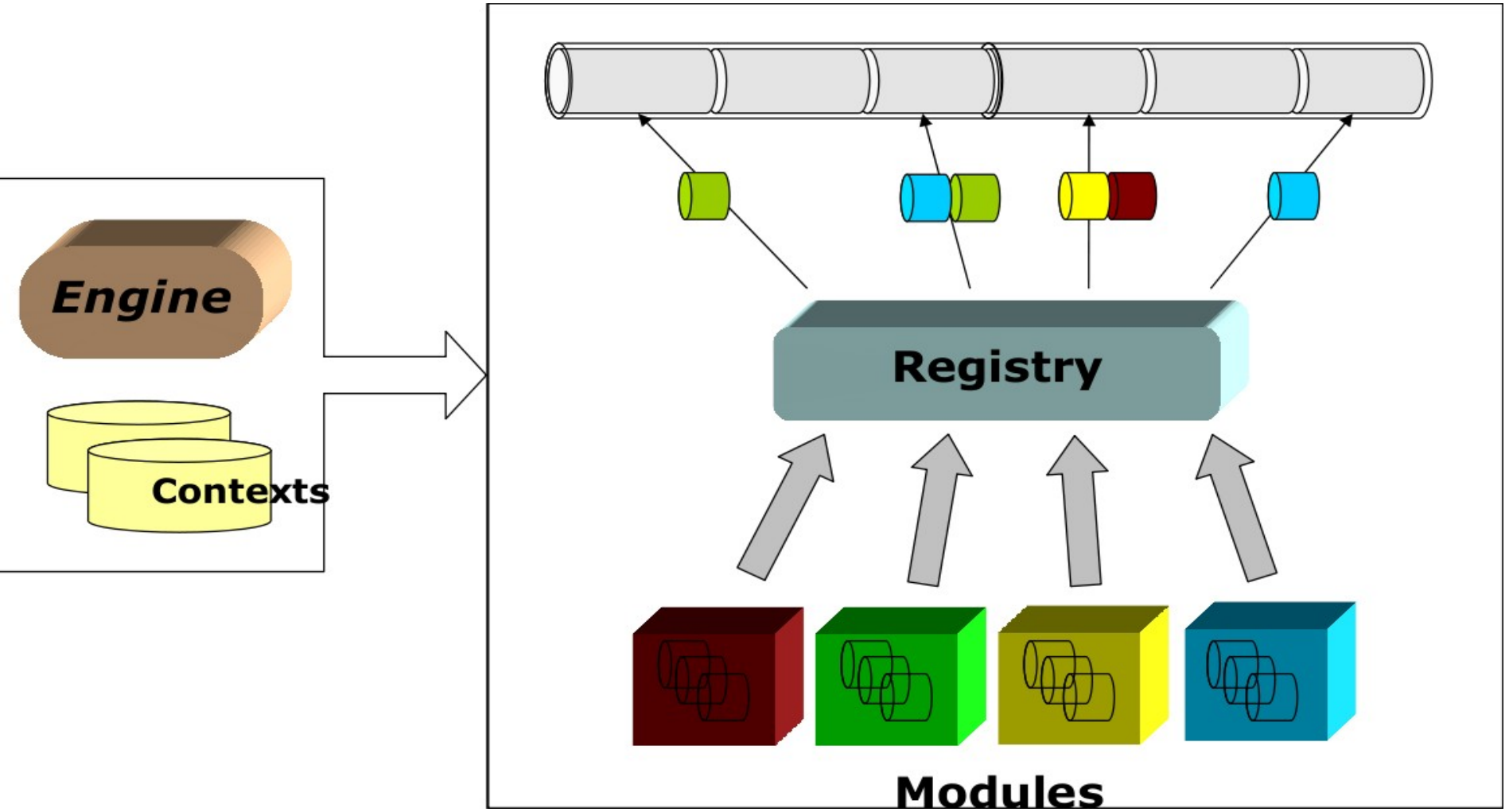
Servicios Web Seguros

Sesión 8: Rampart y Axis2

Puntos a tratar

- Rampart con Axis2
- Configuración de Rampart
- Cliente
- Servidor

Módulos en Axis2



Módulos en Axis2

- Módulos conectables o “pluggable”
- Extensiones que proveen funcionalidades específicas
- Consisten en
  - Manejadores (Handlers)
  - Clase Module
  - Descriptor (module.xml)
- Empaquetados en .mar

Módulos en Axis2

- Engranar (engage) los módulos en Axis2
  - Por sistema
  - Por servicio
  - Por operación

Rampart

- Módulo de seguridad para Axis2
- Rampart provee las características de seguridad especificadas en WS-Security
  - Autentificación (Username Token)
  - Confidencialidad (encriptación)
  - Integridad y no-suplantación (firma digital)
- Módulo adicional Rahas
  - Permite al servicio actuar como Security Token Service (para protocolos WS-SecureConversation)

Instalación de Rampart en Axis2

- Copiar
  - rampart-1.5/modules/rahas-1.5.mar
  - rampart-1.5/modules/rampart-1.5.mar
- a la carpeta de Axis2:
  - axis2-1.5.1/repository/modules
- Copiar las librerías *.jar
  - rampart-1.5/lib/*
- a la carpeta de Axis2:
  - axis2-1.5.1/lib/

Configuración de Rampart

- WSDL: usaremos otras versiones de las políticas de seguridad.
  - xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy"
  - xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
  - Políticas de IncludeToken:
    - http://docs.oasis-open.org/ws-sx/ws-securitypolicy/200702/IncludeToken/*

Configuración de Rampart

- En lugar de crypto.properties usaremos un archivo .xml de configuración de Rampart:

```
<?xml version="1.0" encoding="UTF-8"?>
<wsp:Policy xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
        xmlns="http://ws.apache.org/rampart/policy">
        <RampartConfig>
                <user>boyan</user>
                <userCertAlias>cliente1</userCertAlias>
                <encryptionUser>servidor1</encryptionUser>
                <passwordCallbackClass>
                        es.ua.jtech.seguro.custom.PasswordCallbackHandler
                </passwordCallbackClass>
                <signatureCrypto>
                 ...
                </signatureCrypto>
                <encriptionCrypto>
                 ...
                </encriptionCrypto>
        </RampartConfig>
</wsp:Policy>
```

Configuración de Rampart

```
<signatureCrypto>
        <crypto provider="org.apache.ws.security.components.crypto.Merlin">
                <property name="org.apache.ws.security.crypto.merlin.keystore.
                                                                        type">
                        JKS
                </property>
                <property name="org.apache.ws.security.crypto.merlin.file">
                        /home/servicios/claves/cliente.ks
                </property>
                <property name="org.apache.ws.security.crypto.merlin.keystore.
                                                                    password">
                        cliente1-kspass
                </property>
        </crypto>
</signatureCrypto>
```

Configuración de Rampart

```
<encriptionCrypto>
        <crypto provider="org.apache.ws.security.components.crypto.Merlin">
                <property name="org.apache.ws.security.crypto.merlin.keystore.
                                                                        type">
                        JKS
                </property>
                <property name="org.apache.ws.security.crypto.merlin.file">
                        /home/servicios/claves/cliente.ks
                </property>
                <property name="org.apache.ws.security.crypto.merlin.keystore.
                                                                    password">
                        cliente1-kspass
                </property>
        </crypto>
</encriptionCrypto>
```

Cliente

```
public class Cliente {
        public static void main(String[] args)
          throws RemoteException, FileNotFoundException, XMLStreamException {

                ConfigurationContext context = ConfigurationContextFactory.
                     reateConfigurationContextFromFileSystem("repository");

                SeguroStub servicio = new SeguroStub(context,
                     "http://localhost:1234/axis2/services/Seguro");
                servicio._getServiceClient().engageModule("rampart");

                StAXOMBuilder builder =
                     new StAXOMBuilder("src/main/resources/rampartConfig.xml");

                Policy rampartPolicy =
                     PolicyEngine.getPolicy(builder.getDocumentElement());

                servicio._getServiceClient().getAxisService().getPolicySubject().
                     attachPolicy(rampartPolicy);

                System.out.println(servicio.saluda("Boyan", "Bonev"));
        }
}
```

Cliente

- Para funcionar con las librerías de Rampart, el cliente también necesita los .mar
- Creamos en el proyecto cliente la carpeta
  - repository/modules
- Y le copiamos los .mar:
  - rampart-1.5/modules/rahas-1.5.mar
  - rampart-1.5/modules/rampart-1.5.mar

Servidor

- En /src/main/resources/META-INF/services.xml

```
<module ref="rampart" />
<wsp:Policy
    xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy"
        xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
                            oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="p1">
        <RampartConfig xmlns="http://ws.apache.org/rampart/policy">
                <!-- Pegar aquí el contenido de la RampartConfig de
                                                    rampartConfig.xml -->
                <!-- Cambiar el keystore a servidor.ks y la password
                                                   a servidor1-kspass -->
                <!-- Eliminar <user> y cambiar los siguientes alias     -->
                <userCertAlias>servidor1</userCertAlias>
                <encryptionUser>useReqSigCert</encryptionUser>
        </RampartConfig>
        <!-- Pegar aquí el contenido de la Policy de Seguro.wsdl, incluído
        el AsymmetricBinding, Wss10, SignedParts, EncryptedParts,
                                            SignedSupportingTokens -->


    </wsp:Policy>
```

¿Preguntas...?