Shibboleth Identity Provider 3

Εγκατάσταση και παραμετροποίηση

# Γενικά

Για την ένταξη ενός Φορέα στην Ομοσπονδία ΔΗΛΟΣ του ΕΔΕΤ απαιτείται η υλοποίηση ενός Παρόχου Ταυτότητας συμβατού με τα πρότυπα SAML 2.0 του οργανισμού OASIS. Προτείνεται ο [Shibboleth Identity Provider](https://shibboleth.net/products/identity-provider.html). Απαιτούμενη για τη λειτουργία του Shibboleth Identity Provider είναι η ύπαρξη του ακόλουθου λογισμικού

* Java Runtime Environment (jre)
* Java Servlet Container ( π.χ. Tomcat, Jetty)

Ο Shibboleth Identity Provider εκμεταλλευόμενος τις cross-platform δυνατότητες της Java μπορεί να εγκατασταθεί τόσο σε λειτουργικό σύστημα Microsoft Windows όσο και σε λειτουργικό σύστημα Linux. Πρόταση της Ομοσπονδίας ΔΗΛΟΣ του ΕΔΕΤ είναι ή χρήση λειτουργικού συστήματος Linux, και οι ακόλουθες οδηγίες είναι βασισμένες σε λειτουργικό σύστημα Debian 8.2

# Πριν ξεκινήσετε

## EntityID

Το entityID στο SAML είναι μια συμβολοσειρά που χαρακτηρίζει μοναδικά ένα entity (είτε Identity Provider είτε Service Provider). Συμφωνα με το [SAML 2.0 Core specification](https://www.oasis-open.org/committees/download.php/35711/sstc-saml-core-errata-2.0-wd-06-diff.pdf) θα πρέπει να είναι ένα URI του οποίου το μήκος να μην ξεπερναά τους 1024 χαρακτήρες. Το entityID ενός Identity Provider δεν θα πρέπει να αλλάξει ποτέ και να είναι προφανές πως συνδέεται με τον Φορέα που διαχειρίζεται τον Identity Provider.

Προτείνεται να διαλέξετε ένα entityID με τα ακόλουθα στοιχεία

* Να είναι ένα σωστό URI
* Να είναι ένα absolute URL
* Να περιέχει το πρωτεύον DNS domain του φορέα σας.
* Να περιέχει τη συμβολοσειρά 'idp'

# Εγκατάσταση Shibboleth Identity Provider v3

## Προετοιμασία Λειτουργικού Συστήματος

## Ενημέρωση και αναβαθμίσεις πακέτων

apt-get update && apt-get upgrade
apt-get install curl vim

## Ενεργοποίηση jessie-backports για την εγκατάσταση Java 8

echo deb http://http.debian.net/debian jessie-backports main >> /etc/apt/sources.list
apt-get update
apt-get install openjdk-8-jre

## Εγκατάσταση Tomcat 8 Servlet Container

apt-get install tomcat8

## Eγκατάσταση Apache http server και ενεργοποίηση modules

Εγκαταστήστε τον apache http server

apt-get install apache2

Ενεργοποιήστε τα απαραίτητα modules *proxy, proxy\_ajp* και *ssl*

a2enmod proxy
a2enmod proxy\_ajp
a2enmod ssl

Επανεκκινήστε τον apache http server

service apache2 restart

## Έλεγχος ώρας συστήματος

Η σωστή ώρα συστήματος είναι πολύ σημαντική στο SAML καθώς τα SAML μηνύματα έχουν περιορισμένη χρονική ισχύ (συνήθως της τάξης των μερικών λεπτών) και κατά συνέπεια ο λάθος χρονισμός ενός συστήματος ή η λάθος επιλογή ζώνης ώρας μπορεί να προκαλέσει σημαντικά προβλήματα στην επικοινωνία του Shibboleth Identity Provider με Service Providers που συμμετέχουν στην Ομοσπονδία ΔΗΛΟΣ του ΕΔΕΤ, ή το eduGAIN.

Εγκαταστήστε το πακέτο ntp με την ακόλουθη εντολή

aptitude install ntp

και επιβεβαιώστε ότι συνδέεστε σε κάποιους NTP servers με

ntpq -p

και ότι η ώρα συστήματος σας είναι η σωστή με

date -R

## Εγκατάσταση Βάσης Δεδομένων

apt-get install mysql-server libmysql-java

## Παραμετροποίηση Apache

### Virtual Host

Μπορείτε να επεξεργαστείτε το αρχείο */etc/apache2/sites-available/default-ssl* ή να δημιουργήσετε ένα καινούργιο. Παραθέτουμε παρακάτω ένα ελάχιστο παράδειγμα που προσφέρει την αναγκαία λειτουργικότητα.

<VirtualHost \_default\_:443>
 ServerSignature On
 SSLEngine on
 SSLProtocol TLSv1 +TLSv1.1 +TLSv1.2
 SSLHonorCipherOrder On
 SSLCipherSuite ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES128-SHA256:DHE-RSA-AES256-SHA256:DHE-RSA-AES128-SHA256:ECDHE-RSA-AES256-SHA:ECDHE-RSA-AES128-SHA:DHE-RSA-AES256-SHA:DHE-RSA-AES128-SHA:AES256-GCM-SHA384:AES128-GCM-SHA256:AES256-SHA256:AES128-SHA256
 SSLCertificateFile /path/to/your/TLS/certificate
 SSLCertificateKeyFile /path/to/your/key
 SSLCertificateChainFile /path/to/your/TLS/certificate/chain
 SSLOptions +StdEnvVars +ExportCertData
 SSLVerifyClient none

 BrowserMatch ".\*MSIE.\*" \
 nokeepalive ssl-unclean-shutdown \
 downgrade-1.0 force-response-1.0

 ProxyPass /idp ajp://localhost:8009/idp retry=5
 <Proxy ajp://localhost:8009>
 Require all granted
 </Proxy>
 <Location /idp>
 Order allow,deny
 Allow from all
 </Location>

 LogLevel warn
 ErrorLog ${APACHE\_LOG\_DIR}/idp-error.log
 CustomLog ${APACHE\_LOG\_DIR}/idp-access.log combined
</VirtualHost>

Δοκιμάστε την παραμετροποίηση του Apache httpd με την παρακάτω εντολή για να σιγουρευτείτε ότι δεν υπάρχει κάποιο πρόβλημα:

apache2ctl configtest

Ενεργοποιήστε τώρα το νέο site με:

a2ensite default-ssl

## Παραμετροποίηση Tomcat

### Απενεργοποίηση Connectors

Εφόσον χρησιμοποιείται ο Apache httpd σαν web server μπροστά από τον Apache Tomcat, ο Tomcat δεν χρειάζεται να "ακούει" σε καμία άλλη πόρτα εκτός από εκείνη που απαιτείται από το Apache JServ Protocol (AJP) Χρησιμοποιώντας τον editor της αρεσκείας σας, επεξεργαστείτε το αρχείο */etc/tomcat8/server.xml*. Στο κομμάτι της παραμετροποίησης όπου ορίζονται τα <Connector> elements, φροντίστε ώστε μόνο το ακόλουθο να είναι ενεργοποιημένο, ενώ όλα τα υπόλοιπα <Connector> elements είναι σε σχόλια ή διαγραμμένα.

<Connector port="8009" protocol="AJP/1.3" maxPostSize="100000"/>

### JVM Options

O Tomcat8 χρησιμοποιεί την υλοποίηση της SecureRandom class για τις τυχαίες τιμές που χρειάζεται ( session ids κ.α.). Υπάρχει πιθανότητα κατά την εκκίνηση/επανεκκίνηση του Tomcat να υπάρξουν σημαντικές καθυστερήσεις εάν χρησιμοποιηθεί το JRE σαν πηγή εντροπίας. Για τον λόγο αυτό προτείνεται να χρησιμοποιηθεί το /dev/urandom του συστήματος σας. Ακόμα συνιστούμε την ενεργοποίηση του alternate garbage collector του JVM με την παράμετρο -XX:+UseG1GC για καλύτερη απόδοση με μεγάλα αρχεία metadata. Τέλος αλλάξτε το μέγιστο μέγεθος μνήμης που μπορεί να χρησιμοποιήσει ο Tomcat σε τουλάχιστον 1,5GB. Χρησιμοποιώντας τον editor της αρεσκείας σας, επεξεργαστείτε το αρχείο */etc/default/tomcat8* και αντικαταστήστε την γραμμή που ξεκινά με **JAVA\_OPTS** με την ακόλουθη:

JAVA\_OPTS="-Djava.awt.headless=true -Xmx1500m -XX:+UseG1GC -Djava.security.egd=file:/dev/./urandom"

### Miscellaneous Option

**Session Persistence**

Απενεργοποιήστε το session persistence του Tomcat για να ελευθερώσετε τα log files σας από errors τύπου *lack of persistence of the session objects* όταν κάνετε shutdown. Χρησιμοποιώντας τον editor της αρεσκείας σας, επεξεργαστείτε το αρχείο */etc/tomcat8/context.xml* και ενεργοποιήστε (αφαιρέστε τα σχόλια από) το ακόλουθο element:

<Manager pathname="" />

**Eπιτρέψτε στον Tomcat να χρησιμοποιήσει τον mysql-connector**

Δημιουργείστε ένα symbolic link για το mysql-connector-java.jar στο classpath του Tomcat 8

ln -s /usr/share/java/mysql-connector-java.jar /usr/share/tomcat8/lib/

Επανεκκινήστε τον Tomcat 8 για να ενεργοποιηθούν οι αλλαγές στην παραμετροποίηση:

service tomcat8 restart

## Παραμετροποίηση Βάσης

Συνδεθείτε στην βάση σαν χρήστης με τα κατάλληλα δικαιώματα, και στη συνέχεια

* Δημιουργείστε το instance της βάσης που θα χρησιμοποιήσει ο Shibboleth Identity Provider
* mysql> CREATE DATABASE shibboleth;
* Δημιουργείστε ένα χρήστη με τα κατάλληλα credentials που θα χρησιμοποιεί ο Shibboleth Identity Provider ώστε να διαβάζει και να γράφει στην βάση. Αντικαταστήστε το *password* με έναν ισχυρό κωδικό της επιλογής σας.
* mysql> CREATE USER 'shibboleth'@'localhost' IDENTIFIED BY 'password';
* Δώστε στον χρήστη που δημιουργήσατε τα κατάλληλα privileges
* mysql> GRANT ALL PRIVILEGES ON shibboleth . \* TO 'shibboleth'@'localhost';
mysql> FLUSH PRIVILEGES;
* Δημιουργείστε τον πίνακα που θα χρησιμοποιηθεί για το persistent SAML2 NameID, όπως ορίζεται παρακάτω
* mysql> use shibboleth
mysql> CREATE TABLE shibpid ( \
 localEntity VARCHAR(255) NOT NULL, \
 peerEntity VARCHAR(255) NOT NULL, \
 persistentId VARCHAR(50) NOT NULL, \
 principalName VARCHAR(50) NOT NULL, \
 localId VARCHAR(50) NOT NULL, \
 peerProvidedId VARCHAR(50) NULL, \
 creationDate TIMESTAMP NOT NULL, \
 deactivationDate TIMESTAMP NULL, \
 PRIMARY KEY (localEntity, peerEntity, persistentId) \
);
* Δημιουργείστε τον πίνακα που θα χρησιμοποιηθεί για το server side storage του Shibboleth Identity Provider, όπως ορίζεται παρακάτω:
* mysql> use shibboleth
mysql> CREATE TABLE `StorageRecords` (
 `context` varchar(255) NOT NULL,
 `id` varchar(255) NOT NULL,
 `expires` bigint(20) DEFAULT NULL,
 `value` longtext NOT NULL,
 `version` bigint(20) NOT NULL,
 PRIMARY KEY (`context`,`id`)
);

## Παραμετροποίηση openLDAP

Θα χρειαστεί να ρυθμίσετε τον openLDAP server σας ώστε να γνωρίζει τα 2 schemas που χρησιμοποιούνται στην Ομοσπονδία ΔΗΛΟΣ του ΕΔΕΤ και στο eduGAIN, το eduPerson και το schac.

Κατεβάστε το [schac schema από εδώ](https://wiki.refeds.org/download/attachments/1606048/schac-20150413-1.5.0.schema.txt?version=1&modificationDate=1429044813839&api=v2) και το [eduPerson αντίστοιχα από εδώ](https://spaces.internet2.edu/download/attachments/2309/2xOpenLdapEduPerson-201602.zip?version=1&modificationDate=1469471690870&api=v2)

Για το schac:

* Σώστε το αρχείο ως *schac.schema*
* Δημιουργείστε το directory /tmp/schac

mkdir /tmp/schac

* Δημιουργείστε ένα dummy αρχείο με το όνομα slapd.conf με περιεχόμενα

include /tmp/schac/schac.schema

* Μέσα από το directory /tmp/schema εκτελέστε την ακόλουθη εντολή
* slaptest -f slapd.conf -F .
* που θα δημιουργήσει το αρχείο /tmp/schac/cn\=config/cn\=schema/cn\=\{0\}schac.ldif
* Επεξεργαστείτε το αρχείο ώστε να:
	+ Αφαιρεθούν όλες οι γραμμές που *δεν* ξεκινούν με
	+ dn:
	objectClass:
	cn:
	olcAttributeTypes:
	olcObjectClasses:
	+ Αλλαχθεί το dn: σε dn: cn=schac,cn=schema,cn=config
	+ Αλλάχθεί το cn: σε cn: schac
* Μετονομάστε το αρχείο σε schac.ldif
* mv /tmp/schac/cn\=config/cn\=schema/cn\=\{0\}schac.ldif /tmp/schac/schac.ldif
* Προσθέστε το schema με
* ldapadd -Q -Y EXTERNAL -H ldapi:/// -f /tmp/schac/schac.ldif

Και αντίστοιχα για το eduPerson schema:

* Σώστε το αρχείο ως *eduperson.schema*
* Δημιουργείστε το directory /tmp/eduperson

mkdir /tmp/eduperson

* Δημιουργείστε ένα dummy αρχείο με το όνομα slapd.conf με περιεχόμενα

include /tmp/eduperson/edueprson.schema

* Μέσα από το directory /tmp/eduperson εκτελέστε την ακόλουθη εντολή

slaptest -f slapd.conf -F .

που θα δημιουργήσει το αρχείο `/tmp/eduperson/cn\=config/cn\=schema/cn\=\{0\}eduperson.ldif`

* Επεξεργαστείτε το αρχείο ώστε να:
	+ Αφαιρεθούν όλες οι γραμμές που *δεν* ξεκινούν με
	+ dn:
	objectClass:
	cn:
	olcAttributeTypes:
	olcObjectClasses:
	+ Αλλαχτεί το dn: σε dn: cn=eduperson,cn=eduperson,cn=config
	+ Αλλαχτεί το cn: σε cn: eduperson
* Μετονομάστε το σε eduperson.ldif
* mv /tmp/eduperson/cn\=config/cn\=schema/cn\=\{0\}eduperson.ldif /tmp/eduperson/eduperson.ldif
* Προσθέστε το schema στον openLDAP με
* ldapadd -Q -Y EXTERNAL -H ldapi:/// -f /tmp/eduperson/eduperson.ldif

## Εγκατάσταση Shibboleth Identity Provider

### Δημιουργία του Shibboleth idp.war

Κατεβάστε το αρχείο εγκατάστασης καθώς και το αρχείο με το sha256 hash του αρχείου εγκατάστασης από το <https://shibboleth.net/downloads/identity-provider/latest/>

Για παράδειγμα, για την έκδοση 3.3.0

wget https://shibboleth.net/downloads/identity-provider/latest/shibboleth-identity-provider-3.3.0.tar.gz

wget https://shibboleth.net/downloads/identity-provider/latest/shibboleth-identity-provider-3.3.0.tar.gz.sha256

Υπολογίστε το sha256 hash του αρχείου εγκατάστασης και συγκρίνετε το με την τιμή που βρίσκεται στο shibboleth-identity-provider-3.3.0.tar.gz.sha256

sha256sum shibboleth-identity-provider-3.3.0.tar.gz

Αποσυμπιέστε το αρχείο εγκατάστασης

tar xvfz shibboleth-identity-provider-3.3.0.tar.gz

Εκτελέστε το installation script με

cd shibboleth-identity-provider-3.3.0
JAVA\_HOME=/usr/lib/jvm/java-8-openjdk-amd64/jre/ bin/install.sh

Κατά την διαδικασία της εγκατάστασης θα ερωτηθείτε για τα ακόλουθα:

* *Installation Directory*: Αν δεν υπάρχει κάποιος πολύ σημαντικός λόγος να αλλάξετε το default path της εγκατάστασης, αφήστε το στο */opt/shibboleth-idp*
* *Hostname*: Το FQDN του host
* *SAML EntityID*: Το SAML EntityID του Identity Provider .
* *Attribute Scope*: To Scope επηρεάζει τα λεγόμενα scoped attributes, τα οποία είναι τα ακόλουθα
	+ eduPersonScopedAffiliation
	+ eduPersonPrincipalName
	+ eduPersonUniqueId
* και έχουν σύνταξη της μορφής \*\*value@scope\*\*
* Το scope του Identity Provider δηλώνεται στα Metadata του και η τιμή του συνήθως συμπίπτει με το primary domain του φορέα.
* *Backchannel PKCS12 Password*: Ο κωδικός με τον οποίον θα προστατευτεί το keystore στο οποίο αποθηκεύεται το TLS certificate που θα χρησιμοποιήσει ο Shibboleth Identity Provider για backchannel επικοινωνία.
* *Cookie Encryption Key Password*: Ο Identity Provider θα δημιουργήσει ένα κλειδί συμμετρικής κρυπτογράφησης AES το οποίο θα χρησιμοποιεί για την κρυπτογράφηση των cookies και κάποιων επιπλέον δεδομένων για εσωτερική χρήση. Το κλειδί αυτό αποθηκεύεται σε ενα Java Keystore File (JCEKS). Ο κωδικός που θα δώσετε εδώ θα χρησιμοποιηθεί για την προστασία του κλειδιού στο keystore (είναι ο κωδικός με τον οποίον προστατεύεται η πρόσβαση τόσο στο κλειδί το ίδιο όσο και στο keystore).

Παρακάτω παρουσιάζεται το output από ένα session εγκατάστασης:

Source (Distribution) Directory: [/root/shibboleth-identity-provider-3.3.0]
Installation Directory: [/opt/shibboleth-idp]
Hostname: [idp.grnet.gr]

SAML EntityID: [https://idp.grnet.gr/idp/shibboleth]

Attribute Scope: [vm.okeanos.grnet.gr]
grnet.gr
Backchannel PKCS12 Password:
Re-enter password:
Cookie Encryption Key Password:
Re-enter password:
Warning: /opt/shibboleth-idp/bin does not exist.
Warning: /opt/shibboleth-idp/dist does not exist.
Warning: /opt/shibboleth-idp/doc does not exist.
Warning: /opt/shibboleth-idp/system does not exist.
Warning: /opt/shibboleth-idp/webapp does not exist.
Generating Signing Key, CN = idp.grnet.gr URI = https://idp.grnet.gr/idp/shibboleth ...
...done
Creating Encryption Key, CN = idp.grnet.gr URI = https://idp.grnet.gr/idp/shibboleth ...
...done
Creating Backchannel keystore, CN = idp.grnet.gr URI = https://idp.grnet.gr/idp/shibboleth ...
...done
Creating cookie encryption key files...
...done
Rebuilding /opt/shibboleth-idp/war/idp.war ...
...done

BUILD SUCCESSFUL
Total time: 3 minutes 16 seconds

Μετά την ολοκλήρωση της διαδικασίας εγκατάστασης, θα έχει δημιουργηθεί το idp.war αρχείο που περιέχει το Java web application που υλοποιεί τον Shibboleth Identity Provider.

Το war έχει δημιουργηθεί στο /opt/shibboleth-idp/war/idp.war

### Εγκατάσταση βιβλιοθηκών

Προαπαιτούμενο της παραμετροποιήσης του server side storage για τον Shibboleth Identity Provider είναι και η χρησιμοποίηση μιας βιβλιοθήκης που υποστηρίζει connection pooling. Προτείνεται το HikariCP

Κατεβάστε την τελευταία έκδοση του HikariCP για Java8 (2.5.1 τη στιγμή συγγραφής του οδηγού)

curl -o '/opt/shibboleth-idp/edit-webapp/WEB-INF/lib/HikariCP-2.5.1.jar' 'https://search.maven.org/remotecontent?filepath=com/zaxxer/HikariCP/2.5.1/HikariCP-2.5.1.jar'

Το Shibboleth ΙdP χρειάζεται επίσης την βιβλιοθήκη jstl1.2, η οποία όμως δεν περιέχεται στην αρχική εγκατάσταση. Για τον λόγο αυτό, κατεβάστε το jstl1.2.jar και αποθηκεύστε το τοπικά.

curl -o '/opt/shibboleth-idp/edit-webapp/WEB-INF/lib/jstl1.2' 'https://build.shibboleth.net/nexus/service/local/repositories/thirdparty/content/javax/servlet/jstl/1.2/jstl-1.2.jar'

στο */opt/shibboleth-idp-3.3.0/edit-webapp/WEB-INF/classes/META-INF/* Στη συνέχεια, κάντε build ξανά το idp.war για να περιέχει τα JAR που κατεβάσατε.

JAVA\_HOME=/usr/lib/jvm/java-8-openjdk-amd64/jre/ /opt/shibboleth-idp/bin/build.sh
Installation Directory: [/opt/shibboleth-idp]

Rebuilding /opt/shibboleth-idp/war/idp.war ...
...done

BUILD SUCCESSFUL
Total time: 10 seconds

### Παραμετροποίηση Permissions

Ο χρήστης ως ο οποίος τρέχουν τα processes του Tomcat8 θα πρέπει να έχει δικαίωμα να διαβάσει τα αρχεία που περιέχονται στο */opt/shibboleth-idp/conf/* και στο */opt/shibboleth-idp/credentials/* και να μπορεί να διαβάζει και να γράφει στα directories */opt/shibboleth-idp/logs/* και */opt/shibboleth-idp/metadata/*

Η πρόταση της Δ.Ο. της Ομοσπονδίας ΔΗΛΟΣ του ΕΔΕΤ είναι να γίνει group owner των παραπάνω directories το tomcat8 group και να αποδοθούν τα κατάλληλα permissions στο group αυτό όπως περιγράφεται παρακάτω:

chmod g+r -R /opt/shibboleth-idp/conf/
chown -R root:tomcat8 /opt/shibboleth-idp/conf/

chown -R root:tomcat8 /opt/shibboleth-idp/logs
chmod g+rw -R /opt/shibboleth-idp/logs

chown -R root:tomcat8 /opt/shibboleth-idp/credentials/
chmod g+r -R /opt/shibboleth-idp/credentials/

chown -R root:tomcat8 /opt/shibboleth-idp/metadata
chmod g+rw -R /opt/shibboleth-idp/metadata

### Deployment του Shibboleth Identity Provider στον Tomcat8

Θα πρέπει να δημιουργηθεί ένα deployment descriptor αρχείο που θα περιέχει τις απαιτούμενες λεπτομέρειες για τον Tomcat8 ώστε να διαβάσει και να κάνει deploy το war αρχείο του Shibboleth Identity Provider.

Δημιουργήστε το αρχείο */var/lib/tomcat8/conf/Catalina/localhost/idp.xml* με τα ακόλουθα περιεχόμενα:

<Context docBase="/opt/shibboleth-idp/war/idp.war"
 privileged="true"
 antiResourceLocking="false"
 swallowOutput="true" />

και στην συνέχεια επανεκκινήστε τον Tomcat8 με την ακόλουθη εντολή:

service tomcat8 restart

Για να βεβαιωθείτε ότι το deployment πήγε όπως θα έπρεπε, μπορείτε να ελέγξετε το output στα log files του Tomcat8 και του Shibboleth Identity Provider

Για τον Tomcat8, ελέγξτε το αρχείο */var/log/tomcat8/catalina.out*. Εάν όλα πήγαν καλά, θα δείτε τα ακόλουθα μηνύματα στο τέλος του αρχείου.

Oct 23, 2016 12:27:14 AM org.apache.catalina.startup.HostConfig deployDescriptor
INFO: Deploying configuration descriptor /etc/tomcat8/Catalina/localhost/idp.xml
Oct 23, 2016 12:27:57 AM org.apache.catalina.startup.HostConfig deployDescriptor
INFO: Deployment of configuration descriptor /etc/tomcat8/Catalina/localhost/idp.xml has finished in 43,819 ms

Αντίστοιχα για τον Shibboleth Idp, ελέγξτε το αρχείο */opt/shibboleth-idp/logs/idp-process.log*, στο οποίο θα πρέπει να δείτε τα ακόλουθα:

2016-10-23 00:27:55,926 - INFO [net.shibboleth.ext.spring.context.DelimiterAwareApplicationContext:577] - Refreshing WebApplicationContext for namespace 'idp-servlet': startup date [Sun Oct 23 00:27:55 EEST 2016]; parent: Root WebApplicationContext
2016-10-23 00:27:56,844 - INFO [net.shibboleth.idp.authn.impl.RemoteUserAuthServlet:193] - RemoteUserAuthServlet will process REMOTE\_USER, along with attributes [] and headers []

### Επιτάχυνση εκκίνησης Tomcat μετά την εγκατάσταση του Shibboleth Identity Provider

Για να επιταχύνετε την εκκίνηση και επανεκκίνηση του Tomcat, προτείνουμε να προσθέσετε τη λίστα με τα jar files του Shibboleth Identity Provider, στην λίστα με τα jars που δεν ελέγχονται κατά το startup. Εκτελώντας την παρακάτω εντολή, θα πάρετε μια comma-separated λίστα με όνοματα jar αρχείων. Προσθέστε τη λίστα αυτή στις ήδη υπάρχουσες τιμές του property *tomcat.util.scan.StandardJarScanFilter.jarsToSkip* στο αρχείο */etc/tomcat8/catalina.properties*

ls /opt/shibboleth-idp/webapp/WEB-INF/lib | awk '{print $1",\\"}'

# Παραμετροποίηση Shibboleth Identity Provider

Μετά την εγκατάσταση και το αρχικό deployment του Shibboleth Identity Provider, θα πρέπει να παραμετροποιήσετε τον Identity Provider σας σύμφωνα με τις γνωστές καλές πρακτικές και την [Πολιτική Χρήσης της Ομοσπονδίας](http://aai.grnet.gr/static/policy/policy-el.pdf) .

## Δημιουργία ιδιωτικού κλειδιού

O Shibboleth Identity Provider χρησιμοποιεί ζεύγη κλειδιών για να

* Υπογράφει ψηφιακά τα SAML μηνύματα που στέλνει σε SAML Service Providers
* Αποκρυπτογραφεί μηνύματα που λαμβάνει από SAML Service Providers

Κατα την εγκατάσταση του Shibboleth Identity Provider, το installation script δημιουργεί 3 ζεύγη κλειδιών RSA με μέγεθος 2048 bit μέσα στο directory /opt/shibboleth-idp/credentials

* idp-signing.{key,crt}
	+ Είναι το private key με το οποίο ο Identity Provider θα υπογράφει ψηφιακά τα μηνύματα που στέλνει στους SP
	+ To certificate (self signed και long lived) απλώς περιέχει το public key και δηλώνεται στα metadata ως <md:KeyDescriptor use="signing">
* idp-encryption.{key,crt}
	+ Το public key μπορεί να χρησιμοποιηθεί από SPs για να κρυπτογραφήσουν εισερχόμενα μηνύματα προς τον Identity Provider (τα αποκρυπτογραφεί με το private key)
	+ To μοναδικό ίσως valid use case είναι η κρυπτογράφηση του subject ενός SAML logout request από τον SP (αλλά οι Shibboleth Service Providers δεν το κάνουν by default).
	+ To certificate (self-signed και long-lived) απλώς περιέχει το public key και δηλώνεται στα metadata ως <md:KeyDescriptor use="encryption">
	+ Συνιστούμε να είναι διαφορετικό από το idp-signing keypair για λόγους ασφάλειας. (H παραβίαση της ασφάλειας του encryption key είναι **λιγότερο** σημαντική από την παραβίαση του signing key)
* idp-backchannel.{crt,p12}
	+ Χρησιμοποιείται στο TLS στο backchannel. Για τα δεδομένα της Ομοσπονδίας ΔΗΛΟΣ του ΕΔΕΤ και των περισσοτέρων SPs του eduGAIN, δεν χρειάζεται να υποστηρίζετε backchannel

Εάν επιθυμείτε να δημιουργήσετε εκ νέου τα ζεύγη κλειδιών μόνοι σας, ακολουθήστε τις παρακάτω οδηγίες

openssl req -new -x509 -nodes -sha256\
-newkey rsa:2048 -keyout /opt/shibboleth-idp/credentials/idp-signing.key \
-days 3650 -out /opt/shibboleth-idp/credentials/idp-signing.crt

openssl req -new -x509 -nodes -sha256\
-newkey rsa:2048 -keyout /opt/shibboleth-idp/credentials/idp-encryption.key \
-days 3650 -out /opt/shibboleth-idp/credentials/idp-encryption.crt

openssl req -new -x509 -nodes -sha256\
-newkey rsa:2048 -keyout /opt/shibboleth-idp/credentials/idp-backchannel.key \
-days 3650 -subj /CN=idp.yourdomain.gr -out /opt/shibboleth-idp/credentials/idp-backchannel.crt

## Βασική παραμετροποίηση Shibboleth Identity Provider

H λίστα με όλα τα αρχεία παραμετροποίησης, καθώς και τα κομμάτια της λειτουργικότητας που αυτά επηρεάζουν συντηρείται στο [shibboleth wiki](https://wiki.shibboleth.net/confluence/display/IDP30/ConfigurationFileSummary) και προτείνεται να την διαβάσετε πριν ξεκινήσετε με την παραμετροποίηση. Στις παρακάτω ενότητες παρουσιάζονται κομμάτια της βασικής παραμετροποίησης ενός Shibboleth Identity Provider v3 που προτείνονται από την Δ.Ο. της Ομοσπονδίας ΔΗΛΟΣ του ΕΔΕΤ.

### idp.properties

Στο αρχείο idp.properties, επιβεβαιώστε ή/και αλλάξτε την παραμετροποίηση σας σύμφωνα με τα παρακάτω:

* idp.entityID : Eπιβεβαιώνετε οτι το EntityID είναι αυτό που διαλέξατε κατά την εγκατάσταση του Shibboleth Identity Provider.
* idp.scope : Το scope του Παρόχου Ταυτότητας, το οποίο θα χρησιμοποιείται για να δημιουργηθούν οι τιμές των scoped attributes στο attribute resolver. Η τιμή του θα πρέπει να είναι το πρωτεύον DNS domain του φορέα.
* idp.cookie.secure : Oρίστε την τιμή σε *true*. Η αλλαγή αυτή θα ενεργοποιήσει το secure flag του session cookie που χρησιμοποιεί ο Shibboleth Identity Provider.
* idp.cookie.httpOnly : Oρίστε την τιμή σε *true*. Η αλλαγή αυτή θα ενεργοποιήσει το httpOnly flag του session cookie που χρησιμοποιεί ο Shibboleth Identity Provider.
* idp.cookie.domain : Ορίστε την τιμή στο FQDN του host
* idp.trust.signatures : Ορίστε την τιμή σε shibboleth.ExplicitKeySignatureTrustEngine ώστε ο έλεγχος των ψηφιακών υπογραφών των Service Providers να γίνεται μόνο με βάση τα κλειδιά που γνωρίζει ο Shibboleth Identity Provider από τα metadata
* idp.trust.certificates : Ορίστε την τιμή σε shibboleth.ExplicitKeyX509TrustEngine ώστε ο έλεγχος των ψηφιακών υπογραφών των Service Providers να γίνεται μόνο με βάση τα κλειδιά που γνωρίζει ο Shibboleth Identity Provider από τα metadata
* idp.encryption.optional : Επηρεάζει εάν θα επιτρέπεται να δημιουργεί assertions ο ΙdP σας, όταν δεν θα μπορεί να τα κρυπτογραφήσει (π.χ. Όταν δεν γνωρίζει το κλειδί του SP). Oρίστε την τιμή σε *false*.
* idp.authn.flows : Ορίστε την τιμή σε *Password*

### ldap.properties

Στο αρχείο ldap.properties θα πραγματοποιήσετε την απαιτούμενη παραμετροποίηση για να μπορέσει να χρησιμοποιήσει τον LDAP server σας ο Identity Provider για την αυθεντικοποίηση των χρηστών και την ανάκτηση των στοιχείων τους (attributes).

* idp.authn.LDAP.authenticator: Επιλέξτε ποιον τύπο Authenticator θα χρησιμοποιήσει Identity Provider. Οι επιλογές που έχετε για το πως θα γίνει η αυθεντικοποίηση είναι οι ακόλουθες:
	+ *anonSearchAuthenticator*: Ο Identity Provider Θα επιχειρήσει να κάνει anonymous search για το userDN και στη συνέχεια θα κάνει απόπειρα bind εκ μέρους του χρήστη με DN, password
	+ *bindSearchAuthenticator*: Ο Identity Provider θα επιχειρήσει να κάνει bind με ένα service account DN, password, και στην συνέχεια search για το user DN και απόπειρα bind με DN,password
	+ *directAuthenticator*: Σε περιπτώσεις όπου το DN των χρήστων έχει γνωστο format πχ CN=user\_name,ou=accounts,dc=domain,dc=edu, δηλαδή οι χρήστες είναι όλοι κάτω από το ίδιο OU, δεν γίνεται search αλλά απευθέιας bind
	+ *adAuthenticator* : Παραμετροποίηση που αφορά τη χρήση Active Directory.

Οι ακόλουθες τιμές παραμετροποίησης αφορούν πρόταση της Δ.Ο. της Ομοσπονδίας ΔΗΛΟΣ του ΕΔΕΤ για την χρήση anonymous search σε OpenLDAP server που υποστηρίζει STARTTLS.

* idp.authn.LDAP.authenticator : Ορίστε την τιμή *anonSearchAuthenticator*
* idp.authn.LDAP.ldapURL : Ορίστε την τιμή ως το ldap URL που αντιστοιχει στον host και τo port στο οποίο ακούει για συνδέσεις ο LDAP Server σας. (π.χ. ldap://ldap.grnet.gr:389)
* idp.authn.LDAP.useStartTLS : Ορίστε την τιμή σε *true*
* idp.authn.LDAP.useSSL : Ορίστε την τιμή σε *false*
* idp.authn.LDAP.connectTimeout : Ορίστε την τιμή σε *3000*
* idp.authn.LDAP.trustCertificates : Ορίστε την τιμή ώς το filepath του αρχείου που περιέχει το πιστοποιητικό που χρησιμοποιεί ο LDAP server σας για TLS σε pem format.
* idp.authn.LDAP.baseDN : Ορίστε το baseDN στον LDAP σας όπου θα πρέπει να συνδεθεί ο Identity Provider, π.χ. dc=grnet,dc=gr
* idp.authn.LDAP.subtreeSearch : Ορίστε την τιμή σε *true* αν θέλετε να κάνει subtree search στο baseDN σας ο Identity Provider για να βρεί τους χρήστες
* idp.authn.LDAP.userFilter : Ορίστε την τιμή ως το [ldap filter](https://tools.ietf.org/search/rfc4515) το οποίο θα χρησιμοποιεί ο Identity Provider για να ψάχνει τους χρήστες σας στον LDAP Server. Το macro *{user}* αντιστοιχεί στο input του χρήστη στην φόρμα σύνδεσης, οπότε αν για παράδειγμα οι χρήστες σας κάνουν login χρησιμοποιώντας το uid τους στον LDAP, η τιμή του userFilter θα ήταν *(uid={user})*
* idp.attribute.resolver.LDAP.ldapURL : Ορίστε την τιμή ώς *%{idp.authn.LDAP.ldapURL}*
* idp.attribute.resolver.LDAP.baseDN : Ορίστε την τιμή ώς *%{idp.authn.LDAP.baseDN}*
* idp.attribute.resolver.LDAP.bindDN : Ορίστε την τιμή ώς *%{idp.authn.LDAP.bindDN}*
* idp.attribute.resolver.LDAP.bindDNCredential : Ορίστε την τιμή ώς *%{idp.authn.LDAP.bindDNCredential}*
* idp.attribute.resolver.LDAP.useStartTLS : Ορίστε την τιμή ώς *%{idp.authn.LDAP.useStartTLS:true}*
* idp.attribute.resolver.LDAP.trustCertificates : Ορίστε την τιμή ώς *%{idp.authn.LDAP.trustCertificates}*
* idp.attribute.resolver.LDAP.searchFilter : Ορίστε την τιμή ώς *(uid=$requestContext.principalName)*

## Παραμετροποίηση Αυθεντικοποίησης

Ο τρόπος με τον οποιόν θα γίνεται η αυθεντικοποίηση χρηστών, παραμετροποιείται στα παρακάτω 2 αρχεία:

### ldap-authn-config.xml

Στο αρχείο */opt/shibboleth/conf/authn/ldap-authn-config.xml* αφαιρέστε τα σχόλια από το section

<!-- Anonymous Search Configuration -->
 <bean name="anonSearchAuthenticator" class="org.ldaptive.auth.Authenticator">
 <constructor-arg index="0" ref="anonSearchDnResolver" />
 <constructor-arg index="1" ref="authHandler" />
 </bean>
 <bean id="anonSearchDnResolver" class="org.ldaptive.auth.PooledSearchDnResolver" p:baseDn="%{idp.authn.LDAP.baseDN}"
 p:subtreeSearch="%{idp.authn.LDAP.subtreeSearch:false}" p:userFilter="%{idp.authn.LDAP.userFilter}"
 p:connectionFactory-ref="anonSearchPooledConnectionFactory" />
 <bean id="anonSearchPooledConnectionFactory" class="org.ldaptive.pool.PooledConnectionFactory"
 p:connectionPool-ref="anonSearchConnectionPool" />
 <bean id="anonSearchConnectionPool" class="org.ldaptive.pool.BlockingConnectionPool" parent="connectionPool"
 p:connectionFactory-ref="anonSearchConnectionFactory" p:name="search-pool" />
 <bean id="anonSearchConnectionFactory" class="org.ldaptive.DefaultConnectionFactory" p:connectionConfig-ref="anonSearchConnectionConfig" />
 <bean id="anonSearchConnectionConfig" parent="connectionConfig" />

### password-authn-config.xml

Στο αρχείο */opt/shibboleth-idp/conf/authn/password-authn-config.xml* ενεργοποιήστε την αυθεντικοποίηση μέσω LDAP ενεργοποιώντας το

<import resource="ldap-authn-config.xml" />

και φροντίζοντας να είναι απενεργοποιημένο οποιοδήποτε άλλο import statement.

## Παραμετροποίηση JPA Server Side Storage

Στην ενότητα αυτή περιγράφεται η παραμετροποίηση της επικοινωνίας του Shibboleth Identity Provider με την βάση δεδομένων που χρησιμοποιείται για την αποθήκευση δεδομένων σχετικών με τα Sessions, τα NameIDs των χρηστών κ.α.

Στο αρχείο /opt/shibboleth-idp/conf/global.xml ορίστε τα παρακάτω beans

<bean id="shibboleth.JPAStorageService"
 class="org.opensaml.storage.impl.JPAStorageService"
 p:cleanupInterval="%{idp.storage.cleanupInterval:PT10M}"
 c:factory-ref="shibboleth.JPAStorageService.EntityManagerFactory" />

<bean id="shibboleth.JPAStorageService.EntityManagerFactory"
 class="org.springframework.orm.jpa.LocalContainerEntityManagerFactoryBean">
 <property name="persistenceUnitName" value="storageservice" />
 <property name="packagesToScan" value="org.opensaml.storage.impl" />
 <property name="dataSource" ref="shibboleth.JPAStorageService.DataSource" />
 <property name="jpaVendorAdapter" ref="shibboleth.JPAStorageService.JPAVendorAdapter" />
 <property name="jpaDialect">
 <bean class="org.springframework.orm.jpa.vendor.HibernateJpaDialect" />
 </property>
</bean>

<bean id="shibboleth.JPAStorageService.JPAVendorAdapter"
 class="org.springframework.orm.jpa.vendor.HibernateJpaVendorAdapter">
 <property name="database" value="MYSQL" />
</bean>

<bean id="shibboleth.JPAStorageService.DataSource"
 class="com.zaxxer.hikari.HikariDataSource" destroy-method="close" lazy-init="true"
 p:driverClassName="com.mysql.jdbc.Driver"
 p:jdbcUrl="jdbc:mysql://localhost:3306/shibboleth"
 p:username="Username"
 p:password="Password" />

όπου θα πρέπει να αλλάξετε το *Username* και *Password* ώστε να αντιστοιχούν στα credentials του χρήστη που έχει τα απαιτούμενα δικαιώματα στην βάση δεδομένων.

Δημιουργήστε το αρχείο orm.xml με τα ακόλουθα περιεχόμενα

<?xml version="1.0" encoding="UTF-8"?>
<entity-mappings xmlns="http://java.sun.com/xml/ns/persistence/orm"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:schemaLocation="http://java.sun.com/xml/ns/persistence/orm
 http://java.sun.com/xml/ns/persistence/orm\_1\_0.xsd" version="1.0">
 <package>org.opensaml.storage.impl</package>

 <entity class="JPAStorageRecord" access="PROPERTY">
 <attributes>
 <basic name="value">
 <column name="value" nullable="false"/>
 </basic>
 </attributes>
 </entity>
</entity-mappings>

στο */opt/shibboleth-idp-3.3.0/edit-webapp/WEB-INF/classes/META-INF/*

Στη συνέχεια, κάντε build ξανα το idp.war για να περιέχει το αρχείο αυτό.

JAVA\_HOME=/usr/lib/jvm/java-8-openjdk-amd64/jre/ /opt/shibboleth-idp/bin/build.sh
Installation Directory: [/opt/shibboleth-idp]

Rebuilding /opt/shibboleth-idp/war/idp.war ...
...done

BUILD SUCCESSFUL
Total time: 10 seconds

Tώρα που το server side storage είναι παραμετροποιημένο, μπορεί να χρησιμοποιηθεί για την αποθήκευση sessions κ.α.

Στο αρχείο *idp.properties* ορίστε τα παρακάτω εάν επιθυμείτε να χρησιμοποιήσετε την λειτουργικότητα αυτή

* idp.session.StorageService = shibboleth.JPAStorageService Ελέγχει το που αποθηκεύονται τα user sessions.
* idp.replayCache.StorageService = shibboleth.JPAStorageService Ελέγχει το ποιον αποθηκευτικό χώρο χρησιμοποιεί η προστασία από Replay Attacks
* idp.consent.StorageService = shibboleth.JPAStorageService Ελέγχει που αποθηκεύονται οι προτιμήσεις των χρηστών σε σχέση με τη συγκατάθεση στην απελευθέρωση attributes.

## Παραμετροποίηση Δημιουργίας NameID

Στην έκδοση 3 του Shibboleth Identity Provider, η δημιουργία του SAML2 NameID έχει αποσυνδεθεί από την διαδικασία του attribute resolution. Η ποαραμετροποίηση πραγματοποιείται στα αρχεία *saml-nameid.xml* και *saml-nameid.properties*.

### SAML2 NameID με format transient

Η δημιουργία transient NameIDs υποστηρίζεται με την εγκατάσταση του Identity Provider χωρίς καμία επιπλέον παραμεροποίηση. Το transient NameID δημιουργείται με την χρησιμοποίηση του του idp.transientId.generator που κάνει χρήστη του κλειδιού που που είναι αποθηκευμένο στο JCEKS keystore του Identity Provider και έχει τα παρακάτω χαρακτηριστικά

* Αλλάζει σε κάθε session για κάθε χρήστη
* Διατηρεί την ιδιωτικότητα του χρήστη
* Δεν επιτρέπει στους SPs να κρατάνε state (εάν χρειάζονται)

### SAML2 NameID με format persistent

Προτείνεται η δημιουργία του SAML2 NameID με format persistant με χρησιμοποίηση του *shibboleth.StoredPersistentIdGenerator*. To StoredPersistentIdGenerator δημιουργεί το NameID την πρώτη φορά που θα χρειαστεί για έναν συνδιασμό <χρήστη, Service Provider> και στην συνέχεια το αποθηκεύει τοπικά σε μια βάση δεδομένων για μελλοντική ανάκτηση και χρησιμοποίηση.

#### Παραμετροποίηση

Στο αρχείο */opt/shibboleth-idp/conf/saml-nameid.properties* κάντε την παρακάτω παραμετροποίηση:

* idp.persistentId.sourceAttribute : Ορίστε την τιμή του ως το attribute το οποίο θα χρησιμοποιείται σαν πηγή για την δημιουργία του NameID. Θα πρέπει να προσέξετε τα ακόλουθα:
	+ Το attribute θα πρέπει να υπάρχει και να έχει τιμή για *ολους* τους χρήστες σας.
	+ Η τιμή του attribute θα πρέπει να έιναι σταθερή, μόνιμη και να μην υπάρχει περίπτωση να αποδοθεί σε άλλο χρήστη του Identity Provider σας ποτέ στο μέλλον.
* idp.persistentId.salt = Μια τυχαία τιμή που χρησιμοποιείται σαν salt στο hashing function που υπολογίζει την τιμή του NameID. Μπορείτε να χρησιμοποιήσετε το openssl για να δημιουργήσετε μια τυχαία τιμή
* openssl rand -base64 36
* idp.persistentId.generator : Ορίστε την τιμή ως *shibboleth.StoredPersistentIdGenerator*
* idp.persistentId.dataSource : Oρίστε την τιμή ως το αναγνωριστικό του DataSource που ορίστηκε παραπάνω (π.χ. shibboleth.JPAStorageService.DataSource)

#### Eνεργοποίηση Υποστήριξης SAML2PersistentGenerator

Ενεργοποιήστε το

<ref bean="shibboleth.SAML2PersistentGenerator" />

αφαιρώντας τα σχόλια στο αρχείο */opt/shibboleth-idp/conf/saml-nameid.xml*

## Παραμετροποίηση relying-party.xml

Στο relying-party.xml ορίζεται η παραμετροποίηση για τα SAML2 profiles που υποστηρίζει ο Identity Provider. Σας προτείνουμε

* Να υποστηρίξετε μόνο SAML2 profiles και από τα διαθέσιμα SAML2 μόνο τα front channel specific, δηλαδή τα SAML2.SSO, SAML2.Logout
* Nα δημιουργήσετε ένα <bean> για τον φορέα σας που θα κάνει inherit το *SAML2.SSO* για να μπορέσετε να κάνετε την σχετική παραμετροποίηση πιο εύκολα
* Να ορίσετε <bean> εξαιρέσεων όπου χρειάζεται που θα κάνουν inherit το δικό σας configuration.
* Στο SAML2.SSO να ορίσετε explicitly την σειρά με την οποία υποστηρίζετε την δημιουργία ΝameID, πρώτα με fomat peristent και μετά transient.
* Να ενεργοποιήσετε το User Consent για την απελευθέρωση των attributes των χρηστών

Μια υλοποίηση των προτάσεων αυτών φαίνεται στο παράδειγμα που ακολουθεί

<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
 xmlns:context="http://www.springframework.org/schema/context"
 xmlns:util="http://www.springframework.org/schema/util"
 xmlns:p="http://www.springframework.org/schema/p"
 xmlns:c="http://www.springframework.org/schema/c"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd
 http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context.xsd
 http://www.springframework.org/schema/util http://www.springframework.org/schema/util/spring-util.xsd"

 default-init-method="initialize"
 default-destroy-method="destroy">

 <bean id="shibboleth.UnverifiedRelyingParty" parent="RelyingParty">
 <property name="profileConfigurations">
 <list>
 </list>
 </property>
 </bean>

 <bean id="shibboleth.DefaultRelyingParty" parent="RelyingParty">
 <property name="profileConfigurations">
 <list>
 <bean parent="SAML2.SSO.grnet" p:postAuthenticationFlows="attribute-release" />
 <ref bean="SAML2.Logout" />
 </list>
 </property>
 </bean>

 <bean id="SAML2.SSO.grnet" parent="SAML2.SSO"
 p:nameIDFormatPrecedence="#{{
 'urn:oasis:names:tc:SAML:2.0:nameid-format:persistent',
 'urn:oasis:names:tc:SAML:2.0:nameid-format:transient'}}" />

 <util:list id="shibboleth.RelyingPartyOverrides">
 <!-- Override profile for Office365 that requires unencrypted assertions -->
 <bean parent="RelyingPartyByName" c:relyingPartyIds="urn:federation:MicrosoftOnline">
 <property name="profileConfigurations">
 <list>
 <bean parent="SAML2.SSO.grnet" p:encryptAssertions="false"/>
 </list>
 </property>
 </bean>
 </util:list>
</beans>

## Παραμετροποίηση metadata-providers.xml

Στο αρχείο metadata-providers.xml, ορίζονται οι πηγές από τις οποίες θα διαβάζει και θα εμπιστεύεται metadata ο ΙdP. Η Ομοσπονδία ΔΗΛΟΣ του ΕΔΕΤ παρέχει metadata aggregates για τα Entities της Ομοσπονδίας και για τους SP που δημοσιεύονται στο eduGAIN, αν ο Identity Provider σας συμμετέχει στο eduGAIN.

Τα metadata aggregates της Ομοσπονδίας ΔΗΛΟΣ του ΕΔΕΤ είναι ψηφιακά υπογεγραμμένα με το ιδιωτικό κλειδί της Ομοσπονδίας. Το δημόσιο κλειδί που αντιστοιχεί στο ιδιωτικό αυτό κλειδί δημοσιεύεται [εδώ](https://md.aai.grnet.gr/grnetaai_md_cert.html) και μπορείτε να το κατεβάσετε σε PEM format από [το αντίστοιχο link](https://md.aai.grnet.gr/grnetaai_md_cert.pem).

Για να μπορεί ο Identity Provider να επιβεβαιώσει την γνησιότητα των metadata που διαβάζει από τα aggregates της ομοσπονδίας, θα πρέπει να γνωρίζει το δημόσιο αυτό κλειδί. Η απαιτούμενη παραμετροποίηση έχει ως εξής:

* Κατεβάστε το νέο πιστοποιητικό και αποθηκεύστε το τοπικά
* curl -o grnet-mdsigner.crt https://md.aai.grnet.gr/grnetaai\_md\_cert.pem
* Υπολογίστε τα sha1 και sha256 fingerprints του πιστοποιητικού
* openssl x509 -fingerprint -sha1 -noout -in grnet-mdsigner.crt
openssl x509 -fingerprint -sha256 -noout -in grnet-mdsigner.crt
* Συγκρίνετε τις τιμές που υπολογίστηκαν με τις τιμές που δημοσιεύονται στο <http://md.aai.grnet.gr/grnetaai_md_cert.html>
* Eάν οι τιμές είναι σωστές, αντιγράψτε το αρχείο στο directory */opt/shibboleth-idp/credentials*

Tώρα που o Identity Provider γνωρίζει πως να επιβεβαιώσει την γνησιότητα των metadata της Ομοσπονδίας ΔΗΛΟΣ του ΕΔΕΤ, προχωρήστε στην παραμετροποίηση των Metadata Providers. Στο αρχείο metadata-providers.xml θα πρέπει να ορίσετε τα απαραίτητα <MetadataProvider> elements έτσι ώστε:

* Να καταναλώνονται τα metadata της Ομοσπονδίας ΔΗΛΟΣ του ΕΔΕΤ από το <https://md.aai.grnet.gr/aggregates/grnet-metadata.xml>
* Σε περίπτωση που ο IDP σας συμμετέχει στο eduGAIN, να καταναλωνόνται και τα metadata με τους Service Providers του eduGAIN από το <https://md.aai.grnet.gr/aggregates/grnet-metadata.xml>
* Να χρησιμοποιούνται MetadataProvier τύπου FileBackedHTTPMetadataProvider
* Να ελέγχεται η γνησιότητα των metadata με έλεγχο της ψηφιακής υπογραφής χρησιμοποιώντας SignatureValidation ΜetadataFilter
* Να ελέγχεται η παλαιότητα των metadata έτσι ώστε να μην δέχεται metadata aggregates με πολύ μεγάλη διάρκεια ή aggregates που δεν λήγουν ποτέ με την χρήση RequiredValidUntil MetadataFilter

Οι προτάσεις αυτές υλοποιούνται στο παρακάτω υπόδειγμα metadata-providers.xml

<?xml version="1.0" encoding="UTF-8"?>
<!-- This file is an EXAMPLE metadata configuration file. -->
<MetadataProvider id="ShibbolethMetadata" xsi:type="ChainingMetadataProvider"
 xmlns="urn:mace:shibboleth:2.0:metadata"
 xmlns:resource="urn:mace:shibboleth:2.0:resource"
 xmlns:security="urn:mace:shibboleth:2.0:security"
 xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:schemaLocation="urn:mace:shibboleth:2.0:metadata http://shibboleth.net/schema/idp/shibboleth-metadata.xsd
 urn:mace:shibboleth:2.0:resource http://shibboleth.net/schema/idp/shibboleth-resource.xsd
 urn:mace:shibboleth:2.0:security http://shibboleth.net/schema/idp/shibboleth-security.xsd
 urn:oasis:names:tc:SAML:2.0:metadata http://docs.oasis-open.org/security/saml/v2.0/saml-schema-metadata-2.0.xsd">

 <!-- Μetadata Ομοσπονδίας ΔΗΛΟΣ του ΕΔΕΤ -->
 <MetadataProvider
 xmlns="urn:mace:shibboleth:2.0:metadata"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:schemaLocation="urn:mace:shibboleth:2.0:metadata http://shibboleth.net/schema/idp/shibboleth-metadata.xsd"
 xsi:type="FileBackedHTTPMetadataProvider"
 id="GRNETFederation"
 metadataURL="https://md.aai.grnet.gr/aggregates/grnet-metadata.xml"
 backingFile="%{idp.home}/metadata/grnet-metadata.xml">
 <MetadataFilter xsi:type="RequiredValidUntil" maxValidityInterval="P7D" />
 <MetadataFilter xsi:type="SignatureValidation" requireSignedRoot="true"
 certificateFile="%{idp.home}/credentials/grnet-mdsigner.crt" />
 </MetadataProvider>

 <!-- Metadata των Service Providers που συμμετέχουν στο eduGAIN -->
 <MetadataProvider
 xmlns="urn:mace:shibboleth:2.0:metadata"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:schemaLocation="urn:mace:shibboleth:2.0:metadata http://shibboleth.net/schema/idp/shibboleth-metadata.xsd"
 xsi:type="FileBackedHTTPMetadataProvider"
 id="GRNETEdugainDownstream"
 metadataURL="https://md.aai.grnet.gr/feeds/edugain-sp-samlmd.xml"
 backingFile="%{idp.home}/metadata/edugain-metadata.xml">

 <MetadataFilter xsi:type="RequiredValidUntil" maxValidityInterval="P7D" />
 <MetadataFilter xsi:type="SignatureValidation" requireSignedRoot="true"
 certificateFile="%{idp.home}/credentials/grnet-mdsigner.crt" />
 </MetadataProvider>
 </MetadataProvider>

## Παραμετροποίηση attribute-resolver.xml

Το αρχείο αυτό περιέχει τους ορισμούς των attributes που μπορεί να έχει διαθέσιμα ο Identity Provider από το αποθετήριο χρηστών

Προτείνεται να χρησιμοποιηθεί το αρχειο attribute-resolver-full.xml το οποίο υπάρχει στην εγκατάσταση σαν βάση με τις ακόλουθες αλλαγές:

* Να μετονομαστεί σε attribute-resolver.xml
* Nα ενεργοποιηθούν (αφαιρεθούν τα σχόλια) μόνο οι ορισμοί attributes τα οποία υπάρχουν στον LDAP σας.
* Να αφαιρεθούν τα AttributeEncoders με τύπο *xsi:type="SAML1String"*
* Να προστεθούν οι απαραίτητοι ορισμοί για τα attributes του schac σχήματος που χρησιμοποιείτε και υπάρχουν στον LDAP server σας.
* Να ενεργοποιηθεί το Example LDAP Connector, το οποίο και χρησιμοποιεί παραμετροποίηση που έχει γίνει ήδη στα αρχεία *idp.properties* και *ldap.properties*

Μια τυπική μορφή του αρχείου attribute-resolver.xml για έναν Identity Provider που συμμετέχει στην Ομοσπονδία ΔΗΛΟΣ του ΕΔΕΤ, παρουσιάζεται παρακάτω:

<?xml version="1.0" encoding="UTF-8"?>
<AttributeResolver
 xmlns="urn:mace:shibboleth:2.0:resolver"
 xmlns:sec="urn:mace:shibboleth:2.0:security"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:schemaLocation="urn:mace:shibboleth:2.0:resolver http://shibboleth.net/schema/idp/shibboleth-attribute-resolver.xsd
 urn:mace:shibboleth:2.0:security http://shibboleth.net/schema/idp/shibboleth-security.xsd">

 <!-- ========================================== -->
 <!-- Attribute Definitions -->
 <!-- ========================================== -->

 <!-- Schema: Core schema attributes-->
 <AttributeDefinition xsi:type="Simple" id="uid" sourceAttributeID="uid">
 <Dependency ref="myLDAP" />
 <AttributeEncoder xsi:type="SAML2String" name="urn:oid:0.9.2342.19200300.100.1.1" friendlyName="uid" encodeType="false" />
 </AttributeDefinition>

 <AttributeDefinition xsi:type="Simple" id="email" sourceAttributeID="mail">
 <Dependency ref="myLDAP" />
 <AttributeEncoder xsi:type="SAML2String" name="urn:oid:0.9.2342.19200300.100.1.3" friendlyName="mail" />
 </AttributeDefinition>

 <AttributeDefinition xsi:type="Simple" id="mobileNumber" sourceAttributeID="mobile">
 <Dependency ref="myLDAP" />
 <AttributeEncoder xsi:type="SAML2String" name="urn:oid:0.9.2342.19200300.100.1.41" friendlyName="mobile" encodeType="false" />
 </AttributeDefinition>

 <AttributeDefinition xsi:type="Simple" id="surname" sourceAttributeID="sn">
 <Dependency ref="myLDAP" />
 <AttributeEncoder xsi:type="SAML2String" name="urn:oid:2.5.4.4" friendlyName="sn" encodeType="false" />
 </AttributeDefinition>

 <AttributeDefinition xsi:type="Simple" id="commonName" sourceAttributeID="cn">
 <Dependency ref="myLDAP" />
 <AttributeEncoder xsi:type="SAML2String" name="urn:oid:2.5.4.3" friendlyName="cn" />
 </AttributeDefinition>

 <AttributeDefinition xsi:type="Simple" id="organizationName" sourceAttributeID="o">
 <Dependency ref="myLDAP" />
 <AttributeEncoder xsi:type="SAML2String" name="urn:oid:2.5.4.10" friendlyName="o" encodeType="false" />
 </AttributeDefinition>

 <AttributeDefinition xsi:type="Simple" id="telephoneNumber" sourceAttributeID="telephoneNumber">
 <Dependency ref="myLDAP" />
 <AttributeEncoder xsi:type="SAML2String" name="urn:oid:2.5.4.20" friendlyName="telephoneNumber" encodeType="false" />
 </AttributeDefinition>

 <AttributeDefinition xsi:type="Simple" id="givenName" sourceAttributeID="givenName">
 <Dependency ref="myLDAP" />
 <AttributeEncoder xsi:type="SAML2String" name="urn:oid:2.5.4.42" friendlyName="givenName" encodeType="false" />
 </AttributeDefinition>

 <AttributeDefinition xsi:type="Simple" id="displayName" sourceAttributeID="displayName">
 <Dependency ref="myLDAP" />
 <AttributeEncoder xsi:type="SAML2String" name="urn:oid:2.16.840.1.113730.3.1.241" friendlyName="displayName" encodeType="false" />
 </AttributeDefinition>

 <AttributeDefinition xsi:type="Simple" id="eduPersonAffiliation" sourceAttributeID="eduPersonAffiliation">
 <Dependency ref="myLDAP" />
 <AttributeEncoder xsi:type="SAML2String" name="urn:oid:1.3.6.1.4.1.5923.1.1.1.1" friendlyName="eduPersonAffiliation" encodeType="false" />
 </AttributeDefinition>

 <AttributeDefinition xsi:type="Simple" id="eduPersonEntitlement" sourceAttributeID="eduPersonEntitlement">
 <Dependency ref="myLDAP" />
 <AttributeEncoder xsi:type="SAML2String" name="urn:oid:1.3.6.1.4.1.5923.1.1.1.7" friendlyName="eduPersonEntitlement" encodeType="false" />
 </AttributeDefinition>

 <AttributeDefinition xsi:type="Simple" id="eduPersonPrimaryAffiliation" sourceAttributeID="eduPersonPrimaryAffiliation">
 <Dependency ref="myLDAP" />
 <AttributeEncoder xsi:type="SAML2String" name="urn:oid:1.3.6.1.4.1.5923.1.1.1.5" friendlyName="eduPersonPrimaryAffiliation" encodeType="false" />
 </AttributeDefinition>

 <AttributeDefinition xsi:type="Prescoped" id="eduPersonPrincipalName" sourceAttributeID="eduPersonPrincipalName">
 <Dependency ref="myLDAP" />
 <AttributeEncoder xsi:type="SAML2ScopedString" name="urn:oid:1.3.6.1.4.1.5923.1.1.1.6" friendlyName="eduPersonPrincipalName" encodeType="false" />
 </AttributeDefinition>

 <AttributeDefinition xsi:type="Scoped" id="eduPersonScopedAffiliation" scope="%{idp.scope}" sourceAttributeID="eduPersonAffiliation">
 <Dependency ref="myLDAP" />
 <AttributeEncoder xsi:type="SAML2ScopedString" name="urn:oid:1.3.6.1.4.1.5923.1.1.1.9" friendlyName="eduPersonScopedAffiliation" encodeType="false" />
 </AttributeDefinition>

 <AttributeDefinition xsi:type="Simple" id="eduPersonAssurance" sourceAttributeID="eduPersonAssurance">
 <Dependency ref="myLDAP" />
 <AttributeEncoder xsi:type="SAML2String" name="urn:oid:1.3.6.1.4.1.5923.1.1.1.11" friendlyName="eduPersonAssurance" encodeType="false" />
 </AttributeDefinition>

 <!-- schac -->
 <AttributeDefinition id="schacUserStatus" xsi:type="Simple" sourceAttributeID="schacUserStatus">
 <Dependency ref="myLDAP" />
 <AttributeEncoder xsi:type="SAML2String" name="urn:oid:1.3.6.1.4.1.25178.1.2.19" friendlyName="schacUserStatus" />
 </AttributeDefinition>

 <AttributeDefinition id="schacPersonalUniqueCode" xsi:type="Simple" sourceAttributeID="schacPersonalUniqueCode">
 <Dependency ref="myLDAP" />
 <AttributeEncoder xsi:type="SAML2String" name="urn:oid:1.3.6.1.4.1.25178.1.2.14" friendlyName="schacPersonalUniqueCode" />
 </AttributeDefinition>

 <AttributeDefinition id="schacHomeOrganization" xsi:type="Simple" sourceAttributeID="schacHomeOrganization">
 <Dependency ref="staticConnector" />
 <AttributeEncoder xsi:type="SAML2String" name="urn:oid:1.3.6.1.4.1.25178.1.2.9" friendlyName="schacHomeOrganization" />
 </AttributeDefinition>

 <!-- grEduPerson -->
 <AttributeDefinition id="grEduPersonUndergraduateBranch" xsi:type="Simple" sourceAttributeID="grEduPersonUndergraduateBranch">
 <Dependency ref="myLDAP" />
 <AttributeEncoder xsi:type="SAML2String" name="urn:oid:1.3.6.1.4.1.16515.2.3.2.1" friendlyName="grEduPersonUndergraduateBranch" />
 </AttributeDefinition>

 <!-- ========================================== -->
 <!-- Data Connectors -->
 <!-- ========================================== -->

 <DataConnector id="staticConnector" xsi:type="Static" >
 <Attribute id="eduPersonOrgUnitDN">
 <Value>OU=Technical,dc=domain,dc=gr</Value>
 </Attribute>
 <Attribute id="eduPersonEntitlement">
 <Value>urn:mace:dir:entitlement:common-lib-terms</Value>
 <Attribute id="schacHomeOrganization">
 <Value>domain.gr</Value>
 </Attribute>
 </DataConnector>

 <!-- LDAP Connector -->
 <DataConnector id="myLDAP" xsi:type="LDAPDirectory"
 ldapURL="%{idp.attribute.resolver.LDAP.ldapURL}"
 baseDN="%{idp.attribute.resolver.LDAP.baseDN}"
 useStartTLS="%{idp.attribute.resolver.LDAP.useStartTLS:true}"
 connectTimeout="%{idp.attribute.resolver.LDAP.connectTimeout}"
 responseTimeout="%{idp.attribute.resolver.LDAP.responseTimeout}">
 <FilterTemplate>
 <![CDATA[
 %{idp.attribute.resolver.LDAP.searchFilter}
 ]]>
 </FilterTemplate>
 </DataConnector>

</AttributeResolver>

## Παραμετροποίηση απελευθέρωσης Attributes

Η παραμετροποίηση της απελευθέρωσης attributes γίνεται στο αρχείο */opt/shibboleth-idp/conf/attribute-filter.xml* με την μορφή AttributeFilterPolicy elements , όπου ορίζεται ποια attributes επιτρέπεται ή απαγορεύεται να απελευθερώνει ο Identity Provider σε συγκεκριμένους Service Providers ή ομάδες Service Providers.

Η παραμετροποίηση αυτή είναι καθαρά ευθύνη του διαχειριστή του κάθε Identity Provider και εξαρτάται από την Πολιτική Ασφαλείας του κάθε Ιδρύματος. Οι προτάσεις της Δ.Ο. της Ομοσπονδίας ΔΗΛΟΣ του ΕΔΕΤ είναι :

* Να λάβετε υπ'όψιν την ανάγκη προστασίας της ιδιωτικότητας των χρηστών σας
* Να λάβετε υπ'όψιν την ανάγκη των χρηστών σας να έχουν πρόσβαση σε υπηρεσίες που τους είναι απαραίτητες για να επιτελέσουν το ακαδημαϊκό ή ερευνητικό τους έργο.
* Να ενεργοποιήσετε την λειτουργία User Consent για να είναι οι χρήστες σας οι τελικοί υπεύθυνοι για το αν θα επιτρέψουν την απελευθέρωση των attributes τους προς οποιονδήποτε Service Provider.

Oι Service Providers που λειτουργεί και διαχειρίζεται το ΕΔΕΤ, ανήκουν σε ένα EntityGroup μέσα στα Metadata της Ομοσπονδίας ΔΗΛΟΣ του ΕΔΕΤ με GroupID \*[http://aai.grnet.gr/entities/grnet/\*](http://aai.grnet.gr/entities/grnet/%2A) ώστε να μπορείτε να ρυθμίσετε παρόμοιες πολιτικές απελευθέρωσης για όλους

Ακόμα, για κάθε υπηρεσία, τα attributes τα οποία είναι απαραίτητα για κάθε υπηρεσία, περιέχονται στα metadata κάθε υπηρεσίας μεσα στο [metadata aggregate](https://md.aai.grnet.gr/aggregates/grnet-metadata.xml) της Ομοσπονδίας ΔΗΛΟΣ του ΕΔΕΤ. Για παράδειγμα η Υπηρεσία e:Presence με SAML EntityID το \*[https://new.epresence.grnet.gr/shibboleth\*](https://new.epresence.grnet.gr/shibboleth%2A), περιέχει στα metadata της το ακόλουθο:

<md:RequestedAttribute FriendlyName="givenName" Name="urn:oid:2.5.4.42" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri" isRequired="true"/>
<md:RequestedAttribute FriendlyName="sn" Name="urn:oid:2.5.4.4" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri" isRequired="true"/>
<md:RequestedAttribute FriendlyName="mail" Name="urn:oid:0.9.2342.19200300.100.1.3" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri" isRequired="false"/>
<md:RequestedAttribute FriendlyName="telephoneNumber" Name="urn:oid:2.5.4.20" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri" isRequired="false"/>
<md:RequestedAttribute FriendlyName="schacHomeOrganization" Name="urn:oid:1.3.6.1.4.1.25178.1.2.9" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri" isRequired="true"/>

To οποίο δηλώνει ότι η υπηρεσία για να λειτουργήσει, χρειάζεται από τον Identity Provider να απελευθερώσει για τον χρήστη τα

* givenName (όνομα)
* sn (επώνυμο)
* schacHomeOrganization

Ενώ επιθυμητά (αν και όχι αναγκαία) είναι και τα

* mail
* telephoneNumber

Μια τέτοια πολιτική θα μπορούσε να αποδοθεί στο ακόλουθο Attribute Filter Policy

<AttributeFilterPolicy id="epresence">
 <PolicyRequirementRule xsi:type="Requester" value="https://new.epresence.grnet.gr/shibboleth" />
 <AttributeRule attributeID="email">
 <PermitValueRule xsi:type="ANY" />
 </AttributeRule>
 <AttributeRule attributeID="telephoneNumber">
 <PermitValueRule xsi:type="ANY" />
 </AttributeRule>
 <AttributeRule attributeID="schacHomeOrganization">
 <PermitValueRule xsi:type="ANY" />
 </AttributeRule>
 <AttributeRule attributeID="givenName">
 <PermitValueRule xsi:type="ANY" />
 </AttributeRule>
 <AttributeRule attributeID="surname">
 <PermitValueRule xsi:type="ANY" />
 </AttributeRule>
 </AttributeFilterPolicy>

## Παραμετροποίηση Single Log Out

Ο Shibboleth Identity Provider στις εκδόσεις μετά την 3.2.1 υποστηρίζει 2 τρόπους για Single Log Out. Οι 2 τρόποι ( proprietary και SAML ) περιγράφονται στο SLO Configuration <https://wiki.shibboleth.net/confluence/display/IDP30/LogoutConfiguration>\_.

Η απαιτούμενη παραμετροποίηση στο αρχείο *idp.properties* είναι η ακόλουθη:

* idp.session.trackSPSessions = true
* idp.logout.elaboration = true
* idp.session.secondaryServiceIndex = true

## Παραμετροποίηση αυτόματης ανανέωσης του κλειδιού του DataSealer

Ο Shibboleth Identity Provider χρησιμοποιεί το DataSealer component για να προστατεύσει session cookies, να δημιουργήσει transient NameIDs κ.α. Το DataSealer component αυτό χρησιμοποιεί ένα AES κλειδί που είναι αποθηκευμένο σε ένα JCEKS keystore, προστατευμένο με τον κωδικό τον οποίον ορίσατε κατά την εγκατάσταση. Ακόμα, υπάρχει ένα αρχείο που αποθηκεύει μόνο την τρέχουσα έκδοση του κλειδιού το οποίο χρησιμοποιείται. Ο Shibboleth Identity Provider παρέχει ένα utility (seckeygen) το οποίο μπορεί να χρησιμοποιηθεί για την ανανέωση του κλειδιού αυτού σε τακτά χρονικά διαστήματα. Η ανανέωση προτείνεται να γίνεται καθημερινά, για να μειωθούν οι πιθανότητες να διαρρεύσει το κλειδί και να μπορεί κάποιος κακόβουλος χρήστης να χρησιμοποιήσει το κλειδί αυτό εάν διαρρεύσει. (Γνωρίζοντας την τιμή του κλειδιού, κάποιος μπορεί να κατασκευάσει session cookies και να υποδυθεί οποιονδήποτε χρήστη)

Προτείνεται η χρησιμοποίηση του παρακάτω bash script σε ένα cronjob το οποίο θα τρέχει καθημερινά. Σχόλια για την λειτουργικότητα του script παρέχονται και inline αλλά αυτό που κάνει είναι να χρησιμοποιεί το seckeygen για να δημιουργεί ένα νέο κλειδί (προστατευμένο με τον κωδικό που έχετε ορίσει) και να ανανεώνει το version file

#!/bin/bash

set -e
set -u

# Default IDP\_HOME if not already set
if [ ! -d "${IDP\_HOME:=/opt/shibboleth-idp}" ]
then
 echo "ERROR: Directory does not exist: ${IDP\_HOME}" >&2
 exit 1
fi

function get\_config {
 # Key to lookup (escape . for regex lookup)
 local KEY=${1:?"No key provided to look up value"}
 # Passed default value
 local DEFAULT="${2:-}"
 # Lookup key, strip spaces, replace idp.home with IDP\_HOME value
 local RESULT=$(sed -rn '/^'"${KEY//./\\.}"'\s\*=/ { s|^[^=]\*=(.\*)\s\*$|\1|; s|%\{idp\.home\}|'"${IDP\_HOME}"'|g; p}' ${IDP\_HOME}/conf/idp.properties)
 # Set if no result with default - exit if no default
 echo ${RESULT:-${DEFAULT:?"No value in config and no default defined for: '${KEY}'"}}
}

# Get config values
## Official config items ##
storefile=$(get\_config idp.sealer.storeResource)
versionfile=$(get\_config idp.sealer.versionResource)
storepass=$(get\_config idp.sealer.storePassword)
alias=$(get\_config idp.sealer.aliasBase secret)
## Extended config items ##
count=$(get\_config idp.sealer.\_count 30)
# default cannot be empty - so "self" is the default (self is skipped for syncing)
sync\_hosts=$(get\_config idp.sealer.\_sync\_hosts ${HOSTNAME})

# Run the keygen utility
${0%/\*}/runclass.sh net.shibboleth.utilities.java.support.security.BasicKeystoreKeyStrategyTool \
 --storefile "${storefile}" \
 --storepass "${storepass}" \
 --versionfile "${versionfile}" \
 --alias "${alias}" \
 --count "${count}"

# Display current version
# Βάλτε την επόμενη γραμή σε σχόλιο εάν δεν επιθυμείτε να στέλνεται email επιβεβαίωσης σε κάθε επιτυχή εκτέλεση του script
echo "INFO: $(tac "${versionfile}" | tr "\n" " ")" >&2

Για να λειτουργήσει το παραπάνω script, θα πρέπει να οριστεί μια νέα παράμετρος στο αρχείο *idp.properties* με το όνομα idp.sealer.\_count και τιμή έναν αριθμό που να δείχνει πόσα προηγούμενα versions του κλειδιού να κρατούνται στο JCEKS keystore.

Έχοντας δημιουργήσει το παραπάνω script και με τις παραδοχές οτι

* είναι αποθηκευμένο στο /opt/shibboleth-idp/bin με το όνομα dailyRotateKey.sh
* το dailyRotateKey.sh είναι εκτελέσιμο (π.χ. χρησιμοποιώντας chmod +x)

το cronjob definition Θα είναι της μορφής

::

MAILTO=emailaddresstoreceivealerts 0 01 \* \* \* /opt/shibboleth-idp/bin/dailyRotateKey.sh

για να εκτελείται καθημερινά στη 01:00 .

## Παραμετροποίηση logging

Στο αρχείο logback.xml παραμετροποιούνται τα επίπεδα των Loggers που χρησιμοποιεί ο Identity Provider. Επιγραμματικά, τα σημαντικότερα από αυτά και οι τιμές που προτείνονται:

* <variable name="idp.loglevel.idp" value="INFO" /> : Μηνύματα σχετικά με τον IdP. Χρησιμο να γίνει DEBUG για οποιαδήποτε προσπάθεια troubleshooting.
* <variable name="idp.loglevel.ldap" value="WARN" /> : Mηνύματα σχετικα με την επικοινωνία με τον LDAP server. Χρήσιμο να γίνει DEBUG σε περιπτώσεις προβλημάτων αυθεντικοποίησης.
* <variable name="idp.loglevel.messages" value="INFO" /> : SAML μηνύματα που λαμβάνει και στέλνει ο IdP.
* <variable name="idp.loglevel.encryption" value="INFO" /> : Στο DEBUG level εμφανίζει στο idp-process.log τα SAML Assertions unencrypted. Πολύ χρήσιμο για debugging οποιουδήποτε προβλήματος καθώς μπορείτε να δείτε ακριβως το Assertion που στεέλνεται σε κάθε IdP.
* <variable name="idp.loglevel.opensaml" value="INFO" /> : Μηνύματα από την βιβλιοθήκη OpenSAML.

## Ενεργοποίηση Υποστήριξης Ελληνικών στη διεπαφή χρήστη

Κατεβάστε το αρχείο με τις μεταφράσεις από την [αντίστοιχη σελίδα](https://wiki.shibboleth.net/confluence/display/IDP30/MessagesTranslation) και αποθηκεύστε το αρχείο με όνομα *messages\_el.properties* στον φάκελο /opt/shibboleth-idp/messages. Aν οι χρήστες σας έρθουν με σωστό Accept-Language header, θα δουν την ελληνική διεπαφή χρήστη.