CÓDIGOS Y CRIPTOGRAFÍA. INFORMÁTICA DIEGO RUANO 21/11 2022

FIRMA DIGITAL CON ELGAMAL - REPETICIÓN

- SE GENERAN CLAVES COMO EN EL CRIPTOSISTEMA DE CLAVE PÚBLICA DE EL GAMAL
  - $a \in \{2, \dots, p-2\}$ PRIVADA
  - $(p, g, A = g^a \bmod p)$ PÚBLICA
- SE USA UNA FUNCIÓN HASH

$$h : \{0,1\}^t \to \{1, 2, \dots, p-2\}$$

- GENERACIÓN DE LA FIRMA
  - ALICE ESCOGE AL AZAR $k \in \{1, \dots, p-2\}$ TAL QUE $mcd(k, p-1) = 1$ ⊛
  - ALICE CALCULA PARA UN MENSAJE M

$$r = g^k \bmod p$$

OK POR ⊛

$$s = k^{-1}(h(M) - ar) \bmod (p-1)$$

$\mathbb{Z}_p$

$a^x = a^{x+\lambda(p-1)}$

OJO EXPONENTE

- LA FIRMA DE M ES (r, s)

- VERIFICACIÓN DE LA FIRMA

- BOB PRIMERO COMPRUEBA QUE

$$1 \leq r \leq p-1$$

SI NO ES CIERTO, RECHAZA LA FIRMA

- BOB COMPRUEBA QUE

$$\boxed{A^r r^s \equiv g^{h(M)} \bmod p}$$

LA ACEPTA SI ES CIERTO Y LA RECHAZA EN CASO CONTRARIO

HOY VEREMOS ASPECTOS DE LA SEGURIDAD

← NUEVO, NO DIO TIEMPO EL JUEVES

EJ: COMO EN EL EJEMPLO DE EL GAMAL QUE VIMOS EL MARTES ALICE ESCOGE

$p=23$, $g=7$, $a=6$ Y CALCULA $A = g^a \bmod p$

$= 7^6 \bmod p = 4$

CLAVE PÚBLICA: $(p=23, g=7, A=20)$

CLAVE PRIVADA: $a=13$

ALICE QUIERE FIRMAR M, CON $h(M)=7$.

ALICE ESCOGE $k=$ Y OBTIENE

$$r = g^k \bmod p =$$

$$k^{-1} \bmod p-1 =$$

POR TANTO

$$s = k^{-1}(h(x) - a \cdot r) \bmod p-1 =$$

$$=$$

LA FIRMA ES (,)

- SI BOB QUIERE COMPROBAR LA FIRMA.

CALCULA

$$A^{r} r^{s} \bmod p =$$

$$g^{h(M)} \bmod p =$$

OK

ELECCIÓN DE K ← LO NUEVO DE TEORÍA EMPIEZA AQUÍ

PARA CADA FIRMA, K DEBE SER ESCOGIDO DE FORMA ALEATORIA

SI SIEMPRE USAMOS EL MISMO K:

$$r = g^k \bmod p$$ ES EL MISMO

SI DOS FIRMAS (r_1, s_1) Y (r_2, s_2) TIENEN QUE $r_1 = r_2 = r$ ENTONCES

$$\left.\begin{aligned} s_1 &\equiv k^{-1}(h(M_1) - ar) \bmod p-1 \\ s_2 &\equiv k^{-1}(h(M_2) - ar) \bmod p-1 \end{aligned}\right\} \text{RESTANDO} \Rightarrow$$

"DESAPARECE" ar

$$s_1 - s_2 = k^{-1}(h(M_1) - h(M_2)) \bmod p-1$$

SI $S_1 - S_2$ ES INVERTIBLE $\Rightarrow$

$$K = (S_1 - S_2)^{-1}(h(M_1) - h(M_2)) \mod p-1$$

Y PODEMOS OBTENER K

DE K, S_1, r, $h(M_1)$ EVE PUEDE CALCULAR

LA CLAVE PRIVADA a PORQUE

$$S_1 \equiv K^{-1}(h(M_1) - ar) \mod p-1$$

$\times K$ $\quad S_1 K \equiv h(M_1) - ar \mod p-1$

$\quad ar \equiv h(M_1) - KS_1 \mod p-1$

QUE IMPLICA QUE

$\div r$ $\quad a \equiv r^{-1}(h(M_1) - KS_1) \mod p-1$

$$\boxed{a \equiv r^{-1}(h(M_1) - KS_1) \mod p-1}$$

SI EVE OBTIENE a, PUEDE FIRMAR CUALQUIER COSA

NECESIDAD DE LA FUNCIÓN HASH

SIN LA FUNCIÓN HASH, ESTE MÉTODO SUFRIRÍA UN ATAQUE DE FALSIFICACIÓN EXISTENCIAL

$$A^r r^s \equiv g^M \bmod p$$

(EN LUGAR DE $A^r r^s \equiv g^{h(M)} \bmod p$)

MOSTRAREMOS QUE SE PUEDEN ESCOGER r, S Y M TAL QUE LA ECUACIÓN ANTERIOR SE VERIFICA

EVE ESCOGE u, v CON $mcd(v, p-1) = 1$ (*)

$$r := g^u A^v \bmod p$$

$$s := -r v^{-1} \ (*) \bmod p-1$$

$$M := s u \bmod p-1$$

Y (r, S) ES UNA FIRMA VÁLIDA DE M PORQUE

AL VERIFICAR BOB, TIENE QUE VER QUE $A^r r^s \equiv g^M \mod p$:

$(a^b)^c = a^{bc}$
$a^b \cdot a^c = a^{b+c}$

$$A^r r^s \equiv A^r \left(g^u A^v\right)^{-r v^{-1}} \equiv \cancel{A^r}\, g^{-u r v^{-1}}\, \cancel{A^{-r \cancel{v v^{-1}}}} =$$

$$\equiv g^{(-r v^{-1}) u} \equiv g^{s u} \equiv g^{M} \mod p$$

POR TANTO PASA LA VERIFICACIÓN

CLAVE PARA EVITAR ESTE ATAQUE: USAR UNA FUNCIÓN HASH DE UNA VÍA ← COMO HEMOS HECHO DESDE EL PRINCIPIO

PELIGROSO: M NO ES ESCOGIDO LIBREMENTE, PERO CASI LO ES PUESTO QUE u y v SON ESCOGIDOS LIBREMENTE Y $M = s \cdot u \mod p-1$

IMPORTANCIA VERIFICAR $1 \leq r \leq p-1$

DE OTRA MANERA SE PUEDEN GENERAR FIRMAS NUEVAS A PARTIR DE FIRMAS VIEJAS

SEA (r, s) LA FIRMA DE M.

SEA M' OTRO DOCUMENTO. PARA FIRMAR M' EVE CALCULA

$$u \equiv h(M') h(M)^{-1} \bmod (p-1)$$

$\Rightarrow h(M') = u\, h(M) \bmod p-1$ (*)

SUPONIENDO QUE $h(M)$ ES INVERTIBLE mod $p-1$

EVE CALCULA

$$s' \equiv s u \bmod (p-1)$$

USANDO EL TEOREMA CHINO DE LOS RESTOS EVE CALCULA r', SOLUCIÓN DE

$$\begin{cases} r' \equiv ru \mod p-1 & \longleftarrow \text{EXPONENTE} \\ r' \equiv r \mod p & \longleftarrow \text{ABAJO} \end{cases}$$

UNA FIRMA DE M' ES (r', s') PORQUE LA VERIFICACIÓN FUNCIONA

$$A^{r'}(r')^{s'} \equiv A^{ru} r^{su} \equiv (g^a)^{ru}(g^k)^{us} \equiv g^{u(ar+ks)} \equiv$$

$$\equiv g^u \cdot (g^a)^r \cdot (g^k)^s \equiv g^u A^r r^s \equiv g^u g^{h(M)} \overset{(*)}{\equiv} g^{h(M')} \mod p$$

PORQUE (r,s) ES UNA FIRMA DE M

PERO $\boxed{r' \geq p}$ POR LO QUE ESTE ATAQUE ¡NO FUNCIONA!

$1 \le r \le p-1$

$r \equiv r' \bmod p$

PERO

$r' \equiv r u \not\equiv r \bmod p-1$ PORQUE

$u \equiv h(M')\, h(M)^{-1} \not\equiv 1 \bmod p-1$

h ES RESISTENTE A COLISIONES

→ TAMPOCO PUEDO HACER FALSIFICACIÓN EXISTENCIAL ESCOGIENDO M' PARA QUE $u=1$

LUEGO $r' \not\equiv r \bmod p-1 \Rightarrow r \neq r' \Rightarrow$

$\Rightarrow \boxed{r' \geq p}$ PORQUE $r < p$

EFICIENCIA

EL GAMAL PRECOMPUTADO (NO DEPENDE MENSAJE)

- ALGORITMO EUCLIDES EXTENDIDO : $k^{-1} \bmod p$
- UNA EXPONENCIACIÓN MODULAR: $r = g^k \bmod p-1$

OJO: DEBEN ALMACENARSE DE FORMA SEGURA

EL GAMAL AL FIRMAR:

- 3 MULTIPLICACIONES MODULARES

EL GAMAL AL COMPROBAR FIRMA:

- 3 EXPONENCIACIONES MODULARES

SE PUEDE HACER DE FORMA MÁS EFICIENTE

↳ DSS

FIRMA DSS: DIGITAL SIGNATURE STANDARD

(NO ENTRA EN EL EXAMEN)

BASADO EN ELGAMAL Y PROPUESTO POR EL NIST

3 PARTES

- GENERACIÓN DE LAS CLAVES
- GENERACIÓN DE LA FIRMA
- VERIFICACIÓN DE LA FIRMA

DSS: GENERACIÓN DE CLAVES

SE ELIGE (L, N) ENTRE $(1024, 160)$, $(2048, 224)$, $(2048, 256)$, $(3072, 256)$

LA DIFERENCIA VIENE DE USAR DOS PRIMOS

EL USUARIO DEBE TENER

- UN PRIMO p DE LONGITUD BINARIA L
- UN PRIMO q, DIVISOR $p-1$, DE LONGITUD N
- $g \in \mathbb{Z}_p$, DE ORDEN MULTIPLICATIVO q
- UNA FUNCIÓN HASH (SHA-2). LONGITUD SALIDA SE TRUNCA A L

LOGARITMO DISCRETO

- CLAVE SECRETA: $x = a$, $1 < x < q$
- CLAVE PÚBLICA: $(p, q, g, y = A)$ CON $y = g^x \bmod p$

DSS: GENERACIÓN DE LA FIRMA

PARA FIRMAR M

1) ELIGE AL AZAR $k \in \mathbb{Z}_q$

2) CALCULA $r = (g^k \bmod p) \bmod q$

3) CALCULA $s = k^{-1}(h(M) + xr) \bmod q$

4) SI $r=0$ O $s=0$ SE ELIGE OTRO k

LA FIRMA DIGITAL ES (r, s)

DSS: VERIFICACIÓN DE FIRMA

PARA VERIFICAR FIRMA (r, s) DEL MENSAJE M:

1) RECHAZA FIRMA SI NO SE CUMPLE

$$0 < r < q \quad \text{y} \quad 0 < s < q$$

2) CALCULA $t = s^{-1} \bmod q$

3) CALCULA $u = h(M)\, t \bmod q$

$$v = r t \bmod q$$

4) CALCULA $r' = (g^u y^v \bmod p) \bmod q$

5) ACEPTA LA FIRMA Y EL MENSAJE SI

$r = r'$, LO RECHAZA SI $r \neq r'$

VENTAJAS

- RAPIDEZ EN GENERACIÓN DE CLAVES Y GENERACIÓN DE FIRMA
(r PUEDE SER PRECALCULADO)

- FIRMA ES CORTA (LONGITUD)
- SÓLO DOS EXPONENCIACIONES MOD p. ADEMÁS p ES MÁS PEQUEÑO

CRITICAS

- NO PUEDE SER USADO PARA DISTRIBUIR CLAVES
- NO ES COMPATIBLE CON OTROS SISTEMAS ESTÁNDARD

SEGURIDAD

BÁSICAMENTE IGUAL QUE EN ELGAMAL:

- NECESIDAD FUNCIÓN HASH
- NECESIDAD ELECCIÓN K ALEATORIO
- SE ROMPE SI EVE PUDIERA CALCULAR LOGARITMOS DISCRETOS DE FORMA EFICIENTE MOD P

↳ ÚNICO "PROBLEMA": P ES MÁS PEQUEÑO