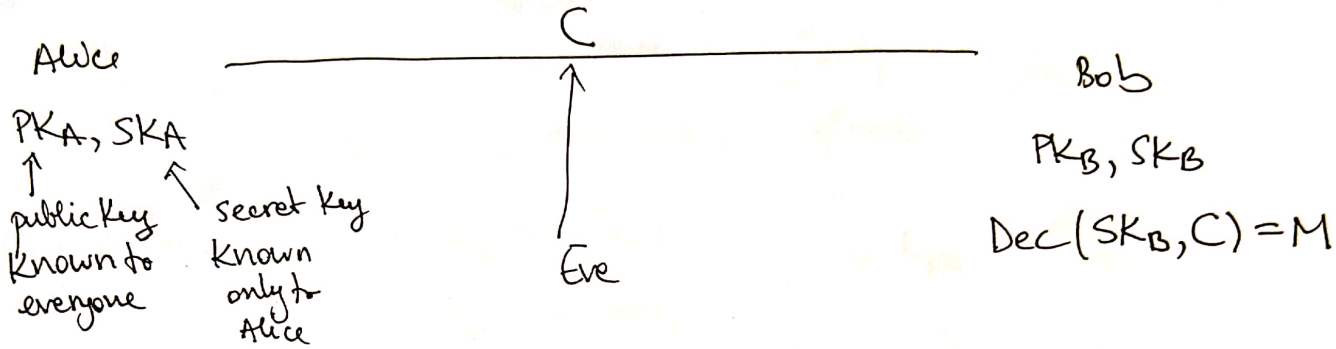
Asymmetric cryptography.



$Enc(PK_B, M) = C$

Syntax:

$Keygen() \to (PK, SK)$

$Enc(PK, m) \to C$

$Dec(SK, C) \to m$

Correctness

$\forall M, \forall SK, PK \leftarrow Keygen(), C \leftarrow Enc(PK, M) : Dec(SK, C) = M$

Security:

One-way functions

A function f is one way if:

(1) Given X, it is easy (poly-time) to compute $f(x)$

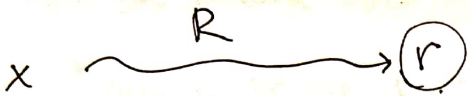
(2) Given y, it is hard (no poly time machine) to find any x s.t. $f(X) = y$.

$f(X) = X$ No, easy to invert

$f(X) = 1$ No, any x leads to 1

$f(X) = E_k(X)$ YES because it is indisti. from random permutation

$f = E_k$ black box don't have key



E_k is indistinguishable from a random permutation

x, $E_k(x) = y$.

Discrete Logarithm Problem (DLP)

$f(X) = \underbrace{g^X \bmod p}_{y}$ where p is a large prime (2048 bits long)

g is a random value in $[2, p-1]$

Assumption: f_{DLP} is OWF

Easy to compute:

Say X is 2048-bit large number.

$\leq 2^{2048}$ $\quad 2^{128}$

repeated squaring

2^{32}

$2^{16} \cdot 2^{16}$

1 mult

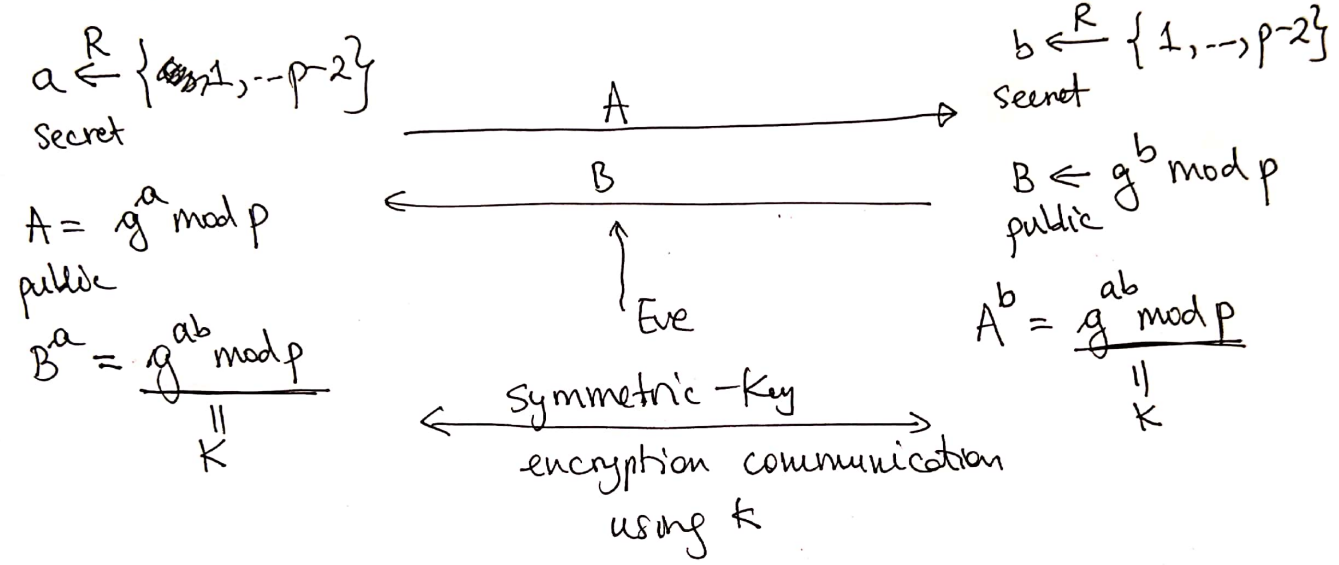
$2^{16} \cdot 2 \cdot 2 \cdot 2 \cdots$

16

Diffie-Hellman Key Exchange (1976)

(Turing award)

large prime p, $1 < g < p-1$ — public



Eve sees: $A = g^a \bmod p \Rightarrow$ cannot compute a

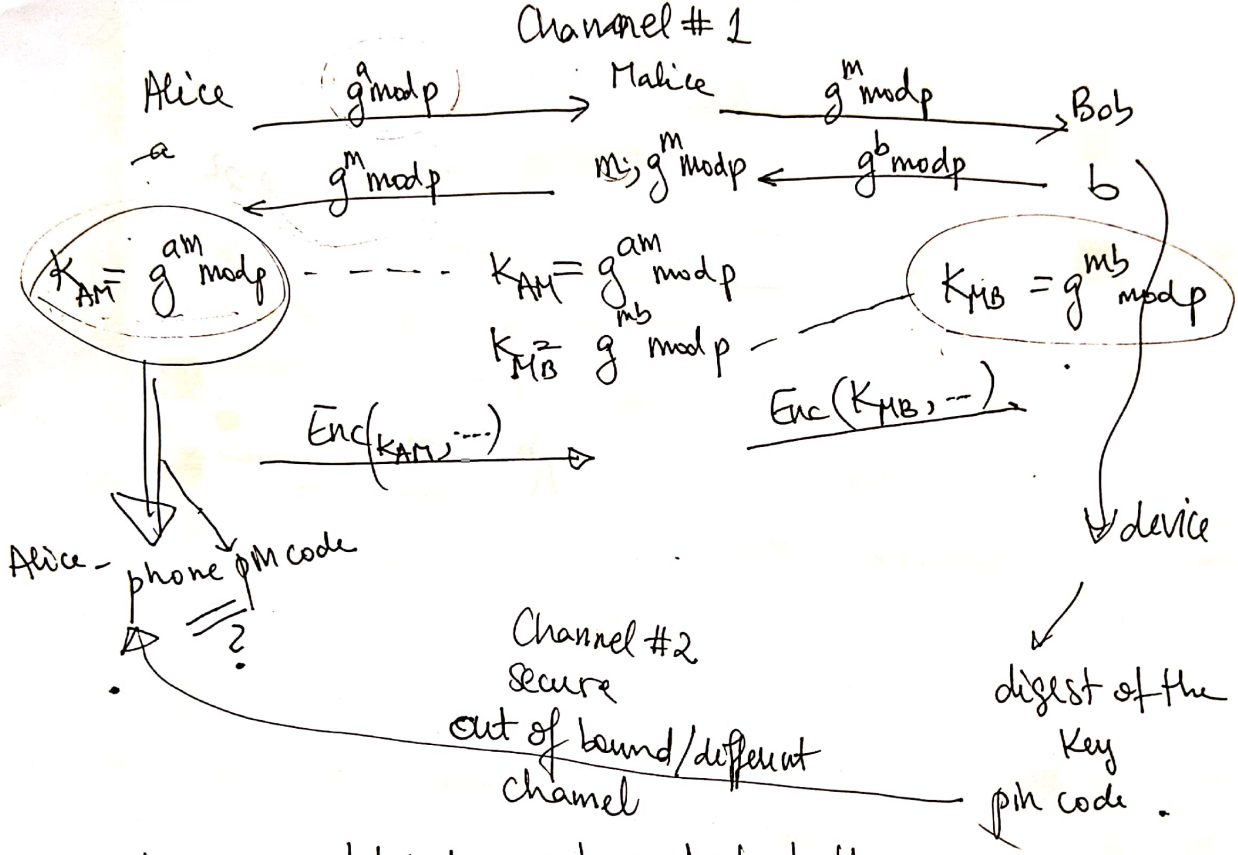
$B = g^b \bmod p \Rightarrow$ cannot compute b

} cannot compute g^{ab}

Assumption: you cannot break DLP (DLP is OWF) ← necessary

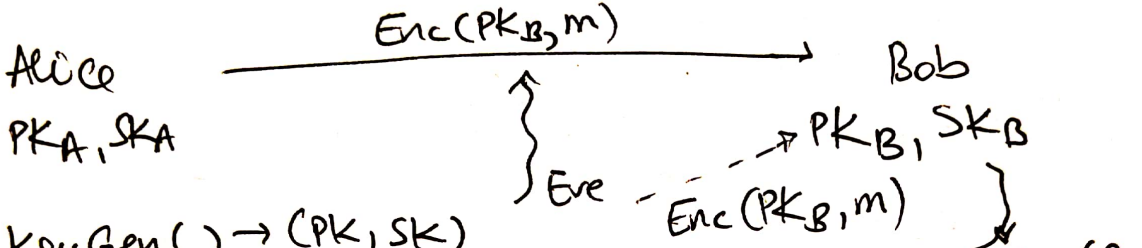
Adv cannot compute $g^{ab} \bmod p$ from $g, g^a, g^b \bmod p$

Man in the middle attack (MITM)



Assumes Adv does not control both channels

Public-key encryption



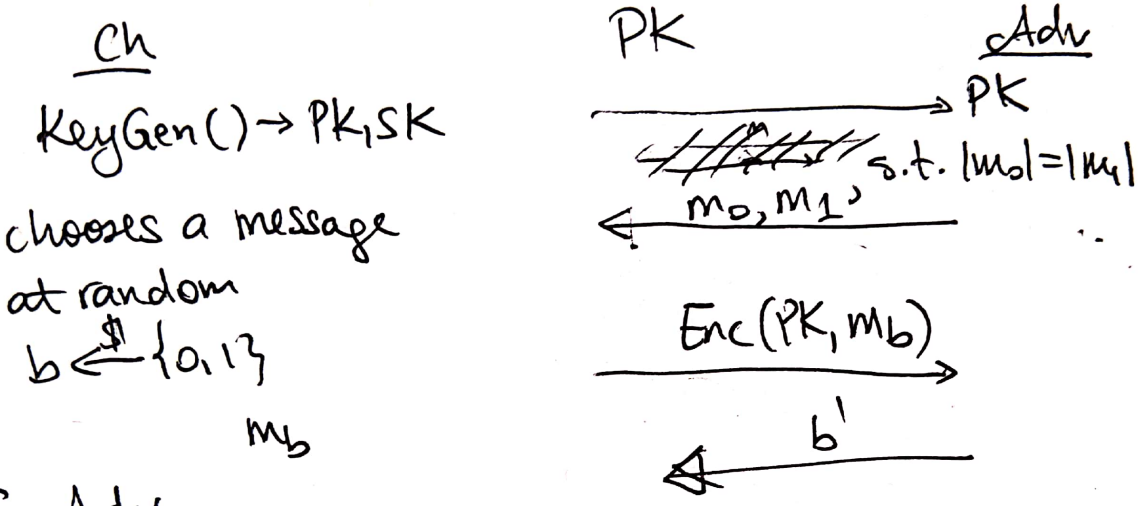
1. KeyGen() → (PK, SK)
2. Enc(PK, m) → C
3. Dec(SK, C) → m

Correctness: $\forall PK, SK \leftarrow KeyGen, \forall m,\ C = Enc(PK, m)$

$Dec(SK, C) = m$

Security: similar in spirit to IND-CPA

Semantic security



$\forall$ Adv,

$Pr[\text{Adv wins } (b' = b)] \leq 1/2 + negl$

ElGamal cryptosystem (1985)

Keygen()

- generate $ a large prime p (2048-bit) $\sim 2^{2048}$
- $g \in [2, p-1]$
- generate $ a secret key k $\in [2, p-2]$

k = SK

- $PK = g^k \bmod p$; $(g, p$ public$)$

Publish PK, keep SK secret

Due to the DLP assumption, cannot guess k

Enc(PK, m): $m \in [1, \ldots, p-1]$

- pick $ $r \in [1, \ldots, p-1]$

$$C = \left(\underbrace{g^r \bmod p}_{C_1} ; \underbrace{m \cdot PK^r \bmod p}_{C_2} \right)$$

Discrete Log Problem must hold (not sufficient)

$(g, p, g^k, C_1, C_2 \approx g, p, g^k, C_1, R)$

Dec(SK, (C_1, C_2)):

$$\frac{C_2}{C_1^k} \bmod p = m$$

k = SK

$$\frac{m \cdot (g^k \bmod p)^r \bmod p \bmod p}{(g^r \bmod p)^k} = m$$

✓ correctness