



**Gobierno
de Canarias**

Consejería de Presidencia,
Justicia y Seguridad
Dirección General de
Telecomunicaciones
y Nuevas Tecnologías



Platino
Plataforma de Interoperabilidad del
Gobierno de Canarias

Interoperabilidad de los servicios telemáticos de la
Administración Pública de la CAC

Página 1 de 23

ACCESO A LOS SERVICIOS DE PLATINO Entorno de Pre-Explotación

Rev.	Fecha	Descripción
0	12/05/2008	Redacción inicial
1	03/06/2008	Actualización clientes de prueba en temisas
2	16/06/2008	Se listan las direcciones IP de los servidores a los que solicitar acceso
3	08/07/2008	Se actualiza documento tras ajustes en entorno de pre explotación
4	26/08/2008	Se necesita acceso por VPN a lagos para el trabajo con los componentes de firma
5	10/11/2008	Se documenta el acceso con la seguridad habilitada a los servicios de Platino
6	18/11/2008	Modificaciones tras la revisión de la DGTNT
7	14/07/2009	Se añade el wsdl de localizaciones del servicio de Terceros
8	08/02/2010	Se corrige la url de la aplicación de pruebas para SGRDE y Firma
9	26/08/2010	Se añade la url del servicio de ForPDF
Documento :		PLA-DOC-BOR-08-05-06-Acceso a los servicios en pre
Ubicación en eRoom:		
Preparado por		Revisado por
D. Gral. de Telecomunicaciones y Nuevas Tecnologías		D. Gral. de Telecomunicaciones y Nuevas Tecnologías
Aprobado por		
D. Gral. de Telecomunicaciones y Nuevas Tecnologías		D. Gral. de Telecomunicaciones y Nuevas Tecnologías
Fecha: 08/02/2010		Fecha: 08/02/2010
		Fecha: 08/02/2010

 <p>Gobierno de Canarias Consejería de Presidencia, Justicia y Seguridad Dirección General de Telecomunicaciones y Nuevas Tecnologías</p>	 <p>Platino Plataforma de Interoperabilidad del Gobierno de Canarias</p>
Acceso a los servicios en pre-explotación	Página 2 de 23

Índice de contenido

1 INTRODUCCIÓN	3
2 REQUISITOS	3
3 ACCESO A LOS SERVICIOS	3
4 CREACIÓN DE CLIENTES CON JAVA	7
4.1 Generar el cliente usando Apache CXF	7
4.2 Consumir el web service generado	9
4.3 Seguridad en PLATINO	10
4.4 Formato de los parámetros adicionales de la cabecera	11
4.5 Ejemplo de petición sin cabeceras	11
4.6 Ejemplo de petición con cabeceras de seguridad habilitadas	12
4.7 Añadiendo cabeceras de seguridad a las peticiones	15
4.7.1 USO DE LA LIBRERÍA	15
4.7.2 FICHERO CLIENT_SIGN.PROPERTIES	17
4.7.3 EJEMPLOS DE PETICIONES	18
4.8 Secuencia de pasos	21
5 ANEXOS	23
5.1 Consulta del alias del certificado almacenado en un PFX	23

 Gobierno de Canarias Consejería de Presidencia, Justicia y Seguridad Dirección General de Telecomunicaciones y Nuevas Tecnologías	 Platino Plataforma de Interoperabilidad del Gobierno de Canarias
Acceso a los servicios en pre-explotación	Página 3 de 23

1 INTRODUCCIÓN

El presente documento pretende exponer todos aquellos elementos técnicos necesarios para que los equipos de desarrollo puedan hacer uso de los diferentes servicios de Platino

2 REQUISITOS

Para poder utilizar el entorno de pre-explotación es necesario contar con acceso a los servidores de Platino, ya sea directamente a través de la red corporativa del Gobierno de Canarias, o mediante una conexión VPN con los permisos adecuados.

Para solicitar un acceso por VPN se deberá dirigir la petición a platino@gobiernodecanarias.org indicando que se solicita “Acceso a Platino PRE”.

El consumidor de Platino también debe poseer un certificado electrónico emitido por Cibercentro que permita el acceso seguro a la plataforma Platino. Este certificado será proporcionado por la Oficina Técnica de Platino como parte del procedimiento de integración con Platino.

3 ACCESO A LOS SERVICIOS

A continuación se detallan los datos técnicos para cada uno de los servicios:

Firma y Sellado de Tiempo	
Definición del servicio:	http://laderas.gobiernodecanarias.net:8077/firma/main.wsd/
Aplicación de Pruebas:	http://lagos.gobiernodecanarias.net:8080/PlatinoClienteFirma/
Administración:	
Otras consideraciones:	

 <p>Gobierno de Canarias Consejería de Presidencia, Justicia y Seguridad Dirección General de Telecomunicaciones y Nuevas Tecnologías</p>	 <p>Platino Plataforma de Interoperabilidad del Gobierno de Canarias</p>
Acceso a los servicios en pre-explotación	Página 4 de 23

Formularios Electrónicos	
Definición del servicio:	http://laderas.gobiernodecanarias.net:8076/formularios/?wsdl
Aplicación de Pruebas:	http://temisas.gobiernodecanarias.net:8080/cliente-sfe
Administración:	
Otras consideraciones:	Editor de formularios en: http://majoreras.gobiernodecanarias.net:8080/formularioeditor/formulario-selector.jsf

Notificaciones Electrónicas	
Definición del servicio:	http://laderas.gobiernodecanarias.net:8077/enotificaciones/notificacion/main.wsdl
Aplicación de Pruebas:	http://temisas.gobiernodecanarias.net:8080/servicio-notificaciones-web
Administración:	
Otras consideraciones:	Acceso al entorno de pre-explotación de <i>Correos</i> : http://193.148.159.24/PortalCiudadano/paginas/comunes/inicio.aspx Se requiere certificado digital para probar el servicio (para entrar al buzón de notificaciones)

Repositorio de Documentos	
Definición del servicio:	http://laderas.gobiernodecanarias.net:8076/sgrde/?wsdl
Aplicación de Pruebas:	http://temisas.gobiernodecanarias.net:8080/sgrde-web-client
Administración:	
Otras consideraciones:	

 <p>Gobierno de Canarias</p> <p>Consejería de Presidencia, Justicia y Seguridad</p> <p>Dirección General de Telecomunicaciones y Nuevas Tecnologías</p>	 <p>Platino Plataforma de Interoperabilidad del Gobierno de Canarias</p>
Acceso a los servicios en pre-explotación	Página 5 de 23

Registro Electrónico

Definición del servicio:	http://laderas.gobiernodecanarias.net:8076/registro/?wsdl
Aplicación de Pruebas:	http://temisas.gobiernodecanarias.net:8080/cliente-sres/
Administración:	
Otras consideraciones:	

Base de Datos de Terceros

Definición del servicio:	http://laderas.gobiernodecanarias.net:8077/terceros/main.wsdl http://laderas.gobiernodecanarias.net:8077/localizaciones/main.wsdl
Aplicación de Pruebas:	
Administración:	http://nayara.gobiernodecanarias.net:8080/gesterceros
Otras consideraciones:	

Pasarela de Pagos

Definición del servicio:	http://laderas.gobiernodecanarias.net:8077/PasarelaPago/main.wsdl
Aplicación de Pruebas:	http://temisas.gobiernodecanarias.net:8080/PasarelaPagoTest/
Administración:	
Otras consideraciones:	

Envío de Mensajes y Correos Electrónicos

Definición del servicio:	http://laderas.gobiernodecanarias.net:8077/edmyce/listaDistribucion/main.wsdl http://laderas.gobiernodecanarias.net:8077/edmyce/area/main.wsdl http://laderas.gobiernodecanarias.net:8077/edmyce/mensaje/main.wsdl
Aplicación de Pruebas:	http://temisas.gobiernodecanarias.net:8080/servicio-mensajes-web/
Administración:	
Otras consideraciones:	

 <p>Gobierno de Canarias Consejería de Presidencia, Justicia y Seguridad Dirección General de Telecomunicaciones y Nuevas Tecnologías</p>	 <p>Platino Plataforma de Interoperabilidad del Gobierno de Canarias</p>
Acceso a los servicios en pre-explotación	Página 6 de 23

Soporte a la tramitación telemática

Definición del servicio:	http://laderas.gobiernodecanarias.net:8077/tramitacion/main.wsdl
Aplicación de Pruebas:	
Administración:	
Otras consideraciones:	

Base de Datos de Procedimientos

Definición del servicio:	http://laderas.gobiernodecanarias.net:8077/dbprocedimientos/main.wsdl
Aplicación de Pruebas:	
Administración:	http://majoreras.gobiernodecanarias.net:8080/bdprocedimientos
Otras consideraciones:	

Base de Datos de Organización

Definición del servicio:	http://laderas.gobiernodecanarias.net:8077/dborganizacion/main.wsdl
Aplicación de Pruebas:	
Administración:	
Otras consideraciones:	

Formularios PDF

Definición del servicio:	http://laderas.gobiernodecanarias.net:8077/forpdf/main.wsdl
Aplicación de Pruebas:	
Administración:	
Otras consideraciones:	

 <p>Gobierno de Canarias</p> <p>Consejería de Presidencia, Justicia y Seguridad</p> <p>Dirección General de Telecomunicaciones y Nuevas Tecnologías</p>	 <p>Platino Plataforma de Interoperabilidad del Gobierno de Canarias</p>
<p>Acceso a los servicios en pre-explotación</p>	<p>Página 7 de 23</p>

4 CREACIÓN DE CLIENTES CON JAVA

Para la creación de los clientes que consuman los servicios web de PLATINO, utilizaremos una herramienta que es capaz de generar automáticamente las clases Java necesarias para invocar estos servicios.

Dado que otros generadores muy populares como Axis, están dando algún problema por el uso de elementos de diferentes versiones de SOAP, recomendamos utilizar Apache CXF. Este producto es un framework de servicios open source que puede descargarse desde la siguiente URL: <http://cxf.apache.org/download.html>

4.1 Generar el cliente usando Apache CXF

Utilizando nuestro entorno IDE favorito, crearemos un nuevo proyecto. Lo primero que debemos hacer es añadir al proyecto las librerías con las que CXF tiene dependencias. Todas estas librerías se encuentran en el directorio 'lib' de la distribución de CXF:

```

aopalliance-1.0.jar
asm-2.2.3.jar
commons-lang-2.1.jar
commons-logging-1.1.jar
cxf-2.1.jar
FastInfoset-1.2.2.jar
geronimo-activation_1.1_spec-1.0.2.jar
geronimo-annotation_1.0_spec-1.1.jar
geronimo-javamail_1.4_spec-1.3.jar
geronimo-ws-metadata_2.0_spec-1.1.1.jar
jaxb-api-2.1.jar
jaxb-impl-2.1.6.jar
jaxws-api-2.1-1.jar
jdom-1.0.jar
neethi-2.0.2.jar
saaj-api-1.3.jar
saaj-impl-1.3.jar
spring-beans-2.0.6.jar
spring-context-2.0.6.jar
spring-core-2.0.6.jar
spring-web-2.0.6.jar
stax-api-1.0.1.jar
wsdl4j-1.6.1.jar
wss4j-1.5.1.jar
wstx-asl-3.2.1.jar
xml-resolver-1.2.jar
XmlSchema-1.3.2.jar
xmlsec-1.3.0.jar

```

 Gobierno de Canarias Consejería de Presidencia, Justicia y Seguridad Dirección General de Telecomunicaciones y Nuevas Tecnologías	 Platino Plataforma de Interoperabilidad del Gobierno de Canarias
Acceso a los servicios en pre-explotación	Página 8 de 23

Para generar un cliente únicamente necesitamos tener conocimiento de la URL del WSDL del servicio web que vamos a consumir y el path del sistema de ficheros local donde queremos que se generen las clases clientes.

El comando que debemos ejecutar es el siguiente:

```
> wsdl2java -d <output-directory> wsdlfile
```

El ejecutable 'wsdl2java' se encuentra en el directorio 'bin' de la distribución de CXF.

Un ejemplo de invocación sería el siguiente:

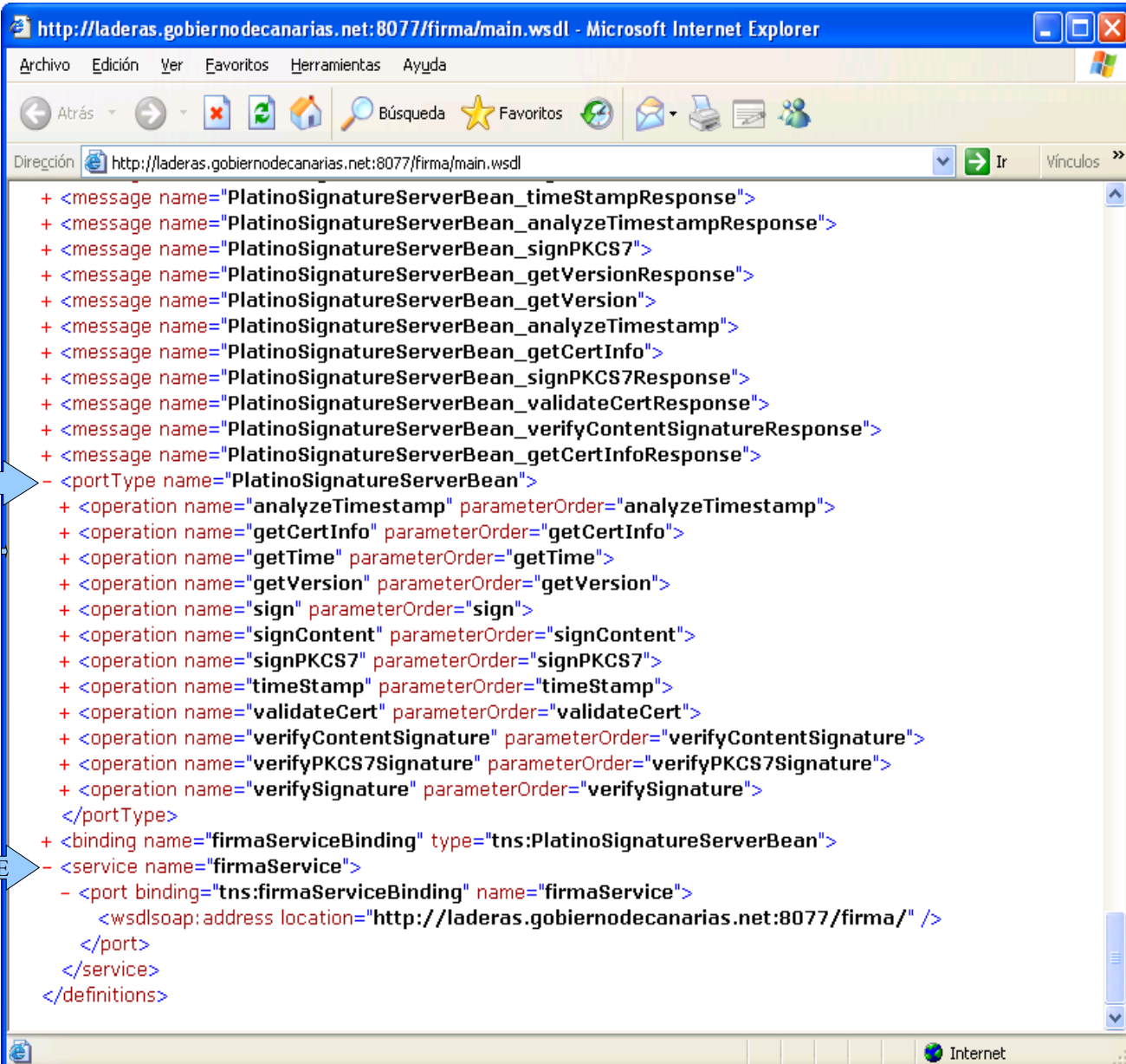
```
> wsdl2java -d c:\workspace\proyecto\src  
http://laderas.gobiernodecanarias.net:8077/firma/main.wsdl
```

Tras la ejecución, el generador CXF habrá creado todos los ficheros java necesarios para la invocación de los métodos del Servicio Web.

 <p>Gobierno de Canarias Consejería de Presidencia, Justicia y Seguridad Dirección General de Telecomunicaciones y Nuevas Tecnologías</p>	 <p>Platino Plataforma de Interoperabilidad del Gobierno de Canarias</p>
<p>Acceso a los servicios en pre-explotación</p>	<p>Página 9 de 23</p>

4.2 Consumir el web service generado

Una vez tengamos las clases generadas, podremos consumir el servicio de Platino. En las clases generadas, habrá una que coincida con el "service name" del WSDL que hemos usado. Esta clase extiende la clase 'Service'.



```

+ <message name="PlatinoSignatureServerBean_timeStampResponse">
+ <message name="PlatinoSignatureServerBean_analyzeTimestampResponse">
+ <message name="PlatinoSignatureServerBean_signPKCS7">
+ <message name="PlatinoSignatureServerBean_getVersionResponse">
+ <message name="PlatinoSignatureServerBean_getVersion">
+ <message name="PlatinoSignatureServerBean_analyzeTimestamp">
+ <message name="PlatinoSignatureServerBean_getCertInfo">
+ <message name="PlatinoSignatureServerBean_signPKCS7Response">
+ <message name="PlatinoSignatureServerBean_validateCertResponse">
+ <message name="PlatinoSignatureServerBean_verifyContentSignatureResponse">
+ <message name="PlatinoSignatureServerBean_getCertInfoResponse">
- <portType name="PlatinoSignatureServerBean">
+ <operation name="analyzeTimestamp" parameterOrder="analyzeTimestamp">
+ <operation name="getCertInfo" parameterOrder="getCertInfo">
+ <operation name="getTime" parameterOrder="getTime">
+ <operation name="getVersion" parameterOrder="getVersion">
+ <operation name="sign" parameterOrder="sign">
+ <operation name="signContent" parameterOrder="signContent">
+ <operation name="signPKCS7" parameterOrder="signPKCS7">
+ <operation name="timeStamp" parameterOrder="timeStamp">
+ <operation name="validateCert" parameterOrder="validateCert">
+ <operation name="verifyContentSignature" parameterOrder="verifyContentSignature">
+ <operation name="verifyPKCS7Signature" parameterOrder="verifyPKCS7Signature">
+ <operation name="verifySignature" parameterOrder="verifySignature">
</portType>
+ <binding name="firmaServiceBinding" type="tns:PlatinoSignatureServerBean">
- <service name="firmaService">
- <port binding="tns:firmaServiceBinding" name="firmaService">
  <wsdlsoap:address location="http://laderas.gobiernodecanarias.net:8077/firma/" />
</port>
</service>
</definitions>

```

Podremos instanciar esta clase y de ella obtener el 'Port' que nos permita invocar los métodos del servicio web:

 <p>Gobierno de Canarias</p> <p>Consejería de Presidencia, Justicia y Seguridad</p> <p>Dirección General de Telecomunicaciones y Nuevas Tecnologías</p>	 <p>Platino Plataforma de Interoperabilidad del Gobierno de Canarias</p>
<p>Acceso a los servicios en pre-explotación</p>	<p>Página 10 de 23</p>

```
FirmaService service = new FirmaService();
PlatinoSignatureServerBean port = firmaService.getFirmaService();
String version = port.getVersion();
```

4.3 Seguridad en PLATINO

La seguridad en Platino se basa en la especificación de Web Service Security (WSS), que propone un conjunto estándar de extensiones que pueden ser usadas para construir Web Services Seguros y para garantizar la integridad del contenido del mensaje.

Se puede consultar esta especificación a través del siguiente enlace:

<http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf>

Platino requiere que la cabecera WSS del mensaje SOAP utilizado para la invocación de cualquier servicio, contenga la siguiente información:

- El actor que debe aparecer en la cabecera de seguridad es:
<http://www.gobiernodecanarias.org/Platino/Authentication/1.0>
- La cabecera WSS debe contener el elemento wsse:UsernameToken, cuyo hijo wsse:Username dependerá del origen de la petición a PLATINO:
 - Si la petición a PLATINO se está realizando en nombre de un tercero que se ha autenticado previamente en el backoffice, deberá pasarse la URI del tercero obtenida del servicio de terceros de PLATINO
 - Si la petición a PLATINO se está realizando en nombre de un empleado público que se ha autenticado previamente en el backoffice, debería pasarse la URI del empleado público obtenida del servicio de BD de Organización.
 - Si la petición a PLATINO se está realizando en nombre del propio backoffice, deberá pasarse la URI que ha sido asignada a ese backoffice en la generación del certificado correspondiente.
- El campo wsse:Password del elemento wsse:UsernameToken es ignorado por las políticas de seguridad de PLATINO, ya que la autorización se realizará a partir de la información extraída de la firma y del campo wsse:UsernameToken. Por este motivo, asociado a este campo puede utilizarse tanto un valor vacío como cualquier cadena de caracteres arbitraria.

 <p>Gobierno de Canarias Consejería de Presidencia, Justicia y Seguridad Dirección General de Telecomunicaciones y Nuevas Tecnologías</p>	 <p>Platino Plataforma de Interoperabilidad del Gobierno de Canarias</p>
Acceso a los servicios en pre-explotación	Página 11 de 23

- El elemento `wsse:UsernameToken` también debe incluir el campo `wsse:Nonce`
- La cabecera WSS debe contener el elemento `wsse:BinarySecurityToken` que contenga la parte pública del certificado con el que se realiza la firma de la petición.
- La cabecera debe contener un elemento que agrupe ciertos parámetros adicionales, que se utilizarán en el ámbito de la autorización de determinadas operaciones sobre ciertos servicios. La definición de este elemento se comenta en el siguiente apartado.
- La cabecera WSS debe contener el elemento `wsse:Signature` donde se refleje que, con el certificado del punto anterior, se ha realizado la firma XML Signature de los elementos `wsse:UsernameToken`, `soap:Body` y `platino:platinoHeaders`

4.4 Formato de los parámetros adicionales de la cabecera

En ciertas ocasiones, es posible que PLATINO requiera que se le proporcionen ciertos parámetros adicionales para poder llevar a cabo la autorización. Estos parámetros se enviarán en la cabecera siguiente la siguiente estructura:

```
<platino:platinoHeaders xmlns:platino="http://platino.gobcan.es">
  <platino:parameter name="nombre1">valor1</platino:parameter>
  <platino:parameter name="nombre2">valor2</platino:parameter>
  ...
  <platino:parameter name="nombreN">valorN</platino:parameter>
</platino:platinoHeaders>
```

En el caso de que no se requiera proporcionar a PLATINO ningún parámetro, el elemento `<platino:platinoHeaders>` estará vacío.

4.5 Ejemplo de petición sin cabeceras

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:sfst="http://platino.gobcan.es/servicios/sfst/">
  <soapenv:Header/>
  <soapenv:Body>
    <sfst:getVersion/>
  </soapenv:Body>
</soapenv:Envelope>
```




```
</ds:Transforms>
<ds:DigestMethod
  Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"
  xmlns:ds="http://www.w3.org/2000/09/xmldsig#" />
<ds:DigestValue
  xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  4VWfEIjnBSCdrbmrGnorh4J5DmM=
</ds:DigestValue>
</ds:Reference>
<ds:Reference URI="#id-8694274"
  xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <ds:Transforms
    xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:Transform
      Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"
      xmlns:ds="http://www.w3.org/2000/09/xmldsig#" />
    </ds:Transforms>
    <ds:DigestMethod
      Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"
      xmlns:ds="http://www.w3.org/2000/09/xmldsig#" />
    <ds:DigestValue
      xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
      tALpI3yyj/TsfnCeVwEWIblUsfI=
    </ds:DigestValue>
  </ds:Reference>
<ds:Reference URI="#id-28681226"
  xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <ds:Transforms
    xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:Transform
      Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"
      xmlns:ds="http://www.w3.org/2000/09/xmldsig#" />
    </ds:Transforms>
    <ds:DigestMethod
      Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"
      xmlns:ds="http://www.w3.org/2000/09/xmldsig#" />
    <ds:DigestValue
      xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
      oZHM1Zd5zkviVBwJ782nQm8i+ao=
    </ds:DigestValue>
  </ds:Reference>
</ds:SignedInfo>
<ds:SignatureValue xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  QzbMKnhAD2sRe5eannzrtK8dSZzs/tbA18/9irOnrhp6f+U0Dru6Ulsbp94q+J4VcRV2FLmETIIK
  16UIJv/98r046r5Y4fXrnGSXNawqFDKfbefBaXfxN/PwscRIT4JOIjM10Kvc4KMLbhKPybECpQa
  ovBKfo5FxyY+rjQIK/ZE=
</ds:SignatureValue>
<ds:KeyInfo Id="KeyId-2884607"
```



```

    xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <wsse:SecurityTokenReference
      xmlns:wssu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
      wsu:Id="STRId-33517025"
      xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd">
      <wsse:Reference URI="#CertId-22603721"
        ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-
token-profile-1.0#X509v3"
        xmlns:wssu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" />
      </wsse:SecurityTokenReference>
    </ds:KeyInfo>
  </ds:Signature>
  <wsse:UsernameToken
    xmlns:wssu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd"
    wsu:Id="UsernameToken-27772036"
    xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
    <wsse:Username
      xmlns:wssu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
      platino://gobcan.es/servicios/terceros/tercero/4525e5da-526f-42d4-bf9b-
64bb02b98399
    </wsse:Username>
    <wsse:Password
      Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-
profile-1.0#PasswordText"
      xmlns:wssu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
    </wsse:Password>
    <wsse:Nonce
      xmlns:wssu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
      Ip81f+qT96Uan0rISx/GKw==
    </wsse:Nonce>
    </wsse:UsernameToken>
  </wsse:Security>
  <platino:platinoHeaders xmlns:platino="http://platino.gobcan.es"
    xmlns:wssu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd"
    wsu:Id="id-28681226" />
  </soap:Header>
  <soap:Body
    xmlns:wssu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd"
    wsu:Id="id-8694274">
    <ns2:getVersion
      xmlns:ns2="http://platino.gobcan.es/servicios/sfst/" />
  </soap:Body>
</soap:Envelope>

```

4.7 Añadiendo cabeceras de seguridad a las peticiones

Para añadir las cabeceras de seguridad necesarias para Platino, se ha desarrollado una librería que facilita mucho esta tarea si se ha utilizado CXF como generador de clientes.

Esta librería podrá descargarse desde la sección "Formación" en el portal WEB de Platino (<http://www.gobiernodecanarias.org/platino>)

Requisitos de la seguridad

- Debemos incorporar al classpath de nuestra aplicación (consumidora de servicios de Platino) la librería platinoWSSInterceptor.jar
- Es necesario estar en posesión del certificado emitido por Cibercentro para el acceso a Platino. Este certificado debe contener la clave privada para poder realizar la firma.
- En el raíz del classpath del cliente debe existir un fichero denominado client_sign.properties que será utilizado para definir las propiedades de acceso al contenedor del certificado con el que realizar la firma de los mensajes.

4.7.1 USO DE LA LIBRERÍA

El método que permite añadir las cabeceras tiene la siguiente interfaz:

addSoapWSSHeader		
Añade la cabecera Web Service Security		
Parámetro	Tipo	Descripción
service	Object	PortType al que se le va a añadir la cabecera de seguridad.
soapVersion	String	Indica la versión soap del endpoint al que vamos a conectarnos. Por defecto, todos los servicios de Platino utilizan la versión Soap 1.1. Los posibles valores para este parámetro son: PlatinoCXFSecurityHeaders.SOAP_11 PlatinoCXFSecurityHeaders.SOAP_12

 <p>Gobierno de Canarias Consejería de Presidencia, Justicia y Seguridad Dirección General de Telecomunicaciones y Nuevas Tecnologías</p>	 <p>Platino Plataforma de Interoperabilidad del Gobierno de Canarias</p>
Acceso a los servicios en pre-explotación	Página 16 de 23

username	String	Valor que se asignará al elemento UsernameToken de la cabecera WSS. Hay tres posibles situaciones: <ul style="list-style-type: none"> • Cuando el backoffice esté realizando una petición a Platino en nombre de un tercero, este parámetro contendrá la URI del tercero en Platino, obtenida a través del servicio de base de datos de terceros. • Cuando el backoffice esté realizando una petición a Platino en nombre de un empleado público, este parámetro contendrá la URI del empleado público en Platino, extraída a través del servicio de base de datos de organización. • Cuando el backoffice esté realizando una petición a Platino en su propio nombre, este parámetro contendrá la URI del backoffice asignada mediante el proceso de integración con Platino.
alias	String	Alias del certificado con el que recuperaremos del almacén de certificados el que se utilizará para realizar la firma del mensaje SOAP.
keystoreCallbackHandler	String	Nombre de la clase que implementa el CallbackHandler que proporcionará el password del almacén de certificados. P. ej. Para una clase llamada KeyStoreCallbackHandler que implemente la interfaz CallbackHandler, la forma de obtener el valor para este parámetro será: KeyStoreCallbackHandler.class.getName()
headers	Map<String, String>	Mapa que almacena las cabeceras adicionales que serán añadidas al mensaje SOAP.

A continuación se muestra un ejemplo de invocación a un servicio de Platino, donde se añaden las cabeceras de seguridad:

```
FirmaService firmaService = new FirmaService();
PlatinoSignatureServerBean service = firmaService.getFirmaService();
PlatinoCXFSecurityHeaders.addSoapWSSHeader(
    service,
    PlatinoCXFSecurityHeaders.SOAP_11,
    "platino://gobcan.es/servicios/terceros/t
ercero/4525e5da-526f-42d4-bf9b-64bb02b98399",
    "cert-alias"
    KeyStoreCallbackHandler.class.getName(),
```

 <p>Gobierno de Canarias Consejería de Presidencia, Justicia y Seguridad Dirección General de Telecomunicaciones y Nuevas Tecnologías</p>	 <p>Platino Plataforma de Interoperabilidad del Gobierno de Canarias</p>
Acceso a los servicios en pre-explotación	Página 17 de 23

```

null);
String version = service.getVersion();

```

Si además fuese necesario inyectar algún parámetro adicional en la cabecera del mensaje, debemos hacer algo equivalente al siguiente ejemplo de código:

```

FirmaService firmaService = new FirmaService();
PlatinoSignatureServerBean service = firmaService.getFirmaService();
Map<String,String> headers = new HashMap<String,String>();
headers.put("cabecera1", "valor1");
headers.put("cabecera2", "valor2");
PlatinoCXFSecurityHeaders.addSoapWSSHeader(
    service,
    PlatinoCXFSecurityHeaders.SOAP_11,
    "platino://gobcan.es/servicios/terceros/tercero/4525e5da-526f-42d4-bf9b-64bb02b98399",
    "cert-alias"
    KeyStoreCallbackHandler.class.getName(),
    headers);
String version = service.getVersion();

```

4.7.2 FICHERO CLIENT_SIGN.PROPERTIES

El fichero contiene las siguientes propiedades:

Propiedad	Descripción
<code>org.apache.ws.security.crypto.provider</code>	Proveedor criptográfico
<code>org.apache.ws.security.crypto.merlin.keystore.type</code>	Formato del almacén de certificados
<code>org.apache.ws.security.crypto.merlin.keystore.password</code>	Password del almacén de certificados
<code>org.apache.ws.security.crypto.merlin.file</code>	Ubicación del almacén de certificados



```

    </ds:DigestValue>
  </ds:Reference>
  <ds:Reference URI="#id-21928017"
    xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:Transforms
      xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
      <ds:Transform
        Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"
        xmlns:ds="http://www.w3.org/2000/09/xmldsig#" />
      </ds:Transforms>
      <ds:DigestMethod
        Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"
        xmlns:ds="http://www.w3.org/2000/09/xmldsig#" />
      <ds:DigestValue
        xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        L/G56fZQzKClXXIO+FRjQr2iBbo=
      </ds:DigestValue>
    </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    TGf0YQiUC/BQG0zZ0amj1tGX0UsZ0gQO9dX+XopC7brT8d/G9IMQhxmN6YG26x0Ty5T2LKAVkoQA
    9LS6QQ+badmEWlfjVj71/hjJYTMzCJYgyTWyFF5WoBnMzkjbiMUQRMiUy27fhV+THKXxOyZ8O8Yn
    acSze0RYMsOLpADciOQ=
  </ds:SignatureValue>
  <ds:KeyInfo Id="KeyId-7718724"
    xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <wsse:SecurityTokenReference
      xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
      wsu:Id="STRId-10816932"
      xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd">
      <wsse:Reference URI="#CertId-22603721"
        ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-
token-profile-1.0#x509v3"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" />
      </wsse:SecurityTokenReference>
    </ds:KeyInfo>
  </ds:Signature>
  <wsse:UsernameToken
    xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd"
    wsu:Id="UsernameToken-26384885"
    xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
    <wsse:Username
      xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">

```

 Gobierno de Canarias Consejería de Presidencia, Justicia y Seguridad Dirección General de Telecomunicaciones y Nuevas Tecnologías	 Platino Plataforma de Interoperabilidad del Gobierno de Canarias
Acceso a los servicios en pre-explotación	Página 21 de 23

64bb02b98399 platino://gobcan.es/servicios/terceros/tercero/4525e5da-526f-42d4-bf9b-

```

    </wsse:Username>
    <wsse:Password
      Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-
profile-1.0#PasswordText"
      xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
seceext-1.0.xsd">
    </wsse:Password>
    <wsse:Nonce
      xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
seceext-1.0.xsd">
      CuU2g5Zq9r/ijr91yETl2Q==
    </wsse:Nonce>
  </wsse:UsernameToken>
</wsse:Security>
<platino:platinoHeaders xmlns:platino="http://platino.gobcan.es"
  xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd"
  wsu:Id="id-21928017">
  <platino:parameter name="cabecera1">valor1</platino:parameter>
  <platino:parameter name="cabecera2">valor2</platino:parameter>
</platino:platinoHeaders>
</soap:Header>
<soap:Body
  xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd"
  wsu:Id="id-10265083">
  <ns2:getVersion
    xmlns:ns2="http://platino.gobcan.es/servicios/sfst/" />
  </soap:Body>
</soap:Envelope>

```

4.8 Secuencia de pasos

La secuencia de pasos que sigue una petición con esta nueva filosofía es la siguiente:

1. Se obtiene el objeto de servicio del cliente CXF: `FirmaService firmaService = new FirmaService(); PlatinoSignatureServerBean service = firmaService.getFirmaSecService();`

2. Se solicita la adición de las cabeceras de seguridad de Platino: `PlatinoCXFSecurityHeaders.addSoapWSSHeader(service, PlatinoCXFSecurityHeaders.SOAP_11,`

 <p>Gobierno de Canarias Consejería de Presidencia, Justicia y Seguridad Dirección General de Telecomunicaciones y Nuevas Tecnologías</p>	 <p>Platino Plataforma de Interoperabilidad del Gobierno de Canarias</p>
<p>Acceso a los servicios en pre-explotación</p>	<p>Página 22 de 23</p>

```

"platino://gobcan.es/servicios/terceros/tercer
o/4525e5da-526f-42d4-bf9b-64bb02b98399",
"cert-alias",
ClientPasswordHandler.class.getName(),
null);

```

3. Se realiza la invocación:

```
String version = service.getVersion();
```

4. Automáticamente se ejecutan los Interceptors que añaden las cabeceras de seguridad:

- 4.1. Se crea el elemento wsse:UsernameToken
- 4.2. Se accede al certificado y se genera el wsse:BinarySecurityToken
- 4.3. Se crea el elemento platino:platinoHeaders con las cabeceras proporcionadas.
- 4.4. Se firma el elemento wsse:UsernameToken, platino:platinoHeaders y soap:Body

5. La petición llega a Platino

6. Proceso de autenticación y autorización:

- 6.1. Se comprueba la validez de la firma
- 6.2. Se comprueba que el certificado con el que se ha firmado pertenece a un consumidor registrado de PLATINO
- 6.3. La petición está autenticada
- 6.4. Se ejecutan las reglas de autorización para comprobar que tiene permisos para realizar la petición
- 6.5. La petición está autorizada

Si el proceso fallase en el punto 6.1, 6.2 o 6.4, el proceso de autorización devolverá un error indicando que la petición ha sido rechazada.

 Gobierno de Canarias Consejería de Presidencia, Justicia y Seguridad Dirección General de Telecomunicaciones y Nuevas Tecnologías	 Platino Plataforma de Interoperabilidad del Gobierno de Canarias
Acceso a los servicios en pre-explotación	Página 23 de 23

5 ANEXOS

5.1 Consulta del alias del certificado almacenado en un PFX

Java proporciona una utilidad llamada 'keytool' (\$JAVA_HOME/bin/keytool) que permite, entre otras muchas cosas, listar el contenido de un PFX. Para ello debemos invocar la herramienta con los siguientes parámetros:

```
> keytool -list -v -keystore <fichero.pfx> -storetype PKCS12
```

A continuación nos solicitará el password del certificado, y nos mostrará una pantalla como la siguiente:

```
Tipo de almacén de claves: PKCS12
Proveedor de almacén de claves: SunJSSE

Su almacén de claves contiene entrada 1
Nombre de alias: ad9a0c6f8127a027e1f977ea1b7d3d43_7f64a424-6531-4367-b918-6e8f5ffd2290
Fecha de creación: 18-sep-2008
Tipo de entrada: keyEntry
Longitud de la cadena de certificado: 2
Certificado[1]:
Propietario: CN=ext-jpadlor, OU=DGTI, O=Gobierno de Canarias, ST=Canarias, C=ES, EMAILADDRESS=ext-jpadlor@canarias.org
Emisor: CN=cibercentro, OU=DGTI, O=Gobierno de Canarias, ST=Canarias, C=ES, EMAILADDRESS=cibercentro@canarias.org
Número de serie: 3dc825b3000200000f4e
Válido desde: Mon Jul 14 09:56:38 BST 2008 hasta: Tue Jul 14 10:06:38 BST 2009
Huellas digitales del certificado:
MD5: 4F:71:B3:43:C8:C5:12:B8:13:8D:FA:B8:7C:BF:1E:2A
SHA1: 96:2A:16:76:19:2C:4A:22:F9:6B:0E:F4:A8:9E:14:3C:27:BA:29:A6
Certificado[2]:
Propietario: CN=cibercentro, OU=DGTI, O=Gobierno de Canarias, ST=Canarias, C=ES, EMAILADDRESS=cibercentro@canarias.org
Emisor: CN=cibercentro, OU=DGTI, O=Gobierno de Canarias, ST=Canarias, C=ES, EMAILADDRESS=cibercentro@canarias.org
Número de serie: 1966c7766cc265ae4084880712eb4a03
Válido desde: Wed Jan 23 21:48:21 GMT 2002 hasta: Wed May 11 14:59:19 BST 2011
Huellas digitales del certificado:
MD5: DE:05:09:88:AB:F6:00:7A:63:7E:C4:CE:B1:A6:7D:96
SHA1: 9D:49:4F:34:E6:CF:53:BB:4F:F2:A3:4B:1A:4F:44:C2:B2:97:37:C6
```

El nombre del alias que debemos usar para pasárselo a la petición sería en este ejemplo el valor:

```
"ad9a0c6f8127a027e1f977ea1b7d3d43_7f64a424-6531-4367-b918-6e8f5ffd2290"
```