Manual del Programador del módulo de Firma web del Servidor @Firma

Versión 4.0

Documento nº:	TI-20-1074-MPF-001
Revisión:	01
Fecha:	06/08/2004
Período de retención:	Permanente durante su período de vigencia + 3 años después de su anulación

CONTROL DE COMPROBACIÓN Y APROBACIÓN

Documento nº: TI-20-1074-MPF-001

Revisión: 1

Fecha: 06/08/2004

REALIZADO

06/08/2004

Moisés Manuel

Infante

Gómez

Analista Programador

COMPROBADO

06/08/2004

José Antonio

Márquez

Contreras

Director @firma

APROBADO

06/08/2004

José Antonio

Márquez

Contreras

Director @firma

CONTROL DE MODIFICACIONES

Documento nº: TI-20-1074-MPF-001

Revisión: 1

Fecha: 06/08/2004

Rev.	1
Fecha	06/08/2004
Autor/es	MMIG
Descripción	Documentación inicial

CONTROL DE DISTRIBUCIÓN

Documento nº: TI-20-1074-MPF-001

Revisión: 1

Fecha: 06/08/2004

Copias Electrónicas:

La distribución de este documento ha sido controlada a través del sistema de información.

Copias en Papel:

La vigencia de las copias impresas en papel está condicionada a la coincidencia de su estado de revisión con el que aparece en el sistema electrónico de distribución de documentos. El control de distribución de copias en papel para su uso en proyectos u otras aplicaciones es responsabilidad de los usuarios del sistema electrónico de información. Fecha de impresión 17/09/04 13:20

Distribución en Papel:

Nombre o Cargo y (Organización)	Nº de Ejemplares	Referencia de la carta de transmisión y fecha

Índice

1 Objeto

El objeto de este documento es describir la utilización del módulo de Firma Web del Servidor @Firma.

El módulo de Firma Web contiene las interfaces y componentes web necesarios para la realización de los siguientes procesos:

- Firma Paginas Web.
- MultiFirma Paginas Web.
- MultiFirma Web Masiva.
- Consulta de Transacciones y Verificación de Firmas.

Los objetivos globales de este proceso son:

- Describir los pasos necesarios para desarrollar una aplicación que utilice las interfaces RMI-IIOP y WebServices, asi como los componentes web disponibles en la plataforma de firma.
- Describir los métodos disponibles en las interfaces anteriores y la lógica de utilización de los mismos.
- Describir detalladamente el ejemplo de utilización de dichas interfaces y componentes web que se adjunta en la plataforma para facilitar el trabajo al nuevo desarrollador de aplicaciones, tanto RMI-IIOP como WebServices.
- Especificar los diferentes tipos de errores que se pueden dar al integrar una aplicación con la plataforma, y su resolución.

2 Alcance

El presente documento recoge la utilización del módulo de Firma Web del Servidor @Firma.

El módulo de Firma Web contiene las interfaces y componentes web necesarios para la realización de los siguientes procesos:

- Firma Paginas Web.
- MultiFirma Paginas Web.
- MultiFirma Web Masiva.
- Consulta de Transacciónes y Verificación de Firmas.

3 Siglas

AC	Autoridad de Certificación
CPD	CRL Distribution Point
CRL	Lista de Revocación de Certificados
FNMT-RCM	Fábrica Nacional de Moneda y Timbre, Real Casa de la Moneda
JSP	JavaServer Pages
LDAP	Lightweight Directory Access Protocol
PC	Ordenador Personal
RSA	Rivest Shamir Adleman
JRE	Java Runtime Environment
JDK	Java Development Kit
PKCS#7	Public Key Cryptography Standard Number 7
ASN.1	Abstract Syntax Notation One
EJB	Enterprise Java Bean
SSL	Secure Socket Layer

4 Documentos de Referencia

- Documento Estándar ASN.1 de la Estructura de Firma Electrónica
- Documento Estándar ASN.1 de la Estructura de Acuse de Recibo

5 Introducción

El módulo de Firma web es una herramienta de Firma Digital basada en Tecnología Java que puede ser integrada en aplicaciones ya existentes o que se vayan a desarrollar. Utiliza certificados digitales (X.509) para firmar los datos digitalmente.

El módulo de Firma Web contiene las interfaces y componentes web necesarios para la realización de los siguientes procesos:

- Firma Paginas Web.
- MultiFirma Paginas Web.
- MultiFirma Web Masiva.
- Consulta de Transacciónes y Verificación de Firmas.

Las interfaces se encuentras disponibles mediante tecnología RMI-IIOP y WEBSERVICES, ambas securizadas mediante SSL y JAAS, lo cual proporciona un doble nivel de seguridad.

5.1 Sobre módulo Firma Web

En el caso de Firma Páginas Web, en una primera fase se pide la página web al Servidor de Firma, este la pide al Servidor de Aplicaciones correspondiente, la registra en el sistema de Custodia y la convierte en firmable. Posteriormente es enviada al usuario el cual la rellena y firma digitalmente desde su propia máquina. La página, la firma y los datos asociados a la transacción de firma quedan almacenados en el sistema de Custodia de la plataforma.

En el caso de MultiFirma Páginas Web, partiendo del identificador de una transacción de firma o multifirma anterior, se recupera la página firmada y se inicia un nuevo proceso de multifirma, el cual genera una información que será firmada por el usuario (referida a la transacción indicada por el identificador anterior). Se permiten dos tipos de multifirma, en paralelo (CoSign) o jerárquica (CounterSign). Este tipo de multifirma está indicado para los casos en que no haya un número excesivo de firmas, ya que todas las firmas se guardan en la estructura PKCS#7 y ésta crecería demasiado. La firma y los datos asociados a la transacción de firma quedan almacenados en el sistema de Custodia de la plataforma.

En el caso de MultiFirma Web Masiva, primero se registra una página web estática mediante la Herramienta de Administración, firmándose digitalmente por el Servidor de Firma en el momento del registro utilizando un certificado digital configurado para ello y se obtiene un identificador de transacción. Con dicho identificador, la aplicación a integrar este tipo de firma, pide la página al sistema el cuál la devuelve con la información a firmar por el usuario (referida a la firma realizada por el servidor), el usuario firma la información digitalmente desde su propia máquina y se devuelve al Servidor de Firma. Este tipo de multifirma está indicado para los casos en los que se vaya a firmar la misma información por un gran número de personas, ya que en cada PKCS#7 sólo se guarda la firma del servidor y la firma de un usuario. La firma y los datos asociados a la transacción de firma quedan almacenados en el sistema de Custodia de la plataforma.

La plataforma proporciona las interfaces RMI-IIOP y WEBSERVICES necesarios para Consultar toda la información disponible sobre cualquier transacción de firma realizada y proceder a la Verificación de cualquier firma.

Con la plataforma de Firma se distribuyen ejemplos que muestran la utilización todas las interfaces y componentes mencionadas. A lo largo de este documento se describen todas las interfaces disponibles y los ejemplos de utilización de las mismas.

Adicionalmente, para facilitar la tarea del desarrollador / integrador de aplicaciones se distribuye el "javadoc" correspondiente a todas las interfaces, en el cual se detallan los métodos, parámetros, excepciones, etc. Esto permitirá que el presente manual se centre principalmente en el funcionamiento omitiendo detalles específicos.

5.2 Sobre Firma Avanzada

Cada aplicación de firma que se integra en la plataforma de firma permite configurar el tipo de Firma Digital que se desea realizar mediante un parámetro. La Firma puede ser básica, en la que solamente se genera la estructura PKCS#7 o avanzada en la que también interviene un Notario Electrónico. A continuación se describen los modos de funcionamiento disponibles:

1- MODO SERVIDOR BÁSICO 0: genera la estructura PKCS#7

2- MODO SERVIDOR AVANZADO 1: genera la estructura PKCS#7 y la Estructura de Firma Electrónica ASN.1 con TimeStamp de Servidor de Notario Electrónico.

3- MODO SERVIDOR AVANZADO 2: genera la estructura PKCS#7, la Estructura de Firma Electrónica ASN.1 con TimeStamp de Servidor de Notario Electrónico y la Estructura de Acuse de Recibo ASN.1 de Servidor de Notario Electrónico.

4- MODO SERIVDOR AVANZADO 3: genera la estructura PKCS#7 y la Estructura de Firma Electrónica ASN.1 con TimeStamp local del Servidor de Firma.

5.3 Interfaces del módulo de Firma Web

A continuación se muestran los nombres de las interfaces disponibles en la plataforma para el módulo de Firma de Web y se describen el ámbito de cada una de ellas.

- Interfaz com.telventi.ejb.FirmaApiFacade

  Permite la consulta de información de transacciones de Firma Web realizadas.

- Interfaz com.telventi.multifirma.MultiFirmaApiFacade

  Permite realizar el proceso MultiFirma Web. Adicionalmente contiene los métodos de consulta específicos de la MultiFirma Web. Se agrupan en esta interfaz por cuestiones de eficiencia en el desarrollo de aplicaciones.

- Interfaz com.telventi.multifirmamasiva.MultiFirmaWebMasiva

Permite realizar el proceso MultiFirma Masiva. Adicionalmente contiene los métodos de consulta específicos de la MultiFirma Masiva. Se agrupan en esta interfaz por cuestiones de eficiencia en el desarrollo de aplicaciones.

5.4 Incorporación de mecanismo de Detección de componente-Cliente en una aplicación

El componente-Cliente se debe instalar en cada máquina cliente desde la cual se deseen realizar transacciones de firma sobre páginas firmables y aplicaciones de multifirma web integradas en el servidor @Firma.

Cuando se integra una aplicación web en el servidor @Firma, es importante incluir un mecanismo de detección de componentes instalados en maquina cliente en una fase temprana de la aplicación web.

Los requisitos mínimos en el cliente son los siguientes:

1. Sistema Operativo Windows: Navegador IE 5.0 o superior o Netscape 4.78 o superior. Plugin de Java JRE 1.4.0 o superior o JVM 1.1.4 presente por defecto en Internet Explorer.

   Sistema Operativo Linux: Navegador Mozilla 1.3 o superior. Plugin de Java JRE 1.4.0 o superior.

2. Componente-Cliente Firma. Consiste en unas librerías que permiten recuperar los certificados almacenados en el sistema operativo de la máquina cliente y firmar en tarjetas criptográficas. Windows: telventsign.dll y telventmultisign.dll. Linux: jss33 de mozilla.

Un mecanismo de detección comprueba si estos componentes están instalados en la máquina cliente y si no están presentes se redirecciona el navegador a una página que contiene links para descargar los componentes necesarios.

El mecanismo consiste en el script *Detector.js*, que debe ser colocado en una página que se muestre al usuario en una fase temprana de la aplicación web que se desea hacer firmable o multifirmable. En otras palabras, el chequeo debe ser realizado antes de que la página firmable o la página que realice la multifirma en cuestión sea mostrada al cliente. Esto es importante a su vez porque es necesario cerrar el navegador una vez ha sido instalado el Plugin de Java. Si se realiza el chequeo en una fase temprana, el flujo de navegación no será interrumpido.

Los pasos necesarios a seguir para instalar el mecanismo de detección serán los siguientes:

1. Copiar los ficheros *TelventDetect.jar, TelventDetect.cab, TelventDetectLinux.jar, Detector.js, XPInstall.jar, XPInstall.cab y XPMensaje.htm* que se encuentran en el directorio del CD Desarrollo */ModuloFirma/Firma Web/HabilitarDetección* en el directorio de nuestra nueva aplicación firmable. En concreto, al mismo lugar donde se encuentre la página que incorporará el mecanismo.

2. Añadir el siguiente código a la página web de la nueva aplicación web que se encargará de realizar la detección:

   <script language="Javascript" SRC="Detector.js"></script>

3. Se deben realizar los siguientes cambios en el fichero *Detector.js*:

   a) En primer lugar se debe localizar la variable *nextURL* y asignarle como valor el nombre de la página de nuestra nueva aplicación a la que será redireccionado el navegador si todos los componentes necesarios están instalados. En caso de asignar el valor cadena vacía (" "), el destino será simplemente la página actual.

      Var nextURL = "comienzo.htm";

   b) En segundo lugar se debe localizar la variable *url_base* y asignarle como valor la URL base de la Fachada de @Firma.

      Var url_base = "https://hostname>:<port>/firmadigital

      Donde <*hostname*> es el nombre de la fachada y <*port*> el puerto de conexión de la Fachada @Firma.

4. Se deben realizar los siguientes cambios en el fichero *XPMensaje.htm*.

   a) En primer lugar se debe localizar la variable *pagina_siguiente_aplicación* y asignarle como valor el nombre de la página de nuestra nueva aplicación a la que será redireccionado el navegador si todos los componentes necesarios están instalados.

      Var pagina_siguiente_aplicación = "comienzo.htm"

   b) En segundo lugar se debe localizar la variable *fachada_firma* y asignarle como valor la URL base de la Fachada @Firma

      Var fachada_firma = "https://hostname>:<port>"

6 Proceso Firma Páginas Web

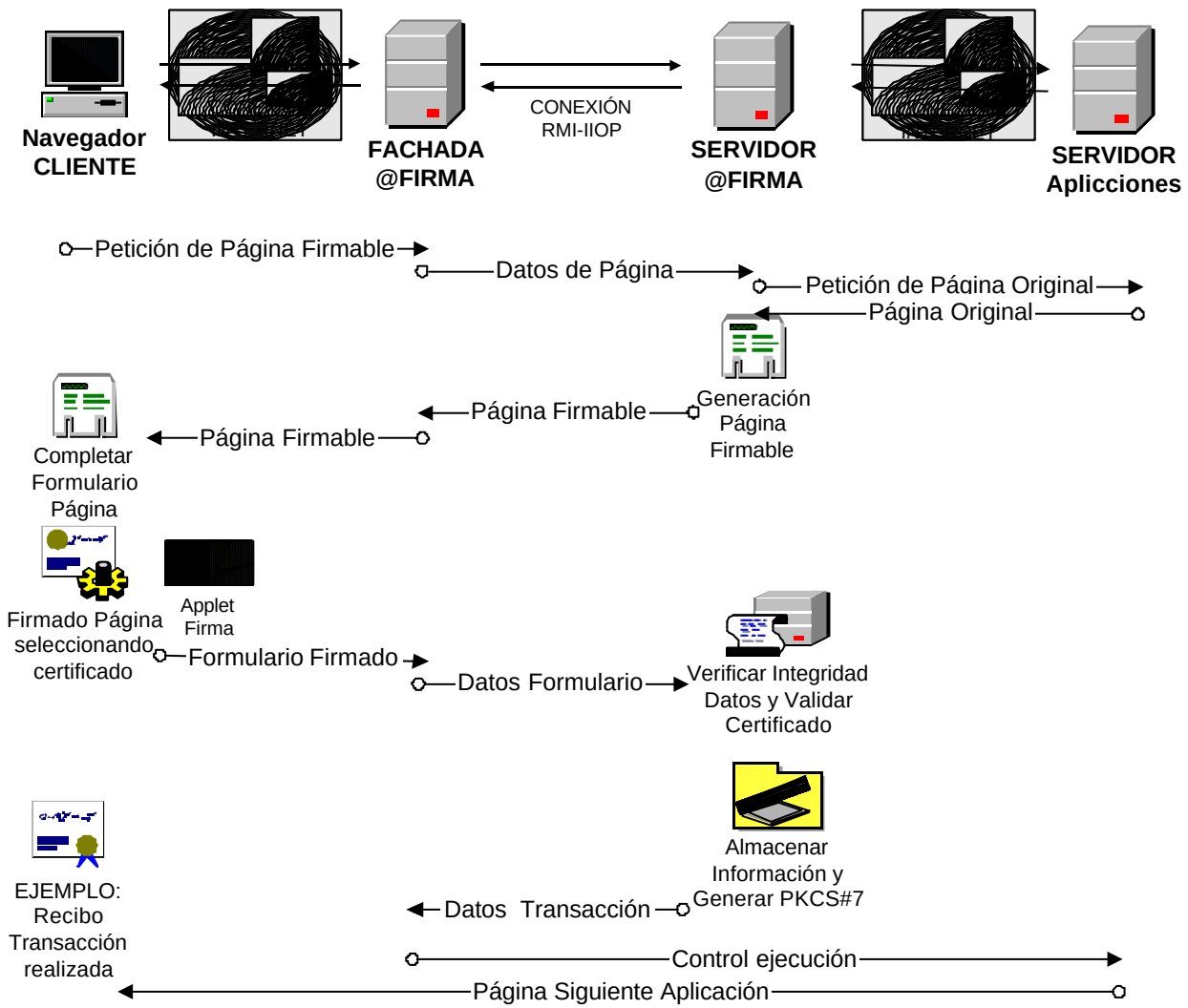
La plataforma de Firma proporciona los mecanismos necesarios para firmar digitalmente una página web perteneciente a una aplicación web. Los desarrolladores deberán seguir una serie de pasos sencillos para la integración de la firma en su aplicación.

La herramienta consiste en un componente-Cliente en forma de plug-in autodescargable que habilita a los usuarios para firmar digitalmente el contenido de una PÁGINA WEB y un componente-Servidor que procesa y verifica las firmas y datos recibidos.

El componente-Servidor utiliza un sistema de Custodia para recoger toda la información necesaria en el proceso de firmado de cada página.

La herramienta permite firmar el contenido de una página WEB. Cualquier tipo de documento que no sea una página WEB se puede incluir en el proceso de firmado como un fichero adjunto en el formulario HTML que se firma.

La siguiente figura resume de forma global el proceso de firma:



A continuación se describe el proceso de firma mostrado en la figura anterior:

1. En primer lugar, el navegador cliente solicita a la Fachada @Firma una página web firmable mediante conexión HTTPS.

2. La Fachada @Firma le transmite, mediante RMI-IIOP, al Servidor @Firma los datos de la página solicitada.

3. El Servidor @Firma genera la página firmable. Para ello recupera la página web original del servidor de aplicaciones correspondiente mediante una conexión HTTPS y realiza las modificaciones pertinentes a la misma. Por otro lado, inicia un proceso de firmado almacenando en el Sistema de Custodia la información adecuada, que consiste en la página HTML modificada, las imágenes de la misma e información relativa a la nueva transacción de firmado.

4. El Servidor @Firma envía la página generada a la Fachada @Firma.

5. El navegador cliente recibe la página web firmable generada por el servidor @Firma.

6. El usuario rellena los datos oportunos en el formulario recibido y pulsa el botón de firmado. En este momento se inicia el Applet Cliente que muestra al usuario una lista con los certificados disponibles en el sistema operativo de la máquina cliente. Cuando el usuario selecciona un certificado, se firma el contenido de la página con el certificado elegido y se envía el formulario incluyendo la firma y el certificado firmador al Servidor @Firma. En caso de existir solamente un certificado en la máquina cliente se firma directamente con ese certificado sin preguntar. Cabe destacar que se firma un resumen calculado al contenido del documento. También se incluye en la firma el HASH calculado a la página HTML.

7. La Fachada @Firma recibe el formulario firmado, obtiene todos los datos del formulario y se los transmite al Servidor @Firma.

8. El Servidor @Firma en primer lugar verifica la integridad de los datos y valida el certificado firmador. En concreto, se comprueba si el certificado ha expirado, si es de confianza y si está revocado. Si el certificado no es válido la transacción de firma se considera inválida y se informa al usuario. A continuación almacena en el Sistema de Custodia la información recibida y genera la estructura PKCS#7 correspondiente, la Estructura de Firma Electrónica ASN.1 y la Estructura de Acuse de Recibo ASN.1. Estos dos últimos elementos los construye basándose en un Servidor de Notario Electrónico.

9. El Servidor @Firma devuelve como resultado de la transacción de firma el identificador si ha sido correcta o un código de error en caso contrario.

10. La Fachada @Firma, en este momento devuelve el control al Servidor de Aplicaciones correspondiente. Pero antes de ello dispone de un API para recuperar toda la información que se desee acerca de la transacción de firmado realizada. En un ejemplo que se incorpora en la plataforma de firma, se genera mediante este API un recibo informativo acerca de la transacción realizada, informando acerca de la integridad de los datos y del certificado firmador.

11. El navegador cliente recibe la página siguiente de la aplicación original resuelta por el servidor de aplicaciones.

12. De esta forma termina un proceso de transacción de firma.

Es importarte tener claro que solamente firma contenido de documentos web (incluyendo hash de página HTML), pero mediante campos de tipo FILE se pueden ADJUNTAR el número que se desee de ficheros externos a la página firmable, los cuales serán incluidos en la firma.

La filosofía de funcionamiento de la herramienta es convertir una página web normal en una página web firmable. Para ello se debe integrar en la herramienta mediante una serie de pasos sencillos. En este manual se describe detalladamente el proceso Integración de una aplicación web en el servidor @Firma.

De esta forma para que una página web cualquiera pueda ser firmada, primero ha tenido que ser incluida en la herramienta. Así pues, queda fuera del alcance de la herramienta poder firmar una página cualquiera que no se ha integrado correctamente.

Por otro lado, es IMPORTANTE destacar el hecho de que la página a convertir en firmable puede ser una página HTML, JSP, ASP, PHP, etc. debido a que cuando el Servidor @firma recupera la página original para convertirla en firmable lo hace mediante una conexión HTTPS, como consecuencia se resuelve en el servidor correspondiente y el Servidor @firma recibe código HTML plano. Véase apartado "Restricciones de generación de página firmable" de este manual para más información acerca de cómo deben ser las páginas que se convierten en firmables digitalmente.

El componente Servidor @Firma almacena en el Sistema de Custodia la siguiente información:

- Página HTML firmable, incluidas las imágenes.
- Contendido de página HTML que se firma, incluidos los ficheros adjuntos
- Estructura PKCS#7 generada
- ASN.1 Estructura de Firma Electrónica
- ASN.1 Estructura Acuse de Recibo
- Hora llegada página firmada a servidor
- Otros datos de funcionamiento interno …

Cuando se verifica la integridad de los datos en componente Servidor, también se tienen en cuenta la página HTML y las imágenes de la misma. Se comprueba que no han sido modificadas en el tránsito, comparando la página e imágenes recibidas con las que tiene almacenadas en la Base de Datos. (Comparación de HASH).

6.1 Integración de una aplicación web con el módulo de Firma web

A continuación se describe el proceso de integración de una aplicación web existente en el módulo de firma digital @Firma.

El proceso implica realizar las siguientes acciones:

1. Localizar la página que deseamos convertir en firmable. Teniendo en cuenta las restricciones necesarias sobre una página firmable. En el apartado 6.1.1 se describe detalladamente este paso.

2. Indicar al Servidor @Firma que una nueva página va a ser firmable. En el apartado 6.1.2 se describe detalladamente este paso.

3. Cuando el proceso de firmado de una página ha finalizado en el Servidor @Firma, se devuelve el control a una página JSP ubicada en la Fachada @Firma ("página destino firmado" a partir de ahora) en la cual se puede incluir libremente cualquier código.

   Se dispone de un API para recuperar información sobre el proceso de firma que se acaba de realizar. Se debe indicar el nombre de la "página destino firmado" al servidor @Firma y desarrollar esta página. En el ejemplo Telvent incluido en la herramienta se genera un recibo al usuario. En el apartado 6.1.3 se describe detalladamente este paso.

   Lo más normal es que esta "página destino firmado" utilice el API disponible para recuperar información sobre el proceso de firmado y posteriormente redireccione el control al Servidor de Aplicaciones original.

   Como resultado del proceso de firmado se genera un *identificador de transacción* (TransactionID) que representa unívocamente la transacción realizada. En esta "página destino firmado", mediante el API, se puede extrae este valor y almacenar donde se desee, para utilizarlo en un futuro y obtener información sobre la transacción realizada.

6.1.1 Localizar la página que se desea convertir en firmable

El primer paso consiste en elegir adecuadamente la página web que se desea convertir en firmable. Se deben tener en cuenta las siguientes restricciones:

1. La página debe tener un FORMULARIO tags <form></form>.

2. Se puede convertir en firmable una página web HTML ,JSP, ASP. El Servidor @Firma establece una conexión HTTPS para recuperar la página a convertir en firmable. Si la página es un HTML no hay ningún problema. Por otro lado, si esta página es un JSP o ASP, se resuelve en su servidor correspondiente y el Servidor @Firma recibe código HTML plano. Pero en este caso, es CONDICIÓN INDISPENSABLE QUE LA ESTRUCTURA del formulario de la página HTML que se genera al resolverse el JSP NO CAMBIE DINAMICAMENTE, en concreto, no pueden cambiar:

   - Propiedad Name de formulario

- Propiedad Action de formulario
- Botón de submit de formulario
- No se pueden añadir dinámicamente imágenes a la página.

3. Con respecto al formulario, se INCLUIRÁN EN LA FIRMA los siguientes elementos:
   - Elementos INPUT (tipo HIDDEN, FILE, RADIO, CHECKBOX, TEXT).
   - Elementos SELECT (tipo SELECT-ONE, SELECT-MULTIPLE).
   - HASH calculado al código HTML de la página firmable.
4. Los campos INPUT y SELECT que se incluyen en la firma deben tener indicado el atributo NAME, en caso contrario no se incluyen en la firma.
5. En el FORMULARIO siempre se debe indicar el METODO, que siempre debe ser POST: method="POST".
6. Si se utilizan campos INPUT de tipo readonly, es necesario indicar el valor booleano, es decir, readonly="true". Ej: <input name="ejemplo" readonly="true" value="……..
7. Para utilizar campos de tipo SELECT se debe incluir el atributo select-one para selección simple y select-multiple para selección múltiple. Ej:

   <select type="select-one" name="..

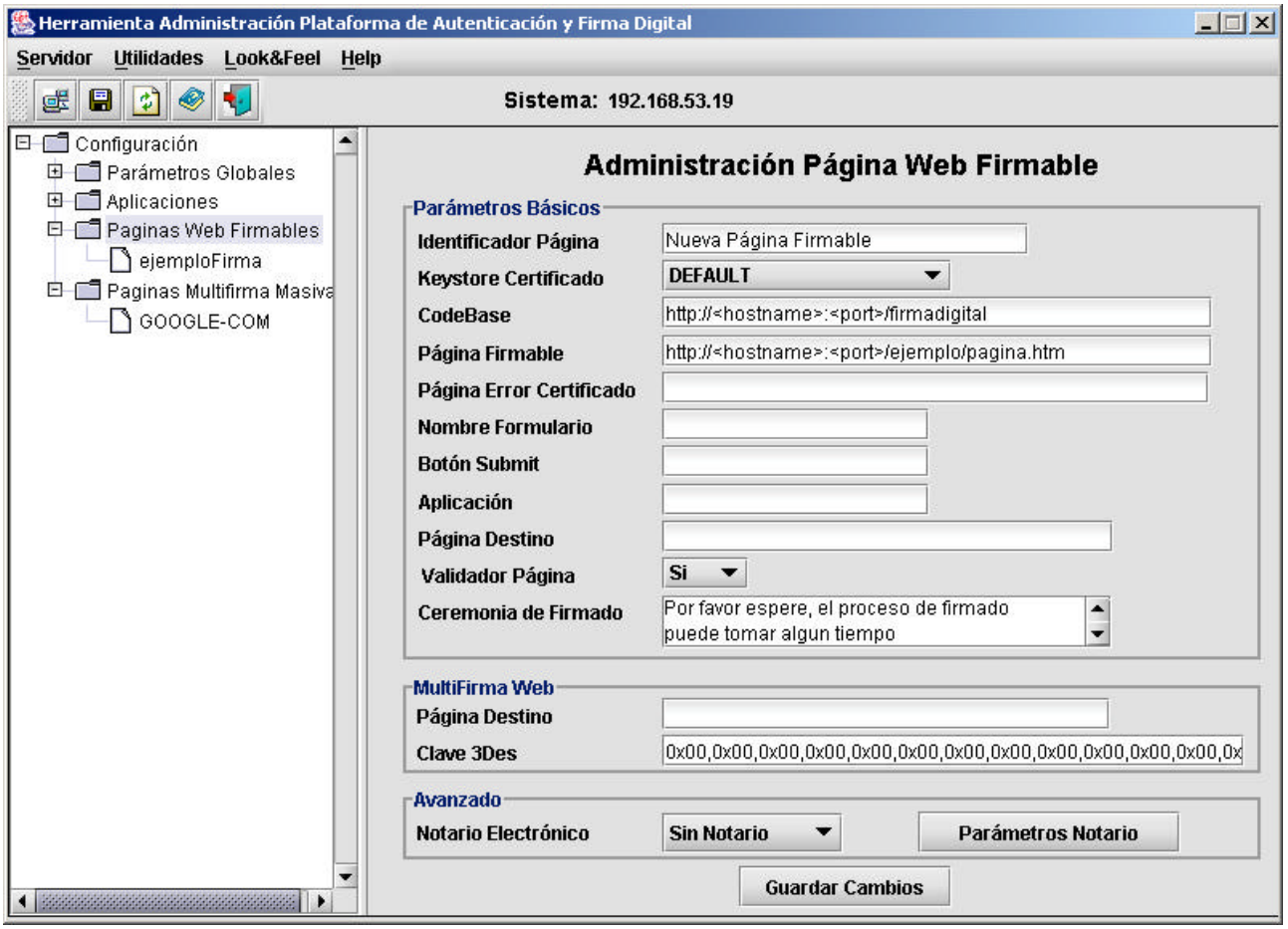
   <select type="select-multiple" name="…
8. Con respecto a las IMÁGENES de las páginas web, se guardan en Base de Datos de componente Servidor y forman parte de la verificación de integridad aquellas incluidas mediante TAGS HTML de tipo IMG. No se guardan aquellas que vengan como atributos, estilos y botones de tipo IMAGE.
9. Si se tiene una página WEB ya incluida en la plataforma y se desea cambiar alguna imagen de la misma, es necesario cambiar el nombre del fichero de la imagen a parte del contenido para que el cambio tenga efecto en la plataforma.
10. Se utilizarán COMILLAS DOBLES en la página firmable y jamás COMILLAS SIMPLES.
11. En el botón de Submit del formulario no se debe utilizar el método OnSubmit, sino solamente el método OnClick(). Es decir, cualquier función que se desee llamar antes de que se inicie el proceso de firmado se colocará en este método. Ej: verificación de campos de formulario, etc.
12. Si se utilizan campos INPUT de tipo FILE, se debe colocar el siguiente atributo en el FORMULARIO enctype="MULTIPART/FORM-DATA". Ej:

<form name="form" action="…" enctype="MULTIPART/FORM-DATA" method="post" >

6.1.2 Indicar al Servidor @Firma que una nueva página va a ser firmable

Para indicar al Servidor @Firma que una nueva página va a ser firmable, se deben seguir los pasos siguientes:

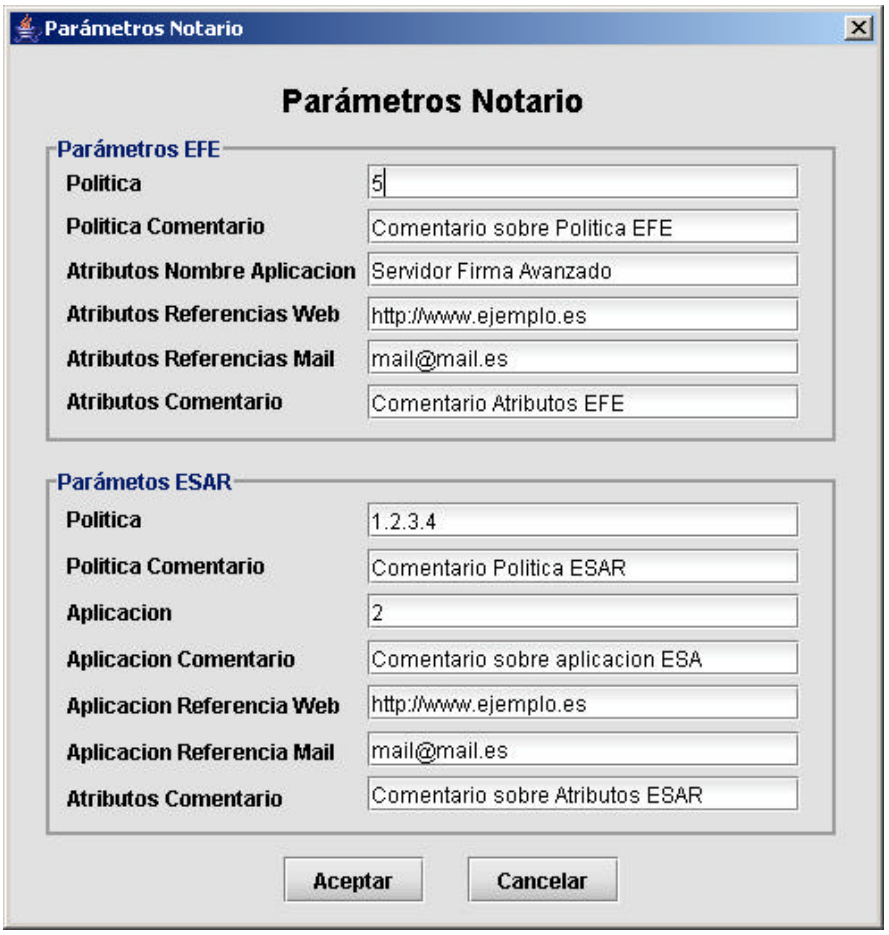
1. Es necesario dar alta la página en la herramienta de Administración de la plataforma de Firma. Una vez inicializada la herramienta seleccionar el nodo "Páginas Web Firmables" del árbol de la izquierda. Esto nos mostrará en la parte izquierda un listado de páginas existentes. A continuación pulsar el botón "Nueva Página Firmable" de la parte izquierda de la pantalla. La siguiente pantalla muestra el resultado:



Los parámetros a introducir son los siguientes:

- Identificador Página: Identificador de la aplicación de firma y/o multifirma web.
- Keystore Certificado: Indica el keystore utilizado para verificar los certificados digitales de usuario utilizados en la firma.
- CodeBase: https://*hostname:port*/firmadigital. Indica la url lógica base del módulo de firma, que será la url de la Fachada. Será necesario establecer el nombre de la Fachada (hostname) y puerto (port). No debe colocarse la "/" del final. NOTA: mantener */firmadigital* como nombre lógico.

- Pagina Firmable: Nombre de página web que deseamos hacer firmable incluida su ruta lógica completa http. Ej.: https://<hostname>:<port>/ruta…./página.html
- Pagina Error Certificado: URL completa de la página de error que toma el control cuando el usuario presenta un certificado no valido. Si se deja vacio se utililza una por defecto.
- Nombre Formulario: Nombre del formulario de la página que deseamos hacer firmable.
- Botón Submit: Nombre del botón de SUBMIT sobre el que opera el proceso de firmado.
- Aplicación: Nombre de la aplicación a la cual pertenece la página firmable actual.
- Pagina Destino: Nombre de "página destino firmado" de la página firmable. Ej.: destino.jsp
- Validador Página: Indica si se usa un validador para comprobar que la página es correcta, cumple las especificaciones para ser firmable, antes de iniciar el proceso de firma.
- Ceremonia de Firmado: Indica el texto que se muestra al cliente cuando se dispone a firmar.
- Notario Electrónico: Indica si queremos utilizar o no Firma Avanzada.
- Parámetros Notario: En caso de utilizar Firma Avanzada deberemos configurar los parámetros del Notario Electrónico. La siguiente figura muestra dichos parámetros.



2. La página que deseamos hacer firmable no puede ser la primera en el flujo de la aplicación web. Debe haber una página anterior que hace referencia a nuestra página firmable. Se debe editar esta página y cambiar la llamada a nuestra página firmable por la llamada al JSP de la Fachada @Firma denominado *entradafirma.jsp*, de la siguiente forma:

https://<hostname>:<port>/firmadigital/servicio/entradafirma.jsp?page=<nombre_sección>&anagrama=<anagrama fiscal largo>

Los elementos <*hostname*> y <*port*> corresponden con la dirección y puerto de la Fachada @Firma.

El elemento <*nombre_sección*> es el parámetro "*Identificador Página*" de la nueva aplicación que hemos creado con la Herramienta de Administración.

El parámetro "*anagrama"* es opcional y si se pone sólo permite que la página sea firmada con un certificado que se corresponda con dicho anagrama. Se usa en aplicaciones que tengan autenticación y sólo se permita firmar a la persona que se autenticó.

Ejemplo:

https://192.168.53.241:444/firmadigital/servicio/entradafirma.jsp?page=ejemploFirma&anagrama=44226299jinfagomm

En caso de que la página firmable deba recibir parámetros por GET, se deberán pasar estos parámetros al JSP *entradafirma.jsp*, adicionalmente al parámetro page y al parámetro anagrama (sí corresponde). Ejemplo:

…..entradafirma.jsp?page=ejemploTelvent.&nombre=javier&edad=15 ……

En caso de que la página que hace referencia a nuestra página firmable tenga un formulario cuya acción apunte a nuestra página firmable y el método usado en el submit sea POST, entonces se debe cambiar el valor del campo *action* del formulario a *entradafirma.jsp* y añadir un campo de tipo *hidden* para el parámetro *page* y otro para el parámetro *anagrama* (si corresponde). Ejemplo:

```
<form name="abc" action="https://88.60.2.253:8080/firmadigital/servicio/entradafirma.jsp" method="POST">
    <input type="hidden" name="page" value="ejemploTelvent">
```

Por ultimo, es muy IMPORTANTE que el atributo ACTION del FORMULARIO de nuestra nueva página firmable siempre tome el valor siguiente:

https://<hostname>:<port>/firmadigital/servicio/formsBackEnd/recepcion.jsp

donde <hostname> y <port> corresponden a la Fachada @Firma

Ejemplo:

```
<form action=" https://192.168.53.241:444/firmadigital /servicio/formsBackEnd/recepcion.jsp ">
```

6.1.3 Generación "página destino firmado"

Finalmente se debe crear una página JSP NUEVA que será la encargada de tomar el control una vez que el servidor @Firma ha finalizado la transacción de firma.

Se debe tener en cuenta lo siguiente:

1. El nombre de esta página se indica en el atributo "*Página Destino"* de la nueva Página Web Firmable que hemos creado con la herramienta de Administración.

2. Esta página JSP NUEVA se debe copiar en el directorio de la Fachada @Firma siguiente:

   /DIR_INSTALACION_JBOSS/server/all/deploy/firmadigital.war/servicio/formsBackEnd

3. En esta página JSP se dispone de un API para recuperar de la Base de Datos del Servidor @Firma toda la información necesaria sobre el proceso de firmado. En este momento se deja el control de la aplicación en manos de la persona que está integrando la página firmable en la herramienta. En este punto se puede realizar cualquier cosa que se desee. Por ejemplo:

   - Generar un recibo al usuario con información recuperada de la Base de Datos del Servidor de @Firma. Es la opción que se ha tomado en el ejemplo TELVENT incluido en la herramienta.

   - Guardar información en otra Base de Datos externa sobre la transacción de firmado realizada con información recuperada de la Base de Datos del Servidor @Firma.

   - Hacer un Redirect a una página del Servidor de Aplicaciones original pasándole parámetros recuperados de la Base de Datos de @Firma por método GET.

   - Etc.…

   Es importarte recuperar mediante el API el valor *TransactionID* que identifica la transacción de firma realizada y almacenarlo donde se desee para poder utilizarlo en un futuro para recuperar información sobre la firma realizada.

   Para poder utilizar el API en página JSP "página destino firmado", simplemente se debe incluir al comienzo de la misma, la declaración de uso de JAVABEAN siguiente:

   ```
   <jsp:useBean id="data" scope="request" class="com.telventi.firma.data.AppData" />
   ```

   A partir de este momento el objeto data nos da acceso a toda la información de la transacción de firma recién terminada.

   Para obtener información detallada sobre los métodos de la clase AppData consultar el "javadoc" proporcionado en el directorio "Documentación / JavaDoc / Modulo Firma / Firma Web" del CD Desarrollo.

4. La plataforma @Firma también dispone de una Interfaz RMI-IIOP FirmaApiFacade para acceder remotamente en cualquier momento futuro y desde cualquier máquina a la información de una transacción de firma. Este API remoto requiere el valor *TransactionID.*

6.2 Acceso mediante RMI-IIOP

Para utilizar la interfaz RMI-IIOP FirmaApiFacade se requieren clientes con máquina virtual de JAVA JDK 1.4 o superior.

En el "CD Desarrollo" se proporcionan las librerías necesarias para acceder a la interfaz. Se encuentran ubicadas en el directorio "\ Modulo Firma \ Firma Web \ apiRMI-IIOP \ librerias". El directorio contiene los siguiente ficheros:

- TelventRemoteApi.jar: Clases e Interfaces de acceso a PKI
- auth.conf : Fichero de para configuración de acceso a interfaz mediante JAAS
- jbossall-client.-jar : Librerías cliente acceso Jboss
- log4j.jar : Librería para la utilización del log.

Las clases necesarias en este caso son las siguientes:

- com.telventi.firma.fachada.ConexionFirmaApiFacade
- com.telventi.ejb.FirmaApiFacade
- com.telventi.firma.data.DatoFirmado
- com.telventi.firma.data.DatoPeticion

A continuación se muestra un pequeño extracto de código JAVA que obtiene una referencia a la interfaz para poder utilizar sus métodos.

```
// IMPORTAMOS LAS CLASES NECESARIAS EN LA CABECERA

import com.telventi.ejb.FirmaApiFacde;

import com.telventi.firma.fachada.ConexionFirmaApiFacade;

import com.telventi.firma.data.*;          ……

                ……

try {

// Creamos un OBJETO ConexionFirmaApiFacade (SERVIDOR, USUARIO, PASSWORD)

String host = "192.168.53.18";
```

```
String usuario = “user01”;

String password = “12345”;

ConexionFirmaApiFacade conexion = new ConexionFirmaApiFacade(host,usuario,password);


// OBTENEMOS UNA REFERENCIA A LA INTERFAZ A PARTIR DE LA CONEXION

FirmaApiFacade api = conexion.getFirmaApiFacade();

// AHORA PODEMOS UTILIZAR CUALQUIERA DE SUS METODOS

String subjectDN  = api.getCertificateSubjectDN(idTransaction);

}catch(java.lang.Exception ex)  { }
```

El fichero auth.conf debe copiarse en el directorio desde el cual se ejecuta la aplicación.

6.3 Códigos de Error en página de Certificado Usuario No Válido

- Error 5: El certificado de Usuario no es válido para la firma, el tipo de certificado no se reconoce.
- Error 6: El anagrama presentado no se corresponde con el anagrama del certificado.
- Error 11: Algoritmo de firma en certificado de cliente invalido.
- Error 12: Clave en certificado cliente no valida.
- Error 13: Proveedor de certificado cliente no encontrado.
- Error 14: Error al comprobar la firma del certificado de cliente.
- Error 15: Error al obtener los campos obligatorios del certificado.
- Error 16: No se ha encontrado el certificado de la CA de este certificado.
- Error 17: No se ha construido el objeto de forma correcta.
- Error 18: Certificado expirado.
- Error 19: No se ha encontrado la CRL.
- Error 20: Error al descargar la CRL.
- Error 21: Error al verificar la crl con el certificado de su CA
- Error 22: No se ha encontrado el certificado de la CA de esta CRL
- Error 24 hasta Error 32: Certificado revocado.

7 Proceso MultiFirma Web

El proceso de MultiFirma Web es implementado por la interfaz com.telventi.multifirma. MultiFirmaApiFacade y unos componentes web que permiten recuperar la pagina firmada y terminar el proceso de multifirma.

En el proceso de multifirma web intervienen tres agentes: usuario (cliente), aplicación que utiliza la interfaz y la plataforma de Firma (interfaz y componentes web).

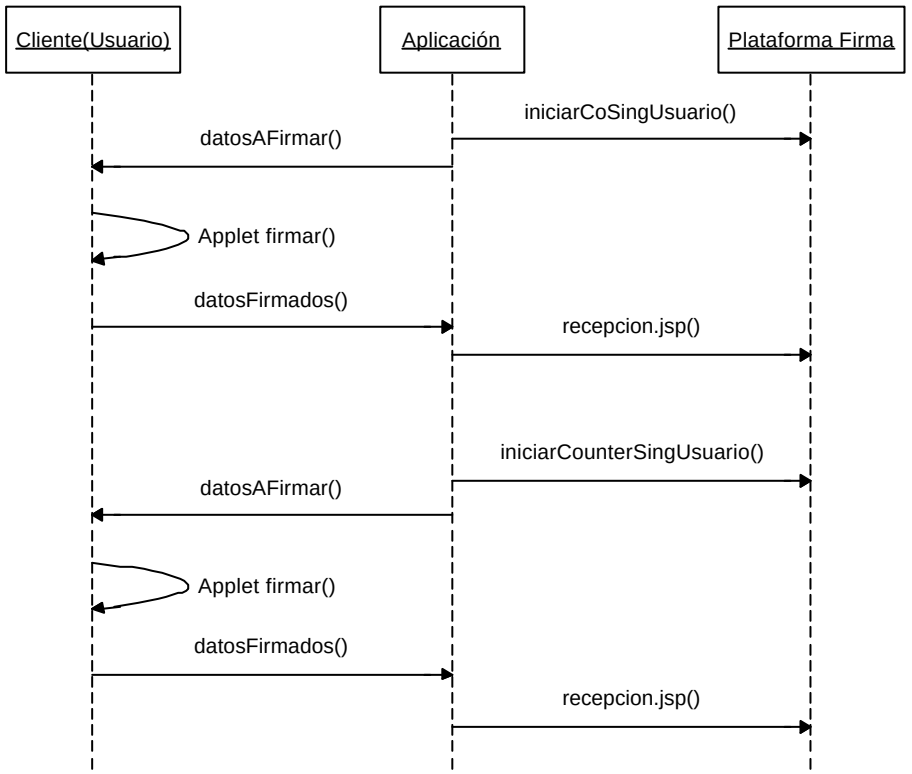
El proceso de multifirma se puede describir como un procedimiento en 3 pasos:

1) Se realiza una firma web normal y se obtiene el identificador de transacción.

2) La aplicación solicita a la plataforma de Firma la generación de los datos a firmar, que serán enviados a la máquina del usuario para ser firmados. (métodos *iniciarCoSignUsuario o iniciarCounterSignUsuario*). También puede solicitar la página firmada, incluidos los adjuntos (componente web visualización *principalpaginafirmada.jsp*)

3) El usuario firma los datos y la aplicación envía el resultado a la plataforma de Firma para terminar el proceso. (componente web de recepción *recepcion.jsp*)

La siguiente figura muestra el proceso de multifirma:



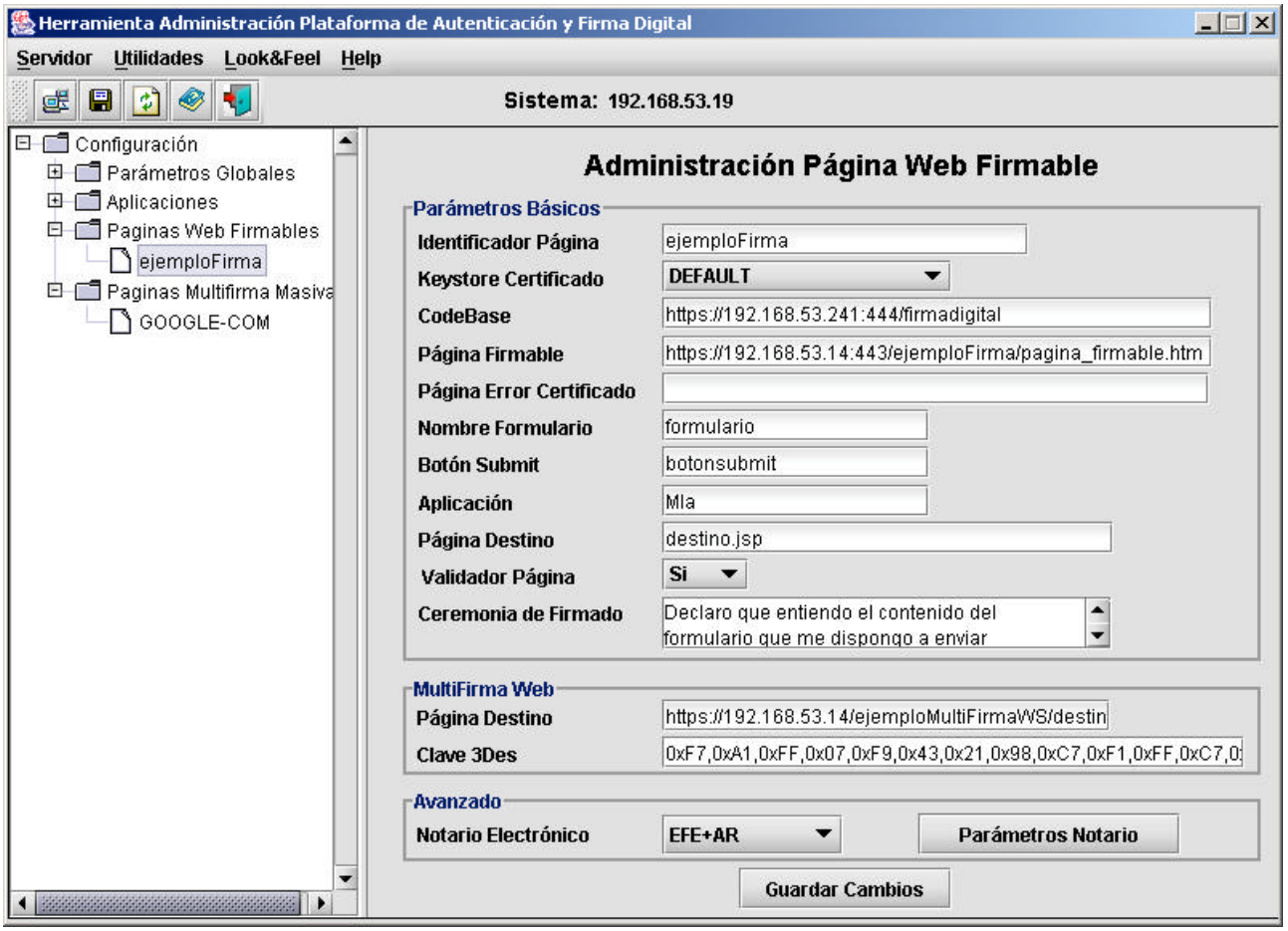
Los métodos de la interfaz permiten realizar algunas consultas frecuentes, la mas normal es la de obtener todos los que han firmado anteriormente una transacción, método *getSigners(…).*

Para obtener información detallada sobre los métodos de la interfaz consultar el "javadoc" proporcionado en el directorio "Documentación / JavaDoc / Modulo Firma / Firma Web" del CD Desarrollo.

Para poder desarrollar una aplicación que utilice la interfaz MultiFirmaApiFAcade y los componentes web citados anteriormente es necesario dar alta una aplicación en la herramienta de Administración de la plataforma de Firma.

A continuación se describen los pasos necesarios a llevar a cabo en dicha herramienta. Para información mas detallada sobre la herramienta de Administración véase el manual del Administrador del Sistema.

Una vez inicializada la herramienta seleccionar por debajo del nodo "Paginas Web Firmables" del árbol de la izquierda la página web que queremos que permita la multifirma. La siguiente pantalla muestra el resultado:



Los parámetros a introducir son los que están dentro del panel MultiFirma Web:

- Página Destino: URL completa de la página que se rediccionará una vez se haya finalizado la multifirma.
- Clave 3Des: Clave utilizada para encriptar el idTransacción que hay que pasar al componente de recuperación de la página firmada *principalpaginafirmada.jsp.*

7.1 Acceso mediante RMI-IIOP

Para utilizar la interfaz RMI-IIOP MultiFirmaApiFacade se requieren clientes con máquina virtual de JAVA JDK 1.4 o superior.

En el "CD Desarrollo" se proporcionan las librerías necesarias para acceder a la interfaz. Se encuentran ubicadas en el directorio "\ Modulo Firma \ MultiFirma Web \ apiRMI-IIOPCliente". El directorio contiene los siguiente ficheros:

- multifirmacliente.jar : Clases e Interfaces de acceso a PKI
- auth.conf : Fichero de para configuración de acceso a interfaz mediante Jaas
- jbossall-client.-jar : Librerías cliente acceso Jboss

Las clases necesarias en este caso son las siguientes:

- com.telventi.conexion.ConexionMultiFirma
- com.telventi.multifirma.MultiFirmaApiFacade
- com.telventi.firmaweb.DTOFirmante
- com.telventi.firmaweb.DTOIniciarMultifirma
- com.telventi.firmaweb.FirmaException

A continuación se muestra un pequeño extracto de código JAVA que obtiene una referencia a la interfaz para poder utilizar sus métodos.

```
// IMPORTAMOS LAS CLASES NECESARIAS EN LA CABECERA

import com.telventi.multifirma.MultiFirmaApiFacde;

import com.telventi.conexion.ConexionMultiFirma;
```

```
import com.telventi.firmaweb.*;          ……

                ……

try {

// Creamos un OBJETO ConexionMultiFirma (SERVIDOR, USUARIO, PASSWORD)

String host = "192.168.53.18";

String usuario = "user01";

String password = "12345";

ConexionMultiFirma conexion = new ConexionMultiFirma(host,usuario,password);


// OBTENEMOS UNA REFERENCIA A LA INTERFAZ A PARTIR DE LA CONEXION

MultiFirmaApiFacade api = conexion.getMultiFirma();

// AHORA PODEMOS UTILIZAR CUALQUIERA DE SUS METODOS

Vector firmantes  = api.getSigners(idTransaction);

}catch(java.lang.Exception ex)  { }
```

El fichero auth.conf debe copiarse en el directorio desde el cual se ejecuta la aplicación.

7.2 Acceso mediante WEBSERVICES

La plataforma de Firma proporciona los Ficheros de Descripción de los Servicios Web (WSDL) publicados en la siguiente URL:

https://<servidor_firma>:<puerto>/axis/servlet/AxisServlet

(Será necesario introducir el usuario/password para Webservices. Consultar con el administrador del sistema.)

Desarrollando una aplicación, para poder comunicarse con el Servicio Web que se desee es necesario generar las clases de acceso al mismo a partir de su fichero descriptor WSDL.

Existen una serie de herramientas que facilitan este trabajo, entre ellas se citan las siguientes:

- Paquete AXIS JAVA: La clase "org.apache.axis.wsdl.WSDL2Java", dado un fichero descriptor WSDL permite generar las clases cliente en tecnología JAVA.
- Paquete AXIS C/C++: La clase "org.apache.axis.wsdl.wsdl2ws.WSDL2Ws", dado un fichero descriptor WSDL permite generar las clases cliente en tecnología C/C++.

- GSoap. Permite generar clases cliente C/C++ a partir de un fichero descriptor WSDL.

- Etc.

A los clientes generados por las herramientas anteriores posiblemente será necesario añadir el código necesario para realizar la comunicación SSL y la autenticación JAAS con la plataforma de firma. En los ejemplos proporcionados con la plataforma (JAVA) se muestran claramente los mecanismos adicionales incorporados.

Como ayuda adicional puede consultarse el "Javadoc" proporcionado en el directorio "Documentación / JavaDoc / Modulo Firma / Firma Web" del CD Desarrollo y los ejemplos desarrollados en JAVA (directorio "Modulo Firma / MultiFirma Web / Ejemplos WebServices" del CD Desarrollo)

Concretamente, para obtener el fichero descriptor WSDL correspondiente a la interfaz MultiFirmaApiFacade conectarse a la siguiente URL:

https://<servidor_firma>:<puerto>/axis/services/MultiFirmaWeb?wsdl

En el menú del navegador *Archivo*, seleccionar *Guardar como...* indicando el nombre multifirma.wsdl.

7.3 Componente JSP que visualiza la página con los datos firmados y los adjuntos.

Si en la aplicación de multifirma se quiere visualizar la pagina web firmada y los adjuntos que contenía existe dentro de la Fachada @firma un componente JSP que devuelve una página HTML que contiene la página original con los campos rellenos con los datos que se han firmado y debajo la lista de ficheros adjuntos a dicha página si los hubiera.

La ruta de acceso es: https://<fachada@firma>:444/multifirma/servicio/principalpaginafirmada.jsp

Este componente necesita dos parámetros que son:

- TransactionID: Identificador de la transacción de firma, encriptado por motivos de seguridad y codificado en Base64 para poderlo poner en la URL.

- Seccion: El "Identificador Página" de la pagina web firmable en la Herramienta de Administración.

La llamada quedaría algo así:

https://172.19.133.128:444/multifirma/servicio/principalpaginafirmada.jsp?TransactionID=<idEncriptado>&Seccion=<id pagina>

El parámetro TransactionID debe de estar codificado, por motivos de seguridad, con la clave DES que tiene asignada cada aplicación de multifirma en la Herramienta de Administración. Esto se puede ver con claridad en el componente *principal.jsp* incluido en los ejemplo de multifirma incluidos en el CD Desarrollo (Modulo Firma / MultiFirma Web / Ejemplos WebServices / ejemplomultifirmaws.war).

7.4 Applets de firma y componente JSP de recepción.

Los applets hay que copiarlos en el directorio de la aplicación y el componente "recepcion.jsp" se encuentra en la Fachada @Firma y los creadores de aplicaciones de multifirma solo tendrán que hacer uso de él. El componente de recepción es:

https://<fachada @Firma>:444/multifirma/servicio/ recepcion.jsp.

Los applets se encuentran en el "CD Desarrollo" en la carpeta "Modulo Firma/Multifirma Web/ Componentes Web" y se llaman "TelventMultiCliente.cab", "TelventMultiCliente.jar", "TelventMultiClienteLinux.jar" y "scriptMultifirma.js". Se deben de copiar en el directorio de la aplicación de multifirma.

Una vez que se haya llevado a cabo todo el proceso de firmado se redireccionará la aplicación hacia la página que se encuentra indicada en la H. De Administración pasándole como parámetro (TransactionID) el identificador de la transacción de firma realizada.

Para hacer uso de los componentes anteriores se debe de realizar una pagina JSP con las siguientes características:

1. Obtener la información a firmar, con uno de los métodos *iniciarC…* del api.
2. La llamada al componente de retorno se debe de hacer desde el campo "action" de un formulario llamado "formulario". Solo debe de haber uno en la página.
3. El formulario debe de tener los siguientes campos ocultos:
   - SignerCertificate: Lo usa el applet para guardar el certificado usado en la firma.
   - Signature: Lo usa el applet para guardar la firma generada.
   - LastTransactionID: Identificador de la transacción de multifirma, se saca del DTOIniciciarMultifirma (*dto.getIdTransaccion()*) devuelto por el método *iniciarC….*
   - PKCS7: informacíon que se va ha firmar, se saca del DTOIniciciarMultifirma (*dto.getHashEnc()*) devuelto por el método *iniciarC….*
   - SignCeremonyText: Texto que se mostrará para confirmar que se quiere firmar.
   - SignatureType: Con el tipo de multifirma que se va ha realizar: "CoSign" o "CounterSign".
4. El formulario debe de tener un campo de tipo submit con lo siguiente onclick="return Telvent_OnFormSubmit()"
5. Se debe incluir en la pagina el siguiente script:

   « <script language="javascript" src= "scriptMultifirma.js"> </script> »

Esto se puede ver con claridad en el componente *firmar.jsp* incluido en el ejemplo de multifirma incluidos en el CD Desarrollo (Modulo Firma / MultiFirma Web / Ejemplos WebServices / ejemplomultifirmaws.war).

8 Proceso de Multifirma Masiva

El proceso de MultiFirma Masiva es implementado por la interfaz com.telventi.multifirmamasiva. MultiFirmaWebMasiva. Esta interfaz contiene métodos para realizar un proceso de multifirma masiva y algunos métodos frecuentes para realizar consultas.

En el proceso de multifirma masiva intervienen tres agentes: usuario (cliente), aplicación que utiliza la interfaz y la plataforma de Firma (interfaz multifirma).

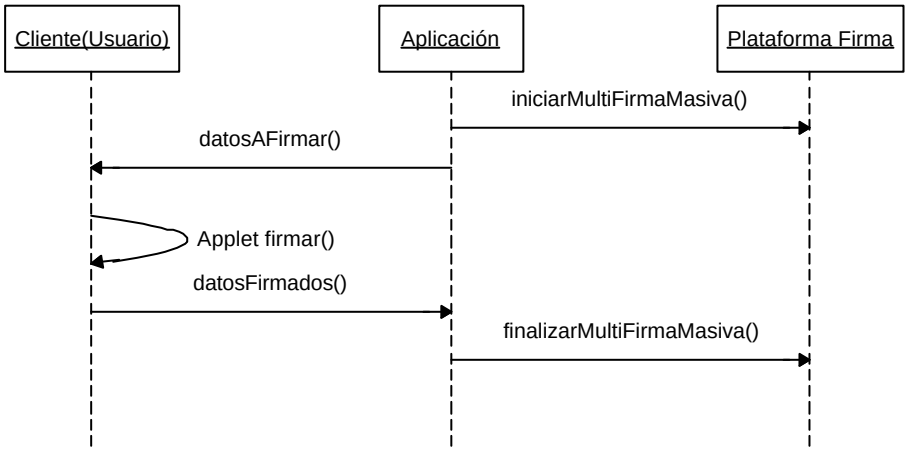
El proceso de multifirma se puede describir como un procedimiento en 3 pasos:

1) Se registra la página que se va ha multifirmar en la Herramienta de Administración y se obtiene el identificador de transacción.

2) La aplicación solicita a la plataforma de Firma la generación de la pagina que se va ha multifirmar la cual incluye los datos a firmar, que serán enviados a la máquina del usuario para ser firmados. (método *iniciarMultiFirmaMasiva*).

3) El usuario firma los datos y la aplicación envía el resultado a la plataforma de Firma para terminar el proceso. (método *finalizarMultiFirmaMasiva*)

La siguiente figura muestra el proceso de multifirma masiva:



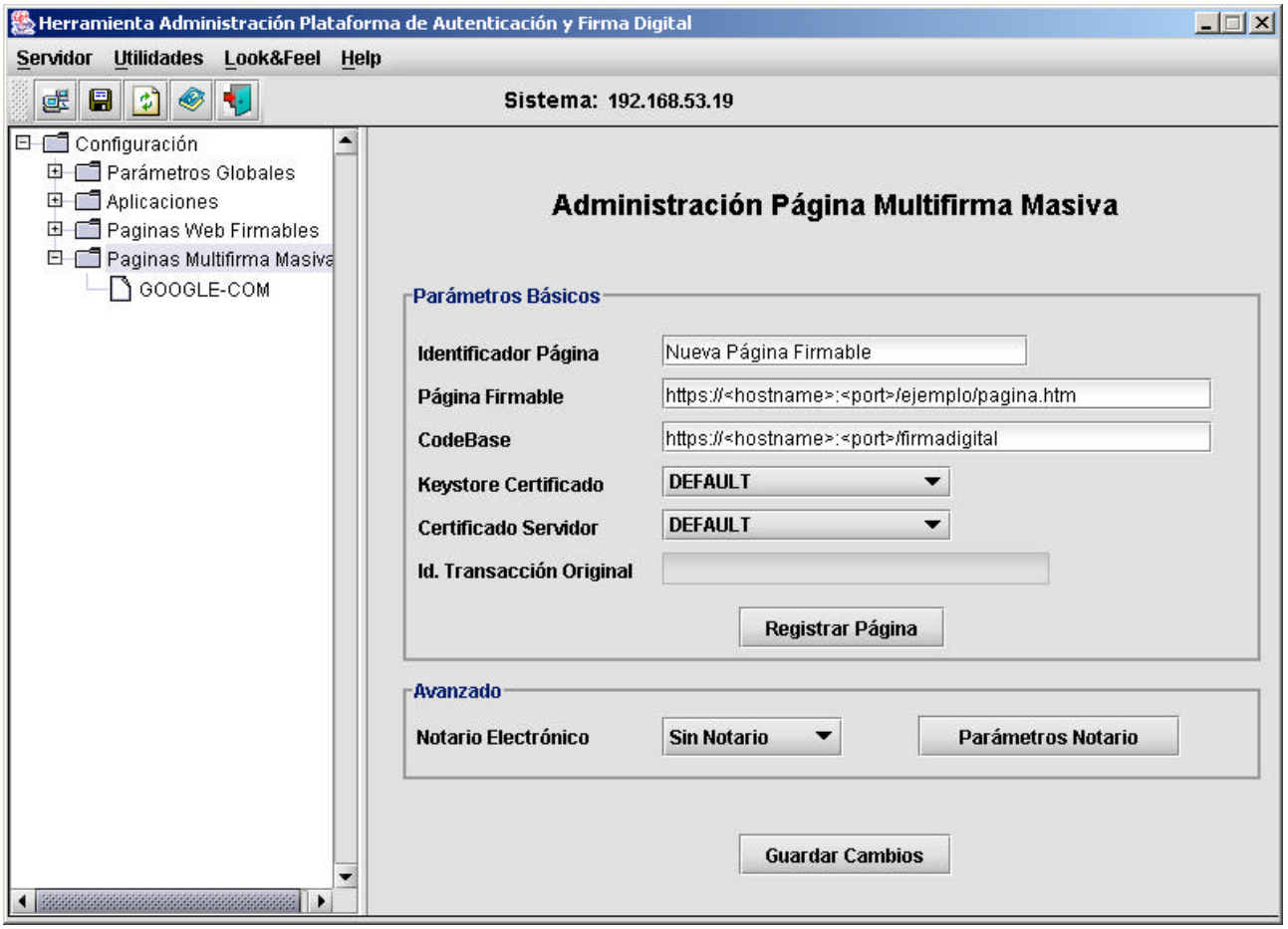
Los métodos de la interfaz permiten realizar algunas consultas frecuentes, la mas normal es la de obtener todos los que han firmado la página, método *getSigners(…).*

Para obtener información detallada sobre los métodos de la interfaz consultar el "javadoc" proporcionado en el directorio "Documentación / JavaDoc / Modulo Firma / Firma Web" del CD Desarrollo.

Para poder desarrollar una aplicación que utilice la interfaz MultiFirmaWebMasiva es necesario dar alta una aplicación en la herramienta de Administración de la plataforma de Firma.

A continuación se describen los pasos necesarios a llevar a cabo en dicha herramienta. Para información mas detallada sobre la herramienta de Administración véase el manual del Administrador del Sistema.

Una vez inicializada la herramienta seleccionar el nodo "Páginas Multifirma Masiva" del árbol de la izquierda. Esto nos mostrará en la parte izquierda un listado de aplicaciones existentes. A continuación pulsar el botón "Nueva Página" de la parte izquierda de la pantalla. La siguiente pantalla muestra el resultado:



Los parámetros a introducir son los que están dentro del panel MultiFirma Web:

- Identificador Página: Identificador de la aplicación de multifirma masiva
- Pagina Firmable: Nombre de página web que deseamos hacer firmable incluida su ruta lógica completa http. Ej.http://<hostname>:<port>/ruta…./página.html
- CodeBase: https://*hostname:port*/firmadigital. Indica la url lógica base del módulo de firma, que será la url de la Fachada. Será necesario establecer el nombre de la Fachada (hostname) y puerto (port). No debe colocarse la "/" del final. NOTA: mantener */firmadigital* como nombre lógico.

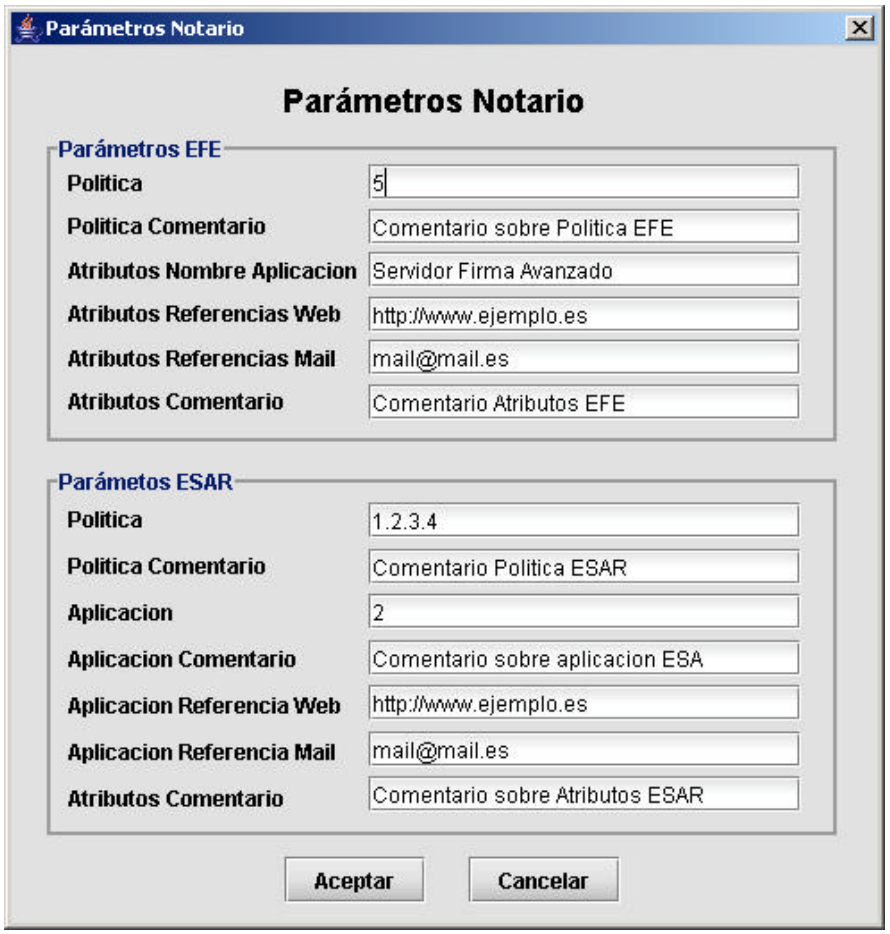
- Keystore Certificado: Indica el keystore utilizado para verificar los certificados digitales de usuario utilizados en la firma. (Ver Keystores CAs en Parámetros Globales).

- Certificado Servidor: Certificado de Servidor utilizado para realizar la primera firma en una multifirma masiva.

- Id. Transacción Original: Identificador de la primera transacción de una multifirma masiva, lo genera la aplicación automáticamente cuando se pulsa le botón "Registrar Página" y no se puede modificar. Este identificador hay que pasárselo a los creadores de aplicaciones de Multifirma Masiva para que lo usen en las llamadas a los métodos de la interfaz proporcionada.

? Notario Electrónico: Indica si queremos utilizar o no Firma Avanzada.

? Parámetros Notario: En caso de utilizar Firma Avanzada deberemos configurar los parámetros del Notario Electrónico. La siguiente figura muestra dichos parámetros.



NOTA: No se puede registrar dos veces la misma Pagina Firmable.

Una vez rellenados los parámetros se pulsa el botón "Registrar Página" con lo que se obtine la página indicada, se registra en el servidor y se realiza una firma de servidor con el certificado indicado. Si todo es correcto en el campo de sólo lectura "Id. Transacción Original" aparece el identificador de dicha firma.

8.1 Acceso mediante RMI-IIOP

Para utilizar la interfaz RMI-IIOP MultiFirmaMasiva se requieren clientes con máquina virtual de JAVA JDK 1.4 o superior.

En el "CD Desarrollo" se proporcionan las librerías necesarias para acceder a la interfaz. Se encuentran ubicadas en el directorio "\ Modulo Firma \ MultiFirma Masiva \ apiRMI-IIOPCliente". El directorio contiene los siguiente ficheros:

- multifirmamasivacliente.jar : Clases e Interfaces de acceso a PKI
- auth.conf : Fichero de para configuración de acceso a interfaz mediante Jaas
- jbossall-client.-jar : Librerías cliente acceso Jboss

Las clases necesarias en este caso son las siguientes:

- com.telventi.conexion.ConexionMultiFirmaMasiva
- com.telventi.multifirmamasiva.MultiFirmaWebMasiva
- com.telventi.firmaweb.DTOFirmante
- com.telventi.firmaweb.FirmaException

A continuación se muestra un pequeño extracto de código JAVA que obtiene una referencia a la interfaz para poder utilizar sus métodos.

```
// IMPORTAMOS LAS CLASES NECESARIAS EN LA CABECERA
import com.telventi.multifirmamasiva.MultiFirmaWebMasiva;
import com.telventi.conexion.ConexionMultiFirmaMasiva;
import com.telventi.firmaweb.*;          ……
                ……
try {
// Creamos un OBJETO ConexionMultiFirma (SERVIDOR, USUARIO, PASSWORD)
String host = "192.168.53.18";
String usuario = "user01";
String password = "12345";
```

```
ConexionMultiFirmaMasiva conexion = new ConexionMultiFirmaMasiva(host,usuario,password);


// OBTENEMOS UNA REFERENCIA A LA INTERFAZ A PARTIR DE LA CONEXION

MultiFirmaWebMasiva api = conexion.getMultiFirmaMasiva();

// AHORA PODEMOS UTILIZAR CUALQUIERA DE SUS METODOS

byte[] pagina  = api.iniciarMultiFirmaMasiva (idTransaction);

}catch(java.lang.Exception ex) { }
```

El fichero auth.conf debe copiarse en el directorio desde el cual se ejecuta la aplicación.

8.2 Acceso mediante WEBSERVICES

La plataforma de Firma proporciona los Ficheros de Descripción de los Servicios Web (WSDL) publicados en la siguiente URL:

https://<servidor_firma>:<puerto>/axis/servlet/AxisServlet

(Será necesario introducir el usuario/password para Webservices. Consultar con el administrador del sistema.)

Desarrollando una aplicación, para poder comunicarse con el Servicio Web que se desee es necesario generar las clases de acceso al mismo a partir de su fichero descriptor WSDL.

Existen una serie de herramientas que facilitan este trabajo, entre ellas se citan las siguientes:

- Paquete AXIS JAVA: La clase "org.apache.axis.wsdl.WSDL2Java", dado un fichero descriptor WSDL permite generar las clases cliente en tecnología JAVA.
- Paquete AXIS C/C++: La clase "org.apache.axis.wsdl.wsdl2ws.WSDL2Ws", dado un fichero descriptor WSDL permite generar las clases cliente en tecnología C/C++.
- GSoap. Permite generar clases cliente C/C++ a partir de un fichero descriptor WSDL.
- Etc....

A los clientes generados por las herramientas anteriores posiblemente será necesario añadir el código necesario para realizar la comunicación SSL y la autenticación JAAS con la plataforma de firma. En los ejemplos proporcionados con la plataforma (JAVA) se muestran claramente los mecanismos adicionales incorporados.

Como ayuda adicional puede consultarse el "Javadoc" proporcionado en el directorio "Documentación / JavaDoc / Modulo Firma / Firma Web" del CD Desarrollo y los ejemplos

desarrollados en JAVA (directorio "Modulo Firma / MultiFirma Masiva / Ejemplos WebServices" del CD Desarrollo)

Concretamente, para obtener el fichero descriptor WSDL correspondiente a la interfaz MultiFirmaApiFacade conectarse a la siguiente URL:

https://<servidor_firma>:<puerto>/axis/services/MultiFirmaWebMasiva?wsdl

En el menú del navegador *Archivo*, seleccionar *Guardar como...* indicando el nombre multifirma.wsdl.

8.3 Utilización del Componente Cliente de Firma (Applet Cliente)

En un proceso de firma de ficheros por usuario también es necesario incorporar un componente que firme los datos en la máquina del usuario. Este componente es un Applet de Firma que debe ser cargado en la página de la aplicación a desarrollar. Para ello se suministran los siguientes ficheros en el directorio "\ Modulo Firma \ Multifirma Masiva \ Componentes Web" del "CD Desarrollo":

- Sign.cab
- SignMozilla.jar
- scriptfirma.js

Estos ficheros se copiarán en el directorio de la aplicación que se esté desarrollando. El procedimiento a seguir consiste en incorporar el fichero javascript "scriptfirma.js" en la página que utilizará el usuario para firmar.

```
<script language="javascript" src ="scriptfirma.js"></script>
```

Una vez copiado el fichero "scripfirma.js", se debe editar y cambiar una variable que representa la URL de la fachada de la plataforma de firma. Esta variable se llama "pginstalacion". Se debe cambiar la IP que aparece por la IP o nombre de la Fachada @Firma.

Este fichero javascript se encarga de cargar un applet de firmado en el document denominado "SignApplet", de esta forma pone a disposición un método denominado "Firma()" que permite firmar datos.

A continuación debemos desarrollar un código javascript que realice una llamada al método "Firma()" de dicho Applet cuando se quiera firmar. Este método devuelve una cadena con los datos firmados, cadena vacía cuando el usuario cancela la firma o "null" cuando ocurrió un error inesperado. Aquí se muestra un ejemplo:

```
<script language=JavaScript>

function clickboton ()

{
```

```
        // LLAMADA A METODO FIRMA DEL APPLET
        var a = document.SignApplet.Firma("datosafirmar");
        if(a == null)          // ERROR FIRMANDO DATOS
                alert("No se ha podido firmar");
        else if(a == "")       // FIRMA CANCELADA POR USUARIO
                alert("Firma cancelada");
        else                   // FIRMA CORRECTA. PASAMOS A SIGUIENTE ACCION …….
        {                      // EJEMPLO: SUBMIT DEL FORMULARIO
                document.formulario.firmagenerada.value = a;
                document.formulario.submit();
        }
}
</script>
```

El valor que debe recibir el método Firma del objeto SignApplet se obtiene del campo HashEnc del formulario FormularioFirma que se ha incluido en la página web devuelta en la llamada al método *iniciarMultiFirmaMasiva(…).* En dicho formulario también existe el campo IdTransaccion que indica la transacción de multifirma masiva que se ha iniciado.

Si la página devuelta por el método *iniciar…* la hemos puesto en un Frame llamado "pagina" una forma de obterner los valores anteriores es:

```
window.top.frames["pagina"].document.FormularioFirma.HashEnc.value
window.top.frames["pagina"].document.FormularioFirma.IdTransaccion.value
```

Los datos generados por el applet de firma y el identificador de la transacción serán los parámetros de entrada del método *finalizarMultiFirmaMasiva(…)* de la interfaz MultiFirmaWebMasiva.

Para ver un ejemplo de una aplicación de firma de ficheros por usuario véase el ejemplo del directorio "Modulo Firma / Multifirma Masiva / Ejemplos * / ejemploMultifirmaMasiva.war" del CD Desarrollo. Aquí se demuestra la utilización del Applet Cliente de Firma.

9 Aplicaciones de Ejemplo Firma Web de la plataforma

9.1 Aplicación "ejemploFirma"

La aplicación "ejemploFirma" muestra un ejemplo de Firma web.

La aplicación está desarrollada en HTML – JSP.

El código de la aplicación se encuentra disponible en el siguiente directorio del CD Desarrollo, respectivamente:

- "Modulo Firma / Firma Web / ejemplofirma.war"

9.1.1 Poner en marcha la aplicación

Es una aplicación WEB, así pues necesitamos un servidor de aplicaciones, por ejemplo JBOSS.

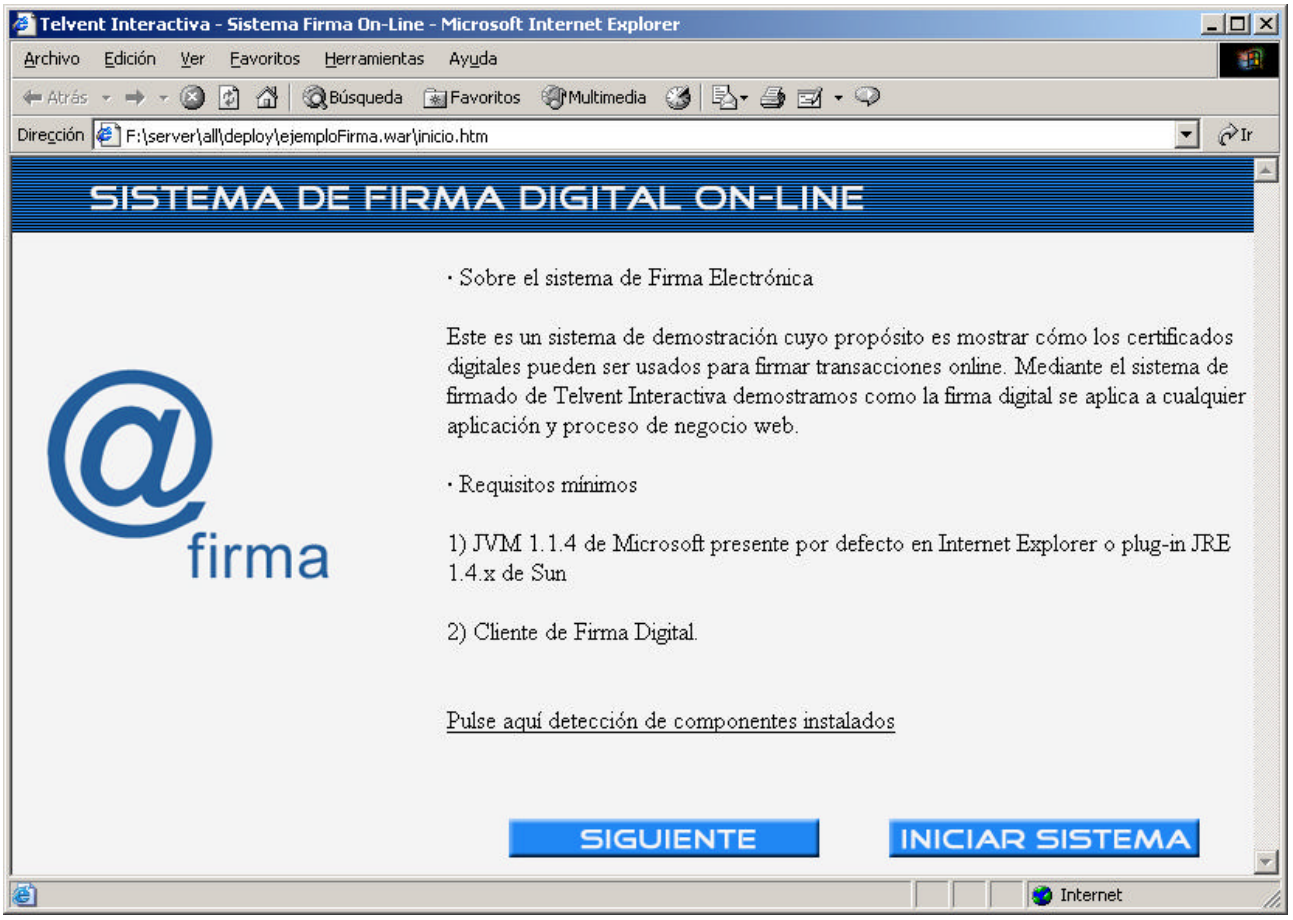
- ejemplofirma.war: aplicación que debe ser desplegada en el directorio de deploy del servidor de aplicaciones. (ej: JBOSS_HOME/server/all/deploy).
- Copiar el fichero "destino.jsp" en la Fachada @Firma en el direcctorio ..\firmadigital.war\servicio\formsBackEnd. (Posiblemente ya esté).
- Dar de alta la aplicación en la Herramienta de Administración con el nombre "ejemploFirma". (Si ya existe modificar la configuración para que sea correcta).

Una vez copiado el directorio modificar los siguientes ficheros:

- comienzo.htm: cambiar la dirección ip que aparece por el nombre o dirección ip de la Fachada @Firma. Cambiar o eliminar tambien el parámetro anagrama.
- detector.js: cambiar la dirección ip de la variable url_base por el nombre o dirección ip de la Fachada @Firma.
- pagina_firmable.htm: cambiar la dirección ip del campo action del formulario por el nombre o dirección ip de la Fachada @Firma.

Una vez finalizados los pasos anteriores, se accederá mediane la url:

https://<servidor_aplicaciones>:<puerto>/ejemplofirma/inicio.htm



9.2 Aplicación "ejemploMultifirma"

La aplicación "ejemploMultifirma" muestra un ejemplo del proceso de Multifirma Web.

Se hace uso de la siguiente interfaz de la plataforma:

- com.telventi.multifirma.MultiFirmaApiFacade: para realizar la multifirma web.

La aplicación está desarrollada en JAVA - JSP y se encuentra disponible utilizando las dos tecnologías de la plataforma: RMI-IIOP y WebServices.

El código de la aplicación se encuentra disponible en los siguientes directorios del CD Desarrollo, respectivamente:

- "Modulo Firma / Multifirma Web / Ejemplo RMI-IIOP / ejemploMultiFirma.war"
- "Modulo Firma / Multifirma Web / Ejemplo WebServices / Codigo"
- "Modulo Firma / Multifirma Web / Ejemplo WebServices / ejemploMultiFirmaWS.war"

9.2.1 Poner en marcha la aplicación RMI-IIOP

Es una aplicación WEB, así pues necesitamos un servidor de aplicaciones, por ejemplo JBOSS.

En el directorio "Modulo Firma / MultiFirma Web / Ejemplo RMI-IIOP / ejemploMultiFirma.war" del CD Desarrollo se encuentran los ficheros necesarios para poner en marcha la aplicación:

- EjemploMultiFirma.war: aplicación que debe ser desplegada en el directorio de deploy del servidor de aplicaciones. (ej: JBOSS_HOME/server/all/deploy)

Configurar los parámetros de "MultiFirma Web" de la aplicación "ejemploFirma" en la Herramienta de administración.

Una vez copiado el directorio modificar los siguientes ficheros:

- principal.jsp: hacer coincidir la clave DES con la de la Herramienta de Adiministración, en la creación del objeto ConexionMultiFirma poner la dirección ip del Servidor @Firma y en la llamada al componente principalpaginafirmada.jsp cambiar la dirección ip por el nombre o dirección ip de la Fachada @Firma.

- firmantes.jsp: en la creación del objeto ConexionMultiFirma poner la dirección ip del Servidor @Firma.

- firmar.jsp: en la creación del objeto ConexionMultiFirma poner la dirección ip del Servidor @Firma, cambiar la dirección ip por el nombre o dirección ip de la Fachada @Firma en el campo action del formulario y en la inclusión del javascript. Poner en el campo CodeBase del formulario la dirección del Servidor de Aplicaciones donde se ha desplegado la aplicación.

Una vez finalizados los pasos anteriores, se accederá mediane la url:

https://<servidor_aplicaciones>:<puerto>/ejemploMultiFirma/Multifirma.htm

9.2.2 Poner en marcha la aplicación WebServices

Es una aplicación WEB, así pues necesitamos un servidor de aplicaciones, por ejemplo JBOSS.

En el directorio "Modulo Firma / Multifirma Web / Ejemplo WebServices / ejemploMultiFirmaWS" del CD Desarrollo se encuentran los ficheros necesarios para poner en marcha la aplicación:

- ejemploMultiFirmaWS.war: aplicación que debe ser desplegada en el directorio de deploy del servidor de aplicaciones. (ej: JBOSS_HOME/server/all/deploy)
- demoMultiFirmaWS.properties: parámetros para poder ejecutar la aplicación. Este fichero debe colocarse en el directorio de ejecución del servidor (e: JBOSS_HOME/bin). Los parámetros a configurar son los siguientes:
    - servidorfirma==<nombre o ip del servidor de firma de la plataforma de firma>
    - usuario=<usuario para acceso JAAS WebServices a servidor de firma>
    - password=<password para acceso JAAS WebServices a servidor de firma>
    - trustedstore=<url del fichero trustkeystore que contiene el certificado digital SSL del servidor de firma de la plataforma de firma>Ej:c:\trustkeystore

  - trustedstorepassword=<password el keystore anterior>
  - idaplicacion=<identificador de aplicación registrada en la herramienta de administracion>Ej: DEMOBLOQUESWS

- trustKeystore: keystore que contiene la clave pública del certificado digital del servidor de firma de la plataforma de firma. Se utiliza para el acceso mediante SSL al servidor de firma.

Configurar los parámetros de "MultiFirma Web" de la aplicación "ejemploFirma" en la Herramienta de administración.

Una vez copiado el directorio modificar los siguientes ficheros:

- principal.jsp: hacer coincidir la clave DES con la de la Herramienta de Adiministración y en la llamada al componente principalpaginafirmada.jsp cambiar la dirección ip por el nombre o dirección ip de la Fachada @Firma.

- firmar.jsp: cambiar la dirección ip por el nombre o dirección ip de la Fachada @Firma en el campo action del formulario y en la inclusión del javascript. Poner en el campo CodeBase del formulario la dirección del Servidor de Aplicaciones donde se ha desplegado la aplicación.

Una vez finalizados los pasos anteriores, se accederá mediane la url:

https://<servidor_aplicaciones>:<puerto>/ejemploMultiFirmaWS/Multifirma.htm

9.3 Aplicación "ejemploMultifirmaMasiva"

La aplicación "ejemploMultifirma" muestra un ejemplo del proceso de Multifirma Masiva.

Se hace uso de la siguiente interfaz de la plataforma:

- com.telventi.multifirmamasiva.MultiFirmaWebMasiva: para realizar la multifirma masiva.

La aplicación está desarrollada en JAVA - JSP y se encuentra disponible utilizando las dos tecnologías de la plataforma: RMI-IIOP y WebServices.

El código de la aplicación se encuentra disponible en los siguientes directorios del CD Desarrollo, respectivamente:

- "Modulo Firma / Multifirma Masiva / Ejemplo RMI-IIOP / ejemploMultiFirmaMasiva.war"
- "Modulo Firma / Multifirma Masiva / Ejemplo WebServices / Codigo"

- "Modulo Firma / Multifirma Masiva / Ejemplo WebServices / ejemploMultiFirmaMasivaWS.war"

9.3.1 Poner en marcha la aplicación RMI-IIOP

Es una aplicación WEB, así pues necesitamos un servidor de aplicaciones, por ejemplo JBOSS.

En el directorio "Modulo Firma / MultiFirma Masiva / Ejemplo RMI-IIOP / ejemploMultiFirmaMasiva.war" del CD Desarrollo se encuentran los ficheros necesarios para poner en marcha la aplicación:

- EjemploMultiFirma.war: aplicación que debe ser desplegada en el directorio de deploy del servidor de aplicaciones. (ej: JBOSS_HOME/server/all/deploy)

Damos de alta una Página Multifirma Masiva en la Herramienta de Administracón y la registramos.

Una vez copiado el directorio modificar los siguientes ficheros:

- paginafirmada.jsp: en la creación del objeto ConexionMultiFirmaMasiva poner la dirección ip del Servidor @Firma.

- procesarmultifirmamasiva.jsp: en la creación del objeto ConexionMultiFirmaMasiva poner la dirección ip del Servidor @Firma.

Una vez finalizados los pasos anteriores, se accederá mediane la url:

https://<servidor_aplicaciones>:<puerto>/ejemploMultiFirmaMasiva/MultifirmaMasiva.htm

9.3.2 Poner en marcha la aplicación WebServices

Es una aplicación WEB, así pues necesitamos un servidor de aplicaciones, por ejemplo JBOSS.

En el directorio "Modulo Firma / Multifirma Masiva / Ejemplo WebServices / ejemploMultiFirmaMasivaWS" del CD Desarrollo se encuentran los ficheros necesarios para poner en marcha la aplicación:

- ejemploMultiFirmaMasivaWS.war: aplicación que debe ser desplegada en el directorio de deploy del servidor de aplicaciones. (ej: JBOSS_HOME/server/all/deploy)
- demoMultiFirmaMasivaWS.properties: parámetros para poder ejecutar la aplicación. Este fichero debe colocarse en el directorio de ejecución del servidor (e: JBOSS_HOME/bin). Los parámetros a configurar son los siguientes:
    - servidorfirma==<nombre o ip del servidor de firma de la plataforma de firma>
    - usuario=<usuario para acceso JAAS WebServices a servidor de firma>
    - password=<password para acceso JAAS WebServices a servidor de firma>
    - trustedstore=<url del fichero trustkeystore que contiene el certificado digital SSL del servidor de firma de la plataforma de firma>Ej:c:\trustkeystore

- o trustedstorepassword=<password el keystore anterior>
- o idaplicacion=<identificador de aplicación registrada en la herramienta de administracion>Ej: DEMOBLOQUESWS

- trustKeystore: keystore que contiene la clave pública del certificado digital del servidor de firma de la plataforma de firma. Se utiliza para el acceso mediante SSL al servidor de firma.

Damos de alta una Página Multifirma Masiva en la Herramienta de Administracón y la registramos.

Una vez finalizados los pasos anteriores, se accederá mediane la url:

https://<servidor_aplicaciones>:<puerto>/ejemploMultiFirmaMasivaWS/Multifirma.htm