SAML

MyTimetable supports SAMLv2 authentication, using Spring Security SAML and OpenSAML as underlying libraries. The following information is necessary to set up SAML authentication:

- Identity Provider (IdP) metadata URL
- IdP entity ID (consult the metadata for the available entity IDs)
- Java keystore containing a public/private key pair (we generate it below)
- Attribute name to use as username and display name (optionally)

First generate a key pair using Java keytool. We recommend to use a 2048 bit (or higher) RSA key. Depending on the policy of your IdP a self-signed key could suffice – most IdP's require HTTPS connections for both the IdP and SP's, which makes the certificate chain of the SAML key less important. Please use the same password for the keystore and the key.

```
root@d-xx-tc-01:/opt/tomcat/conf# /opt/jdk1.7.0_45/jre/bin/keytool -genkey -alias samlKey -keyalg RSA -keysize
2048 -validity 3650 -keystore saml.jks
Enter keystore password:
Re-enter new password:
What is your first and last name?
  [Unknown]:  xx.dev.eveoh.nl
What is the name of your organizational unit?
  [Unknown]:  Eveoh
What is the name of your organization?
  [Unknown]:  Eveoh
What is the name of your City or Locality?
  [Unknown]:  The Hague
What is the name of your State or Province?
  [Unknown]:  Zuid-Holland
What is the two-letter country code for this unit?
  [Unknown]:  NL
Is CN=xx.dev.eveoh.nl, OU=Eveoh, O=Eveoh, L=Den Haag, ST=Zuid-Holland, C=NL correct?
  [no]:  yes
Enter key password for <samlKey>
        (RETURN if same as keystore password):
```

Now you can configure MyTimetable to use SAML authentication. This is done in the MyTimetable properties file, usually located in `$tomcat/mytimetable/config`. We define the `spring.profiles.active` parameter to activate saml, and set various properties:

```
# use auth-saml,ec when using Exchange/O365/GCal push-sync
spring.profiles.active = auth-saml

# SAML keystore information
saml.keystore = file:/location/to/keystore.jks
saml.keyname = samlKey
saml.keypass = keypass

# Our own entity ID and URL
# You can use any entity ID, we usually set it to be the same as the URL
saml.entity_id = https://our.entity.id
saml.entity_baseurl = https://our.entity.base_url

# IDP metadata URL and entity ID
saml.idp_url = https://idp/metadata/url
saml.idp_entity_id = https://idp.entity.id

# Attribute containing the username to use - value has to be unique since it is used in our data store for
storing user data
# Specify @null to use the NameID
# This username is also used when connecting to external systems (e.g., connecting to Blackboard to retrieve
timetables)
# Default to the eduPersonPrincipalName
saml.attribute.username = urn:mace:dir:attribute-def:eduPersonPrincipalName

# OPTIONAL: attribute to use as display name in the user interface, if not specified the username will be shown
#saml.attribute.displayName = urn:mace:dir:attribute-def:eduPersonPrincipalName

# OPTIONAL: maximum age of the authentication tocal
#saml.max_authentication_age = 43200

# OPTIONAL: set to true to force asking the user for username/password (disable SSO)
#saml.forceauthn = false

# OPTIONAL: set to false to support and initiate SAML Single Logout (SLO)
#saml.local_logout_only = true

# OPTIONAL: Signature to use for SAML metadata and signatures
# Possible values: sha1, sha256, sha384, sha512 (default = sha256, recommended = sha256)
#saml.signature_algorithm = sha256

# OPTIONAL: set to true to redirect the user to your own logout page, and specify the URL of the logout page
#LogoutUrl.AlwaysUseTarget = false
#LogoutUrl.Target = /
```

Please note that our SAML library checks the URL's specified in the incoming SAML message. Because of this, the URL in the SAML message **must** match with the actual URL Tomcat is providing to the Servlet. If you are using a proxy in front of MyTimetable, you will need to specify the scheme, proxyHost and proxyPort properties in the Connector element of your server.xml:

```
    <Connector port="8080" protocol="HTTP/1.1"
               connectionTimeout="20000"
               scheme="https" proxyName="xx.dev.eveoh.nl" proxyPort="443" /> <!-- This last line specifies the
actual protocol/host/port the end-user is using -->
```

ADFS 2/3 specifics

Using the SAML authentication module it is possible to authenticate using ADFS 2 or 3. There are a couple of things to keep in mind though:

- For ADFS 2: Install all update rollups and hotfixes available for ADFS. See the links mentioned in this Stackoverflow answer for more information.
- You can normally find the ADFS metadata at the URL: `https://<adfsserver>/FederationMetadata/2007-06/FederationMetadata.xml`
- Make sure you send a NameID: add a claim rule to send an LDAP attribute, with the NameID as outgoing type. You can usually send the samAccountName as NameID.
- If you get a '401' or '403' error after being redirected to MyTimetable, check the logs at your ADFS IdP for more information. If these logs are not showing any error, enable debug logging in MyTimetable and check the MyTimetable logs for errors.

For older MyTimetable versions (< 3.7) or older JRE's without the JCE policy files:

- Disable encryption of the SAML assertion, using the Powershell command:

  ```
  Set-ADFSRelyingPartyTrust -TargetName “target” -EncryptClaims $False
  ```

  By default the JVM does not support the large key sizes in use by ADFS, and in almost all cases your assertions/claims won't contain any confidential data. If you still want to have the assertion encrypted, try installing the Java Cryptography Extension poilcy available at the bottom of this page. If that doesn't work, please contact us for more information, we will probably have to patch your MyTimetable installation with the Bouncy Castle Crypto APIs.
- Set the 'secure hash algorithm', found under the Advanced tab of the Relying Party properties, to SHA-1 instead of SHA-256.