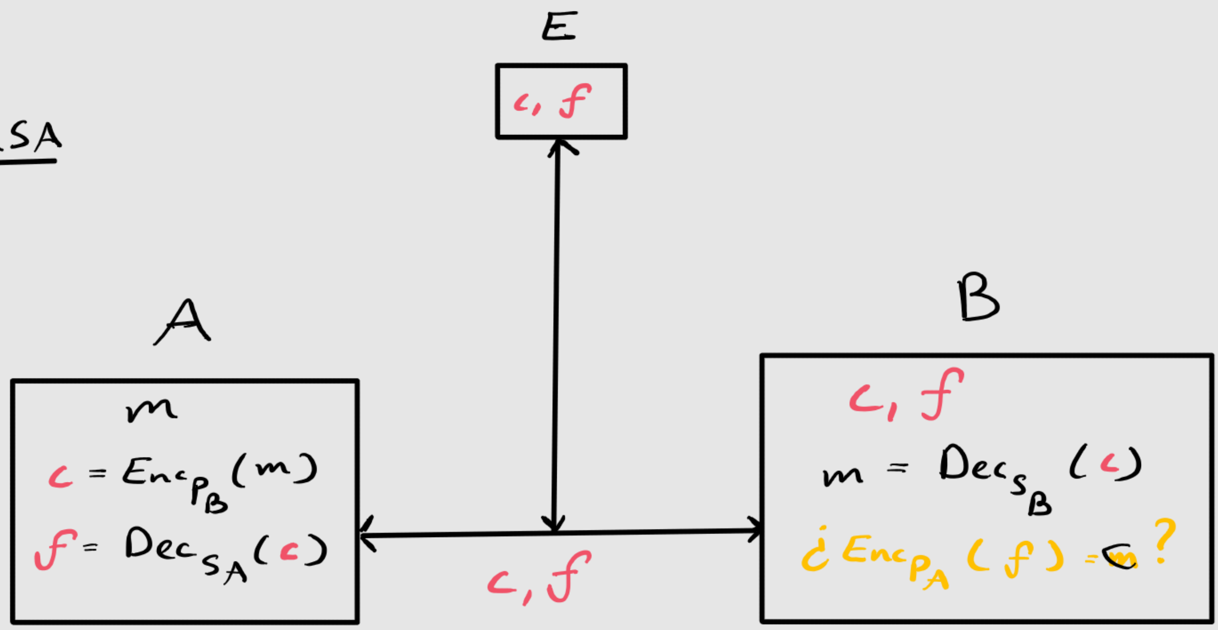
Repaso

Firmas con RSA



Firma de Schnorr

A: $P_A = x \in \{1, \dots, q-1\}$

$S_A = g^x$

1. $r = g^k$, $k \in \{1, \dots, q-1\}$
2. $v = h(r \| m)$
3. $s = k + v \cdot x \quad \overset{\text{manda}}{\Longrightarrow} \quad (v, s)$

Lo que B realmente hace

B: 1. $\alpha = g^s = g^{k+vx} = g^k \cdot g^{vx}$

2. $\beta = \alpha \cdot (g^x)^{q-v} = \alpha \cdot (g^x)^q \cdot (g^x)^{-v}$

$= \alpha \cdot (g^q)^x \cdot ((g^x)^v)^{-1}$

$= g^k \cdot \underbrace{(g^q)^x}_{e^x} \cdot \underbrace{(g^x)^v \cdot ((g^x)^v)^{-1}}_{e^x}$

$= g^k$

$\Longrightarrow h(\beta \| m) = v = h(g^k \| m)$

Grupos

por $(G, *)$ con:

G conjunto

$* : G \times G \to G$ tal que

Neutro:

$\forall a \in G \ \exists e$ tq $e * a = a$

Inverso:

$\forall a \in G \ \exists a^{-1} \in G$ tq $a^{-1} * a = e$

Asociatividad:

$\forall a, b, c \in G$: $(a * b) * c = a * (b * c)$

E1

Lagrange

G grupo finito y H subgrupo de G, entonces:

$$|H| \text{ divide a } |G|$$

Subgrupos generados

$\mathbb{Z}_{10}^{*} = (\{1,3,7,9\}, \cdot \text{ mod } 10)$ $\qquad |\mathbb{Z}_{10}^{*}| = |\{1,3,7,9\}| = 4$

¿Cuáles son los generados? verifique con Lagrange

$$\langle 1 \rangle = \{1\}$$

$$\langle 3 \rangle = \{3, 9, \underbrace{27}_{7}, \underbrace{21}_{1}, 3, \ldots\} = \{1,3,7,9\} \qquad 4|4 \checkmark$$

$$\langle 7 \rangle = \{7, \underbrace{49}_{9}, \underbrace{63}_{3}, 9, 7, 1, 3, \ldots\} = \{1,3,7,9\} \qquad 4|4 \checkmark$$

como arriba ...

$$\langle 9 \rangle = \{9, 81, 1, 1, \ldots\} = \{1, 9\} \qquad 2|4 \checkmark$$

E2 Schnorr: Efficient Signature Generation by Smart Cards

a) describa que hace el Verificador y Usuario

b) describa el protocolo, paso a paso

c) demuestre su correctitud

Verificador (KAC: key auth center)

- S_B, P_B • romper $\longleftrightarrow$ log discreto
- p : primo
- q : coprimo de $p-1$
- α : $\alpha^q = 1 \mod p$ (pequeño Fermat)
  - ↳
    - $\alpha \in \mathbb{Z}_p$
    - $|\langle \alpha \rangle| = q$
- h : one-way hash function

priv: S_B

pub: p, q, α, h, P_B

User

- s : $s \in \{1, \dots, q\}$
- v : $\alpha^{-s} \pmod p$ • romper $\longleftrightarrow$ log discreto
- I : identificación / ID del usuario

priv: s

pub: (I, v)

Genera firmas más cortas, menos cómputo, y preprocesar en Idle time.

Pocos bytes de comunicación

Autenticación

0. **Setup:** ver arriba

1. **Preprocesar:** A elige $r \in \{1, \dots, q\}$ y calcula $x = \alpha^r \bmod p$

2. **Inicialización:**

$$\boxed{A} \xrightarrow[\cdot\, x]{\cdot\, s=(I,v)} \boxed{B}$$

3. **Intercambio 1:**
$e \in \{0, \dots, 2^{t-1}\}$

$$\boxed{A} \xleftarrow[e]{} \boxed{B}$$

4. **Intercambio 2:**
$y = r + s \cdot e \pmod q$

$$\boxed{A} \xrightarrow[y]{} \boxed{B}$$

5. **Verificación:** $x \stackrel{?}{=} \alpha^y v^e \bmod p$

$\rightarrow$ True/False

Para firmar msg, basta con reemplazar e aleatorio por:

$$e = h(x, m)$$

Demostración

$$\alpha^r = \alpha^y v^e \bmod p$$

$$
\begin{aligned}
&\alpha^{r+se} \cdot v^e \\
&= \alpha^r \alpha^{se} v^e \\
&= \alpha^r (\alpha^s)^e v^e \\
&= \alpha^r \cdot (\alpha^s v)^e \\
&= \alpha^r \cdot (\alpha^s \cdot \alpha^{-s})^e \\
&= \alpha^r \cdot 1^e \\
&= \alpha^r
\end{aligned}
$$

$\Longrightarrow$ Sólo alguien con llave privada s puede autenticar al usuario

- Banco tiene (I, v)