Servicios Web Seguros

Sesión 11: Metro. Seguridad de nivel de transporte

Puntos a tratar

- Introducción a Metro
- Comparativa entre Axis 2 y Metro
- Tipos de seguridad en NetBeans
- Configuración de la seguridad en NetBeans
- Seguridad a nivel de transporte con SSL
- Autentificación con nombre de usuario en SSL
- Autentificación con certificado digital en SSL
- Autentificación con SAML en SSL

Introducción a Metro

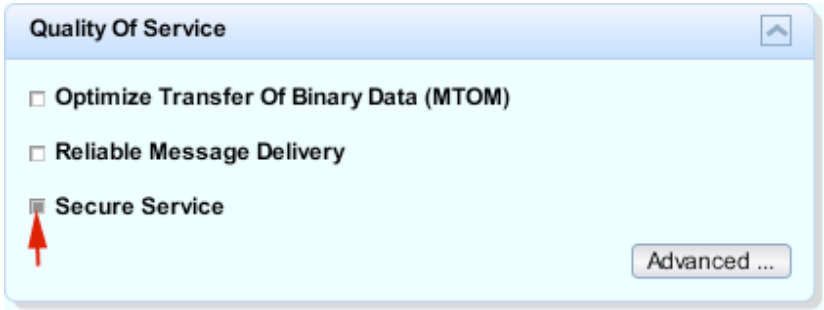
- Pila para servicios web
- Comprende los estándares de Sun
  - JAX-WS
    - Servicios web básicos
  - JAXB
    - Vinculación XML - Java
  - WSIT
    - Extensiones SOAP (*WS-**)
    - Incluye *WS-Security* y *WS-Trust*

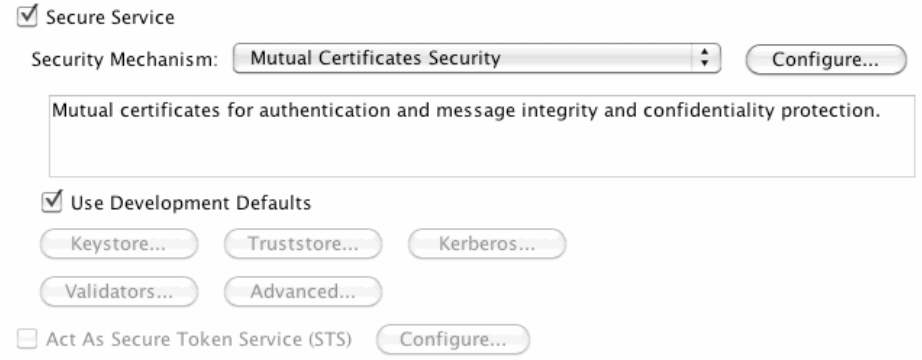
Axis 2 y Metro

Axis 2	*Metro*
Abierto	Estándares de Sun
Módulos adicionales	Todo viene de serie
Flexible	Configuración sencilla
Despliegue en AAR	Despliegue en WAR

GlassFish y NetBeans

- Metro forma parte de GlassFish
- NetBeans proporciona una interfaz gráfica para configurar los servicios de Metro
- El entorno NetBeans-GlassFish simplifica la creación de servicios web seguros





Tokens de seguridad

- Username token
  - Login y password
- X.509 token
  - Certificado digital
- SAML token
  - Información de autentificación y autorización (XML)
- Kerberos token
  - Tickets proporcionados por centro de claves
- STS Issued token

Almacenes de claves y certificados

- *Keystore*
  - Pares de claves
    - Certificados propios
    - Clave privada asociada
- *Truststore*
  - Certificados en los que confiamos
    - Certificados raíz
    - Certificados de sujetos con los que nos comunicamos
- GlassFish contiene su propio *keystore* y *truststore* ya configurado

Seguridad en NetBeans (transporte)

- Transport Security (SSL)
  - Seguridad a nivel de transporte
- Message Authentication over SSL
  - Seguridad a nivel de transporte
  - Autentificación con login y password o certificado digital
- SAML Authentication over SSL
  - Autentificación con tokens SAML *Sender Vouches*
  - Cliente actúa en nombre de otro sujeto
  - Relación de confianza entre cliente-servicio

Seguridad en NetBeans (mensaje)

- Mutual Certificates Security
  - Seguridad a nivel de mensaje con clave asimétrica
  - Cliente y servicio tienen sus propios certificados
- Username Authentication with Symmetric Key
  - Seguridad a nivel de mensaje con clave simétrica
  - Autentificación con login y password
- Endorsing Certificate
  - Seguridad a nivel de mensaje con clave simétrica
  - Autentificación con certificado digital
  - Se incluye como *EndorsingSupportingToken*

Seguridad en NetBeans (otros tokens)

- Orientados a *Single Sign-On (SSO)*
  - SAML Sender Vouches with Certificates
  - SAML Holder of Key
  - SAML Authentication over SSL
  - Symmetric Binding with Kerberos Tokens
  - STS Issued Token
  - STS Issued Token with Service Certificate
  - STS Issued Endorsing Token
  - STS Issued Supporting Token

Seguridad a nivel de transporte con SSL

- El cliente debe crearse con la dirección segura

  `https://localhost:8181/.../MyService?wsdl`

- Si se accede al servicio sin inyección
  - Especificar la dirección del WSDL al instanciar el *stub* del servicio en el cliente
- Si se accede fuera de GlassFish
  - Incluir librerías de Metro en el proyecto
  - Configurar manualmente *keystore* y *truststore*
  - Hacer *Clean and Build* siempre que sea necesario

Autentificación con login y password

- Por defecto utiliza los usuarios del *realm* `file`
- Para modificar esto podemos:
  - Cambiar *realm* por defecto en GlassFish
  - Crear el servicio dentro de una aplicación enterprise (EAR), y modificar el *realm* de la aplicación en `sun-application.xml`
  - Crear un *validator* (ignorado por GlassFish)
  - Reemplazar `CallbackHandler` por defecto de GlassFish para implementar autentificación programada

Autentificación en el servidor

- Dentro del servicio podemos obtener el nombre del usuario
- Inyectar propiedad

```
@Resource
WebServiceContext context;
```

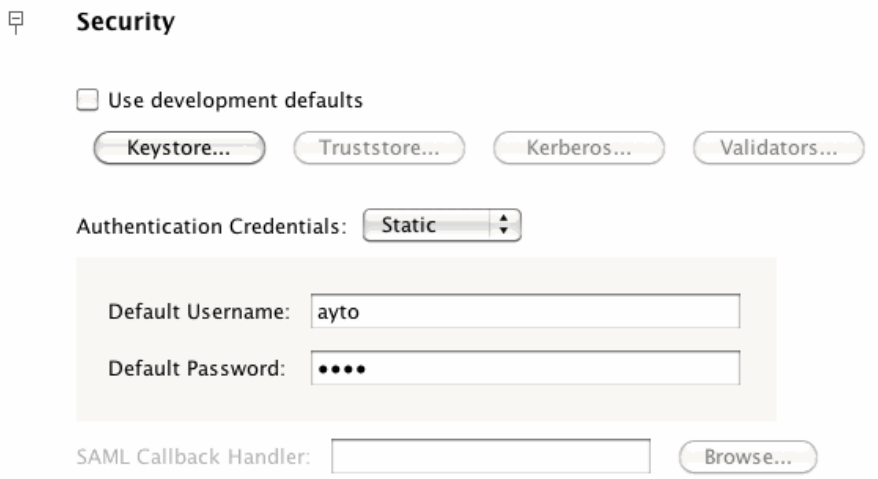
- Obtener principal

```
context.getUserPrincipal().getName();
```

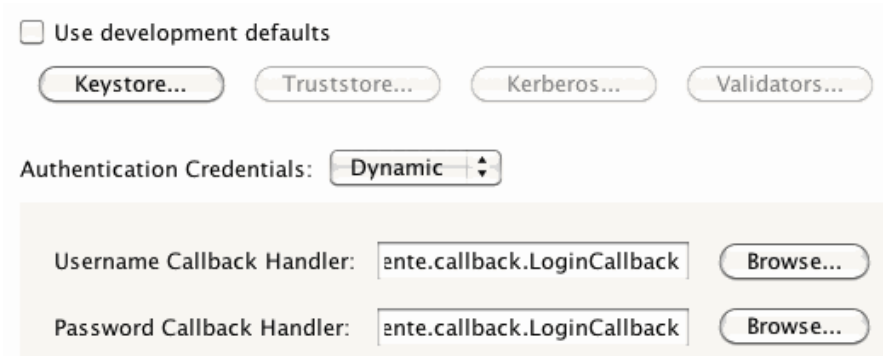
- Este mismo método sirve para obtener los datos del certificado digital

Autentificación en el cliente

- Credenciales estáticos
  - Se incluye el login y password en el fichero de configuración de WSIT
  - Problema de seguridad



- Credenciales dinámicos
  - Se utiliza un `CallbackHandler` para solicitar login y password
  - No funciona en aplicaciones web



Autentificación con certificado digital

- Configurar el servicio
  - Autentificación con X.509
  - Establecer *truststore*
- Configurar el cliente
  - Establecer *keystore* y alias

- Clientes y servicios en GlassFish
  - Siempre utiliza el almacén de GlassFish
- Clientes independientes
  - Especificar manualmente la ruta del *keystore*

Autentificación con SAML SV sobre SSL

- Cliente y servicio confían de antemano
- El cliente accede en nombre de otro sujeto
- En el *token* SAML se proporciona información de autentificación y autorización del sujeto
- Marcando *Use development defaults* en el cliente crea un *callback* SAML de ejemplo
- Si estamos fuera de GlassFish, deberemos modificar el *callback* para indicar la ruta exacta de los almacenes de certificados

¿Preguntas...?