MOOC de Criptología Matemática. Firmas Digitales

Leandro Marín

Módulo II. Sesión 5.
Dificultad Alta

Introducción

- Una de las principales aplicaciones de la criptografía de clave pública es la posibilidad de generar firmas digitales.

Introducción

- Una de las principales aplicaciones de la criptografía de clave pública es la posibilidad de generar firmas digitales.
- Una firma digital es una información asociada a un mensaje que garantiza la autoría del mismo.

Introducción

- Una de las principales aplicaciones de la criptografía de clave pública es la posibilidad de generar firmas digitales.
- Una firma digital es una información asociada a un mensaje que garantiza la autoría del mismo.
- Esa autoría debe poder comprobarse por el receptor del mensaje de forma inequívoca.

Introducción

- Una de las principales aplicaciones de la criptografía de clave pública es la posibilidad de generar firmas digitales.
- Una firma digital es una información asociada a un mensaje que garantiza la autoría del mismo.
- Esa autoría debe poder comprobarse por el receptor del mensaje de forma inequívoca.
- Vamos a basar este tema en dos de los estándares que aparecen el el documento FIPS PUB 186-4 que es accesible en `http://csrc.nist.gov/publications/fips/fips186-4/fips-186-4.pdf`

Parámetros DSA

- El DSA (Digital Signature Algorithm) necesita los siguientes parámetros:

Parámetros DSA

- El DSA (Digital Signature Algorithm) necesita los siguientes parámetros:
  - Un número primo p de L bits.

Parámetros DSA

- El DSA (Digital Signature Algorithm) necesita los siguientes parámetros:
  - Un número primo p de L bits.
  - Un número primo q que divida a $p-1$ de N bits.

Parámetros DSA

- El DSA (Digital Signature Algorithm) necesita los siguientes parámetros:
  - Un número primo p de L bits.
  - Un número primo q que divida a $p-1$ de N bits.
  - Un elemento $g \in \mathbb{Z}_p$ distinto de 1 tal que $g^q = 1(mod\ p)$.

Parámetros DSA

- El DSA (Digital Signature Algorithm) necesita los siguientes parámetros:
  - Un número primo p de L bits.
  - Un número primo q que divida a $p-1$ de N bits.
  - Un elemento $g \in \mathbb{Z}_p$ distinto de 1 tal que $g^q = 1(mod\ p)$.
  - Un número aleatorio x en el intervalo $[1, q-1]$. Este valor es la clave privada.

Parámetros DSA

- El DSA (Digital Signature Algorithm) necesita los siguientes parámetros:
  - Un número primo p de L bits.
  - Un número primo q que divida a $p-1$ de N bits.
  - Un elemento $g \in \mathbb{Z}_p$ distinto de 1 tal que $g^q = 1(mod\ \ p)$.
  - Un número aleatorio x en el intervalo $[1, q-1]$. Este valor es la clave privada.
  - El valor $y = g^x\ (mod\ \ p)$ será la clave pública.

Parámetros DSA

- El DSA (Digital Signature Algorithm) necesita los siguientes parámetros:
  - Un número primo p de L bits.
  - Un número primo q que divida a $p-1$ de N bits.
  - Un elemento $g \in \mathbb{Z}_p$ distinto de 1 tal que $g^q = 1(mod\ p)$.
  - Un número aleatorio x en el intervalo $[1, q-1]$. Este valor es la clave privada.
  - El valor $y = g^x\ (mod\ p)$ será la clave pública.
  - Un elemento k aleatorio en el intervalo $[1, q-1]$ dependiente de cada mensaje.

Parámetros DSA

- El DSA (Digital Signature Algorithm) necesita los siguientes parámetros:
  - Un número primo p de L bits.
  - Un número primo q que divida a $p-1$ de N bits.
  - Un elemento $g \in \mathbb{Z}_p$ distinto de 1 tal que $g^q = 1(mod\ \ p)$.
  - Un número aleatorio x en el intervalo $[1, q-1]$. Este valor es la clave privada.
  - El valor $y = g^x\ (mod\ \ p)$ será la clave pública.
  - Un elemento k aleatorio en el intervalo $[1, q-1]$ dependiente de cada mensaje.
- Los tamaños adecuados para L y N podrían ser, por ejemplo, $L = 1024$ y $N = 160$.

Método de Generación de Claves

- Para generar valores cumpliendo estas propiedades se empieza buscando el primo q del tamaño adecuado.

Método de Generación de Claves

- Para generar valores cumpliendo estas propiedades se empieza buscando el primo q del tamaño adecuado.
- Se calculan valores $p = qn + 1$ para valores de n que nos den el tamaño adecuado hasta que p sea primo.

Método de Generación de Claves

- Para generar valores cumpliendo estas propiedades se empieza buscando el primo q del tamaño adecuado.
- Se calculan valores $p = qn + 1$ para valores de n que nos den el tamaño adecuado hasta que p sea primo.
- Se toman valores h y si $h^n (mod\ p)$ no es 1, este valore h^n puede tomarse como g.

Método de Generación de Claves

- Para generar valores cumpliendo estas propiedades se empieza buscando el primo q del tamaño adecuado.
- Se calculan valores $p = qn + 1$ para valores de n que nos den el tamaño adecuado hasta que p sea primo.
- Se toman valores h y si $h^n (mod\ p)$ no es 1, este valore h^n puede tomarse como g.
- El estaándar nos determina formas para generar todos estos parámetros de forma aleatoria.

Generación de Firma DSA

- Tomaremos como función de resumen digital la función sha1 que vimos anteriormente. Calcularemos los siguiente valores:
- $r = (g^k \pmod{p}) \pmod{q}$

Generación de Firma DSA

- Tomaremos como función de resumen digital la función sha1 que vimos anteriormente. Calcularemos los siguiente valores:
- $r = (g^k \ (mod \ p)) \ (mod \ q)$
- $z = sha1(M)$

Generación de Firma DSA

- Tomaremos como función de resumen digital la función sha1 que vimos anteriormente. Calcularemos los siguiente valores:
- $r = (g^k \ (mod \ p)) \ (mod \ q)$
- $z = sha1(M)$
- $s = (k^{-1}(z + xr)) \ (mod \ q)$

Generación de Firma DSA

- Tomaremos como función de resumen digital la función sha1 que vimos anteriormente. Calcularemos los siguiente valores:
- $r = (g^k \ (mod \ p)) \ (mod \ q)$
- $z = sha1(M)$
- $s = (k^{-1}(z + xr)) \ (mod \ q)$
- Se debe comprobar que r y s no son 0. Si fueran cero, se generaría un nuevo valor de k y se recalcularían de nuevo r y s.

Generación de Firma DSA

- Tomaremos como función de resumen digital la función sha1 que vimos anteriormente. Calcularemos los siguiente valores:
- $r = (g^k \ (mod \ p)) \ (mod \ q)$
- $z = sha1(M)$
- $s = (k^{-1}(z + xr)) \ (mod \ q)$
- Se debe comprobar que r y s no son 0. Si fueran cero, se generaría un nuevo valor de k y se recalcularían de nuevo r y s.
- La firma digital es el par de números (r, s).

Verificación de Firma DSA

- Supongamos que tenemos el mensaje M y la firma (r, s) que queremos comprobar si es correcta. Si $r = 0$ ó $s = 0$ la daremos como incorrecta. Si son distintos de cero haremos los siguientes cálculos:

Verificación de Firma DSA

- Supongamos que tenemos el mensaje M y la firma (r, s) que queremos comprobar si es correcta. Si $r = 0$ ó $s = 0$ la daremos como incorrecta. Si son distintos de cero haremos los siguientes cálculos:
  - $w = s^{-1} \ (mod \ q)$

Verificación de Firma DSA

- Supongamos que tenemos el mensaje M y la firma (r, s) que queremos comprobar si es correcta. Si $r = 0$ ó $s = 0$ la daremos como incorrecta. Si son distintos de cero haremos los siguientes cálculos:
  - $w = s^{-1} \ (mod \ q)$
  - $z = sha1(M)$

Verificación de Firma DSA

- Supongamos que tenemos el mensaje M y la firma (r, s) que queremos comprobar si es correcta. Si $r = 0$ ó $s = 0$ la daremos como incorrecta. Si son distintos de cero haremos los siguientes cálculos:
  - $w = s^{-1} \ (mod \ q)$
  - $z = sha1(M)$
  - $u_1 = zw \ (mod \ q)$

Verificación de Firma DSA

- Supongamos que tenemos el mensaje M y la firma (r, s) que queremos comprobar si es correcta. Si $r = 0$ ó $s = 0$ la daremos como incorrecta. Si son distintos de cero haremos los siguientes cálculos:
  - $w = s^{-1} \ (mod \ q)$
  - $z = sha1(M)$
  - $u_1 = zw \ (mod \ q)$
  - $u_2 = rw \ (mod \ q)$

Verificación de Firma DSA

- Supongamos que tenemos el mensaje M y la firma (r, s) que queremos comprobar si es correcta. Si $r = 0$ ó $s = 0$ la daremos como incorrecta. Si son distintos de cero haremos los siguientes cálculos:
  - $w = s^{-1} \ (mod \ q)$
  - $z = sha1(M)$
  - $u_1 = zw \ (mod \ q)$
  - $u_2 = rw \ (mod \ q)$
  - $(v = g^{u_1} y^{u_2} \ (mod \ p)) \ (mod \ q)$

Verificación de Firma DSA

- Supongamos que tenemos el mensaje M y la firma (r, s) que queremos comprobar si es correcta. Si $r = 0$ ó $s = 0$ la daremos como incorrecta. Si son distintos de cero haremos los siguientes cálculos:
  - $w = s^{-1} \pmod q$
  - $z = sha1(M)$
  - $u_1 = zw \pmod q$
  - $u_2 = rw \pmod q$
  - $(v = g^{u_1} y^{u_2} \pmod p) \pmod q$
- Si el valor v coincide con r entonces la firma será correcta, si no lo es la firma es incorrecta.

Verificación del Algoritmo

Se puede ver la demostración de la validez del algoritmo en el Apéndice E del documento FIPS PUB 184-4 que hemos referenciado al inicio.

El Estándar RSAVP1

- Existen varias formas de generar firmas digitales usando RSA.

El Estándar RSAVP1

- Existen varias formas de generar firmas digitales usando RSA.
- En el *Public-Key Cryptography Standard* PKCS #1, *RSA Cryptography Standard v2.2* podemos encontrar varios de ellos.

El Estándar RSAVP1

- Existen varias formas de generar firmas digitales usando RSA.
- En el *Public-Key Cryptography Standard* PKCS #1, *RSA Cryptography Standard v2.2* podemos encontrar varios de ellos.
- Nosotros nos centraremos en uno de los más simples, el RSAVP1.

Generación de Firma

- Supongamos que tenemos un mensaje M y nuestra clave privada (n, d).

Generación de Firma

- Supongamos que tenemos un mensaje M y nuestra clave privada (n, d).
- Generaremos un valor m que represente al mensaje. Este valor m debe estar entre 0 y $n-1$ y se puede generar mediante una función de resumen digital junto con alguna información adicional relativa al mensaje que deseemos introducir.

Generación de Firma

- Supongamos que tenemos un mensaje M y nuestra clave privada (n, d).
- Generaremos un valor m que represente al mensaje. Este valor m debe estar entre 0 y $n - 1$ y se puede generar mediante una función de resumen digital junto con alguna información adicional relativa al mensaje que deseemos introducir.
- La firma digital será el valor $s = m^d \pmod{n}$ calculado mediante la exponenciación modular.

Verificación de Firma

- Supongamos que recibimos un mensaje M procedente de un usuario cuya clave pública es (n, e).

Verificación de Firma

- Supongamos que recibimos un mensaje M procedente de un usuario cuya clave pública es (n, e).
- Generamos el valor m que representa al mensaje según el método acordado.

Verificación de Firma

- Supongamos que recibimos un mensaje M procedente de un usuario cuya clave pública es (n, e).
- Generamos el valor m que representa al mensaje según el método acordado.
- Calculamos $m' = s^e \ (mod \ n)$.

Verificación de Firma

- Supongamos que recibimos un mensaje M procedente de un usuario cuya clave pública es (n, e).
- Generamos el valor m que representa al mensaje según el método acordado.
- Calculamos $m' = s^e \ (mod \ n)$.
- Si $m' = m$ entonces la firma es válida.

Verificación de Firma

- Supongamos que recibimos un mensaje M procedente de un usuario cuya clave pública es (n, e).
- Generamos el valor m que representa al mensaje según el método acordado.
- Calculamos $m' = s^e \ (mod \ n)$.
- Si $m' = m$ entonces la firma es válida.
- Si no lo es, la firma no es correcta.

Verificación del Método

- Al ser (n, e, d) claves RSA, sabemos que $ed = 1 + \varphi(n)t$ para algún $t \in \mathbb{Z}$.
- Si $s = m^d \ (mod \ n)$ y calculamos $m' = s^e \ (mod \ n)$ obtenemos

$$m' = s^e = m^{ed} = m^{1+\varphi(n)t} = m\underbrace{\left(m^{\varphi(n)}\right)}_{1}^{t} = m \ (mod \ n)$$

Verificación del Método

- Al ser (n, e, d) claves RSA, sabemos que $ed = 1 + \varphi(n)t$ para algún $t \in \mathbb{Z}$.
- Si $s = m^d \ (mod \ n)$ y calculamos $m' = s^e \ (mod \ n)$ obtenemos

$$m' = s^e = m^{ed} = m^{1+\varphi(n)t} = m\underbrace{\left(m^{\varphi(n)}\right)^t}_{1} = m \ (mod \ n)$$

- Por lo tanto el valor m' y m coincidirían y la firma sería correcta.

Verificación del Método

- Al ser (n, e, d) claves RSA, sabemos que $ed = 1 + \varphi(n)t$ para algún $t \in \mathbb{Z}$.
- Si $s = m^d \ (mod \ n)$ y calculamos $m' = s^e \ (mod \ n)$ obtenemos

$$m' = s^e = m^{ed} = m^{1+\varphi(n)t} = m\underbrace{\left(m^{\varphi(n)}\right)^t}_{1} = m \ (mod \ n)$$

- Por lo tanto el valor m' y m coincidirían y la firma sería correcta.
- El único usuario capaz de generar la firma sería el que conociera la clave privada d, por lo tanto nos garantiza la autenticidad del mensaje.

Verificación del Método

- Al ser (n, e, d) claves RSA, sabemos que $ed = 1 + \varphi(n)t$ para algún $t \in \mathbb{Z}$.
- Si $s = m^d \pmod{n}$ y calculamos $m' = s^e \pmod{n}$ obtenemos

$$m' = s^e = m^{ed} = m^{1+\varphi(n)t} = m\underbrace{\left(m^{\varphi(n)}\right)}_{1}^{t} = m \pmod{n}$$

- Por lo tanto el valor m' y m coincidirían y la firma sería correcta.
- El único usuario capaz de generar la firma sería el que conociera la clave privada d, por lo tanto nos garantiza la autenticidad del mensaje.
- El receptor puede verificar la firma puesto que sólo utiliza información pública del emisor (los valores (n, e)).