Seguridad en XML

Jose Emilio Labra Gayo

Noviembre 2006

Contenidos

Requisitos de seguridad

- *Confidencialidad*: garantizar que la información no es visible a extraños
- *Autenticación*: garantiza que el acceso a la información sólo puede ser realizado por quienes proporcionan la identidad adecuada.
- *Integridad*: asegurar que el mensaje no ha sido modificado accidental o deliberadamente
- *No repudiación*: garantiza que el emisor del mensaje no puede negar haberlo enviado

Claves simétricas vs asimétricas

Un sistema utiliza claves simétricas si la misma clave sirve para cifrar y para descifrar un documento
En un sistema de claves asimétricas se utiliza una clave para cifrar y otra para descifrar el documento

Canonización

Un mismo contenido puede ser representado de varias formas en XML

Doc1.xml

```
<img src="pp.jpg" alt="Foto de Pepe"></img>
```

Doc2.xml

```
<img src="pp.jpg" alt="Foto de Pepe" />
```

Doc3.xml

```
<img alt="Foto de Pepe" src="pp.jpg" />
```

Canonización

Varias propuestas de W3C:

- C14N (Marzo 2001)
- Exclusive C14N (Julio 2002) soluciona algunos problemas del anterior.

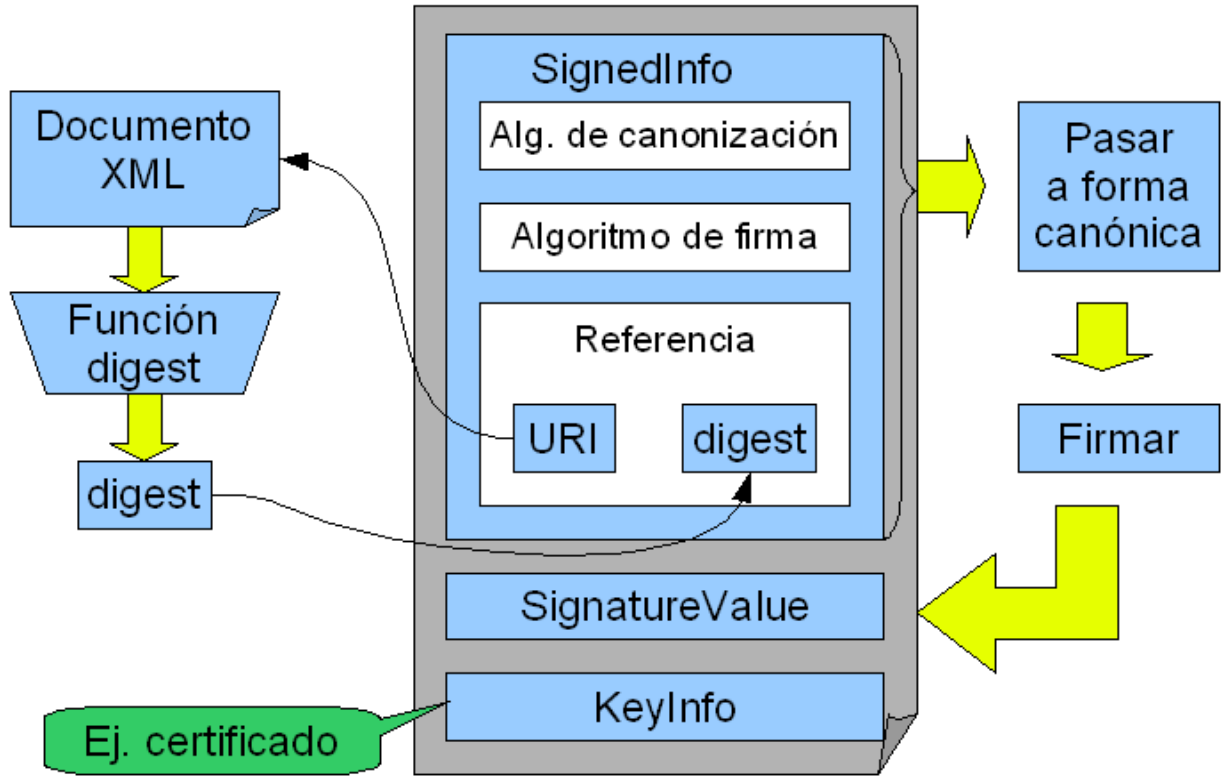
XML Signature

XML Digital Signature (2002) permite demostrar la identidad y autenticidad del documento
Objetivo: Integridad y autenticación
Es básica la canonización para identificar documentos comunes

Estructura de XML SIgnature

- `SignedInfo` contiene la información que es firmada. Incluye:
  - `CanonizalizationMethod`: algoritmo utilizado para canonizar el elemento
  - `SignatureMethod`: algoritmo para convertir `signedInfo` en `signatureValue`
  - `Reference` incluye el método para obtener la firma (`digest`) y el valor, así como otras transformaciones realizadas
- `SignatureValue` incluye el valor de la firma
- `KeyInfo` indica la la clave a utilizar para validar la firma

Estructura de XML Signature



Ejemplo de XML Digital Signature (1)

```
<Signature Id="MyFirstSignature"
 xmlns="http://www.w3.org/2000/09/xmldsig#">
<SignedInfo>
 <CanonicalizationMethod
    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315" />
 <SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#dsa
 <Reference URI="http://www.w3.org/TR/2000/REC-xhtml1-20000126/">
 <Transforms>
  <Transform
    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315" />
 </Transforms>
 <DigestMethod
    Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
 <DigestValue>j6lwx3rvEPO0vKtMup4NbeVu8nk=</DigestValue>
 </Reference>
</SignedInfo>
...
```

Ejemplo de XML Digital Signature (2)

```
...
 <SignatureValue>MC0CFFrVLtRlk=...</SignatureValue>
 <KeyInfo>
  <KeyValue>
  <DSAKeyValue>
    <p>...</p><Q>...</Q><G>...</G><Y>...</Y>
  </DSAKeyValue>
 </KeyValue>
</KeyInfo>
</Signature>
```

XML Encryption

Objetivo: proteger una información que se envía del acceso no autorizado por terceras partes
Permite cifrar:

- Un documento XML completo
- Un elemento de un documento XML
- El contenido de un elemento
- Datos textuales que no son XML
- Contenidos ya cifrados

Componentes de XML Encryption

El elemento `EncryptedData` consta de:

- `EncryptionMethod`
- `KeyInfo`
- `CipherData`

Ejemplo: cifrado de una parte

```
<pedido><nombre>Pepe</nombre>
<formaPago><tarjeta tipo="Visa">
 <num>2323 4121 2445 8976</num>
 <titular>Jose Pablo Herrera</titular>
 <fecha>02/09</fecha>
</tarjeta></formaPago>
</pedido>
```

```
<pedido><nombre>Pepe</nombre>
<formaPago>
<EncryptedData Type="http://www.w3.org/2001/04/xmlenc#Element"
 xmlns="http://www.w3.org/2001/04/xmlenc#">
<CipherData><CipherValue>C5723A...</CipherValue></CipherData>
</EncryptedData>
</formaPago>
</pedido>
```

Ejemplo: cifrado de un valor

Puede cifrarse únicamente el valor de un elemento. Por ejemplo el número de tarjeta

```
<pedido><nombre>Pepe</nombre>
<formaPago><tarjeta tipo="Visa">
 <num><EncryptedData Type="http://www.w3.org/2001/04/xmlenc#Conten
 xmlns="http://www.w3.org/2001/04/xmlenc#">
<CipherData><CipherValue>B6235...</CipherValue></CipherData>
</EncryptedData>
 </num>
 <titular>Jose Pablo Herrera</titular>
 <fecha>02/09</fecha>
</tarjeta>
</formaPago>
</pedido>
```

Ejemplo: cifrado de un valor

Puede cifrarse únicamente el valor de un elemento. Por ejemplo el número de tarjeta

```
<EncryptedData
 MimeType="text/xml"
 xmlns="http://www.w3.org/2001/04/xmlenc#">
<CipherData><CipherValue>B6235...</CipherValue></CipherData>
</EncryptedData>
```

XKMS

Objetivo: Ocultar al usuario la complejidad subyacente en PKI

2 partes:

- Registro de clave pública
  - X-KISS (*XML Key Information Service Specification*)
  - Procesar el campo `KeyInfo` contenido en las firmas digitales
- Información de clave pública
  - X-KRSS (*XML key Registration Service Specification*)
  - Buscar la clave pública de alguien y comprobar si es buena

SAML

SAML (*Security Assertion Markup Language*) permite compartir información de autentificación entre aplicaciones
Se basa en autoridades que emiten aserciones
Transmite varios tipos de aserciones:

- Aserción de autenticación
- Aserción de atributo

XACML

XACML (*XML Access Control Markup Language*) permite expresar políticas de control de acceso
Codifica en reglas XML expresiones del tipo:

- Una persona puede leer un informe si es el propio paciente

XrML

XrML (*eXtensible Rights Markup Language*)
Permite expresar reglas sobre derechos
Codifica expresiones del tipo:

- Un consumidor puede ver una película 6 veces durante 15 días

Fin de la presentación