Consejería de Hacienda y Administración Pública

Módulo de firmas XML

Sevilla, noviembre de 2010

Hoja de control		
Fecha	**Autor**	**Descripción**
03/11/2010	RPV	Adaptación módulo de firmas XML MPR

1 Introducción 5

2 Objetivos 5

3 Formatos de firma soportados 6

3.1 Matriz de formatos soportados por el módulo 6

3.2 Uso de los parámetros de funcionamiento 7

3.3 Parámetros de funcionamiento 8

3.4 XML Digital Signature (XMLDSig) 10

3.4.1 XMLDSig Detached 10

3.4.2 XMLDSig Enveloping 13

3.4.3 XMLDSig Enveloped 14

3.4.4 Co-Firmas en XMLDSig 15

3.4.4.1 Cofirmas cruzadas entre XMLDSig y XAdES 15

3.4.5 Contrafirmas en XMLDSig 16

3.5 XML Advanced Digital Signature (XAdES) 17

3.6 Open Document Format (ODF) 18

3.7 Office Open XML (OOXML) 18

4 Información sobre el módulo 19

4.1 Compatibilidad 19

4.2 Firmas de contenido binario en XMLDSig y XAdES 20

4.2.1 Deshabilitación de las transformaciones Base64 20

4.3 Firma de hojas de estilo en XMLDSig y XAdES 21

4.3.1 Notas importantes sobre la firma de hojas de estilo XML 21

4.3.1.1 Desactivación y activación de firma de las hojas de estilo 21

4.3.1.2 Hojas de estilo anidadas 22

4.3.2 Transformaciones XML 22

4.4 Incidencias conocidas 24
4.4.1 Mensajes extraños en consola 24
4.5 Diferencias en las firmas generadas respecto a la versión anterior del cliente y sus módulos 24
5 Ejemplos de Estructuras XML generadas por el módulo . 25
5.1 XMLDSig 25
5.1.1 Detached Interno Explícito / Implícito 25
5.1.2 Enveloping Explícito / Implícito 27
5.1.3 Enveloped 28
5.1.4 Co-Firmas 29
5.1.5 Contrafirmas 31
5.2 XAdES 34
5.2.1 Detached Interno Explícito / Implícito 34
5.2.2 Enveloping Explícito / Implícito 36
5.2.3 Enveloped 38
5.2.4 Co-Firmas 40
5.2.5 Contrafirmas 44
5.2.6 Distintas versiones de XAdES 49
6 Información de utilidad 50
7 Glosario de términos 50
8 Información de contacto 53

1 Introducción

El Cliente de Firma es una herramienta de Firma Electrónica que funciona en forma de Applet de Java integrado en una página Web mediante JavaScript.

El Cliente hace uso de los certificados digitales X.509 y de las claves privadas asociadas a los mismos que estén instalados en el repositorio o almacén de claves y certificados (*keystore*) del navegador web (*Internet Explorer, Mozilla, Firefox*) o el sistema operativo así como de los que estén en dispositivos (tarjetas inteligentes, dispositivos *USB*) configurados en el mismo (el caso de los DNI-e).

El Cliente de Firma, como su nombre indica, es una aplicación que se ejecuta en cliente (en el ordenador del usuario, no en el servidor Web). Esto es así para evitar que la clave privada asociada a un certificado tenga que "salir" del contenedor del usuario (tarjeta, dispositivo USB o navegador) ubicado en su PC. De hecho, nunca llega a salir del navegador, el Cliente le envía los datos a firmar y éste los devuelve firmados.

El Cliente de Firma contiene las interfaces y componentes web necesarios para la realización de los siguientes procesos (además de otros auxiliares como cálculos de hash, lectura de ficheros, etc…):

- Firma de formularios Web.
- Firma de datos y ficheros.
- Multifirma masiva de datos y ficheros.
- Cofirma (CoSignature) → Multifirma al mismo nivel.
- Contrafirma (CounterSignature) → Multifirma en cascada.

Como complemento al cliente de firma, se encuentra un cliente de cifrado que nos permite realizar las funciones de encriptación y desencriptación de datos atendiendo a diferentes algoritmos y configuraciones. Además permite la generación de sobres digitales.

2 Objetivos

El objetivo del presente documento es reflejar las capacidades del Cliente @firma en cuanto a firmas electrónicas basadas en formatos XML (*Extensible Markup Language*), las peculiaridades de este relativas a estos formatos, sus opciones de configuración y su compatibilidad.

3 Formatos de firma soportados

3.1 Matriz de formatos soportados por el módulo

		Detached				Enveloping		Enveloped
		Interna		Externa				
		Implícito	Explícito	Implícito	Explícito	Implícito	Explícito	Implícito
XMLDS	Firma							
	Cofirma							
	Contrafirma							
XAdES	Firma							
	Cofirma							
	Contrafirma							
SOAP	Firma							

Leyenda:	
	Soportado y acorde a estándar
	Soportado
	No soportado
	No contemplado en el estándar

Dentro de la normativa XAdES, la versión actual del módulo soporta únicamente las variante básica, conocida como XAdES BES y la acorde a política de firma, EPES, no soportándose ninguna otra (X, XL, etc.):

	BES	EPES	T	C	X	XL	A
Variante							

Adicionalmente, el módulo es capaz de hacer firmas XML según la normativa Open Document Format (ODF) acorde con la implementación de OpenOffice.org 3.0 y superiores.

Debemos recordar que los documentos ODF están basados en XML, y sus firmas digitales son una variante de XMLDSig.

La matriz de compatibilidad (siguiendo la misma leyenda que la matriz anterior), es la siguiente:

	OpenOffice.org		
	2.4	3.0	3.1
Firmas			
Cofirma			
Contrafirma			

En el caso de las "cofirmas" sobre documentos ODF, estas deben tratarse como la aplicación de varias firmas al mismo documento a un mismo nivel jerárquico, en lo que también puede llamarse "multifirma" de un documento.

Otra variante de firmas XML soportada por el Cliente es OOXML, formato de documentos ofimáticos de Microsoft Office 2007, Office 2008 for Mac OS X y Office 2010 (x86 y x64).

La matriz de compatibilidad (siguiendo la misma leyenda que la matriz anterior), es la siguiente:

	Microsoft Office		
	2007	2008	2010
Firmas			
Cofirma			
Contrafirma			

En el caso de las "cofirmas" sobre documentos OOXML, estas deben tratarse como la aplicación de varias firmas al mismo documento a un mismo nivel jerárquico, en lo que también puede llamarse "multifirma" de un documento.

3.2 Uso de los parámetros de funcionamiento

Los formatos de firma XML (XAdES, XMLDSig, ODF y OOXML) está únicamente disponible en las construcciones Media y Completa del cliente de firma.

Para realizar una firma en este formato sólo es necesario su configuración como formato de firma mediante el método del Applet cliente:

```
setSignatureFormat(String format)
```

Puede llamarse a esta función desde las páginas Web que integren el cliente por medio de la sentencia JavaScript:

```
clienteFirma.setSignatureFormat(String format)
```

Esto configurará el cliente de firma para realizar alguna de las firmas XML (consultar el apartado "**Parámetros de funcionamiento**", para conocer las referencias concretas para cada uno de los formatos), siendo necesario también completar la configuración del cliente que se considere necesaria (algoritmo de firma, modo, tratamiento de errores,…), establecer los datos que se desean firmar y ordenar el proceso de firma.

Esta y el resto de funciones del cliente están documentadas en el JavaDoc del Applet y el manual del integrador del cliente @firma. Remítase a estos documentos para más información.

3.3 Parámetros de funcionamiento

Cadenas de identificación de formato (varias alternativas insensibles a mayúsculas y minúsculas por cada formato, por flexibilidad de uso):

- XAdES Detached Interna

"XAdES Detached" / "XAdES_Detached" / "XADES" / "XADES-BES"

- XAdES Enveloped

"XAdES Enveloped" / "XAdES_Enveloped"

- XAdES Enveloping

"XAdES Enveloping" / "XAdES_Enveloping"

- XMLDSig Detached

"XMLdSig Detached" / "XMLdSign Detached" / "XMLdSig_Detached" / "XMLdSign_Detached" / "XMLDSIG" / "XMLDSIGN"

- XMLDSig Enveloped

"XMLdSig Enveloped" / "XMLdSign Enveloped" / "XMLdSig_Enveloped" / "XMLdSig_Enveloped"

- XMLDSig Enveloping

"XMLdSig Enveloping" / "XMLdSign Enveloping" / "XMLdSig_Enveloping" / "XMLdSign_Enveloping"

- ODF

"ODF" / "ODT" / "ODP" / "ODS" / "OpenOffice" / "OpenOffice.org"

- OOXML

"OOXML" / "OOXML (Office Open XML)" / "DOCX" / "PPTX" / "XLSX"

Cadenas de identificación del modo de firma (insensibles a mayúsculas y minúsculas):

- Explícita

"Explicit"

- Implícita

"Implicit"

Ficheros de entrada

- XMLDSig y XAdES, Enveloping y Detached, Implícitas y Explícitas

Binarios (*.*) y XML (*.xml)

- XMLDSig y XAdES, Enveloped

XML (*.xml)

- ODF

ODF (*.odt, *.ods, *.odp)

- OOXML

OOXML (*.docx, *.pptx, *.xlsx)

Ficheros de salida

- XAdES

Fichero de firma avanzada (*.xsig)

- XMLDSig

Fichero de firma (*.xsig)

- ODF

ODF (*.odt, *.ods, *.odp)

- OOXML

OOXML(*.docx, *.pptx, *.xlsx)

Notas importantes

- El algoritmo de huella digital **MD2 no está soportado** en ninguno de los formatos y variantes de firma aportados por este módulo, por motivos de obsolescencia. Esta es una limitación impuesta por los estándares y normativas afectadas.
- En las operaciones de firma, las algoritmos de huella digital **SHA-256, SHA-384 y SHA-512 no están soportados** en ningún sistema basado en CAPI (Windows + Internet Explorer) ni, aun sobre sistemas diferentes a CAPI, en los entornos de ejecución de Java anteriores a la **versión 1.6u18.**
- El algoritmo de huella digital **SHA-224 no está soportado**, y no se soportará en un futuro.
- No se soportan firmas XML **sin realizar huellas digitales**, por no estar contemplada esta posibilidad en ninguna normativa ni estándar.
- **No se soporta la firma en más de una fase en ningún formato o variante de firma realizado por este módulo.**
- Ciertos entornos de ejecución de Java (Java 1.6 anteriores a la versión 1.6.0_10) están afectados por un error (http://bugs.sun.com/bugdatabase/view_bug.do?bug_id=6696582) que impide la realización de firmas de documentos ODF. **Es necesario un Java 6 en versión**

superior a la 1.6.0_10 para realizar firmas ODF. Este error no se da en Java 5, por lo que sus usuarios no necesitan actualizar (aunque siempre se recomienda la actualización a Java 1.6.0_18).

- El módulo es incompatible con las versiones **preliminares (beta / alpha)** de **Java 7**, por lo que no se recomienda su uso hasta que no esté oficialmente soportada en una versión final.

3.4 XML Digital Signature (XMLDSig)

3.4.1 XMLDSig Detached

La firma XML en modo "Detached" permite tener una firma de forma separada e independiente del contenido firmado, pudiendo relacionar firma con contenido firmado mediante una referencia de tipo URI. Este tipo de firmas es útil cuando no se puede modificar el contenido original pero se desea constatar su autenticidad, procedencia, etc.

Un uso común de este formato es en la descarga de ficheros, pudiendo poner a disposición del internauta, junto al contenido a descargar, un pequeño fichero de firma para verificar la integridad del primero.

Un ejemplo de este tipo de firmas sería la siguiente estructura (resumida) XML:

```
<?xml version="1.0" encoding="UTF-8"?>
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <ds:SignedInfo>
    <ds:CanonicalizationMethod
         Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
    <ds:SignatureMethod
         Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
    <ds:Reference URI="http://www.mpr.es/contenido">
      <ds:DigestMethod
          Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
      <ds:DigestValue/>
    </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue/>
</ds:Signature>
```

En este ejemplo, los datos firmados se encuentran en un servidor Web accesible públicamente: `http://www.mpr.es/contenido`, y se referencia como tal, conformando lo que se denomina "Externally Detached" o "Detached Externa".

Cuando se desea firmar un contenido con un formato "Detached", pero se quiere eliminar la dependencia de la disponibilidad externa del contenido firmado, es posible crear una estructura XML que contenga los propios contenidos y la firma, pero cada uno en una subestructura independiente, manteniendo así el concepto de "Detached" (firma y contenido firmado no se interrelacionan directamente).

Un ejemplo de estructura XML sería:

```
<?xml version="1.0" encoding="UTF-8"?>
<internally-detached>
  <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:SignedInfo>
      <ds:CanonicalizationMethod
         Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
      <ds:SignatureMethod
         Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
      <ds:Reference URI="#data">
        <ds:DigestMethod
          Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
        <ds:DigestValue/>
      </ds:Reference>
    </ds:SignedInfo>
    <ds:SignatureValue/>
  </ds:Signature>

  <document Id="data">
     <title>title</title>
     <author>writer</author>
     <date>today</date>
     <content>
        <para>First paragraph</para>
        <para>Second paragraph</para>
     </content>
  </document>
</internally-detached>
```

En este caso, la estructura "internally-detached" contiene dos subestructuras, la firma ("Signature") y el propio contenido firmado ("document"). La forma de relacionar ambos es, como ocurría en el primer ejemplo, con una URI, solo que en este caso es interna al documento XML, referenciando el identificador de la subestructura del contenido firmado ("data").

A esta variante de firma "Detached" se la conoce como "Internally Detached", o "Detached Interna".

Para unificar las superestructuras creadas dentro de un formato "Detached Interno", el Cliente @firma construye siempre el siguiente esqueleto XML:

```
<CONTENT Id="id" Encoding="codificacion" MimeType="MimeType" Algorithm="…">
  <! – CONTENIDO FIRMADO -->
</CONTENT>
```

Es decir, el contenido a firmar, ya sea XML o no-XML, se encapsula dentro de una etiqueta XML llamada CONTENT, en la que se indica la codificación del contenido (UTF-8, Base64, etc.), el tipo de contenido (imagen JPEG, texto, XML, etc.) y el algoritmo utilizado para calcular la huella digital de este (por ejemplo, SHA-1).

Como la superestructura es XML, si el contenido también es XML la inserción es directa (como en el primer ejemplo de "Detached Interna", pero si no es XML se codifica en Base64 antes de insertarse, resultando una estructura con una forma similar a la siguiente:

```
<CONTENT Id="id" Encoding="Base64" MimeType="application/octect-stream"
Algorithm="…">
  SFGJKASGFJKASEGUYFGEYGEYRGADFJKASGDFSUYFGAUYEGWEYJGDFYKGYKGWJKEGYFWYJ
</CONTENT>
```

La larga cadena de caracteres sería una codificación Base64 del original interpretado en su forma binaria pura.

Como la variante "Detached Interna" almacena dentro de la superestructura XML el contenido firmado, la firma de un contenido de gran tamaño generaría igualmente un fichero de firma de gran tamaño, aspecto claramente no deseable (especialmente en un formato "Detached"), por lo que este módulo introduce una sub-variante llamada "Detached Interna Explícita" (quedando la anteriormente descrita como "Detached Interna Implícita". En esta sub-variante, se sustituyen los datos firmados (ya sean binarios o XML) dentro de la superestructura XML que los contenía tanto a ellos como a la firma por su huella digital calculada mediante el algoritmo SHA-1, y codificada en Base64.

Es importante reseñar aquí que **la sub-variante Detached Interna Explícita no está respaldada directamente por el estándar XML Digital Signature** (aunque su estructura interna cumple con el formato).

Debido a que las firmas Detached Internas, tanto explícita como implícita, comparten la misma estructura y no es posible distinguirlas, **el cliente marcará las firmas explícitas con el MimeType no estándar "hash/sha1", independientemente de que se especifique el MimeType de los datos**. En caso de generar la firma explícita indicando directamente al Cliente @firma el hash que debe firmar en lugar de los datos, se firmará este hash y se establecerá como MimeType el valor

"hash/*algoritmo*" en donde '*algoritmo*' será el algoritmo de hash utilizado en la firma. Por ejemplo, en una firma "SHA256withRSA" se deberá indicar un hash de tipo SHA256.

La importancia de recalcar este punto reside en que un "validador" de firmas XMLDSig estándar dará siempre por buena la firma Detached Explícita, pero no la validará contra el contenido original, por lo que debemos asegurarnos o que se soporta ésta sub-variante o que la validación se realiza contra la huella digital del contenido firmado y luego se compara esta huella digital con una obtenida directamente de este último.

3.4.2 XMLDSig Enveloping

Otra variante de firma es la "Enveloping", en la que la estructura XML de firma es la única en el documento de firma, y esta contiene internamente el contenido firmado (en un nodo propio).
Un posible ejemplo de este tipo de firma podría ser:

```
<?xml version="1.0" encoding="UTF-8"?>
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <ds:SignedInfo>
    <ds:CanonicalizationMethod
         Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
    <ds:SignatureMethod
         Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
    <ds:Reference URI="#obj">
      <ds:DigestMethod
          Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
      <ds:DigestValue/>
    </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue/>
  <ds:Object Id="obj">SFGJKASGFJKASEGUYFGEYGEYRGADFJKASGDFSUYFG=</ds:Object>
</ds:Signature>
```

En este caso, los datos firmados se encuentran en el nodo "Object" referenciados internamente al XML mediante el identificador "obj".

Al igual que ocurría con el formato "Detached", si los datos no son XML, no es posible insertarlos directamente dentro de una estructura XML, por lo que se codifican previamente en Base64.

Al contener el resultado de la firma los datos firmados empotrados internamente, esta formato sufre el mismo problema que la variante "Detached Interna" (Implícita), por lo que también se ha optado por incluir una sub-variante denominada "Enveloping Explícita" (definiendo la "Enveloping" como "Enveloping Implícita"), en la que de nuevo se sustituyen los datos originalmente firmados por su huella digital calculada mediante el algoritmo SHA-1.

Es importante reseñar aquí que **la sub-variante Enveloping Explícita no está respaldada directamente por el estándar XML Digital Signature** (aunque su estructura interna cumple con el formato).

La importancia de recalcar este punto reside en que un "validador" de firmas XMLDSig estándar dará siempre por buena la firma Enveloping Explícita, pero no la validará contra el contenido original, por lo que debemos asegurarnos o que se soporta esta sub-variante o que la validación se realiza contra la huella digital del contenido firmado y luego se compara esta huella digital con una obtenida directamente de este último.

Debido a que las firmas Enveloping explícita e implícita comparten la misma estructura y no es posible distinguirlas, **el cliente marcará las firmas explícitas con el MimeType no estándar "hash/sha1", independientemente de que se especifique el MimeType de los datos**. En caso de generar la firma explícita indicando directamente al Cliente @firma el hash que debe firmar en lugar de los datos, se firmará este hash y se establecerá como MimeType el valor "hash/*algoritmo*" en donde '*algoritmo*' será el algoritmo de hash utilizado en la firma. Por ejemplo, en una firma "SHA256withRSA" se deberá indicar un hash de tipo SHA256.

3.4.3 XMLDSig Enveloped

Este formato de firma XMLDSig está pensado para que un contenido XML pueda auto-contener su propia firma digital, insertándola en un nodo propio interno, por lo que, al contrario que en los formatos anteriores, no es posible firmar contenido que no sea XML.

Un ejemplo simple del resultado de una firma "Enveloped" podría ser el siguiente:

```
<!DOCTYPE Envelope [
  <!ENTITY ds "http://www.w3.org/2000/09/xmldsig#">
  <!ENTITY c14n "http://www.w3.org/TR/2001/REC-xml-c14n-20010315">
  <!ENTITY enveloped "http://www.w3.org/2000/09/xmldsig#enveloped-signature">
  <!ENTITY xslt "http://www.w3.org/TR/1999/REC-xslt-19991116">
  <!ENTITY digest "http://www.w3.org/2000/09/xmldsig#sha1">
]>
<Letter>
   <Return-address>address</Return-address>
   <To>You</To>
   <Message>msg body</Message>
   <From>
      <ds:Signature xmlns:ds="&ds;">
         <ds:SignedInfo>
            <ds:CanonicalizationMethod Algorithm=
```

```
                "http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
            <ds:SignatureMethod Algorithm=
                "http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
            <ds:Reference URI="">
               <ds:Transforms>
                  <ds:Transform Algorithm="&enveloped;">
                  </ds:Transform>
               </ds:Transforms>
               <ds:DigestMethod Algorithm="&digest;"/>
               <ds:DigestValue></ds:DigestValue>
            </ds:Reference>
         </ds:SignedInfo>
         <ds:SignatureValue/>
      </ds:Signature>
   </From>
   <Attach>attachement</Attach>
</Letter>
```

En este caso, el documento original ("Letter"), contiene internamente la estructura de firma digital ("Signature").

Una peculiaridad de la estructura generada es que esta referenciada mediante una URI vacía ("URI=""), lo cual indica que la firma aplica a la totalidad del documento original.

Aunque en este caso el módulo no ha definido sub-variantes, para mantener una coherencia en la nomenclatura, ha denominado a este formato "Enveloped Implícita".

3.4.4 Co-Firmas en XMLDSig

Cuando un mismo documento es firmado (en una misma jerarquía de firma) por varios firmantes, de produce una operación de co-firma.

A nivel interno, una co-firma no es más que una sucesión de objetos o estructuras XML de tipo "Signature" (como los que figuran en los ejemplos anteriores) que se refieren a los mismos datos. Las co-firmas XMLDSig pueden aplicarse a todos los formatos y variantes del estándar.

El módulo de firmas XML del Cliente @firma permite co-firmas en todas las variantes y sub-variantes soportadas de XMLDSig.

3.4.4.1 Cofirmas cruzadas entre XMLDSig y XAdES

Las cofirmas de un documento dan como resultado dos firmas sobre este mismo documento que se encuentran a un mismo nivel, es decir, que ninguna envuelve a la otra ni una prevalece sobre la otra.

A nivel de formato interno, esto quiere decir que cuando cofirmamos un documento ya firmado previamente, esta firma previa no se modifica. Si tenemos en cuenta que XAdES es en realidad un subconjunto de XMLDSig, el resultado de una cofirma XAdES sobre un documento firmado previamente con XMLDSig (o viceversa), son dos firmas independientes, una en XAdES y otra en XMLDSig. Dado que todas las firmas XAdES son XMLDSig pero no todas las firmas XMLDSig son XAdES, el resultado global de la firma se adecúa al estándar más amplio, XMLDSig en este caso.

3.4.5 Contrafirmas en XMLDSig

Una contrafirma es un tipo de multi-firma (varios firmantes en un único documento), en el que las firmas de los firmantes no están a un mismo nivel, sino que siguen una jerarquía de firmas, donde un firmante asume la firma de otro firmante previo, formándose en el proceso un árbol de firmas (o firmantes).

Un ejemplo de uso de este tipo de multi-firmas podríamos encontrarlo en una revisión en cadena de un documento:

1. El autor del documento firma el original.
    1. El documento pasa al gerente de ventas, el cual firma sobre la firma del autor
        1. El documento pasa al director de ventas, el cual asume el buen trabajo de su gerente y firma sobre su firma.
    2. El documento pasa, también desde el propio autor, al gerente de calidad, el cual firma sobre la firma del autor.
        1. El documento pasa al director de calidad, el cual asume el buen trabajo de su gerente y firma sobre su firma.
2. El documento pasa al director general, el cual firma las firmas de sus directores, asumiendo igualmente su trabajo.

Adicionalmente, es posible combinar co-firmas con contrafirmas, pudiendo tener así árboles de firma complejos, incluso con varios nodos raíz.

La especificación del estándar XMLDSig no define la forma de contrafirmar un documento, aunque deja abierta la posibilidad de realizar contrafirmas. No obstante, la especificación XAdES [ETSI TS 101 903 V1.3.2 (2006-03)] especifica un formato de contrafirma retro-compatible con el formato XMLDSig mediante referencias de tipo "CountersignedSignature" (http://uri.etsi.org/01903#CountersignedSignature).

El módulo de firmas XML del cliente @firma implementa las contrafirmas según este método, pero es importante resaltar que, dado que la normativa XMLDSig no adopta este formato de contrafirmas directamente (sino que se define en la normativa XAdES), **las contrafirmas XMLDSig creadas mediante el cliente @firma pueden no reconocerse en algunas plataformas de validación de firmas**.

3.5 XML Advanced Digital Signature (XAdES)

Las firmas XAdES son una evolución de las firmas XMLDSig a la que añaden ciertas extensiones y en la que se concretan con más definición algunas operaciones, como las contrafirmas.

El formato XAdES admite múltiples variantes, con distintas aplicaciones (desde sellos de tiempo hasta archivo longevo). La actual versión del módulo de firmas XML del cliente @firma soporta únicamente la versiones básica (BES), de aplicación general no especializada y la acorde a política de firma (EPES).

Variantes XAdES						
BES	EPES	T	C	X	XL	A

La tabla anterior muestra las variantes de XAdES existentes, estando marcadas únicamente las básica (BES) y acorde a política de firma (EPES) en verde, como indicativo de soporte por el módulo.

Dado que XAdES es una extensión retro-compatible con XMLDSig, los formatos soportados son los mismos que en este segundo, aunque encontrando las siguientes diferencias y aspectos a recalcar:

- Las contrafirmas se realizan mediante un atributo especialmente especificado por XAdES ("CounterSignature"), por lo que, al contrario que en XMLDSig, sí están plenamente respaldadas por el estándar.
- Al igual que ocurría con XMLDSig, todas las sub-variantes explícitas **no están directamente respaldadas por el estándar**.

Adicionalmente, es importante comentar que los atributos específicos XAdES implementados por el módulo de firmas XML son (además de los relativos a las políticas de firma):

- SigningTime
- SigningCerticate
- IssuerSerial

- SignedDataObjectProperties

3.6 Open Document Format (ODF)

ODF, la normativa de documentos ofimáticos implementada por OpenOffice.org, implementa para sus firmas digitales una variante específica de XMLDSig.

El módulo es capaz de realizar las firmas según esta variante, resultando en documentos firmados reconocidos por los productos OpenOffice.org 3.0 y 3.1, no previéndose que existan problemas de compatibilidad en futuras versiones superiores a la 3.1.

Por problemas de evolución del propio formato ODF, las firmas generadas no son válidas en el formato ODF generado por OpenOffice.org 2.4.

La compatibilidad en cuanto a tipo de documento es la siguiente:

OpenOffice.org 3.0 / 3.1		
Impress (Presentaciones)	Calc (Hojas de Cálculo)	Writer (Textos)

Se soportan documentos Impress, Calc y Writer.

Detalles a tener en cuenta del formato de firma ODF son:

- No contempla la operación de contrafirma
- Las firmas son siempre implícitas, por lo que no se atenderá a la configuración de modo del cliente.

El único algoritmo de firma soportado por el formato es SHA1withRSA, por lo que no se atenderá a la configuración del cliente.

3.7 Office Open XML (OOXML)

OOXML, la normativa de documentos ofimáticos implementada por Microsoft Office, implementa para sus firmas digitales una variante específica de XMLDSig.

El módulo es capaz de realizar las firmas según esta variante, resultando en documentos firmados reconocidos por los productos Office 2007, Office 2008 for Mac OS X y Office 2010, no previéndose que existan problemas de compatibilidad en futuras versiones.

La compatibilidad en cuanto a tipo de documento es la siguiente:

Office 2007/2008/2010		
PowerPoint (Presentaciones)	Excel (Hojas de Cálculo)	Word (Textos)

Se soportan documentos Word, Excel y PowerPoint.

Detalles a tener en cuenta del formato de firma OOXML son:

- No contempla la operación de contrafirma.
- Las firmas son siempre implícitas, por lo que no se atenderá a la configuración de modo del cliente.

ADVERTENCIA: La herramienta de validación VALIDe, en su versión actual, no soporta la validación de firmas OOXML. Es posible comprobar la validez de estas firmas abriendo el documento con Microsoft Office 2007 o superior.

4 Información sobre el módulo

4.1 Compatibilidad

El módulo de firmas XML del cliente @firma es compatible con la versión 3 y superiores del cliente, en todas las plataformas, sistemas operativos, arquitecturas y versiones de entorno de ejecución de Java soportadas por este.

El módulo es únicamente compatible con la versión V3 y superiores del Cliente de la Plataforma @firma 5.

Las firmas que genera el módulo son interpretadas correctamente solo por OpenOffice.org 3.0, 3.1 y superiores.

4.2 Firmas de contenido binario en XMLDSig y XAdES

Cuando se indica al Cliente que se desea firmar datos binarios y estos se facilitan efectivamente en formato binario, el cliente los transforma a Base64 antes de firmarlos, de forma que estos puedan ser insertados en el XML.

La normativa (http://www.w3.org/TR/xmldsig-core/#sec-Object) indica que siempre que sobre los datos originales se realice una transformación de este tipo, esta debe declararse en la referencia de la firma XML mediante el algoritmo http://www.w3.org/2000/09/xmldsig#base64.

Declarando esta transformación, la huella digital firmada corresponderá con el original binario, y no con su versión codificada en Base64.

No obstante, el cliente incorpora ciertas peculiaridades en la aplicación de esta norma:

- Únicamente se declaran trasformaciones Base64 en las firmas XAdES y XMLDSig. OOXML y ODF no necesitan estas transformaciones nunca.
- Únicamente se declaran transformaciones Base64 de forma automática si es el cliente el que realiza la transformación de binario a Base64. Si los datos se proporcionan ya en Base64 (la transformación se ha realizado externamente), es el integrador quien debe asegurarse de que, si desea que se declare esta transformación, se añade de forma manual.

4.2.1 Deshabilitación de las transformaciones Base64

Por si se produjesen problemas de compatibilidad con otros sistemas no compatibles con la normativa XMLDSig / XAdES en cuanto a transformaciones Base64, el cliente establece un mecanismo de desactivación de esta característica mediante el siguiente método del Applet (que es posible invocar vía JavaScript): `SignApplet.addExtraParam(String paramName, String paramValue)`, y el uso concrerto:

```
signApplet.addExtraParam("avoidBase64Transforms", "true");
```

Una vez deshabilitada la declaración de las transformaciones Base64, estas se dejarán de añadir hasta que se reinicie el Applet o se vuelvan a habilitar mediante la llamada complementaria:

```
signApplet.addExtraParam("avoidBase64Transforms", "false");
```

Puede encontrar más información sobre el método `addExtraParam` en la documentación JavaDoc.

4.3 Firma de hojas de estilo en XMLDSig y XAdES

La normativa XMLDSig (punto 8.1.3) indica que si un XML declara una hoja de estilo, esta debe ser firmada junto al XML para que la firma asocie a este último con su visualización, respetando el concepto "lo que se ve es lo que se firma".

Dado que las hojas de estilo pueden declararse de distintas formas, el cliente adopta distintas estrategias para cada forma de declaración y según la variante de firma.

Las formas de declarar una hoja de estilo y la forma de firmar el XML en ese caso por el Cliente son las siguientes:

- La hoja de estilo está empotrada dentro del XML, y se declara con una referencia local (el valor de el atributo `href` de la declaración del XSL es un nombre de identificador de nodo XML precedido por "#").
  - En este caso no es necesaria ninguna estrategia adicional, pues al ser parte la hoja de estilo del XML, siempre que se firma uno, se firma también el otro. Esto aplica a la totalidad de las firmas XML.
- La hoja de estilo está accesible remotamente por protocolo HTTP o HTTPS (el valor del atributo `href` es una URL válida con esquema `http` ó `https`).
  - En este caso se añade una referencia a la firma que apunta a la hoja de estilo mediante la misma URL (una referencia Externally Detached). Esto aplica a la totalidad de las firmas XML).
- Se referencia a la hoja de estilo mediante una referencia relativa local.
  - En este caso, dado que las referencias relativas locales se pierden al firmar (el Applet no sabe en qué directorio o carpeta estaba el XML para localizar el XSL, y no puede asumir dónde se guardará la firma generada), las hojas de estilo no se firman.

4.3.1 Notas importantes sobre la firma de hojas de estilo XML

4.3.1.1 Desactivación y activación de firma de las hojas de estilo

El cliente dispone de un mecanismo de desactivación /activación de las firmas de hojas de estilo mediante el método del Applet (que es posible invocar vía JavaScript):

`SignApplet.addExtraParam(String paramName, String paramValue)`, y el siguiente uso:

Desactivar firma de hoja de estilo (comportamiento por defecto):

```
signApplet.addExtraParam("ignoreStyleSheets", "true");
```

Activar firma de hoja de estilo (comportamiento por defecto):

```
signApplet.addExtraParam("ignoreStyleSheets", "false");
```

Una vez deshabilitada o habilitada la firma de hojas de estilo, se mantendrá ese comportamiento hasta que se reinicie el Applet o se vuelva a cambiar mediante la llamada complementaria.
Por defecto el Applet de firma tiene desactivada la firma de hojas de estilo asociadas a documentos XML.

Puede encontrar más información sobre el método `addExtraParam` en la documentación JavaDoc.

4.3.1.2 Hojas de estilo anidadas

Las hojas de estilo XSL pueden referenciar a su vez a otras hojas de estilo mediante las clausulas `<xsl:include>` y `<xsl:import>`. Mediante estas etiquetas, se puede construir una cadena de ficheros de definición de estilos en forma de árbol, donde unas referencian a otras tanto de forma remota como local, en modo relativo o absoluto.

En la actualidad, el Cliente @firma no sigue la cadena de referencias para firmar la totalidad de los ficheros que definen el estilo del XML, sino que opera únicamente sobre el primer fichero referenciado por el XML en el atributo `href` de la etiqueta `xml-stylesheet`.

Intente en la medida de lo posible evitar el uso de `<xsl:include>` y `<xsl:import>` en hojas de estilo, y valore la posibilidad de incluir clausulas en su aplicación reflejando esta limitación.

4.3.2 Transformaciones XML

Los formatos de firma XAdES y XMLdSig permiten la configuración de transformaciones XML personalizadas. Estas transformaciones se aplican a un XML de datos antes de firmarlo, de tal forma que no se firma el XML original, sino el transformado. Las transformaciones quedan declaradas en la firma para que quede constancia de ellas y sea posible validar la firma por medio del XML de datos original.

La declaración de transformaciones XML para la firma es útil para la ejecución de firmas con un formato específico. Por ejemplo, la especificación de factura electrónica establece que no se debe firmar el XML de datos al completo, sino partes específicas del mismo.

Se permite la declaración de distintos tipos de transformaciones, algunos de los cuales tienen subtipos. La lógica de una transformación se especifica mediante un cuerpo, salvo en casos concretos en los que la lógica está predefinida por el tipo de transformación (caso de las transformaciones Base64 y Enveloped).

Los tipos de transformaciones soportados son:

- Transformación **XPATH**:
  - **Tipo:** `http://www.w3.org/TR/1999/REC-xpath-19991116`
  - **Subtipos:** No tiene subtipos.
  - **Cuerpo:** Especificado mediante sentencias de tipo XPATH.
- Transformación **XPATH2**:
  - **Tipo:** `http://www.w3.org/2002/06/xmldsig-filter2`
  - **Subtipos:**
    - **`subtract`**: Operación de substracción.
    - **`intersect`**: Operación de intersección.
    - **`union`**: Operación de unión.
  - **Cuerpo:** Especificado mediante sentencias de tipo XPATH2.
- Transformación **XSLT**:
  - **Tipo:** `http://www.w3.org/TR/1999/REC-xslt-19991116`
  - **Subtipos:** No tiene subtipos.
  - **Cuerpo:** Especificado mediante sentencias de tipo XSLT.
- Transformación **BASE64** (esta transformación ya se establece por defecto cuando se firman datos binarios, por lo que no se recomienda su uso de forma independiente):
  - **Tipo:** `http://www.w3.org/2000/09/xmldsig#base64`
  - **Subtipos:** No tiene subtipos.
  - **Cuerpo:** No tiene cuerpo.
- Transformación **ENVELOPED** (esta transformación ya se establece por defecto en los formatos de firma enveloped, por lo que no se recomienda su uso de forma independiente):
  - **Tipo:** `http://www.w3.org/2000/09/xmldsig#enveloped-signature`
  - **Subtipos:** No tiene subtipos.
  - **Cuerpo:** No tiene cuerpo.

No es posible especificar transformaciones complejas que incluyan varias sentencias. En su lugar, puede declararse una sucesión de transformaciones simples que produzcan el mismo resultado. Cada una de las transformaciones se aplicará de forma ordenada sobre el resultado de la anterior.

Las transformaciones personalizadas se declaran a través del método:

```
addXMLTransform(String tipo, String subtipo, String cuerpo)
```

Este método puede utilizarse sucesivas veces para establecer varias transformaciones que se apliquen ordenadamente.

Para eliminar todas las transformaciones declaradas se utiliza el método:

```
resetXMLTransforms()
```

4.4 Incidencias conocidas

4.4.1 Mensajes extraños en consola

El cliente, al firmar ficheros no-XML en modalidades implícitas o al firmar cualquier tipo de fichero (XML o no) en modalidades explícitas puede escribir el siguiente mensaje en consola:

```
[Fatal Error] :1:1: Content is not allowed in prolog.
```

Este mensaje debe ignorarse, ya que no se genera desde la aplicación, sino desde las clases de Sun Microsystems / Apache destinadas a identificar si el contenido del fichero es XML o no. No es posible ocultar este mensaje o evitar que se escriba en la consola.

4.5 Diferencias en las firmas generadas respecto a la versión anterior del cliente y sus módulos

- En las contrafirmas, la versión anterior del cliente identificaba las firmas según el IssuerName del certificado firmante. Si firma el documento dentro de una misma organización es muy probable que el IssuerName sea el mismo para todos los firmantes (normalmente, el certificado lo habrá emitido la misma autoridad de certificación para los empleados de una determinada organización), por lo que resultaba imposible discernir entre los firmantes. Cuando se recibe un documento ya firmado, para aceptar ese documento y firma, lo

importante es conocer quién ha firmado ese documento, por lo que es aconsejable que se identifiquen según el SubjectName o el CommonName de su certificado.

- Las versiones v.2.x y v3.0 del Cliente @firma nunca declaraban (en ningún formato XML) transformaciones Base64 cuando realizaba estas conversiones. La versión actual (ver puntos anteriores) declara estas transformaciones (aspecto requerido por las normativas) para las firmas XAdES y XMLDSig.
- Si se firma un XML que declara una hoja de estilo XSL en modo Enveloped, las versiones del cliente v2.x y 3.0 eliminaban la declaración de la hoja de estilo, mientras que el v3.1 mantiene esta declaración.
- Las versiones del cliente v.2.x y v3.0 no firmaban hojas de estilo asociadas a un XML, mientras que la versión v3.1 si las firma.

5 Ejemplos de Estructuras XML generadas por el módulo

5.1 XMLDSig

5.1.1 Detached Interno Explícito / Implícito

```
<?xml version="1.0" encoding="UTF-8" ?>
<AFIRMA>
 <CONTENT Encoding="..." Id="ID-CONTENT" MimeType="...">...</CONTENT>
 <ds:Signature xmlns:ds=http://www.w3.org/2000/09/xmldsig# Id="ID-Signature">
  <ds:SignedInfo>
   <ds:CanonicalizationMethod
    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
   <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
   <ds:Reference Id="ID-Reference" URI="#ID-CONTENT">
    <ds:Transforms>
     <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <ds:DigestValue>...</ds:DigestValue>
   </ds:Reference>
   <ds:Reference URI="#ID-KeyInfo">
    <ds:Transforms>
     <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <ds:DigestValue>...</ds:DigestValue>
   </ds:Reference>
  </ds:SignedInfo>
```

```
  <ds:SignatureValue Id="ID-SignatureValue">...</ds:SignatureValue>
  <ds:KeyInfo Id="2e9ff06f-a4cf-48f1-b670-b0dcdec56331-KeyInfo">
   <ds:KeyValue>
    <ds:RSAKeyValue><ds:Modulus>...</ds:Modulus><ds:Exponent>...</ds:Exponent></ds:RSAKeyValue>
   </ds:KeyValue>
   <ds:X509Data><ds:X509Certificate>...</ds:X509Certificate></ds:X509Data>
  </ds:KeyInfo>
 </ds:Signature>
</AFIRMA>
```

5.1.2 Enveloping Explícito / Implícito

```
<?xml version="1.0" encoding="UTF-8" ?>
<ds:Signature xmlns:ds=http://www.w3.org/2000/09/xmldsig# Id="ID-Signature">
  <ds:SignedInfo>
   <ds:CanonicalizationMethod
    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
   <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
   <ds:Reference Id="ID-Reference"
    Type="http://www.w3.org/2000/09/xmldsig#Object" URI="#ID-Object">
    <ds:Transforms>
     <ds:Transform
      Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <ds:DigestValue>...</ds:DigestValue>
   </ds:Reference>
   <ds:Reference URI="#ID-KeyInfo">
    <ds:Transforms>
     <ds:Transform
      Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <ds:DigestValue>...</ds:DigestValue>
   </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue Id="ID-SignatureValue">...</ds:SignatureValue>
  <ds:KeyInfo Id="ID-KeyInfo">
   <ds:KeyValue>
    <ds:RSAKeyValue>
     <ds:Modulus>...</ds:Modulus>
     <ds:Exponent>...</ds:Exponent>
    </ds:RSAKeyValue>
   </ds:KeyValue>
   <ds:X509Data>
    <ds:X509Certificate>...</ds:X509Certificate>
   </ds:X509Data>
  </ds:KeyInfo>
  <ds:Object Encoding="..." Id="ID-Object" MimeType="...">...</ds:Object>
</ds:Signature>
```

5.1.3 Enveloped

```
<?xml version="1.0" encoding="UTF-8" ?>
<documento>
 ...
  <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#" Id="ID-Signature">
   <ds:SignedInfo>
    <ds:CanonicalizationMethod
     Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
    <ds:Reference Id="ID-Reference" URI="">
     <ds:Transforms>
      <ds:Transform
       Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />
      <ds:Transform
       Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
     </ds:Transforms>
     <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
     <ds:DigestValue>...</ds:DigestValue>
    </ds:Reference>
    <ds:Reference URI="#ID-KeyInfo">
     <ds:Transforms>
      <ds:Transform
       Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
     </ds:Transforms>
     <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
     <ds:DigestValue>...</ds:DigestValue>
    </ds:Reference>
   </ds:SignedInfo>
   <ds:SignatureValue Id="ID-SignatureValue">...</ds:SignatureValue>
   <ds:KeyInfo Id="ID-KeyInfo">
    <ds:KeyValue>
     <ds:RSAKeyValue>
      <ds:Modulus>...</ds:Modulus>
      <ds:Exponent>...</ds:Exponent>
     </ds:RSAKeyValue>
    </ds:KeyValue>
    <ds:X509Data>
    <ds:X509Certificate>...</ds:X509Certificate>
    </ds:X509Data>
   </ds:KeyInfo>
  </ds:Signature>
</documento>
```

5.1.4 Co-Firmas

Ejemplo con "Detached Interna" y dos firmantes:

```
<?xml version="1.0" encoding="UTF-8" ?>
<AFIRMA>
 <CONTENT Encoding="base64" Id="ID-CONTENT" MimeType="...">...</CONTENT>
 <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  Id="ID-Signature">
  <ds:SignedInfo>
   <ds:CanonicalizationMethod
    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
   <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
   <ds:Reference Id="ID-Reference" URI="#ID-CONTENT">
    <ds:Transforms>
     <ds:Transform
      Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <ds:DigestValue>...</ds:DigestValue>
   </ds:Reference>
   <ds:Reference URI="#ID-KeyInfo">
    <ds:Transforms>
     <ds:Transform
      Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <ds:DigestValue>...</ds:DigestValue>
   </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue Id="ID-SignatureValue">...</ds:SignatureValue>
  <ds:KeyInfo Id="ID-KeyInfo">
   <ds:KeyValue>
    <ds:RSAKeyValue>
     <ds:Modulus>...</ds:Modulus>
     <ds:Exponent>...</ds:Exponent>
    </ds:RSAKeyValue>
   </ds:KeyValue>
   <ds:X509Data>
    <ds:X509Certificate>...</ds:X509Certificate>
   </ds:X509Data>
  </ds:KeyInfo>
 </ds:Signature>
 <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  Id="ID-Signature">
  <ds:SignedInfo>
   <ds:CanonicalizationMethod
    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
```

```
    <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
    <ds:Reference Id="ID-Reference" URI="#ID-CONTENT">
     <ds:Transforms>
      <ds:Transform
       Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
     </ds:Transforms>
     <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
     <ds:DigestValue>...</ds:DigestValue>
    </ds:Reference>
    <ds:Reference URI="#ID2-KeyInfo">
     <ds:Transforms>
      <ds:Transform
       Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
     </ds:Transforms>
     <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
     <ds:DigestValue>...</ds:DigestValue>
    </ds:Reference>
   </ds:SignedInfo>
   <ds:SignatureValue Id="ID-SignatureValue">...</ds:SignatureValue>
   <ds:KeyInfo Id="ID2-KeyInfo">
    <ds:KeyValue>
     <ds:RSAKeyValue>
      <ds:Modulus>...</ds:Modulus>
      <ds:Exponent>...</ds:Exponent>
     </ds:RSAKeyValue>
    </ds:KeyValue>
    <ds:X509Data>
     <ds:X509Certificate>...</ds:X509Certificate>
    </ds:X509Data>
   </ds:KeyInfo>
  </ds:Signature>
</AFIRMA>
```

5.1.5 Contrafirmas

Ejemplo con "Detached Interna" y dos firmantes:

```
<?xml version="1.0" encoding="UTF-8" ?>
<AFIRMA>
 <CONTENT Encoding="…" Id="ID-CONTENT" MimeType="...">...</CONTENT>
 <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  Id="ID-Signature1">
  <ds:SignedInfo>
   <ds:CanonicalizationMethod
    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
   <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
   <ds:Reference Id="ID-Reference" URI="#ID-CONTENT">
    <ds:Transforms>
     <ds:Transform
      Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <ds:DigestValue>sEfy8aIUCmWhhBUOsUiHyxvmLvs=</ds:DigestValue>
   </ds:Reference>
   <ds:Reference URI="#ID-KeyInfo">
    <ds:Transforms>
     <ds:Transform
      Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <ds:DigestValue>...</ds:DigestValue>
   </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue Id="ID-SignatureValue1">...</ds:SignatureValue>
<ds:KeyInfo Id="ID-KeyInfo">
   <ds:KeyValue>
    <ds:RSAKeyValue>
     <ds:Modulus>...</ds:Modulus>
     <ds:Exponent>...</ds:Exponent>
    </ds:RSAKeyValue>
   </ds:KeyValue>
   <ds:X509Data>
    <ds:X509Certificate>...</ds:X509Certificate>
   </ds:X509Data>
  </ds:KeyInfo>
 </ds:Signature>
 <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  Id="ID-Signature2">
  <ds:SignedInfo>
   <ds:CanonicalizationMethod
    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
```

```
  <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
  <ds:Reference Id="ID-Reference" URI="#ID-CONTENT">
   <ds:Transforms>
    <ds:Transform
     Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
   </ds:Transforms>
   <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
   <ds:DigestValue>…</ds:DigestValue>
  </ds:Reference>
  <ds:Reference URI="#ID-KeyInfo">
   <ds:Transforms>
    <ds:Transform
     Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
   </ds:Transforms>
   <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
   <ds:DigestValue>...</ds:DigestValue>
  </ds:Reference>
 </ds:SignedInfo>
 <ds:SignatureValue Id="ID-SignatureValue2">...</ds:SignatureValue>
 <ds:KeyInfo Id="ID-KeyInfo">
  <ds:KeyValue>
   <ds:RSAKeyValue>
    <ds:Modulus>...</ds:Modulus>
    <ds:Exponent>...</ds:Exponent>
   </ds:RSAKeyValue>
  </ds:KeyValue>
  <ds:X509Data>
   <ds:X509Certificate>...</ds:X509Certificate>
  </ds:X509Data>
 </ds:KeyInfo>
</ds:Signature>
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
 Id="ID-Signature3">
 <ds:SignedInfo>
  <ds:CanonicalizationMethod
   Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
  <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
  <ds:Reference Id="ID-Reference"
   Type="http://uri.etsi.org/01903#CountersignedSignature" URI="#ID-SignatureValue1">
   <ds:Transforms>
    <ds:Transform
     Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
   </ds:Transforms>
   <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
   <ds:DigestValue>...</ds:DigestValue>
  </ds:Reference>
  <ds:Reference URI="#ID-KeyInfo">
   <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
```

```
      <ds:DigestValue>...</ds:DigestValue>
     </ds:Reference>
    </ds:SignedInfo>
    <ds:SignatureValue Id="ID-SignatureValue3" />
    <ds:KeyInfo Id="ID-KeyInfo">
     <ds:KeyValue>
      <ds:RSAKeyValue>
       <ds:Modulus>...</ds:Modulus>
       <ds:Exponent>...</ds:Exponent>
      </ds:RSAKeyValue>
     </ds:KeyValue>
     <ds:X509Data>
      <ds:X509Certificate>...</ds:X509Certificate>
     </ds:X509Data>
    </ds:KeyInfo>
   </ds:Signature>
  </AFIRMA>
```

5.2 XAdES

5.2.1 Detached Interno Explícito / Implícito

```
<?xml version="1.0" encoding="UTF-8" ?>
<AFIRMA>
 <CONTENT Encoding="..." Id="ID-CONTENT" MimeType="...">...</CONTENT>
 <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  Id="ID-Signature">
  <ds:SignedInfo>
   <ds:CanonicalizationMethod
    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
   <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
   <ds:Reference Id="ID-Reference" URI="#ID-CONTENT">
    <ds:Transforms>
     <ds:Transform
      Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <ds:DigestValue>...</ds:DigestValue>
   </ds:Reference>
   <ds:Reference Type="http://uri.etsi.org/01903/v1.3.2#SignedProperties"
    URI="#ID-SignedProperties">
    <ds:Transforms>
     <ds:Transform
      Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
     </ds:Transforms>
     <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
     <ds:DigestValue>...</ds:DigestValue>
   </ds:Reference>
   <ds:Reference URI="#ID-KeyInfo">
    <ds:Transforms>
     <ds:Transform
      Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <ds:DigestValue>...</ds:DigestValue>
   </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue Id="ID-SignatureValue" />
  <ds:KeyInfo Id="ID-KeyInfo">
   <ds:KeyValue>
    <ds:RSAKeyValue>
     <ds:Modulus>...</ds:Modulus>
     <ds:Exponent>...</ds:Exponent>
    </ds:RSAKeyValue>
   </ds:KeyValue>
```

```
    <ds:X509Data>
     <ds:X509Certificate>...</ds:X509Certificate>
    </ds:X509Data>
   </ds:KeyInfo>
   <ds:Object>
    <xades:QualifyingProperties xmlns:dsign="http://www.w3.org/2000/09/xmldsig#"
     xmlns:xades="http://uri.etsi.org/01903/v1.3.2#" Id="ID-QualifyingProperties"
     Target="#ID-Signature">
     <xades:SignedProperties Id="ID-SignedProperties">
      <xades:SignedSignatureProperties>
       <xades:SigningTime>...</xades:SigningTime>
       <xades:SigningCertificate>
        <xades:Cert>
         <xades:CertDigest>
          <ds:DigestMethod Algorithm="..." />
          <ds:DigestValue>...</ds:DigestValue>
         </xades:CertDigest>
         <xades:IssuerSerial>
          <ds:X509IssuerName>...</ds:X509IssuerName>
          <ds:X509SerialNumber>...</ds:X509SerialNumber>
         </xades:IssuerSerial>
        </xades:Cert>
       </xades:SigningCertificate>
      </xades:SignedSignatureProperties>
      <xades:SignedDataObjectProperties>
       <xades:DataObjectFormat ObjectReference="#ID-Reference">
        <xades:Description />
        <xades:MimeType>...</xades:MimeType>
        <xades:Encoding>...</xades:Encoding>
       </xades:DataObjectFormat>
      </xades:SignedDataObjectProperties>
     </xades:SignedProperties>
    </xades:QualifyingProperties>
   </ds:Object>
  </ds:Signature>
</AFIRMA>
```

5.2.2 Enveloping Explícito / Implícito

```
<?xml version="1.0" encoding="UTF-8" ?>
 <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  Id="ID-Signature">
  <ds:SignedInfo>
   <ds:CanonicalizationMethod
    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
   <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
   <ds:Reference Id="ID-Reference"
    Type="http://www.w3.org/2000/09/xmldsig#Object" URI="#ID-Object">
    <ds:Transforms>
     <ds:Transform
      Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <ds:DigestValue>eg9yVHgqINMyyztUbeqlfJozTzY=</ds:DigestValue>
   </ds:Reference>
   <ds:Reference Type="http://uri.etsi.org/01903/v1.3.2#SignedProperties"
    URI="#ID-SignedProperties">
    <ds:Transforms>
     <ds:Transform
      Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <ds:DigestValue>8v1ohri++lnTP4D0b3P0gY9D7QI=</ds:DigestValue>
   </ds:Reference>
   <ds:Reference URI="#ID-KeyInfo">
    <ds:Transforms>
     <ds:Transform
      Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <ds:DigestValue>...</ds:DigestValue>
   </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue Id="ID-SignatureValue">...</ds:SignatureValue>
  <ds:KeyInfo Id="ID-KeyInfo">
   <ds:KeyValue>
    <ds:RSAKeyValue>
     <ds:Modulus>...</ds:Modulus>
     <ds:Exponent>...</ds:Exponent>
    </ds:RSAKeyValue>
   </ds:KeyValue>
   <ds:X509Data>
    <ds:X509Certificate>...</ds:X509Certificate>
   </ds:X509Data>
  </ds:KeyInfo>
```

```
<ds:Object Encoding="..." Id="ID-Object" MimeType="...">...</ds:Object>
<ds:Object>
 <xades:QualifyingProperties xmlns:dsign="http://www.w3.org/2000/09/xmldsig#"
  xmlns:xades="http://uri.etsi.org/01903/v1.3.2#" Id="ID-QualifyingProperties"
  Target="#ID-Signature">
  <xades:SignedProperties Id="ID-SignedProperties">
   <xades:SignedSignatureProperties>
    <xades:SigningTime>...</xades:SigningTime>
    <xades:SigningCertificate>
     <xades:Cert>
      <xades:CertDigest>
       <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
        <ds:DigestValue>...</ds:DigestValue>
       </xades:CertDigest>
       <xades:IssuerSerial>
        <ds:X509IssuerName>...</ds:X509IssuerName>
        <ds:X509SerialNumber>...</ds:X509SerialNumber>
       </xades:IssuerSerial>
     </xades:Cert>
    </xades:SigningCertificate>
   </xades:SignedSignatureProperties>
   <xades:SignedDataObjectProperties>
    <xades:DataObjectFormat ObjectReference="#ID-Reference">
     <xades:Description />
     <xades:MimeType>...</xades:MimeType>
     <xades:Encoding>...</xades:Encoding>
    </xades:DataObjectFormat>
   </xades:SignedDataObjectProperties>
  </xades:SignedProperties>
 </xades:QualifyingProperties>
</ds:Object>
</ds:Signature>
```

5.2.3 Enveloped

```
<?xml version="1.0" encoding="UTF-8" ?>
<documento>
 ...
 <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  Id="ID-Signature">
  <ds:SignedInfo>
   <ds:CanonicalizationMethod
    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
   <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
   <ds:Reference Id="ID-Reference" URI="">
    <ds:Transforms>
     <ds:Transform
      Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />
     <ds:Transform
      Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <ds:DigestValue>...</ds:DigestValue>
   </ds:Reference>
   <ds:Reference Type="http://uri.etsi.org/01903/v1.3.2#SignedProperties"
    URI="#ID-SignedProperties">
    <ds:Transforms>
     <ds:Transform
      Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <ds:DigestValue>...</ds:DigestValue>
   </ds:Reference>
   <ds:Reference URI="#ID-KeyInfo">
    <ds:Transforms>
     <ds:Transform
      Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <ds:DigestValue>...</ds:DigestValue>
   </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue Id="ID-SignatureValue">...</ds:SignatureValue>
  <ds:KeyInfo Id="ID-KeyInfo">
   <ds:KeyValue>
    <ds:RSAKeyValue>
     <ds:Modulus>...</ds:Modulus>
     <ds:Exponent>...</ds:Exponent>
    </ds:RSAKeyValue>
   </ds:KeyValue>
   <ds:X509Data>
```

```
      <ds:X509Certificate>...</ds:X509Certificate>
     </ds:X509Data>
    </ds:KeyInfo>
    <ds:Object>
     <xades:QualifyingProperties xmlns:dsign="http://www.w3.org/2000/09/xmldsig#"
      xmlns:xades="http://uri.etsi.org/01903/v1.3.2#" Id="ID-QualifyingProperties"
      Target="#ID-Signature">
      <xades:SignedProperties Id="ID-SignedProperties">
       <xades:SignedSignatureProperties>
        <xades:SigningTime>...</xades:SigningTime>
        <xades:SigningCertificate>
         <xades:Cert>
          <xades:CertDigest>
           <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
           <ds:DigestValue>...</ds:DigestValue>
          </xades:CertDigest>
          <xades:IssuerSerial>
           <ds:X509IssuerName>...</ds:X509IssuerName>
           <ds:X509SerialNumber>...</ds:X509SerialNumber>
          </xades:IssuerSerial>
         </xades:Cert>
        </xades:SigningCertificate>
       </xades:SignedSignatureProperties>
       <xades:SignedDataObjectProperties>
        <xades:DataObjectFormat ObjectReference="#ID-Reference">
         <xades:Description />
         <xades:MimeType>...</xades:MimeType>
         <xades:Encoding>...</xades:Encoding>
        </xades:DataObjectFormat>
       </xades:SignedDataObjectProperties>
      </xades:SignedProperties>
     </xades:QualifyingProperties>
    </ds:Object>
   </ds:Signature>
  </documento>
```

5.2.4 Co-Firmas

Ejemplo con "Detached Interna" y dos firmantes:

```
<?xml version="1.0" encoding="UTF-8" ?>
<AFIRMA>
 <CONTENT Encoding="..." Id="ID-CONTENT" MimeType="...">...</CONTENT>
  <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
   Id="ID-Signature">
   <ds:SignedInfo>
    <ds:CanonicalizationMethod
     Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
    <ds:Reference Id="ID-CONTENT">
     <ds:Transforms>
      <ds:Transform
       Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
     </ds:Transforms>
     <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
     <ds:DigestValue>...</ds:DigestValue>
    </ds:Reference>
    <ds:Reference Type="http://uri.etsi.org/01903/v1.3.2#SignedProperties"
     URI="#ID-SignedProperties">
     <ds:Transforms>
      <ds:Transform
       Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
     </ds:Transforms>
     <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
     <ds:DigestValue>...</ds:DigestValue>
    </ds:Reference>
    <ds:Reference URI="#ID-KeyInfo">
     <ds:Transforms>
      <ds:Transform
       Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
     </ds:Transforms>
     <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
     <ds:DigestValue>...</ds:DigestValue>
    </ds:Reference>
   </ds:SignedInfo>
   <ds:SignatureValue Id="..."></ds:SignatureValue>
   <ds:KeyInfo Id="ID-KeyInfo">
    <ds:KeyValue>
     <ds:RSAKeyValue>
      <ds:Modulus>...</ds:Modulus>
      <ds:Exponent>...</ds:Exponent>
     </ds:RSAKeyValue>
    </ds:KeyValue>
    <ds:X509Data>
```

```
    <ds:X509Certificate>...</ds:X509Certificate>
   </ds:X509Data>
  </ds:KeyInfo>
  <ds:Object>
   <xades:QualifyingProperties xmlns:dsign="http://www.w3.org/2000/09/xmldsig#"
    xmlns:xades="http://uri.etsi.org/01903/v1.3.2#" Id="ID-QualifyingProperties"
    Target="#ID-Signature">
    <xades:SignedProperties Id="ID-SignedProperties">
     <xades:SignedSignatureProperties>
      <xades:SigningTime>...</xades:SigningTime>
      <xades:SigningCertificate>
       <xades:Cert>
        <xades:CertDigest>
         <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
         <ds:DigestValue>...</ds:DigestValue>
        </xades:CertDigest>
        <xades:IssuerSerial>
         <ds:X509IssuerName>...</ds:X509IssuerName>
         <ds:X509SerialNumber>...</ds:X509SerialNumber>
        </xades:IssuerSerial>
       </xades:Cert>
      </xades:SigningCertificate>
     </xades:SignedSignatureProperties>
     <xades:SignedDataObjectProperties>
      <xades:DataObjectFormat ObjectReference="#ID-Reference">
       <xades:Description />
       <xades:MimeType>...</xades:MimeType>
       <xades:Encoding>...</xades:Encoding>
      </xades:DataObjectFormat>
     </xades:SignedDataObjectProperties>
    </xades:SignedProperties>
   </xades:QualifyingProperties>
  </ds:Object>
 </ds:Signature>
 <dsign:Signature xmlns:dsign="http://www.w3.org/2000/09/xmldsig#"
  Id="ID-Signature">
  <dsign:SignedInfo>
   <dsign:CanonicalizationMethod
    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
   <dsign:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
   <dsign:Reference Id="ID-Reference" URI="#ID-CONTENT">
    <dsign:Transforms>
     <dsign:Transform
      Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    </dsign:Transforms>
    <dsign:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <dsign:DigestValue>...</dsign:DigestValue>
   </dsign:Reference>
```

```
 <dsign:Reference Type="http://uri.etsi.org/01903/v1.3.2#SignedProperties"
  URI="#ID-SignedProperties">
  <dsign:Transforms>
   <dsign:Transform
    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
  </dsign:Transforms>
  <dsign:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
  <dsign:DigestValue>...</dsign:DigestValue>
 </dsign:Reference>
 <dsign:Reference URI="#ID2-KeyInfo">
  <dsign:Transforms>
   <dsign:Transform
    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
  </dsign:Transforms>
  <dsign:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
  <dsign:DigestValue>...</dsign:DigestValue>
 </dsign:Reference>
</dsign:SignedInfo>
<dsign:SignatureValue Id="ID-SignatureValue">...</dsign:SignatureValue>
<dsign:KeyInfo Id="ID2-KeyInfo">
 <dsign:KeyValue>
  <dsign:RSAKeyValue>
   <dsign:Modulus>...</dsign:Modulus>
   <dsign:Exponent>...</dsign:Exponent>
  </dsign:RSAKeyValue>
 </dsign:KeyValue>
 <dsign:X509Data>
  <dsign:X509Certificate>...</dsign:X509Certificate>
 </dsign:X509Data>
</dsign:KeyInfo>
<dsign:Object>
 <xades:QualifyingProperties xmlns:xades="http://uri.etsi.org/01903/v1.3.2#"
  Id="ID-QualifyingProperties" Target="#ID-Signature">
  <xades:SignedProperties Id="ID-SignedProperties">
   <xades:SignedSignatureProperties>
    <xades:SigningTime>...</xades:SigningTime>
    <xades:SigningCertificate>
     <xades:Cert>
      <xades:CertDigest>
       <dsign:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
       <dsign:DigestValue>...</dsign:DigestValue>
      </xades:CertDigest>
      <xades:IssuerSerial>
       <dsign:X509IssuerName>...</dsign:X509IssuerName>
       <dsign:X509SerialNumber>...</dsign:X509SerialNumber>
      </xades:IssuerSerial>
     </xades:Cert>
    </xades:SigningCertificate>
```

```
        </xades:SignedSignatureProperties>
        <xades:SignedDataObjectProperties />
       </xades:SignedProperties>
      </xades:QualifyingProperties>
    </dsign:Object>
  </dsign:Signature>
</AFIRMA>
```

5.2.5 Contrafirmas

Ejemplo con "Detached Interna" y dos firmantes:

```
<?xml version="1.0" encoding="UTF-8" ?>
<AFIRMA>
<CONTENT Encoding="..." Id="ID-CONTENT" MimeType="...">...</CONTENT>
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
 Id="ID-Signature">
 <ds:SignedInfo>
  <ds:CanonicalizationMethod
   Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
  <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
  <ds:Reference Id="ID-Reference" URI="#ID-CONTENT">
   <ds:Transforms>
    <ds:Transform
     Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
   </ds:Transforms>
   <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
   <ds:DigestValue>...</ds:DigestValue>
  </ds:Reference>
  <ds:Reference Type="http://uri.etsi.org/01903/v1.3.2#SignedProperties"
   URI="#ID-SignedProperties">
   <ds:Transforms>
    <ds:Transform
     Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
   </ds:Transforms>
   <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
   <ds:DigestValue>...</ds:DigestValue>
  </ds:Reference>
  <ds:Reference URI="#ID-KeyInfo">
   <ds:Transforms>
    <ds:Transform
     Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
   </ds:Transforms>
   <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
   <ds:DigestValue>...</ds:DigestValue>
  </ds:Reference>
 </ds:SignedInfo>
<ds:SignatureValue Id="..."></ds:SignatureValue>
<ds:KeyInfo Id="ID-KeyInfo">
 <ds:KeyValue>
  <ds:RSAKeyValue>
   <ds:Modulus>...</ds:Modulus>
   <ds:Exponent>...</ds:Exponent>
   </ds:RSAKeyValue>
 </ds:KeyValue>
 <ds:X509Data>
```

```
    <ds:X509Certificate>...</ds:X509Certificate>
   </ds:X509Data>
  </ds:KeyInfo>
  <ds:Object>
   <xades:QualifyingProperties xmlns:dsign="http://www.w3.org/2000/09/xmldsig#"
     xmlns:xades="http://uri.etsi.org/01903/v1.3.2#" Id="ID-QualifyingProperties"
     Target="#ID-Signature">
    <xades:SignedProperties Id="ID-SignedProperties">
     <xades:SignedSignatureProperties>
      <xades:SigningTime>...</xades:SigningTime>
      <xades:SigningCertificate>
       <xades:Cert>
        <xades:CertDigest>
         <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
         <ds:DigestValue>...</ds:DigestValue>
        </xades:CertDigest>
        <xades:IssuerSerial>
         <ds:X509IssuerName>...</ds:X509IssuerName>
         <ds:X509SerialNumber>...</ds:X509SerialNumber>
        </xades:IssuerSerial>
       </xades:Cert>
      </xades:SigningCertificate>
     </xades:SignedSignatureProperties>
     <xades:SignedDataObjectProperties>
      <xades:DataObjectFormat ObjectReference="#ID-Reference">
       <xades:Description />
       <xades:MimeType>...</xades:MimeType>
       <xades:Encoding>...</xades:Encoding>
      </xades:DataObjectFormat>
     </xades:SignedDataObjectProperties>
    </xades:SignedProperties>
   </xades:QualifyingProperties>
  </ds:Object>
 </ds:Signature>
 <dsign:Signature xmlns:dsign="http://www.w3.org/2000/09/xmldsig#"
  Id="ID-Signature">
  <dsign:SignedInfo>
   <dsign:CanonicalizationMethod
    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
   <dsign:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
   <dsign:Reference Id="ID-Reference" URI="#ID-CONTENT">
    <dsign:Transforms>
     <dsign:Transform
      Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
    </dsign:Transforms>
    <dsign:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <dsign:DigestValue>5nfu4+SLSMHDs+CwGRGGoSjNyhw=</dsign:DigestValue>
   </dsign:Reference>
```

```
 <dsign:Reference Type="http://uri.etsi.org/01903/v1.3.2#SignedProperties"
  URI="#26561cf7-e5c0-455a-b220-554a822227b8-SignedProperties">
  <dsign:Transforms>
   <dsign:Transform
    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
  </dsign:Transforms>
  <dsign:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
  <dsign:DigestValue>...</dsign:DigestValue>
 </dsign:Reference>
 <dsign:Reference URI="#ID-KeyInfo">
  <dsign:Transforms>
   <dsign:Transform
    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
  </dsign:Transforms>
  <dsign:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
  <dsign:DigestValue>...</dsign:DigestValue>
 </dsign:Reference>
</dsign:SignedInfo>
<dsign:SignatureValue Id="ID-SignatureValue">...</dsign:SignatureValue>
<dsign:KeyInfo Id="ID-KeyInfo">
 <dsign:KeyValue>
  <dsign:RSAKeyValue>
   <dsign:Modulus>...</dsign:Modulus>
   <dsign:Exponent>...</dsign:Exponent>
  </dsign:RSAKeyValue>
 </dsign:KeyValue>
 <dsign:X509Data>
  <dsign:X509Certificate>...</dsign:X509Certificate>
 </dsign:X509Data>
</dsign:KeyInfo>
<dsign:Object>
 <xades:QualifyingProperties xmlns:xades="http://uri.etsi.org/01903/v1.3.2#"
  Id="ID-QualifyingProperties" Target="#ID-Signature">
  <xades:SignedProperties Id="ID-SignedProperties">
   <xades:SignedSignatureProperties>
    <xades:SigningTime>...</xades:SigningTime>
    <xades:SigningCertificate>
     <xades:Cert>
      <xades:CertDigest>
       <dsign:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
       <dsign:DigestValue>...</dsign:DigestValue>
      </xades:CertDigest>
      <xades:IssuerSerial>
       <dsign:X509IssuerName>...</dsign:X509IssuerName>
       <dsign:X509SerialNumber>...</dsign:X509SerialNumber>
      </xades:IssuerSerial>
     </xades:Cert>
    </xades:SigningCertificate>
```

```
 </xades:SignedSignatureProperties>
 <xades:SignedDataObjectProperties />
</xades:SignedProperties>
<UnsignedProperties>
 <UnsignedSignatureProperties>
  <CounterSignature>
   <dsign:Signature Id="ID-Signature">
    <dsign:SignedInfo>
     <dsign:CanonicalizationMethod
      Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
     <dsign:SignatureMethod
      Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
     <dsign:Reference Id="ID-Reference" URI="#ID-SignatureValue">
      <dsign:Transforms>
       <dsign:Transform
        Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
      </dsign:Transforms>
      <dsign:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
      <dsign:DigestValue>...</dsign:DigestValue>
     </dsign:Reference>
     <dsign:Reference
      Type="http://uri.etsi.org/01903/v1.3.2#SignedProperties" URI="#ID-SignedProperties">
      <dsign:Transforms>
       <dsign:Transform
        Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
      </dsign:Transforms>
      <dsign:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
      <dsign:DigestValue>...</dsign:DigestValue>
     </dsign:Reference>
     <dsign:Reference URI="#ID-KeyInfo">
      <dsign:Transforms>
       <dsign:Transform
        Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments" />
      </dsign:Transforms>
      <dsign:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
      <dsign:DigestValue>...</dsign:DigestValue>
     </dsign:Reference>
    </dsign:SignedInfo>
    <dsign:SignatureValue Id="ID-SignatureValue">...</dsign:SignatureValue>
    <dsign:KeyInfo Id="ID-KeyInfo">
     <dsign:KeyValue>
      <dsign:RSAKeyValue>
       <dsign:Modulus>...</dsign:Modulus>
       <dsign:Exponent>...</dsign:Exponent>
      </dsign:RSAKeyValue>
     </dsign:KeyValue>
     <dsign:X509Data>
      <dsign:X509Certificate>...</dsign:X509Certificate>
```

```
          </dsign:X509Data>
         </dsign:KeyInfo>
         <dsign:Object>
          <xades:QualifyingProperties Id="ID-QualifyingProperties" Target="#ID-Signature">
           <xades:SignedProperties Id="ID-SignedProperties">
           <xades:SignedSignatureProperties>
           <xades:SigningTime>...</xades:SigningTime>
           <xades:SigningCertificate>
            <xades:Cert>
             <xades:CertDigest>
              <dsign:DigestMethod
               Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
              <dsign:DigestValue>...</dsign:DigestValue>
             </xades:CertDigest>
             <xades:IssuerSerial>
              <dsign:X509IssuerName>...</dsign:X509IssuerName>
              <dsign:X509SerialNumber>...</dsign:X509SerialNumber>
             </xades:IssuerSerial>
            </xades:Cert>
           </xades:SigningCertificate>
          </xades:SignedSignatureProperties>
          <xades:SignedDataObjectProperties />
         </xades:SignedProperties>
        </xades:QualifyingProperties>
       </dsign:Object>
      </dsign:Signature>
     </CounterSignature>
    </UnsignedSignatureProperties>
   </UnsignedProperties>
  </xades:QualifyingProperties>
</dsign:Object>
</dsign:Signature>
</AFIRMA>
```

5.2.6 Distintas versiones de XAdES

Por defecto, el cliente genera un tipo de firma que se adecúa a dos versiones de XAdES, la 1.3.2 y la 1.4.1 (BES y EPES), y declara el espacio de nombres según la 1.3.2 (http://uri.etsi.org/01903/v1.3.2#).

Si queremos que las firmas se declaren con el espacio de nombres de la versión 1.4.1 (útil si queremos que la firma esté preparada para añadir características avanzadas de sello de tiempo según la última versión de XAdES), debemos indicárselo expresamente al Applet mediante un "parámetro extraordinario".

La forma de realizar estas indicaciones es mediante el método del Applet

`SignApplet.addExtraParam(String paramName, String paramValue)`, y el siguiente uso:

```
signApplet.addExtraParam("xadesNamespace",
"http://uri.etsi.org/01903/v1.4.1#");
```

Los espacios de nombres que podemos indicar mediante esta táctica (siempre indicando como nombre de parámetro "`xadesNamespace`") y están aceptados por la normativa son:

"`http://uri.etsi.org/01903/v1.3.2#`"
Versión v1.3.2 de XAdES.

"`http://uri.etsi.org/01903/v1.4.1#`"
Versión v.1.4.1 de XAdES.

Cuando establezcamos un valor para el espacio de nombres de XAdES, este se utilizará para todas las firmas XAdES que se realicen a partir de entonces (o hasta una re-inicialización del Applet). Para restablecer el valor por defecto debemos deshacer el establecimiento de parámetros extraordinarios con el método equivalente del Applet, `SignApplet.removeExtraParam(String paramName)`, que en nuestro caso se concretaría en:

```
signApplet.removeExtraParam("xadesNamespace");
```

Estos métodos pueden llamarse desde JavaScript, y puede encontrar información adicional en la documentación JavaDoc del Applet @firma.

6 Información de utilidad

Normativa XMLDSig

http://www.w3.org/TR/xmldsig-core/

Normativa XAdES

http://www.etsi.org

7 Glosario de términos

Firma electrónica

Es el conjunto de datos, en forma electrónica, anejos a otros datos electrónicos o asociados funcionalmente con ellos, utilizados como medio para identificar formalmente al autor o a los autores del documento que la recoge.

XML Digital Signature (XMLDSig)

Es una recomendación del W3C que define una sintaxis XML para la firma digital

XML Advanced Signature (XAdES)

Es un conjunto de extensiones a las recomendaciones XML-DSig haciéndolas adecuadas para la firma electrónica avanzada.

RSA

Es un sistema criptográfico de clave pública desarrollado en 1977. En la actualidad, RSA es el primer y más utilizado algoritmo de este tipo y es válido tanto para cifrar como para firmar digitalmente.

XML

Es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium (W3C). Es una simplificación y adaptación del SGML y permite definir la gramática de lenguajes específicos (de la misma manera que HTML es a su vez un lenguaje definido por SGML). Por lo tanto XML no es realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades. Algunos de estos lenguajes que usan XML para su definición son XHTML, SVG, MathML.

Office Open XML (OOXML)

Es un formato de archivo abierto y estándar cuyas extensiones más comunes son .docx, .xlsx y .pptx. Se le utiliza para representar y almacenar hojas de cálculo, diagramas, presentaciones y documentos de

texto. Un archivo Office Open XML contiene principalmente datos basados en el lenguaje de marcado XML, comprimidos en un contenedor .zip específico.

Open Document Format (ODF)

Es un formato de fichero estándar para el almacenamiento de documentos ofimáticos tales como hojas de cálculo, memorandos, gráficas y presentaciones. Aunque las especificaciones fueron inicialmente elaboradas por Sun, el estándar fue desarrollado por el comité técnico para Open Office XML de la organización OASIS y está basado en un esquema XML inicialmente creado e implementado por la suite ofimática OpenOffice.org (ver OpenOffice.org XML).

ZIP

Es un formato de almacenamiento sin pérdida, muy utilizado para la compresión de datos como imágenes, programas o documentos.

PDF

Es un formato de almacenamiento de documentos, desarrollado por la empresa Adobe Systems. Este formato es de tipo compuesto (imagen vectorial, mapa de bits y texto).

SHA

Es un sistema de funciones hash criptográficas relacionadas de la Agencia de Seguridad Nacional de los Estados Unidos y publicadas por el National Institute of Standards and Technology (NIST). El primer miembro de la familia fue publicado en 1993 es oficialmente llamado SHA. Sin embargo, hoy día, no oficialmente se le llama SHA-0 para evitar confusiones con sus sucesores. Dos años más tarde el primer sucesor de SHA fue publicado con el nombre de SHA-1. Existen cuatro variantes más que se han publicado desde entonces cuyas diferencias se basan en un diseño algo modificado y rangos de salida incrementados: SHA-224, SHA-256, SHA-384, y SHA-512 (llamándose SHA-2 a todos ellos).

PKCS

Se refiere a un grupo de estándares de criptografía de clave pública concebidos y publicados por los laboratorios de RSA en California. A RSA Security se le asignaron los derechos de licenciamiento para la patente de algoritmo de clave asimétrica RSA y adquirió los derechos de licenciamiento para muchas otras patentes de claves.

W3C

Es un consorcio internacional que produce recomendaciones para la World Wide Web. Está dirigida por Tim Berners-Lee, el creador original de URL (Uniform Resource Locator, Localizador Uniforme de

Recursos), HTTP (HyperText Transfer Protocol, Protocolo de Transferencia de HiperTexto) y HTML (Lenguaje de Marcado de HiperTexto) que son las principales tecnologías sobre las que se basa la Web.

OpenOffice.org

Es una suite ofimática libre (código abierto y distribución gratuita) que incluye herramientas como procesador de textos, hoja de cálculo, presentaciones, herramientas para el dibujo vectorial y base de datos. Está disponible para varias plataformas, tales como Microsoft Windows, GNU/Linux, BSD, Solaris y Mac OS X. Soporta numerosos formatos de archivo, incluyendo como predeterminado el formato estándar ISO/IEC OpenDocument (ODF), entre otros formatos comunes. A febrero de 2010, OpenOffice soporta más de 110 idiomas.

Base64

Es un sistema de numeración posicional que usa 64 como base. Es la mayor potencia de dos que puede ser representada usando únicamente los caracteres imprimibles de ASCII. Esto ha propiciado su uso para codificación de correos electrónicos, PGP y otras aplicaciones. Todas las variantes famosas que se conocen con el nombre de Base64 usan el rango de caracteres A-Z, a-z y 0-9 en este orden para los primeros 62 dígitos, pero los símbolos escogidos para los últimos dos dígitos varían considerablemente de unas a otras. Otros métodos de codificación como UUEncode y las últimas versiones de binhex usan un conjunto diferente de 64 caracteres para representar 6 dígitos binarios, pero éstos nunca son llamados Base64.

ASN.1

Es una norma para representar datos independientemente de la máquina que se esté usando y sus formas de representación internas. Es un protocolo de nivel de presentación en el modelo OSI.

Autoridad de Certificación (CA)

Es una entidad de confianza, responsable de emitir y revocar los certificados digitales o certificados, utilizados en la firma electrónica, para lo cual se emplea la criptografía de clave pública. Jurídicamente es un caso particular de Prestador de Servicios de Certificación.

Certificado Digital

Es un documento digital mediante el cual un tercero confiable (una autoridad de certificación) garantiza la vinculación entre la identidad de un sujeto o entidad y su clave pública.

Infraestructura de Clave Pública (PKI)

Es una combinación de hardware y software, políticas y procedimientos de seguridad que permiten la ejecución con garantías de operaciones criptográficas como el cifrado, la firma digital o el no repudio de transacciones electrónicas.

8 Información de contacto

Soporte a la Administración Electrónica
Consejería de Hacienda y Administración Pública
Dirección General de Tecnologías para Hacienda y la Administración Electrónica
soporte.admonelectronica@juntadeandalucia.es