

**DOCUMENTOS TÉCNICOS
DIVISIÓN INFORMÁTICA
DESARROLLO**



**Servicios Web Externos DGI
Factura Electrónica**

CÓDIGO: T-5.020.00.001-000005

CÓDIGO:	T-5.020.00.001-000005	DOCUMENTOS TÉCNICOS INFORMÁTICA – DESARROLLO <i>Servicios Web Externos DGI</i>
VERSIÓN:	1.9	
FECHA:	13/05/2024	

Índice

Índice.....	2
Actualizaciones.....	3
Introducción	3
Servicios Disponibles.....	3
WS_eFacSolicitud / EFACSLACTUALIZARCONTACTO.....	5
Desarrollo de un cliente para nuestro webservice.....	6
Pruebas previas	6
Ejemplo de Pruebas con SoapUI	6
WS-Security.....	7

CÓDIGO:	T-5.020.00.001-000005	DOCUMENTOS TÉCNICOS INFORMÁTICA – DESARROLLO <i>Servicios Web Externos DGI</i>
VERSIÓN:	1.9	
FECHA:	13/05/2024	

Actualizaciones

- **Versión 1.0:** Se incluye el nuevo método:
EFACSLACTUALIZARCONTACTO : Solicitud de Actualización de Contactos del Emisor Electrónico(Códigos habilitados en esta versión para actualizar 19 : URL Webservice y 20 : Mail Contacto Técnico)
-

Introducción

DGI publica un Servicio Web externo para el manejo de solicitudes de factura electrónica.

Este Servicio Web utiliza un esquema de seguridad basado en “WS-Security” permitiendo así garantizar un nivel de seguridad tal -a través del uso de certificados y firmas digitales- que se logra cumplir con los siguientes requisitos de seguridad:

- Autenticación
- Confidencialidad (vía https)
- Integridad
- No repudio

Quien desarrolle un cliente externo de este Servicio Web deberá tener en cuenta el uso de los certificados digitales para poder lograr una invocación exitosa. Más adelante en este documento encontrará un ejemplo de invocación exitosa de nuestro Servicio Web hecho con SoapUI (www.soapui.org), una de las herramientas más difundidas para testeo de webservices.

Servicios Disponibles

El servicio publicado (ws_efacsolicitud) expone los siguientes métodos que resuelven el manejo de solicitudes:

Métodos del WebService	
Nombre	Descripción
EFACSLACTUALIZARCONTACTO	Actualiza el valor del contacto del Emisor Electrónico. (Códigos habilitados en esta versión para actualizar 19 : URL Webservice y 20 : Mail Contacto Técnico)

CÓDIGO:	T-5.020.00.001-000005	DOCUMENTOS TÉCNICOS INFORMÁTICA – DESARROLLO <i>Servicios Web Externos DGI</i>
VERSIÓN:	1.9	
FECHA:	13/05/2024	

Servicios Disponibles

Servicio	Método	Parám.Request	Parám.Respuesta Correcta
ws_efacsolicitud	EFACSLACTUALIZARCONTACTO	eFacSolActualizarContactoParm	eFacSolActualizarContactoRes

Para la invocación de los servicios, se definen los parámetros de entrada y salida como XML que debe contener en cada caso:

CÓDIGO:	T-5.020.00.001-000005	DOCUMENTOS TÉCNICOS INFORMÁTICA – DESARROLLO <i>Servicios Web Externos DGI</i>
VERSIÓN:	1.9	
FECHA:	13/05/2024	

WS_eFacSolicitud / EFACSOACTUALIZARCONTACTO

Dado el RUC, el Código de Contacto y el Valor del Contacto, genera una solicitud de Actualización de datos del contacto con el nuevo valor indicado y devuelve el estado de la operación indicando un Código y un Detalle. Los tres campos son obligatorios.

Entrada	Salida
<pre><dgi:WS_eFacSolicitud.EFACSOACTUALIZARCONTACTO> <dgi:Efacsoactualizarcontactoparm> <dgi:ContactoRUC>...</dgi:ContactoRUC> <dgi:ContactoCodigo>...</dgi:ContactoCodigo> <dgi:ContactoValor>...</dgi:ContactoValor> </dgi:Efacsoactualizarcontactoparm> </dgi:WS_eFacSolicitud.EFACSOACTUALIZARCONTACTO></pre>	<pre><WS_eFacSolicitud.EFACSOACTUALIZARCONTACTOResponse xmlns="http://dgi.gub.uy"> <Efacsoactualizarcontactores> <Codigo>...</Codigo> <Detalle>...</Detalle> </Efacsoactualizarcontactores> </WS_eFacSolicitud.EFACSOACTUALIZARCONTACTOResponse></pre>

-- Entrada

```
<dgi:Efacsoactualizarcontactoparm>
  <dgi:ContactoRUC>219999830019</dgi:ContactoRUC>
  <dgi:ContactoCodigo>20</dgi:ContactoCodigo>
  <dgi:ContactoValor>15_21_10@dgi.gub.uy</dgi:ContactoValor>
</dgi:Efacsoactualizarcontactoparm>
```

-- Salida

```
<Efacsoactualizarcontactores>
  <Codigo>0</Codigo>
  <Detalle>Contacto: 20 ( Mail Contacto Técnico ) para el RUC: 219999830019
actualizado con el valor: 15_21_10@dgi.gub.uy . Número solicitud generado:
202404000026</Detalle>
</Efacsoactualizarcontactores>
```

CÓDIGO:	T-5.020.00.001-000005	DOCUMENTOS TÉCNICOS INFORMÁTICA – DESARROLLO <i>Servicios Web Externos DGI</i>
VERSIÓN:	1.9	
FECHA:	13/05/2024	

Desarrollo de un cliente para nuestro webservice

Pruebas previas

Antes de desarrollar un programa cliente para el webservice de DGI, recomendamos hacer pruebas con la herramienta SoapUI (www.soapui.org) para hacer una validación básica del funcionamiento del servicio. De ese modo se podrán descartar antes los problemas que no pertenecen al desarrollo del cliente (ej: redes, validez del xml, etc.).

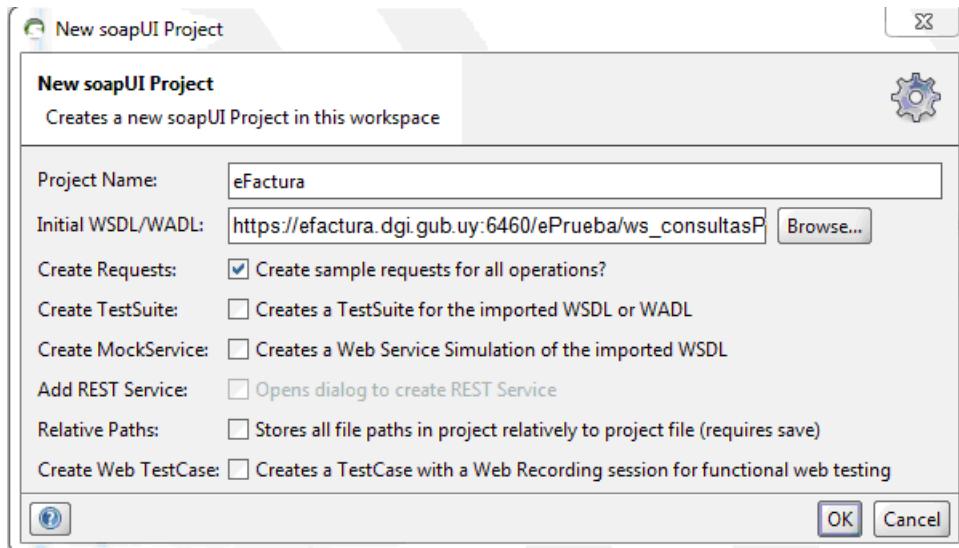
Una vez que se supere exitosamente esta prueba, resultará más accesible para el desarrollador implementar un cliente que interactúe del mismo modo que lo hace SoapUI con nuestro webservice.

Ejemplo de Pruebas con SoapUI

El WSDL del webservice para ambiente de TEST está publicado en:

https://efactura.dgi.gub.uy:6460/ePrueba/ws_consultasPrueba?wsdl

Una vez instalado SoapUI, se debe crear un nuevo proyecto de la siguiente manera:



CÓDIGO:	T-5.020.00.001-000005	DOCUMENTOS TÉCNICOS INFORMÁTICA – DESARROLLO	DGI DIRECCIÓN GENERAL IMPOSITIVA
VERSIÓN:	1.9		
FECHA:	13/05/2024	Servicios Web Externos DGI	

WS-Security

Dado que la información es confidencial, el sistema incorpora una estrategia de seguridad basada en el estándar WS-Security. Este estándar utiliza una infraestructura de clave pública (PKI o public key infrastructure).

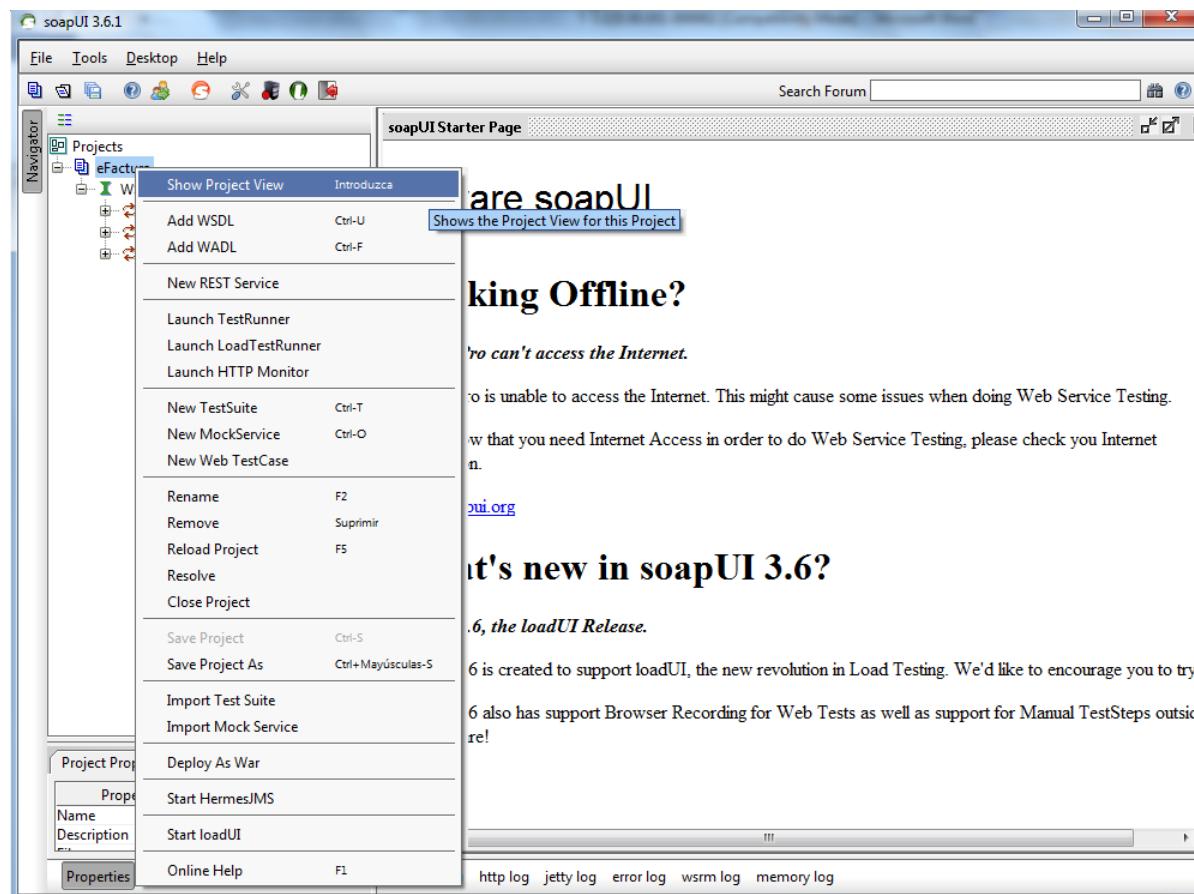
El certificado a utilizar podrá ser un "Certificado Nominado de Persona Jurídica" o "Certificado Innominado de Persona Jurídica" que al día de hoy solo emite el Correo del Uruguay (Única CA por lo pronto que formará parte de PKI Uruguay). La decisión de cuál utilizar será resorte de cada Empresa/Organismo.

Una vez obtenido el certificado, deberá utilizarse el mismo para firmar digitalmente las consultas realizadas.

La manera más sencilla de trabajar con WS-Security es utilizando certificados almacenados en keystores.

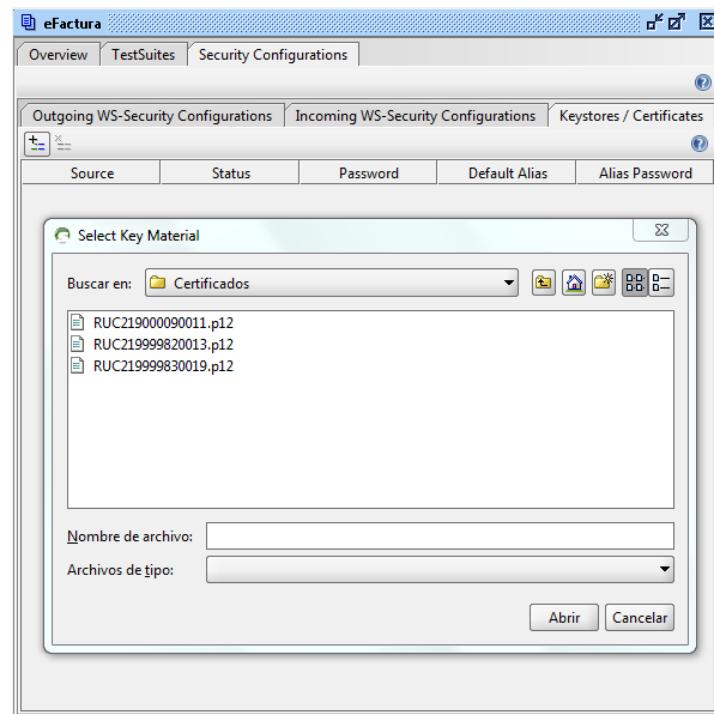
Una vez obtenidos los certificados brindados por la autoridad certificadora, los keystores se pueden crear mediante la aplicación Portecle, disponible en (<http://portecle.sourceforge.net>) o mediante openssl (<http://www.openssl.org/>).

Para poder utilizar firma digital en SoapUI, debemos hacer los siguientes ajustes:

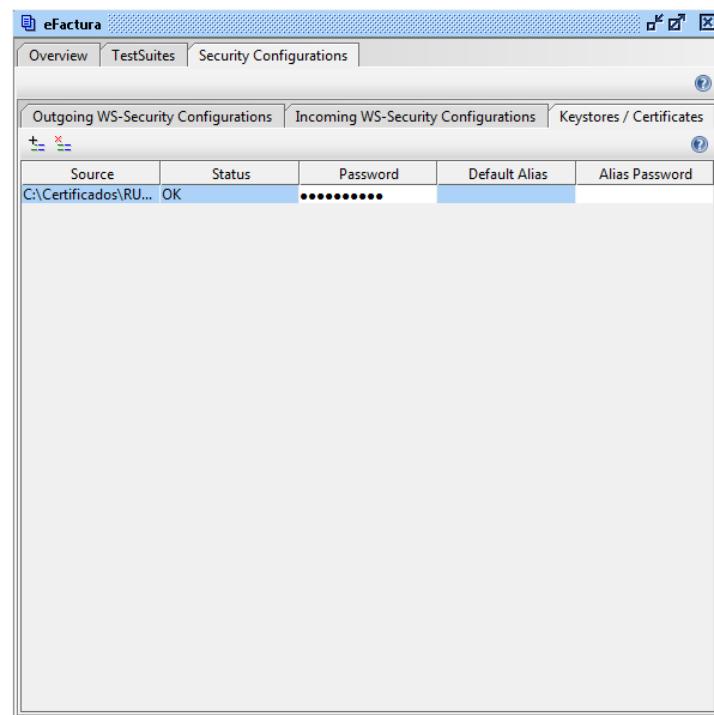


Agregamos el keystore:

CÓDIGO:	T-5.020.00.001-000005	DOCUMENTOS TÉCNICOS INFORMÁTICA – DESARROLLO	DGI DIRECCIÓN GENERAL IMPOSITIVA
VERSIÓN:	1.9		
FECHA:	13/05/2024	Servicios Web Externos DGI	

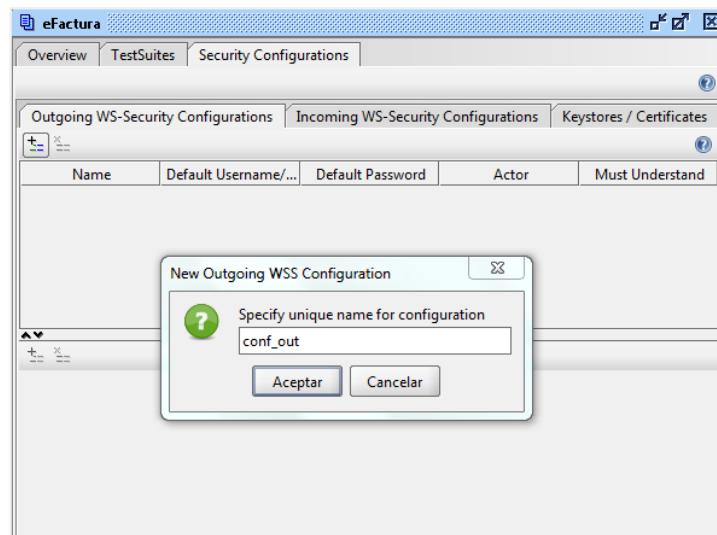


Ingresamos la contraseña necesaria para abrir el archivo KeyStore

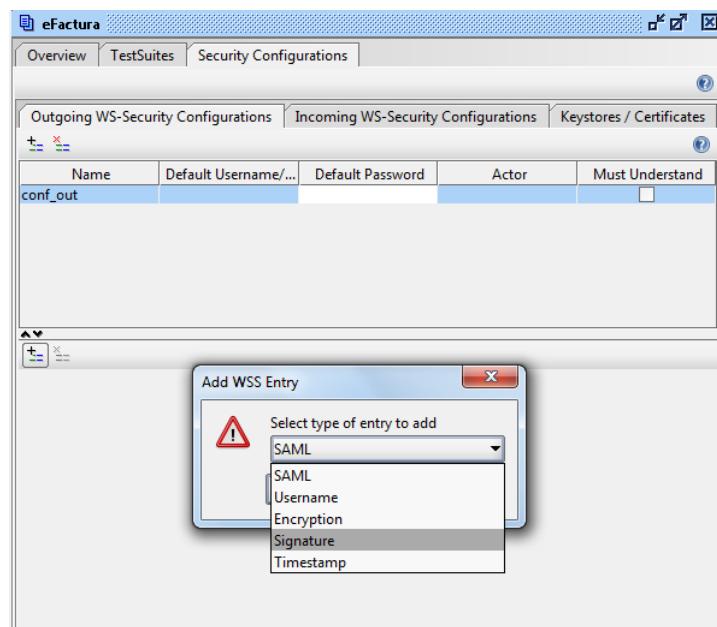


Creamos una configuración de salida:

CÓDIGO:	T-5.020.00.001-000005	DOCUMENTOS TÉCNICOS INFORMÁTICA – DESARROLLO	DGI DIRECCIÓN GENERAL IMPOSITIVA
VERSIÓN:	1.9		
FECHA:	13/05/2024	Servicios Web Externos DGI	

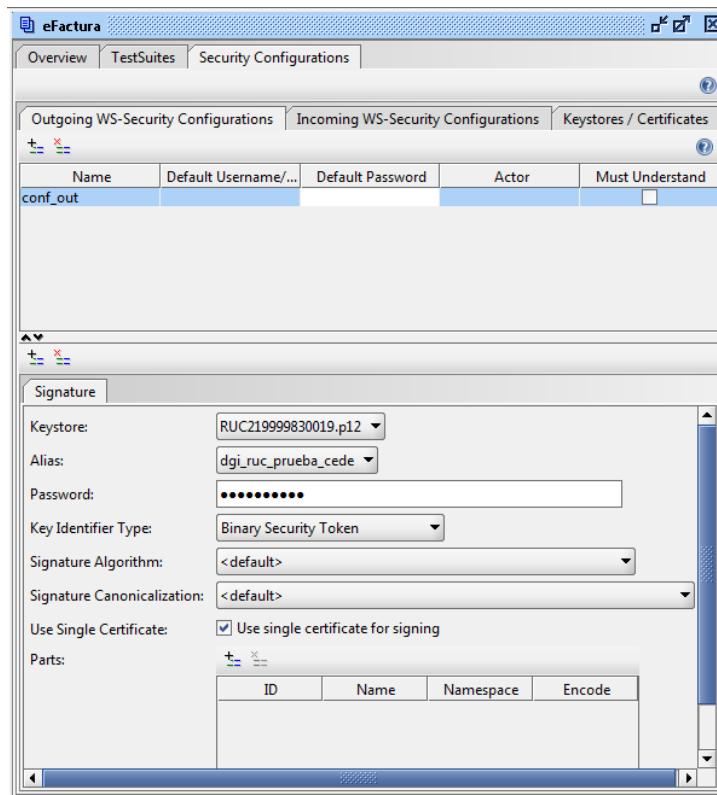


Seleccionamos que en esa configuración se agregue firma digital (signature):



En esta pantalla nos pide el nombre del keystore (seleccionamos el que agregamos antes), el alias es el nombre que le dimos a la clave privada del cliente dentro del keystore. El resto de los parámetros se configuran como se ve en la imagen.

CÓDIGO:	T-5.020.00.001-000005	DOCUMENTOS TÉCNICOS INFORMÁTICA – DESARROLLO	DGI DIRECCIÓN GENERAL IMPOSITIVA
VERSIÓN:	1.9		
FECHA:	13/05/2024	Servicios Web Externos DGI	



Nota importante: Si en SoapUI creamos un request antes de configurar estos pasos de seguridad, luego de configurar la seguridad debemos borrar el request y crear uno nuevo para que tome los cambios.

CÓDIGO:	T-5.020.00.001-000005	DOCUMENTOS TÉCNICOS INFORMÁTICA – DESARROLLO	DGI DIRECCIÓN GENERAL IMPOSITIVA
VERSIÓN:	1.9	Servicios Web Externos DGI	

Vea que en la zona inferior del request aparece una opción “Aut” (Authentication)

Property	Value
Name	Request 1
Description	366
Message Size	UTF-8

Username: _____
 Password: _____
 Domain: _____
 Outgoing WSS: _____
 Incoming WSS: _____

Aut Headers (0) Attachments (0) WS-A WS-RM JMS Headers JMS Properties (0) Headers (0)

soapUI log http log jetty log error log wsrm log memory log

Al seleccionarla nos permite elegir la configuración de salida (Outgoing WSS) que creamos antes (“conf_out” era el nombre en nuestro ejemplo).

Sustituyendo los signos de interrogación por sus valores en el request ya podemos invocar al webservice utilizando la firma digital correspondiente.