

Inverso modular

# Inverso modular

## Definición

*b es inverso de a en módulo n si  $(a \cdot b) \equiv_n 1$ .*

# Inverso modular

## Definición

*b es inverso de a en módulo n si  $(a \cdot b) \equiv_n 1$ .*

## Ejemplo

3 es el inverso de 2 en módulo 5 puesto que  $3 \cdot 2 \equiv_5 1$ .

# Inverso modular

## Definición

*b es inverso de a en módulo n si  $(a \cdot b) \equiv_n 1$ .*

## Ejemplo

3 es el inverso de 2 en módulo 5 puesto que  $3 \cdot 2 \equiv_5 1$ .

▶ O, equivalentemente, se tiene que  $(3 \cdot 2) \bmod 5 = 1$ .

# Inverso modular

## Definición

*b es inverso de a en módulo n si  $(a \cdot b) \equiv_n 1$ .*

## Ejemplo

3 es el inverso de 2 en módulo 5 puesto que  $3 \cdot 2 \equiv_5 1$ .

▶ O, equivalentemente, se tiene que  $(3 \cdot 2) \bmod 5 = 1$ .

¿Todo número tiene inverso modular?

# Inverso modular

## Definición

*b es inverso de a en módulo n si  $(a \cdot b) \equiv_n 1$ .*

## Ejemplo

3 es el inverso de 2 en módulo 5 puesto que  $3 \cdot 2 \equiv_5 1$ .

▶ O, equivalentemente, se tiene que  $(3 \cdot 2) \bmod 5 = 1$ .

¿Todo número tiene inverso modular?

▶ No. Por ejemplo, 2 no tiene inverso en módulo 4.

# Inverso modular

## Definición

*b es inverso de a en módulo n si  $(a \cdot b) \equiv_n 1$ .*

## Ejemplo

3 es el inverso de 2 en módulo 5 puesto que  $3 \cdot 2 \equiv_5 1$ .

- ▶ O, equivalentemente, se tiene que  $(3 \cdot 2) \bmod 5 = 1$ .

¿Todo número tiene inverso modular?

- ▶ No. Por ejemplo, 2 no tiene inverso en módulo 4.
- ▶ ¿Bajo qué condiciones  $a$  tiene inverso en módulo  $n$ ?

# Inverso modular: existencia

## Teorema

*$a$  tiene inverso en módulo  $n$  si y sólo si  $MCD(a, n) = 1$ .*



# Inverso modular: existencia

## Teorema

*$a$  tiene inverso en módulo  $n$  si y sólo si  $MCD(a, n) = 1$ .*

**Demostración:** ( $\Rightarrow$ ) Suponga que  $b$  es inverso de  $a$  en módulo  $n$ .

# Inverso modular: existencia

## Teorema

*$a$  tiene inverso en módulo  $n$  si y sólo si  $\text{MCD}(a, n) = 1$ .*

**Demostración:** ( $\Rightarrow$ ) Suponga que  $b$  es inverso de  $a$  en módulo  $n$ .

▶ Entonces:  $a \cdot b \equiv_n 1$ .

# Inverso modular: existencia

## Teorema

*$a$  tiene inverso en módulo  $n$  si y sólo si  $\text{MCD}(a, n) = 1$ .*

**Demostración:** ( $\Rightarrow$ ) Suponga que  $b$  es inverso de  $a$  en módulo  $n$ .

▶ Entonces:  $a \cdot b \equiv_n 1$ .

Se deduce que  $a \cdot b - 1 = \alpha \cdot n$ , por lo que  $1 = a \cdot b - \alpha \cdot n$ .

# Inverso modular: existencia

## Teorema

*$a$  tiene inverso en módulo  $n$  si y sólo si  $\text{MCD}(a, n) = 1$ .*

**Demostración:** ( $\Rightarrow$ ) Suponga que  $b$  es inverso de  $a$  en módulo  $n$ .

▶ Entonces:  $a \cdot b \equiv_n 1$ .

Se deduce que  $a \cdot b - 1 = \alpha \cdot n$ , por lo que  $1 = a \cdot b - \alpha \cdot n$ .

Concluimos que si  $c|a$  y  $c|n$ , entonces  $c|1$ .

# Inverso modular: existencia

## Teorema

*$a$  tiene inverso en módulo  $n$  si y sólo si  $\text{MCD}(a, n) = 1$ .*

**Demostración:** ( $\Rightarrow$ ) Suponga que  $b$  es inverso de  $a$  en módulo  $n$ .

▶ Entonces:  $a \cdot b \equiv_n 1$ .

Se deduce que  $a \cdot b - 1 = \alpha \cdot n$ , por lo que  $1 = a \cdot b - \alpha \cdot n$ .

Concluimos que si  $c|a$  y  $c|n$ , entonces  $c|1$ .

▶ Por lo tanto  $c$  debe ser igual a 1, de lo que concluimos que  $\text{MCD}(a, n) = 1$ .

# Inverso modular: Existencia

( $\Leftarrow$ ) Suponga que  $\text{MCD}(a, n) = 1$ .

# Inverso modular: Existencia

( $\Leftarrow$ ) Suponga que  $\text{MCD}(a, n) = 1$ .

Ejecutando el algoritmo extendido de Euclides para el cálculo del máximo común divisor obtenemos  $s$  y  $t$  tales que:

$$1 = s \cdot n + t \cdot a$$

# Inverso modular: Existencia

( $\Leftarrow$ ) Suponga que  $\text{MCD}(a, n) = 1$ .

Ejecutando el algoritmo extendido de Euclides para el cálculo del máximo común divisor obtenemos  $s$  y  $t$  tales que:

$$1 = s \cdot n + t \cdot a$$

Por lo tanto:  $a \cdot t \equiv_n 1$ .



# Inverso modular: Existencia

( $\Leftarrow$ ) Suponga que  $\text{MCD}(a, n) = 1$ .

Ejecutando el algoritmo extendido de Euclides para el cálculo del máximo común divisor obtenemos  $s$  y  $t$  tales que:

$$1 = s \cdot n + t \cdot a$$

Por lo tanto:  $a \cdot t \equiv_n 1$ .

► Concluimos que  $a$  tiene inverso en módulo  $n$ .



# El pequeño teorema de Fermat

# Aritmética modular: Una propiedad fundamental

## Teorema (Fermat)

Sea  $p$  un número primo. Si  $a \in \{0, \dots, p-1\}$ , entonces  $a^p \bmod p = a$ .

# Aritmética modular: Una propiedad fundamental

## Teorema (Fermat)

Sea  $p$  un número primo. Si  $a \in \{0, \dots, p-1\}$ , entonces  $a^p \bmod p = a$ .

**Demostración:** Por inducción en  $a$ .

# Aritmética modular: Una propiedad fundamental

## Teorema (Fermat)

Sea  $p$  un número primo. Si  $a \in \{0, \dots, p-1\}$ , entonces  $a^p \bmod p = a$ .

**Demostración:** Por inducción en  $a$ .

Para  $a = 0$  y  $a = 1$  se cumple trivialmente. Suponga que  $a^p \bmod p = a$  y  $2 \leq (a+1) < p$ .

# Aritmética modular: Una propiedad fundamental

## Teorema (Fermat)

Sea  $p$  un número primo. Si  $a \in \{0, \dots, p-1\}$ , entonces  $a^p \bmod p = a$ .

**Demostración:** Por inducción en  $a$ .

Para  $a = 0$  y  $a = 1$  se cumple trivialmente. Suponga que  $a^p \bmod p = a$  y  $2 \leq (a+1) < p$ .

▶ O, equivalentemente, tenemos que  $a^p \equiv_p a$ .

# Aritmética modular: Una propiedad fundamental

## Teorema (Fermat)

Sea  $p$  un número primo. Si  $a \in \{0, \dots, p-1\}$ , entonces  $a^p \bmod p = a$ .

**Demostración:** Por inducción en  $a$ .

Para  $a = 0$  y  $a = 1$  se cumple trivialmente. Suponga que  $a^p \bmod p = a$  y  $2 \leq (a+1) < p$ .

▶ O, equivalentemente, tenemos que  $a^p \equiv_p a$ .

Sabemos que:

$$(a+1)^p = \sum_{k=0}^p \binom{p}{k} a^k$$

# Aritmética modular: Una propiedad fundamental

Por lo tanto:

$$(a+1)^p - (a+1) = (a^p - a) + \sum_{k=1}^{p-1} \binom{p}{k} a^k$$



# Aritmética modular: Una propiedad fundamental

Por lo tanto:

$$(a+1)^p - (a+1) = (a^p - a) + \sum_{k=1}^{p-1} \binom{p}{k} a^k$$

Lema

*Si  $k \in \{1, \dots, p-1\}$ , entonces  $p \mid \binom{p}{k}$*

# Aritmética modular: Una propiedad fundamental

Por lo tanto:

$$(a+1)^p - (a+1) = (a^p - a) + \sum_{k=1}^{p-1} \binom{p}{k} a^k$$

**Lema**

*Si  $k \in \{1, \dots, p-1\}$ , entonces  $p \mid \binom{p}{k}$*

**Demostración del lema:** Sabemos que:

$$\binom{p}{k} = \frac{p!}{k! \cdot (p-k)!} = \frac{p \cdot (p-1) \cdot \dots \cdot (p-k+1)}{k!}$$

# Aritmética modular: Una propiedad fundamental

Por lo tanto:

$$(a+1)^p - (a+1) = (a^p - a) + \sum_{k=1}^{p-1} \binom{p}{k} a^k$$

**Lema**

*Si  $k \in \{1, \dots, p-1\}$ , entonces  $p \mid \binom{p}{k}$*

**Demostración del lema:** Sabemos que:

$$\binom{p}{k} = \frac{p!}{k! \cdot (p-k)!} = \frac{p \cdot (p-1) \cdot \dots \cdot (p-k+1)}{k!}$$

Como  $k \in \{1, \dots, p-1\}$  y  $p$  es un número primo:

$\frac{(p-1) \cdot \dots \cdot (p-k+1)}{k!}$  es un número entero

# Aritmética modular: Una propiedad fundamental

Por lo tanto:  $\binom{p}{k} = p \cdot \alpha$ , donde  $\alpha$  es un número entero.

# Aritmética modular: Una propiedad fundamental

Por lo tanto:  $\binom{p}{k} = p \cdot \alpha$ , donde  $\alpha$  es un número entero.

▶ Concluimos que  $p \mid \binom{p}{k}$ .



# Aritmética modular: Una propiedad fundamental

Por lo tanto:  $\binom{p}{k} = p \cdot \alpha$ , donde  $\alpha$  es un número entero.

► Concluimos que  $p \mid \binom{p}{k}$ .



Del lema concluimos que  $p \mid \sum_{k=1}^{p-1} \binom{p}{k} a^k$ .

# Aritmética modular: Una propiedad fundamental

Por lo tanto:  $\binom{p}{k} = p \cdot \alpha$ , donde  $\alpha$  es un número entero.

► Concluimos que  $p \mid \binom{p}{k}$ .



Del lema concluimos que  $p \mid \sum_{k=1}^{p-1} \binom{p}{k} a^k$ .

Por lo tanto, dado que  $p \mid (a^p - a)$  por hipótesis de inducción, tenemos que  $p \mid ((a+1)^p - (a+1))$ .

# Aritmética modular: Una propiedad fundamental

Por lo tanto:  $\binom{p}{k} = p \cdot \alpha$ , donde  $\alpha$  es un número entero.

► Concluimos que  $p \mid \binom{p}{k}$ .



Del lema concluimos que  $p \mid \sum_{k=1}^{p-1} \binom{p}{k} a^k$ .

Por lo tanto, dado que  $p \mid (a^p - a)$  por hipótesis de inducción, tenemos que  $p \mid ((a+1)^p - (a+1))$ .

► Concluimos que  $(a+1)^p \equiv_p (a+1)$ ,



# Aritmética modular: Una propiedad fundamental

Por lo tanto:  $\binom{p}{k} = p \cdot \alpha$ , donde  $\alpha$  es un número entero.

► Concluimos que  $p \mid \binom{p}{k}$ .



Del lema concluimos que  $p \mid \sum_{k=1}^{p-1} \binom{p}{k} a^k$ .

Por lo tanto, dado que  $p \mid (a^p - a)$  por hipótesis de inducción, tenemos que  $p \mid ((a+1)^p - (a+1))$ .

► Concluimos que  $(a+1)^p \equiv_p (a+1)$ , y de esto concluimos que  $(a+1)^p \bmod p = (a+1)$ .



# Aritmética modular: Una propiedad fundamental

## Corolario (Fermat)

Sea  $p$  un número primo. Si  $a \in \{1, \dots, p-1\}$ , entonces  $a^{p-1} \bmod p = 1$ .

# Aritmética modular: Una propiedad fundamental

## Corolario (Fermat)

Sea  $p$  un número primo. Si  $a \in \{1, \dots, p-1\}$ , entonces  $a^{p-1} \bmod p = 1$ .

**Demostración:** Por el teorema anterior sabemos que

$$a^p \equiv_p a$$

# Aritmética modular: Una propiedad fundamental

## Corolario (Fermat)

Sea  $p$  un número primo. Si  $a \in \{1, \dots, p-1\}$ , entonces  $a^{p-1} \bmod p = 1$ .

**Demostración:** Por el teorema anterior sabemos que

$$a^p \equiv_p a$$

Como  $a \in \{1, \dots, p-1\}$  y  $p$  es primo, se tiene que  $\text{MCD}(a, p) = 1$ .

# Aritmética modular: Una propiedad fundamental

## Corolario (Fermat)

Sea  $p$  un número primo. Si  $a \in \{1, \dots, p-1\}$ , entonces  $a^{p-1} \bmod p = 1$ .

**Demostración:** Por el teorema anterior sabemos que

$$a^p \equiv_p a$$

Como  $a \in \{1, \dots, p-1\}$  y  $p$  es primo, se tiene que  $\text{MCD}(a, p) = 1$ .

- ▶ Por lo tanto,  $a$  tiene inverso en módulo  $p$ . Vale decir, existe  $b$  tal que  $(a \cdot b) \equiv_p 1$ .

# Aritmética modular: Una propiedad fundamental

Considerando que  $p \geq 1$ , tenemos que:

$$\begin{aligned}a^p \equiv_p a &\Rightarrow (a^p \cdot b) \equiv_p (a \cdot b) \\&\Rightarrow ((a^{p-1} \cdot a) \cdot b) \equiv_p 1 \\&\Rightarrow (a^{p-1} \cdot (a \cdot b)) \equiv_p 1 \\&\Rightarrow (a^{p-1} \cdot 1) \equiv_p 1 \\&\Rightarrow a^{p-1} \equiv_p 1\end{aligned}$$

# Aritmética modular: Una propiedad fundamental

Considerando que  $p \geq 1$ , tenemos que:

$$\begin{aligned} a^p \equiv_p a &\Rightarrow (a^p \cdot b) \equiv_p (a \cdot b) \\ &\Rightarrow ((a^{p-1} \cdot a) \cdot b) \equiv_p 1 \\ &\Rightarrow (a^{p-1} \cdot (a \cdot b)) \equiv_p 1 \\ &\Rightarrow (a^{p-1} \cdot 1) \equiv_p 1 \\ &\Rightarrow a^{p-1} \equiv_p 1 \end{aligned}$$

Concluimos que  $a^{p-1} \bmod p = 1$ .



# La extensión a todos los enteros

## Corolario

Sea  $p$  un número primo.



# La extensión a todos los enteros

## Corolario

Sea  $p$  un número primo.

1. Para cada  $a \in \mathbb{Z}$  se tiene que  $a^p \equiv_p a$ .

# La extensión a todos los enteros

## Corolario

Sea  $p$  un número primo.

1. Para cada  $a \in \mathbb{Z}$  se tiene que  $a^p \equiv_p a$ .
2. Para cada  $a \in \mathbb{Z}$  tal que  $a \not\equiv_p 0$ , se tiene que  $a^{p-1} \bmod p = 1$ .

# La extensión a todos los enteros

## Corolario

Sea  $p$  un número primo.

1. Para cada  $a \in \mathbb{Z}$  se tiene que  $a^p \equiv_p a$ .
2. Para cada  $a \in \mathbb{Z}$  tal que  $a \not\equiv_p 0$ , se tiene que  $a^{p-1} \bmod p = 1$ .

## Ejercicio

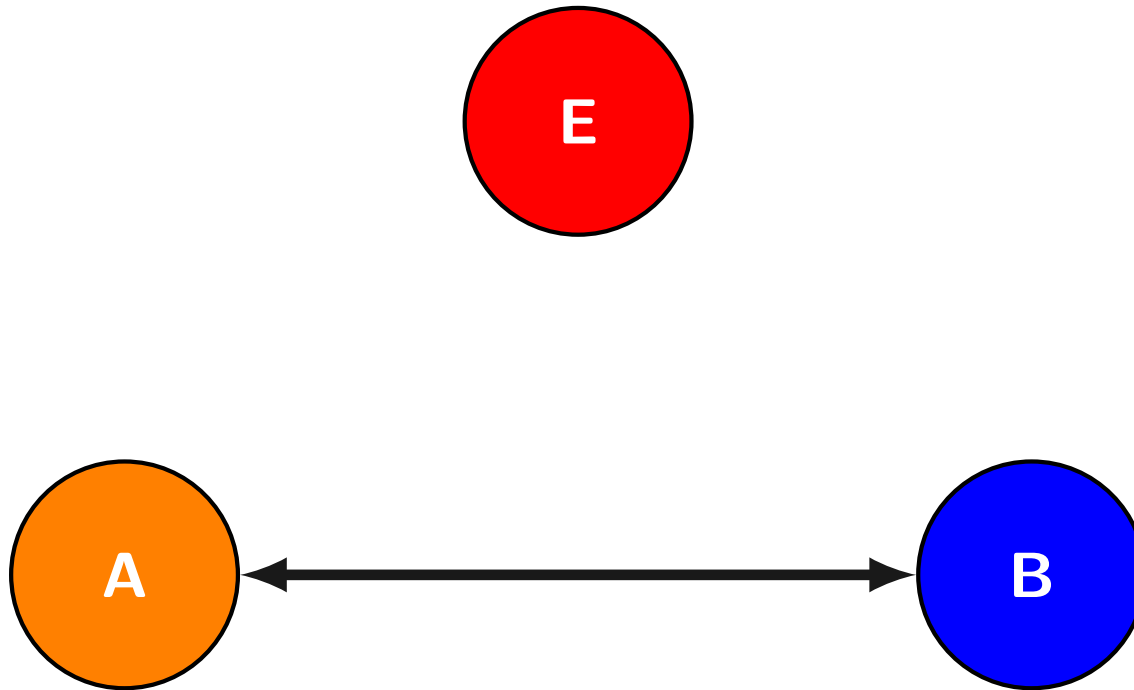
Demuestre el corolario.

# Criptografía de clave pública

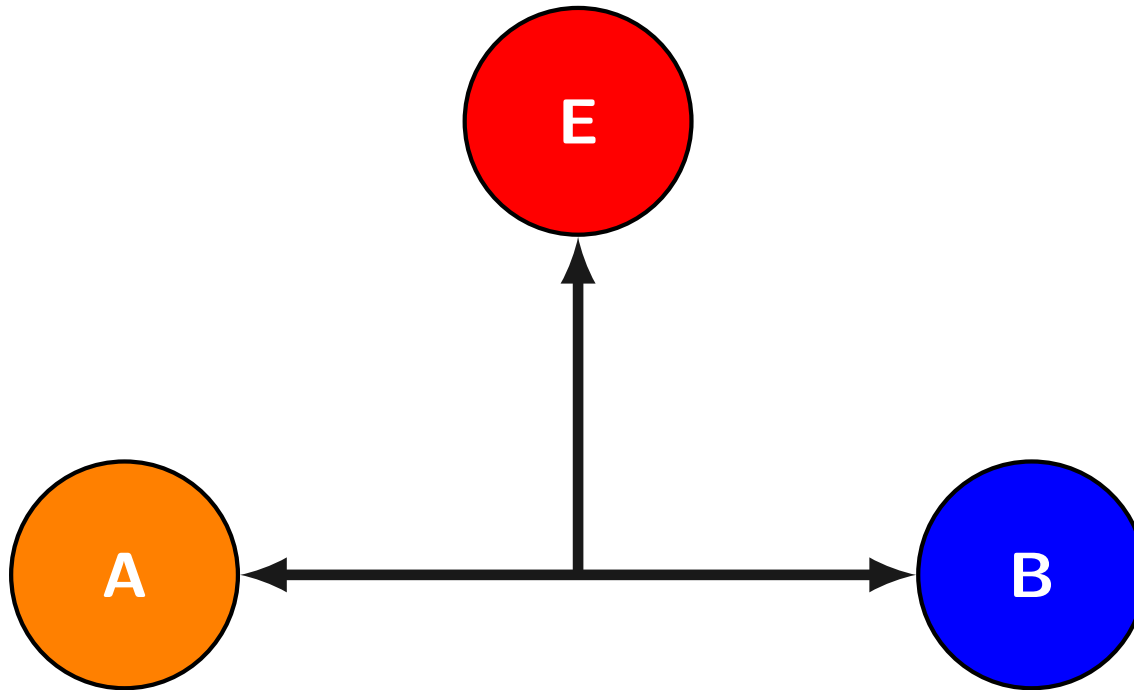
El escenario clásico: criptografía de clave secreta



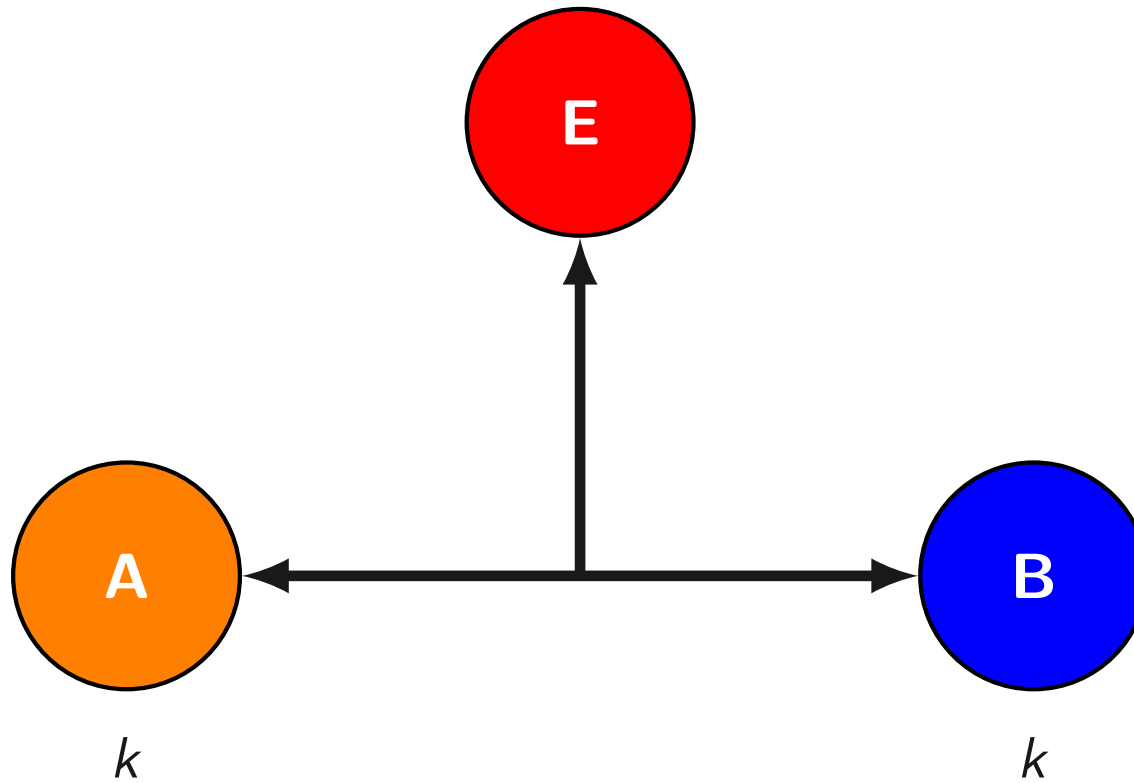
El escenario clásico: criptografía de clave secreta



El escenario clásico: criptografía de clave secreta

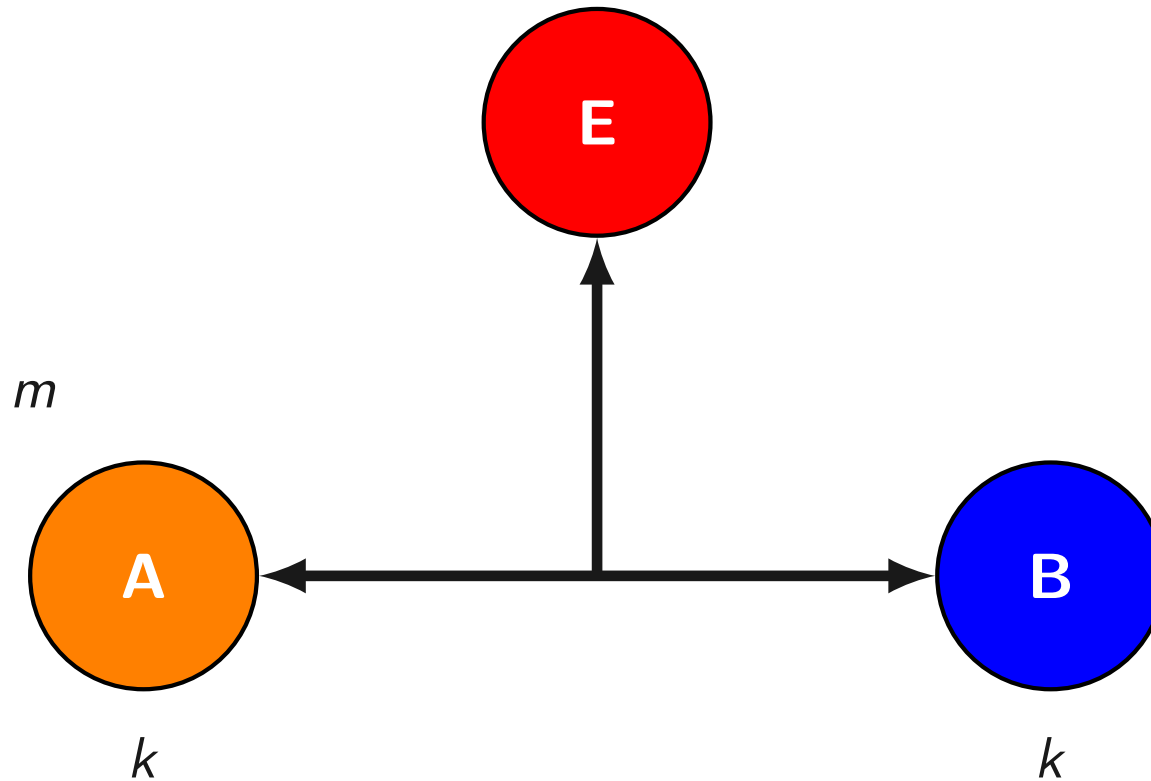


El escenario clásico: criptografía de clave secreta

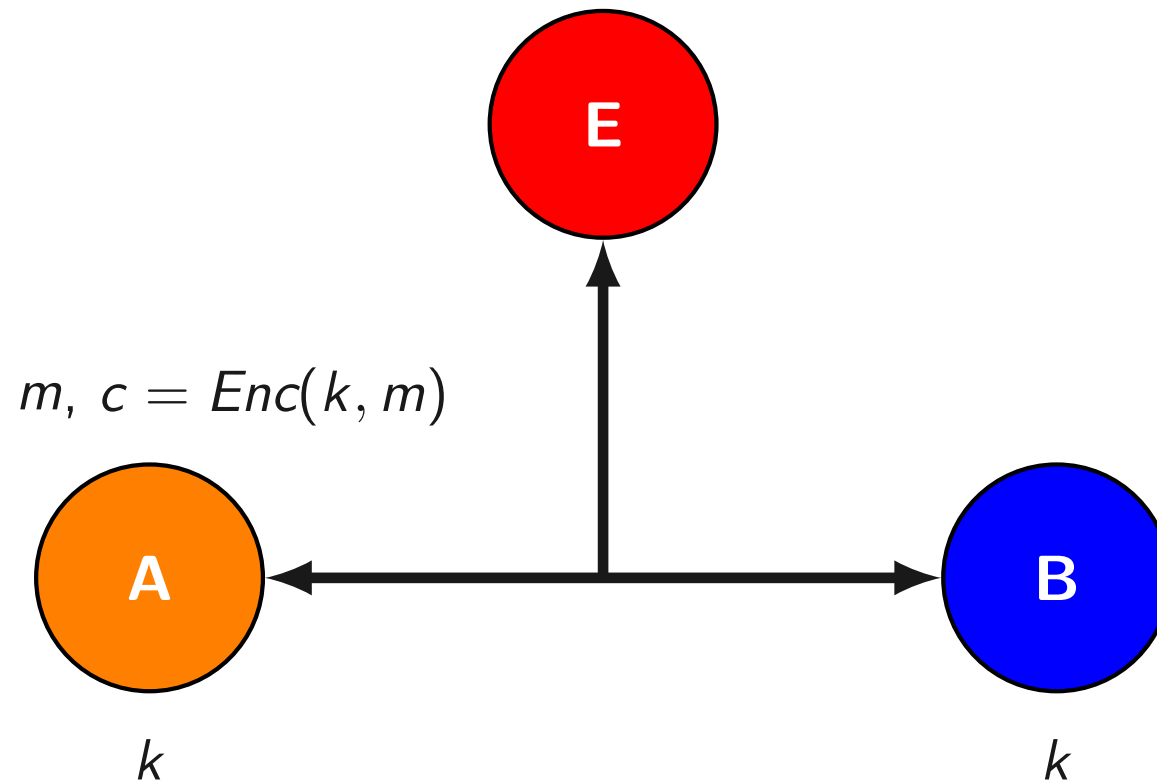




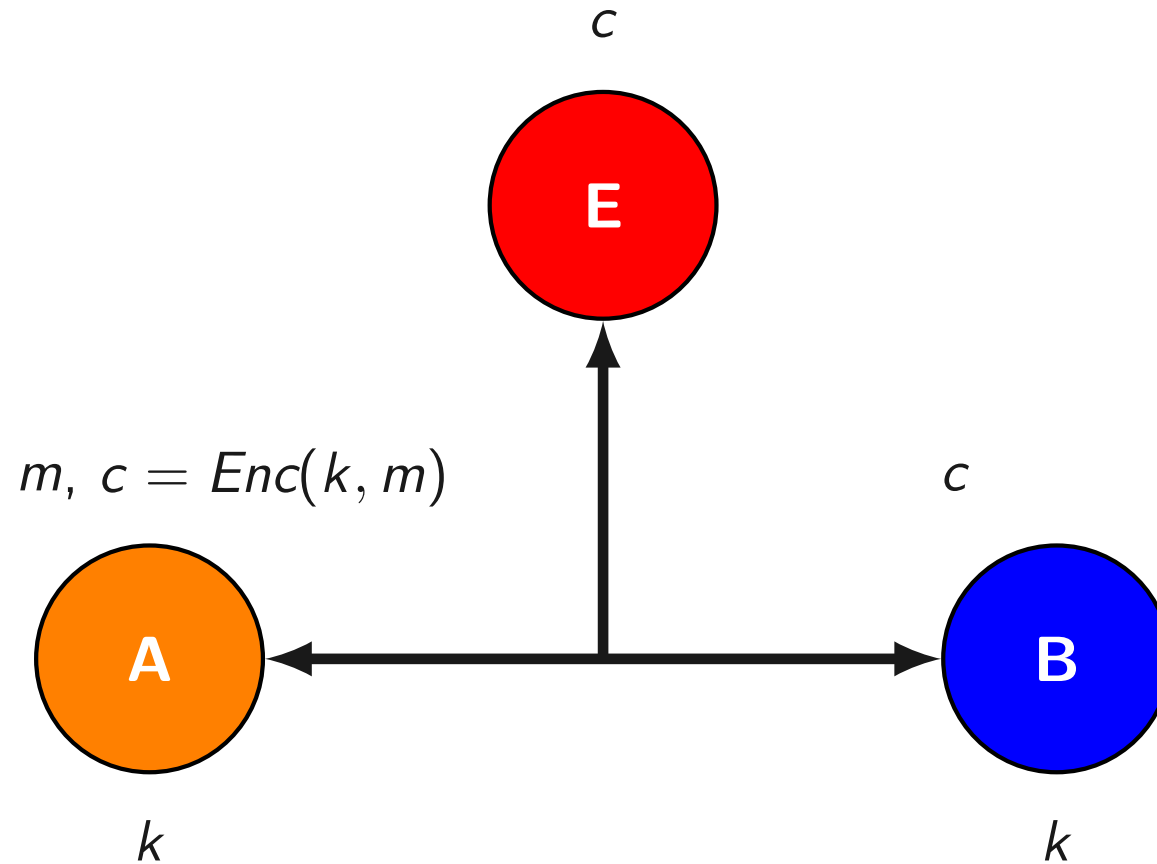
## El escenario clásico: criptografía de clave secreta



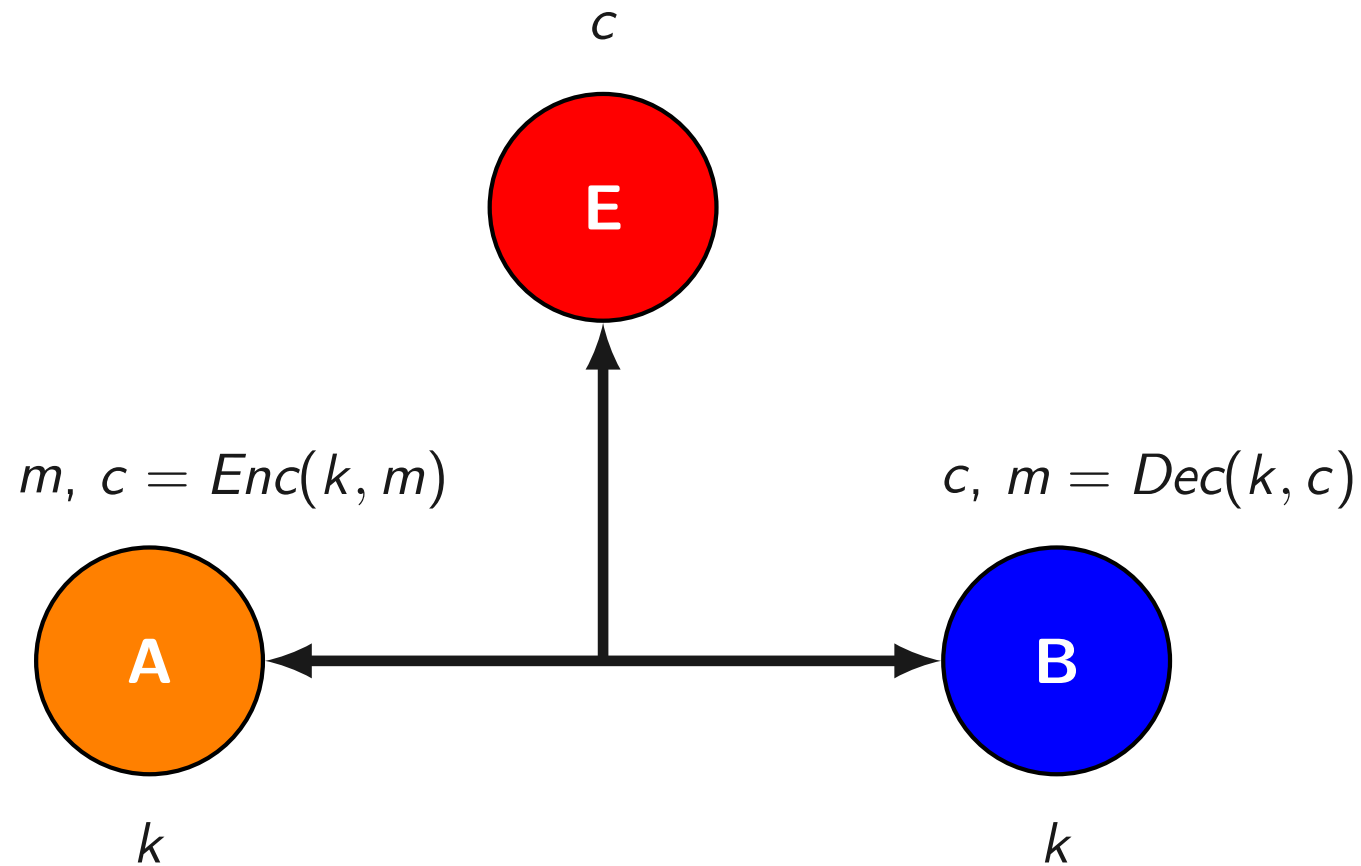
## El escenario clásico: criptografía de clave secreta



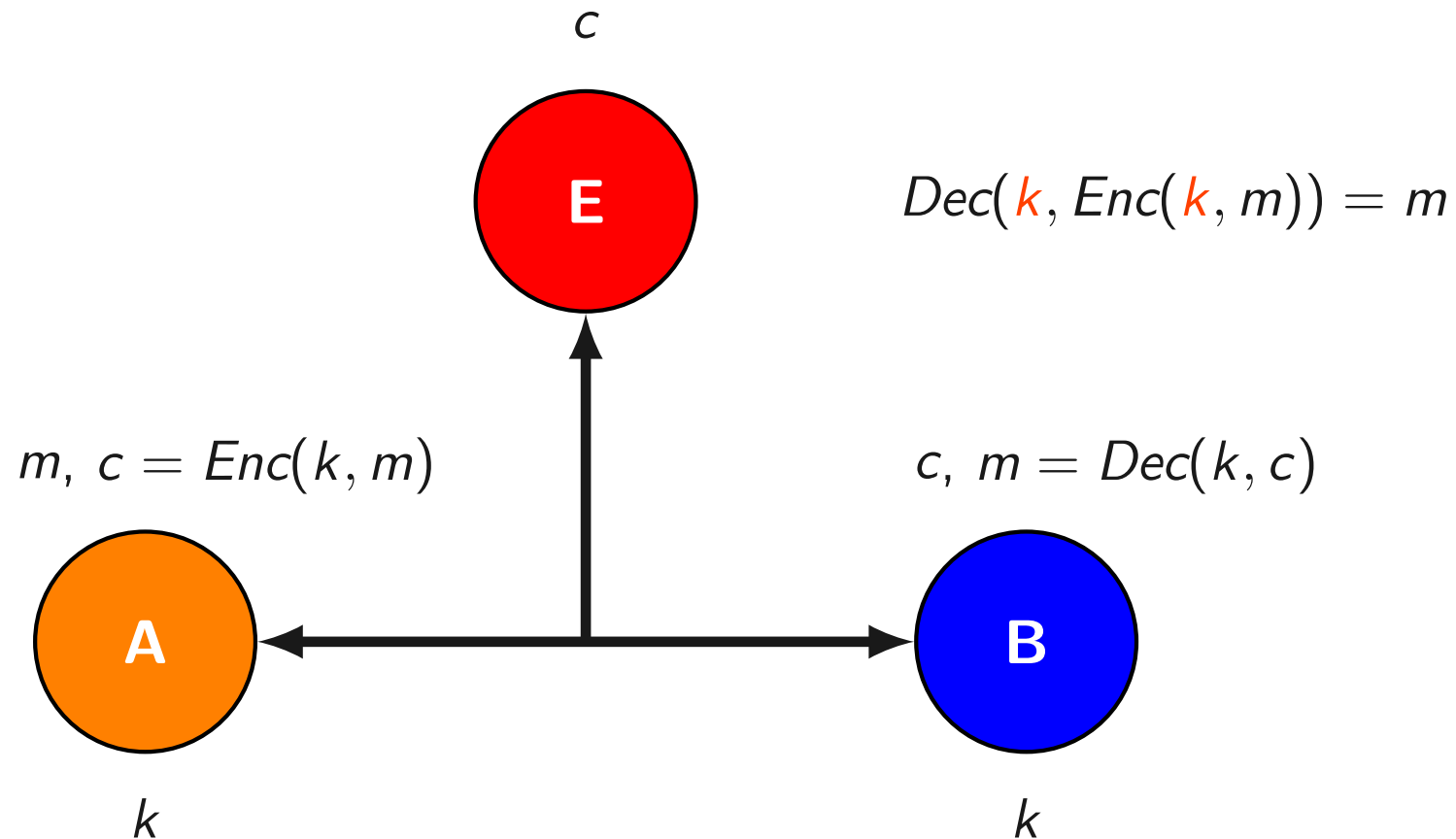
# El escenario clásico: criptografía de clave secreta



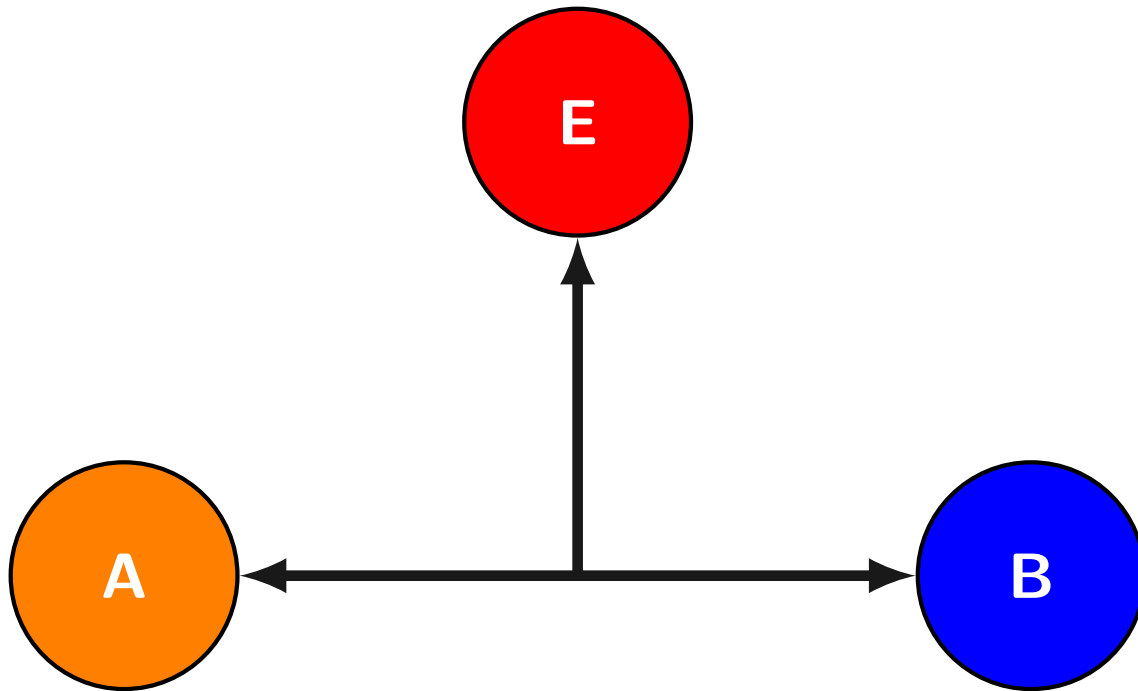
## El escenario clásico: criptografía de clave secreta



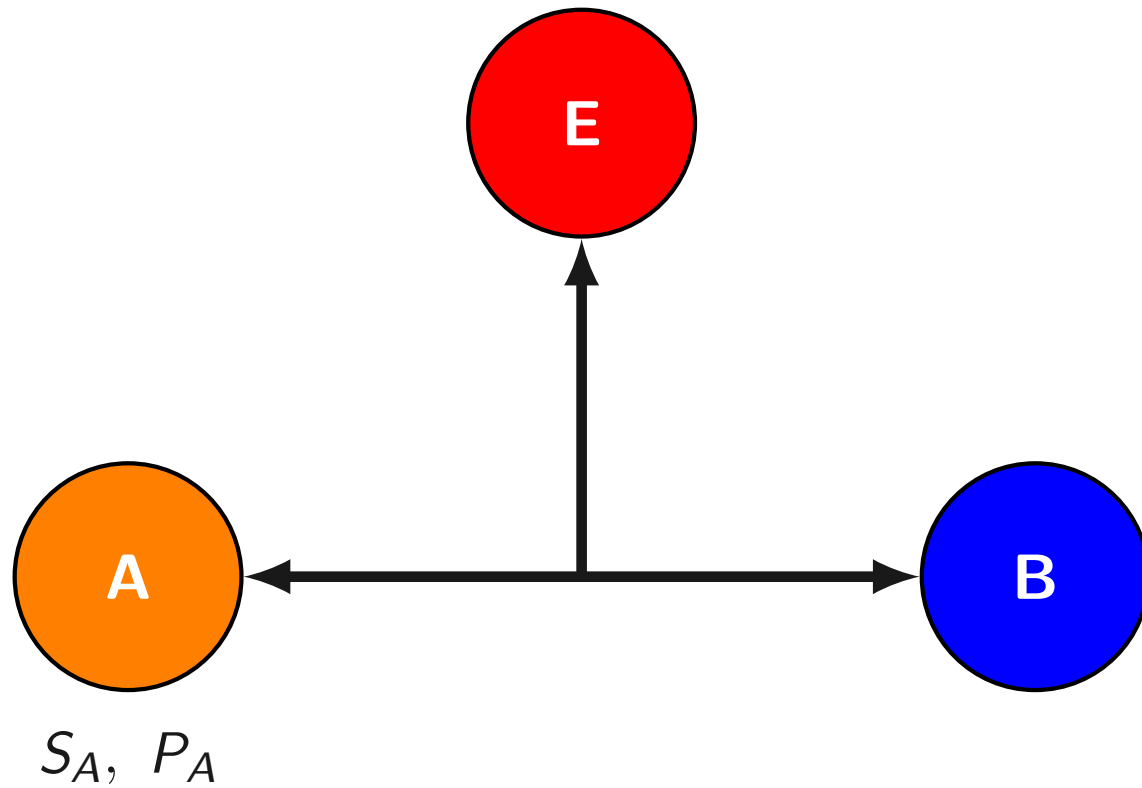
# El escenario clásico: criptografía de clave secreta



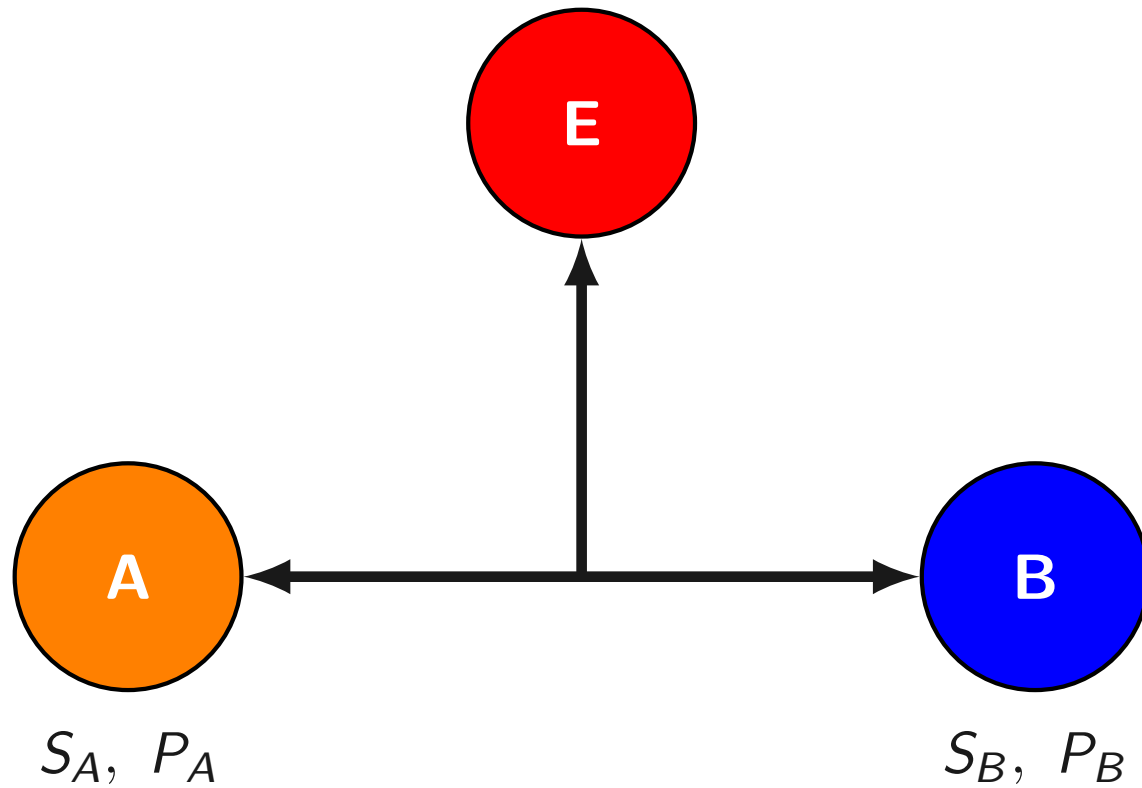
## El escenario moderno: criptografía de clave pública



# El escenario moderno: criptografía de clave pública

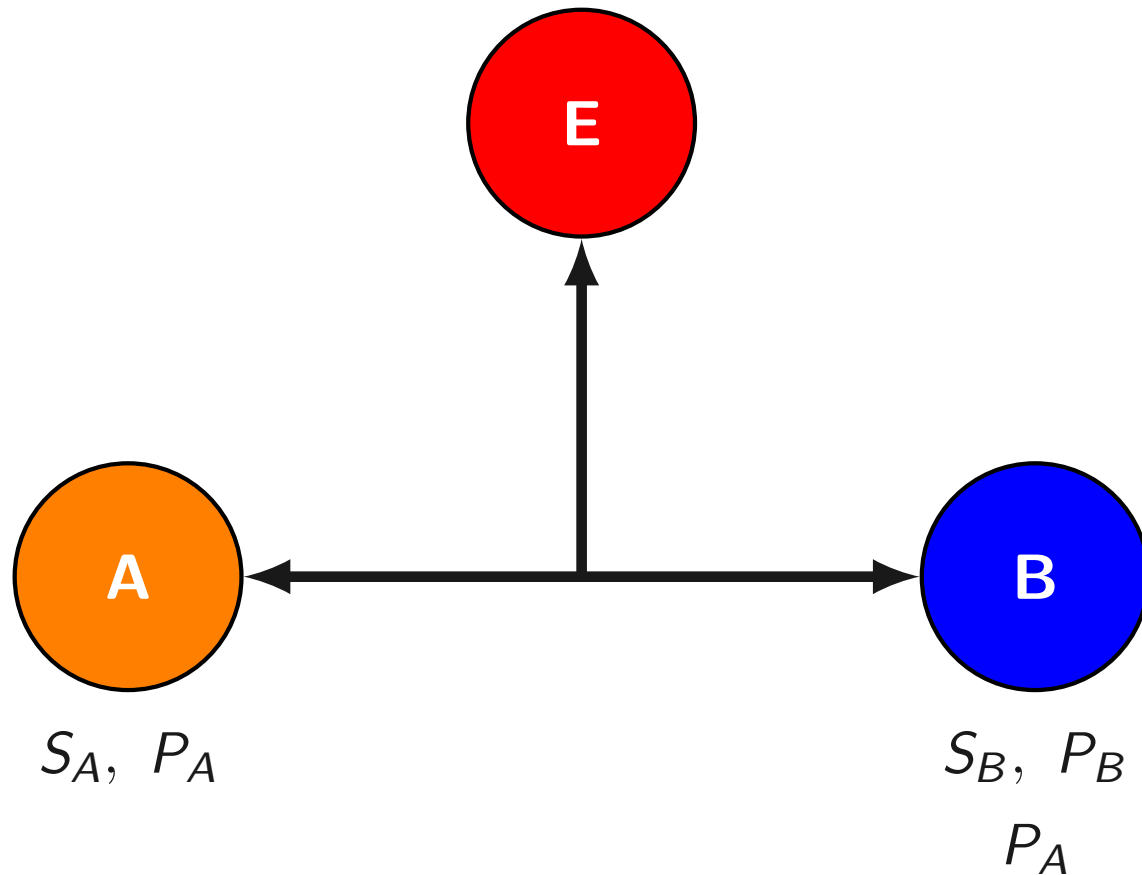


# El escenario moderno: criptografía de clave pública

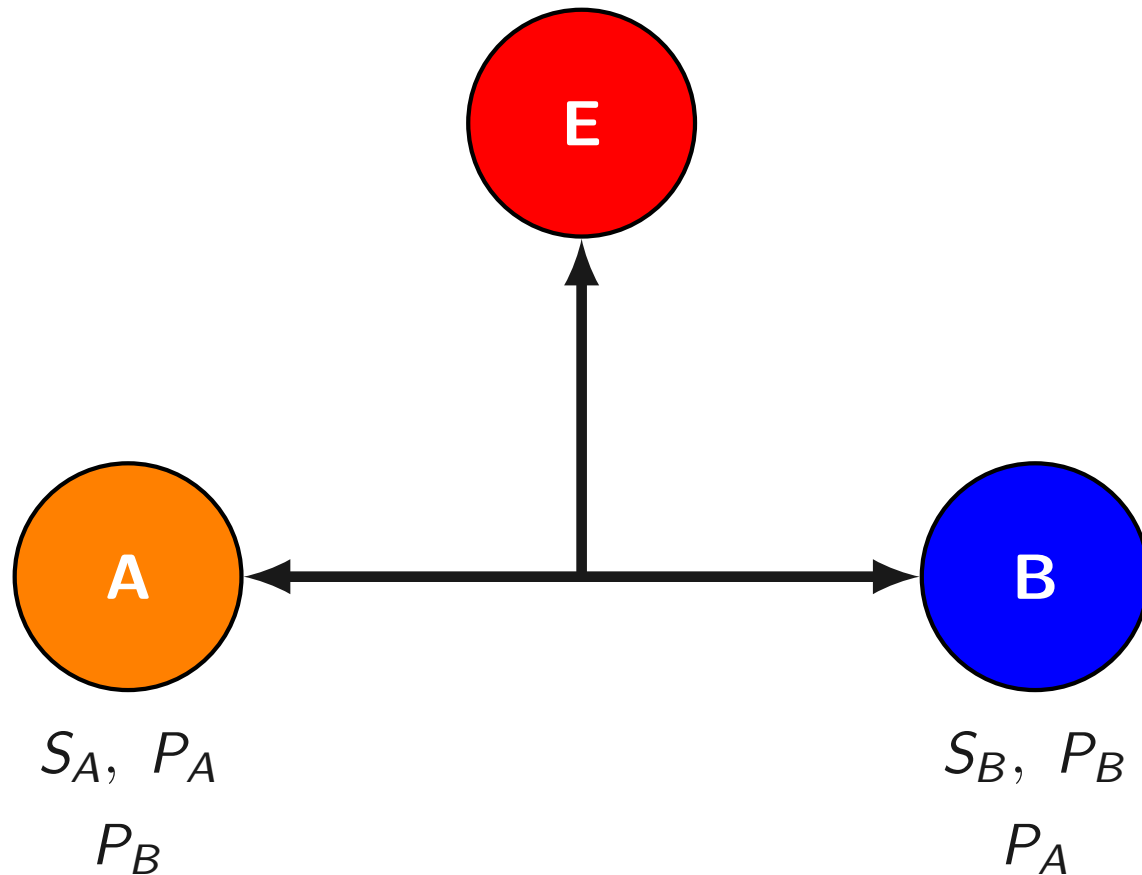




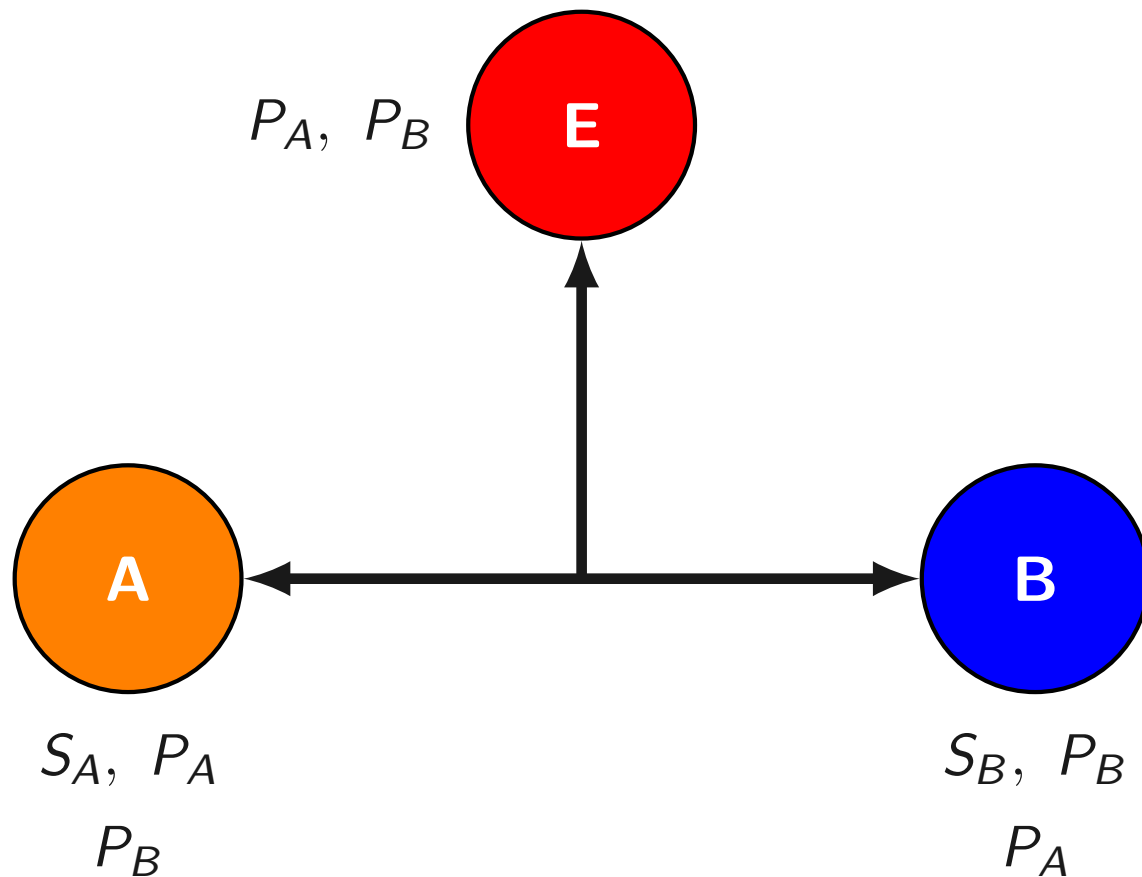
# El escenario moderno: criptografía de clave pública



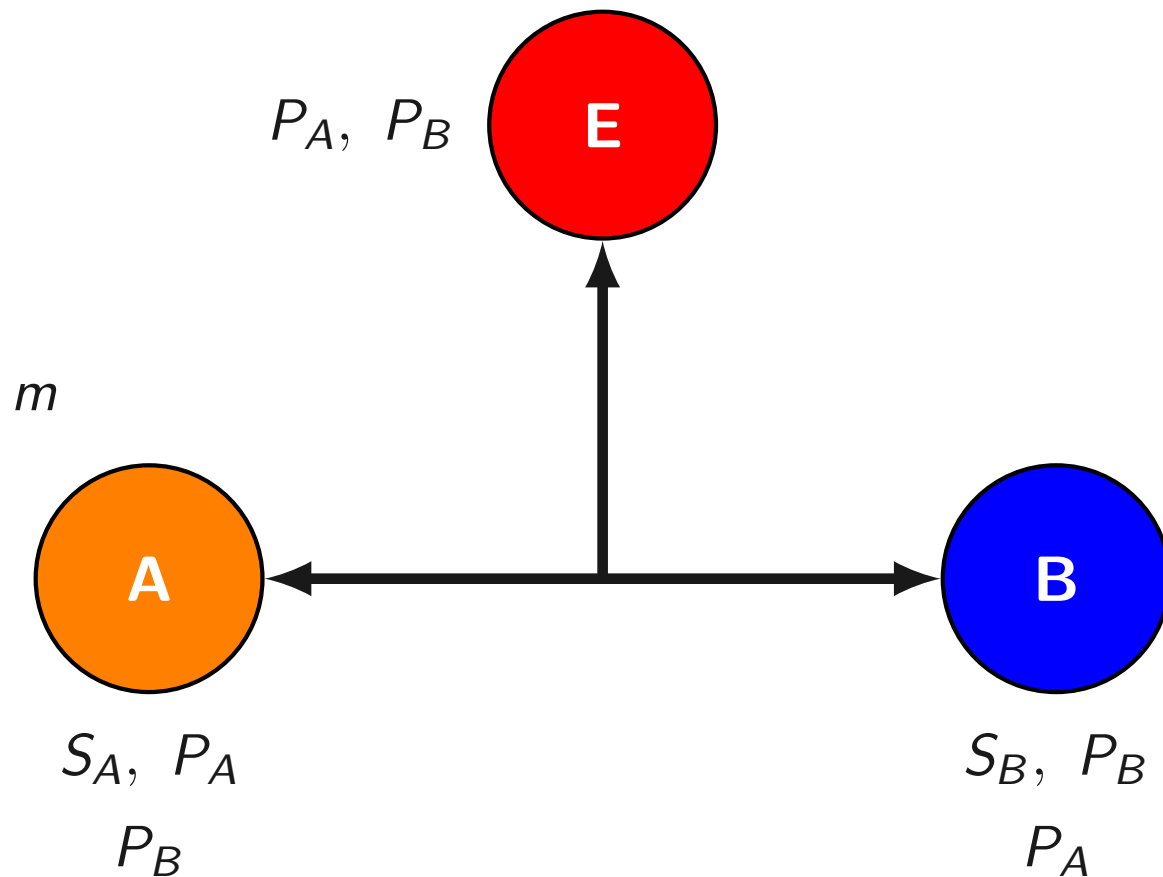
# El escenario moderno: criptografía de clave pública



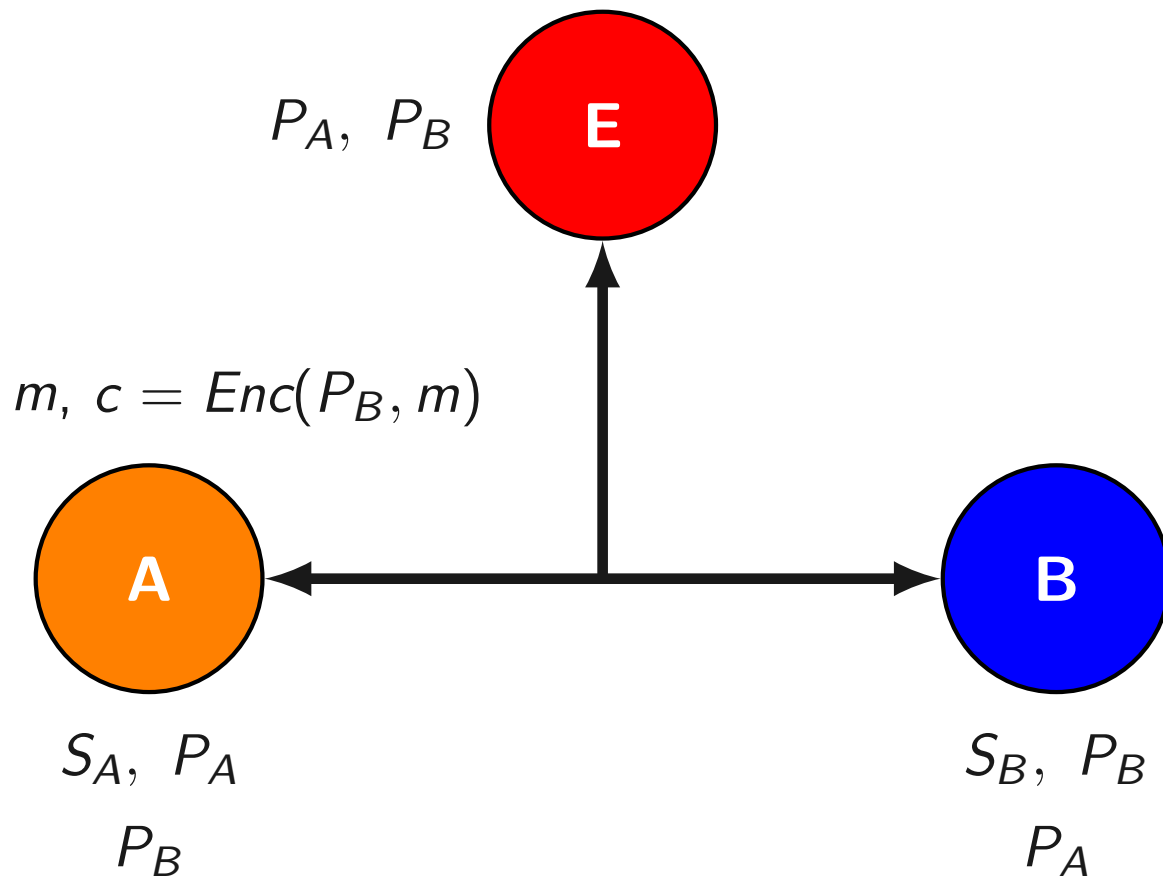
# El escenario moderno: criptografía de clave pública



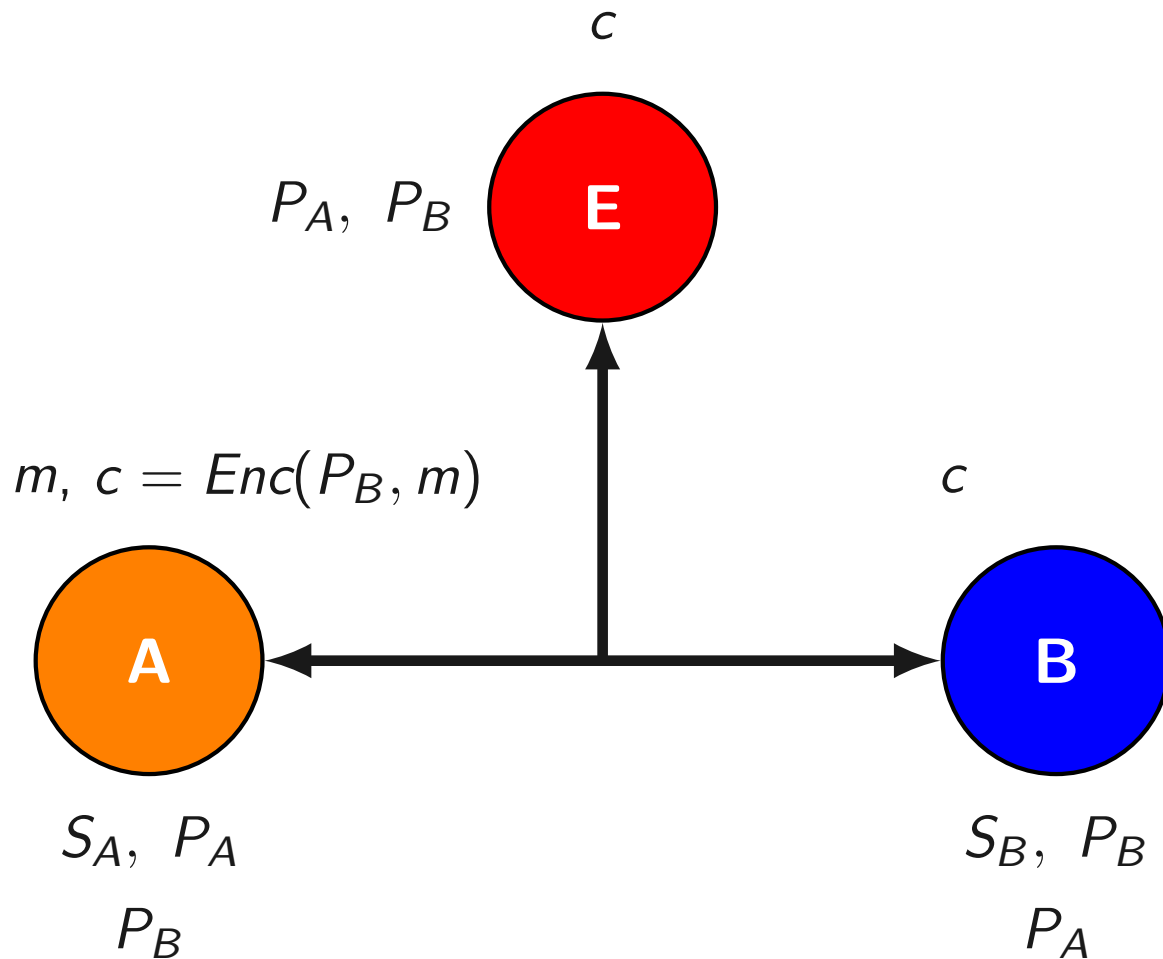
# El escenario moderno: criptografía de clave pública



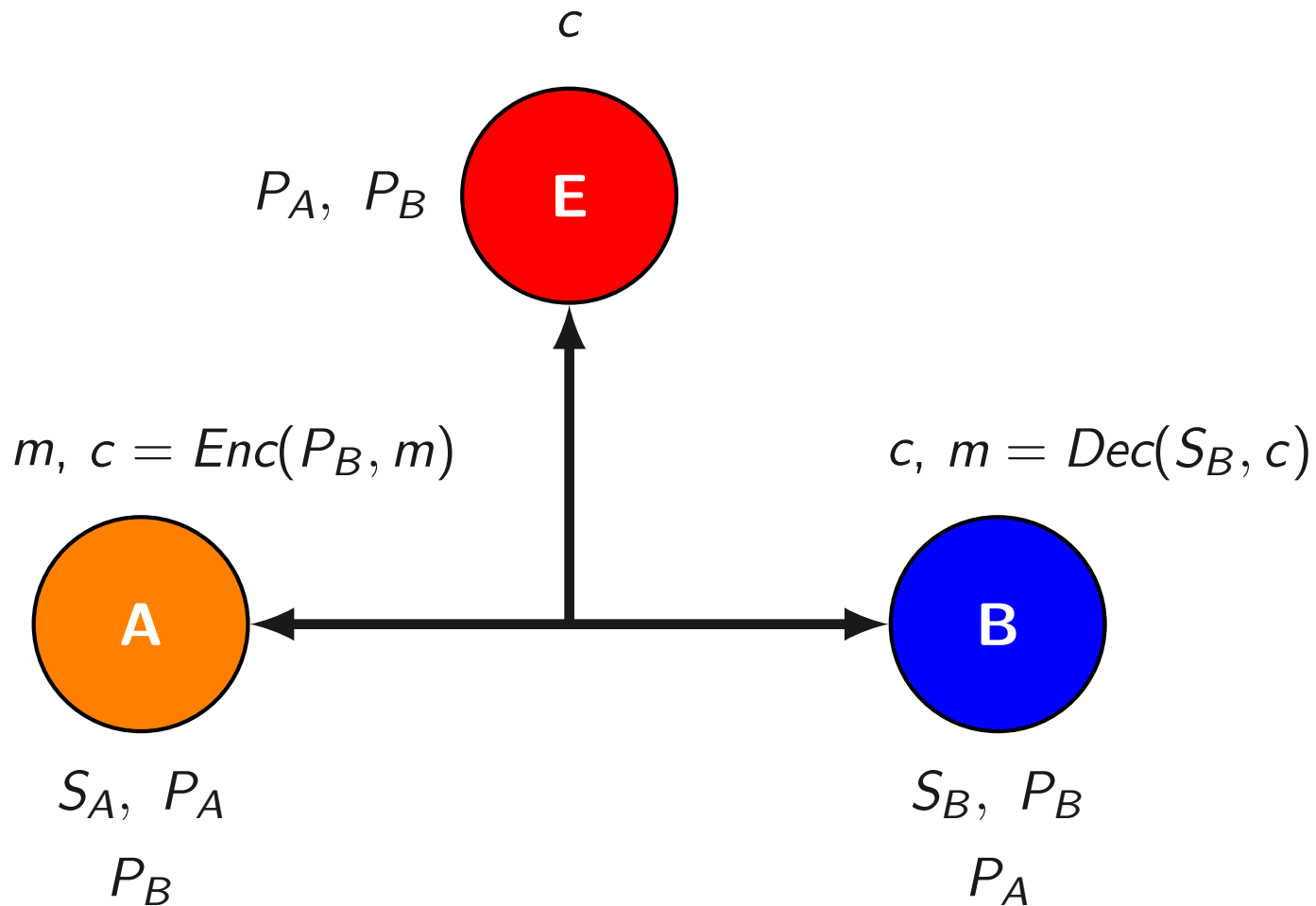
# El escenario moderno: criptografía de clave pública



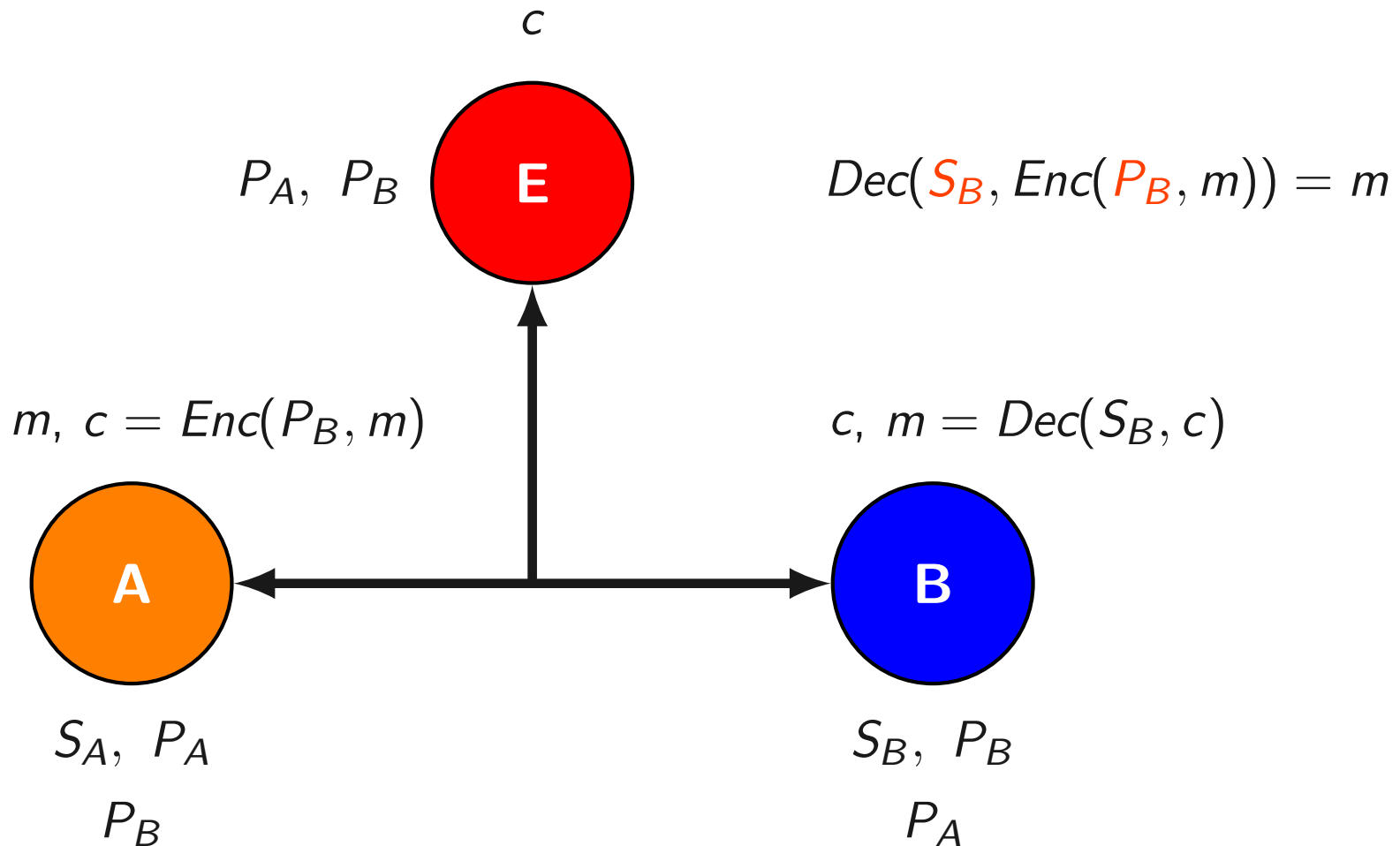
# El escenario moderno: criptografía de clave pública



# El escenario moderno: criptografía de clave pública

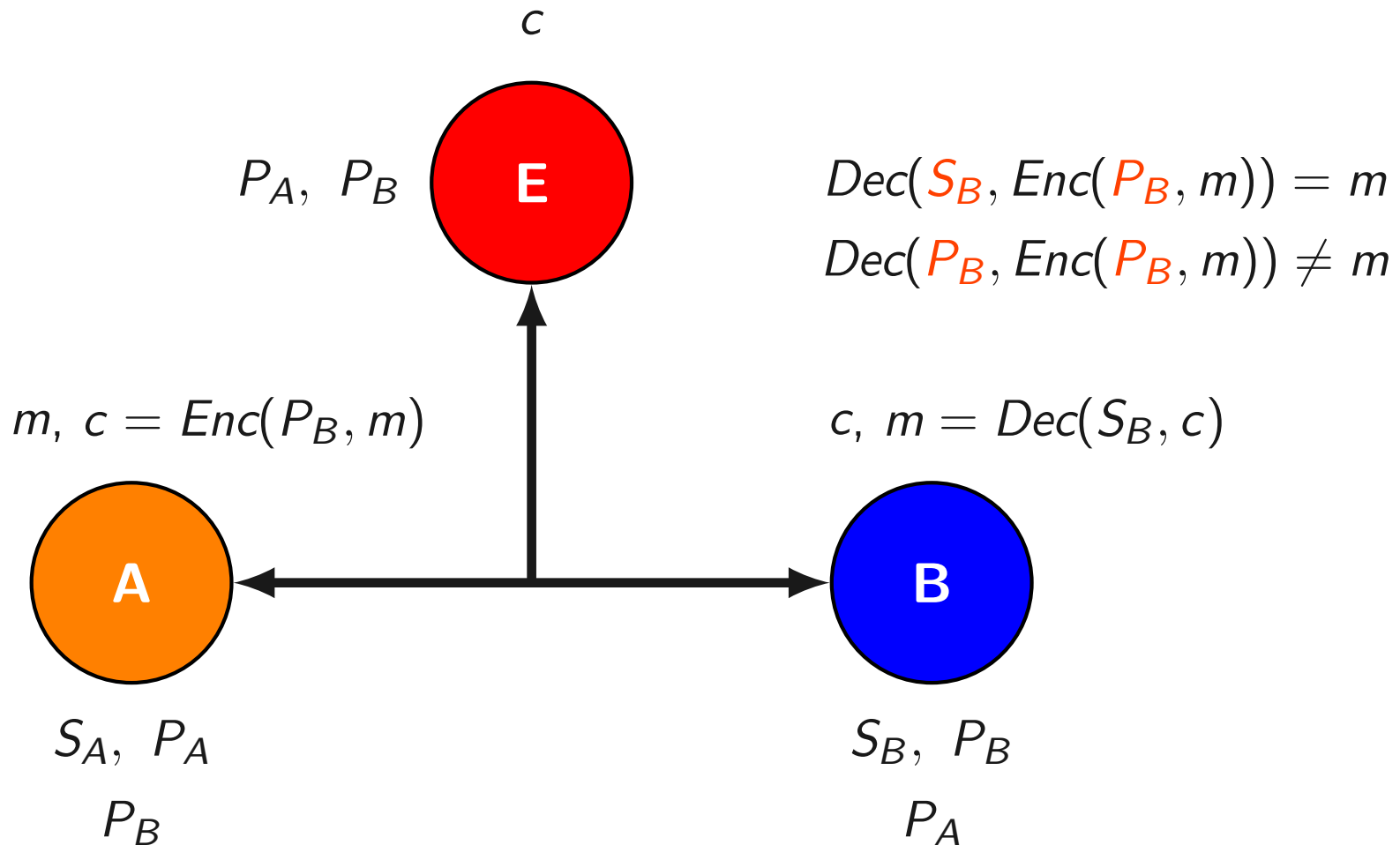


# El escenario moderno: criptografía de clave pública





# El escenario moderno: criptografía de clave pública



# El sistema criptográfico RSA

Un usuario  $A$  ejecuta los siguientes pasos para construir sus claves pública  $P_A$  y secreta  $S_A$ .

# El sistema criptográfico RSA

Un usuario  $A$  ejecuta los siguientes pasos para construir sus claves pública  $P_A$  y secreta  $S_A$ .

1. Genera al azar dos números primos distintos  $P$  y  $Q$

# El sistema criptográfico RSA

Un usuario  $A$  ejecuta los siguientes pasos para construir sus claves pública  $P_A$  y secreta  $S_A$ .

1. Genera al azar dos números primos distintos  $P$  y  $Q$
2. Define  $N = P \cdot Q$  y  $\varphi(N) = (P - 1) \cdot (Q - 1)$

# El sistema criptográfico RSA

Un usuario  $A$  ejecuta los siguientes pasos para construir sus claves pública  $P_A$  y secreta  $S_A$ .

1. Genera al azar dos números primos distintos  $P$  y  $Q$
2. Define  $N = P \cdot Q$  y  $\varphi(N) = (P - 1) \cdot (Q - 1)$
3. Genera al azar un número  $d \in \{0, \dots, \varphi(N)\}$  tal que  $\text{MCD}(d, \varphi(N)) = 1$

# El sistema criptográfico RSA

Un usuario  $A$  ejecuta los siguientes pasos para construir sus claves pública  $P_A$  y secreta  $S_A$ .

1. Genera al azar dos números primos distintos  $P$  y  $Q$
2. Define  $N = P \cdot Q$  y  $\varphi(N) = (P - 1) \cdot (Q - 1)$
3. Genera al azar un número  $d \in \{0, \dots, \varphi(N)\}$  tal que  $\text{MCD}(d, \varphi(N)) = 1$
4. Calcula un número  $e$  tal que  $(e \cdot d) \bmod \varphi(N) = 1$

# El sistema criptográfico RSA

Un usuario  $A$  ejecuta los siguientes pasos para construir sus claves pública  $P_A$  y secreta  $S_A$ .

1. Genera al azar dos números primos distintos  $P$  y  $Q$
2. Define  $N = P \cdot Q$  y  $\varphi(N) = (P - 1) \cdot (Q - 1)$
3. Genera al azar un número  $d \in \{0, \dots, \varphi(N)\}$  tal que  $\text{MCD}(d, \varphi(N)) = 1$
4. Calcula un número  $e$  tal que  $(e \cdot d) \bmod \varphi(N) = 1$
5. Define  $P_A = (e, N)$  y  $S_A = (d, N)$

# El sistema criptográfico RSA

Sea  $A$  un usuario con claves  $P_A = (e, N)$  y  $S_A = (d, N)$ .



# El sistema criptográfico RSA

Sea  $A$  un usuario con claves  $P_A = (e, N)$  y  $S_A = (d, N)$ .

Para cifrar un mensaje con la clave pública de  $A$  se utiliza la siguiente función.  
Dado  $m \in \{0, \dots, N - 1\}$ :

$$Enc(P_A, m) = m^e \bmod N$$

# El sistema criptográfico RSA

Sea  $A$  un usuario con claves  $P_A = (e, N)$  y  $S_A = (d, N)$ .

Para cifrar un mensaje con la clave pública de  $A$  se utiliza la siguiente función. Dado  $m \in \{0, \dots, N - 1\}$ :

$$Enc(P_A, m) = m^e \bmod N$$

Para descifrar un mensaje con la clave privada de  $A$  se utiliza la siguiente función. Dado  $m \in \{0, \dots, N - 1\}$ :

$$Dec(S_A, m) = m^d \bmod N$$

# El sistema criptográfico RSA

Ejemplo

Sean  $P = 7$  y  $Q = 11$ .

# El sistema criptográfico RSA

## Ejemplo

Sean  $P = 7$  y  $Q = 11$ .

- ▶ Se tiene que  $N = 77$  y  $\varphi(N) = 60$ .

# El sistema criptográfico RSA

## Ejemplo

Sean  $P = 7$  y  $Q = 11$ .

▶ Se tiene que  $N = 77$  y  $\varphi(N) = 60$ .

Sean  $d = 37$ .

# El sistema criptográfico RSA

## Ejemplo

Sean  $P = 7$  y  $Q = 11$ .

- ▶ Se tiene que  $N = 77$  y  $\varphi(N) = 60$ .

Sean  $d = 37$ .

- ▶ Se tiene que  $\text{MCD}(37, 60) = 1$ .

# El sistema criptográfico RSA

## Ejemplo

Sean  $P = 7$  y  $Q = 11$ .

- ▶ Se tiene que  $N = 77$  y  $\varphi(N) = 60$ .

Sean  $d = 37$ .

- ▶ Se tiene que  $\text{MCD}(37, 60) = 1$ .

Sean  $e = 13$ .

# El sistema criptográfico RSA

## Ejemplo

Sean  $P = 7$  y  $Q = 11$ .

- ▶ Se tiene que  $N = 77$  y  $\varphi(N) = 60$ .

Sean  $d = 37$ .

- ▶ Se tiene que  $\text{MCD}(37, 60) = 1$ .

Sean  $e = 13$ .

- ▶ Se tiene que  $(13 \cdot 37) \bmod 60 = 1$ .



# El sistema criptográfico RSA

## Ejemplo

Sean  $P = 7$  y  $Q = 11$ .

- ▶ Se tiene que  $N = 77$  y  $\varphi(N) = 60$ .

Sean  $d = 37$ .

- ▶ Se tiene que  $\text{MCD}(37, 60) = 1$ .

Sean  $e = 13$ .

- ▶ Se tiene que  $(13 \cdot 37) \bmod 60 = 1$ .

Definimos  $P_A = (13, 77)$  y  $S_A = (37, 77)$

# El sistema criptográfico RSA

Ejemplo (continuación)

Ciframos y desciframos un mensaje  $m \in \{0, \dots, 76\}$  de la siguiente forma:

# El sistema criptográfico RSA

Ejemplo (continuación)

Ciframos y desciframos un mensaje  $m \in \{0, \dots, 76\}$  de la siguiente forma:

$$Enc(P_A, m) = m^{13} \bmod 77$$

# El sistema criptográfico RSA

Ejemplo (continuación)

Ciframos y desciframos un mensaje  $m \in \{0, \dots, 76\}$  de la siguiente forma:

$$Enc(P_A, m) = m^{13} \bmod 77$$

$$Dec(S_A, m) = m^{37} \bmod 77$$

# El sistema criptográfico RSA

Ejemplo (continuación)

Ciframos y desciframos un mensaje  $m \in \{0, \dots, 76\}$  de la siguiente forma:

$$Enc(P_A, m) = m^{13} \bmod 77$$

$$Dec(S_A, m) = m^{37} \bmod 77$$

El sistema funciona correctamente:

# El sistema criptográfico RSA

Ejemplo (continuación)

Ciframos y desciframos un mensaje  $m \in \{0, \dots, 76\}$  de la siguiente forma:

$$Enc(P_A, m) = m^{13} \bmod 77$$

$$Dec(S_A, m) = m^{37} \bmod 77$$

El sistema funciona correctamente:

$$Enc(P_A, 5) = 5^{13} \bmod 77 = 26$$

# El sistema criptográfico RSA

Ejemplo (continuación)

Ciframos y desciframos un mensaje  $m \in \{0, \dots, 76\}$  de la siguiente forma:

$$Enc(P_A, m) = m^{13} \bmod 77$$

$$Dec(S_A, m) = m^{37} \bmod 77$$

El sistema funciona correctamente:

$$Enc(P_A, 5) = 5^{13} \bmod 77 = 26$$

$$Dec(S_A, 26) = 26^{37} \bmod 77 = 5$$

# ¿Por qué funciona RSA?

¿Qué propiedades debemos demostrar que son ciertas?



# ¿Por qué funciona RSA?

¿Qué propiedades debemos demostrar que son ciertas?

- ▶  $Dec(S_A, Enc(P_A, m)) = m$  para todo  $m \in \{0, \dots, N - 1\}$

# ¿Por qué funciona RSA?

¿Qué propiedades debemos demostrar que son ciertas?

▶  $Dec(S_A, Enc(P_A, m)) = m$  para todo  $m \in \{0, \dots, N - 1\}$

¿Qué problemas debemos demostrar que pueden ser resueltos de manera eficiente para que el sistema puede ser utilizado?

# ¿Por qué funciona RSA?

¿Qué propiedades debemos demostrar que son ciertas?

- ▶  $Dec(S_A, Enc(P_A, m)) = m$  para todo  $m \in \{0, \dots, N - 1\}$

¿Qué problemas debemos demostrar que pueden ser resueltos de manera eficiente para que el sistema puede ser utilizado?

- ▶ Generar primos distintos  $P$  y  $Q$ .

# ¿Por qué funciona RSA?

¿Qué propiedades debemos demostrar que son ciertas?

- ▶  $Dec(S_A, Enc(P_A, m)) = m$  para todo  $m \in \{0, \dots, N - 1\}$

¿Qué problemas debemos demostrar que pueden ser resueltos de manera eficiente para que el sistema puede ser utilizado?

- ▶ Generar primos distintos  $P$  y  $Q$ .
- ▶ Generar un número  $d$  tal que  $MCD(d, \varphi(N)) = 1$ .

# ¿Por qué funciona RSA?

¿Qué propiedades debemos demostrar que son ciertas?

- ▶  $Dec(S_A, Enc(P_A, m)) = m$  para todo  $m \in \{0, \dots, N - 1\}$

¿Qué problemas debemos demostrar que pueden ser resueltos de manera eficiente para que el sistema puede ser utilizado?

- ▶ Generar primos distintos  $P$  y  $Q$ .
- ▶ Generar un número  $d$  tal que  $\text{MCD}(d, \varphi(N)) = 1$ .
- ▶ Generar un número  $e$  tal que  $(e \cdot d) \bmod \varphi(N) = 1$ .

# ¿Por qué funciona RSA?

¿Qué propiedades debemos demostrar que son ciertas?

- ▶  $Dec(S_A, Enc(P_A, m)) = m$  para todo  $m \in \{0, \dots, N - 1\}$

¿Qué problemas debemos demostrar que pueden ser resueltos de manera eficiente para que el sistema puede ser utilizado?

- ▶ Generar primos distintos  $P$  y  $Q$ .
- ▶ Generar un número  $d$  tal que  $\text{MCD}(d, \varphi(N)) = 1$ .
- ▶ Generar un número  $e$  tal que  $(e \cdot d) \bmod \varphi(N) = 1$ .
- ▶ Calcular funciones  $Enc$  y  $Dec$ .

# ¿Por qué funciona RSA?

¿Qué propiedades debemos demostrar que son ciertas?

- ▶  $Dec(S_A, Enc(P_A, m)) = m$  para todo  $m \in \{0, \dots, N - 1\}$

¿Qué problemas debemos demostrar que pueden ser resueltos de manera eficiente para que el sistema puede ser utilizado?

- ▶ Generar primos distintos  $P$  y  $Q$ .
- ▶ Generar un número  $d$  tal que  $\text{MCD}(d, \varphi(N)) = 1$ .
- ▶ Generar un número  $e$  tal que  $(e \cdot d) \bmod \varphi(N) = 1$ .
- ▶ Calcular funciones  $Enc$  y  $Dec$ .

¿Qué problemas no pueden ser resueltos de manera eficiente para que el sistema sea seguro?

- ▶ Dado  $(e, N)$  calcular  $d$ , lo cual se reduce a encontrar los divisores de  $N$ .

# RSA funciona correctamente

Sean  $P_A$ ,  $S_A$ ,  $Enc$  y  $Dec$  definidas como en las láminas anteriores.



# RSA funciona correctamente

Sean  $P_A$ ,  $S_A$ ,  $Enc$  y  $Dec$  definidas como en las láminas anteriores.

- ▶ En particular  $P_A = (e, N)$  y  $S_A = (d, N)$ , con  $N = P \cdot Q$ .

# RSA funciona correctamente

Sean  $P_A$ ,  $S_A$ ,  $Enc$  y  $Dec$  definidas como en las láminas anteriores.

- ▶ En particular  $P_A = (e, N)$  y  $S_A = (d, N)$ , con  $N = P \cdot Q$ .

## Teorema (Rivest-Shamir-Adleman)

Para cada  $m \in \{0, \dots, N - 1\}$ , se tiene que  $Dec(S_A, Dec(P_A, m)) = m$ .

# RSA funciona correctamente

Sean  $P_A$ ,  $S_A$ ,  $Enc$  y  $Dec$  definidas como en las láminas anteriores.

- ▶ En particular  $P_A = (e, N)$  y  $S_A = (d, N)$ , con  $N = P \cdot Q$ .

## Teorema (Rivest-Shamir-Adleman)

Para cada  $m \in \{0, \dots, N - 1\}$ , se tiene que  $Dec(S_A, Dec(P_A, m)) = m$ .

**Demostración:** Sabemos que

$$\begin{aligned} Dec(S_A, Enc(P_A, m)) &= (m^e \bmod N)^d \bmod N \\ &= (m^e)^d \bmod N \\ &= m^{e \cdot d} \bmod N \end{aligned}$$

# RSA funciona correctamente

Sean  $P_A$ ,  $S_A$ ,  $Enc$  y  $Dec$  definidas como en las láminas anteriores.

- ▶ En particular  $P_A = (e, N)$  y  $S_A = (d, N)$ , con  $N = P \cdot Q$ .

## Teorema (Rivest-Shamir-Adleman)

Para cada  $m \in \{0, \dots, N - 1\}$ , se tiene que  $Dec(S_A, Dec(P_A, m)) = m$ .

**Demostración:** Sabemos que

$$\begin{aligned} Dec(S_A, Enc(P_A, m)) &= (m^e \bmod N)^d \bmod N \\ &= (m^e)^d \bmod N \\ &= m^{e \cdot d} \bmod N \end{aligned}$$

Por lo tanto, tenemos que demostrar que  $m^{e \cdot d} \equiv_N m$

# RSA funciona correctamente: Demostración

Sabemos que  $(e \cdot d) \bmod \varphi(N) = 1$ .

# RSA funciona correctamente: Demostración

Sabemos que  $(e \cdot d) \bmod \varphi(N) = 1$ .

▶ Por lo tanto,  $e \cdot d = k \cdot \varphi(N) + 1$ .

# RSA funciona correctamente: Demostración

Sabemos que  $(e \cdot d) \bmod \varphi(N) = 1$ .

▶ Por lo tanto,  $e \cdot d = k \cdot \varphi(N) + 1$ .

Tenemos que demostrar que  $m^{k \cdot \varphi(N) + 1} \equiv_N m$ .

# RSA funciona correctamente: Demostración

Sabemos que  $(e \cdot d) \bmod \varphi(N) = 1$ .

▶ Por lo tanto,  $e \cdot d = k \cdot \varphi(N) + 1$ .

Tenemos que demostrar que  $m^{k \cdot \varphi(N) + 1} \equiv_N m$ .

▶ El siguiente lema es fundamental para la demostración.



# RSA funciona correctamente: Demostración

Sabemos que  $(e \cdot d) \bmod \varphi(N) = 1$ .

▶ Por lo tanto,  $e \cdot d = k \cdot \varphi(N) + 1$ .

Tenemos que demostrar que  $m^{k \cdot \varphi(N) + 1} \equiv_N m$ .

▶ El siguiente lema es fundamental para la demostración.

Lema

$$m^{k \cdot \varphi(N) + 1} \equiv_P m \text{ y } m^{k \cdot \varphi(N) + 1} \equiv_Q m$$

# RSA funciona correctamente: Demostración

Sabemos que  $(e \cdot d) \bmod \varphi(N) = 1$ .

▶ Por lo tanto,  $e \cdot d = k \cdot \varphi(N) + 1$ .

Tenemos que demostrar que  $m^{k \cdot \varphi(N) + 1} \equiv_N m$ .

▶ El siguiente lema es fundamental para la demostración.

## Lema

$$m^{k \cdot \varphi(N) + 1} \equiv_P m \text{ y } m^{k \cdot \varphi(N) + 1} \equiv_Q m$$

**Demostración:** Primero suponemos que  $P \mid m$ .

# RSA funciona correctamente: Demostración

$$\begin{aligned} \text{Entonces: } m^{k \cdot \varphi(N) + 1} \bmod P &= (m \bmod P)^{k \cdot \varphi(N) + 1} \bmod P \\ &= 0^{k \cdot \varphi(N) + 1} \bmod P \\ &= 0 \end{aligned}$$

# RSA funciona correctamente: Demostración

$$\begin{aligned}\text{Entonces: } m^{k \cdot \varphi(N) + 1} \bmod P &= (m \bmod P)^{k \cdot \varphi(N) + 1} \bmod P \\ &= 0^{k \cdot \varphi(N) + 1} \bmod P \\ &= 0\end{aligned}$$

Por lo tanto:  $m^{k \cdot \varphi(N) + 1} \equiv_P m$ .

# RSA funciona correctamente: Demostración

$$\begin{aligned}\text{Entonces: } m^{k \cdot \varphi(N) + 1} \bmod P &= (m \bmod P)^{k \cdot \varphi(N) + 1} \bmod P \\ &= 0^{k \cdot \varphi(N) + 1} \bmod P \\ &= 0\end{aligned}$$

Por lo tanto:  $m^{k \cdot \varphi(N) + 1} \equiv_P m$ .

En segundo lugar, suponemos que  $P \nmid m$ .

# RSA funciona correctamente: Demostración

$$\begin{aligned}\text{Entonces: } m^{k \cdot \varphi(N)+1} \bmod P &= (m \bmod P)^{k \cdot \varphi(N)+1} \bmod P \\ &= 0^{k \cdot \varphi(N)+1} \bmod P \\ &= 0\end{aligned}$$

Por lo tanto:  $m^{k \cdot \varphi(N)+1} \equiv_P m$ .

En segundo lugar, suponemos que  $P \nmid m$ .

Dado que  $m \not\equiv_P 0$ , por pequeño teorema de Fermat:

$$m^{P-1} \equiv_P 1$$

# RSA funciona correctamente: Demostración

De esto concluimos que:

$$\begin{aligned} m^{k \cdot \varphi(N) + 1} \bmod N &= ((m^{P-1})^{k \cdot (Q-1)} \cdot m) \bmod P \\ &= ((m^{P-1} \bmod P)^{k \cdot (Q-1)} \cdot m) \bmod P \\ &= (1^{k \cdot (Q-1)} \cdot m) \bmod P \\ &= m \bmod P \end{aligned}$$

# RSA funciona correctamente: Demostración

De esto concluimos que:

$$\begin{aligned} m^{k \cdot \varphi(N) + 1} \bmod N &= ((m^{P-1})^{k \cdot (Q-1)} \cdot m) \bmod P \\ &= ((m^{P-1} \bmod P)^{k \cdot (Q-1)} \cdot m) \bmod P \\ &= (1^{k \cdot (Q-1)} \cdot m) \bmod P \\ &= m \bmod P \end{aligned}$$

Concluimos que  $m^{k \cdot \varphi(N) + 1} \equiv_P m$ .



# RSA funciona correctamente: Demostración

De esto concluimos que:

$$\begin{aligned} m^{k \cdot \varphi(N) + 1} \bmod N &= ((m^{P-1})^{k \cdot (Q-1)} \cdot m) \bmod P \\ &= ((m^{P-1} \bmod P)^{k \cdot (Q-1)} \cdot m) \bmod P \\ &= (1^{k \cdot (Q-1)} \cdot m) \bmod P \\ &= m \bmod P \end{aligned}$$

Concluimos que  $m^{k \cdot \varphi(N) + 1} \equiv_P m$ .

▶ De la misma forma se demuestra que  $m^{k \cdot \varphi(N) + 1} \equiv_Q m$ .



# RSA funciona correctamente: Demostración

Del lema concluimos que:

$$m^{k \cdot \varphi(N) + 1} - m = \alpha \cdot P$$

$$m^{k \cdot \varphi(N) + 1} - m = \beta \cdot Q$$

# RSA funciona correctamente: Demostración

Del lema concluimos que:

$$\begin{aligned} m^{k \cdot \varphi(N) + 1} - m &= \alpha \cdot P \\ m^{k \cdot \varphi(N) + 1} - m &= \beta \cdot Q \end{aligned}$$

Por lo tanto,  $\alpha \cdot P = \beta \cdot Q$ . Tenemos entonces que  $P | (\beta \cdot Q)$ .

# RSA funciona correctamente: Demostración

Del lema concluimos que:

$$\begin{aligned} m^{k \cdot \varphi(N) + 1} - m &= \alpha \cdot P \\ m^{k \cdot \varphi(N) + 1} - m &= \beta \cdot Q \end{aligned}$$

Por lo tanto,  $\alpha \cdot P = \beta \cdot Q$ . Tenemos entonces que  $P | (\beta \cdot Q)$ .

Entonces, dado que  $P$  y  $Q$  son primos distintos tenemos que  $P | \beta$ .

# RSA funciona correctamente: Demostración

Del lema concluimos que:

$$\begin{aligned} m^{k \cdot \varphi(N) + 1} - m &= \alpha \cdot P \\ m^{k \cdot \varphi(N) + 1} - m &= \beta \cdot Q \end{aligned}$$

Por lo tanto,  $\alpha \cdot P = \beta \cdot Q$ . Tenemos entonces que  $P | (\beta \cdot Q)$ .

Entonces, dado que  $P$  y  $Q$  son primos distintos tenemos que  $P | \beta$ .

► Por lo tanto,  $\beta = \gamma \cdot P$ .

# RSA funciona correctamente: Demostración

Del lema concluimos que:

$$\begin{aligned} m^{k \cdot \varphi(N)+1} - m &= \alpha \cdot P \\ m^{k \cdot \varphi(N)+1} - m &= \beta \cdot Q \end{aligned}$$

Por lo tanto,  $\alpha \cdot P = \beta \cdot Q$ . Tenemos entonces que  $P | (\beta \cdot Q)$ .

Entonces, dado que  $P$  y  $Q$  son primos distintos tenemos que  $P | \beta$ .

► Por lo tanto,  $\beta = \gamma \cdot P$ .

Concluimos que  $m^{k \cdot \varphi(N)+1} - m = \gamma \cdot P \cdot Q$ .

# RSA funciona correctamente: Demostración

Del lema concluimos que:

$$\begin{aligned} m^{k \cdot \varphi(N)+1} - m &= \alpha \cdot P \\ m^{k \cdot \varphi(N)+1} - m &= \beta \cdot Q \end{aligned}$$

Por lo tanto,  $\alpha \cdot P = \beta \cdot Q$ . Tenemos entonces que  $P | (\beta \cdot Q)$ .

Entonces, dado que  $P$  y  $Q$  son primos distintos tenemos que  $P | \beta$ .

► Por lo tanto,  $\beta = \gamma \cdot P$ .

Concluimos que  $m^{k \cdot \varphi(N)+1} - m = \gamma \cdot P \cdot Q$ .

► Vale decir,  $m^{k \cdot \varphi(N)+1} \equiv_N m$ .



# RSA: comentarios finales

¿Cómo se pueden resolver los siguientes problemas de manera eficiente?



# RSA: comentarios finales

¿Cómo se pueden resolver los siguientes problemas de manera eficiente?

1. Generar primos distintos  $P$  y  $Q$ .

# RSA: comentarios finales

¿Cómo se pueden resolver los siguientes problemas de manera eficiente?

1. Generar primos distintos  $P$  y  $Q$ .
2. Generar un número  $d$  tal que  $\text{MCD}(d, \varphi(N)) = 1$ .

# RSA: comentarios finales

¿Cómo se pueden resolver los siguientes problemas de manera eficiente?

1. Generar primos distintos  $P$  y  $Q$ .
2. Generar un número  $d$  tal que  $\text{MCD}(d, \varphi(N)) = 1$ .
3. Generar un número  $e$  tal que  $(e \cdot d) \bmod \varphi(N) = 1$ .

# RSA: comentarios finales

¿Cómo se pueden resolver los siguientes problemas de manera eficiente?

1. Generar primos distintos  $P$  y  $Q$ .
2. Generar un número  $d$  tal que  $\text{MCD}(d, \varphi(N)) = 1$ .
3. Generar un número  $e$  tal que  $(e \cdot d) \bmod \varphi(N) = 1$ .
4. Calcular las funciones  $Enc$  y  $Dec$ .

# RSA: comentarios finales

Los problemas 1. y 2. están fuera del alcance de este curso.

# RSA: comentarios finales

Los problemas 1. y 2. están fuera del alcance de este curso.

- ▶ Son estudiados en IIC3253 Criptografía y Seguridad Computacional.

# RSA: comentarios finales

Los problemas 1. y 2. están fuera del alcance de este curso.

- ▶ Son estudiados en IIC3253 Criptografía y Seguridad Computacional.

El problema 3. ya sabemos cómo resolverlo usando el algoritmo extendido de Euclides.

# RSA: comentarios finales

Los problemas 1. y 2. están fuera del alcance de este curso.

- ▶ Son estudiados en IIC3253 Criptografía y Seguridad Computacional.

El problema 3. ya sabemos cómo resolverlo usando el algoritmo extendido de Euclides.

- ▶ ¿Cómo se hace esto?



# RSA: comentarios finales

Los problemas 1. y 2. están fuera del alcance de este curso.

- ▶ Son estudiados en IIC3253 Criptografía y Seguridad Computacional.

El problema 3. ya sabemos cómo resolverlo usando el algoritmo extendido de Euclides.

- ▶ ¿Cómo se hace esto?

## Ejercicio

De un algoritmo eficiente para resolver el problema 4.

# RSA: comentarios finales

Los problemas 1. y 2. están fuera del alcance de este curso.

- ▶ Son estudiados en IIC3253 Criptografía y Seguridad Computacional.

El problema 3. ya sabemos cómo resolverlo usando el algoritmo extendido de Euclides.

- ▶ ¿Cómo se hace esto?

## Ejercicio

De un algoritmo eficiente para resolver el problema 4.

- ▶ Piense en cómo puede calcular  $a^{100}$  con menos de 99 multiplicaciones.

# RSA: comentarios finales

Los problemas 1. y 2. están fuera del alcance de este curso.

- ▶ Son estudiados en IIC3253 Criptografía y Seguridad Computacional.

El problema 3. ya sabemos cómo resolverlo usando el algoritmo extendido de Euclides.

- ▶ ¿Cómo se hace esto?

## Ejercicio

De un algoritmo eficiente para resolver el problema 4.

- ▶ Piense en cómo puede calcular  $a^{100}$  con menos de 99 multiplicaciones. Luego extienda su idea para calcular  $a^{100} \bmod b$ .