Firma No Criptográfica

Manual de Integración

Versión 1.0

Fecha de la versión 01/12/2022

Madrid, 01 de diciembre del 2022

Elaborado por la Secretaría General de Administración Digital

NIPO: Pendiente de asignación.

ÍNDICE

1. OBJETO

El objeto de este manual es servir de guía para el conocimiento del servicio de Firma No Criptográfica proporcionado por Autentica, tanto desde un punto de vista de funcionalidad del mismo, como de los mecanismos de integración disponibles para las aplicaciones que deseen utilizarlo.

2. DESCRIPCIÓN

La Resolución de 23 de febrero de 2022, de la Secretaría General de Administración Digital, por la que se establecen las condiciones de uso de firma electrónica no criptográfica vinculada a "AutenticA", para la relación con la Administración General del Estado y sus organismos públicos y entidades de derecho público vinculados o dependientes describe el funcionamiento del Sistema de Firma No Criptográfica (en adelante FNC) basado en Autentica.

Dicho sistema de firma no criptográfica, permitirá a los usuarios de las aplicaciones integradas con Autentica la firma de documentos PDF (firma PAdES) o XML (firma XAdES) sin necesidad de utilizar aplicaciones terceras, como Autofirma, simplificando de esta manera el proceso de firma y evitando los problemas habituales del uso y configuración de este tipo de herramientas.

El sistema de FNC basado en Autentica contempla tres posibles mecanismos de invocación:

- Mediante el envío de un XML que contenga el fichero a firmar
- Por medio de una página del propio FNC de Autentica
- Mediante la invocación de servicios web.

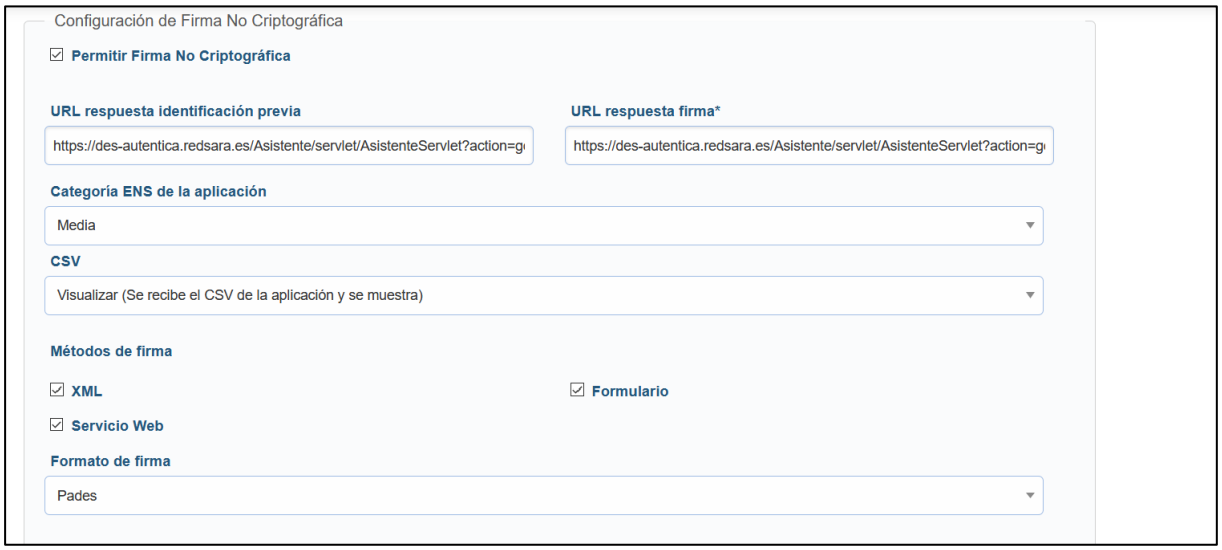
Asimismo, se ha implementado una funcionalidad dentro de la aplicación Autentica que permitirá consultar y descargar las evidencias recopiladas durante el proceso de firma de un documento mediante el sistema de FNC.

Para dar una mayor seguridad jurídica a la FNC, así como mejorar la presentación de los documentos generados, se ha integrado el sistema de FNC con la librería CSV Creator.

3. MÓDULO DE APROVISIONAMIENTO

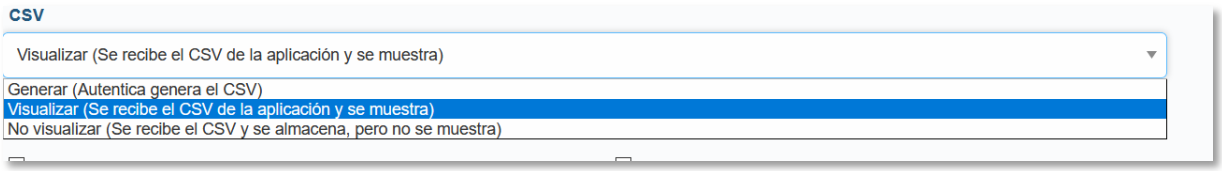
3.1. Configuración

Para que una aplicación pueda utilizar los servicios de FNC proporcionados por Autentica, deberá estar dada de alta en Autentica y tener activada la configuración correspondiente.



Para ello, deberá tener marcado el check “Permitir Firma No Criptográfica”, de tal forma que se activarán el resto de opciones de configuración.

- URL respuesta identificación previa: se indica la URL de respuesta a la que se debe devolver el hash generado en el caso de las opciones de invocación XML y servicios web.
- URL respuesta firma: se indica la URL de respuesta a la que se devolverá el documento firmado así como el justificante.
- Categoría ENS de la aplicación: puede ser “Alta”, “Media” o “Baja”. Es un dato descriptivo.
- CSV: se indica si la aplicación envía el CSV o no, pudiendo tener tres opciones:



  - Generar (Autentica genera el CSV): en este caso, desde Autentica se solicita la generación del CSV con el prefijo “AUT-” a la librería CSV Creator. Aunque la aplicación que utiliza FNC envíe un CSV, éste no será tenido en cuenta.

- Visualizar (Se recibe el CSV de la aplicación y se muestra). La aplicación que utiliza FNC deberá enviar un CSV que será el que se agregue al documento firmado. En el caso de que la aplicación no envíe un CSV no se podrá continuar con la operación solicitada.
- No visualizar (Se recibe el CSV y se almacena, pero no se muestra). Como en el caso anterior, la aplicación que utiliza FNC deberá enviar un CSV, pero este no será agregado al documento firmado. En el caso de que la aplicación no envíe un CSV, no se podrá continuar con la operación solicitada.

- Métodos de firma: se trata de tres opciones teniendo que marcar al menos una de ellas, ya que no son excluyentes entre sí. No se podrá utilizar la modalidad de FNC correspondiente si no está seleccionada en la configuración de la aplicación.
  - XML
  - Formulario
  - Servicio web
- Formato de firma: se seleccionará el formato PAdes (para documentos en formato PDF) o XAdes.. FNC comprobará que el formato seleccionado para la aplicación corresponde al fichero que se está intentando firmar.

- Otras opciones:
  - Previsualizar documento: si desea que FNC muestre una pantalla desde la que el usuario pueda visualizar y revisar el documento antes de la firma (sólo para ficheros PDF y no disponible para invocación mediante servicios web)
  - Emitir justificante: si se desea que FNC genere un justificante de la operación de firma realizada.

- Recabar consentimiento: si se desea que FNC muestre en pantalla una cláusula para informar al usuario y recabar su consentimiento de la operación de firma que se va a realizar (no disponible para invocación mediante servicios web).

3.2. Consulta de Evidencias de Firma No Criptográfica

3.2.1. Módulo de aprovisionamiento

Existe en el módulo de aprovisionamiento de Autentica una opción para obtener los datos de trazabilidad de las operaciones FNC que se realicen.

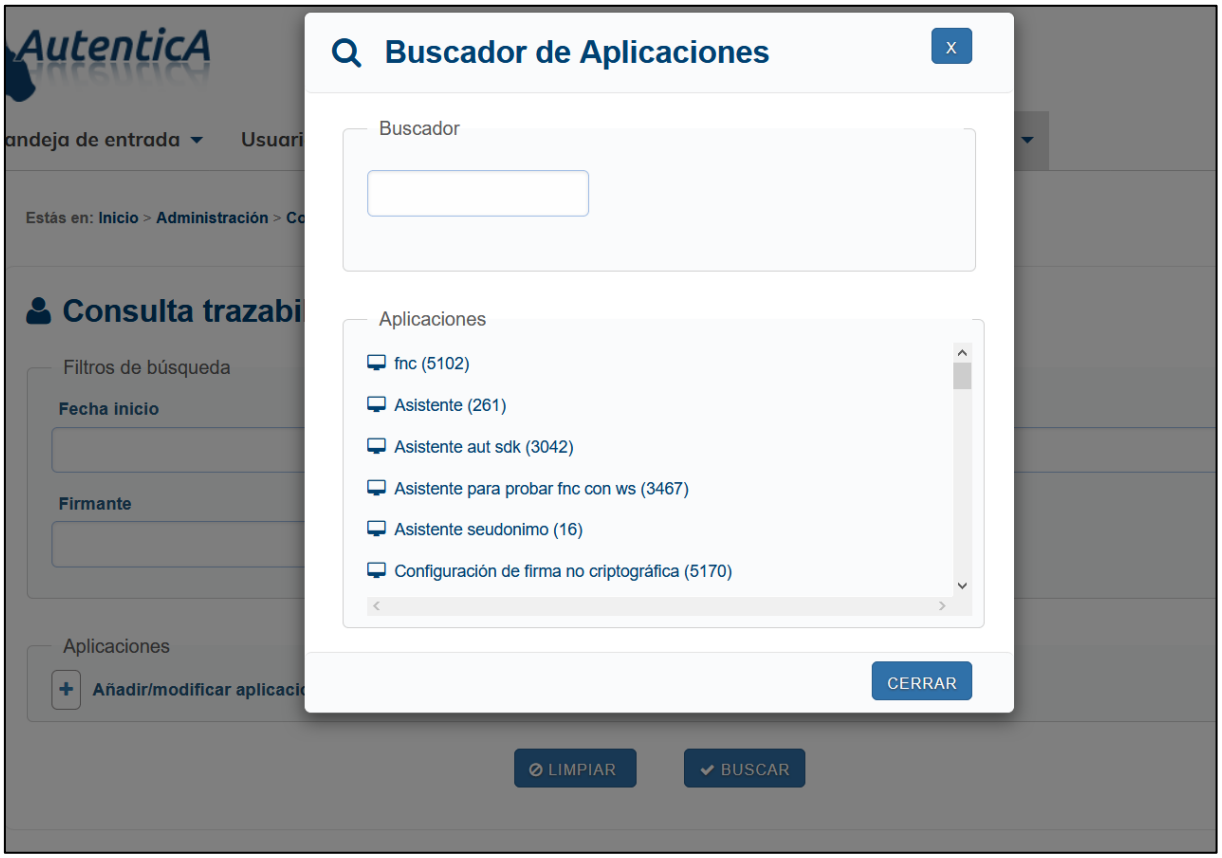
Mediante el formulario, se podrá filtrar por diversos criterios, que serán los siguientes:

- Fecha de inicio: se busca desde una fecha determinada.
- Fecha de fin: se busca hasta una fecha determinada.
- Firmante: se busca por un NIF/NIE de un usuario que haya realizado la operación de FNC.

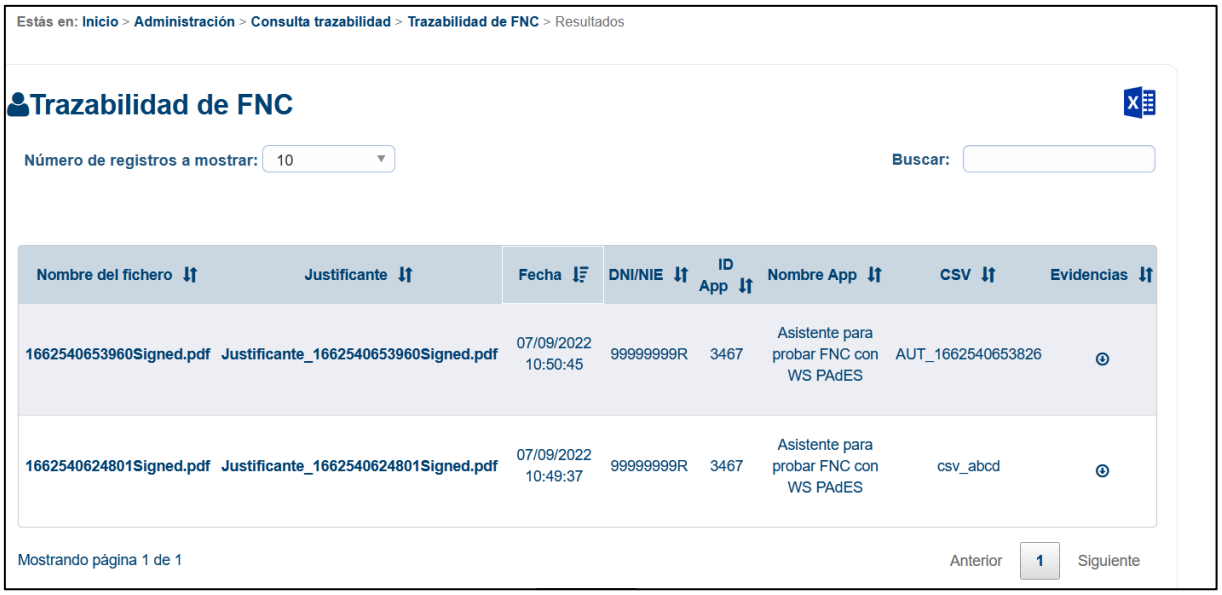
- CSV: se busca por un CSV.
- Aplicaciones: se selecciona una o más aplicaciones desde la que se haya utilizado FNC.

En la imagen se puede ver el formulario de búsqueda.

En la siguiente imagen se puede observar el buscador de aplicaciones.

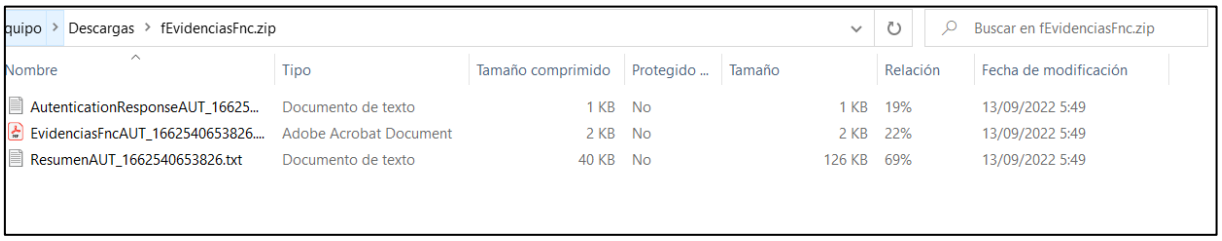


De esta forma, se obtendrán una serie de resultados que podrán ser consultados por un usuario con permisos de administrador.



En la columna “Evidencias” se podrán descargar las evidencias de una operación concreta de FNC realizada por medio de un archivo .zip.

A continuación se muestra el contenido del fichero .zip con las evidencias.



3.2.2. Servicios web

Por medio de los servicios web también se podrá realizar la descarga de las evidencias.

Para ello, se encuentra el servicio web “getEvidence”, dentro del catálogo de servicios web de FNC. Para su uso se deberán indicar los siguientes parámetros:

- WebUser: objeto que indica el identificador de la aplicación que está invocando el servicio web “getEvidence”.
- String sCSV: String con el CSV del documento del cual se quieren descargar las evidencias.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:fnc="http://fnc.ws.babel.es" xmlns:obj="http://obj.fnc.ws.babel.es">
  <soapenv:Header/>
  <soapenv:Body>
    <fnc:getEvidence>
      <fnc:webUser>
        <obj:webName>?</obj:webName>
      </fnc:webUser>
```

```
        <fnc:sCSV>?</fnc:sCSV>
      </fnc:getEvidence>
    </soapenv:Body>
</soapenv:Envelope>
```

El servicio web devuelve el mismo fichero de evidencias .zip que se obtiene si se realiza esta operación desde el módulo de aprovisionamiento.

Como requisitos para la utilización de este servicio web se indican los siguientes:

- Debe ser invocado desde una IP que se encuentre dada de alta en la ficha de la aplicación
- Se está utilizando el certificado configurado en la ficha de la aplicación para la utilización de los servicios web.
- El id de aplicación que se indica en el objeto WebUser es el mismo que la aplicación con la que se firmó el documento.

El servicio devuelve un objeto de tipo “EvidenceResponse” que contiene dos elementos:

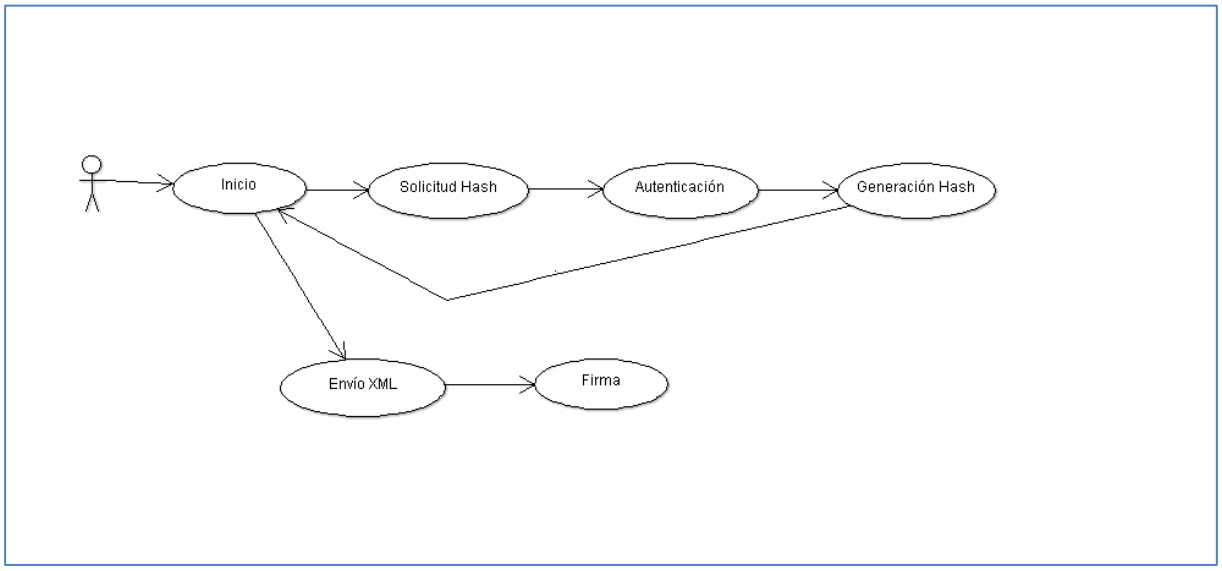
- evidence: String con un mensaje indicando un posible error que se ha producido o null, en el caso de que se haya devuelto la evidencia.
- message. String en base64 que contiene el fichero .zip con la evidencia solicitada.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <getEvidenceResponse xmlns="http://fnc.ws.babel.es">
      <getEvidenceReturn>
        <evidence xsi:nil="true"/>
        <message>UEsDBBQACAgIAB06LVUAAAAA[….]AAAA==</message>
      </getEvidenceReturn>
    </getEvidenceResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

4. FORMAS DE INVOCACIÓN

4.1. XML

La invocación por medio de XML se realiza en dos pasos, en un primer paso, se invoca una acción de FNC para obtener un hash que asociará el usuario que realiza la petición y la aplicación desde la que se llama al módulo de FNC y que deberá ser proporcionado en el momento de realizar la solicitud de firma. Para ello, se solicitará, por medio de Autentica, que el usuario se valide, ya sea por medio de certificado o usuario y contraseña.



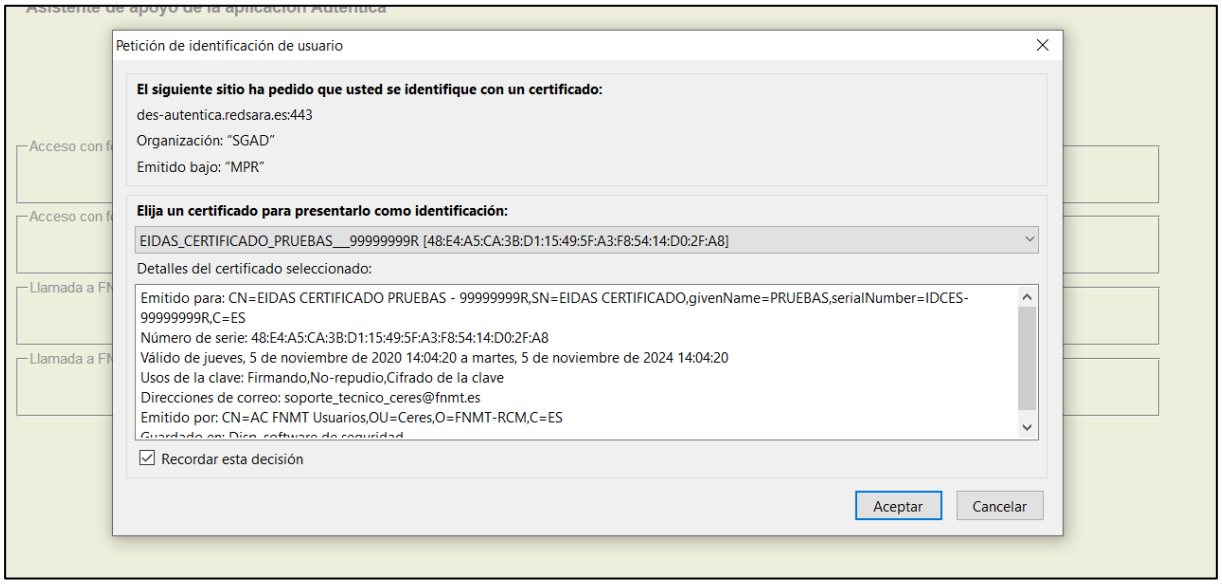
4.1.1. Generación de hash

Se invoca la acción “goToAutenticaFnc” indicando como parámetro el id de aplicación, de tal forma que se solicita la autenticación del usuario generándose, en caso de éxito, un hash que asocia la aplicación desde la que se va a realizar la solicitud con el fichero a firmar y el usuario que la ha realizado, teniendo que ser proporcionada en el momento de realizar la solicitud de firma.

A continuación se indica un ejemplo de invocación de este caso:

http://des-autentica.redsara.es/Autentica/servlet/AutenticaServlet?action=goToAutenticaFnc&appId=261

- appId indica el identificador del método de generación de hash.



La respuesta, con el hash que se ha generado, se devolverá a la URL indicada en el apartado "URL respuesta configuración previa" de la configuración de la aplicación, por medio del parámetro "FNC_XML", que contendrá un XML cuyo xsd es el siguiente:

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="fnc">
    <xs:complexType>
      <xs:sequence>
        <xs:element type="xs:long" name="hash"/>
        <xs:element type="xs:string" name="userName"/>
        <xs:element type="xs:short" name="appId"/>
        <xs:element type="xs:short" name="signId"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

A continuación se indica un XML de ejemplo, donde el hash generado se encuentra dentro del elemento "hash".

```
<?xml version='1.0' encoding='ISO-8859-1'?>
<fnc>
        <hash>1628680368248</hash>
        <userName>1R</userName>
        <appId>261</appId>
```

```
        <signId>200</signId>
</fnc>
```

Como paso previo a la generación del hash, se comprobará que la aplicación tenga configurada la invocación por medio de un XML, ya que en el caso de que no sea así no se permitirá realizar la operación, mostrándose una pantalla como la que sigue.

4.1.2. Invocación

Desde la aplicación externa se enviará un XML que contendrá el fichero PDF a firmar. A continuación se indica el xsd de la petición:

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="fnc">
    <xs:complexType>
      <xs:sequence>
        <xs:element type="xs:string" name="fileName"/>
        <xs:element type="xs:string" name="file"/>
        <xs:element type="xs:string" name="ext"/>
        <xs:element type="xs:string" name="userName"/>
        <xs:element type="xs:short" name="appId"/>
        <xs:element type="xs:long" name="hash"/>
        <xs:element type="xs:string" name="csv"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Los elementos a los que se hace referencia son los siguientes:

- fileName: indica el nombre del fichero que se va a firmar.

- file: contiene el contenido del fichero en formato base 64.
- ext: contiene la extensión del fichero.
- userName: contiene el identificador (NIF/NIE habitualmente) del usuario que ha realizado la solicitud de FNC.
- appId: indica el identificador numérico de la aplicación externa desde la que se está invocando la operación de FNC.
- hash: se indica el código hash que se ha obtenido en el punto anterior.
- csv: se indica el CSV del documento en caso de que se proporcione por parte de la aplicación. Si no se proporciona ningún valor, el sistema de FNC solicitará la generación de un CSV a la librería CSV Creator con el siguiente formato: “AUT-[código alfanumérico]”. El CSV se añadirá al documento firmado mediante la copia realizada a través de la librería antes mencionada.

Se indica a continuación un XML de ejemplo:

```
<?xml version="1.0" encoding="UTF-8"?>
<fnc>
        <fileName>nombreFichero.pdf</fileName>
        <file>JVBERi0xLjQKJeLjz9MKMy[….]</file>
        <ext>.PDF</ext>
        <userName>1R</userName>
        <appId>261</appId>
        <hash>1628680368248</hash>
        <csv>asiscsv_1628680368450</csv>
</fnc>
```

Para llamar a la acción de FNC, invocamos la acción “goToXMLFncDocument”; un ejemplo de invocación desde un formulario sería como el que sigue:

```
<form name="fEnviar" id="fEnviar" action="https://des-autentica.redsara.es/Fnc/servlet/FncServlet" method="post">
        <br/>
        <input type="submit" value="Enviar" id="submitEnviar" />
        <br/>
        <input type="hidden" id="action" name="action" value="goToXMLFncDocument"/>
        <input type="hidden" id="type" name="type" value="JSP"/>
        <input type="hidden" id="userName" name="userName" value="1R"/>
        <input type="hidden" id="appId" name="appId" value="261"/>
        <input type="hidden" id="hash" name="hash" value="1628680368248"/>
        <input type="hidden" id="csv" name="csv" value="ASISCSV_1629875667140"/>
        <input type="hidden" id="sFileXML" name="sFileXML" value="<?xml version='1.0' encoding='UTF-
```

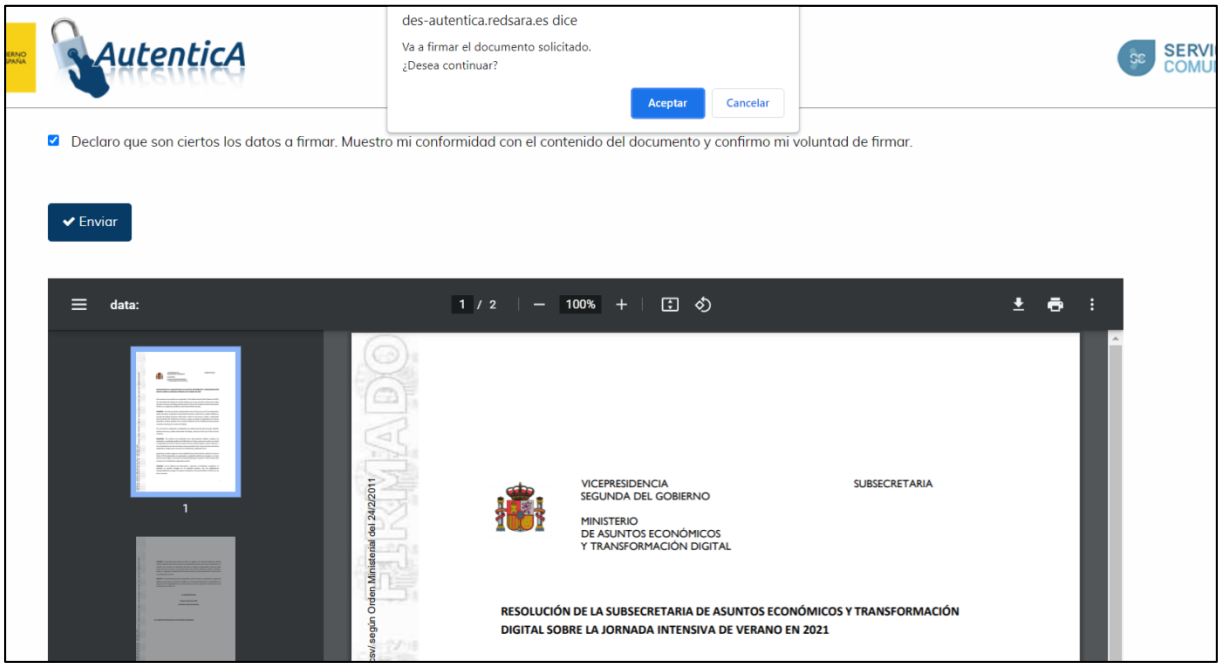
```
8'?><fnc><fileName>fichero.pdf</fileName><file>JVBERi0xLjQKJeLjz[...]</file><ext>.pdf</ext><userName>1R</
userName><appId>261</appId><hash>1628680368248</hash><csv> asiscsv_1628680368450</csv></fnc>"/>
</form>
```

Posteriormente, se firmará el documento o, dependiendo de la configuración de la aplicación, se mostrará alguna pantalla intermedia.

Las opciones posibles son:

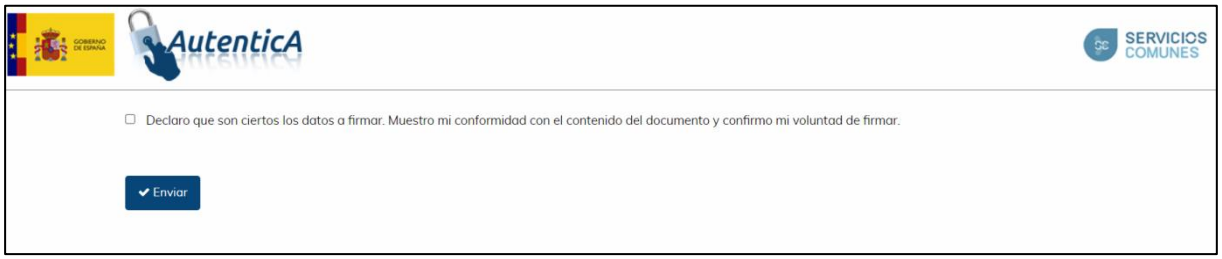
- firma directa del documento.
- Mostrar una pantalla intermedia para previsualización del documento junto con un una opción donde se solicita, de forma obligatoria, el consentimiento del usuario a la operación de firma.

Será necesario, en este caso, aceptar el consentimiento para poder continuar con el proceso de FNC.

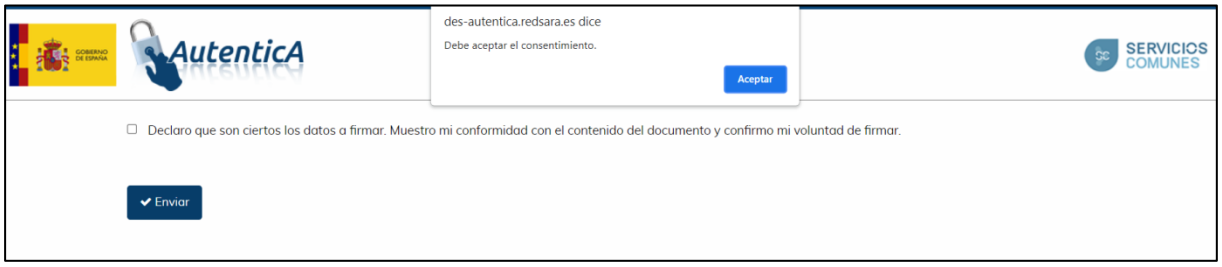


- Mostrar una pantalla intermedia para previsualización del documento, sin que sea necesario el consentimiento del usuario a la operación de firma.

- Mostrar una pantalla intermedia para recabar el consentimiento por parte del usuario.

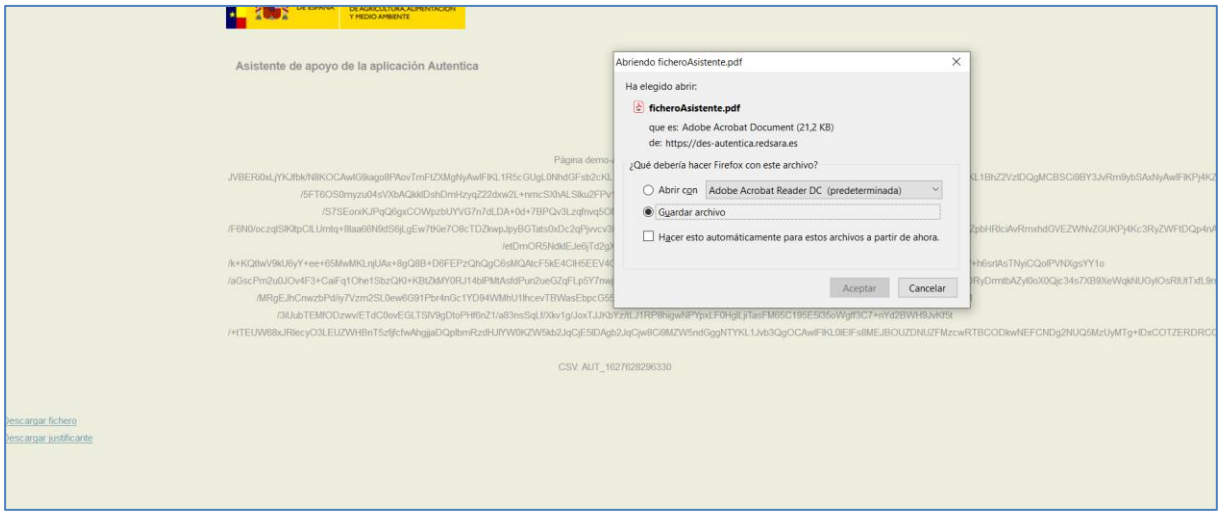


Como en el resto de casos, será necesario aceptar el consentimiento.



Una vez que se haya realizado la firma, se invocará la URL de respuesta proporcionada por parte de la aplicación y configurada en el apartado “URL respuesta firma”, devolviendo los siguientes tres parámetros:

- fileData: contiene el documento firmado en formato base64.
- justitData: contiene el justificante firmado en formato base64.
- csv: contiene el csv del documento.



Con los datos recibidos en el parámetro “fileData”, el usuario podrá descargarse el documento firmado:

Así mismo, con los datos recibidos en el parámetro "justifData", el usuario podrá descargarse el justificante de firma:

El documento PDF contendrá datos como

- Nombre del usuario.
- DNI/NIE del usuario.
- Fecha y hora de la firma.
- Nombre del documento.
- URL de validación del documento.

- CSV del documento.

4.2. Formulario FNC

4.2.1. Invocación

Desde una aplicación externa, se invocará la acción “goToAutentica” junto con un identificador que será proporcionado por los responsables de FNC y que dependerá del entorno, de tal forma que se accederá a un formulario del módulo de FNC mediante el cual se podrá seleccionar el fichero que vaya a ser firmado y con cuyo contenido se compondrá de forma interna un XML similar al visto en el punto anterior.

Para ello, será necesario indicar los siguientes parámetros:

- appId: identificador del formulario FNC.
- appParam: identificador de la aplicación que invoca el módulo de FNC.

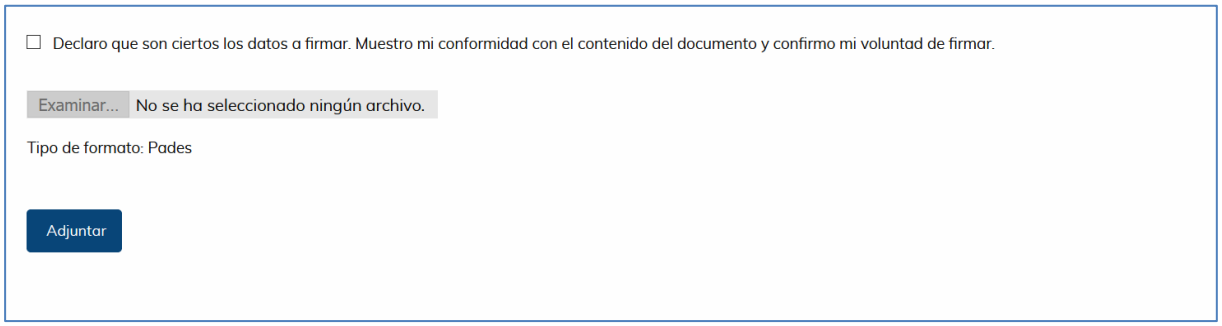
A continuación se indica un ejemplo de invocación:

```
<form id="f1" action="https://des-autentica.redsara.es/Autentica/servlet/AutenticaServlet" method="post">
        <input type="hidden" name="action" id="action" value="goToAutentica"/>
        <input type="hidden" name="appId" id="appId" value="4883"/>
        <input type="hidden" name="appParam" id="appParam" value="261"/>
        <input type="submit" value="Acceso" id="Acceso" />
</form>
```

Para este caso no será necesario generar un código hash de invocación.

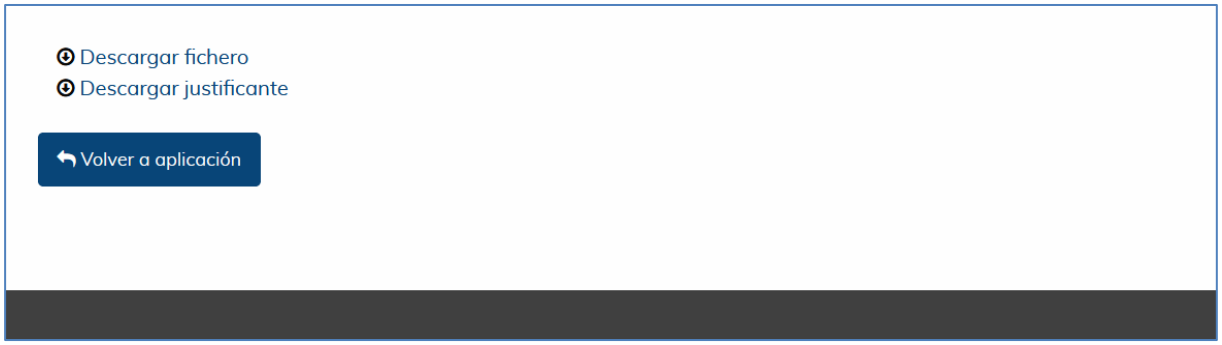
Asimismo, será un requisito que la aplicación esté configurada para poder utilizar este medio de invocación, en caso contrario, se mostrará la siguiente pantalla:

Una vez que se haya comprobado que el método de acceso es correcto, se mostrará el formulario para adjuntar el documento, así como la solicitud de consentimiento en el caso de que la aplicación esté así configurada.

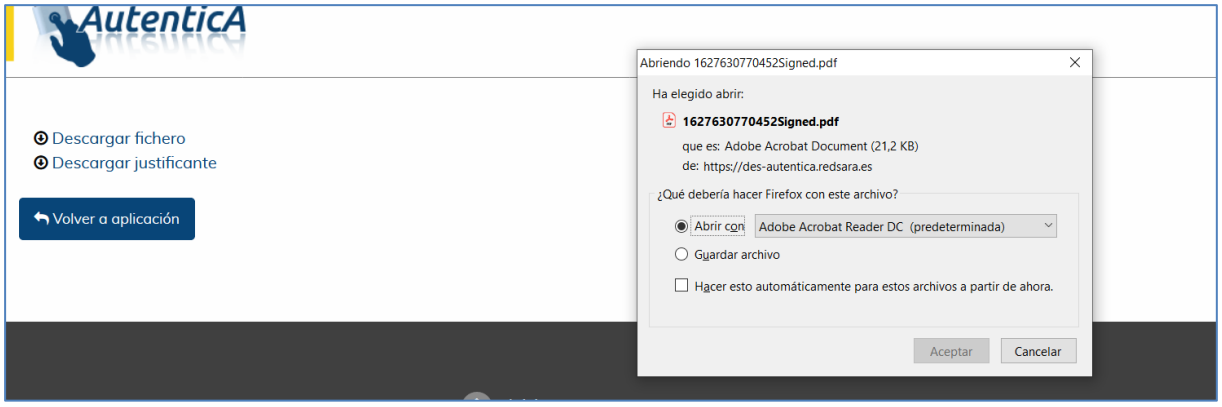


Además, una vez seleccionado el fichero, se mostrará una pantalla de previsualización del documento en el caso de que la aplicación así esté configurada.

Una vez que se haya realizado la firma, se mostrará una página del módulo de FNC con los enlaces correspondientes al fichero firmado y al justificante que se ha generado.,



Por medio de estos enlaces, el usuario podrá descargarse el documento firmado y el justificante generado de la operación de firma realizada.



4.3. Servicio web

La invocación del método de FNC por medio de servicios web es similar al visto para el caso de XML, salvo que se invocarán diversos métodos de servicios web para las distintas operaciones a realizar.

Al igual que en el supuesto de invocación por medio de XML, se deben realizar dos pasos:

- Invocación de un servicio web con el que se obtendrá un hash que asociará al usuario que realiza la petición y que se ha autenticado por medio de este servicio web y la aplicación desde la que se llama al módulo de FNC.
- Invocación de un servicio web para realizar la solicitud de firma y que, entre otros datos, recibirá el hash generado anteriormente.

A continuación se indica el wsdl de los servicios web, cuyo endpoint sería, dependiendo del entorno, similar al que sigue y donde se pueden revisar los servicios web disponibles.

http://autentica.redsara.es/FncWS/services/FncWS?wsdl.

4.3.1. Generación de hash

La aplicación externa invocará el servicio web “getHash” del catálogo de servicios web de FNC para obtener un hash generado desde FNC y al cual deberá indicar los siguientes parámetros:

- WebUser: objeto que indica el identificador de la aplicación que está invocando los servicios web.
- String sUserName: String con el NIF/NIE del usuario.
- String sCert: String con la clave pública del certificado del usuario.

A continuación se indica un ejemplo de invocación del servicio web:

```
HashResponse hashResponse = fncWS.getHash(webUser, sUserName, sCert);
```

El servicio web validará, mediante @firma, si el certificado recibido es válido. En el caso de que así sea, se cotejará que corresponde al DNI/NIF indicado en el elemento sUserName. Posteriormente, se comprobará la existencia de ese usuario en el repositorio de usuarios de Autentica.

Además, se comprobará que la aplicación tenga definida la invocación por medio de servicios web, ya que en el caso de que no sea así no se permitirá realizar la operación.

El servicio devolverá un objeto de tipo “HashResponse” que, entre otros datos, contendrá el hash que será utilizado para la invocación posterior de la operación de FNC. Este objeto contiene las siguientes propiedades:

- Hash: String con el hash generado.
- hashXML: String con el hash generado en formato XML, tal como se vio en el punto referente a la invocación XML.

En todos los casos en los que se produzca algún tipo de error, se devolverá en la propiedad hash el texto “ERROR”, junto con la descripción del mismo.

4.3.2. Invocación

Una vez que se ha obtenido el hash, se invocará al servicio web “getFncSign” enviando este dato como parámetro como propiedad de un objeto FncFileBean que contendrá, además, el fichero que se va a firmar en formato base64. Los parámetros que se indicarán al servicio web son:

- WebUser: objeto que indica el identificador de la aplicación que está invocando los servicios web.
- FncFileBean: Objeto que contiene las siguientes propiedades:
    - sName: String que indica el nombre del fichero.
    - sExt: String que indica la extensión del fichero.
    - sContent: String con el contenido del fichero en base64.
    - sUserName: String con el NIF/NIE del usuario.
    - sAppId: String con el identificador de la aplicación invocante.
    - sHash: String con el hash generado.
    - sCsv: String con el CSV.

A continuación se indica un ejemplo de invocación del servicio web:

```
SignResponse signResponse = fncWS.getFncSign(webUser, fncFileBean);
```

Al invocar el servicio web se comprobará que

- la aplicación invocante tiene permiso de uso del método de servicios web (al igual que ocurre con el servicio “getHash”)
- el hash es correcto y no ha caducado.
- el usuario que realiza la petición se encuentra registrado en Autentica.

Una vez que se haya realizado la firma, el servicio web devolverá, en el caso de que la operación se haya realizado correctamente, un objeto de tipo “SignResponse”, que contendrá las propiedades:

- fileContent: con el contenido del fichero en base64.
- fileJustif: con el contenido del justificante en base64.
- Csv: String con el CSV del documento.

En el caso de que la operación no se haya realizado correctamente, estos campos no tendrán contenido, indicándose en la propiedad message un texto con el error.

4.3.3. Ejemplo de invocación SoapUI

4.3.3.1. Configuración

Se indica la configuración a realizar en la herramienta SoapUI para invocar los servicios web de FNC.

Se puede obtener más información en la propia página de ayuda de SoapUI:

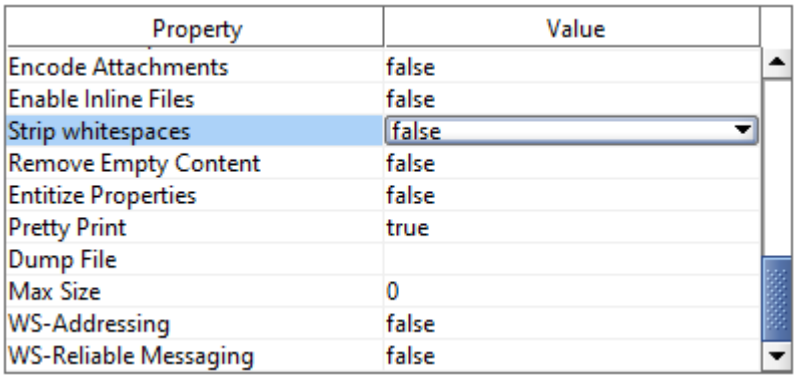
https://www.soapui.org/soapui-projects/ws-security.html

Para realizar este manual se ha utilizado la versión 5.5.0.

- Se accede a las propiedades del proyecto (doble clic sobre el mismo u opción del botón derecho “Show Project View”).
- Pestaña “WS-Security Configurations”
- Pestaña “Keystores”. Se añade el archivo .keystore que está integrado en los servicios web de Autentica
    - Source: [...]\sello_dtic.ks
    - Status: OK (valor que indica SoapUI si la contraseña introducida es correcta).
    - Password: [……]
  - Pestaña “Outgoing WS-Security Configurations”: se añade una nueva configuración especificando un nombre único, ej. “outgoing_fnc” y se completan los siguientes datos:
    - Default Username/Alias: [en blanco]
    - Default Password: [en blanco]
    - Actor: fncSig
    - Must Understand: [Seleccionado]

  - En el apartado inferior se añade una nueva entrada WSS (WSS Entry) seleccionando el tipo “Signature” en la ventana de diálogo “Add WSS Entry”.

- Se rellena la siguiente configuración de la entrada “Signature”
  - Keystore: sello_dtic.ks
  - Alias: sello entidad sgad pruebas
  - Password: [……]
  - Key Identifier Type: BinarySecurityToken
  - Signature Algorithm: <default>
  - Signature Canonicalization: <default>
  - Digest Algorithm: <default>
- Se crea una nueva petición y se completan los datos correspondientes.
  - Se añade la cabecera con la opción inferior “Auth” y se abre un formulario:
    - Authorization: Basic
    - Outgoing WSS: outgoing_fnc
  - En la esquina inferior izquierda se habilita la opción de whitespaces seleccionando el valor “true”.



4.3.3.2. Invocación de los servicios web

En primer lugar, hay que invocar el servicio web “getHash” para que el usuario se valide en Autentica y obtener el hash que se debe proporcionar posteriormente.

Ejemplo de petición (certificado cortado por longitud)

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:fnc="http://fnc.ws.babel.es" xmlns:obj="http://obj.fnc.ws.babel.es">
  <soapenv:Header/>
  <soapenv:Body>
    <fnc:getHash>
      <fnc:webUser>
        <obj:webName>3467</obj:webName>
      </fnc:webUser>
      <fnc:sUserName>99999999R</fnc:sUserName>

        <fnc:sCert>MIIHmDCCBoCgAwIBAgIQTeR66BdqdRpcpwM07UCRMjANBgkqhkiG9w0BAQsFADBLMQ
[……]du8gj4IkXpAxmQ==</fnc:sCert>
```

```
    </fnc:getHash>
  </soapenv:Body>
</soapenv:Envelope>
```

Ejemplo de respuesta:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <getHashResponse xmlns="http://fnc.ws.babel.es">
      <getHashReturn>
        <hash>1645700394233</hash>
        <hashXML><![CDATA[<?xml version='1.0' encoding='ISO-8859-1'?><fnc><hash>1645700394233</hash><userName>99999999R</userName><appId>3467</appId><signId>2965</signId></fnc>]]></hashXML>
      </getHashReturn>
    </getHashResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Posteriormente, para no tener un error de caducidad del hash, se deberá invocar el servicio web “getFncSign”

Ejemplo de petición:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:fnc="http://fnc.ws.babel.es" xmlns:obj="http://obj.fnc.ws.babel.es" xmlns:bean="http://beans.fnc.ws.babel.es">
  <soapenv:Header/>
  <soapenv:Body>
    <fnc:getFncSign>
      <fnc:webUser>
        <obj:webName>3467</obj:webName>
      </fnc:webUser>
      <fnc:fncFileBean>
        <bean:sAppId>3467</bean:sAppId>
        <bean:sCSV>csv_1357</bean:sCSV>

<bean:sContent>JVBERi0xLjQKJeLjz9MKMyAwIG9iago8PC9MZW5ndGggMjQzL0ZpbHRlci9GbGF0ZURlY29kZT4+c3RyZWFtCnicrZFNT8MwDIbv+RU+jktJss7tOLFV2wEJJCA7IQ5h88qmdtnccNi/p5nEl4QC0iJLthT79ePoPYipEUOEUiKYlZgZcS803IRXBbKPkMtcg2nF5VyBkmDWYnBhtmH2a0TCsv0u0hKK8SiT6kOoMQjlqc21GFRu5+3SwwPtHfuwT0L9y86n576uTjfFaUPMRvnnmfonbdER39mWrhKAVAxUOWZy51NwHKM87hJ8BIsowjORPxejdP6HM6pUiCiLaQpS1JravvCGmuujfXUuoy4BMO7SonuzvHEw2ZMKSacgRk2rbNMQ3Nqj4773L9w7EmnVuAplbmRzdH
```

JlYW0KZW5kb2JqCjEgMCBvYmoKPDwvR3JvdXA8PC9UeXBlL0dyb3VwL0NTL0RldmljZVJHQi9TL1RyYW5zcGFyZW5jeT4+L1BhcmVudCA0IDAgUi9Db250ZW50cyAzIDAgUi9UeXBlL1BhZ2UvUmVzb3VyY2VzPDwvUHJvY1NldCBbL1BERiAvVGV4dCAvSW1hZ2VCIC9JbWFnZUMgL0ltYWdlSV0vQ29sb3JTcGFjZTw8L0NTL0RldmljZVJHQj4+L0ZvbnQ8PC9GMSAyIDAgUj4+Pj4vTWVkaWFCb3hbMCAwIDU5NSA4NDJdPj4KZW5kb2JqCjUgMCBvYmoKWzEgMCBSL1hZWiAwIDg1MiAwXQplbmRvYmoKMiAwIG9iago8PC9CYXNlRm9udC9IZWx2ZXRpY2EvVHlwZS9Gb250L0VuY29kaW5nL1dpbkFuc2lFbmNvZGluZy9TdWJ0eXBlL1R5cGUxPj4KZW5kb2JqCjQgMCBvYmoKPDwvRGVzdHMgNiAwIFI+PgplbmRvYmoKOCAwIG9iago8PC9OYW1lcyA3IDAgUi9UeXBlL0NhdGFsb2cvVmlld2VyUHJlZmVyZW5jZXM8PC9Db250ZW50cyAzIDAgUi9kgMCBvYmoKPDwvQ3JlYXRvcihKYXNwZXJSZXBvcnRzIFwoYXV0ZW50aWNhX3JlcG9ydF9mb3JtX2Byb2R1Y2VyKGlUZXh0IDIuMS43IGJ5IDFUM1hUKS9Nb2REYXRlKEQ6MjAxNDA2MTAxNDUzMzYrMDInMDAnKS9DcmVhdGlvbkRhdGUoRDoyMDE0MDYxMDE0NTMzNiswMicwMCcpPj4KZW5kb2JqCnhyZWYKMCAxMAowMDAwMDAwMDAwIDY1NTM1IGYgCjAwMDAwMDAzMjUgMDAwMDAgbiAKMDAwMDAwMDU5NCAwMDAwMCBuIAowMDAwMDE1IDAwMDAwIG4gCjAwMDAwMDA2ODIgMDAwMDAgbiAKMDAwMDAwMDU1OSAwMDAwMCBuIAowMDAwMDAwNzQ1IDAwMDAwIG4gCjAwMDAwMDA3OTkgMDAwMDAgbiAKMDAwMDAwMDgzMSAwMDAwMCBuIAowMDAwMDAwOTM0IDAwMDAwIG4gCnRyYWlsZXIKPDwvUm9vdCA4IDAgUi9JRCBbPDBiYTlmYzVmZTM3MGUwYjg5MDRhYjQ4NjVkOTM1MjE4PjxiOTZkZDRiYjIxNTU2YTAwMjU2N2I1M2E2NDUxNmNkYT5dL0luZm8gOSAwIFIvU2l6ZSAxMD4+CnN0YXJ0eHJlZgoxMTA1CiUlRU9G</bean:sContent>

<bean:sExt>.pdf</bean:sExt>

<bean:sHash>1645700394233</bean:sHash>

<bean:sName>autentica_report_form.pdf</bean:sName>

<bean:sUserName>99999999R</bean:sUserName>

</fnc:fncFileBean>

</fnc:getFncSign>

</soapenv:Body>

</soapenv:Envelope>

Ejemplo de respuesta (cortado por longitud)

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

<soapenv:Body>

<getFncSignResponse xmlns="http://fnc.ws.babel.es">

<getFncSignReturn>

<csv>csv_1357</csv>

<fileContent>JVBERi0xLjYKJfbk/N8KMyAwIG9iago8PAovTGVuZ3RoIDI0MwovRmlsdGVyIC9GbGF0ZURlY29kZQo+PgpzdHJlYW0NCnicrZFNT8MwDIbv+RU+jktJss7tOLFV2wEJJCA7IQ5h88qmdtnccNi/p5nEl4QC0iJLthT79ePoPYipEUOEUiKYlZgZcS803

[……]

+XQovSW5mbyA5IDAgUgovUHJldiAxMDg3Ci9UeXBlIC9YUmVmCi9TaXplIDE4Ci9JbmRleCBbMCAxIDggMSAxMCAxIDEyIDEgMTQgM10KL1cgWzEgMiAwXQovRmlsdGVyIC9GbGF0ZURlY29kZQo+PgpzdHJlYW0NCnicY2BgYGRNZGQTY2R9zMi2ljFliTHICQAZMwMsDQplbmRzdHJlYW0KZW5kb2JqCnN0YXJ0eHJlZgoyMTE2NAolJUVPRgo=</fileContent>

<fileJustif>JVBERi0xLjYKJfbk/

EPQ+HvH+E6tPgf6i0we8miB3sDQtaf4XOMJQWNaEDvhKUVTl5aV3PtK5Y3JF0TlnToCN/Y2tr67bBpLIky+t+

[……]

gMjUKL0luZGV4IFswIDIgMTcgMSAxOSAxIDIxIDNdCi9XIFsxIDIgMF0KL0ZpbHRlciAvRmxhdGVEZWNvZGUKPj4Kc3RyZWFtDQp4nGNgYGCMLWWM/cEYe5Qxrp9xFQvjKhUANuMFuQ0KZW5kc3RyZWFtCmVuZG9iagpzdGFydHhyZWYKNDM2NjIKJSVFT0YK</fileJustif>

<message xsi:nil="true"/>

```
        </getFncSignReturn>
      </getFncSignResponse>
   </soapenv:Body>
</soapenv:Envelope>
```

4.3.4. Ejemplo de invocación proyecto web

A continuación se detalla un ejemplo de invocación de los servicios web desde una aplicación:

- Se generaran las clases cliente a partir del fichero FncWS.wsdl o de una URL como la siguiente: (https://se-autentica.redsara.es/FncWS/services/FncWS?wsdl el entorno de servicios estables y https://autentica.redsara.es/FncWS/services/FncWS?wsdl en el entorno de producción).

  - Se añadirán los siguientes ficheros para WS-Security (ubicados todos en la carpeta classes)
    - **fncClient_deploy**.wsdd (en negrita se indican valores de ejemplo a configurar, comenzando por el propio nombre del fichero).

```
<deployment xmlns="http://xml.apache.org/axis/wsdd/"
xmlns:java="http://xml.apache.org/axis/wsdd/providers/java">
        <transport name="http" pivot="java:org.apache.axis.transport.http.HTTPSender"/>
        <globalConfiguration>
                <requestFlow>
                        <handler type="java:org.apache.ws.axis.security.WSDoAllSender">
                                <parameter name="action" value="Signature"/>
                                <parameter name="actor" value="fncSig"/>
                                <parameter name="mustUnderstand" value="1"/>
                                <parameter name="user" value="sello entidad sgad pruebas"/>
                                <parameter name="passwordCallbackClass"
value="es.sag.fnc.callback.FncPWCallback"/>
                                <parameter name="signaturePropFile"    value="fncCrt-crypto.properties"
/>
                                <parameter name="signatureKeyIdentifier" value="DirectReference" />
                        </handler>
                </requestFlow >
        </globalConfiguration>
</deployment>
```

    - **fncCrt-crypto.properties** (en negrita se indican valores de ejemplo a configurar, comenzando por el propio nombre del fichero).

```
org.apache.ws.security.crypto.provider          = org.apache.ws.security.components.crypto.Merlin
org.apache.ws.security.crypto.merlin.keystore.type     = PKCS12
org.apache.ws.security.crypto.merlin.keystore.password     = [….]
org.apache.ws.security.crypto.merlin.file          = sello_dtic.ks
org.apache.ws.security.crypto.merlin.keystore.alias          = sello entidad sgad pruebas
```

- Será necesario desarrollar una clase java que herede de CallbackHandler (PWCallback): se muestra ejemplo con valores en negrita a configurar.

```
package es.sag.fnc.callback;

import javax.security.auth.callback.Callback;
import javax.security.auth.callback.CallbackHandler;
import javax.security.auth.callback.UnsupportedCallbackException;

import org.apache.ws.security.WSPasswordCallback;

import java.io.IOException;

public class FncPWCallback implements CallbackHandler {

   public void handle(Callback[] callbacks)
         throws IOException, UnsupportedCallbackException {

      for (int i = 0; i < callbacks.length; i++) {
         if (callbacks[i] instanceof WSPasswordCallback) {
            WSPasswordCallback pc = (WSPasswordCallback) callbacks[i];

            if (pc.getIdentifer().equals("fnc")) {
               pc.setPassword("fncKeyWS");
            } else {
               throw new UnsupportedCallbackException(callbacks[i], "Usuario
desconocido("+pc.getIdentifer()+")");
            }
         } else {
            throw new UnsupportedCallbackException(callbacks[i],
                  "Manejador desconocido");
         }
      }
   }
}
```

5. DESCARGA DE DOCUMENTO FIRMADO

Dentro del catálogo de servicios web de FNC se encuentra el servicio web "getSignedDocument", por medio del cual se podrá descargar el documento firmado. Para ello, se deberán indicar los siguientes parámetros:

- WebUser: objeto que indica el identificador de la aplicación que está invocando el servicio web "getSignedDocument".
- String sCSV: String con el CSV del documento del cual se quieren descargar del documento firmado.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:fnc="http://fnc.ws.babel.es" xmlns:obj="http://obj.fnc.ws.babel.es">
  <soapenv:Header/>
  <soapenv:Body>
    <fnc:getSignedDocument>
      <fnc:webUser>
        <obj:webName>?</obj:webName>
      </fnc:webUser>
      <fnc:sSCV>?</fnc:sSCV>
    </fnc:getSignedDocument>
  </soapenv:Body>
</soapenv:Envelope>
```

Como requisitos para la utilización de este servicio web se indican los siguientes:

- Se está utilizando el certificado configurado en la ficha de la aplicación para la utilización de los servicios web.
- El id de aplicación que se indica en el objeto WebUser es el mismo que la aplicación con la que se firmó el documento.

El servicio devuelve un objeto de tipo "SignedDocument" que contiene dos elementos:

- fileContent: String en base64 que contiene el fichero firmado o null en el caso de que se haya producido alguna incidencia en la petición.
- message. String con el CSV en el caso de que el documento haya sido devuelto o un mensaje indicando el error producido.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <getSignedDocumentResponse xmlns="http://fnc.ws.babel.es">
```

```
        <getSignedDocumentReturn>
          <fileContent>JVBERi0xLjYKJfb[……]o=</fileContent>
           <message>AUT_1663165368106</message>
        </getSignedDocumentReturn>
      </getSignedDocumentResponse>
   </soapenv:Body>
</soapenv:Envelope>
```