Arquitectura e Integración de Aplicaciones Empresariales

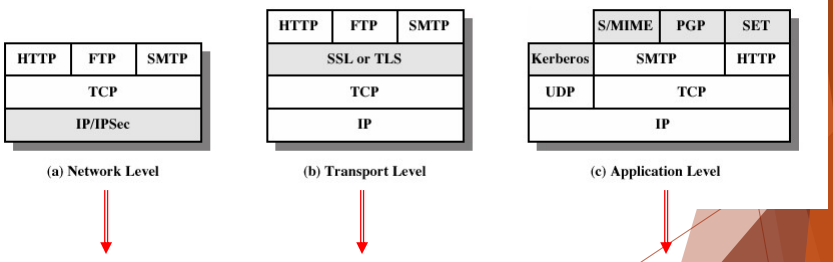
Novena Sesión
Protocolos de Seguridad

Universidad Autónoma Metropolitana
Casa abierta al tiempo Azcapotzalco

Dra. Maricela Bravo
Cubículo H-287-B
mari_clau_18@hotmail.com

¿Dónde ofrecer Seguridad?

- Discusión bizantina sin respuesta final



Conceptos generales

- Llaves pública/privada:
- Información encriptada con una llave solo puede ser desencriptada por su par.
- La base es mantener una llave guardada y distribuir la otra.
- El par de llaves está basado en un número primo.

Conceptos Generales

- Llave Simétrica:
- Usa la misma llave para encriptar y desencriptar (más práctico)
- Mucho más veloz que algoritmo asimétrico.
- Potencialmente inseguro.
- Transmitir llave simétrica con ayuda de algoritmos asimétricos.

Conceptos Generales

- Certificado Digital:
- Certifica que una persona o entidad es quien dice ser.
- Dirección de Correo, nombre, uso del certificado, ID de entidad que lo firma.
- CA almacenados en Navegador.
- Todo certificado es no confiable si no se ha firmado.

Protocolos de Seguridad

- HTTP no es un protocolo seguro
- Es simple y no se establece un estado cliente/servidor. Ejecuta sobre TCP/IP
- Es necesario instrumentar medidas de seguridad
  - SSL (Secure Socket Layer)
  - TLS (Transport Layer Security)
  - HTTPS
- Protocolo seguro HTTP
- El uso de SSL se aplica también a otras capas TCP/IP, por ejemplo,
- POP3, SMTP, FTP, SSH, etc.

6

- Transport Layer Security (TLS)
- Secure Sockets Layer (SSL)
- Son protocolos criptográficos, que proporcionan comunicaciones seguras por una red, comúnmente Internet.
-

SSL

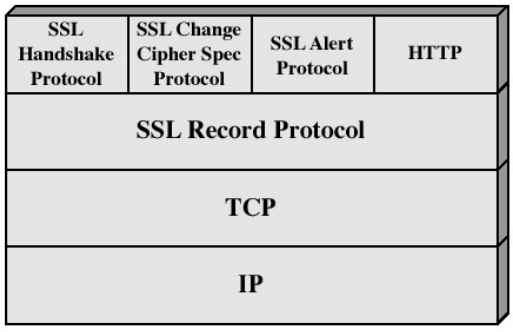
- SSL: Secure Sockets Layer, capa de sockets seguros.
- Establece un canal seguro en el nivel de transporte entre dos partes.
- Ofrece privacidad en las comunicaciones al utilizar cifrado con llave simétrica y usa códigos de autentificación de mensajes.

SSL

- SSL proporciona autenticación y privacidad de la información entre extremos sobre Internet mediante el uso de criptografía. Habitualmente, solo el servidor es autenticado (es decir, se garantiza su identidad) mientras que el cliente se mantiene sin autenticar.
- SSL implica una serie de fases básicas:
  - Negociar entre las partes el algoritmo que se usará en la comunicación
  - Intercambio de claves públicas y autenticación basada en certificados digitales.
  - Cifrado del tráfico basado en cifrado simétrico.
- Durante la primera fase, el cliente y el servidor negocian qué algoritmos criptográficos se van a usar. Las implementaciones actuales proporcionan las siguientes opciones:
  - Para criptografía de clave pública: RSA, Diffie-Hellman, DSA (Digital Signature Algorithm) o Fortezza.
  - Para cifrado simétrico: RC2, RC4, IDEA (International Data Encryption Algorithm), DES (Data Encryption Standard), Triple DES y AES (Advanced Encryption Standard).
- Con funciones hash: MD5 o de la familia SHA.

Arquitectura SSL

- Utiliza TCP (transferencia de datos confiable)

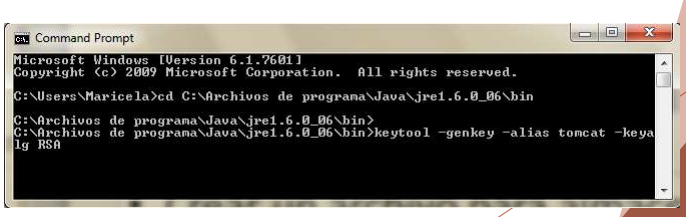


Instalación y configuración de SSL en Tomcat 8

Creación de la llave para el servidor

- Crear un archivo para almacenar la llave privada del servidor y un certificado auto-firmado (esto normalmente se solicita a una certificadora)

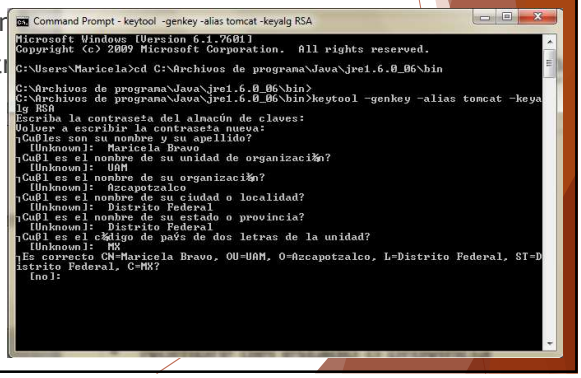
%JAVA_HOME%\bin\keytool -genkey -alias tomcat -keyalg RSA



Proporcionar los datos para crear la clave

- La contraseña del almacén de claves (6 digitos)
- Nombre y apellido
- Nombre de la unidad: Azcapotzalco
- Nombre de la organización: UAM
- Nombre de la ciudad o localidad
- Nombre del estado o provin
- Código del país con dos let

Buscar el archivo *.keystore

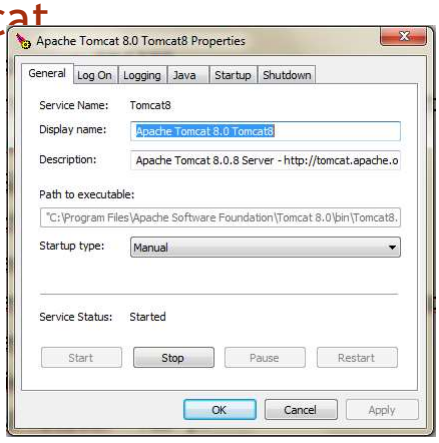


- Tomcat solamente opera con formatos de llave JKS, PKCS11 o PKCS12.
- El formato de JKS es el estándard de java “Java Keystore”, y es el formato creado por el comando keytool.

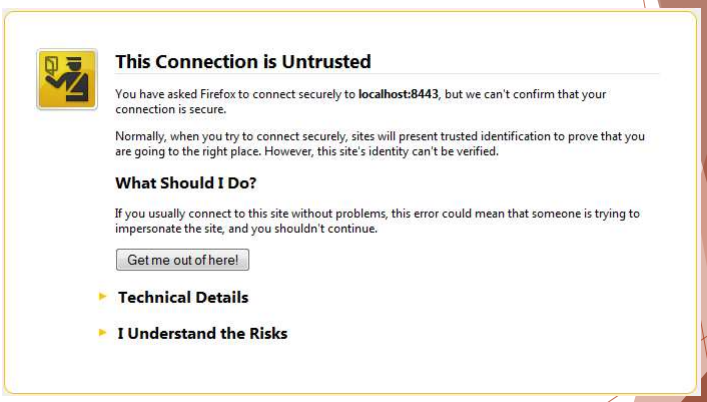
Abrir, modificar y guardar el archivo de configuración del servidor conf/server.xml

```
<Connector port="8443"
    protocol="org.apache.coyote.http11.Http11NioProtocol“
    maxThreads="200"
    SSLEnabled="true"
    scheme="https"
    secure="true“
    keystoreFile="C:/Users/Maricela/.keystore"
    keystorePass="123456“
    clientAuth="false"
    sslProtocol="TLS" />
```

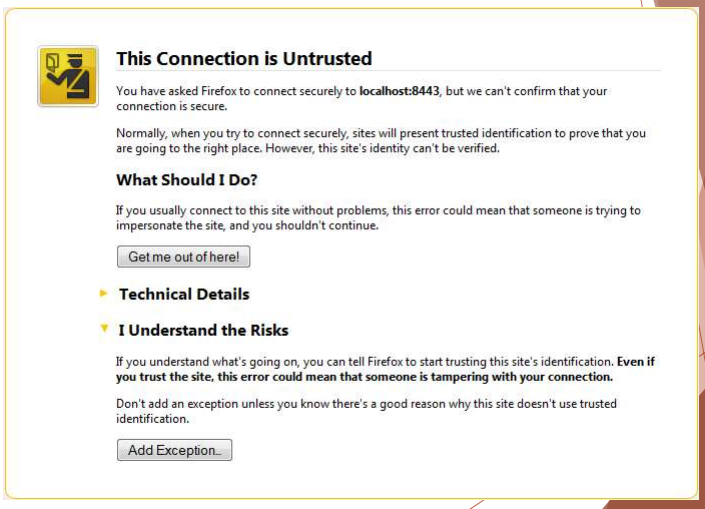
Detener y reiniciar el servicio de Tomcat



Probar en el navegador con: https://localhost:8443



Agregar la excepción



Agregar la excepción

Finalmente