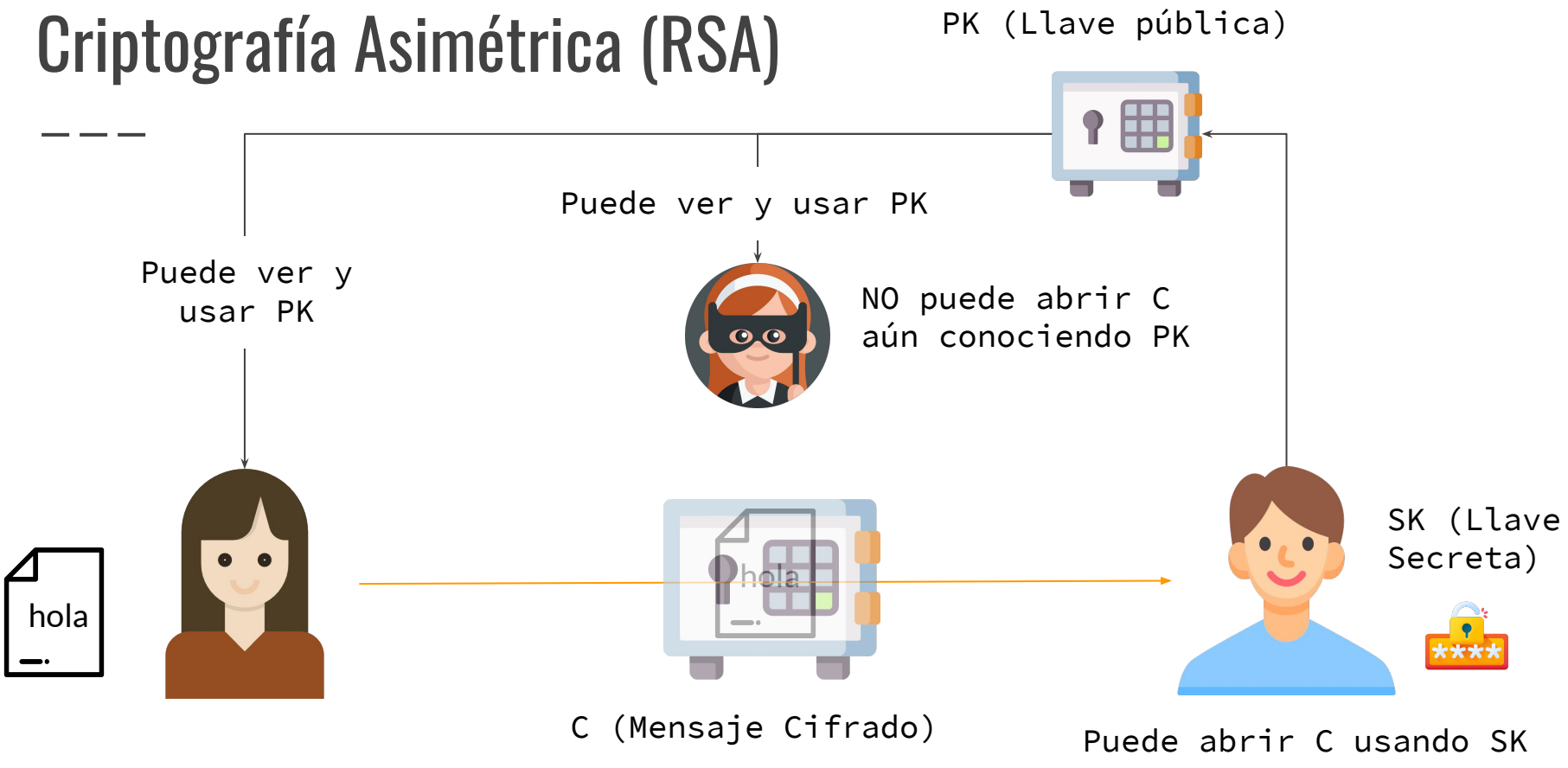
# Criptografía Moderna II: Cripto Asimétrica

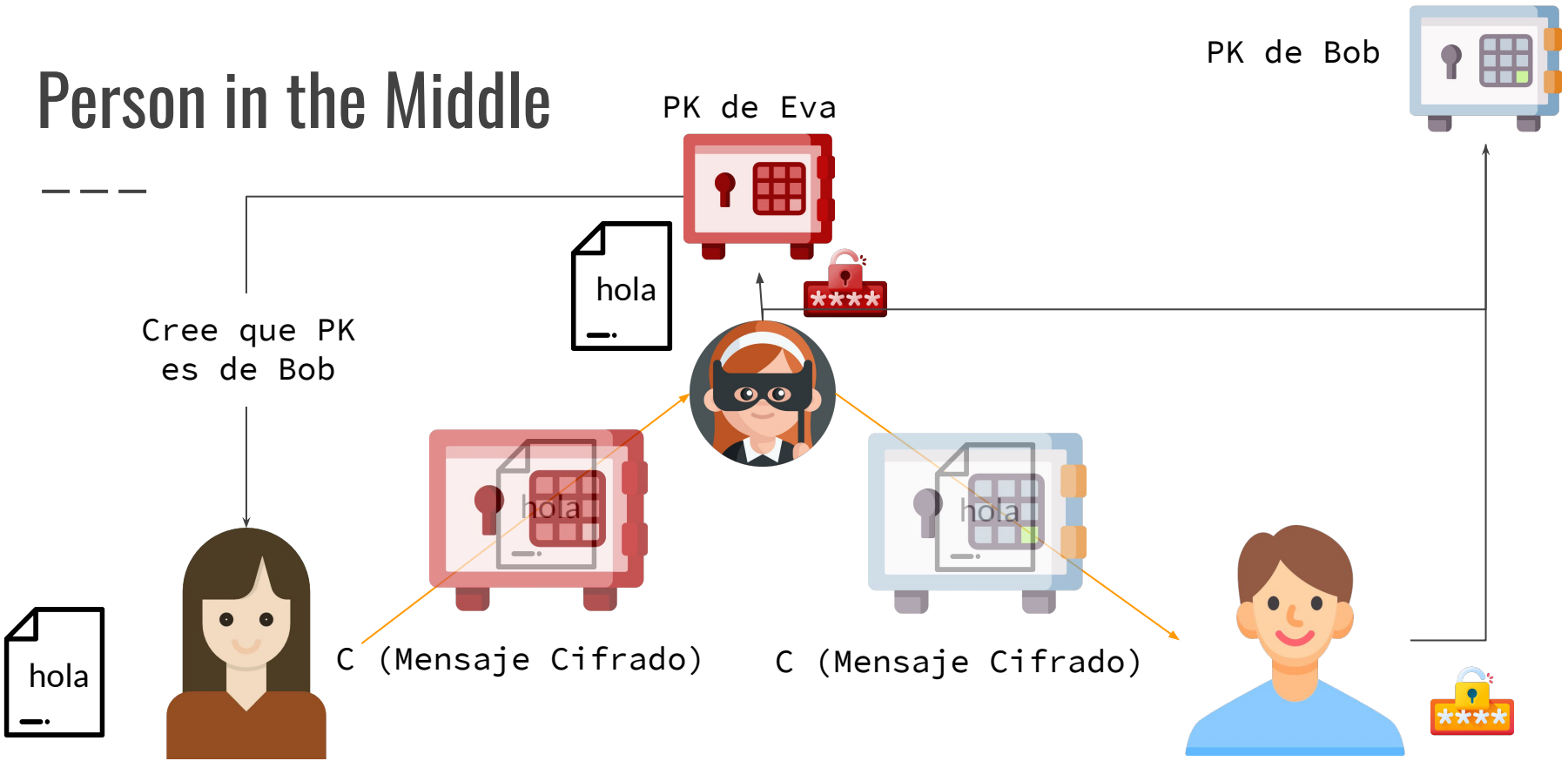
CC5325 - Taller de Hacking Competitivo

# Criptografía Asimétrica (RSA)




Iconos de https://flaticon.com (Freepik)

# Person in the Middle




Iconos de https://flaticon.com (Freepik)

# Cifrado RSA "de libro"

---

$$c = x^e \bmod n \qquad x = c^d \bmod n$$

Mensaje (x) = Lo que queremos cifrar, codificado a un número.

Módulo (**n**) = **pq** (p y q primos grandes, **aleatorios** y **secretos**)
Exp. público **e:** Número menor a **(p-1)(q-1).** Gralmente **65537 o 3**

Exp. Secreto **d** = **1/e mod (p-1)(q-1)** (calculable si conoces **p** y **q**)

Llave pública: **(n, e)**  Llave privada: **d**

# Ejemplo muy básico

$p=7,\ q=5 \Rightarrow n=35,\ r=(p-1)(q-1)=24$

$e=5 \Rightarrow d=29\ (ed=1 \bmod 24)$

cifrado: $x=12 \Rightarrow c=17\ (12^5 \bmod 35)$

descifrado: $c^d \bmod 35=(17^{29} \bmod 35)=12$

# Problemas típicos en cifrado RSA de libro

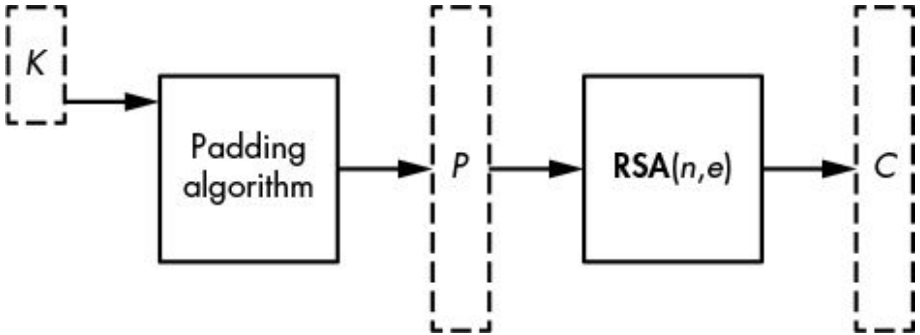
---

- Si **x** y **e** son chicos, puedo sacar raíz e-ésima de **c** y obtener **x**
  - Ej: Mismos parámetros que antes, pero mensaje m es 2
    - => **c**=$2^5$=32
    - **=> puedo obtener x calculando raíz quinta de 32**
      - (Aplica igual para **n** mayores si **x** y **e** son chicos)
- **Maleabilidad:** Si recibo 2 mensajes **x1** y **x2** cifrados con pk=(n, e), puedo calcular la versión cifrada del mensaje **x1*x2** sin conocer **x1** ni **x2**
  - **c1=x1^e mod n, c2=x2^e mod n => c1c2=x1^e*x2^e mod n=(x1*x2)^e mod n.**
- **Valor de c es determinista para un x particular.**

# Padding RSA

**Arregla algunos problemas del RSA de libro:**

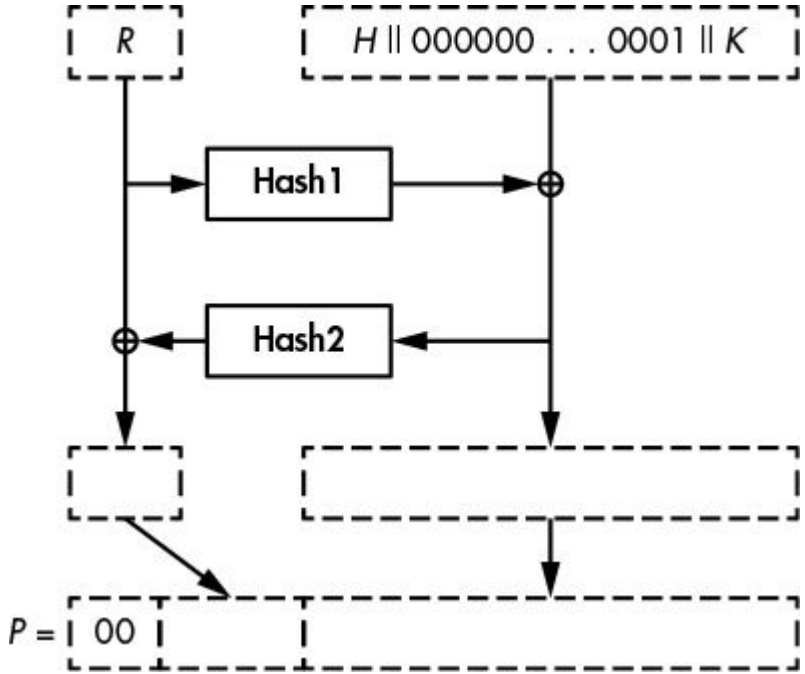
- **Agrega aleatoriedad.**
- **Ahora números siempre son grandes.**




Libro "Serious Cryptography" / Jean-Philippe Amuasson

# Ejemplo de Padding: OAEP

- **H:** Constante estándar de tamaño **h**
- **K:** x
- **M**: **H**||00..00||**K**
- **R:** Número aleatorio de tamaño **h**
- **H||00..00||K** de tamaño igual al de **n**
- **H1, H2:** Funciones de hash
- **P**: valor paddeado y cifrable




Libro "Serious Cryptography" / Jean-Philippe Amuasson

# Problemas típicos en cifrado RSA (en general)

---

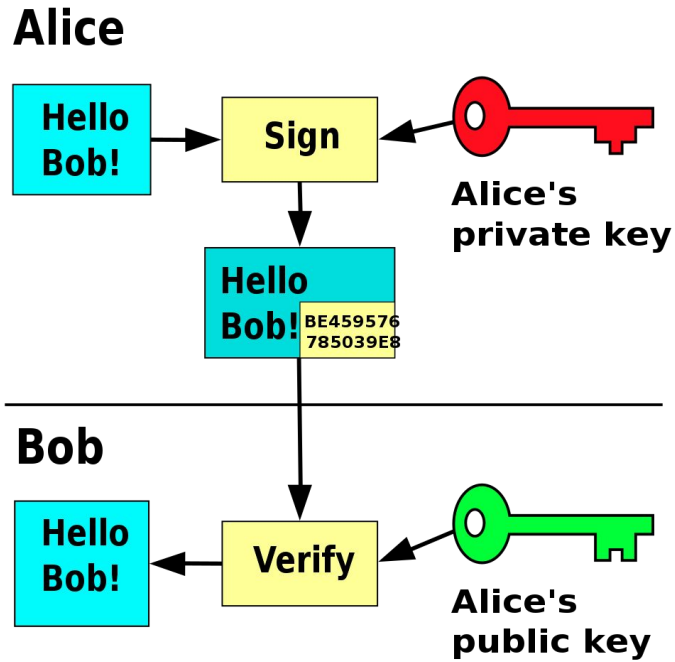
- Si **n** es muy chico (menor a 1024 bits), puedo factorizarlo con un computador potente.
  - El número más largo factorizado es **RSA-250 (250 dígitos, 829 bits)**
  - **Requirió 2700 CPU Core-years para ser factorizado**
    - **1 CPU-Segundo = 1 GigaFLOPS (Floating Point Operations per Second)**
  - **RSA Numbers**
- Si mi generador de números aleatorios es malo y **p** y **q** dependen entre sí, puedo usar esta info para factorizar **n. (Lo veremos en la clase en vivo)**

# Firmas en RSA

---

Podemos usar el mismo esquema criptográfico para generar firmas con la llave pública, validables con la llave privada:

- Calculas el hash **H** de un mensaje **M** si es muy largo
- La firma **S** será **S=H^d mod n**
- Se valida la firma calculando **H** a partir del mensaje **M,** y verificando que **S^e mod n=H**



# Ataques a firmas RSA (si no se usa hash)

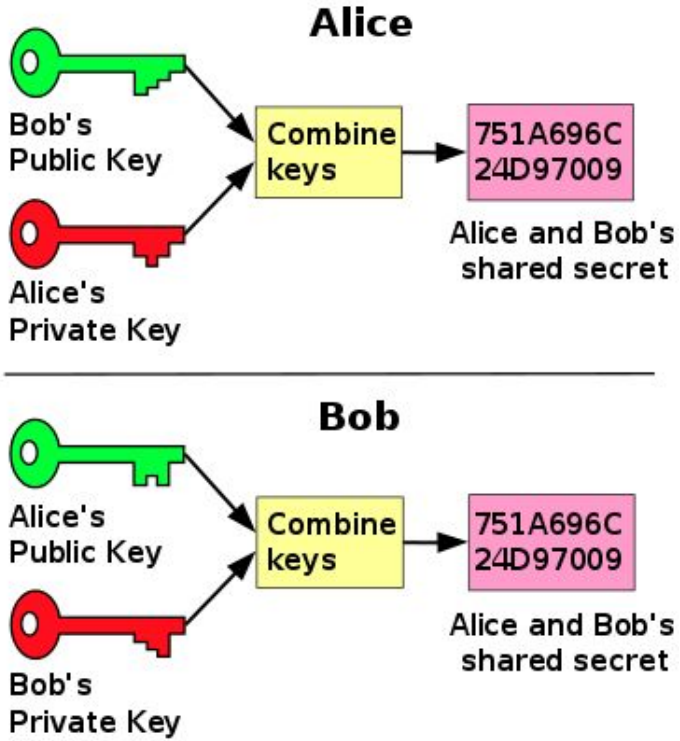
---

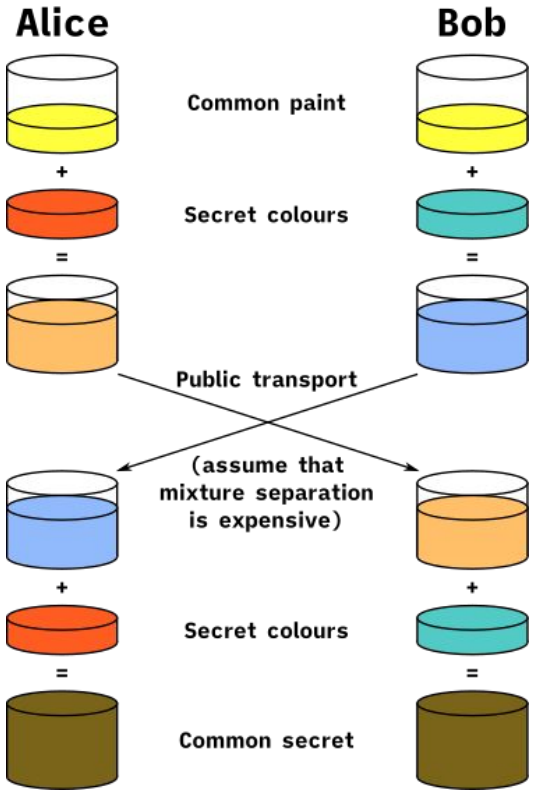
- **Firmas "triviales":** Si se firma el mensaje **M** y no el hash:
  - Si **M** es **0**, **S=0^d mod n=0**
  - Si **M** es **1**, **S=1^d mod n=1**
  - Si **M** es **(n-1)**, **S=(n-1)^d mod n=(n-1)**

Cualquiera puede crear una firma para estos mensajes

- **Blinding Attack:** Hacer que una persona firme un mensaje que no quiere firmar **M**
  - Encontrar **R** tal que **R^eM** sea un mensaje que la persona sí firmaría.
  - **S=(R^eM)^d mod n = R^(ed)M^d = RM^d,** por lo que dividiendo por R obtenemos la firma de **M^d**.

# Acuerdo de llaves Diffie-Hellman

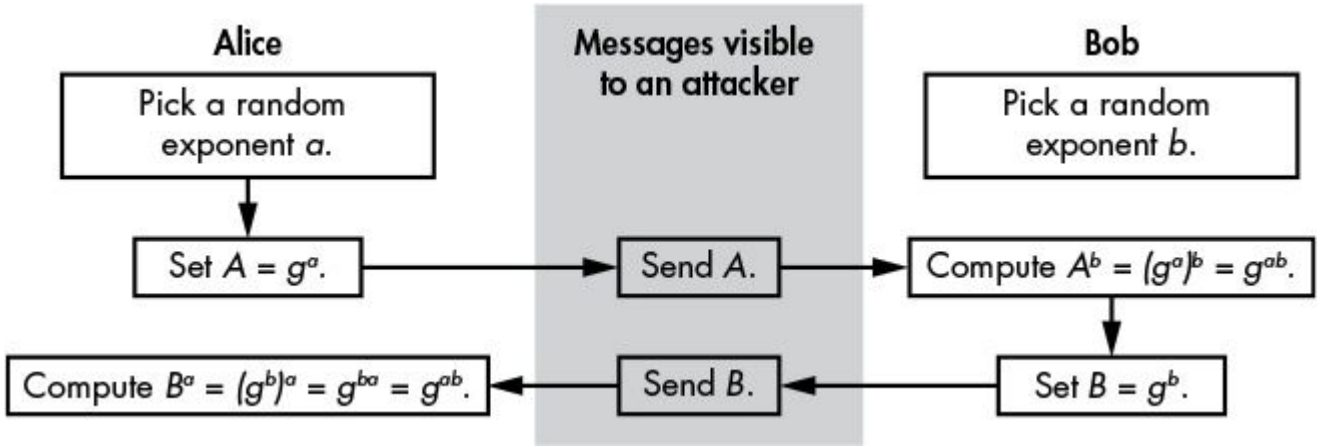


# ¿Y con números?

**p=**primo grande tal que (p-1)/2 también es primo

**g=**Generador En el grupo multiplicativo Z*_p (Generalmente se usa 2)

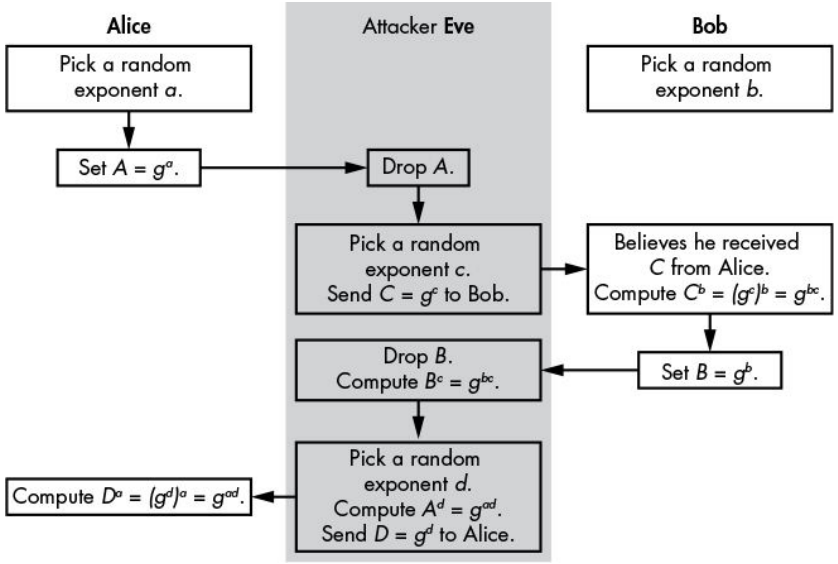



Libro "Serious Cryptography" / Jean-Philippe Amuasson

# Cosas que pueden salir mal en DH de libro:

---

- **Mala aleatoriedad para g^(ab)**
  - **Hashear el secreto sirve para evitar sesgos del grupo.**
- **Parámetros de grupo inseguros**
  - **Si el grupo contiene subgrupos chicos facilita encontrar los valores secretos vía ensayo y error.**
- **Person in the middle**
  - **Autentificar con una llave precompartida.**




Libro "Serious Cryptography" / Jean-Philippe Amuasson

# Conclusiones

- CTFs de Criptografía Moderna: Campo MUY amplio
  - Desafíos serios requieren demasiado conocimiento criptográfico de fondo.
- En CTFs chicos y variados los problemas suelen tratar de cosas más simples
  - Protocolos obsoletos y con ataques conocidos, mal implementados, etc.
- Para adentrarse más, recomendamos nuevamente tomar un curso de criptografía introductorio.