Authorizing Access to SPs

SWITCHaai Team
aai@switch.ch

Berne, 13 August 2014

```
Require valid-user
```

"Considered harmful!"

Don't accept just any valid user

- The single access rule `Require valid-user` is usually not well-suited. This would allow any AAI user to access your resource, including guest users and VHO users. In most cases, that's not what you want to allow.

- You should require specific attribute values, e.g. specific affiliations like `staff/student/faculty`. (Guest and VHO users just have affiliation `affiliate`).

- You should take care while designing access control rules.

Content Protection and Session Initiation

- Before access control can occur, a Shibboleth session must be initiated on the SP.
  - Session initiation and content protection go hand in hand.
  - Session initiation is done by the Shibboleth SP software.

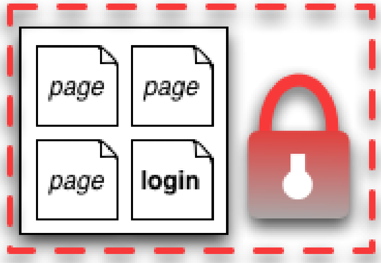
- Requiring a session means the user has to authenticate.

- Only authenticated users can access protected content.

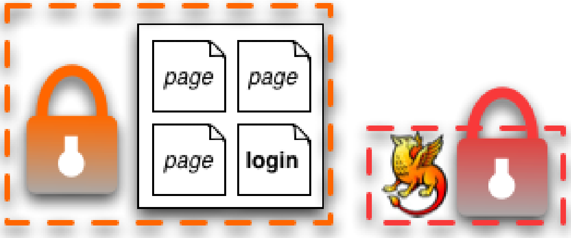
- AAI attributes are available only if a valid session has been initiated.

Where to Require a Shibboleth Session

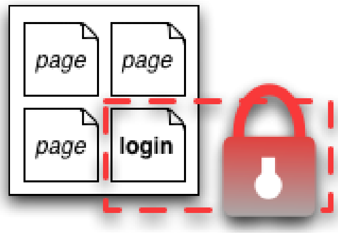
- **Whole application with "required" Shibboleth session**
  - Easiest way to protect a set of documents
  - No other authentication methods possible like this
  - Problems with lost HTTP POST requests



- **Whole application with "lazy" Shibboleth session**
  - Also allows for other authentication methods
  - Authorization can only be done in application



- **Only page that sets up application session**
  - Well-suited for dual login
  - Application can control session time-out
  - Generally the best solution



Options for Access Control

3 ways to protect an application with access rules:

- **Apache Access Rules (Apache only)**
  - Static configuration
    (Changes require restart of Apache)
  - Directory configuration (.htaccess) file
    (Restricted to existing directories in the filesystem)
- **Shibboleth XML Access Control (Apache, IIS, others)**
  - Configuration in shibboleth2.xml (or via .htaccess)
- **Application Access Control (Apache, IIS, others)**
  - Access control done by application itself based on attribute values

Options for Access Control: Overview

	1.a apache2.conf <VirtualHost>	1.b .htaccess	2. XML AccessControl *	3. Application Access Control
⊕	Easy to configure Can also protect locations or virtual files URL Regex	Dynamic Easy to configure	Platform independent Powerful boolean rules URL Regex Dynamic	Very flexible and powerful with arbitrarily complex rules URL Regex Support
⊖	*Only works for Apache* Not dynamic Very limited rules	*Only works for Apache* Only usable with "real" files and directories	XML editing Configuration error can prevent SP from restarting	You have to implement it yourself You have to maintain it yourself

* Configured in RequestMap or referenced by an .htaccess file

Apache Access Rules

Example:

```
# Force user to authenticate on protected-directory
<Location /protected-directory>
  AuthType shibboleth
  ShibCompatWith24 On
  ShibRequestSetting requireSession true
  Require shib-attr homeOrganizationType university uas
</Location>
```

- Enforces Shibboleth session for all resources at the path `/protected-directory`
- User must be member of a university or a university of applied sciences (*`university uas`*).

Notes for Apache 2.2

- The option `ShibCompatWith24 On` is recommended in case Apache 2.2 is used (to simplify a later migration).
- This option is provided by the Shibboleth SP Apache module. It adds support for extended "Require" rules that the Shibboleth SP supports in Apache 2.4.

In case you already use Apache 2.4, you need to remove the option `ShibCompatWith24 On`*.*

Apache: Static vs. Directory Configuration

- **Static configuration:**
  - Access rules are configured in main configuration.
    (e.g. `/etc/apache/sites-available/www.example.org`)
  - Changes require restart of Apache.
  - Applicable to "real" files and directories as well as to virtual files and locations

```
# Force user to authenticate on protected-directory
<Location /protected-directory>
  AuthType shibboleth
  ShibCompatWith24 On
  ShibRequestSetting requireSession true
  Require shib-attr homeOrganizationType university uas
</Location>
```

Apache: Static vs. Directory Configuration

- **Directory configuration:**
  - Access rules are configured in `.htaccess` files in the (filesystem) directories that need to be protected.
    (e.g. `/var/www/protected-directory/.htaccess`)
  - Changes take effect immediately.
  - Not applicable to virtual files and locations

Example:
`/var/www/protected-directory/.htaccess:`

```
# Force user to authenticate
AuthType shibboleth
ShibCompatWith24 On
ShibRequestSetting requireSession true
Require shib-attr homeOrganizationType university uas
```

Shibboleth XML Access Control

- Access rules are directly embedded in shibboleth2.xml file or included from external file.
- The Shibboleth SP dynamically loads access rules. Changes take effect immediately.
- If using Apache, XML access rules defined in an external file might be included in an .htaccess file.
  (Not discussed here; refer to the comprehensive documentation on our SWITCHaai website.)

Shibboleth XML Access Control: shibboleth2.xml

Proper place of XML access rules in shibboleth2.xml:

```
<SPConfig ...>
    [...]
    <RequestMapper type="Native">
        <RequestMap applicationId="default">
            <Host name="www.example.com">
                 [...]
            </Host>
            [...]
        </RequestMap>
    </RequestMapper>

    <ApplicationDefaults ...>
        [...]
    </ApplicationDefaults>
    [...]
</SPConfig>
```

Shibboleth XML Access Control: Example

```
...
<Host name="www.example.org">
  <Path name="protected-directory" authType="shibboleth" requireSession="true">
    <AccessControl>
      <AND>
        <Rule require="affiliation">student</Rule>
        <OR>
          <Rule require="homeOrganization">ethz.ch</Rule>
          <Rule require="homeOrganization">uzh.ch</Rule>
        </OR>
        <NOT>
          <!-- assert that VHO users are never allowed -->
          <Rule require="homeOrganization">vho-switchaai.ch</Rule>
        </NOT>
      </AND>
    </AccessControl>
    <Path name="unprotected" authType="shibboleth" requireSession="false" />
  </Path>
</Host>
...
```

Shibboleth XML Access Control: Example

Meaning:

- Affiliation MUST be "student"
- Home Organization MUST be either "ethz.ch" or "uzh.ch"
- Home Organization MUST NOT be "vho-switchaai.ch"
  (Although this last rule is always fulfilled because of the previous rules, this requirement is explicitly expressed, using a NOT operator.)

Shibboleth XML Access Control: Apache

- Using Apache, to support XML Access Rules embedded in shibboleth2.xml, you still need something similar to the following configuration (else, the rules won't take effect).

```
# Activate Shibboleth but don't enforce a session
<Location />
  AuthType shibboleth
  Require shibboleth
</Location>
```

Application Access Control

- Application can access and use Shibboleth attributes by reading them from the web server environment.
- The Shibboleth SP exports the attributes to a set of environment variables (Apache) or HTTP request headers (IIS)
- Attributes then can be used for access control.
- The names of the attributes may differ between various application containers (e.g. prefixed with "AJP_" if using Apache and Tomcat).

Application Access Control

- See the appropriate pages on the SWITCHaai website and on the Shibboleth Wiki for details:

  https://www.switch.ch/aai/support/serviceproviders/sp-access-rules.html
  https://wiki.shibboleth.net/confluence/display/SHIB2/NativeSPAttributeAccess

- Many applications, such as e-learning systems, have built-in support for Shibboleth (e.g. Moodle, Ilias).
  They don't need manual modifications.

Application Access Control: Example

PHP:

```
$affiliations = preg_split("/\s*;\s*/",
    $_SERVER['affiliation']);

if (in_array("staff", $affiliations)) {
  grantAccess();
}
```

（Affiliation: "staff;member"）

Pitfalls

- If you have run your Shibboleth SP for a long time, you may still use deprecated configuration directives. You may want to update them to simplify a later migration.

  Example:
  Old: `ShibRequireSession On`
  New: `ShibRequestSetting requireSession true`

  Consult the Shibboleth Wiki for details about configuration changes and to find deprecated directives:

  https://wiki.shibboleth.net/confluence/display/SHIB2/NativeSPApacheConfig

Pitfalls

- If you use Apache together with XML access rules, and if you have configured multiple hostnames in your virtual hosts in Apache, make sure that the option `UseCanonicalName` is set to `On` in Apache. Else, the XML access rules might be bypassed.

Further Information

- You can find detailed information about access control for SWITCHaai, including a lot of examples, on the following web page:
  - Shibboleth Service Provider Access Control
    https://www.switch.ch/aai/support/serviceproviders/sp-access-rules.html

- Comprehensive information and examples:
  Shibboleth Service Provider Training March 2014, "Hands-On":
  - https://www.switch.ch/aai/support/presentations/sp-training-2014/
    Shibboleth SP Training Hands-On, slides 75 to 104

Further Information

- General documentation from the Shibboleth Project:
  - Apache Configuration:
    https://wiki.shibboleth.net/confluence/display/SHIB2/NativeSPApacheConfig
  - Apache .htaccess:
    https://wiki.shibboleth.net/confluence/display/SHIB2/NativeSPhtaccess
  - XML-based mechanism:
    https://wiki.shibboleth.net/confluence/display/SHIB2/NativeSPXMLAccessControl