Universidad
Carlos III de Madrid
COSEC Lab · *Dpto. Informática*

TEMA 2. CRIPTOGRAFÍA
T 2.6 FIRMA DIGITAL

Criptografía y seguridad informática
Seguridad en las tecnologías de la información
@ COSEC

Curso 2016-2017

Índice

- Introducción
- Propiedades
- Variantes
- Algoritmos:
  - DSS/ElGamal
  - RSA
- Representación y formatos
- Anexo

Definición según estándares

- [RFC 4949:2007]
  - Un valor calculado con un algoritmo criptográfico y que se asocia con un objeto de datos de tal manera que cualquier destinatario de los datos puede utilizar la firma para verificar el origen de los datos y la integridad
  - [Norma ISO/IEC 7498-2:1989] Datos añadidos a un conjunto de datos, o transformación de éstos, que permiten al receptor probar el origen y la integridad de los datos recibidos, así como protegerlos contra falsificaciones
- [SP800-57:2007] El resultado de una transformación criptográfica de datos que, si se aplica correctamente, según la infraestructura y políticas, proporciona los servicios de:
  - Autenticación del origen,
  - Integridad de los datos, y
  - No repudio del firmante

Ir ligada a un mensaje, si es valida para uno no vale para otro
Solo puede ser escrita por la persona a la que legalmente corresponde
Publicamente verificable
Lo habitual es usar un cifrador simetrico y una función resumen.

Firma digital con clave pública

- Introducida por Diffie y Hellman en 1976
- Analogía electrónica de la firma manual
- Propiedades de una firma manual:
  - Fácil y barata de producir
  - Fácil de reconocer
  - Imposible de rechazar por el propietario
  - Infalsificable (teóricamente)
- La firma digital debería cumplir las mismas propiedades, pero:
  - No puede ser siempre la misma ya que sería fácilmente falsificable.

Propiedades de seguridad

1. **Autentica** indubitablemente al signatario de una información.
2. Garantía de la **integridad** del mensaje recibido al imposibilitar su modificación fraudulenta.
3. Garantía de **no repudio**: medio de prueba en la resolución de disputas.

No asegura la **confidencialidad.**

Componentes

- Un esquema de firma digital comporta dos partes:

1. Algoritmo de firma
2. Algoritmo de verificación de la firma

Firma digital: Determinista vs Aleatorio

- Un esquema de firma digital comporta dos partes:

1. Algoritmo de firma
2. Algoritmo de verificación de la firma

- El algoritmo de firma puede ser:
  - **Determinista:** Dos firmas del mismo mensaje producen el mismo resultado (por ejemplo, las firmas basadas en el algoritmo RSA)
  - **Aleatoria:** dependiente de un conjunto de índices (por ejemplo las basadas en el algoritmo ElGamal)

Firma digital: Tipos (clasificación "local")

- **TIPO I:** La firma se vuelca en un apéndice. Se denomina **firma separada del mensaje o con apéndice** (se envían el apéndice F y M)
- **TIPO II:** La firma está integrada en el propio mensaje transformado. Se denomina **firma con recuperación del mensaje** a partir de la firma (se envía F únicamente)
- **TIPO III:** Esquema de firma con recuperación del mensaje transformado en esquema de firma separada con ayuda de una función resumen
- **Otros modelos** (Basada en MAC con clave secreta, Firma opaca…)

Firma digital: Separada del mensaje

- **TIPO I:** La firma se vuelca en un apéndice. Se denomina **firma separada del mensaje o con apéndice** (se envían el apéndice F y M)
  - DSA y ElGamal
  - Requieren el mensaje original como entrada para la verificación de la firma
  - Se aplica al mensaje original una función resumen antes de firmar

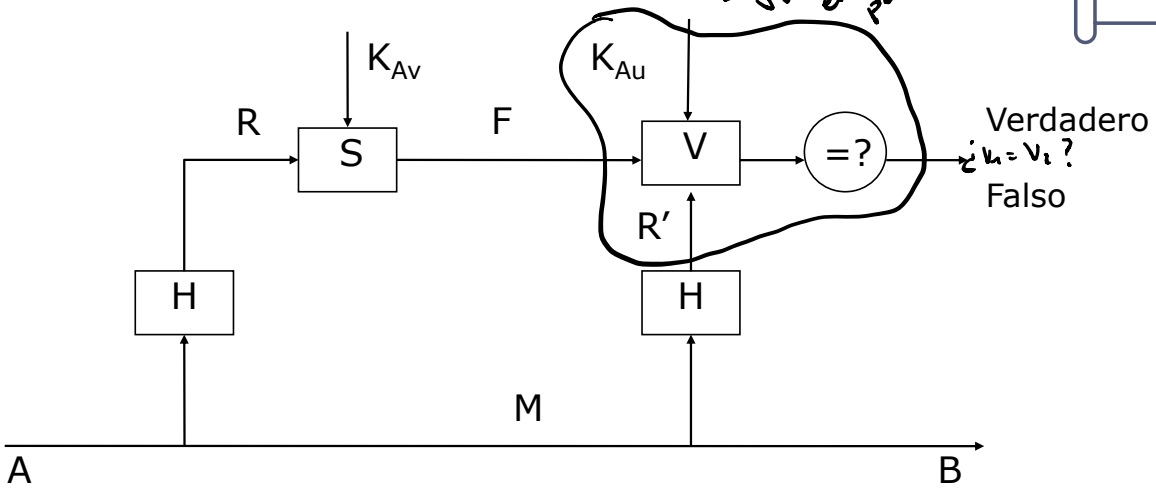
Firma separada del mensaje

Tipo I

- Protocolo de firma digital con apéndice F de un mensaje M:
  1. Obtener el comprimido resumen R=H(M) y su firma F=S(R) (obtenida con el algoritmo de firma S y la clave privada del remitente K_{Av})
  2. Enviar el par (M, F)
  3. El receptor calcula R' = H(M) a partir del primer elemento del par (el mensaje M)
  4. El receptor evalúa la validez de la firma recibida ejecutando el algoritmo de verificación de firma V con la clave pública del remitente K_{Au} y a partir del resumen calculado R' y la firma recibida F. Se acepta el mensaje si el resultado del algoritmo es verdadero y se rechaza en caso contrario

Firma separada del mensaje

Tipo I

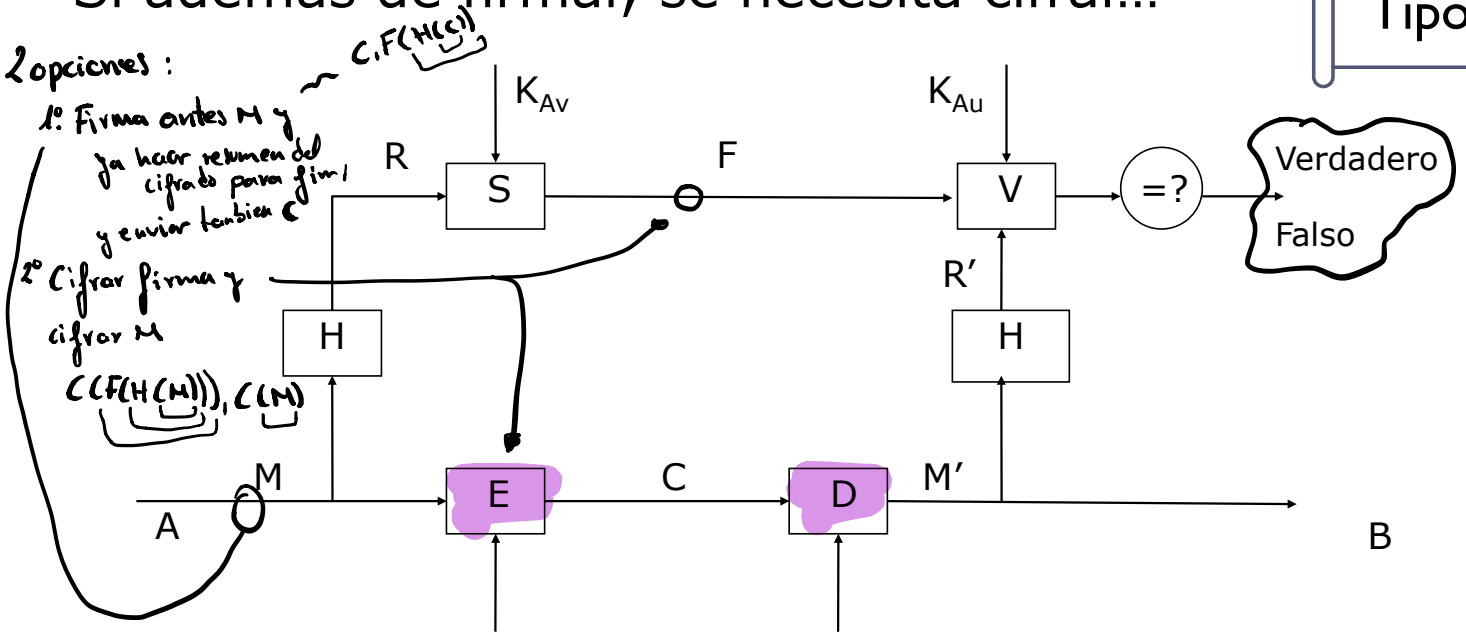


$R=H(M)$ $F=S(K_{Av}, R)$

Mensaje cifrado con firma separada

Si además de firmar, se necesita cifrar…

Tipo I



$C=E(K_{Bu}, M)$ $R=H(M)$ $F=S(K_{Av}, R)$

Firma digital: Con recuperación del mensaje

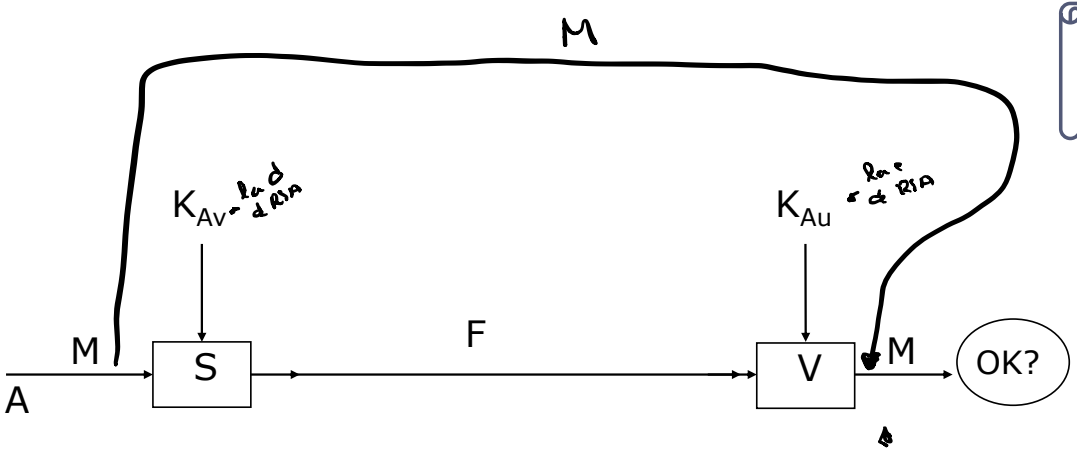
- **TIPO II:** La firma está integrada en el propio mensaje transformado. Se denomina **firma con recuperación del mensaje** a partir de la firma (se envía F únicamente)
  - RSA
  - El mensaje original se recupera durante el proceso de verificación de la firma

Firma con recuperación del mensaje

Tipo II

- Protocolo de firma digital con recuperación del mensaje M:
  1. Obtener la firma F=S(M) (obtenida con la clave privada del remitente K_{Av})
  2. Enviar F
  3. El receptor calcula M a partir de F utilizando el algoritmo de verificación de firma V y la clave pública del remitente K_{Au} . Se acepta el mensaje si el resultado del algoritmo es correcto (el mensaje obtenido pertenece al espacio de mensajes permitidos)

Firma con recuperación del mensaje



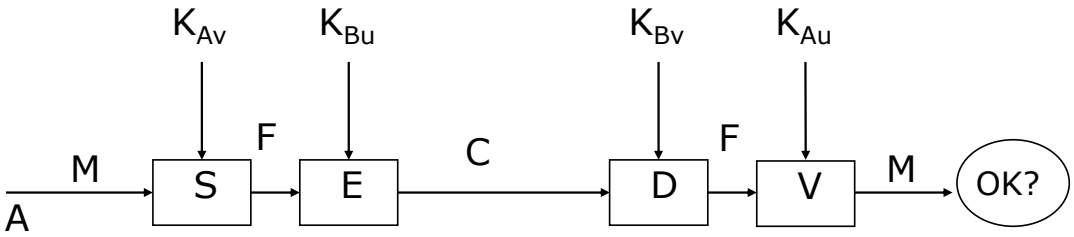
Tipo II

$$F = S(K_{Av}, M)$$

Mensaje cifrado y firmado con esquema de firma con recuperación del mensaje

Si además de firmar, se necesita cifrar…

Tipo II



$$C = E(K_{Bu}, F) = E(K_{Bu}, S(K_{Av}, M))$$

Firma con recuperación del mensaje. Desventajas

Tipo II

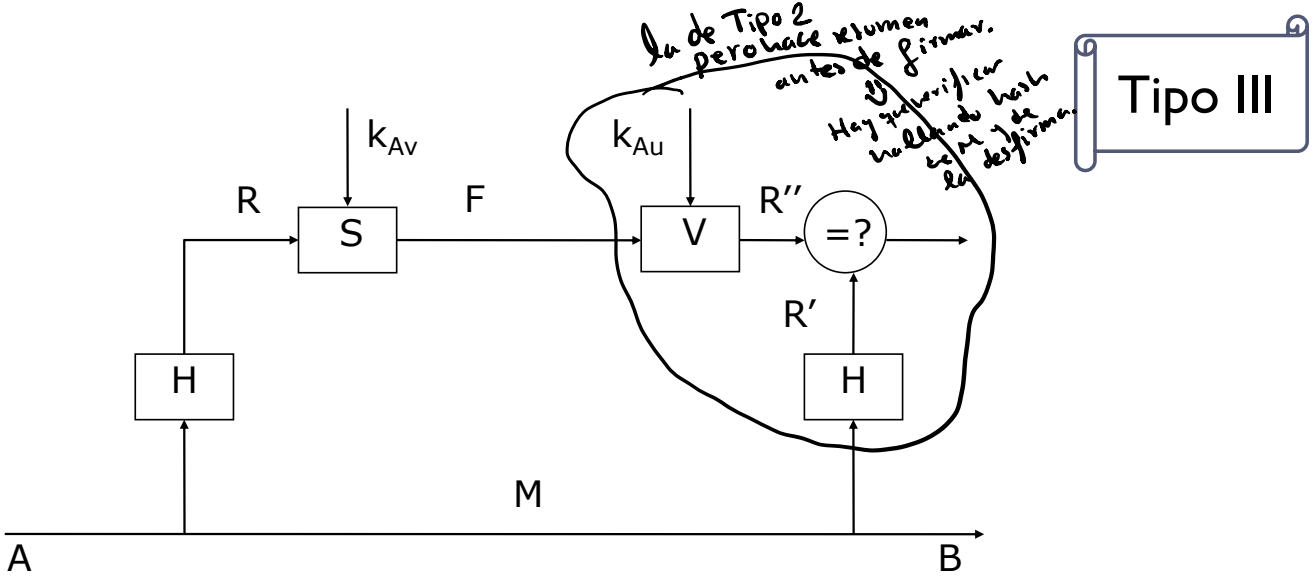
- Las firmas con recuperación del mensaje tienen el inconveniente de tener que cifrar (dos veces si se desea garantizar secreto y autenticidad) todo el mensaje a autenticar, el cual puede ser muy largo, con clave pública (que es muy lenta)
- Hay que dividir el mensaje bloques tal que 0 < M < n y firmar cada uno de ellos
  - Firma = concatenación de firmas
- No existe conexión entre los fragmentos
  - El receptor no puede comprobar si han llegado todos y en orden correcto
- Hay que aplicar el algoritmo de firma n veces
  - n * lento = poco eficiente

Firma con recuperación del mensaje combinada con función resumen

- **TIPO III: Esquema de firma con recuperación del mensaje transformado en esquema de firma separada con ayuda de una función resumen**

1. Obtener el resumen R = H(M) del mensaje M
2. Firmar R con la clave privada del emisor $F = S(R, K_{Av})$
3. A envía a B: (M, F)
4. El receptor calcula el resumen del mensaje por dos caminos distintos:
   1. R' = H(M)
   2. R'' = $V(K_{Au}, F)$
5. El receptor compara R' con R'', aceptando el mensaje si coinciden y rechazándolo en caso contrario

Firma con recuperación del mensaje combinada con función resumen



$R=H(M)$ $F=S(k_{Av}, R)$

Firma opaca

- Permiten a una entidad A conseguir que otra B (cierta autoridad) firme un mensaje M sin que en el proceso B pueda conocer el contenido de M
- Pasos
  - A envía el mensaje M, encubierto, al Notario N, quien lo firma y remite a A
  - A invierte el encubrimiento y dispone del mensaje M firmado
  - El notario no ha conocido M
  - El encubrimiento debe ser compatible con el algoritmo de firma

Tipos de ataques

- El objetivo para un atacante a un proceso de firma digital es crear firmas que sean aceptadas como válidas.
  - Rotura total: El atacante posee un algoritmo de firma funcionalmente equivalente al auténtico.
  - Rotura selectiva: El atacante es capaz de forjar una firma para un tipo particular de mensaje.
  - Rotura existencial: El atacante es capaz de forjar una firma para al menos un mensaje.

Algoritmos

- ElGamal (DSA, DSS)
  - Tipo I
- RSA
  - Tipo II
  - Tipo III

Firma digital ElGamal

Sobre M se hace una función resumen que nos muestra. Separado F y M

- El NIST (*National Institute of Standards and Technology)* propone en 1991 el DSA (*Digital Signature Algorithm),* una variante de los algoritmos de ElGamal y Schnoor (problemas de autoría). En 1994 aparece el DSS (*Digital Signature Standard)* nombre dado por el NIST a su primer algoritmo DSA
- Esquema de firma digital aleatorio y con apéndice (separada del mensaje) (TIPO I):
- Inicialización:
  - Se ha elegido un primo adecuado p (con p ~ 200 bits) y un elemento primitivo g (generador de CG(p))
  - El firmante elige una clave secreta x_A, $1 < x_A < p - 1$ y hace pública $y_A \equiv g^{XA}$ (mód. p)

Firma digital ElGamal

- **Creación de la firma por A**
  - A, con claves x_A e $y_A = g^{XA}$ mód. p, para firmar M (ver Nota) elige un entero k (coprimo con p-1) y calcula la firma, el par (r,s):
    - $r = g^k$ (mód. p)
    - $s = (M - x_A . r) \cdot k^{-1}$ mód. (p -1)

$M = x_A \cdot r + k \cdot s$ (mód $p-1$)

- **A envía a B: (M, r, s)**
- **Verificación de la firma por B**
  - B, recibidos M y (r, s), acepta la firma si coinciden las dos expresiones:
    - $V_1 = y_A{}^r . r^s$ (mód. p)
    - $V_2 = g^M$ (mód. p)

Debe conocer y_A, g, p

Nota: Trabajaremos directamente sobre M, aunque en realidad el algoritmo se aplicaría sobre H(M)

Ejemplo Firma digital ElGamal

- **Calcular y verificar la firma, mediante El Gamal, del mensaje M=5, con g=2, p=11, x_A=8, y k=9.**
- Primero se comprueba:
  - g es raíz primitiva de CG(p)
  - $1<x_A<p-1 \rightarrow 1<8<10$
  - $1<k<p-1 \rightarrow 1<9<10$
- $M = x_A \cdot r + k \cdot s \ (\text{mód } p-1)$
  - $r = g^k \ (\text{mód. } p) = 2^9 \bmod 11 = 6$
  - $s=(M - x_A . r) \cdot k^{-1} \ \text{mód. } (p-1)$
    - $k^{-1} \bmod (p-1) = 9^{-1} \bmod 10 = 9$
    - $s= (5 - 8 \cdot 6) \cdot 9 \bmod 10 = \mathbf{3} \text{ mód } 10$
  - El emisor envía entonces (M, r, s): (5,6,3)

Ojo, el ejemplo no considera la aplicación de la función resumen sobre el mensaje

Ejemplo Firma digital ElGamal (cont.)

- Verificamos la firma:
  - $V_1 = y_A^r \cdot r^s \pmod p$;
  - $y_A = g^{xA} \pmod p = 2^8 \pmod{11} = 3 \bmod 11$
  - $V_1 = y_A^r \cdot r^s \pmod p = 3^6 \cdot 6^3 \pmod{11} = 32 \bmod 11 = 10$

  - $V_2 = g^M \pmod p = 2^5 \pmod{11} = 10$

Como V_1 y V_2 coinciden, la firma es válida.

Firma digital con RSA (tipo II)

- Esquema de firma digital determinista y con recuperación del mensaje (TIPO II):
  - Alicia quiere mandar un mensaje firmado a Benito:
    1. Alicia crea una clave pública e (o K_{Au}) y una privada d (o K_{Av})
    2. **Generación** de la firma :
       1. Alicia cifra el mensaje con su clave privada $F=D_{RSA}(M, d)$
    3. Alicia envía F a Benito
    4. **Verificación** de la firma:
       1. Benito obtiene el mensaje utilizando la clave pública de Alicia $M=E_{RSA}(M, e)$

La firma es cifrar con su privada

Para que pueda descifrar la hace con la pública

Ejemplo Firma digital con RSA

Sea un sistema RSA con p= 13 y q=19, donde se desea firmar digitalmente el mensaje M=10. Supóngase e= 11.

- Comprobación inicial:
  - $N=p \cdot q=13 \cdot 19=247$
  - $\phi(N)=12 \cdot 18=216$
  - $1<e<N \rightarrow 1<11<247$ Cierto
  - Mcd $(e , \phi(N)) =1 \rightarrow$ Mcd (11, 216)=1 Cierto
  - p y q números primos grandes Falso, pero nos vale para el ejercicio

- Primero necesitamos calcular la clave privada para poder firmar:
  - $d \cdot e = 1$ mód $(\phi(N))$
  - $11 \cdot d = 1$ mód (216) -> por euclides modif. -> d= 59

Ejemplo Firma digital con RSA (cont.)

- Firma
  - $F=S(M)=M^d$ (mód. N)$=10^{59}$ (mód. 247)=212 (mód. 247)

- Verificación:
  - $M=V(F)=F^e$ (mód. N)$=212^{11}$(mód. 247) =10 (mód. 247)

Firma digital con RSA (tipo III)

- En la práctica se usa siempre transformado en firma digital con apéndice mediante la aplicación de una función resumen inicial al mensaje (TIPO III)
  1. A genera la firma
     1. A obtiene el resumen R = H(M) del mensaje M
     2. A firma con su clave privada el resumen: $F = D_{RSA}(R, K_{vA})$
  2. A envía a B: (M, F)
  3. B verifica la firma
     1. Obtiene R" aplicando el algoritmo de cifrado E_{RSA} sobre F con la clave pública K_{Au}: $R'' = E_{RSA}(K_{Au}, F)$
     2. A partir del M recibido, calcula el resumen de nuevo R'
     3. Compara R' con R"

Mensaje cifrado y firmado con RSA (tipo III)

- Para mantener la confidencialidad (envío del mensaje cifrado):
  1. A obtiene el resumen R = H(M) del mensaje M
  2. A firma con su clave privada el resumen: $F = D_{RSA}(K_{Av}, R)$
  3. A cifra M con la clave pública del receptor B: $C = E_{RSA}(K_{uB}, M)$
  4. A envía a B: $(C, F) = (E_{RSA}(K_{uB}, M), D_{RSA}(K_{Av}, R))$
  5. B descifra el mensaje y verifica la firma
     1. Descifrando C con su clave ~~pública~~ privada, por lo que obtiene M
     2. Aplica el algoritmo de cifrado de E_{RSA} sobre F con la clave pública del emisor, por lo que obtiene R' (verificar)
     3. A partir del M obtenido al descifrar C, calcula de nuevo el resumen R"
     4. Compara R' con R"

Formatos de firmas digitales

- Existen diferentes formatos para almacenar una firma digital.
- La mayoría encapsula en un sobre los datos, la identidad del firmante y la firma.
- Algunos ejemplos de formatos son:
  - Privacy-Enhanced Mail (PEM) – RFC 1421
  - PKCS#7
  - S/MIME – RFC 2634
  - ISO 9796-2
  - XMLDsig
  - XAdES

Formato de Sobre PKCS#7

Esquema que define como debe estar estructurados los datos.

- PKCS #7: Cryptographic Message Syntax Standard
  - Estándar desarrollado por RSA Laboratories Inc.
  - Define varios formatos de mensajes:
    - Data, EnvelopedData, SignedData, etc.

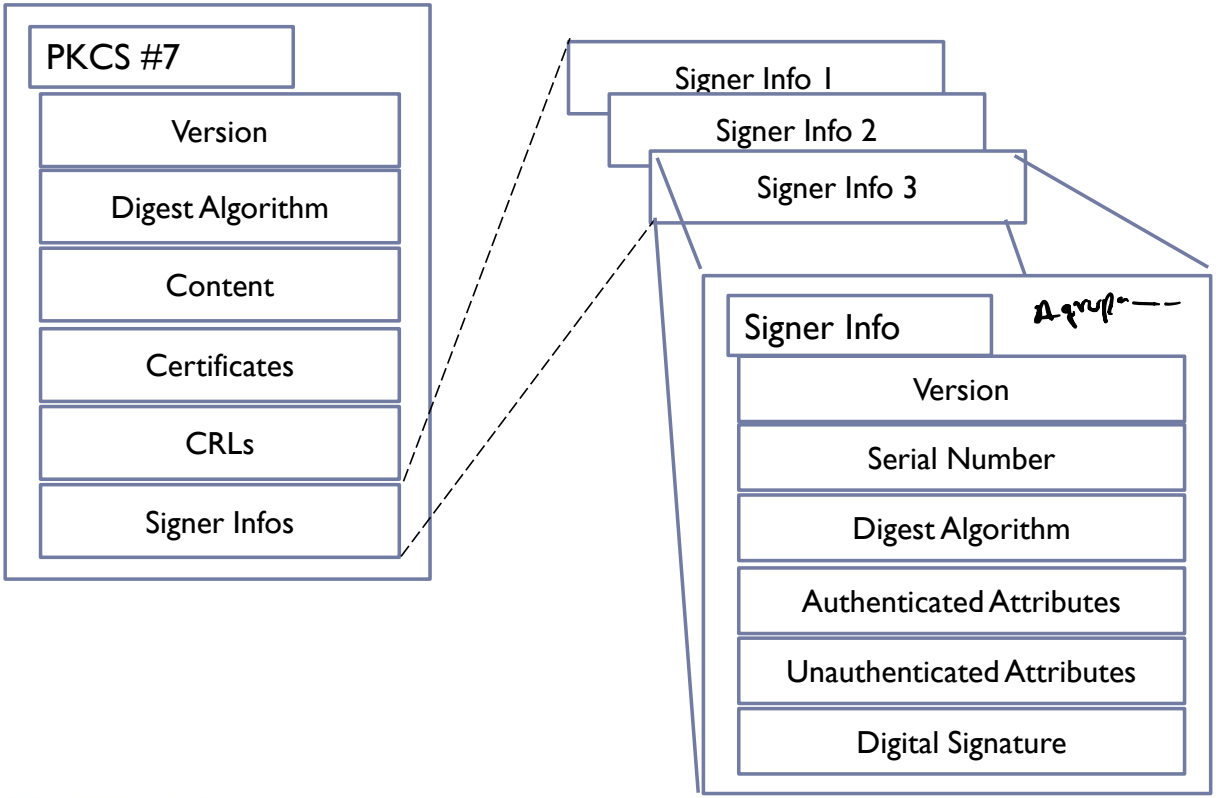
Solo datos sin encript

Datos para cifrados, firmar (confidencial)

Para autenticar

```
SignedData ::= SEQUENCE {
  version            Version,
  digestAlgorithms   DigestAlgorithmIdentifiers,
  contentInfo        ContentInfo,
  certificates       [0] IMPLICIT ExtendedCertificatesAndCertificates OPTIONAL,
  crls               [1] IMPLICIT CertificateRevocationLists OPTIONAL,
  signerInfos        SignerInfos }
```

PKCS#7 SignedData



Firma Digital XML

Permite la firma completa o parcial utilizando el lenguaje XML

- XMLDSIG: Propuesta conjunta de estándar IETF/W3C
  - RFC 3075 y http://www.w3.org/Signature/
- Define un mecanismo para firmar:
  - Documentos XML
  - Fragmentos de un documento XML
- Proporciona tres métodos de firma: → contenido dentro de la firma
  - "Wrapped", el formato de firma incluye el contenido. Contiene los datos
  - "Detached", la firma está separada del contenido. Esta separada de XML
  - "Embedded", la firma es parte del contenido firmado
- El formato XMLDsig no requiere infraestructura de certificados
  - Se incluye información de la clave
  - Se reserva espacio para proporcionar información del certificado

Formato de Firma XMLDsig

```
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
<element name="Signature" type="ds:SignatureType"/>
<complexType name="SignatureType">
    <sequence>
        <element ref="ds:SignedInfo"/>
        <element ref="ds:SignatureValue"/>
        <element ref="ds:KeyInfo" minOccurs="0"/>
        <element ref="ds:Object" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
    <attribute name="Id" type="ID" use="optional"/>
</complexType>
```

```
<Signature xmlns="http://www.w3.org/2000/09/xmldsig#">

  <SignedInfo>
    <CanonicalizationMethod Algorithm="http://www.w3.org/TR/2000/..." />
    <SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
    <Reference URI="#PurchaseOrder">
      <DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
      <DigestValue>qZk+nkcGcWq6p1VxeFdcbJzQ2JO=</DigestValue>
    </Reference>
  </SignedInfo>

  <SignatureValue>
   IW1jxQjUrcXBYc0el4QxjWo9Kg8Dep9tlWoT4SdeRT87GR03dgh
  </SignatureValue>

  <KeyInfo>
     <X509Data>
       <X509SubjectName>CN=Alice Smith, STREET=742 Park Avenue,
         L=New York, ST=NY, C=US</X509SubjectName>
     </X509Data>
  </KeyInfo>

</Signature>
```

```
<X509Certificate>
      MIID5jCCA0+gA...lVN
</X509Certificate>
```

Universidad
Carlos III de Madrid

COSEC Lab · *Dpto. Informática*

ANEXO

Criptografía y seguridad informática

Seguridad en las tecnologías de la información

@ COSEC

Curso 2016-2017