Práctica Seguridad 2013
Leandro J Guillén Moreno

Perfiles SAML

Práctica 1 de Seguridad 2012-13

Leandro J. Guillén Moreno 48510228P leandrojesus.guillen@um.es

Índice

1. Diseño

El escenario se compone de tres entidades (ver figura 1):

- Service Provider.
- Service Provider 2.
- Identity Provider.

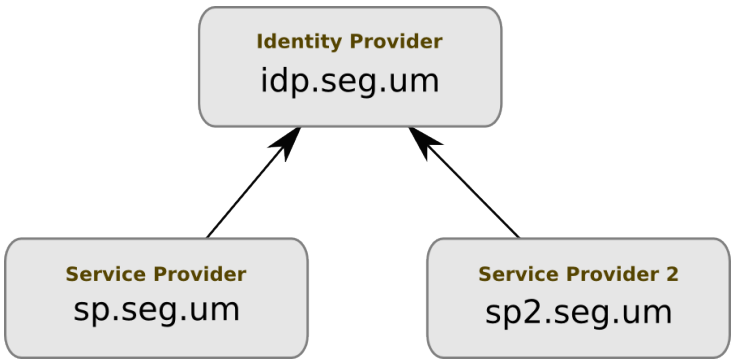

Figura 1: Escenario de la práctica y relaciones de confianza.

Cada una de las máquinas ha sido montada en una máquina virtual con VirtualBox (ver figura 2). Es requisito indispensable que los nombres de dominio indicados en la figura 1 sean resolubles en el cliente. Por ejemplo, configurándolos en el fichero */etc/hosts* en linux.

El entorno de ejecución es un contenedor Apache Tomcat, utilizando la tecnología Java EE para cada sitio. Un servlet es el encargado de recibir las peticiones y realizar la lógica del sitio.

Las redirecciones en todas las entidades se realizan con un POST.

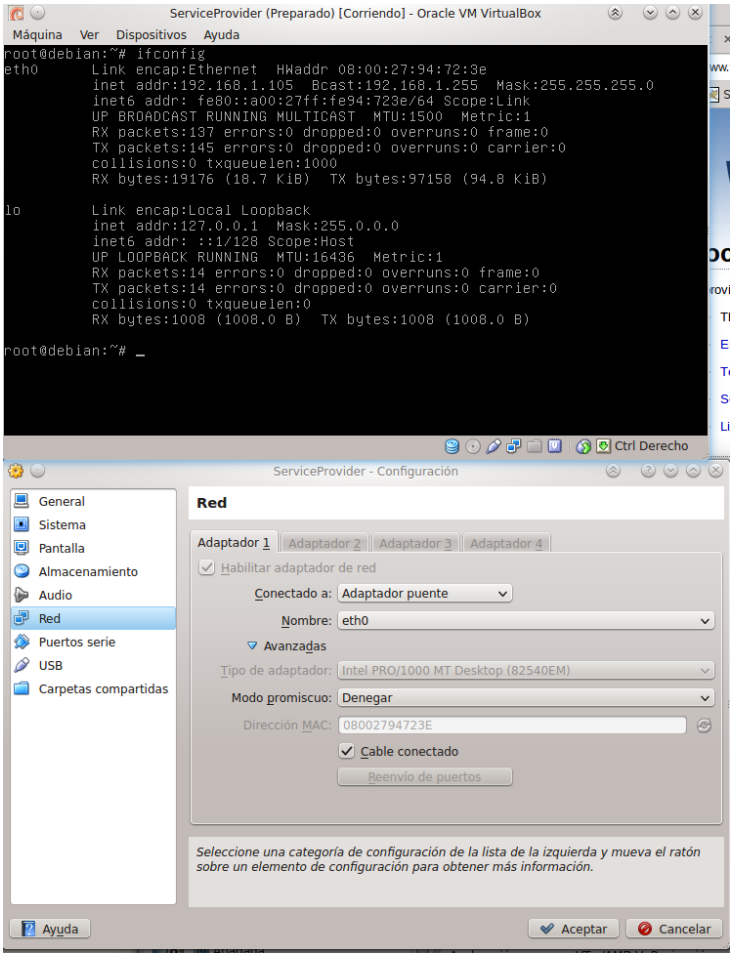

Figura 2: Maquina virtual ejecutandose en modo bridge.

2. Intercambios entre entidades

Para mostrar un ejemplo de intercambio entre entidades he realizado tres pruebas:

1. El cliente pide un recurso al Service Provider y no está autenticado.
2. El cliente pide un recurso al Service Provider y está autenticado.
3. El cliente pide un recurso al Service Provider 2 y está autenticado.

Las pruebas se han realizado de manera consecutiva. La correspondencia de las direcciones IP es la siguiente:

Entidad	IP
Cliente	192.168.1.12
Service Provider	192.168.1.140
Service Provider 2	192.168.1.134
Identity Provider	192.168.1.135

Primer intercambio

En la figura 3 se ve el intercambio número uno. Este consiste en la petición de un recurso al Service Provider sin que el cliente esté autenticado.

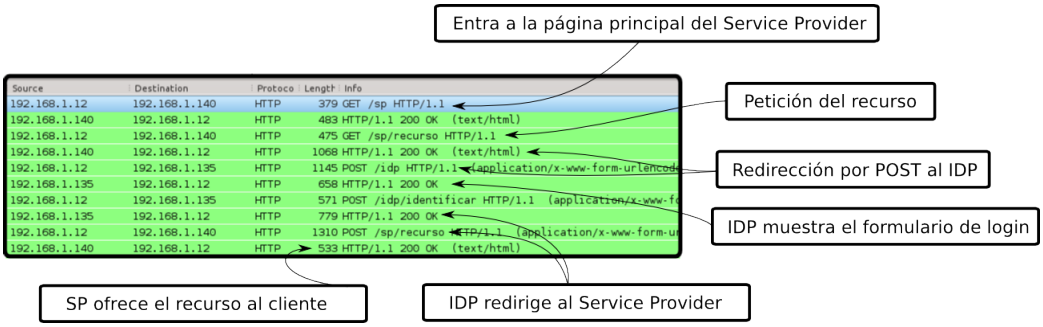

Figura 3: Petición inicial del recurso.

A continuación describo con detalle el intercambio:

1. El cliente entra en la página principal del Service Provider, http://sp.seg.um/sp.
2. El cliente solicita un recurso protegido.
3. El Service Provider ve que la sesión del cliente no está autenticada, por lo que emite una página que le redirige al Identity Provider (por el método POST) junto con un Authentication Request.
4. El cliente reenvía la petición al Identity Provider.
5. El Identity Provider responde al cliente mostrándole el formulario de autenticación, donde el cliente introduce su usuario y contraseña.
6. El Identity Provider autentica correctamente al usuario y le redirige al recurso solicitado del Service Provider enviando un Response.
7. El Service Provider recibe el Response y ve que el usuario fue autenticado correctamente. A partir de ahora, el cliente tiene acceso a los recursos.
8. El Service Provider ofrece el recurso al cliente.

Segundo intercambio

El segundo intercambio (figura 4) consiste en una petición al Service Provider estando ya autenticado el cliente:

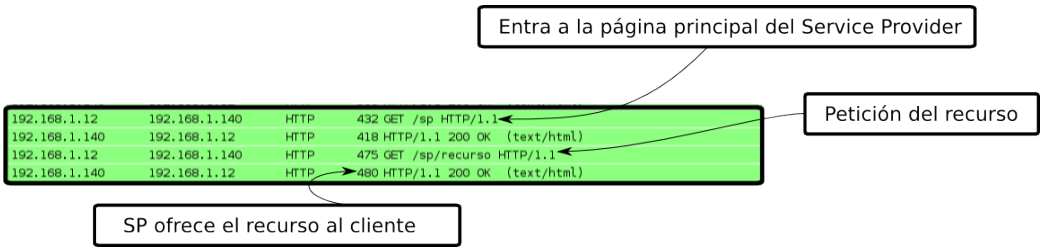

Figura 4: Petición del recurso ya autenticado.

1. El cliente entra en la página principal.
2. El cliente solicita el recurso.
3. El Service Provider recibe los datos de la sesión del cliente y ve que ya está autenticado en el Identity Provider.
4. El Service Provider ofrece el recurso directamente al cliente.

Tercer intercambio

El tercer intercambio (figura 5) muestra al cliente accediendo a un recurso en otro Service Provider (llamado Service Provider 2).

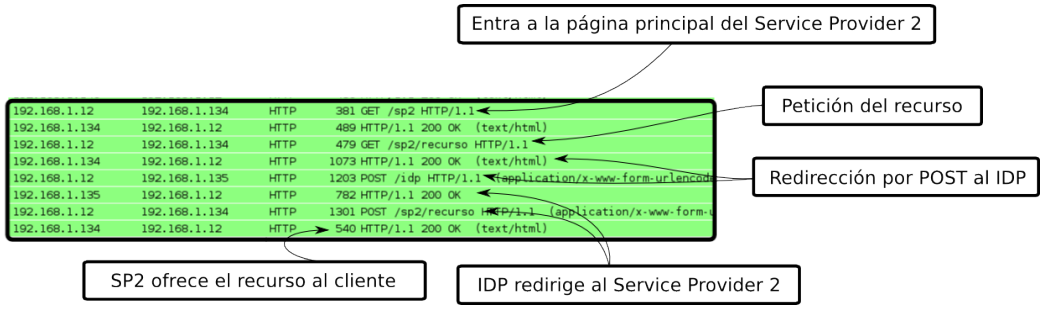

Figura 5: Petición de un recurso en otro Service Provider ya autenticado.

1. El cliente, tras acceder a la página principal (http://sp2.seg.um/sp2) solicita acceso al recurso protegido.

2. El cliente es redirigido por POST al Identity Provider para autenticación, ya que éste no está autenticado en el sistema propio.
3. El Identity Provider identifica al cliente automáticamente y le considera autenticado.
4. Reenvía un Response al Service Provider 2.
5. Service Provider 2 devuelve el recurso.

Figura 6: Página inicial del Service Provider 2 sin el cliente autenticado.

3. Despliegue

A continuación se detallan los pasos seguidos para desplegar el escenario:

1. Crear tres máquinas virtuales.
    - Instalar Tomcat 7 y Oracle Java 7.
    - No hace falta configurar nada a nivel de red, tan solo saber la IP asignada en caso de modo puente, o el puerto para la traducción en modo NAT.
    - Instalar el paquete tomcat-manager para gestionar aplicaciones con interfaz web.
2. Desplegar sitios web.
    - Accediendo al manager de Tomcat.
    - Se suben y despliegan los archivos .war de cada aplicacion en cada máquina virtual.
3. Configurar el cliente para que los nombres de dominio (descritos en la figura 1) sean resolubles.
4. Probar el escenario entrando en http://sp.seg.um/sp, por ejemplo. Se pueden seguir los pasos descritos en la sección 2.

El código está accesible en un repositorio público en Github: `https://github.com/LeandroGuillen/SAML.git`.

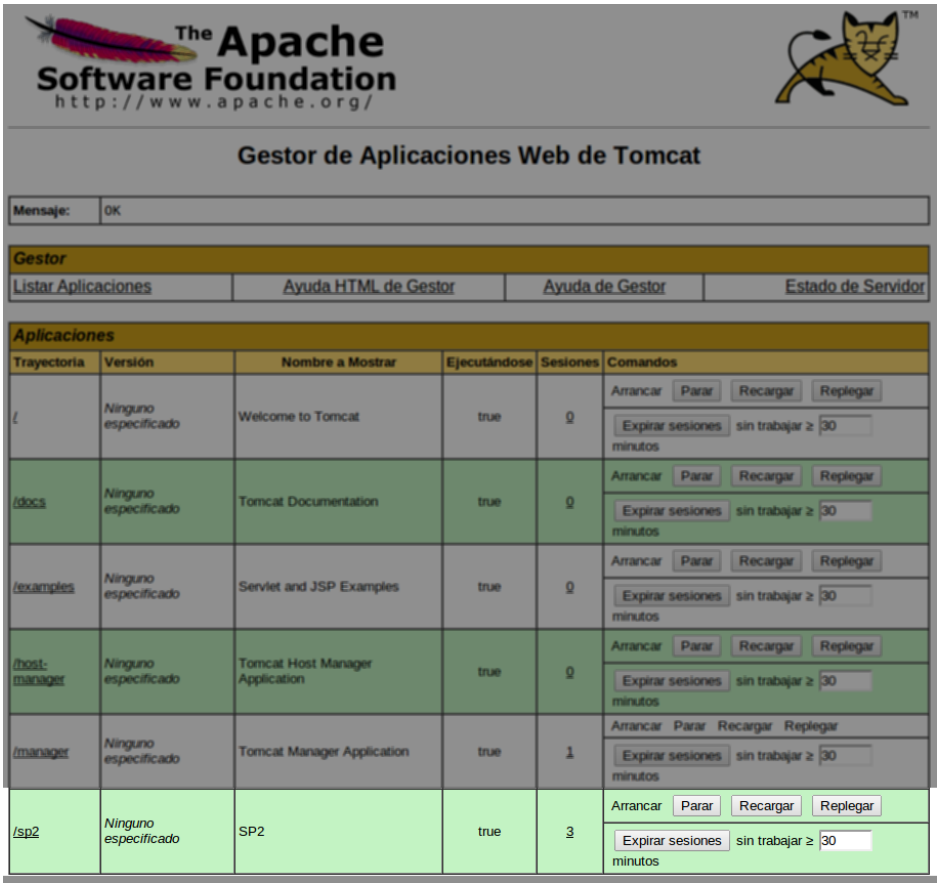

Figura 7: Tomcat tiene desplegada la aplicacion *sp2* (Service Provider 2).

Referencias

[1] Suresh Attanayake. How To Read A SAML 2.0 Response With OpenSAML. `http://sureshatt.blogspot.com.es/2012/11/how-to-read-saml-20-response-with.html`, 2012.

[2] MSDN. SAMLWriter Class.

[3] OASIS. Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0. `https://www.oasis-open.org/committees/download.php/35711/sstc-saml-core-errata-2.0-wd-06-diff.pdf`, 2005.

[4] OASIS. Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0. `http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf`, 2005.

[5] Oracle. Java EE API Documentation. http://docs.oracle.com/javaee/6/api/.

[6] Oracle. VirtualBox Documentation. https://www.virtualbox.org/wiki/Documentation.

[7] Johns Hopkins University. Session Tracking. http://www.apl.jhu.edu/~hall/java/Servlet-Tutorial/Servlet-Tutorial-Session-Tracking.html.