="urn:mace:shibboleth:2.0:resolver:ad" >
      <resolver:Dependency ref="storedIdConnector" />
      <resolver:AttributeEncoder xsi:type="enc:SAML2ScopedString"
                                 name="urn:oid:1.3.6.1.4.1.5923.1.1.1.10" friendlyName="buggy-
eduPersonTargetedID"
                                 xmlns="urn:mace:shibboleth:2.0:attribute:encoder" />
    </resolver:AttributeDefinition>
```

```
vim [/path/to]/shibboleth-idp/conf/attribute-filter.xml
```

Fontos, hogy amennyiben az ajánlásnak megfelelően a Resource Registry által előállított attribute filtert használjuk, akkor ne ezt a dinamikusan frissülő fájlt szerkesszük, hanem az attribute-filter-local.xml-t, és ebben a fájlban végezzük el az alábbi módosításokat.

1. Szúrjuk be az alábbi részletet, amely megmondja, hogy az Elsevier SP számára mely attribútumok adandók ki:

```
    <AttributeFilterPolicy id="buggy-eptid">
            <PolicyRequirementRule xsi:type="basic:AttributeRequesterString"
value="https://sdauth.sciencedirect.com/" />
            <AttributeRule attributeID="elsevierId">
                <PermitValueRule xsi:type="basic:ANY" />
            </AttributeRule>
            <AttributeRule attributeID="eduPersonScopedAffiliation">
                <PermitValueRule xsi:type="basic:ANY" />
            </AttributeRule>
    </AttributeFilterPolicy>
```

2. A `releaseIDsToAnyone` alapértelmezett szabályt módosítsuk az alábbiakra:

```
    <!--  Release IDs to anyone -->
    <AttributeFilterPolicy id="releaseIDsToAnyone">
        <PolicyRequirementRule xsi:type="basic:NOT">
                <basic:Rule xsi:type="basic:AttributeRequesterString"
value="https://sdauth.sciencedirect.com/" />
        </PolicyRequirementRule>
        <AttributeRule attributeID="transientId">
            <PermitValueRule xsi:type="basic:ANY" />
        </AttributeRule>

        <AttributeRule attributeID="persistentId">
            <PermitValueRule xsi:type="basic:ANY" />
        </AttributeRule>

    </AttributeFilterPolicy>
```

Ez utóbbi módosításra azért van szükség, hogy a mindenki számára kiadható szabványos persistentID ne csapja felül ez Elsevier számára kiadandót.

SimpleSAMLphp alatt

```
vim [/path/to]/simplesaml/metadata/saml20-sp-remote.php
```

Beállítjuk, hogy csak az elsevier SP-je esetén az attribútum előállítás és kiadás folyamatát toldja meg még egy lépéssel. Ehhez szükséges, hogy a fenti fájlba vegyük fel állandó elemként az Elsevire SP metadatájának kiegészítéseként az alábbi PHP tömböt. Fontos, hogy a tömb egyes elemei előtt szereplő számok magasabbak legyenek, mint az IdP általános feloldási szabályainál megadott számok, de alacsonyabbak, mint a jellemzően a folyamat végén beállított 'name2oid'

mappelések.

```
$metadata['https://sdauth.sciencedirect.com/']['authproc'] = array (
                96 => array(
                         'class' => 'core:TargetedID'
                ),
                97 => array(
                        'class' => 'core:AttributeAlter',
                        'subject' => 'eduPersonTargetedID',
                        'pattern' => '/^.*$/',
                        'replacement' => '${0}@intezmenyiScope.hu',
                ),
);
```

A fenti kódrészlet annyit teszi, hogy újragenerálja az alapértelmezett eduPersonTargetedID-t egyszerű formában (csak a stringet, a NameID-s xml struktúra nélkül), majd mögé teszi az intézményi scope-ot. Fontos, hogy a megoldás feltételezi azt, hogy az Elsevier SP metadatájának további részei betöltésre kerülnek pl. a metarefresh modul által.

GoogleAuth

Alapok

A wikiben részletezett többi megoldáshoz képest ez a fejezet kilóg a sorból, hiszen ez a megoldás "a federáción" belül csak egy Identity Providert használ - a google központi kapuját. Ezt bárki használhatja a saját szolgáltatásában (SP) történő felhasználó-hitelesítésére, természetesen annyi megszorítás van, hogy az adott felhasználóknak rendelkezniük kell Google Accounttal (gmail-es e-mail címmel).

A google ekkor 2.0-ás szabványú OpenID protokoll szerint működő OpenID IdP-ként viselkedik, emellett támogatja az OpenID Attribute Exchange 1.0 alapú attribútum kiadást, így ha a felhasználó az autentikáció során hozzájárul a felsorolt adatainak az SP részére történő kiadásához, akkor azt a google ki is adja. (A kiadható attribútumok köre korántsem teljes, lévén jelenleg egyedül az e-mail cím kiadatására van lehetősége az SP-nek)

Beállítás

Az autentikációhoz használt google-végpont a

https://www.google.com/accounts/o8/id

címen érhető el, a megfelelő paramétereket ide kell átadni.

Rendelkezésre álló paraméterek

openid.ns	**Kötelező - a használandó OpenID protokollt definiálja, ajánlott érték: "http://specs.openid.net/auth/2.0"**
openid.claimed_id	Opcionális - a kérelem típusát azonosítja, ajánlott érték: "http://specs.openid.net/auth/2.0/identifier_select"
openid.identity	Opcionális, egy alternatív azonosítót definiál, ajánlott érték: "http://specs.openid.net/auth/2.0/identifier_select"
openid.return_to	Kötelező - azt az oldalt definiálja, amelyre a google visszairányít az autentikáció után. Támogatott http és https protokollú visszatérő oldal egyaránt.

openid.ns	Kötelező - a használandó OpenID protokollt definiálja, ajánlott érték: "http://specs.openid.net/auth/2.0"
openid.realm	Opcionális - azt a domaint definiálja, amelyre a felhasználót a google-el autentikáltatni szeretnénk. Alapértelmezés szerint az openid.return_to-nál megadott honlapra mutat. Amennyiben megadásra kerül, akkor a domainnek mindenképp egyeznie kell az openid.return_to domainnevével.
openid.assoc_handle	Opcionális - részletes ismetető: http://openid.net/specs/openid-authentication-2_0.html#associations
openid.mode	Kötelező - a kérés módját adja meg, a két lehetséges érték: "checkid_immediate" (szinkron kommunikáció) "checkid_setup" (aszinkron kommunkikáció)
openid.ns.ext1	Opcionális (attribútum kiadatáskor kötelező) - ajánlott érték: "http://openid.net/srv/ax/1.0"
openid.ext1.mode	Opcionális (attribútum kiadatáskor kötelező) - ajánlott érték: "fetch_request"
openid.ext1.type.email	Opcionális (attribútum kiadatáskor kötelező) - ajánlott érték: "http://axschema.org/contact/email"
openid.ext1.required	Opcionális (attribútum kiadatáskor kötelező) - ajánlott érték: "email" (Jelenleg csak az email attribútum elérhető)

Minta kérés

Az alábbi kérés igényli az email-cím attribútum kiadását is

```
https://www.google.com/accounts/o8/ud
?openid.ns=http%3A%2F%2Fspecs.openid.net%2Fauth%2F2.0
&openid.claimed_id=http%3A%2F%2Fspecs.openid.net%2Fauth%2F2.0%2Fidentifier_select
&openid.identity=http%3A%2F%2Fspecs.openid.net%2Fauth%2F2.0%2Fidentifier_select
&openid.return_to=http%3A%2F%2Fwww.example.com%2Fcheckauth
&openid.realm=http%3A%2F%2Fwww.example.com%2F
&openid.assoc_handle=ABSmpf6DNMw
&openid.mode=checkid_setup
&openid.ns.ext1=http%3A%2F%2Fopenid.net%2Fsrv%2Fax%2F1.0
&openid.ext1.mode=fetch_request
&openid.ext1.type.email=http%3A%2F%2Faxschema.org%2Fcontact%2Femail
&openid.ext1.required=email
```

Minta válasz sikeres autentikáció esetén

```
http://www.example.com:8080/checkauth
?openid.ns=http%3A%2F%2Fspecs.openid.net%2Fauth%2F2.0
&openid.mode=id_res&openid.op_endpoint=https%3A%2F%2Fwww.google.com%2Faccounts%2Fo8%2Fud
&openid.response_nonce=2008-09-18T04%3A14%3A41Zt6shNlcz-MBdaw
&openid.return_to=http%3A%2F%2Fwww.example.com%3A8080%2Fcheckauth
&openid.assoc_handle=ABSmpf6DNMw
&openid.signed=op_endpoint%2Cclaimed_id%2Cidentity%2Creturn_to%2Cresponse_nonce%2Cassoc_handle
%2Cext1.mode%2Cext1.type.email%2Cext1.value.email
&openid.sig=s%2FgfiWSVLBQcmkjvsKvbIShczH2NOisjzBLZOsfizkI%3D
&openid.identity=https%3A%2F
%2Fwww.google.com%2Faccounts%2Fo8%2Fid%3Fid%3DACyQatixLeLODscWvwqsCXWQ2sa3RRaBhaKTkcsvUElI6tNH
IQ1_egX_wt1x3fAY983DpW4UQV_U
&openid.claimed_id=https%3A%2F%2Fwww.google.com%2Faccounts%2Fo8%2Fid%3Fid%3DACyQatixLeLODscWvw
qsCXWQ2sa3RRaBhaKTkcsvUElI6tNHIQ1_egX_wt1x3fAY983DpW4UQV_U
&openid.ns.ext1=http%3A%2F%2Fopenid.net%2Fsrv%2Fax%2F1.0
&openid.ext1.mode=fetch_response
&openid.ext1.type.email=http%3A%2F%2Faxschema.org%2Fcontact%2Femail
&openid.ext1.value.email=fred.example%40gmail.com
```

Minta választ sikertelen autentikáció esetén

```
http://www.example.com/checkauth
?openid.mode=cancel
&openid.ns=http%3A%2F%2Fspecs.openid.net%2Fauth%2F2.0
```

Válaszok feldolgozása

A megfelelő kérés elküldése után a rendszer átirányít a google autentikációs oldalára, ahol a felhasználónak meg kell adnia google accounttal kapcsolatos adatait, majd sikeres bejelentkezés után jóvá kell hagynia, hogy a SP részére a google átadja az információt, hogy bejelentkezett ill. az egyéb attribútumokat, melyeket adott esetben az SP igényelt. Pozitív válasz esetén a google beállítva a megfelelő értékeket meghívja a visszatérő oldalt, ahol a megfelelő GET paraméterek feldolgozásával már SP-hatáskörben irányíthatók tovább a felhasználók.

Küls? hivatkozások

- http://code.google.com/intl/hu-HU/apis/accounts/docs/OpenID.html

Eszközök

Eszközök

- uApprove: User-consent (beleegyezési nyilatkozat) eszköz
- SamlSign: Metaadat aláíró és ellenőrző eszköz

Eszközök

SamlSign

Parancssoros eszköz, melyhez debian alatt az `opensaml2-tools` csomagot kell telepíteni. A program kétféle üzemmódban képes működni: **metaadat aláírása** és **metaadat ellenőrzése**.

Metaadat aláírása

```
samlsign -s -k /path/to/mainkey.key -f /path/to/metadatatosign.xml
```

Alapértelmezés szerint a samlsign az eredményeket az alapértelmezett kimenetre írja ki (STDOUT), így célszerű ezt egy új fájlba átirányítani:

```
samlsign -s -k /path/to/mainkey.key -f /path/to/metadatatosign.xml >
/path/to/metadatasigned.xml
```

Metaadat ellenőrzése

```
samlsign -c /path/to/maincert.crt -f /path/to/metadatatosign.xml
```

Samlsign legfontosabb kapcsolói

- -s ez határozza meg, hogy aláírunk, vagy ellenőrzünk. Ha megadtuk kapcsolóként, akkor a program megpróbálja aláírni a megadott xml fájlt, ha nem, akkor ugyanezt a fájlt ellenőrizni fogja.
- -f az ellenőrzendő/aláírandó fájl elérhetősége **abszolút útvonallal** megadva
- -k a privát kulcs elérhetősége **abszolút útvonallal** megadva
- -c az ellenőrzésre használt publikus kulcs elérhetősége **abszolút útvonallal** megadva
- További részletes leírás a samlsign man oldalán

További fontos tudnivalók A samlsign nem szereti a metadatában szereplő `Organization`-nel kapcsolatos adatokat, mivel ilyen tag-ekben kötelezően megadandó `xml:lang` attribútumot `lang`-ra alakítja át, ami által viszont nem lesz érvényes (valid) maga a metaadat, így pl. a shibboleth sem fog tudni vele mit kezdeni. A **megoldás** (nem szép, de hasznos): az aláírás előtt álló metaadatokból ki kell szedni az `Organization`-nel kapcsolatos adatokat. Ezek után már gond nélkül aláírja és az eredmény is érvényes lesz.

Apró trükk a privát kulcs kinyerésére `jks`-ből

Tekintettel arra, hogy a `keytool` nem teszi lehetővé a privát kulcs kihalászását JavaKeystore-ból, így külső segítséget kell igénybe vennünk. A segédalkalmazás `ExportPrivateKey` névre hallgat, és innen letölthető egy darab zip fájl. Használata rendkívül egyszerű:

```
java -jar ExportPrivateKey.zip {jks fájl elérhetősége} JKS {jks jelszó} {alias} {célfájl}
```

Ezek után a létrehozott kulccsal már használhatjuk is a samlsign-t.

Eszközök

Xmltooling Log4cpp patch

```
--- xmltooling/soap/impl/SOAPClient.cpp.orig    2008-03-14 23:50:29.000000000 +0100
+++ xmltooling/soap/impl/SOAPClient.cpp 2008-04-25 13:14:29.000000000 +0200
@@ -89,8 +89,11 @@
     XercesJanitor<DOMDocument> janitor(doc);

     Category& log = Category::getInstance(XMLTOOLING_LOGCAT".SOAPClient");
-    if (log.isDebugEnabled())
-        log.debugStream() << "received XML:\n" << *(doc->getDocumentElement()) <<
logging::eol;
+    if (log.isDebugEnabled()) {
+        string serializedXml;
+        XMLHelper::serialize (doc->getDocumentElement(),serializedXml,false);
+        log.debugStream() << "received XML:\n" << serializedXml << logging::eol;
+    }

     auto_ptr<XMLObject> xmlObject(XMLObjectBuilder::buildOneFromElement(doc-
>getDocumentElement(), true));
     janitor.release();
```

Intézmény átnevezés

Sok esetben merül fel az a kérdés, hogy eduID tagintézmény névváltozása esetén mi a teendő. Alább egy konkrét levelezés másolata következik, amely remélhetőleg általános információkat tartalmaz.

Kérdés

> Kecskemét és Szolnok egyesítésével létrejött a Pallasz Athéné Egyetem. Mit kell ilyenkor tennünk? Mert a weboldalakat át kéne neveznem az új dominre. Kell csinálnom új IDP-t és új SP-ket? vagy csak SP-ket?

Szerz?dés

Ha a PAE jogutódja a KeFo-nak, akkor nem kell új szerződést kötni. Egyébként igen.

Átnevezni vagy nem átnevezni?

entityID

Az entitások átnevezése problémás, ezért azt célszerű kerülni. A legtöbb gond az eduPersonTargetedId-vel van, ugyanis az "targeted" azonosító, ami azt jelenti, hogy akár az IdP, akár az SP entityID-je megváltozik, az azonosító értéke is megváltozik, azaz a felhasználó "élettörténete" újraindul az SP-nél. Ráadásul az azonosítók kézi átállítása is meglehetősen körülményes. Kérdés, hogy mely általatok használt SP-k használnak eduPersonTargetedId-t. A Videotorium például igen, ill. lásd még a href.xml-ben a RequestedAttributes elemet. Mivel az entityID nem jelenik meg a felhasználóknak (normális esetben), emiatt nem valószínű, hogy bárki arra kötelezne titeket, hogy változtassátok meg.

További probléma az entityID megváltoztatásával, hogy a kereskedelmi adatbázis-szolgáltatók jellemzően kézzel tartják karban azt a listát, hogy mely intézmények számára elérhető a szolgáltatás, és az entityID változtatása miatt az intézményünk kikerülhet ezekből a listákból.

Scope

Az eduPersonPrincipalName és az eduPersonScopedAffiliation használja a scope-okat. Az minden további nélkül működik, hogy újabb scope-okat **IS** elkezdjetek használni (új felhasználóknál), de ha egy létező felhasználó scope-ját lecserélitek, akkor ő a fentihez hasonló módon új felhasználóként jelenik meg az összes SP-nél. És eduPersonPrincipalname-et és/vagy eduPersonScopedAffiliationt nagyon sok SP használ, valamint a legtöbb belső SP is valószínűleg úgy van konfigurálva, hogy scope alapon végezze az autorizációt. Az autorizációs szabályok (pl. `require affiliation member@kefo.hu` vagy `require eppn bekre.pal@kefo.hu`) szabályok módosítása triviális. Az élettörténet megtartása sajnos már alkalmazásspecifikus, jellemzően kézi adatbázisműveleteket igényel. Annyival egyszerűbb ez, mint az eduPersonTargetedId módosítása, hogy pontosan tudod az átírási szabályt, pl:

```
s/([^@]+)@kefo.hu/$1@pae.hu/
```

A tanácsom az, hogy nézzétek át, hogy mely SP-k esetén fontos az élettörténet megtartása, és ezeket kérjétek meg a fenti változtatásra. (Pl. ha vannak olyan HBONE-adminisztrációs szolgáltatások, amiket eddig kefo.hu-s azonosítóval használtál, akkor azok ilyenek.)

mail

A mail cím megváltoztatása jellemzően nem jelent problémát. Azt érdemes észben tartani, hogy a Drupal megköveteli az e-mail címek egyediségét, azaz ha megváltozik az eppn, miközben nem változik a mail, akkor az "új" felhasználó létrehozása nem fog sikerülni.

IdP URL

Mellékhatások nélkül módosítható, hogy az IdP milyen URL-en azonosítsa a felhasználókat (vagy válaszoljon az AttributeQuery-kre, de ez nagyon ritka funkció). Ehhez a `SingleSignOnService` és `SingleLogoutService` végpontokat kell lecserélni a Resource Registry-ben a haladó beállítások között, valamint természetesen az IdP webszerverét megfelelően kell konfigurálni. Ez a módosítás kizárólag azt befolyásolja, hogy milyen URL jelenik meg a felhasználók böngészőjében, amikor azonosításra átirányítják őket az SP-k. Nyilván megfelelő SSL tanúsítvány is kell az új URL-re.

Discovery leírás

Ez is mellékhatások nélkül módosítható, és ez az, ami a leginkább szembetűnő a felhasználók számára. Ezt is a Resource Registry-ben, a Leírók-nál kell módosítani. (Légyszi küldjetek egy értesítést az info @ eduid.hu-ra, ha módosítjátok, mert a központi Discovery Service-t jelenleg még kézzel kell frissítenünk.)

Erasmusplus ESI

Erasmus+ és ESI

Az Európai Hallgatói Kártya Kezdeményezés részeként az Európai Bizottság 2021 júniusától kezdve digitális formában kívánja támogatni az Erasmus programhoz kapcsolódó összes szolgáltatást, mint például az Online Learning Agreement (OLA) és az Erasmus+ mobilalkalmazást. A digitalizálás alapvető része a hallgatók azonosítása, amely az eduGAIN-en keresztül fog történni.Az eduGAIN szövetség hazai képviselője a KIFÜ, aki Magyarországon az eduID.hu szövetséget működteti, hogy az oktatás, kutatás és közgyűjtemények szereplői az intézményi hitelesítő adataikkal bejelentkezhessenek az eduGAIN szolgáltatásokba. Az eduGAIN-ben a szolgáltatások sikeres azonosítás esetén azonnal megkapják a felhasználók helyes azonosításához és jogosultság kezeléséhez szükséges információkat. Az MyAcademicID működéséhez, az európai hallgatói azonosító (European Student Identifier) helyes létrehozásához és az Erasmus+ szolgáltatásokban szükséges attribútumok beállításhoz szükséges információk az alábbiak.

MyAcademicID proxy

Az Erasmus+ szolgáltatásokhoz való hozzáférés a MyAcademicID projektben létrehozott és a GEANT által kezelt proxyn keresztül történik. Ez azt jelenti, hogy az Erasmus+ összes szolgáltatásának egyetlen szolgáltatója van a következő entitásazonosítóval:

> “ https://proxy.prod.erasmus.eduteams.org/metadata/backend.xml

A Szolgáltatót az eduID.hu az eduGAIN-en keresztül teszi közzé, így ha az intézménye tagja az eduGAIN-nek is (lásd: Metaadatok), akkor nem kell további lépéseket tenni. Ellenkező esetben lépjen kapcsolatba először a KIFÜ eduID.hu csapatával, hogy kérje identitásszolgáltatójának exportálását az eduGAIN-ba, majd konfigurálja be az eduID.hu szolgáltatás által terjesztett eduGAIN metaadatokat (lásd: Metadata).

Attributumok

Az Erasmus + szolgáltatások eléréséhez szükséges attribútumok a következők.

Attributumok	Leírás	Példa

eduPersonPrincipalName	Állandó, nem célzott, nem újra kiosztható globálisan egyedi azonosító	`username@foobar.hu`
eduPersonTargetedID	Nem átlátszó, célzott (minden szolgáltatónál más) azonosító, amely nem osztható ki újra	`https://idp.foobar.hu/idp/shibboleth!https://sp.example.com/shibboleth!a1b2c3d4-789a-4ca7-8ff9-43216789bd`
displayName	A felhasználó megjelenítendő neve. **Szükséges, ha nincsen cn, vagy givenName+sn.**	
cn	A felhasználó teljes neve. **Szükséges, ha nincsen displayName vagy givenName+sn.**	`Gipsz Jakab Aladár`
givenName	Felhasználó keresztneve. **Szükséges, ha nincsen displayName vagy cn.**	`Jakab`
sn	Felhasználó családi neve. **Szükséges, ha nincsen displayName vagy cn.**	`Gipsz`
eduPersonAffiliation	Felhasználó és intézmény közti viszony leírása.	`[student](member,)`
eduPersonScopedAffiliation	Felhasználó és intézmény közti viszony leírása. scope-al	`[student@foobar.hu](member@foobar.hu,)`
schacPersonalUniqueCode	A szervezet által hozzárendelt személyes egyedi kód (URN). '''Az európai hallgatói azonosító kódolására és továbbítására szolgál.	`<nowiki>urn:schac:personalUniqueCode:int:esi:foobar.hu:123456789</nowiki>`
schacHomeOrganization	Szervezete domainje.	`foobar.hu`
schacHomeOrganizationType	Szervezet típusa (URN)	`<nowiki>urn:schac:homeOrganizationType:eu:higherEducationalInstitution</nowiki>`

European Student Identifier

Az European Student Identifier (ESI) globálisan egyedi, tartós, nem célzott azonosító (minden szolgáltatónál ugyanaz marad). Titkosítva kerül továbbításra a `schacPersonalUniqueCode` attribútumon keresztül. Az ESI két fő módját támogatott:

1. országos kód:ha rendelkezésre áll nemzeti hallgatói kód
2. intézményenként:jelenleg ez támogatható Magyarországi szinten

Az ESI három részből áll:

- változatlan URN névtér: `<nowiki>urn:schac:personalUniqueCode:int:esi:</nowiki>:`
- szervezet domainje: foobar.hu
- hallgatói azonosító kód (pl. belső nyilvántartási szám): 123456789

Az így kapott teljes ESI: `<nowiki>urn:schac:personalUniqueCode:int:esi:foo.bar:123456789</nowiki>`

Az ESI teljes specifikációja elérhető a GEANT wiki-n:

https://wiki.geant.org/display/SM/European+Student+Identifier

EWP adminisztrátori funkció

Az eduGAIN közösség 2022 őszén definiálta **EWP Admin szerepkört**. EWP Admin szerepkör (Erasmus Without Paper Administrator szerepkör) lehetővé teszi az Erasmus+ tevékenységeiben részt vevő felsőoktatási intézmények (HEI) felhatalmazott képviselői számára, hogy szabványos módon bejelentkezzenek az EWP-eszközökbe és EWP-információik és -beállításaik kezelése egységes legyen. Erről bővebb információ itt található:

https://wiki.geant.org/display/SM/EWP+Admin+Role

További információk

https://wiki.geant.org/display/SM/Identifier+in+Erasmus+mobility

Tanúsítványok a föderációban

SAML2 föderációkban az entitásoknak ismerniük kell egymás publikus kulcsát (amelyet X.509 tanúsítványokban osztanak meg egymással) ahhoz, hogy biztonságosan kommunikálhassanak egymással. Eközben a felhasználókkal is interakcióba lépnek, ami miatt könnyű összekeverni a két fajta tanúsítványt:

1. amit az IdP/SP a felhasználó felé használ;
2. amit másik entitások (SP/IdP) felé használ.

A **felhasználók felé** olyan tanúsítványt kell használni, amelyben a felhasználók böngészője megbízik. Ez a tanúsítvány nem szerepel a föderációs metadatában, ellenben a webszerver konfigurációjában hivatkozni kell rá. Jellemzően valamilyen jól ismert CA-val (pl. DigiCert vagy letsencrypt) aláírt tanúsítványt, ami azt is jelenti, hogy rendszeresen cserélni kell őket.

A **föderációs metadatában** szereplő tanúsítványt elsősorban a föderációs alkalmazás (Shibboleth , SimpleSAMLphp) konfigurációjában kell megadni, mert ez az, amivel alá tudja írni az általa küldött üzeneteket, illetve dekódolni tudja a fogadott titkosított adatokat. Ez a tanúsítvány lehetőség szerint hosszú (10+ éves) lejáratú, self-signed tanúsítvány legyen.

- A webszerver (Apache, Jetty) konfigurációban csak akkor szerepeljen a föderációs metadatában szereplő tanúsítvány, ha azt szeretnénk, hogy az IdP támogassa az attribútumok back-channel történő letöltését (a Shibboleth IdP ilyen), esetleg ha valamilyen oknál fogva önálló AttributeAuthority-t építünk. Ha valami fut a szabványos https porton (pl. az IdP SSO szolgáltatása vagy egy SP), akkor az AttributeAuthority szolgáltatást nem tehetjük ide, ezért az jellemzően a 8443-as porton szokott figyelni.

Interoperabilitás mátrix

Interoperabilitás mátrix

Tesztelt szoftverek

- Shibboleth 2.0 IdP
  - metadata: papigw-shibboleth2-idp.xml
  - telepítési útvonal: /usr/local/shibboleth-idp-2.0.0
  - protokollok: SAML1.1, Shibboleth1.3, SAML2.0
- Shibboleth 2.0 SP
  - metadata: papigw-shibboleth2-sp.xml
  - telepítés Debian csomagból, konfiguráció /etc/shibboleth/ alatt
  - protokollok: SAML1.1, Shibboleth1.3, SAML2.0
- OpenSSO/FAM 8.0 CVS build - IdP
  - metadata: maszat-opensso-idp.xml
  - host: maszat.sch.bme.hu
  - protokollok: SAML2.0
- simpleSAMLphp (?) - SP
  - entityID: https://papigw.aai.niif.hu/simplesaml
  - protokollok: Shibboleth1.3, SAML2.0

Tesztelt protokollok és bindingok

- SAML2.0 AuthnRequest/AuthnResponse protokoll (Web browser SSO profil)
  - HTTP-GET / HTTP-POST binding
  - HTTP-GET / HTTP-Artifact binding
- SAML2.0 AttributeQuery protokoll
- SAML2.0 Single Logout

SAML2.0 Interoperabilitás mátrix

Jelmagyarázat:

- Single Sign on - AuthnRequest/Response (Attribute push-sal együtt)

- HTTP-POST - SAML2.0 HTTP-Post binding
- HTTP-Artifact - SAML2.0 HTTP-Artifact binding
- Attribute query - SAML2.0 Attribute Query protocol
- Signing / encryption - az Assertion aláírása, aláírt és titkosított Assertion feldolgozása
- Metadata management - mennyire egyszerű megoldani hogy az IdP és az SP ismerje egymást

A zöld-del jelölt funkciók tökéletesen működnek, a narancssárgák nem triviálisan, de működésre bírhatók (ilyenkor mindig van megjegyzés is hozzájuk), a pirossal jelölt funkciók nem működtek. Az áthúzott funkciókat nem implementálja az adott szoftverpáros.

Sure, here's the HTML conversion of the provided MediaWiki table:

	Shibboleth2 SP	OpenSSO SP	simpleSAMLphp SP
Shibboleth2 IdP	**Shib2-Shib2** Single Sign on HTTP-POST HTTP-Artifact Attribute query Signing / encryption Metadata management	**Shib2-OpenSSO** Single Sign on HTTP-POST HTTP-Artifact Attribute query Signing / encryption Metadata management	**Shib2-SimpleSAMLphp** Single Sign on HTTP-POST ~~HTTP-Artifact~~ ~~Attribute query~~ Signing / encryption Metadata management
OpenSSO IdP	**OpenSSO-Shib2** Single Sign on HTTP-POST HTTP-Artifact ~~Attribute query~~ Signing / encryption Metadata management	**OpenSSO-OpenSSO** Single Sign on HTTP-POST HTTP-Artifact Attribute query Signing / encryption Metadata management	**OpenSSO-simpleSAML** Single Sign on HTTP-POST HTTP-Artifact Attribute query Signing / encryption Metadata management
simpleSAMLphp IdP	**simpleSAML-Shib2** Single Sign on HTTP-POST HTTP-Artifact Attribute query Signing / encryption Metadata management	**simpleSAML-OpenSSO** Single Sign on HTTP-POST HTTP-Artifact Attribute query Signing / encryption Metadata management	**simpleSAML-simpleSAML** Single Sign on HTTP-POST HTTP-Artifact Attribute query Signing / encryption Metadata management

AA Testing

The following shell script uses *curl* to query a SAML2 Attribute Authority.

You need a valid principal (eduPersonPrincipalName) and the X.509 credentials of an existing Service Provider to use this script.

Source

```
#!/bin/bash

basedir=$(dirname $0)

# URL of the Attribute Authority
AA_URI="https://hexaa.eduid.hu:8443/simplesaml/module.php/aa/attributeserver.php"

# Testing principal (subject)
Principal="bajnokk@niif.hu"

# HEXAA cert
AACert="$basedir/keys/hexaa.eduid.hu-aa.crt"

# EntityID and credentials of the SP on behalf of which
# the request is made
ReqSP="https://sp.hexaa.eduid.hu/test"
ReqCert="$basedir/keys/test.sp.hexaa.eduid.hu-fed.crt"
ReqKey="$basedir/keys/test.sp.hexaa.eduid.hu-fed.key"



usage () {
        cat <<EOS
Usage: $0 [options]

Options:
  -a uri       Attribute Authority URI. Defaults to '$AA_URI'
  -C certfile  Attribute Authority metadata certificate in PEM format. Defaults to '$AACert'.
```

```
  -p principal Testing principal (user name / subject). Defaults to '$Principal'.
  -s entity    EntityID of the SP on behalf of which the request is made. Defaults to '$ReqSP'
  -k keyfile   Key file in PEM format containing the key of the SP used for the request.
Defaults to '$ReqKey'
  -c certfile  Cert file in PEM format containing the certificate of the SP used for the
request. Defaults to '$ReqCert'
EOS
        exit 3
}

# Get command line arguments
while getopts "a:p:s:k:c:h" opt; do
        case $opt in
                a)
                        AA_URI=$OPTARG
                        ;;
                C)
                        AACert=$OPTARG
                        ;;
                p)
                        Principal=$OPTARG
                        ;;
                s)
                        ReqSP=$OPTARG
                        ;;
                k)
                        ReqKey=$OPTARG
                        ;;
                c)
                        ReqCert=$OPTARG
                        ;;
                h)
                        usage
                        ;;
                \?)
                        usage
                        ;;
        esac
done
```

```
DATE=$(date --utc +%FT%TZ)
ReqID=$(hexdump -n 16 -e '4/4 "%08x" 1 "\n"' /dev/urandom)


read -r -d '' REQ_XML <<EOS
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <samlp:AttributeQuery xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion" ID="_$ReqID" IssueInstant="$DATE"
Version="2.0">
      <saml:Issuer>$ReqSP</saml:Issuer>
      <saml:Subject>
        <saml:NameID Format="urn:oid:1.3.6.1.4.1.5923.1.1.1.6">$Principal</saml:NameID>
      </saml:Subject>
    </samlp:AttributeQuery>
  </S:Body>
</S:Envelope>
EOS

#debug echo "$REQ_XML"

echo "$REQ_XML" | \
  curl --silent --show-error --cacert $AACert --cert $ReqCert --key $ReqKey \
       --header "Content-Type: text/xml;charset=UTF-8" --data @- $AA_URI
```

Validation of response

Signature validation:

```
xmlsec1 --verify --id-attr:ID "urn:oasis:names:tc:SAML:2.0:protocol:Response" --trusted-pem
$aacert $response 2>/dev/null
```

Content validation:

```
xmllint --xpath "//*['Attribute'](local-name()#bkmrk-)[@Name'$attribute']/*[local-
name()='AttributeValue']/text()" $response
```

EduIDTest

hosts file használata

A hosts file meglehetősen egyszerűen használható eszköz arra, hogy élesben működő szolgáltatás átalakítása során tudjunk. Mivel az eduID-ban elterjedten használt SAML profilokban az üzenetváltás a legtöbb esetben a böngészőn keresztül zajlik, ezért elegendő a böngészőt arra rávenni, hogy a tesztelni kívánt SP-t vagy IdP-t ne a világ által látott helyen, hanem a tesztelni kívánt ponton keresse. Ez a módszer **mind SP, mind IdP tesztelésre használható**.

Néhány tipp:

- A tesztelni kívánt IdP vagy SP föderációs konfigjában megadott tanúsítványok egyezzenek meg az éles rendszerben használt tanúsítvánnyal. Ennek hiányában az aláírás nem lesz valid, illetve az esetlegesen titkosítottan küldött adatokat nem lehet kibontani.
- Ha végül az új gép a régi nevén fog hallgatni, akkor nincs szükség a Resource Registry-ben módosítani az adatokon.
- *Bármely* SP-vel / IdP-vel tesztelhetjük az IdP-nket / SP-nket, hacsak nem rontunk el valamit (vö. első két pont) a távoli fél nem fogja észrevenni, hogy nem az "éles" partnerrel beszél.
- Tesztelésre privát IP cím is megfelelő, feltéve, hogy a saját gépünk eléri azt a tartományt.
- [SP]: HEXXA kapcsolat is tesztelhető a módszerrel, feltéve, hogy a tesztelt SP képes kapcsolatot nyitni a hexaa.eduid.hu 8443-as portjára. A HEXAA kizárólag a használt SSL tanúsítvány alapján azonosítja be a kérdezőt, így ugyanazt a tanúsítványt és entityID-t használva ugyanazt a választ kapjuk a tesztelt és az éles SP-n is.
- [IdP]: Az IdP-nk helyes működését például itt tudjuk ellenőrizni: https://attributes.eduid.hu

SP tesztelés

A szolgáltatásunk SAML integrációjának tesztelésére remekül használható eszköz a **https://samlidp.io**, amellyel pár kattintással készíthetünk magunknak teszt célokra IdP-t, amelynek megadható - akár a föderációban még nem regisztrált - SP metaadata is, így különböző felhasználói profilokkal próbálkozhatunk.

Publikációk

- AAI előadás a 2007-es Networkshopon
- AAI és Shibboleth (HBONE Workshop)

SSP EntityCategories

Ez a lap a simpleSAMLphp EntityCategories moduljának továbbfejlesztett változatához tartozó beállításokat ismerteti. Reményeink szerint a modul valamikor a simpleSAMLphp alapcsomagjának is része lesz, ám amíg ez nem történik meg, addig az alábbiak szerint érdemes eljárni, ha használni szeretnénk.

A modul forrása: https://github.com/sitya/simplesamlphp-module-entitycategories.git

Fontos, hogy a modul nem helyettesíti a core:AttributeLimit, vagy a niif:AttributeLimit modult, valamelyikkel együtt használandó!

Telepítés composeren keresztül

```
cd /path/to/simplesamlphp
composer require simplesamlphp/simplesamlphp-module-entitycategories:dev-master
composer config repositories.simplesamlphp/simplesamlphp-module-entitycategories vcs
https://github.com/sitya/simplesamlphp-module-entitycategories.git
composer update
```

Beállítás

Lévén egy AuthProc modullal van dolgunk, így a rendes authproc szekcióba kell elhelyeznünk valahol a sorban, a core:AttributeLimit, vagy niif:AttrobuteLimit előtt. Az alább részletezett alapbeállításokon túl (`default, strict, allowRequestedAttributes`) az egyes entityCategory-k URI-jait kell megadni, mint egy tömb kulcsát, és a hozzátartozó tömb értékeiként pedig a megengedett attribútumok URN-jeit. Tetszőleges számú entityCategory-t megadhatunk, a korábbi beállítások elsősorban azt szabályozzák, hogy mi történjen akkor, olyan entitással van dolgunk, akinek nincs a metadatájában megadva EntityCategory, vagy ha van, nem illeszkedik az általunk explicit beállított listában található EntityCategory-k valamelyikére.

M?köd? példa

```
75  => array(
       'class' => 'entitycategories:EntityCategory',
       'default' => true,
```

```
        'strict' => true,
        'allowRequestedAttributes' => true,
        'http://eduid.hu/category/registered-by-eduidhu' => array (),
        'http://www.geant.net/uri/dataprotection-code-of-conduct/v1' => array (),
        'http://refeds.org/category/research-and-scholarship' => array(
            'urn:oid:2.16.840.1.113730.3.1.241', #displayName
            'urn:oid:2.5.4.4', #sn
            'urn:oid:2.5.4.42', #givenName
            'urn:oid:0.9.2342.19200300.100.1.3', #mail
            'urn:oid:1.3.6.1.4.1.5923.1.1.1.6', #eduPersonPrincipalName
            'urn:oid:1.3.6.1.4.1.5923.1.1.1.9', #eduPersonScopedAffiliation
        ),
    ),

    80 => array(
        'class' => 'niif:AttributeLimit',
        'default' => true,
        'bilateralSPs' => array(
            'google.com' => array('mail'),
            'urn:federation:MicrosoftOnline' => array('IDPEmail', 'ImmutableID'),
        ),
    ),
```

A fenti példa az alábbiakat végzi:

1. a magyar föderáció által regisztrált entitások számára a Resource Registry-ben beállított attribútumok (`RequestedAttributes`) alapján történik az attribútum kiadás;
2. a GÉANT Code of Conduct entitás kategóriájú eduGAIN-es SP-k számára szintén `RequestedAttributes` alapján történik az attribútum kiadás;
3. a Research & Scholarship entitás kategóriájú eduGAIN-es SP-k számára kiadjuk a kategória által igényelt attribútumcsomagot
   - az ilyen SP-k `RequestedAttributes` alapján módosíthatnak az attribútum igényeiken
4. a fenti kategóriáknak nem megfelelő SP-k közül kizárólag a *bilateralSPs* tömbben megadott entitásoknak adunk ki attribútumot.

> **Warning**
>
> Az *entitycategories:EntityCategory* modul *strict* beállítása esetén a lokálisan felvett SP-k esetén nem lesz figyelembe véve az *attributes* tömbelem! Ez azt jelenti, hogy a nem a központi metaadatokból származó SP-ket fel kell venni az *niif:AttributeLimit* modul listájába. Ebből fakadóan ilyen esetben nem is használhatjuk a *core:AttributeLimit* modult.

Opciók

`default`

Logikai kapcsoló, `true/false` értékeket vehet fel. Amennyiben `true`, úgy a beállított EntityCategory-k alatt megadott attribútumkészletet akkor is kiadjuk az adott EntityCategory-val érkező SP-nek, ha az attribútumokat explicit, a `RequestedAttribute`-ok között nem kérte felsorolva. Ennek az R&S EntityCategorynál van különös jelentősége (a példában is ez szerepel), mert ott a specifikáció kimondja, hogy a meghatározott attributúmkészletet akkor is ki kell adni az R&S-t támogató IdP-nek, amennyiben nincsenek ezen attribútumok tételesen felsorolva, mint `RequestedAttributes`.

`strict`

Logikai kapcsoló, `true/false` értékeket vehet fel. Amennyiben `true`, úgy csak és kizárólag a modul konfigurációjában megadott EntityCategory-val érkező SP-k számára kerülnek attribútumok kiadásra, "ismeretlen" EntityCategory-val bíró, és EntityCategory nélküli SP-k számára nem kerül kiadásra semmi.

`allowRequestedAttributes`

Logikai kapcsoló, `true/false` értékeket vehet fel. Amennyiben `true`, úgy megengedjük, hogy egy SP a hozzátartozó EntityCategory konfigurációjában megadott attribútumlistán túl kért attribútumot kiadásra `RequestedAttributes`-on keresztül, `false` esetén csak a listában szereplő attribútumok kiadását engedi a modul. Amennyiben az érték `true` és a `strict` értéke `false`, úgy az "ismeretlen" EntityCategory-val bíró SP-k számára is a `RequestedAttributes` alapján kerülnek kiadásra az attribútumok.

NIIFSchema

NIIF LDAP Schema

Versioning

Current version: **2.3**

Change log

- changes in 2.4
  - extend niifAuthenticatedObject with niifEduroamVLANI
- changes in 2.3
  - extend niifAuthenticatedObject with niifValidityStart and niifExpireTime attributes
- changes since 2.1b2
  - add niifAuthenticatedObject and its niifEduroamPassword
- changes since 2.1b1
  - updated schema files
- changes since 2.0b5
  - add niifCertificateSHA1Fingerprint
  - add niifEduPersonArchiveCourse
  - add niifEduPersonHeldCourse
  - mark niifCertificateSubjectDN as obsolete

Schema files

- Sun DS format: 99-niifschema.ldif
- OpenLDAP format: niif-openldap.schema
- niifauthentication.ldif{:doenload="niifauthentication.ldif"}

ObjectClasses

niifPerson

	niifPerson
Parent	inetOrgPerson
OID	1.3.6.1.4.1.11914.0.0.0
Description	-
Mandatory attributes	-
Optional attributes	• niifPersonPrefix • niifPersonDegree • niifPersonPosition • ~~niifPrefix~~ • ~~niifStatus~~ • ~~niifTitle~~ • ~~niifCertificateSubjectDN~~ • niifPersonDateOfBirth • niifPersonActivityStatus • niifActiveMemberOf • niifPersonJoinDate • niifPersonQuitDate • niifPersonOrgID • niifPersonCityOfBirth • niifPersonCountryOfBirth • niifPersonMothersName • niifPersonIdentityNumber • niifPersonResidentalAddress • niifUniqueId

niifEduPerson

	niifEduPerson
Parent	eduPerson
OID	1.3.6.1.4.1.11914.0.0.9
Description	-
Mandatory attributes	-

	niifEduPerson
Optional attributes	• niifEduPersonFaculty • niifEduPersonFacultyDN • niifEduPersonMajor • niifEduPersonAcademicYear • niifEduPersonAttendedCourse • niifEduPersonArchiveCourse • niifEduPersonHeldCourse

niifAuthenticatedObject

	niifAuthenticatedObject
Parent	-
OID	1.3.6.1.4.1.11914.0.0.10
Description	An object having extra passwords or time-limited validity
Mandatory attributes	-
Optional attributes	• niifEduroamPassword • niifValidityStart • niifExpireTime • niifEduroamVLANID

Defined attributes

Attributes of niifPerson

niifUniqueID

	niifUniqueID
OID	1.3.6.1.4.1.11914.0.1.3
Description	Unique ID of a person.

	niifUniqueID
Semantics	<unique-local-ID>@<organization-domain> The <organization-domain> part equals to the main internet domain of the organization (i.e.: 'sztaki.hu').The <unique-local-ID> part is a sequence of letters (case insensitive) and numbers. It can be freely chosen by the home organization provided that the ID is unique within the scope of the organization and one and only one ID is assigned to every single person.
Values	nincs korlátozás
# of values	`single`
Availability	`organizational`
Syntax	`Directory String`
Examples	gmx3f0@bme.hu
Notes	It is up to the local policy to define how the <unique-local-ID> is generated and how long does it represent a user. However, assigning the same ID to another person (after the first person entry has been removed from the directory) is deprecated.It is recommended to import the user ID into <unique-local-ID> from a comprehensive user database (like Neptun and ETR at Hungarian Universities) if such a database exists.
Use in federation	-

niifPrefix

	niifPrefix
OID	1.3.6.1.4.1.11914.0.1.0
Description	**OBSOLETED**, use niifPersonPrefix
Semantics	-
Values	nincs korlátozás
# of values	`multi`
Availability	`-`
Syntax	`Directory String`
Examples	-
Notes	-
Use in federation	-

niifPersonPrefix

	niifPersonPrefix

OID	1.3.6.1.4.1.11914.0.1.165
Description	Prefix of the person's name
Semantics	A name should have only one prefix, multiple entitlements may be listed one after the other in the same value
Values	nincs korlátozás
# of values	`single`
Availability	`public`
Syntax	`Directory String`
Examples	Prof. Dr.
Notes	-
Use in federation	HREF_attribútum_specifikáció#schacPersonalTitle

niifStatus

	niifStatus
OID	1.3.6.1.4.1.11914.0.1.1
Description	**OBSOLETED**, use niifPersonDegree
Semantics	-
Values	nincs korlátozás
# of values	`multi`
Availability	`-`
Syntax	`Directory String`
Examples	-
Notes	-
Use in federation	-

niifPersonDegree

	niifPersonDegree
OID	1.3.6.1.4.1.11914.0.1.166
Description	Scientific degree of the person
Semantics	Only the highest degree should be stored.
Values	nincs korlátozás
# of values	`multi`
Availability	`public`

	niifPersonDegree
Syntax	`Directory String`
Examples	kandidátus
Notes	May be specified in multiple languages by the use of language tags
Use in federation	-

niifTitle

	niifTitle
OID	1.3.6.1.4.1.11914.0.1.2
Description	**OBSOLETED**, use niifPersonPosition
Semantics	-
Values	nincs korlátozás
# of values	`multi`
Availability	`-`
Syntax	`Directory String`
Examples	-
Notes	-
Use in federation	-

niifPersonPosition

	niifPersonPosition
OID	1.3.6.1.4.1.11914.0.1.167
Description	Position of the person within a department or organization
Semantics	-
Values	nincs korlátozás
# of values	`multi`
Availability	`-`
Syntax	`Directory String`
Examples	dékán

	niifPersonPosition
Notes	May be specified in multiple languages by the use of language tags.When a person is in relationship with multiple departments or organizations, no exact match between a status and a particular organization can be defined within an entry. To solve this situation, some relation-representing entry may be used, eg. use of the schacPersonalPosition attribute, which defines an URN format for this purpose.
Use in federation	-

niifCertificateSubjectDN

	niifCertificateSubjectDN
OID	1.3.6.1.4.1.11914.0.1.151
Description	Subject DN of the certificate of the entity. This attribute is **OBSOLETED**, use niifCertificateSHA1Fingerprint instead.
Semantics	The value must describe a DN which identifies the certificate subject of the entity.
Values	Unlike standard LDAP DN, this value must contain the same number of spaces between DN elements as it is tied in the certificate.
# of values	`multi`
Availability	`-`
Syntax	`Directory String`
Examples	-
Notes	Although it is possible for an entity to have more than one certificates at the same time, this kind of usage is deprecated.When used, this attribute must be indexed. This attribute may be applied to any kind of entities that can be certified with an X.509 certificate, including persons, servers, network nodes, etc.
Use in federation	-

niifCertificateSHA1Fingerprint

	niifCertificateSHA1Fingerprint
OID	1.3.6.1.4.1.11914.0.1.173
Description	Fingerprint of a certificate which belongs to the subject.
Semantics	Multiple fingerprints may be stored in this attribute, if a subject has multiple valid certificates. This attribute uses case insensitive matching rule.

	niifCertificateSHA1Fingerprint
Values	SHA-1 hash in hexadecimal string format without any separator characters.
# of values	`multi`
Availability	`-`
Syntax	`IA5String`
Examples	fe6d5980e2c02912024054cec114ee53ebeb2e6c
Notes	-
Use in federation	-

niifPersonDateOfBirth

	niifPersonDateOfBirth
OID	1.3.6.1.4.1.11914.0.1.152
Description	Date of birth of the person
Semantics	-
Values	YYYYMMDD date format according to RFC 3339 'full-date' format
# of values	`single`
Availability	`confidential`
Syntax	`Directory String`
Examples	19800316
Notes	It's recommended to use the schacDateOfBirth attribute instead, as it has the same syntax and semantics.
Use in federation	HREF_attribútum_specifikáció#schacDateOfBirth, HREF_attribútum_specifikáció#schacYearOfBirth

niifPersonActivityStatus

	niifPersonActivityStatus
OID	1.3.6.1.4.1.11914.0.1.153
Description	Activity status
Semantics	Describes whether the person is an active employee/student of the home organization
Values	One of the term 'active' or 'inactive'
# of values	`single`

	niifPersonActivityStatus
Availability	`organizational`
Syntax	`Directory String`
Examples	-
Notes	-
Use in federation	-

niifActiveMemberOf

	niifActiveMemberOf
OID	1.3.6.1.4.1.11914.0.1.168
Description	DN of a group entry to which the entity currently belongs.
Semantics	-
Values	nincs korlátozás
# of values	`multi`
Availability	`-`
Syntax	`Directory String`
Examples	-
Notes	As a special case, this attribute may be used to keep a record of a student's active major(s), but it's recommended to use niifEduPersonMajor instead.
Use in federation	-

niifPersonJoinDate

	niifPersonJoinDate
OID	1.3.6.1.4.1.11914.0.1.169
Description	Date of joining to the organization
Semantics	Date of joining to the organization. For students it may represent the first date of enrollment.
Values	YYYYMMDD date format according to RFC 3339 'full-date' format
# of values	`single`
Availability	`organizational`
Syntax	`Integer`
Examples	19980901

	niifPersonJoinDate
Notes	-
Use in federation	-

niifPersonQuitDate

	niifPersonQuitDate
OID	1.3.6.1.4.1.11914.0.1.170
Description	Date of leaving the organization
Semantics	-
Values	YYYYMMDD date format according to RFC 3339 'full-date' format.
# of values	`single`
Availability	`organizational`
Syntax	`Integer`
Examples	20030627
Notes	If this date is in the past, niifPersonActivityStatus must be 'inactive', and the user should be locked out.
Use in federation	-

niifPersonOrgID

	niifPersonOrgID
OID	1.3.6.1.4.1.11914.0.1.154
Description	Organizational ID of a person
Semantics	ID of a person in a comprehensive organizational user database if such a database exists. This ID shall be unique within the organization.It is strongly recommended to use the <unique-local-ID> part of the niifUniqueID as a value for niifPersonOrgID.
Values	For integration with niifUniqueID, value must not contain the '@' mark.
# of values	`single`
Availability	`-`
Syntax	`Directory String`
Examples	-

	niifPersonOrgID
Notes	It is recommended to import the user ID into <unique-local-ID> from a comprehensive user database (like Neptun and ETR at Hungarian Universities) if such a database exists.This attribute is for facilitating the use of user ID's in intra-organizational applications in cases when standard uid attribute can not be applied for some reason.
Use in federation	-

niifPersonCityOfBirth

	niifPersonCityOfBirth
OID	1.3.6.1.4.1.11914.0.1.155
Description	The city or settlement where the person was born
Semantics	Name of the city or settlement where the person was born. If the place of birth is outside the borders of Hungary, the name may be given in Hungarian.
Values	nincs korlátozás
# of values	`single`
Availability	`confidential`
Syntax	`Directory String`
Examples	Kolozsvár
Notes	It's recommended to use the schacPlaceOfBirth attribute instead.
Use in federation	-

niifPersonCountryOfBirth

	niifPersonCountryOfBirth
OID	1.3.6.1.4.1.11914.0.1.156
Description	The country where the person was born
Semantics	Name of the country where the person was born. If the place of birth is outside the borders of Hungary, the name must be given in Hungarian.
Values	nincs korlátozás
# of values	`single`
Availability	`confidential`
Syntax	`Directory String`
Examples	Románia

	niifPersonCountryOfBirth
Notes	It's recommended to use the schacPlaceOfBirth attribute instead.
Use in federation	-

niifPersonMothersName

	niifPersonMothersName
OID	1.3.6.1.4.1.11914.0.1.157
Description	Name of the mother of the person
Semantics	Maiden name of the mother of the person.
Values	nincs korlátozás
# of values	`single`
Availability	`confidential`
Syntax	`Directory String`
Examples	-
Notes	-
Use in federation	-

niifPersonIdentityNumber

	niifPersonIdentityNumber
OID	1.3.6.1.4.1.11914.0.1.158
Description	Number of the Identity Card
Semantics	Number of the Identity Card of the person or Passport Number for those who are non-Hungarian citizens
Values	nincs korlátozás
# of values	`single`
Availability	`confidential`
Syntax	`Directory String`
Examples	329906AA
Notes	Every Hungarian citizen by the age of 14 receives an Identity Card. For foreigners, Passport Number should be used. This number should never be made public.Format of the code may vary as numbering scheme has been changed in the recent years.It's recommended to use the URN-formatted schacPersonalUniqueID attribute instead.
Use in federation	-

niifPersonResidentialAddress

	niifPersonResidentialAddress
OID	1.3.6.1.4.1.11914.0.1.159
Description	Home address of the person
Semantics	Permanent home address of the person. The postal code, the name of the city, street and apartment number shall be included.
Values	nincs korlátozás
# of values	`single`
Availability	`confidential`
Syntax	`Directory String`
Examples	1234 Budapest, Harap u. 3.
Notes	-
Use in federation	-

Attributes of niifEduPerson

niifEduPersonFaculty

	niifEduPersonFaculty
OID	1.3.6.1.4.1.11914.0.1.160
Description	Faculty of the person
Semantics	Full name of the faculty the person belongs to. List of accredited faculties can be found here (in institutional order): http://www.mab.hu/doc/akkrint040106.doc (in Hungarian)
Values	nincs korlátozás
# of values	`multi`
Availability	`organizational`
Syntax	`Directory String`
Examples	Villamosmérnöki és Informatikai Kar
Notes	Local directory policy may mark this attribute as mandatory.It is recommended if an LDAP entry has niifEduPersonFaculty attribute set, it should also have an eduPersonFacultyDN attribute pointing to the entry of the faculty.
Use in federation	-

niifEduPersonFacultyDN

	niifEduPersonFacultyDN
OID	1.3.6.1.4.1.11914.0.1.161
Description	Pointer to the faculty of the person
Semantics	-
Values	nincs korlátozás
# of values	`multi`
Availability	`organizational`
Syntax	`DN`
Examples	-
Notes	This attribute has a meaning only if the person participates in education (eduPersonAffiliation=student or faculty)Local directory policy may mark this attribute as mandatory.
Use in federation	-

niifEduPersonMajor

	niifEduPersonMajor
OID	1.3.6.1.4.1.11914.0.1.162
Description	Major(s) of the student
Semantics	Majors are defined at http://www.mab.hu/listak2.html (in Hungarian)
Values	nincs korlátozás
# of values	`multi`
Availability	`organizational`
Syntax	`Directory String`
Examples	műszaki informatikai
Notes	This attribute has a meaning only if the person is a student (eduPersonAffiliation=student)Local directory policy may mark this attribute as mandatory for students.
Use in federation	-

niifEduPersonAcademicYear

	niifEduPersonAcademicYear
OID	1.3.6.1.4.1.11914.0.1.163

	niifEduPersonAcademicYear
Description	Current academic year of the student. This attribute is **DEPRECATED**.
Semantics	-
Values	Integer numbers from 1 to the number of years required for graduation
# of values	`multi`
Availability	`confidential`
Syntax	`Directory String`
Examples	-
Notes	This attribute has a meaning only if the person is a student (eduPersonAffiliation = student)It is unclear that if a student has more than one majors, the number of which should be stored in this attribute and how a connection between major and year can be set up. As the concept of academic year gets lesser important (and not well-defined), depending on the value of this attribute is deprecated.
Use in federation	-

niifEduPersonAttendedCourse

	niifEduPersonAttendedCourse
OID	1.3.6.1.4.1.11914.0.1.164
Description	Code of the courses the student attends to in the current semester
Semantics	-
Values	Course codes defined in the student calendar.
# of values	`multi`
Availability	`-`
Syntax	`Directory String`
Examples	BMEVIMM1234BMEVIMA3214
Notes	This attribute has a meaning only if the person is a student (eduPersonAffiliation = student), and the values should be taken from the student calendar.
Use in federation	-

niifEduPersonArchiveCourse

	niifEduPersonArchiveCourse

OID	1.3.6.1.4.1.11914.0.1.171
Description	Code of the courses the student have ever attended
Semantics	-
Values	Course codes defined in the student calendar.
# of values	`multi`
Availability	`-`
Syntax	`Directory String`
Examples	-
Notes	This attribute has a meaning only if the person is a student (eduPersonAffiliation = student), and the values should be taken from the student calendar.
Use in federation	-

niifEduPersonHeldCourse

	niifEduPersonHeldCourse
OID	1.3.6.1.4.1.11914.0.1.172
Description	Code of the courses which are associated with the faculty member or professor in the current semester.
Semantics	-
Values	Course codes defined in the student calendar.
# of values	`multi`
Availability	`-`
Syntax	`Directory String`
Examples	-
Notes	This attribute has a meaning only if the person is a faculty (eduPersonAffiliation = faculty), the values should be taken from the student calendar.For authorization decision reasons, courses from previous semester(s) might appear in this attribute if the local policy needs these additional values.
Use in federation	-

Attributes of niifAuthenticatedObject

niifEduroamPassword

	niifEduroamPassword

OID	1.3.6.1.4.1.11914.0.1.4
Description	Clear text password for eduroam authentication
Semantics	-
Values	nincs korlátozás
# of values	`multi`
Availability	`-`
Syntax	`Octet String`
Examples	-
Notes	SUP userPassword
Use in federation	-

niifValidityStart

	niifValidityStart
OID	1.3.6.1.4.1.11914.0.1.5
Description	When to enable this account
Semantics	-
Values	nincs korlátozás
# of values	`single`
Availability	`-`
Syntax	`Generalized Time`
Examples	-
Notes	-
Use in federation	-

niifExpireTime

	niifExpireTime
OID	1.3.6.1.4.1.11914.0.1.6
Description	When to disable this account
Semantics	-
Values	nincs korlátozás
# of values	`single`
Availability	`-`
Syntax	`Generalized Time`

	niifExpireTime
Examples	-
Notes	-
Use in federation	-

niifEduroamVLANID

	niifEduroamVLANID
OID	1.3.6.1.4.1.11914.0.1.7
Description	VLAN ID for eduroam
Semantics	-
Values	-
# of values	`single`
Availability	`-`
Syntax	`Numeric String`
Examples	-
Notes	-
Use in federation	-

Szövetségi alapon történ? felhasználó azonosítás Huntéka könyvtári alkalmazásokban

Könyvtári alkalmazás használati esetei

Könyvtári és föderatív azonosságok

A könyvtári alkalmazás olvasói nyilvántartásában szereplő és a föderatív azonosítás szolgáltatónál lévő azonosságokat a könyvtári alkalmazásban célszerűen össze kell kapcsolni. Ez az identitás összekapcsolás mindig feltételezi a felhasználó aktív közreműködését, e nélkül az identitások összekapcsolása nem jöhet létre.

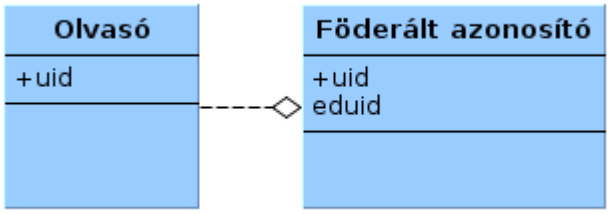
Egy föderatív azonosítással elérhető, SP-vel védett Online könyvtári beiratkozás szolgáltatásnál ez egy tájékoztató szöveg megjelenése és tudomásul vétele után történik meg, a már hagyományos módon beiratkozott, az alkalmazás olvasói nyilvántartásában már szereplő olvasók “könyvtári identitását” viszont külön eljárással kell összekapcsolni az IdP által azonosítottakkal.

Ha a könyvtári alkalmazásban már létezik webes azonosítás, akkor célszerű egy olyan komponenst beilleszteni, ahol a felhasználó mind a könyvtári azonosságát (könyvtári azonosító - jelszó), mind a föderatív azonosságát tudja bizonyítani, és a komponens összeköti a két azonosítót a kapcsoló táblában (account linking). A tájékoztató szöveg megjelenítése és annak elfogadása ekkor történhet meg.

Az MTA SZTAKI könyvtára esetében a felhasználók nem tudtak bejelentkezni a webes felületen, az identitás összekapcsolást segítő kulcs az email cím lehetett, ez alapján az összerendelések nagy tömegét elvégezhettük, a maradék azonosítót pedig kézzel, további adatok összehasonlításával, vagy közvetlenül az olvasó megkeresésével lehetett elvégezni.

Identitások összekapcsolása

A könyvtári rendszert fel kell készíteni a szövetségi azonosítás során használt azonosítójukkal érkező felhasználók fogadására. Ehhez szükség van egy kapcsoló táblára, ahol a könyvtári azonosítót és a föderált azonosítókat tároljuk.ű



Bejelentkezés során bármely eduid-vel azonosítja a felhasználót a szóbanforgó Home IdP, az alkalmazás kikeresi a megfelelő könyvtári alkalmazásbeli azonosítót, és a továbbiakban az alkalmazás ezt használja a munkamenet során.

Beiratkozás

Ha a felhasználó olyan azonosítóval jön a Home IdP-től, ami nincs egyetlen alkalmazás azonosítóhoz sem kapcsolva, a beiratkozás folyamatát kell végrehajtani.

A beiratkozás folyamán létre kell hozni az olvasó táblában az új olvasót, és a Home IdP-től kapott személyes adatokkal fel kell tölteni az olvasó rekordot.

A “Föderált azonosító” kapcsolótáblában létre kell hozni a könyvtári azonosítót és a föderált azonosítót összerendelő rekordot.

Kiiratkozás

Felhasználó kiiratását az OIOSAML.java nem támogatja, az azonosító lejáratásával lehet a felhasználó jogosultságait korlátozni. Tervezésnél meg kell fontolni, hogy egy-egy IdP által azonosított felhasználó beiratásánál mennyi az az idő, amíg az olvasót aktívnak szeretnénk tudni. Megvizsgálva néhány könyvtár Használati szabályzatát a megoldás megfelel az általános gyakorlatnak. A könyvtárak beiratkozáskor kiállított olvasójegyét bizonyos időre szóló érvényességgel (általában 1 év) állítják ki, ha az olvasó ennek elteltével nem hosszabbítja meg olvasójegye érvényességét, az lejár.

Azonosítás és attribútumok átadása

A LoginFilter által védett URL-re irányuló kéréskor a SAML Assertion és az általa hordozott attribútumok a dk.itst.oiosaml.sp.UserAssertion objektumból lekérdezhetőek. Ezen objektum csak a védett URL-eken érhető el a dk.itst.oiosaml.sp.UserAssertionHolder.get() metódus meghívásával.

Ha csak a login funkciót védjük SAML SP-vel, érdemes a user session-be átmenteni a szükséges attribútumokat, ha azokat más funkciókkal is el kell érni.

Attribútum mappingről igény szerint a fejlesztés során kell gondoskodnunk, az OIOSAML.java nem támogat ilyen funkciót.

Példa az eduPersonPrincipalName attribútum kinyeréséről:

```
UserAssertion ua = UserAssertionHolder.get();
String eppn = ua.getAttribute(“urn:oid:1.3.6.1.4.1.5923.1.1.1.6 ”);
```

A rendszerfejlesztéshez kapcsolódó dokumentáció az alábbi URL-en található:

[https://svn.softwareborsen.dk/oiosaml.java/sp/trunk/docs/developers.html]

Attribútumok frissítése

Hogy a könyvtári rendszerben nyilvántartott olvasók adatai naprakészek legyenek, a Home IdP-től származó attribútumokat nem csak a beiratkozáskor, hanem a mindennapi használatkor is igénybe vehetjük.

Az IdP-től kapott attribútumokat a könyvtári alkalmazás ellenőrzi, és ha az adott attribútum még nem szerepel a nyilvántartásában, azt rögzíti.

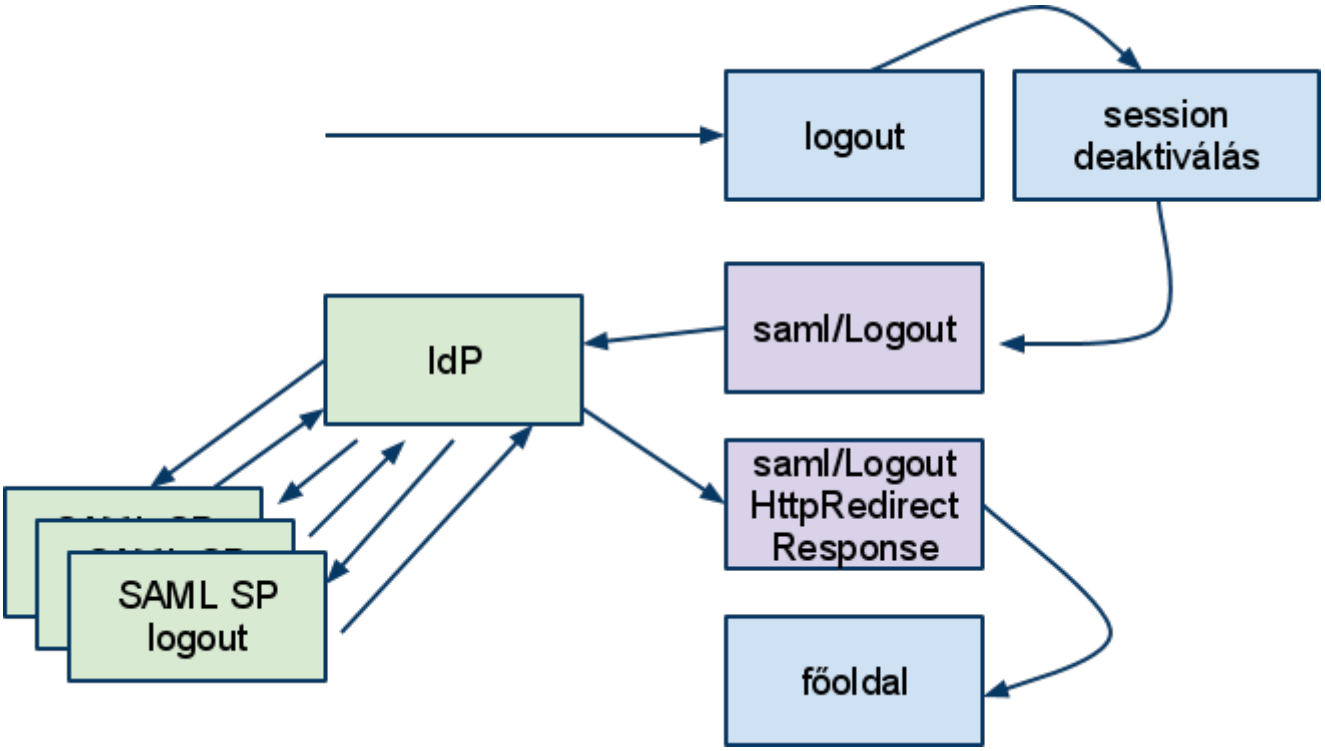
Az alkalmazás saját szabály-rendszere döntheti el, hogy a meglévő értékeket lecseréli az Home IdP-től kapottal, vagy csak hozzáadja új értékként a már meglévőhöz.

Kijelentkezés

SP által kezdeményezett SLO

Az alkalmazásból történő kijelentkezést a session-változók törlésével kell megoldani, és biztosítani kell a Single Logout folyamat elindítását. Az OIOSAML.java esetében ezt a saml/Logout URL-re történő redirect-tel lehet kezdeményezni.

```
HttpSession session = req.getSession(false);
res.sendRedirect("saml/Logout");
```



IdP által kezdeményezett SLO

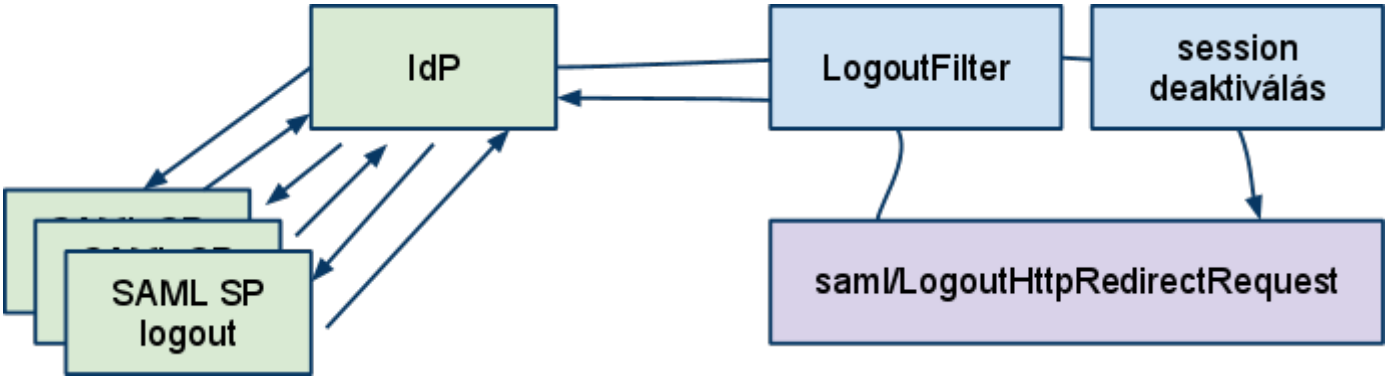
Ha a Single Logout-ot a felhasználó egy másik alkalmazásnál kezdeményezte, és kérte, hogy valamennyi aktív alkalmazásból léptesse ki a rendszer, az IdP a könyvtári alkalmazásnak egy Logout Request-et küld a böngésző átirányításának segítségével. A kérésről a SAML SP-n kívül az alkalmazásnak is tudnia kell, hogy a felhasználó session-jén a kilépést érvényre juttassa. Ezt a SAML SP SLO endpoint-jai elé tett filter segítségével lehet a legegyszerűbben végrehajtani.

```
LogoutFilter.java

import java.io.IOException;
import javax.servlet.Filter;
import javax.servlet.FilterChain;
import javax.servlet.FilterConfig;
import javax.servlet.ServletException;
import javax.servlet.ServletRequest;
import javax.servlet.ServletResponse;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpSession;

import org.apache.log4j.Logger;

/**
* Servlet Filter implementation class LogoutFilter
*/
```

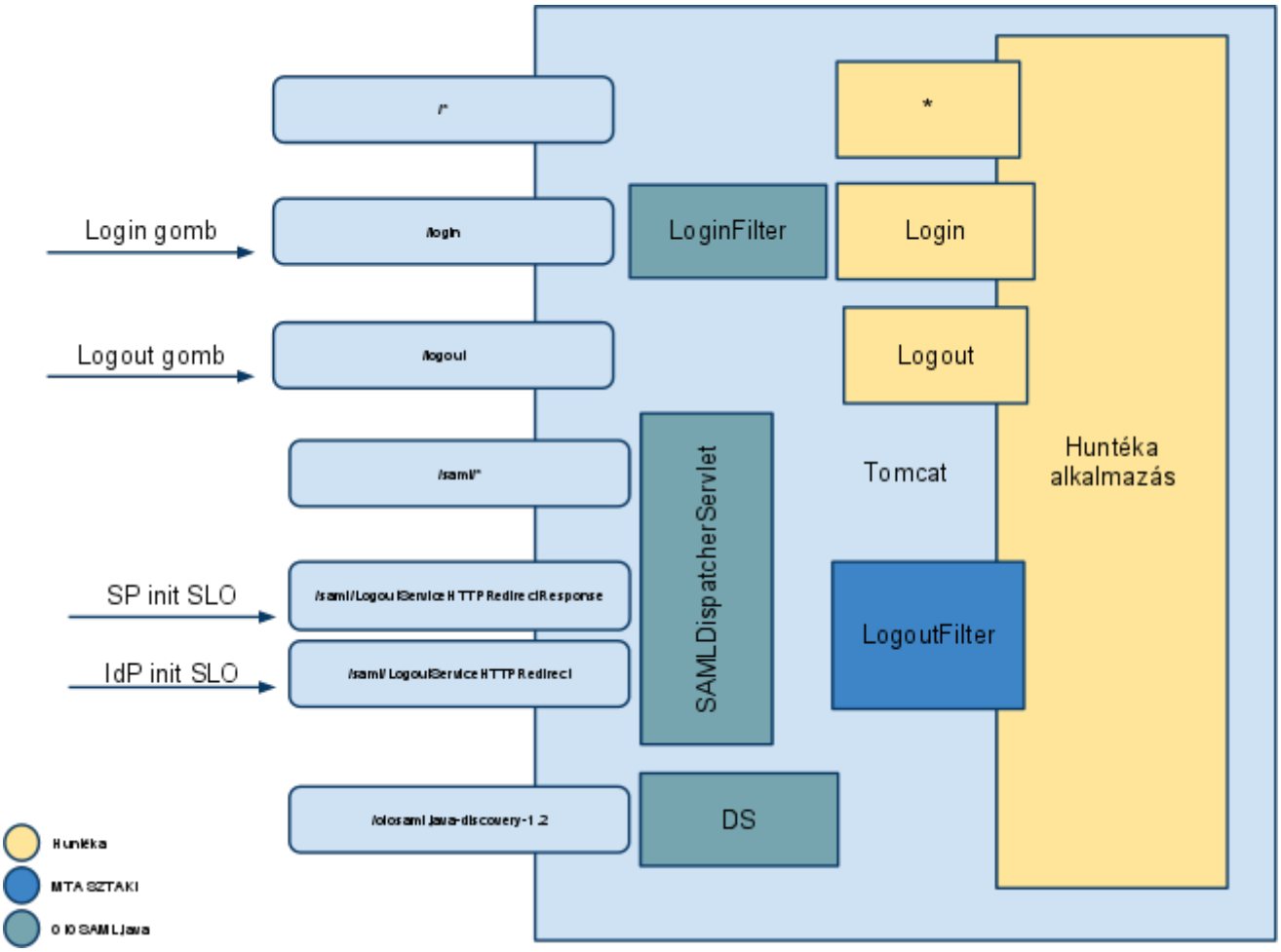
```
public class LogoutFilter implements Filter {

   /**
    * Default constructor.
    */
  private static final Logger log = Logger.getLogger(LogoutFilter.class);

   public LogoutFilter() {
   }

  /**
   * @see Filter#destroy()
   */
  public void destroy() {
  }

  /**
   * @see Filter#doFilter(ServletRequest, ServletResponse, FilterChain)
   */
  public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain)
throws IOException, ServletException {
      chain.doFilter(request, response);
      log.debug("Enter into LoginFilter");
      HttpServletRequest servletRequest = ((HttpServletRequest) request);
      HttpSession session = servletRequest.getSession(false);
      if (session != null){
          log.debug("Remove application related session attributes.");
          session.removeAttribute("aai_auth");
      }
  }

  /**
   * @see Filter#init(FilterConfig)
   */
  public void init(FilterConfig fConfig) throws ServletException {
  }
}
```

web.xml

```
...
 <filter>
   <filter-name>LogoutFilter</filter-name>
   <filter-class>LogoutFilter</filter-class>
</filter>
<filter-mapping>
   <filter-name>LogoutFilter</filter-name>
   <url-pattern>/saml/LogoutServiceHTTPRedirect</url-pattern>
   <url-pattern>/saml/LogoutServiceHTTPRedirectResponse</url-pattern>
</filter-mapping>
...
```



Rendszer architektúra



OIOSAML.java, mint SP komponens

Telepítés

Az oiosaml.java letöltése után a következő lépésekkel telepíthetjük alkalmazásunkhoz az oiosaml.java SP komponenst:

1. Másoljuk a lib/* fileokat a könyvtári alkalamzás WEB-INF/lib könyvtárába
2. Egészítsük ki az alkalmazás web.xml leírófile-ját a következőkkel:

```
<context-param>
   <param-name>oiosaml-j.home</param-name>
   <param-value>/path/to/oiosaml.home</param-value>
</context-param>

<servlet>
   <servlet-name>SAMLDispatcherServlet</servlet-name>
   <servlet-class>dk.itst.oiosaml.sp.service.DispatcherServlet</servlet-class>
</servlet>
```

```
<servlet-mapping>
   <servlet-name>SAMLDispatcherServlet</servlet-name>
   <url-pattern>/saml/*</url-pattern>
</servlet-mapping>

<filter>
   <filter-name>LoginFilter</filter-name>
   <filter-class>dk.itst.oiosaml.sp.service.SPFilter</filter-class>
</filter>
<filter-mapping>
   <filter-name>LoginFilter</filter-name>
   <url-pattern>/protected/*</url-pattern>
</filter-mapping>
```

A SAMLDispatcherServlet servlet végzi el a SAML-lal kapcsolatos kéréseket, ajánlott a /saml/* URL alá definiálni.

A LoginFilter filter által védett URL-lek (példánkban /protected/*) csak SAML-os azonosítással érhetőek el, ezen URL-ek alatt érhetők el különböző objektumokban TODO a SAML Assertion-ben átadott attribútumértékek.

Telepítés után a az OIOSAML.java ellenőrzi konfigurációját, és ha a rendszer nincs helyesen konfigurálva, elindul az autoconfig folyamat, ahol a rendszert a webes felületről konfigurálhatjuk. A konfigurációs folyamat végén az SP aláírt meta-adata letölthető.

Konfiguráció

Az oiosaml.java konfigurációja az oiosaml-sp.properties file-ban található.

Telepítés után a következő módosításokat érdemes megtenni a fileban:

```
oiosaml-sp.assurancelevel=0
```

Az elfogadott azonosítási szintet állíthatjuk itt be. Mivel a föderációban ilyet egyelőre nem használunk, és egyes IdP-k nem támogatják (simpleSAMLphp) ezt az attribútumot, állítsuk 0-ra.

```
oiosaml-sp.uri.home=../
```

Az alkalmazásunk kezdő lapja. SLO után ide irányítódik át a felhasználó böngészője.

A konfigurációs beállítások teljes dokumentációja az alábbi URL-en található:

https://svn.softwareborsen.dk/oiosaml.java/sp/trunk/docs/configuration.html

Meta-adat kezelés

A SAML SP komponens működéséhez elengedhetetlen, hogy a vele bizalmi szövetségben lévő IdP-k meta-adatait ismerje.

Az OIOSAML.java meta-adatformátumának a legfőbb jellemzője, hogy az egyes entitásokat egyenként külön file-ban kell eltárolni. Az szóban forgó file-okat a metadata/IdP alkönyvtárba kell helyezni XML formátumban, a név-konvenció csak annyit követel meg, hogy .xml-re végződjön a file neve.

Fontos tudni, hogy az OIOSAML.java csak induláskor tölti be a meta-adatokat, így ha azokat változtatjuk, az alkalmazást újra kell indítani, hogy a változtatások érvényre jussanak.

Discovery Service

Ha az SP több IdP-vel is bizalmi szövetséget alkot, szükség van Discovery Service-re is, hogy a felhasználó eldönthesse, melyik IdP-nél kívánja magát azonosítani.

Az OIOSAML.java tartalmaz egy discovery service servlet-et, ami az IdP metadatáiból állít össze egy weboldalt, ahol a felhasználó kiválaszthatja a saját IdP-jét.

Telepítése az oiosaml.java-discovery..*war file egyszerű deploy-olásával történik, beállításra az oiosaml-sp.properties file oiosaml-sp.discovery.* változók használhatóak.

A discovery service-hez kapcsolódó dokumentáció az alábbi URL-en található:

https://svn.softwareborsen.dk/oiosaml.java/sp/trunk/docs/discovery.html

Metadata frissítés

A meta-adatok frissítése kapcsán meg kell oldani a HREF letölthető teljes meta-adatának entitásonként külön file-ba történő szétvágását.

Ezt a feladatot a mdsigner-j-cli-2.0-rc2-jar-with-dependencies.jar nevű program végzi el.

Működéséhez szükséges a metadata aláíró root certificate, ezzel lehet ellenőrizni a metadata hitelességét.

A programot célszerű rendszeresen, pl. naponta futtatni, egy külön folyamatban leválogatni a releváns IdP-ket, ellenőrizni a módosulásokat, módosulás esetén frissíteni a SAML SP metadata könyvtárát és újraindítani a webalkalmazást.

IdP whichone

A legfontosabb, hogy föderációs oldalról nincs különbség a Shibboleth és a SimpleSAML között, azaz bármelyik IdP bármelyik SP-vel képes együttműködni. Föderációs szempontból a különbség mindössze két **apróság** :

1. *Single Logout*: A Shibboleth nem támogatja a Single Logout-ot, mert a fejlesztők úgy gondolják, hogy ezt nem lehet rendesen implementálni. A régi IdP-hez az NIIF csinált SLO-t, de ez nem került a hivatalos kiadásba és a friss verziókkal már nem működik. Az új IdP-kben van olyan funkcionalitás, hogy a logout-ot kezdeményező SP-ből és az IdP-ből kiléphet a felhasználó, de ez a többi SP-nél esetleg érvényes session-jét nem érinti. A SimpleSAMLphp támogatja az SLO-t, minden örökletes betegségével együtt.
2. *ECP*: A Shibboleth támogatja az ECP profilt, ami akkor használható, ha nem webes kliensekkel akarjuk használni az IdP-t. Praktikusan az Office365 és néhány (nálunk nem használatos) grides alkalmazás tartozik ide. A SimpleSAML ECP támogatásán még dolgoznak, de addig is van arra megoldás, hogy ECP nélkül is lehessen Office365-öt használni.

Ezek a gyakorlatban nem fontos dolgok. Az érdemi döntési pont az, hogy **melyik környezethez értetek jobban**:

- **ha Apache + PHP, akkor SimpleSAMLphp;**
- **ha Jetty + Java, akkor Shibboleth.**

Az IdP-hez javasolt konfiguráció:

- Linux (tetszőleges)
- 2GB RAM, 2 VCPU
- felhasználók által ismert tanúsítvány

VO

Terminológia

- **Virtual Organization (VO)**: Felhasználókból álló csoport
- **VO Service**: Alkalmazás, melyet az egyes VO tagjai használnak
- **VO Service Group**: Alkalmazások egy csoportja, melyek egy közös entityID alatt kerülnek összefogásra, s melyek számára egy IdP ugyanazt a targetedID-t adja ki
- **VO Platform**: Az a szoftverkörnyezet, amely által lehetővé válik a VO tagok egyes VO Service-kbeli munkája.
- **VO Attributes**: Azok az attribútumok, melyek az adott felhasználó VO Platform-beli munkájához kapcsolódnak.
- **VO Management**: Felület, amelyen az felhasználók VO attribútumait lehet konfigurálni.
- **VO Attribute Authority**: Speciális IdP, melyet az egyes VO Service-k kérdeznek, ha egy-egy felhasználó VO Platform-beli attribútumára van szükségük.

Koncepció

Az alapprobléma

Az egyre inkább terebélyesedő föderációkban előbb-utóbb mindenhol előkerül a kérdés: miként lehet egyszerűen és hatékonyan megvalósítani egymástól független, különböző intézmények által azonosított felhasználók projektszintű együttműködését, tehát ad-hoc csoportok létrehozását? Mindezt úgy, hogy ez ne kívánjon speciális IdP/SP konfigurációt, .htaccess bütykölést, stb... hanem az egyes alkalmazások egy adott helyről be tudják szerezni a felhasználó az adott alkalmazáshoz kapcsolódó attribútumait.

Az ötlet

- Az egyes csoportokhoz tartozó felhasználók csoportmunkával kapcsolatos attribútumait gyűjtsük egy adatbázisba, tegyünk mellé egy webes felületet, így lehetőségünk lesz rá, hogy a felügyeletünk alatt álló csoportokat könnyen tudjuk adminisztrálni.
- Állítsunk üzembe egy IdP-t, amely autentikációt nem végez, kizárólag Attribute Authority szerepkörben dolgozik.

Tehát ha ehhez az IdP-hez fordul egy SP egy állandóazonosítóval, akkor az IdP az adott állandóazonosítóhoz tartozó attribútumokat kiadja számára, ezzel lehetővé téve, hogy az SP ismerhesse, hogy az adott felhasználó milyen csoportokban milyen szerepeket tölt be, hogy az SP

által védett alkalmazás ezek alapján állapíthassa meg a felhasználó alkalmazásszintű jogosultságait. Például: a felhasználó belép egy wiki alkalmazásba föderatív azonosítással, ekkor a felhasználót azonosító IdP-től az SP kap néhány attribútumot, ám azt - nyilvánvaló okokból - nem tárolja az IdP, hogy a felhasználó milyen ad-hoc csoportoknak tagjai, és ezeken a csoportokon belül milyen szerepkört tölt be, így az SP tovább kérdez, és ha a VO AA-tól megtudja, hogy a felhasználó tagja a wiki-adminisztrátorok csoportjának, akkor ezt az információt is tovább adja az alkalmazásnak valamilyen attribútum értékeként.

Az ötlet fenti két pontjának együttes megvalósítását nevezzük **VO Platformnak**. Egy felhasználó bármilyen ad-hoc csoportnak tagja lehet, így nem szükségszerű, hogy

--> folyt köv innen.

Ha az egyes VO-k által használt szolgáltatásokat (VO Service) egy csoportba rendezzük (VO Service Group), akkor lehetőségünk lesz a

Kérdések