TUTORIAL

II Convegno IDEM

Bari, 9-10 Marzo 2010

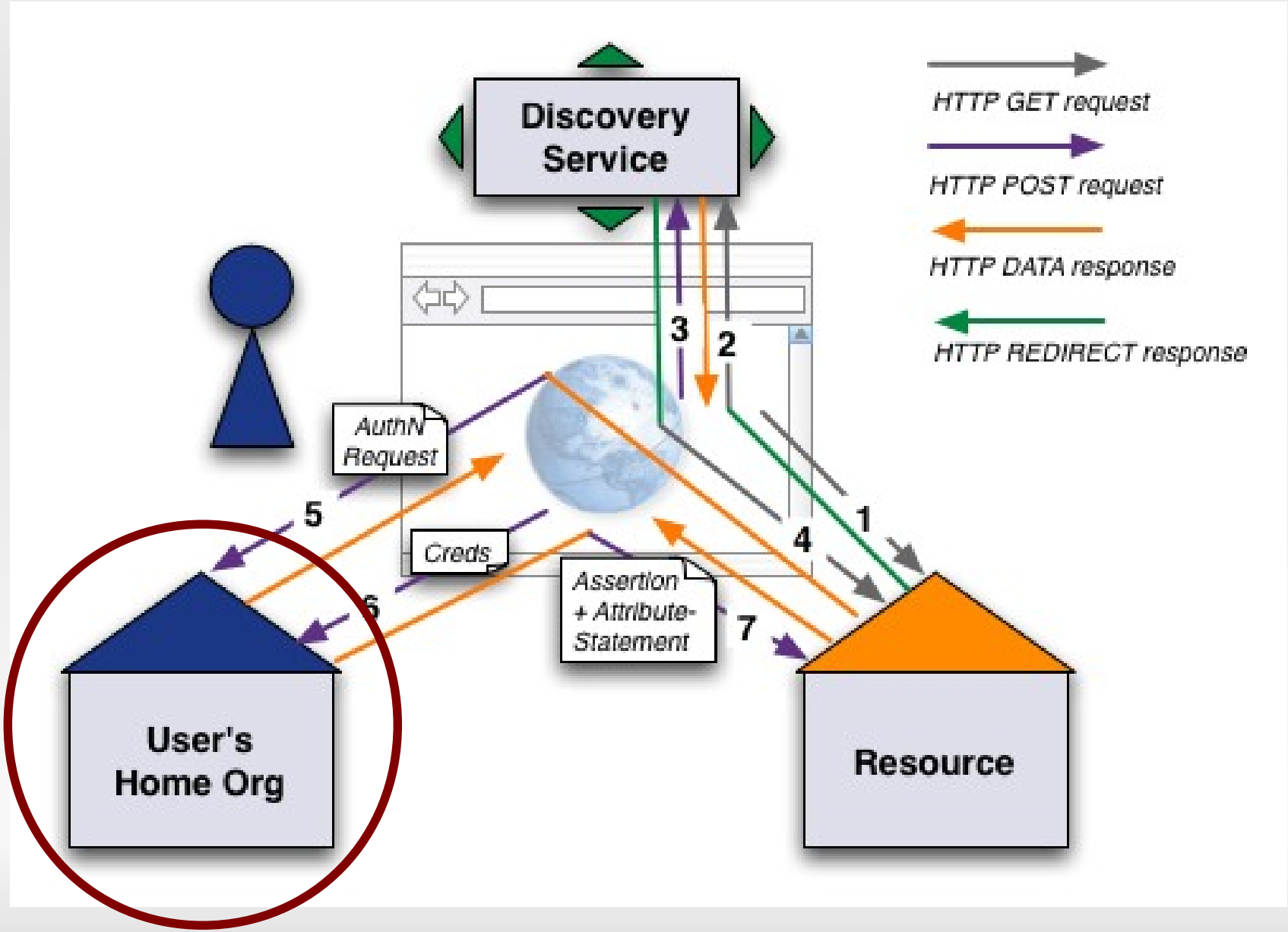
IdP, passo dopo passo!

Shibboleth 2.1 on CentOS 5.4

Relatore:
Claudio Marotta

Tutor:
Virginia Calabritto
Ilaria De Marinis
Massimo Ianigro
Maria Laura Mantovani

Déjà vu


Discovery Service
HTTP GET request
HTTP POST request
HTTP DATA response
HTTP REDIRECT response
3
2
AuthN Request
5
Creds
6
Assertion + Attribute-Statement
4
1
7
User's Home Org
Resource

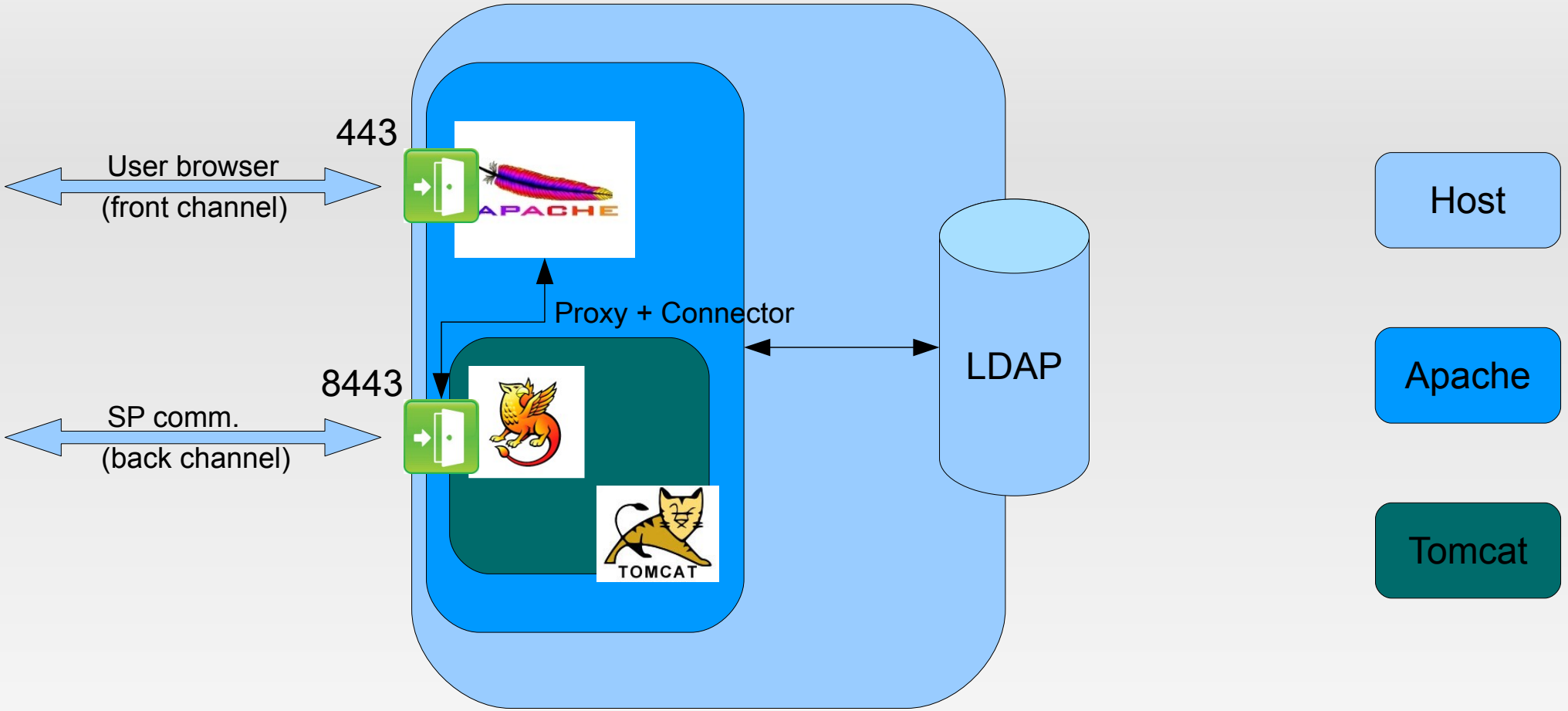
Prima di cominciare

- Configurate la rete utilizzando i parametri che vi sono stati consegnati.
- Se siete guru, sapete come usare la console per farlo
- Se siete un po' meno guru, utilizzate l'interfaccia grafica
  - sistema → amministrazione → rete
    - Dispositivi → eth0 → ip, mask, gw
    - DNS → hostname, server
  - Riavviate i servizi di rete (service network restart)
- Check: controllate se siete online.

Come lavoreremo

- Macchina Virtuale, installazione 'base' di CentOS 5.4
- Server LDAP locale (molto semplice)
- **/home/idem/Desktop/src,** sorgenti dei software da installare
- **/home/idem/Desktop/esempi,** frammenti di configurazioni

Cosa 'costruiremo'



- Flussi di comunicazione sicuri (https, tls, etc..)
- Apache: Location /idp attiva proxy su 8009 e un Connector di Tomcat redirige su 8443 (loopback)
- Server LDAP puo' essere interno o esterno allo host

Agenda

- Installazione dei pacchetti sofware
  - Apache 2.2 & ntp (via yum)
  - Java (jre6), Tomcat 6, Shibboleth 2.1 (manuale)
- Configurazione

  Apache, Tomcat, Shibboleth (attributi, metadati..)
- Test

Agenda

Installazione

Installazione – Certificato

```
#> su -
(tutorial)
#> cd /etc/pki/tls/certs
#> openssl req -newkey rsa:1024 -x509 -nodes \
   -out tutorialXX.pem -keyout tutorialXX.key \
   -days 365
(inserite le info richieste)
- [Country] IT
- [Province] Puglia
- [City ]Bari
- [Organization] IDEM
- [Section] Tutorial
- [CN] tutorialXX.poliba.it
- [email] ...
```

Lavoreremo con un certificato auto-firmato. **(questo e' MALE®)**

Installazione – Ldap TLS

Attiviamo TLS su LDAP

```
#> vi /etc/openldap/slapd.conf
   Decommentiamo le righe relative a TLS (30-34)
   Modifichiamo i nomi dei file certificato e
   chiave  sostituendo le XX
(ESC :wq)

#> service ldap restart
```

Installazione – Java


```
#> cd /home/idem/Desktop/src/
#> wget -O ./jre.rpm.bin <UrlDownload>[1]
#> chmod +x ./jre.rpm.bin
#> ./jre.rpm.bin
```

Scarichiamo e installiamo la Java VM.
Non e' possibile utilizzare i pacchetti presenti nei repo ufficiali (gcj, openjdk)

1 - <UrlDownload>
- Fra i boomark del browser nella vostra VM
- http://javadl.sun.com/webapps/download/AutoDL?BundleId=37391 (puo` variare)
- Oggi lavoriamo con i sorgenti presenti nella directory /home/idem/Desktop/src

Installazione – Java


```
#> touch /etc/profile.d/idp.sh
#> chmod +x /etc/profile.d/idp.sh
#> cat >>/etc/profile.d/idp.sh
export JAVA_HOME=/usr/java/default
export JAVA_OPT=”-server”
(ctrl+d)
#> source /etc/profile
```

Creiamo le variabili d'ambiente necessarie alla JRE

Il comando 'source' serve a renderle attive nella sessione di bash corrente

Installazione – Apache

```
#> yum install httpd mod_ssl
#> chkconfig httpd on
```

NB:L'installazione del pacchetto e` gia' stata effettuata sulla vostra VM
Eseguiamo il comando #2

Installazione – Tomcat 1/2

```
#> wget <UrlDownload>[1]
#> tar xzvf ./apache-tomcat-6.x.y.tar.gz \\
   -C /usr/local
#> ln -s /usr/local/apache-tomcat-6.x.y  \\
   /usr/local/apache-tomcat

#> cat >> /etc/profile.d/idp.sh
export CATALINA_HOME=/usr/local/apache-tomcat
(ctrl+d)
#> source /etc/profile
```

1 - <UrlDownload>

- Fra i boomark del browser nella vostra VM
- http://mirror.nohup.it/apache/tomcat/tomcat-6/v6.0.24/bin/apache-tomcat-6.0.24.tar.gz (puo` variare)
- Oggi Lavoriamo con i sorgenti presenti nella directory /home/idem/Desktop/src

Installazione – Tomcat


```
#> useradd tomcat -d /usr/local/apache-tomcat
#> chown -R tomcat:tomcat \
/usr/local/apache-tomcat-6.0.24
```

Creiamo utente responsabile del run di tomcat e assegnamovi la proprieta` dei file

```
#> vi $CATALINA_HOME/conf/tomcat-users.xml
..
<tomcat-users>
   <role rolename="admin"/>
   <role rolename="manager"/>
   <user username=”admin” password=”tutorial” roles=”admin,manager”/>
</tomcat-users>
..
(ESC & :wq)
```

Modifichiamo le righe indicate per aggiundere un utente di amministrazione di tomcat

Installazione – Shibboleth 1/3

```
#> wget <UrlDownload>[1]
#> unzip shibboleth-identityprovider-
2.1.5.bin.zip
#> cd shibboleth-identityprovider-2.1.5
#> sh install.sh
(inseriamo i parametri richiesti)
1 - /usr/local/shibboleth-idp    (directory target)
2 - tutorialXX.poliba.it         (hostname)
3 - tutorial                     (password)
```

1 - <UrlDownload>

- Fra i boomark del browser nella vostra VM
- http://shibboleth.internet2.edu/downloads/shibboleth/idp/latest/shibboleth-identityprovider-2.1.5-bin.zip (puo` variare)
- Oggi lavoriamo con i sorgenti presenti nella directory /home/idem/Desktop/src

Installazione – Shibboleth


```
#> chown tomcat:tomcat /usr/local/shibboleth-idp -R

#> vi $CATALINA_HOME/conf/catalina.properties
..
common.loader=PATH1,PATH2,/usr/local/shibboleth-idp/lib/endorsed/*.jar
..

(ESC & :wq)
```

Assegnamo all'utente tomcat la proprieta' dei file di shibboleth.

Appendiamo il percorso delle classi endorsed in cosa a quelle già definite nel common loader di tomcat utilizzando un carattere ”,” per concatenarlo a quelli presenti.

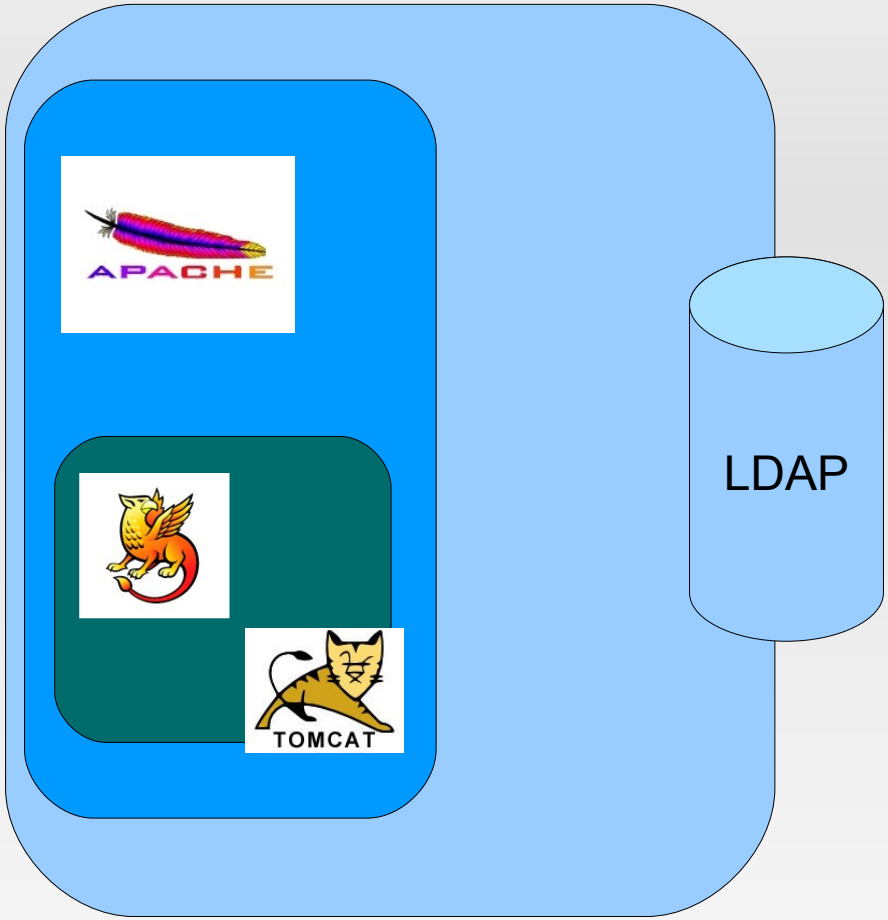
Installazione – Shibboleth

Effettuiamo il deploy dell'applicazione copiando il Web Application Archive fra le webapps di tomcat

```
#> cp /usr/local/shibboleth-idp/war/idp.war   \\
   /usr/local/apache-tomcat/webapps
```

NB: esistono diversi modi per il deply dell'applicazione. Si può utilizzare anche il tool web di tomcat alla pagina http://tutorialxx.poliba.it:8080 , link tomcat manager. (Accedere utilizzando l'utente admin con password tutorial)

A che punto siamo?



Host

Apache

Tomcat

- Abbiamo installato quello che ci serve.
- Ora possiamo configurarlo

Agenda

Configurazione

Configurazione – Apache 1/4

Creare un copia di backup del file **/etc/httpd/conf.d/ssl.conf** e modifichiamolo

```
#> cd /etc/httpd/conf.d
#> cp ssl.conf ssl.conf.orig
#> vi ssl.conf

   (eliminare tutta la sezione compresi i tag)
   <VirtualHost _default_:443>
   ..
   ..
   </VirtualHost>

(ESC & :wq)
```

Configurazione – Apache 2/4

Creare il file **/etc/httpd/conf.d/vhosts.conf** e aggiungervi il seguente frammento: (potete copiare e adattare il template nel file **~/Desktop/esempi/vhosts.example**

```
<VirtualHost tutorialxx.poliba.it:443>
    SSLEngine on
    SSLProtocol all -SSLv2
    SSLCipherSuite ALL:!ADH:!EXPORT:!SSLv2:RC4+RSA:+HIGH:+MEDIUM:+LOW
    SSLOPtions +StdEnvvars

    SSLCertificateFile "/etc/pki/tls/certs/tutorialXX.pem"
    SSLCertificateKeyFile "/etc/pki/tls/certs/tutorialXX.key"

    ErrorLog logs/idp_error.log
    TransferLog logs/idp_access.log
    LogLevel warn
</VirtualHost>
```

Configurazione – Apache 3/4

```
<VirtualHost tutorialXX.poliba.it:8443>
    SSLEngine on
    SSLProtocol all -SSLv2
    SSLCipherSuite ALL:!ADH:!EXPORT:!SSLv2:RC4+RSA:+HIGH:+MEDIUM:+LOW
    SSLOPtions +StdEnvvars +ExportCertData
    SSLVerifyClient optional_no_ca
    SSLVerifyDepth 10

    SSLCertificateFile "/usr/local/shibboleth-idp/credentials/idp.crt"
    SSLCertificateKeyFile "/usr/local/shibboleth-idp/credentials/idp.key"

    ErrorLog logs/idp_error.log
    TransferLog logs/idp_access.log
    LogLevel warn
</VirtualHost>
```

Configurazione – Apache 4/4

Focus sui certificati

- **Front Channel**: certificato firmato da un CA riconosciuta dalla federazione (cfr. Documento Specifiche Tecniche)
  - **`SSLCertificateFile:`** file contentente il certificato
  - **`SSLCertificateKeyFile:`** file contenente la chiave privata
  - **`SSLCertificateChainFile:`** file 'bundle' contenente i certificati delle CA intermedie fino alla root CA
    - **`Cybertrust Educational CA (sureserverEDU.pem)`**
    - **`GTE CyberTrust Solutions (ct_root.pem)`**

```
#> cat sureserverEDU.pem ct_root.pem >> myCertChain.pem
```

- **Back Channel**: certificato auto-firmato generato da shibboleth durante l'installazione

Configurazione – Tomcat

Aggiungere nel file **/etc/httpd/conf.d/proxy_ajp.conf:**

```
ProxyPass /idp/ ajp://localhost:8009/idp/
ProxyPassReverse /idp/ ajp://localhost:8009/idp/
```

Configurare un connettore modificando **/usr/local/apache-tomcat/conf/server.xml**

```
<Connector
    port="8009"
    request.tomcatauthentication="false"
    address="127.0.0.1"
    enableLookup="false"
    protocol="AJP/1.3"
    redirectPort="8443"
/>
```

Configurazione

Rendiamo effettive le modifiche ai servizi..

```
#> service httpd restart
#> usr/local/apache-tomcat/bin/startup.sh
```

… verificando che siano state apportate correttamente

NB: sara' necessario configurare tomcat affinche' si avvii allo startup

Configurazione – Shibboleth

Panoramica sui file su cui lavoreremo

- **login.config:** Java Authentication & Authorization Service
- **handler.xml:** come l'IdP risponde ai vari messaggi ricevuti
- **attribute-resolver.xml:** recupero, manipolazione ed encoding degli attributi
- **attribute-filter.xml:** cosa rilasciare agli SP
- **relying-party.xml:** configuraz. specifiche & metadati

Configurazione – Shibboleth

Focus su LDAP

```
#> ldapsearch -x -ZZ
#tutorial.idem
dn: dc=tutorial,dc=idem
dc: tutorial
objectClass: domain
# people, tutorial.idem
dn: ou=people,dc=tutorial,dc=idem
ou: people
objectClass: organizationalUnit
# idem01, people, tutorial.idem
dn: uid=idem01,ou=people,dc=tutorial,dc=idem
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
objectClass: eduPerson
objectClass: top
cn: Utente Test
eduPersonOrgDN: uid=idem01,ou=people,dc=tutorial,dc=idem
eduPersonOrgUnitDN: ou=people,dc=tutorial,dc=idem
eduPersonPrincipalName: idem01@tutorial.idem
eduPersonScopedAffiliation: staff@tutorial.idem
givenName: Utente
mail: idem01@tutorial.idem
sn: Test
uid: idem01
userPassword:: aWRlbWlkZW0=
```

Albero molto semplice.

- Base DN: dc=tutorial,dc=idem
- 1 OU: ou=people
- 1 entry: uid=idem01

NB: attivare il TLS lato client inserendo il certificato in /etc/openldap/ldap.conf oppure elliminare l'opzione -ZZ dalla linea di comando

Configurazione – Shibboleth

Importiamo il certificato di LDAP nel keystore di default di JAVA

```
#> /usr/java/default/bin/keytool -import -trustcacerts \
   -alias "myldap" \
   -file /etc/pki/tls/certs/tutorialXX.pem \
   -keystore /usr/java/default/lib/security/cacerts

   (password: changeit)
   (rispondere "si" alla richiesta di conferma)
```

Configurazione – Shibboleth

Nel file **/usr/local/shibboleth-idp/conf/login.config** decommentare la seguente sezione e modificare. Istruiremo il JAAS con i parametri del nostro server LDAP

```
/* (uncomment)
   edu.vt.middleware.ldap.jaas.LdapLoginModule required
      host="tutorialXX.poliba.it" # o altro, se esterno
      port="389"
      base="dc=tutorial,dc=idem"
      tls="true"
      userField="uid"
      subtreeSearch="true"
      serviceUser="cn=manager,dc=tutorial,dc=idem"
      serviceCredential="tutorial";

*/
```

Configurazione – Shibboleth

Attivare il metodo di login *'UsernamePassword'* nella sezione *'Login Handlers'* del file **/usr/local/shibboleth-idp/conf/handler.xml.** Attiviamo cosi' JAAS appena configurato

```
<!-- Login Handlers -->
<!-- (comment out)
<LoginHandler xsi:type="RemoteUser">
    <AuthenticationMethod>
        urn:oasis:names:tc:SAML:2.0:ac:classes:unspecified
    </AuthenticationMethod>
</LoginHandler>
-->


<!--  Username/password login handler -->
<!-- (uncomment)
<LoginHandler xsi:type="UsernamePassword"
    jaasConfigurationLocation="file:///usr/local/shibboleth-idp/conf/login.config">
    <AuthenticationMethod>
        urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport
    </AuthenticationMethod>
   </LoginHandler>
-->
```

Configurazione – Shibboleth

Focus su **attribute-resolver.xml**

- **<resolverDataConnector> :** definiscono le sorgenti degli attibuti
  - **static**
  - **computed**
  - **stored ID**
  - **relational database**
  - **LDAP**
- **<resolverAttributeDefinition>:** definiscono i tipi e le mappature con la sorgente
  - **simple**
  - **scoped**
  - **computed**
  - **...altri 9...**

Configurazione – Shibboleth

/usr/local/shibboleth-idp/conf/attribute-resolver.xml

```
<resolver:DataConnector
      id="myLDAP"
      xsi:type="LDAPDirectory"
      xmlns="urn:mace:shibboleth:2.0:resover:dc"
      ldapURL="ldap://tutorialXX.poliba.it
      baseDN="dc=tutorial,dc=idem"
      principal="cn=manager,dc=tutorial,dc=idem"
      principalCredential="tutorial"
      useStartTLS="true">
   <FilterTemplate>
      <![CDATA[(uid=$requestContext.principalName)]]>
   </FilterTemplate>

</resolver:DataConnector>
```

Security Hint #2: vedi #1

Configurazione – Shibboleth

/usr/local/shibboleth-idp/conf/attribute-resolver.xml

Analizziamo la configurazione di esempio: /home/idem/Desktop/esempi
e copiamola nella directory di configurazione di shibboleth

```
#> cd /usr/local/shibboleth-idp/conf/
#> mv attribute-resolver.xml attribute-resolver.xml.dist
#> cp /home/idem/Desktop/esempi/attribute-resolver.example \
./attribute-resolver.xml
#> chown tomcat:tomcat ./attribute-resolver.xml
```

Configurazione – Shibboleth

/usr/local/shibboleth-idp/conf/attribute-filter.xml

Un frammento d'esempio:

```
</AttributeFilterPolicy id=”releaseToAnyone”>

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

</AttributeFilterPolicy>
```

Configurazione – Shibboleth

/usr/local/shibboleth-idp/conf/attribute-filter.xml

Analizziamo la configurazione di esempio: /home/idem/Desktop/esempi
e copiamola nella directory di configurazione di shibboleth

```
#> cd /usr/local/shibboleth-idp/conf/
#> mv attribute-filter.xml attribute-filter.xml.dist
#> cp /home/idem/Desktop/esempi/attribute-filter.example \
./attribute-filter.xml
#> chown tomcat:tomcat ./attribute-filter.xml
```

Configurazione – Shibboleth

/usr/local/shibboleth-idp/conf/relying-party.xml

```
<MetadataProvider id="idem"
        xsi:type="FileBackedHTTPMetadataProvider"
        xmlns="urn:mace:shibboleth:2.0:metadata"
        metadataURL="https://www.idem.garr.it/docs/conf/signed-metadata.xml"
        backingFile="/usr/local/shibboleth-idp/metadata/idem-metadata.xml">

    <MetadataFilter xsi:type="ChainingFilter"
            xmlns="urn:mace:shibboleth:2.0:metadata">

    <MetadataFilter xsi:type="SignatureValidation"
            xmlns="urn:mace:shibboleth:2.0:metadata"
            trustEngineRef="shibboleth.MetadataTrustEngine"
            requireSignedMetadata="true" />
     </MetadataFilter>

</MetadataProvider>
```

Configurazione – Shibboleth

/usr/local/shibboleth-idp/conf/relying-party.xml

Analizziamo la configurazione di esempio: /home/idem/Desktop/esempi
e copiamola nella directory di configurazione di shibboleth

```
#> cd /usr/local/shibboleth-idp/conf/
#> mv relying-party.xml relying-party.xml.dist
#> cp /home/idem/Desktop/esempi/relying-party.example \
./relying-party.xml
#> chown tomcat:tomcat ./relying-party.xml
```

NB: i metadati hanno bisogno di essere verificati tramite il certificato signer_bundle.pem che e' stato preventivamente scaricato nella directory /etc/pki/tls/cacerts dalla pagina

https://www.idem.garr.it/index.php/it/informazioni-tecniche/certificati

Configurazione – Shibboleth

/usr/local/shibboleth-idp/metadata/idp-metadata.xml
Analizziamo il file:

- E' la carta d'identita' del nostro IdP
- Contiente informazioni di sicurezza (certificati)
- Consente l'interazione coni altri attori della federazione

Il frammento deve essere inviato (tramite mezzo sicuro, es mail firmata) alla federazione per essere inserito nei metadati ufficiali di IDEM

Configurazione – Shibboleth

Compiti per casa:

Modificare il file
/usr/local/apache-tomcat/webapps/idp/web.xml
per abilitare la pagina di login web.

NB: il file viene creato da Tomcat che 'esplode' il file idp.war

Agenda

Test

Test

Test #1 - funzionale: successo nell'installazione

Digitate nel browser:

https://tutorialxx.poliba.it/idp/profile/status

Se l'installazione di shibboleth e' avvenuta correttamente, nel browser apparira' **OK**

Test

Test #2 - attributi

Utilizzare l'utility **aacli.sh** (Attribute Authority, Command Line Interface) per verificare il corretto processo di prelievo e rilascio degli attributi

```
#> cd /usr/local/shibboleth-idp
#> sh bin/aacli.sh --configDir=conf -principal=idem01 \
      --requester=https://sp-test.garr.it/secure
```

Test

Test #3: www.testshib.org

Registrarsi al servizio: http://www.testshib.org/testshib-two/join.jsp

- Modificare il file **relying-party.xml** come descritto in http://www.testshib.org/testshib-two/configure.jsp
- Aggiungere una policy in attribute-filter.xml per rilasciare al https://sp.testshib.org/shibboleth-sp gli attributi:
  - eduPersonScopedAffiliation
  - EduPersonTargetedID
- Riavviare l'IdP (restart Tomcat)
- Recarsi alla pagina https://sp.testshib.org, inserire il nome DNS del proprio IdP ed effettuare il login con account di test

NB: l'IdP deve avere un indirizzo pubblico, deve essere registrato nel DNS e le porte TCP (443, 8443) devono essere aperte

Riferimenti

- Documenti tecnici della federazione
  - Specifiche tecniche
  - Specifiche tecniche per la compilazione e uso degli attributi
  - Howtos
- Documentazione varia
  - Documentazione ufficiale di Shibboleth 2
  - FAQ di OpenLDAP su TLS/SSL
  - Shibboleth IdP Test Installation at the University of Canterbury
  - Shibboleth 2 IdP Setup Guide (UK Federation)

Andate...
.. e federatevi!