



Servicios Web Seguros

Sesión 10: Seguridad a nivel de transporte con Rampart



Puntos a tratar

- Configuración de Tomcat
- Servicio con SSL
- Cliente con SSL



Introducción

- La seguridad a nivel de transporte consiste en enviar los mensajes por conexión SSL
- El cliente necesita ser capaz de establecer ese tipo de conexión
- El servidor debe estar preparado para conexiones https y tener su almacén de certificados correspondiente
- Tanto cliente como servidor usarán Rampart



Configuración de Tomcat

- En apache-tomcat/conf/server.xml:

```
<Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"
    maxThreads="150" scheme="https" secure="true"
    clientAuth="false" sslProtocol="TLS"
    minSpareThreads="25" maxSpareThreads="75"
    enableLookups="false"
    keystoreFile="/home/servicios/claves/tomcat.keystore"
    keystorePass="tomcat-kspass"
    acceptCount="100" />
```

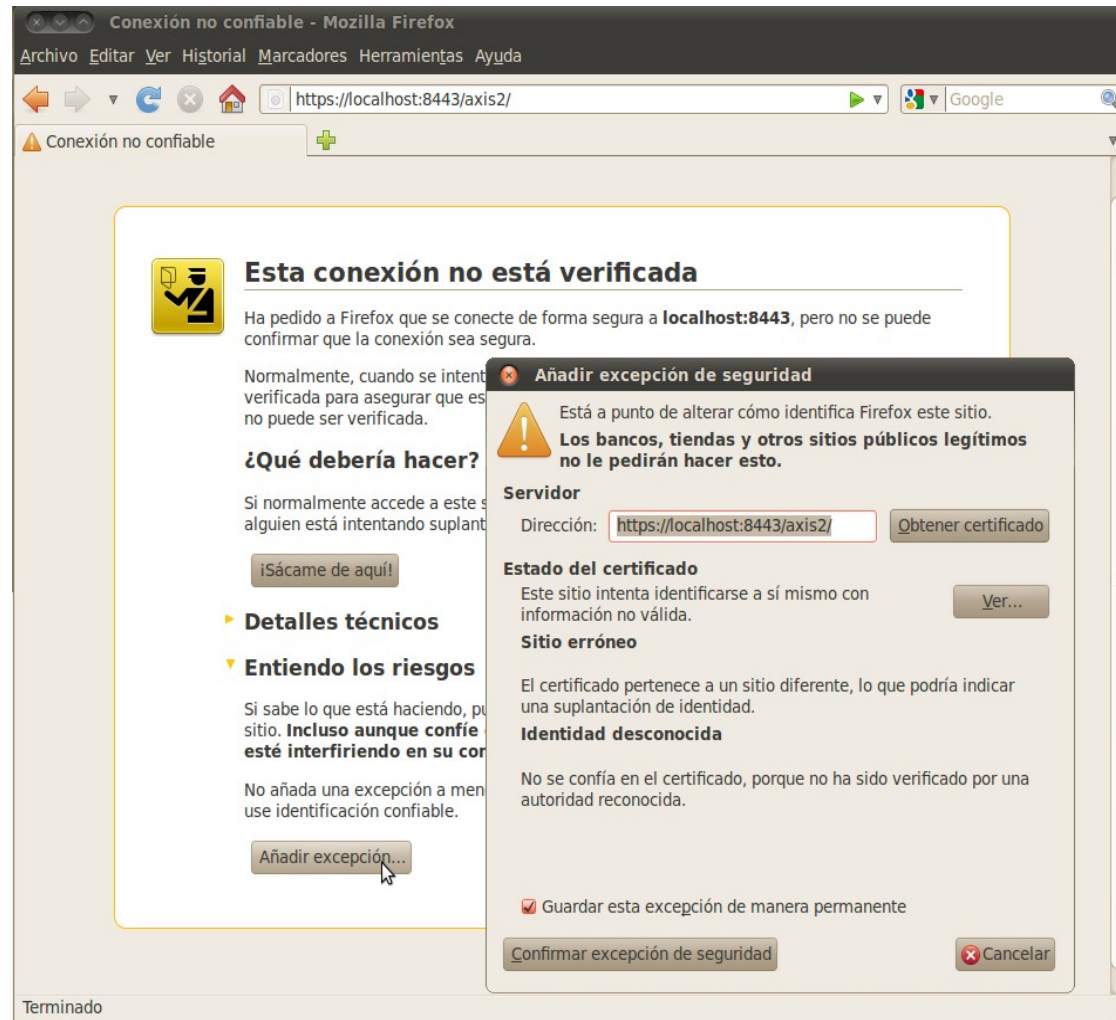


Configuración de Tomcat

- Crear un almacén de certificados JKS para Tomcat
- `keytool -genkey -alias tomcat -keyalg RSA -keystore ./tomcat.keystore`
- Ponemos la misma password para el almacén y para el certificado
- Intentamos acceder a través de https y por el puerto 8443:

Configuración de Tomcat

- Añadir excepción de seguridad al acceder por primera vez a `https://localhost:8443/`





Servicio con SSL

- Configuración de la política en el WSDL y en services.xml

```
<sp:TransportBinding
  xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
  <wsp:Policy>
    <sp:TransportToken>
      <wsp:Policy>
        <sp:HttpsToken RequireClientCertificate="false"/>
      </wsp:Policy>
    </sp:TransportToken>
    <sp:AlgorithmSuite>
      <wsp:Policy>
        <sp:Basic256/>
      </wsp:Policy>
    </sp:AlgorithmSuite>
    <sp:Layout>
      <wsp:Policy>
        <sp:Lax/>
      </wsp:Policy>
    </sp:Layout>
    <sp:IncludeTimestamp/>
  </wsp:Policy>
</sp:TransportBinding>
```



Servicio con SSL

- Configuración de la política en el WSDL y en services.xml

```
<sp:SignedSupportingTokens
  xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
  <wsp:Policy>
    <sp:UsernameToken sp:IncludeToken="http://schemas.xmlsoap.org/
      ws/2005/07/securitypolicy/IncludeToken/AlwaysToRecipient" />
  </wsp:Policy>
</sp:SignedSupportingTokens>
```




Cliente con SSL

```
ConfigurationContext context = ConfigurationContextFactory.  
    createConfigurationContextFromFileSystem("repository");  
SegTransporteStub stub = new SegTransporteStub(context,  
    "https://localhost:8443/axis2/services/SegTransporte");  
stub._getServiceClient().engageModule("rampart");  
Policy rampartConfig = new Policy();  
rampartConfig.addPolicyComponent(new RampartConfig());  
stub._getServiceClient().getAxisService().getPolicySubject().  
    attachPolicy(rampartConfig);  
stub._getServiceClient().getOptions().setPassword("boyan-pass");  
stub._getServiceClient().getOptions().setUserName("boyan");  
  
System.setProperty("javax.net.ssl.trustStore",  
    "/home/servicios/claves/tomcat.keystore");  
System.setProperty("javax.net.ssl.trustStorePassword", "tomcat-kspass");  
  
System.out.println(stub.saluda("Boyan", "Bonev"));
```



¿Preguntas...?