

Shibboleth SP

installazione e configurazione di base per SSO

Agenda

- Verifica VM – istruzioni operative
- Intro sui service provider
- Installazione
- Configurazione
- Protezione di una risorsa web

Istruzioni operative

Visualizzazione pagine web

- *Browser* (si può usare quello Host oppure quello della VM)

Configurazione shibboleth/apache/etc..

- Copia dei file da terminale (`sudo -i`)
- Edit dei file (`gedit`, `vi`, etc)

Materiale per configurazione in

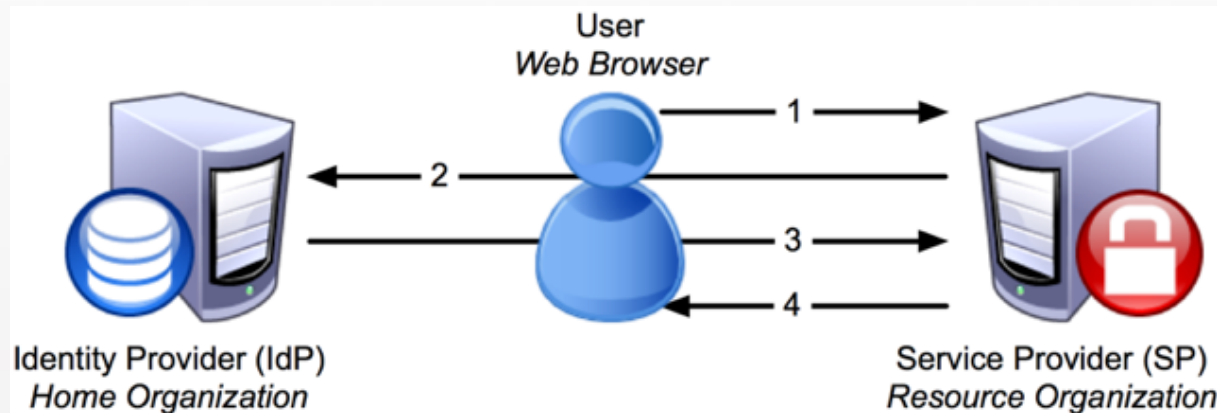
`/home/testuser/CORSO_IDEM`

Agenda

Intro sui service provider

- Flusso informazioni
- Relazioni di Trust
- Framework SAML

Flusso delle informazioni



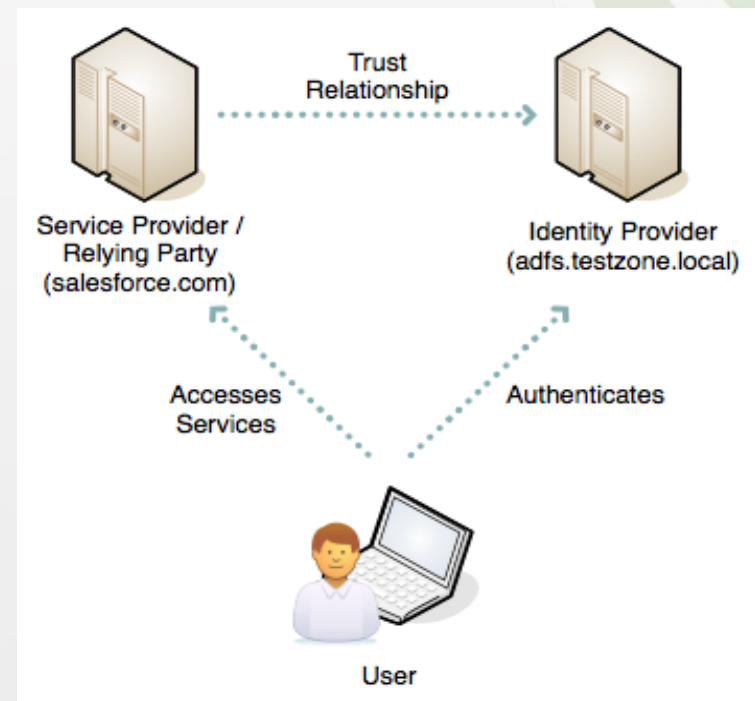
1. The SP detects the user attempting to **access restricted content** within the resource.
2. The SP generates an **authentication request**, then sends the request, and the user, to the user's IdP.
3. The **IdP authenticates** the user, then sends the **authentication response**, and the user, back to the SP.
4. The **SP verifies the IdP's response** and sends the request through to the resource

Trust Relationship

Nelle applicazioni legacy il trust è implicito,
tutte le funzioni sono interne alla applicazione stessa

Con il paradigma SP – IdP ci deve essere
una relazione di fiducia fra SP e IdP

La fiducia reciproca fra IdP ed SP
si ottiene attraverso i **metadati** ed i
certificati



Trust Relationship - metadati

Esempio di metadati

```
<EntitiesDescriptor ID="IDEM"
```

```
.....
```

```
<ds:X509Certificate>
```

```
MIIDnzCCAoeg.....
```

```
</ds:X509Certificate>
```



```
<EntityDescriptor entityID="https://IDP.example.it">
```

```
.....
```

```
<ds:X509Certificate>
```

```
MIIDEzCCAfugAwIBAg.....
```

```
</ds:X509Certificate>
```



```
<EntityDescriptor entityID="https://SP.example.it">
```

```
.....
```

```
<ds:X509Certificate>
```

```
MQswCQY.....
```

```
</ds:X509Certificate>
```



Trust Relationship - asserzioni

Esempio di asserzione

```
<saml2p:Response Destination="https://filesender.garr.it" .....>
<saml2:Issuer>https://idp.uniparthenope.it</saml2:Issuer>
<ds:Signature...>
...
<ds:X509Certificate>
    MIIDOzCCAiO...
</ds:X509Certificate>

<saml2:EncryptedAssertion>
.....
<ds:X509Certificate>
    MIIEZTCCA02gAw....
</ds:X509Certificate>

<xenc:CipherValue>
UKuw3E7+YXdLrVUFjy+eFFf5aQRDjdAvdiYNZd51823d
YDo734pJiaKSWQgfZLLxeVHVC5SwwLK2axe6rZFZpm/a
```


Framework SAML

Shibboleth SP - modulo di Apache/IIS, utilizzabile con tutti i linguaggi di programmazione supportati dal webserver.

SimpleSAMLphp - implementazione PHP nativa Integrabile direttamente nella nostra applicazione (PHP)

Elenco dei software

http://en.wikipedia.org/wiki/SAML-based_products_and_services

Estensioni SAML

Le applicazioni più note hanno dei plug-in SAML già disponibili



<https://wiki.shibboleth.net/confluence/display/SHIB2/ShibEnabled>

<https://rnd.feide.no/federated-software/>

Agenda

Intro sui service provider

Installazione

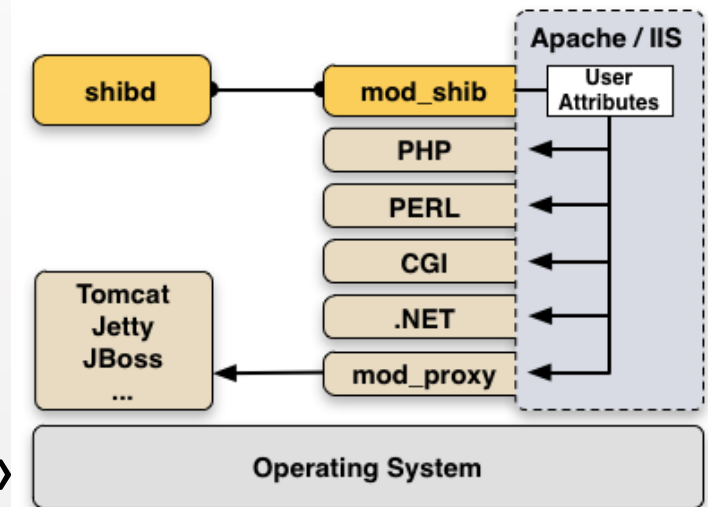
Configurazione

Proteggere risorsa web

Shibboleth SP

Shibboleth Service Provider è composto da

- mod_shib (Apache /IIS)
- Demone SHIBD



Caratteristiche

- Proteggere l'accesso con «Require»
- Attributi utente accessibili nell'ambiente del web server da tutte le applicazioni (PHP, Perl, .Net, ASP, CGI, ...) es. `$_SERVER['mail']`.
- Servlet container, (es. Tomcat) devono operare con Apache or IIS come front-end

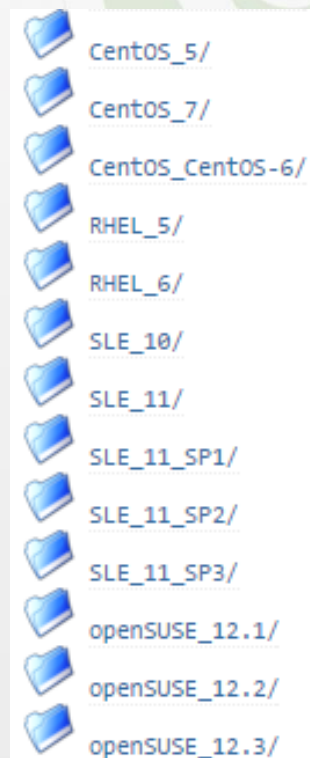
Installazione

- pacchetti per Debian/Ubuntu

```
Sudo apt-get install apache2 libapache2-mod-shib2
openssl php5 ntp
```

- distribuzioni RPM based

<http://download.opensuse.org/repositories/security://shibboleth/>



Struttura delle directory

- **Modulo Apache**
 - `/etc/apache2/mod-available/shib2.load`
 - `/usr/lib/apache2/modules/mod_shib2.so`
- **Demone**
 - `/usr/bin/`
 - `/usr/lib/x86_64-linux-gnu/shibboleth/*`
 - `/usr/sbin/shib-keygen`
 - `/usr/sbin/shibd`
- **CFG**
 - `/etc/shibboleth/*`
- **Elenco files**

Comando : `dpkg -L libapache2-mod-shib2`

<http://packages.ubuntu.com/trusty/amd64/libapache2-mod-shib2/filelist>

Agenda

Intro sui service provider

Installazione

Configurazione

Proteggere risorsa web

Agenda

Configurazione - certificati

- Scarichiamo il certificato per la verifica metadati
- Generiamo il certificato del nostro service provider
- *Nota: Il certificato usato dal webserver per l'HTTPS è completamente indipendente da questo.*

Configurazione - certificati

Salvare i certificati per la **VERIFICA** del file metadati

- **Federazione IDEM**

Wget

https://www.idem.garr.it/documenti/doc_download/321-idem-metadata-signer-2019

-O /etc/shibboleth/metadata-signer.crt

- ***Questo corso***

Wget <https://sp.lab.unimo.it/metadata-signer.crt>

-O /etc/shibboleth/metadata-signer.crt

Configurazione - certificati

Creare i certificati per la **firma** e **cifratura** delle asserzioni

- Generazione certificati

`/usr/sbin/shib-keygen`

`/etc/shibboleth/sp-cert.pem`

`/etc/shibboleth/sp-key.pem`

- *Solo per il corso*

```
cp /home/testuser/CORSO_IDEM/1_SESSIONE/shibboleth/sp-*.pem  
/etc/shibboleth
```

Agenda

Configurazione -shibd

- Set *entityID*
- Set *SSO*
- Set *metadata provider*

Configurazione – shibboleth2.xml -1/3

Editing di /etc/shibboleth/shibboleth2.xml

entityID

Sostituire

```
<ApplicationDefaults entityID="https://sp.example.org/shibboleth"  
REMOTE_USER="eppn persistent-id targeted-id">
```

con

```
<ApplicationDefaults entityID="https://sp1.local/shibboleth"  
REMOTE_USER="eppn persistent-id targeted-id">
```

Configurazione – shibboleth2.xml -2/3

Editing di /etc/shibboleth/shibboleth2.xml

SSO (caso singolo IdP)

Sostituire

```
<SSO entityID="https://idp.example.org/shibboleth"  
  discoveryProtocol="SAMLDS"  
discoveryURL="https://ds.example.org/DS/WAYF">  
SAML2 SAML1  
</SSO>
```

Con

```
<SSO entityID="https://idp-corso.ircs.garr.it/idp/shibboleth"  
  discoveryProtocol="SAMLDS"  
discoveryURL="https://ds.example.org/DS/WAYF">  
SAML2 SAML1  
</SSO>
```

Configurazione – shibboleth2.xml -3/3

Editing di /etc/shibboleth/shibboleth2.xml

MetadataProvider

Inserire il seguente <MetadataProvider>:

```
<MetadataProvider type="XML"
uri="https://sp.lab.unimo.it/rr3/signedmetadata/federation/fed-
corso/metadata.xml"
backingFilePath="/etc/shibboleth/signed-test-metadata.xml"
reloadInterval="7200">
<MetadataFilter type="Signature" certificate="metadata-signer.crt"/>
</MetadataProvider>
```

Configurazione – check & restart

- Verifichiamo la configurazione

```
Shibd -t
```

- Riavviamo il servizio

```
Services shibd restart
```

- Attiviamo il modulo shib2

```
a2enmod shib2
```

```
service apache2 restart
```

I nostri metadati

Rivediamo i nostri metadati

<https://sp1.local/Shibboleth.sso/Metatada>

Analizziamo i vari TAG

<Request Initiator>

<Assertion consumer service>

Agenda

Intro sui service provider

Installazione

Configurazione

Proteggere risorsa web



Protezione di una risorsa Locale

Pagina da proteggere

<https://sp1.local/intranet/intranet.html>

Accesso libero

Configurazione - Apache

- configurazione della Location da proteggere

Modificare /etc/apache2/sites-available/service_provider.conf

```
<Location /intranet>  
    AuthType shibboleth  
    ShibRequestSetting requireSession true  
    Require shib-session  
</Location>
```

- Attivare il modulo shib2

```
service apache2 restart
```

Protezione di una risorsa Locale

VERIFICA

<https://sp1.local/intranet/intranet.html>

Accesso previo autenticazione

Sistemiamo le cose.....

Lanciamo lo script di update per allineare tutte le VM

```
cd /home/testuser  
./CORSO_IDEM/1_SESSIONE/update_stato_1.sh
```

Speriamo vi sia piaciuto....

.....e tutto abbia funzionato!!!

