SD*Interceptor* for Apache 2 Web Server

Configuration Instructions

28 November 2016

This short guide illustrates how to enable the SD*Interceptor* for protecting a regular Web Server and OGC Web Services using a GeoXACML PDP.

SD*Interceptor* - Foreward

In order to enforce Access Rights, you will need a GeoPDP Service!

You can obtain your GeoPDP Service for a free evaluation. Please visit the GeoXACML homepage http://www.geoxacml.org select your location and register with your email address.

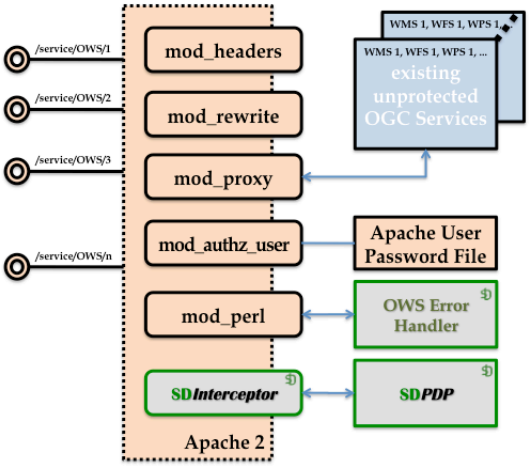
After registration, you will see the menu item „Evaluation“ which enables you to see your GeoPDP URL. Please use this GeoPDP URL for completing the SD*Interceptor* configuration.

The Evaluation section provides additional functionality for uploading and viewing a GeoXACML or XACML Policy. You can also send GeoXACML and XACML compliant Authorization Decision Requests and see the Authorization Decision.

SD*Interceptor* – Deployment options

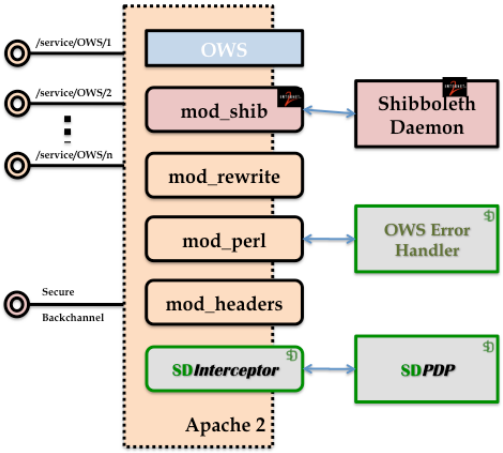
The **SD*Interceptor*** can be deployed as an Apache 2 Module to realize different setups. Some possible deployment variations are illustrated below. Please note that these deployments are just commonly used examples but that other deployment constellations are supported!

Example deployment #1



The Apache Web Server, configured as a Reverse Proxy to existing OGC Web Service(s), requires HTTP Authentication using a simple username/password file.

Example deployment #2



The Apache Web Server, hosting OGC Web Service(s), requires a SAML2 based authentication as supported by a Sibboleth IdP.

SD*Interceptor* - Activation via License Key

The **SD*Interceptor*** module requires a license key to run!

Three different license types are available:

1) Evaluation license
2) Web Server license
3) Web Server URI license

Please contact us at support@secure-dimensions.de and send us the following information for

- License 1) the value of the Apache 2 configuration entry for ServerName + end time of the evaluation period
- License 2) the value of the Apache 2 configuration entry for ServerName
- License 3) the value of the Apache 2 configuration entry for ServerName and Location

The resulting license key must be put inside the <Location> or <Directory> for the Apache 2 configuration.

In case you do not request an evaluation key, the **SD*Interceptor*** module will function for 24 hours only.

SD*Interceptor* - Activation and Configuration to use HTTP Attributes

In cases, where HTTP Header or Apache Environment attributes from the intercepted request shall be populated into the XACML Authorization Decision Request (ADR), sent to the GeoPDP, the following keys can be used. It is possible to repeat the entries in case more HTTP Header attributes shall be mapped into the same category.

This feature is switched off by default.

`SubjectAttribute` – allows adding HTTP Header attributes to the <Subject> section of the ADR
`ActionAttribute` - allows adding HTTP Header attributes to the <Action> section of the ADR
`ResourceAttribute` - allows adding HTTP Header attributes to the <Resource> section of the ADR
`EnvironmentAttribute` - allows adding HTTP Header attributes to the <Environment> section of the ADR

Each Key is followed by two string entries with the following pattern/meaning:

```
Key HTTP_Header_Attribute_Name XACML_Attribute_Name XACML_Type
```

Example:
For adding the HTTP Header Attribute **User-Agent** to the **<Environment>** section of the ADR as an XACML Attribute named **urn:example:client:user:agent** with data type **String**, the following entry provides that (please note that there is no line break in the configuration file!):

```
EnvironmentAttribute User-Agent urn:example:client:user:agent
http://www.w3.org/2001/XMLSchema#string

<Location /a/shibboleth/protected/service>
     AuthType shibboleth
     ShibRequireSession On
     ShibUseHeaders On
     Require valid-user
     SubjectAttribute affilitation urn:example:user:affiliation
                      http://www.w3.org/2001/XMLSchema#string
</Location>
```

SD*Interceptor* for Apache 2 Web Server – Configuration Instructions

The **SD*Interceptor*** for Apache 2 Web Server consists of one Apache 2 module, compiled as a Dynamic Shared Object (DSO): **`mod_authz_geoxacml.so`**

It can be loaded (as other Apache 2 DSO modules) using the LoadModule directive in the Apache 2 configuration file. Assuming the Apache 2 DSO modules directory is **`/usr/local/apache2/modules`**, and that the file **`mod_authz_geoxacml.so`** is in that directory, then the following example instruction loads the **SD*Interceptor*** for the Apache 2 Web Server (please note that there is no line break in the configuration file):

```
LoadModule authz_geoxacml_module
/usr/local/apache2/modules/mod_authz_geoxacml.so
```

The **SD*Interceptor*** module gets executed as Apache hook fixup MIDDLE. Please observe the order of execution with other modules due to loading sequence.

The **SD*Interceptor*** module requires the installation of the following libraries, which have different version numbers depending on your computer's operating system:

1) libcurl
2) libxml2
3) libxslt
4) gd
5) boost

Loading the Perl Error Handler

The following pre-requisites exist – please see related manuals how to perform the installation:

1. Perl must be installed
2. mod_perl must be installed
3. Apache must be configured for Perl support
4. GD support for Perl must be installed - Because the Error Handler for WMS must be capable to return images (in cases where the WMS parameter Exceptions is set to „in_image"), this is required.
5. Image format support - Depending on the image format support, you must also install support for png, jpg, gif, etc.
   Note: In case you do not support all the image formats that are supported by the WMS to be protected, please correct the supported images formats in the Capabilities document!

In order to enable the Error Handler, the file **OWSErrorHandler.pm** must be copied into a directory named **SD**, located in the Perl search path.
For example, if your Perl installation is in /usr/lib/perl5 (typically you find the directories named GD and ModPerl in there) you must create the directory **SD** and copy the OWSErrorHandler.pm:

```
mkdir -p /usr/lib/perl5/SD
cp OWSErrorHandler.pm /usr/lib/perl5/SD
```

Make sure that the file OWSErrorHandler.pm is executable. If necessary apply

```
chmod +x OWSErrorHandler.pm
```

Configuration of the Error Handler

The configuration can take place by defining a URL for the Error Handler, which can only be accessed from local host.

```
<Location /OWSErrorHandler>
  Order deny,allow
  Allow from localhost
  PerlOptions +GlobalRequest
  SetHandler perl-script
  PerlHandler SD::OWSErrorHandler
</Location>
```

Configuration of Proxy Settings

For all installations, where network communication between the **SD*Interceptor*** module and the **SD*PDP*** is via a web proxy, the following keys can be used for that configuration.

```
GeoPDPProxyIP <the IP address of the web proxy>
GeoPDPProxyPort <port number of the web proxy>
```

In case, theweb proxy requires HTTP 1.0 communication, the following key will enforce that.

```
GeoPDPProxyHTTPVersion 1.0
```

In case this key is missing, the HTTP version is 1.1.

Activation for protecting a Web Server

The configuration of this module takes place in the httpd.conf file (or the one applicable for the vhost). Inside a <Location> or <Direcotry> tag, the module can be activated by the following three statements:

```
<Location /my/protected/path>
  # enable the checking per GeoXACML PDP
  # on = access control is on
  # off = no access control; all requests are granted
  GeoXACML-PEP On|Off
  # The GeoPDP Serivce URL where the XACML 2 Authorization Decision
Requests are sent
  GeoPDP-URL <http://the URL to your GeoPDP Serivce>
  XACML-LicenseKey <your lience key goes here>
  XACML -LicenseEnd <for an evaluation license the expiration date>
...
</Location>
```

Please note that the term in front of "PEP" reflects the license type.

Activation for protecting a OGC Web Map Service

```
<Location /my/protected/WMS>
  # enable the checking per GeoXACML PDP
  # on = access control is on
  # off = no access control; all requests are granted
  WMS-PEP On|Off #Off by default
  # The GeoPDP Serivce URL where the XACML 2 Authorization Decision
Requests are sent
  GeoPDP-URL http://the URL to your GeoPDP Serivce
  WMS-LicenseKey <your lience key goes here>
  WMS-LicenseEnd <for an evaluation license the expiration date>
  ErrorDocument 403 /OWSErrorHandler
  ErrorDocument 500 /OWSErrorHandler
...
</Location>
```

Activation for protecting a OGC Web Feature Service (Transactional)

```
<Location /my/protected/WFS>
  # enable the checking per GeoXACML PDP
  # on = access control is on
  # off = no access control; all requests are granted
  WFS-PEP On|Off
  # The GeoPDP Serivce URL where the XACML 2 Authorization Decision
Requests are sent
  GeoPDP-URL <http://the URL to your GeoPDP Serivce>
  WFS-LicenseKey <your lience key goes here>
```

```
  WFS-LicenseEnd <for an evaluation license the expiration date>
  ErrorDocument 403 /OWSErrorHandler
  ErrorDocument 500 /OWSErrorHandler
...
</Location>
```

Activation for protecting a OGC Web Processing Service

```
<Location /my/protected/WPS>
  # enable the checking per GeoXACML PDP
  # on = access control is on
  # off = no access control; all requests are granted
  WPS-PEP On|Off
  # The GeoPDP Serivce URL where the XACML 2 Authorization Decision
Requests are sent
  GeoPDP-URL <http://the URL to your GeoPDP Serivce>
  WPS-LicenseKey <your lience key goes here>
  WPS-LicenseEnd <for an evaluation license the expiration date>
  ErrorDocument 403 /OWSErrorHandler
  ErrorDocument 500 /OWSErrorHandler
...
</Location>
```

Activation for protecting a OGC Web 3D Service

```
<Location /my/protected/WPS>
  # enable the checking per GeoXACML PDP
  # on = access control is on
  # off = no access control; all requests are granted
  W3DS-PEP On|Off
  # The GeoPDP Serivce URL where the XACML 2 Authorization Decision
Requests are sent
  GeoPDP-URL <http://the URL to your GeoPDP Serivce>
  W3DS-LicenseKey <your lience key goes here>
  W3DS-LicenseEnd <for an evaluation license the expiration date>
  ErrorDocument 403 /OWSErrorHandler
  ErrorDocument 500 /OWSErrorHandler
...
</Location>
```

Questions & Feedback

In case you have any additional questions or feedback, please contact us at support@secure-dimensions.com

SD*Interceptor* for Apache 2 Web Server – ADR Structure

After the **SD*Interceptor*** module is loaded and activated, it will generate XACML / GeoXACML based Authorization Decision Requests that are sent to the **SD*PDP***.

In order to write a GeoXACML policy that can use the information from the ADR to get an authorization decision, it is important to know the structure of the ADR and the attribute Ids used.

The following sections describe the mapping from the domain specific request to the ADR.

Common **Environment** Attributes (valid for all service types)

	ADR AttributeId	Data Type
current date	urn:oasis:names:tc:xacml:1.0:environment:current-date	Date
current-time	urn:oasis:names:tc:xacml:1.0:environment:current-time	Time
version	urn:oasis:names:tc:xacml:1.0:environment:current-dateTime	dateTime
resource-id	urn:oasis:names:tc:xacml:1.0:resource:resource-id	xpath-expression
hostname	urn:SD:def:xacml:2.0:hostname	String
protocol	urn:SD:def:xacml:2.0:protocol	String
uri	urn:SD:def:xacml:2.0:uri	String

Common **Subject** Attributes (valid for all service types)

	ADR AttributeId	Data Type
subject	urn:oasis:names:tc:xacml:1.0:subject:subject-id	String
role	urn:oasis:names:tc:xacml:2.0:subject:role	String
scope (MRP)	urn:oasis:names:tc:xacml:2.0:profile:multiple:scope	String

Common **Action** Attributes (valid for all service types)

	ADR AttributeId	Data Type
HTTP method	urn:SD:def:xacml:2.0:action	String
request	urn:SD:def:xacml:2.0:request	String

OGC Web Services Common parameters as **Resource** Attributes

request parameter	ADR AttributeId	Data Type
service	urn:SD:def:xacml:2.0:service	String
request	urn:SD:def:xacml:2.0:request	String
version	urn:SD:def:xacml:2.0:version	String
update_sequence	urn:SD:def:xacml:2.0:updatesequence	String

language	urn:SD:def:xacml:2.0:language	String

OGC Web Map Service parameters as **Resource** Attributes

request parameter	ADR AttributeId	Data ype
layers	urn:SD:def:xacml:2.0:layers	String
styles	urn:SD:def:xacml:2.0:styles	String
srs	urn:SD:def:xacml:2.0:srs	String
bbox	urn:SD:def:geoxacml:1.0:bbox	Geometry
width	urn:SD:def:xacml:2.0:width	Integer
height	urn:SD:def:xacml:2.0:height	Integer
format	urn:SD:def:xacml:2.0:format	String
transparent	urn:SD:def:xacml:2.0:transparent	String
bgcolor	urn:SD:def:xacml:2.0:bgcolor	String
exceptions	urn:SD:def:xacml:2.0:exceptions	String
wms_time	urn:SD:def:xacml:2.0:time	String
elevation	urn:SD:def:xacml:2.0:elevation	String
sld	urn:SD:def:xacml:2.0:sld	anyURI
wfs	urn:SD:def:xacml:2.0:wfs	anyURI
query_layers	urn:SD:def:xacml:2.0:querylayers	String
info_format	urn:SD:def:xacml:2.0:infoformat	String
feature_count	urn:SD:def:xacml:2.0:featurecount	Integer
x	urn:SD:def:xacml:2.0:x	Integer
y	urn:SD:def:xacml:2.0:y	Integer
info_point	urn:SD:def:geoxacml:1.0:infopoint	Geometry

OGC Web Map Tile Service parameters as **Resource** Attributes

request parameter	ADR AttributeId	Data ype
format	urn:SD:dev:xacml:2.0:format	String
tilematrixset	urn:SD:dev:xacml:2.0:tilematrixset	String
tilematrix	urn:SD:dev:xacml:2.0:tilematrix	String
layer	urn:SD:dev:xacml:2.0:layers	String
style	urn:SD:dev:xacml:2.0:styles	String
tilerow	urn:SD:dev:xacml:2.0:tilerow	Integer
tilecol	urn:SD:dev:xacml:2.0:tilecol	Integer
elevation	urn:SD:dev:xacml:2.0:elevation	Double
othersampledimensions	urn:SD:dev:xacml:2.0:othersampledimensions	String
infoformat	urn:SD:dev:xacml:2.0:inforformat	String
i	urn:SD:dev:xacml:2.0:i	Integer
j	urn:SD:dev:xacml:2.0:j	Integer

OGC Web Feature Service Transactional parameters as **Resource** Attributes

request parameter	ADR AttributeId	Data ype
typename(s)	urn:SD:def:xacml:2.0:typename	String
outputformat	urn:SD:def:xacml:2.0:outputformat	String
valuereference	urn:SD:def:xacml:2.0:valuereference	String
startindex	urn:SD:def:xacml:2.0:startindex	String
count	urn:SD:def:xacml:2.0:count	String
resulttype	urn:SD:def:xacml:2.0:resulttype	String
resolve	urn:SD:def:xacml:2.0:resolve	String
resolvedepth	urn:SD:def:xacml:2.0:resolvedepth	String
resolvetimeout	urn:SD:def:xacml:2.0:resolvetimeout	String
resolvepath	urn:SD:def:xacml:2.0:resolvepath	String
namespaces	urn:SD:def:xacml:2.0:namespaces	String
aliases	urn:SD:def:xacml:2.0:aliases	String
filter	urn:SD:def:xacml:2.0:filter	String
srsname	urn:SD:def:xacml:2.0:srsname	String
filter_language	urn:SD:def:xacml:2.0:filter_language	String
sortby	urn:SD:def:xacml:2.0:sortby	String
propertyname	urn:SD:def:xacml:2.0:propertyname	String
storedquery_id	urn:SD:def:xacml:2.0:storedquery_id	String
lockid	urn:SD:def:xacml:2.0:lockid	String
expiry	urn:SD:def:xacml:2.0:expiry	String
lockaction	urn:SD:def:xacml:2.0:lockaction	String
releaseaction	urn:SD:def:xacml:2.0:releaseaction	String
featureid	urn:SD:def:xacml:2.0:featureid	String
traversexlinkdepth	urn:SD:def:xacml:2.0:traversexlinkdepth	String
traversexlinkexipry	urn:SD:def:xacml:2.0:traversexlinkexipry	String
proptravxlinkdepth	urn:SD:def:xacml:2.0:proptravxlinkdepth	String
proptravxlinkexpiry	urn:SD:def:xacml:2.0:proptravxlinkexpiry	String
gmlobjectid	urn:SD:def:xacml:2.0:gmlobjectid	String

OGC Web Processing Service parameters as **Resource** Attributes

request parameter	ADR AttributeId	Data ype
input	urn:SD:def:xacml:2.0:input	anyURI
process	urn:SD:def:xacml:2.0:process	String
datainputs	urn:SD:def:xacml:2.0:datainputs	String
responseform	urn:SD:def:xacml:2.0:responseform	String

OGC Catalog Service for the Web parameters as **Resource** Attributes

request parameter	ADR AttributeId	Data ype

namespace	urn:SD:def:xacml:2.0:namespace	String
typename	urn:SD:def:xacml:2.0:typename	String
schemalanguage	urn:SD:def:xacml:2.0:schemalanguage	String
parametername	urn:SD:def:xacml:2.0:parametername	String
requestid	urn:SD:def:xacml:2.0:requestid	String
outputschema	urn:SD:def:xacml:2.0:outputschema	String
startposition	urn:SD:def:xacml:2.0:startposition	String
maxrecords	urn:SD:def:xacml:2.0:maxrecords	String
elementsetname	urn:SD:def:xacml:2.0:elementsetname	String
elementname	urn:SD:def:xacml:2.0:elementname	String
constraintlanguage	urn:SD:def:xacml:2.0:constraintlanguage	String
constraint	urn:SD:def:xacml:2.0:constraint	String
distributedsearch	urn:SD:def:xacml:2.0:distributedsearch	Boolean
hopcount	urn:SD:def:xacml:2.0:hopcount	Boolean
responsehandler	urn:SD:def:xacml:2.0:responsehandler	String
recordid	urn:SD:def:xacml:2.0:recordid	String
source	urn:SD:def:xacml:2.0:source	String
resourcetype	urn:SD:def:xacml:2.0:resourcetype	String
resourceformat	urn:SD:def:xacml:2.0:resourceformat	String
harvestinterval	urn:SD:def:xacml:2.0:harvestinterval	String

OGC Web 3D Service parameters as Resource Attributes

request parameter	ADR AttributeId	Data ype
acceptversions	urn:SD:def:xacml:2.0:acceptversions	String
acceptformats	urn:SD:def:xacml:2.0:acceptformats	String
sections	urn:SD:def:xacml:2.0:sections	String
minheight	urn:SD:def:xacml:2.0:minheight	Double
maxheight	urn:SD:def:xacml:2.0:maxheight	Double
spatialselection	urn:SD:def:xacml:2.0:spatialselection	String
lods	urn:SD:def:xacml:2.0:lods	String
lodselection	urn:SD:def:xacml:2.0:lodselection	String
offset	urn:SD:def:xacml:2.0:offset	Geometry
background	urn:SD:def:xacml:2.0:background	String
light	urn:SD:def:xacml:2.0:light	Boolean
viewpoints	urn:SD:def:xacml:2.0:viewpoints	String
coordinate	urn:SD:def:xacml:2.0:coordinate	Geometry
columnames	urn:SD:def:xacml:2.0:columnames	String
tilelevel	urn:SD:def:xacml:2.0:tilelevel	Integer
tilerow	urn:SD:def:xacml:2.0:tilerow	Integer
tilecolumn	urn:SD:def:xacml:2.0:tilecolumn	Integer