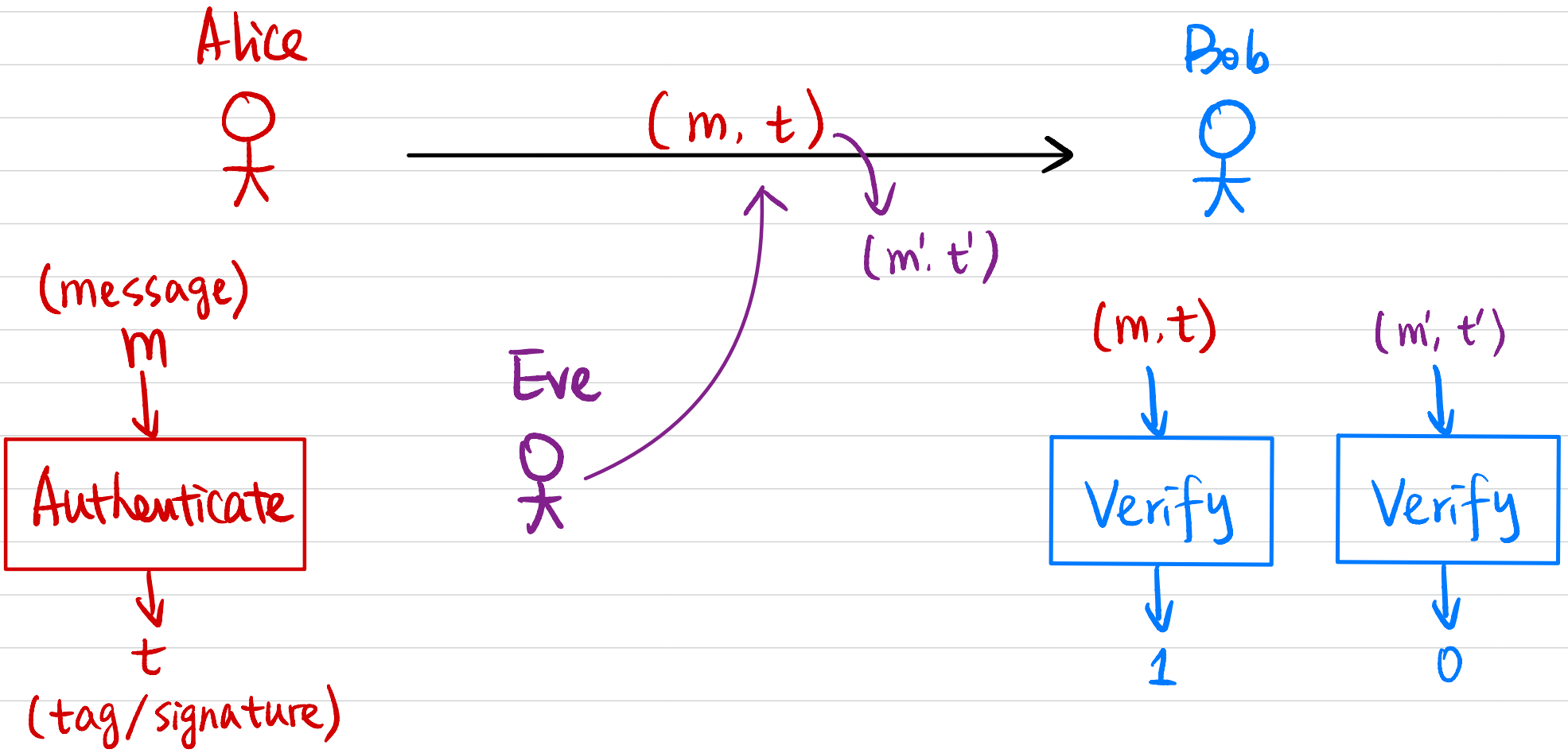
CSCI 1515 Applied Cryptography

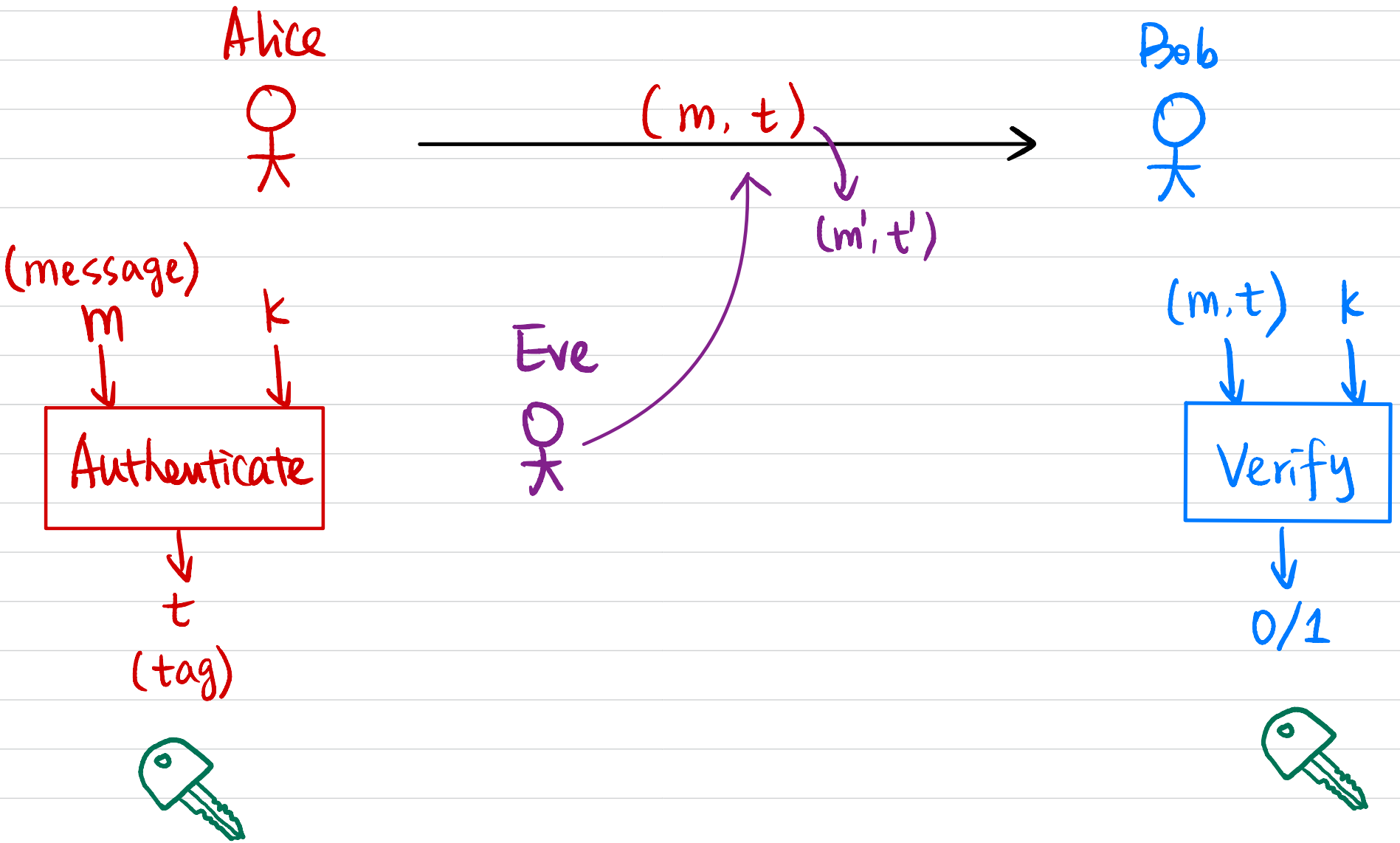
This Lecture:

- Message Integrity & CMA Security
- RSA Signature Scheme
- Authenticated Encryption
- Cryptographic Hash Function

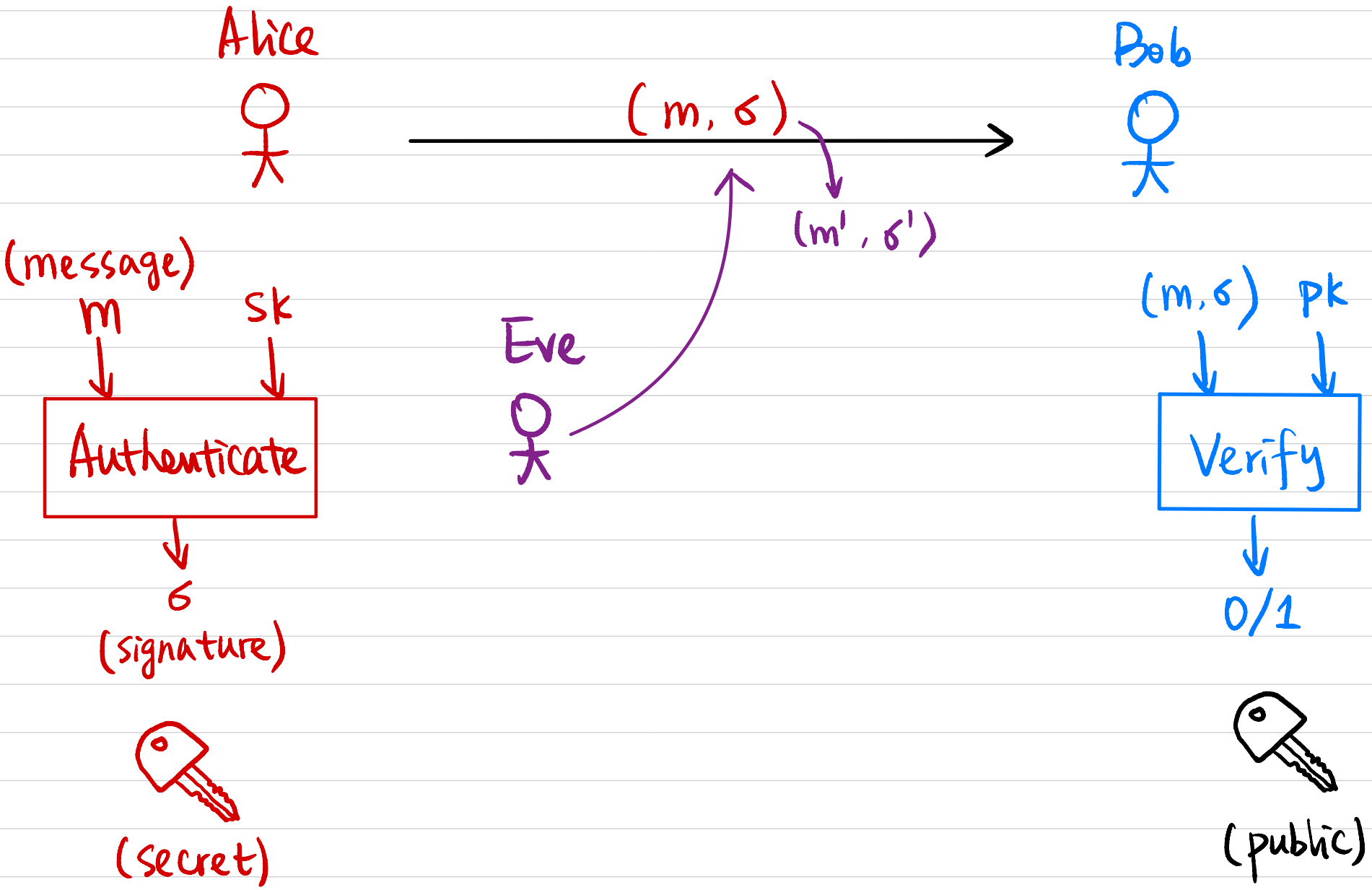
Message Integrity



Message Authentication Code (MAC)



Digital Signature



Syntax

Message Authentication Code (MAC) Scheme $\Pi = (Gen, Mac, Vrfy)$

$k \leftarrow Gen(1^\lambda)$

$t \leftarrow Mac_k(m)$

$0/1 := Vrfy_k(m, t)$

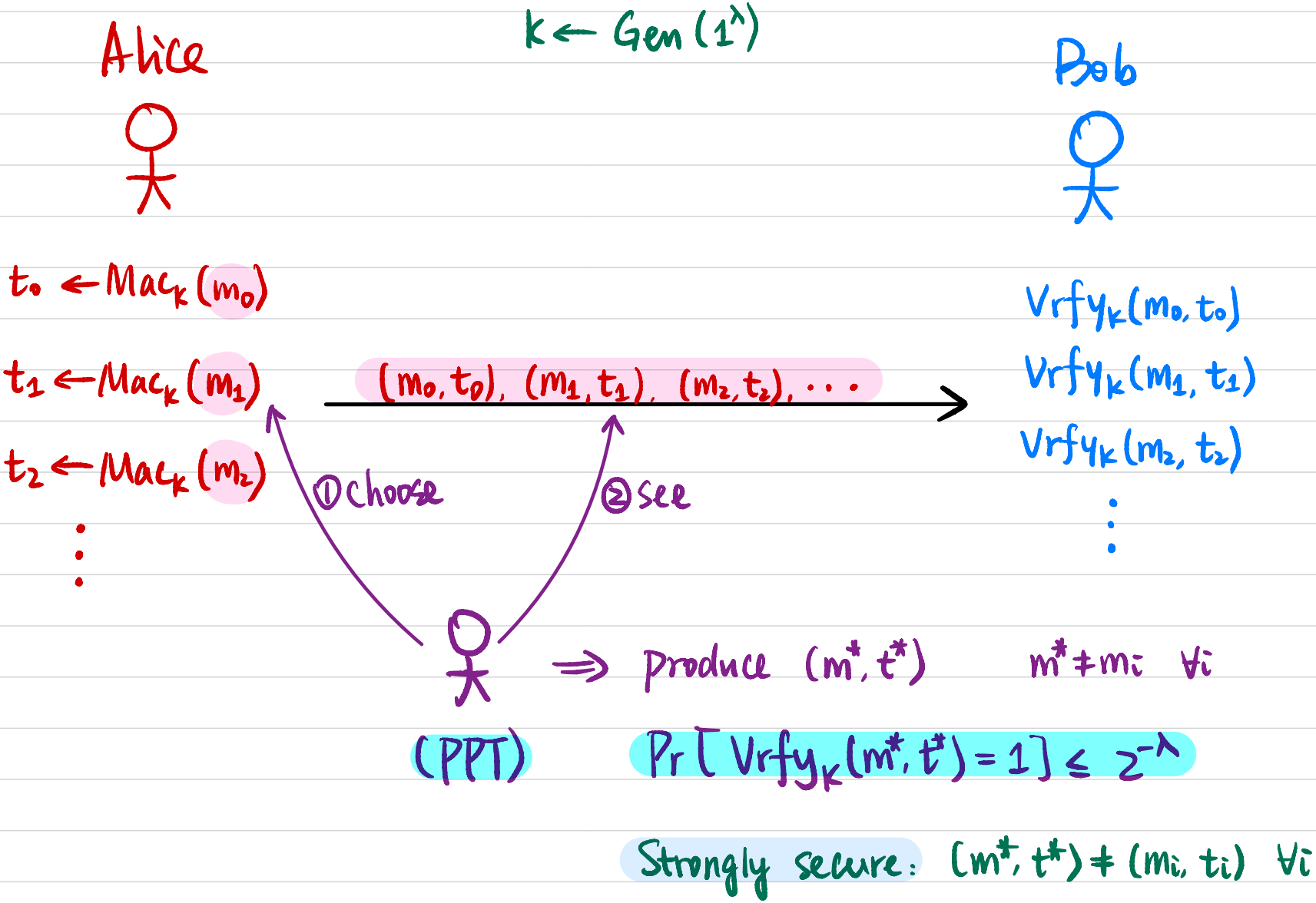
Digital Signature Scheme $\Pi = (Gen, Sign, Vrfy)$

$(pk, sk) \leftarrow Gen(1^\lambda)$

$\sigma \leftarrow Sign_{sk}(m)$

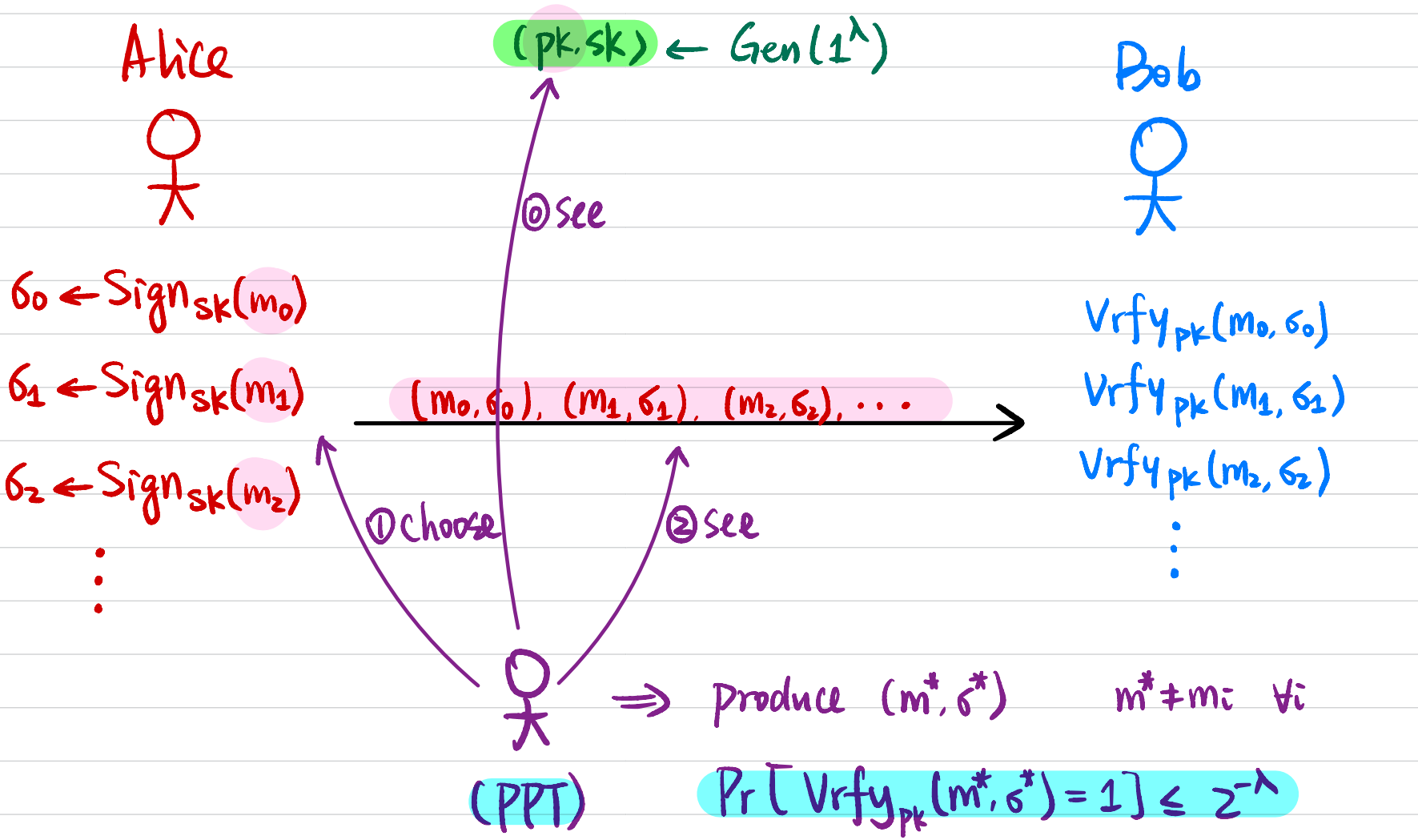
$0/1 := Vrfy_{pk}(m, \sigma)$

Chosen-Message Attack (CMA)



Does Mac have to be randomized?

Chosen-Message Attack (CMA)



Does Sign have to be randomized?

RSA Signature

Generate two n-bit primes p, q $(p \neq q)$

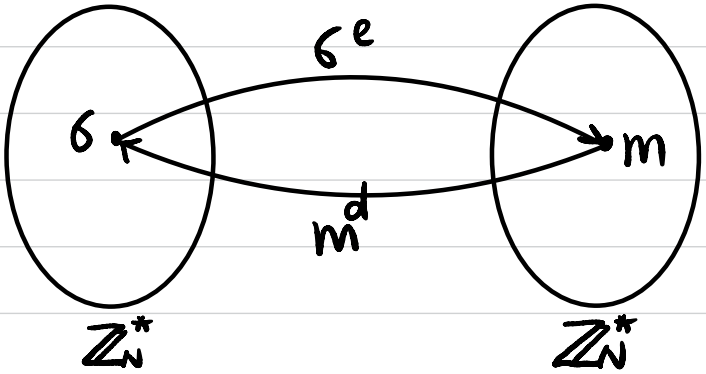
Compute $N = p \cdot q$, $\phi(N) = (p-1)(q-1)$

Choose e s.t. $\gcd(e, \phi(N)) = 1$

Compute $d = e^{-1} \bmod \phi(N)$

Given (N, e) & a random $y \xleftarrow{\$} \mathbb{Z}_N^*$, it's computationally hard to find x s.t.

$$x^e \equiv y \pmod N$$



$sk = d$ $\quad$ $pk = (N, e)$

$Sign_{sk}(m) = m^d \bmod N$

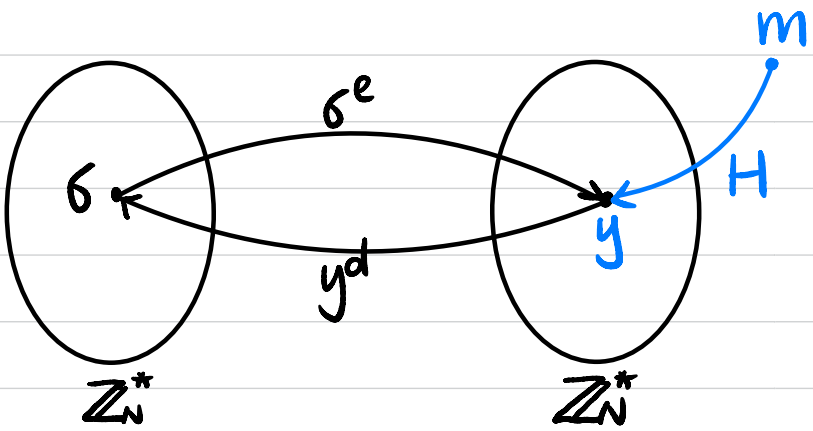
$Vrfy_{pk}(m, \sigma)$: $\sigma^e \stackrel{?}{\equiv} m \pmod N$

CMA Secure ?

RSA Signature

Given (N, e) & a random $y \xleftarrow{\$} \mathbb{Z}_N^*$, it's computationally hard to find x s.t.

$$x^e \equiv y \pmod N$$



$sk = d$ $\quad$ $pk = (N, e)$

$Sign_{sk}(m) = H(m)^d \bmod N$

$Vrfy_{pk}(m, \sigma)$: $\sigma^e \stackrel{?}{\equiv} H(m) \pmod N$

$H: \{0,1\}^* \to \mathbb{Z}_N^*$

↑

public, deterministic function that gives a (pseudo)random output

CMA Secure ?

Constructions for Digital Signature

RSA Signature : RSA Assumption

Schnorr / DSA Signature : Discrete Logarithm Assumption

Lattice-Based Signature Schemes (Post-Quantum Security)

Constructions for MAC

From block cipher: CBC-MAC

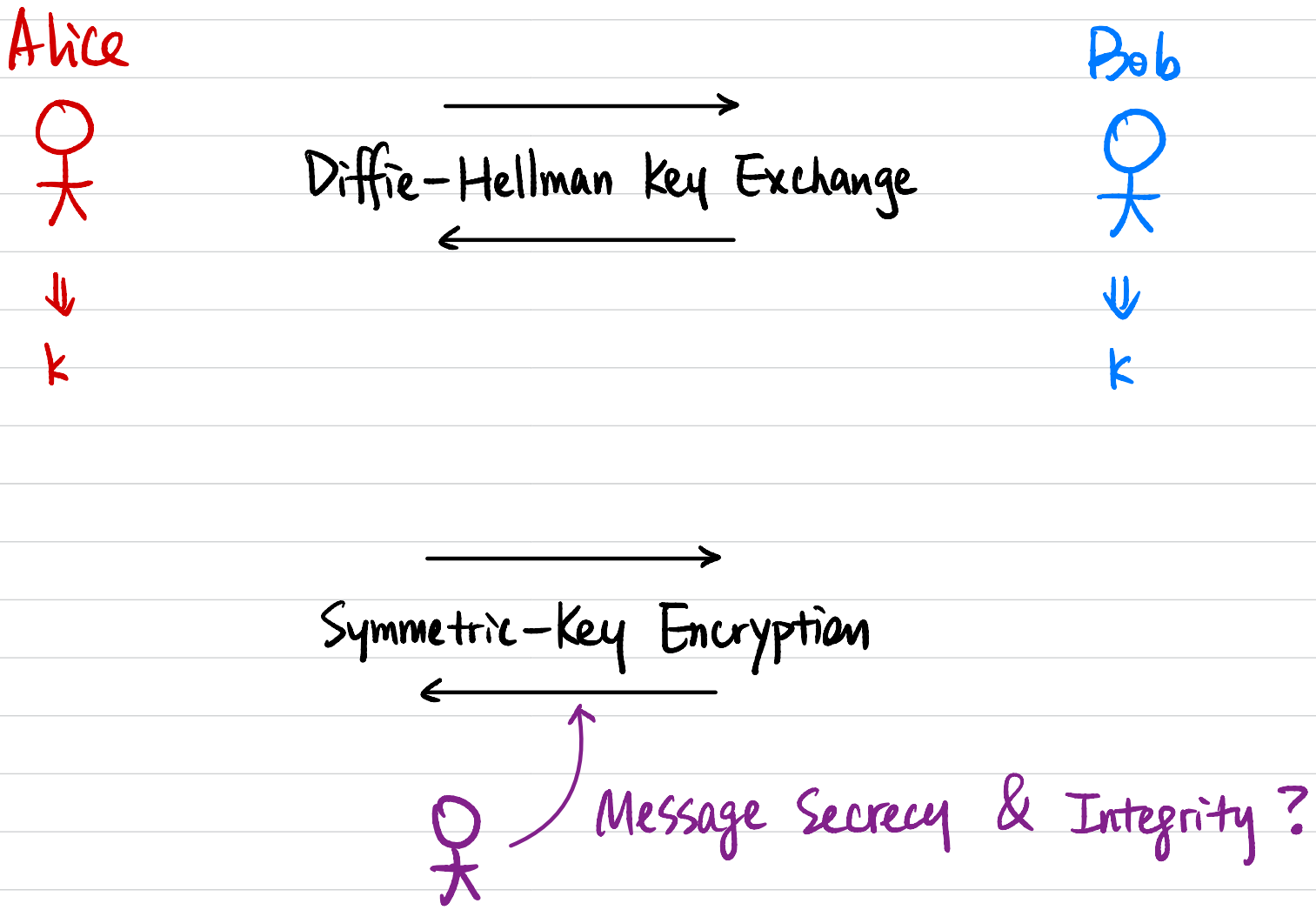
From hash function: HMAC

Computational Assumption: "The construction is secure"

Summary

	Symmetric-Key	Public-Key
Message Secrecy	Primitive: SKE Construction: block Cipher	Primitive: PKE Constructions: RSA / ElGamal
Message Integrity	Primitive: MAC Constructions: CBC-MAC / HMAC	Primitive: Signature Constructions: RSA / DSA
Key Exchange		Construction: Diffie-Hellman
Important Tool	Primitive: Hash function Construction: SHA	

What happens in practice



Authenticated Encryption (AE)

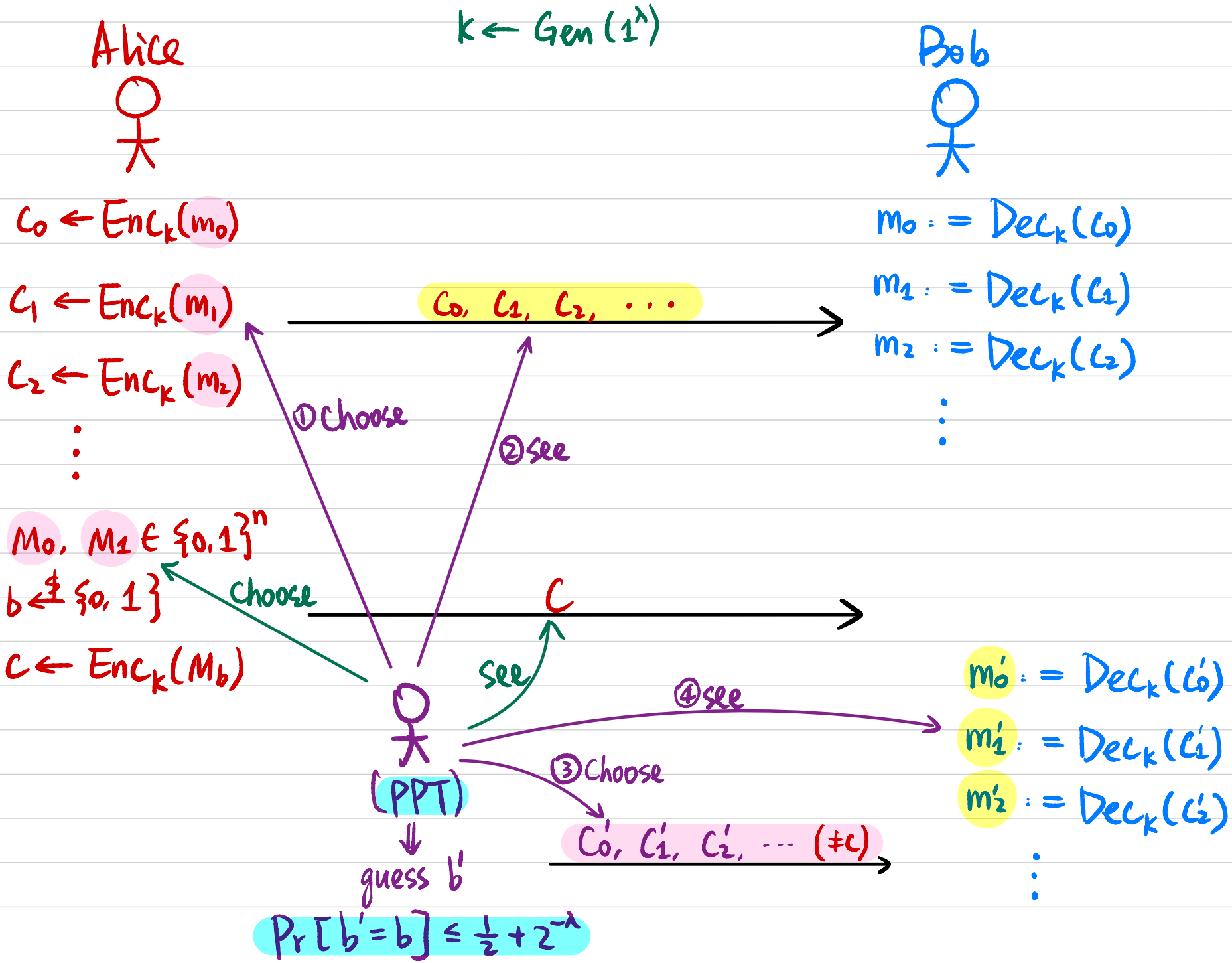
Def An authenticated encryption scheme is a symmetric-key encryption that is CCA-secure & unforgeable.

Construction

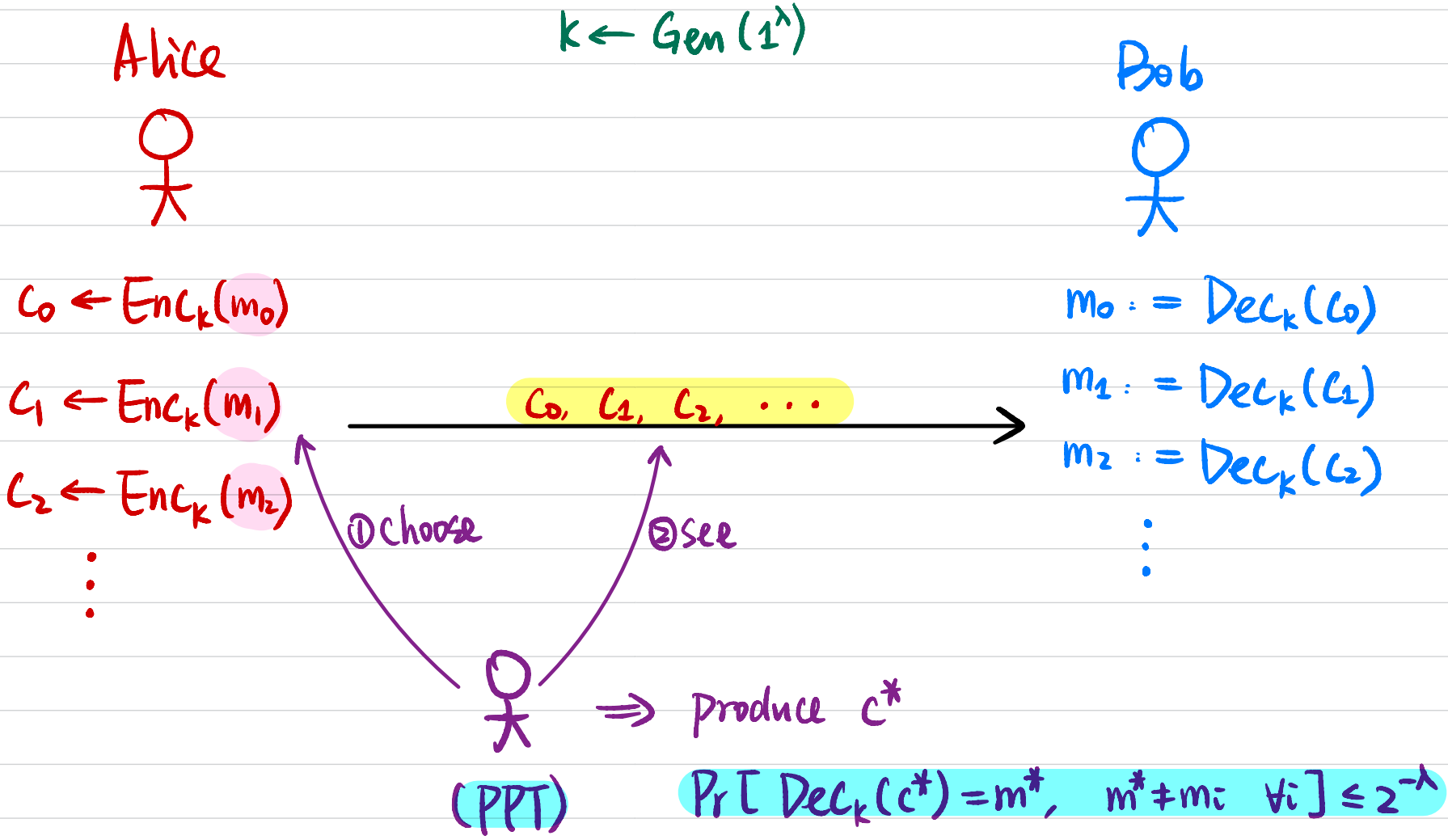
CPA-secure SkE
+
CMA-secure MAC
$\Rightarrow$ AE $\langle$ CCA-secure, Unforgeable

① Encryt-and-MAC
② MAC-then-Encryt
③ Encryt-then-MAC

Chosen-Ciphertext Attack (CCA) Security



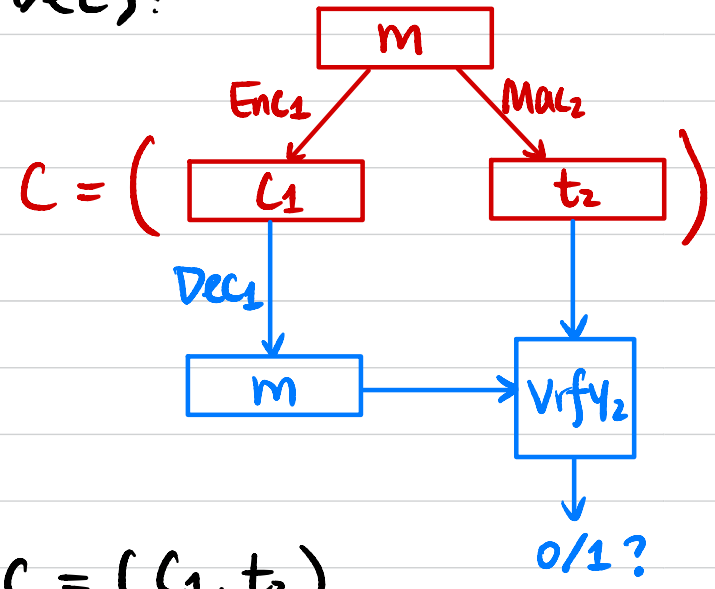
Unforgeablity



Encryt-and-MAC

Given a CPA-secure SKE scheme $\Pi_1 = (Gen_1, Enc_1, Dec_1)$
& a strongly CMA-secure MAC scheme $\Pi_2 = (Gen_2, Mac_2, Vrfy_2)$

Construct an AE scheme $\Pi = (Gen, Enc, Dec)$:



$Gen(1^\lambda)$:

$k_1 \leftarrow Gen_1(1^\lambda)$

$k_2 \leftarrow Gen_2(1^\lambda)$

Output $k = (k_1, k_2)$

$Enc_k(m)$:

$c_1 \leftarrow Enc_1(k_1, m)$

$t_2 \leftarrow Mac_2(k_2, m)$

output $c = (c_1, t_2)$

$Dec_k(c)$: $c = (c_1, t_2)$

$m := Dec_1(k_1, c_1)$

$b := Vrfy_2(k_2, (m, t_2))$

If $b=1$, output m

Otherwise output $\perp$

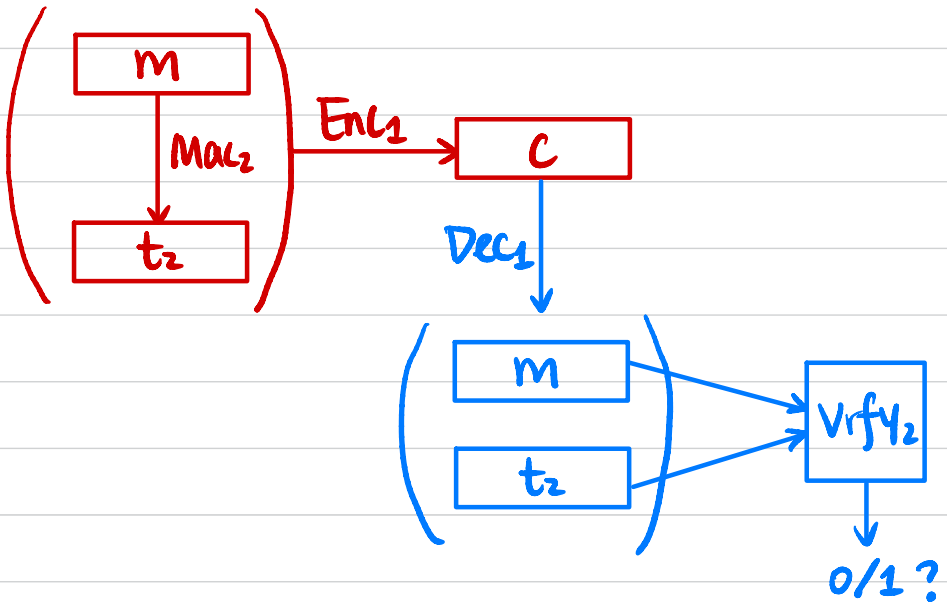
MAC-then-Encryt

$Gen(1^\lambda)$:

$k_1 \leftarrow Gen_1(1^\lambda)$

$k_2 \leftarrow Gen_2(1^\lambda)$

Output $k = (k_1, k_2)$



$Enc_k(m)$:

$t_2 \leftarrow Mac_2(k_2, m)$

$c_1 \leftarrow Enc_1(k_1, m \| t_2)$

output $c = c_1$

$Dec_k(c)$: $c = c_1$

$m \| t_2 := Dec_1(k_1, c_1)$

$b := Vrfy_2(k_2, (m, t_2))$

If $b=1$, output m

Otherwise output $\perp$

Encryt-then-MAC

$Gen(1^\lambda)$:

$k_1 \leftarrow Gen_1(1^\lambda)$

$k_2 \leftarrow Gen_2(1^\lambda)$

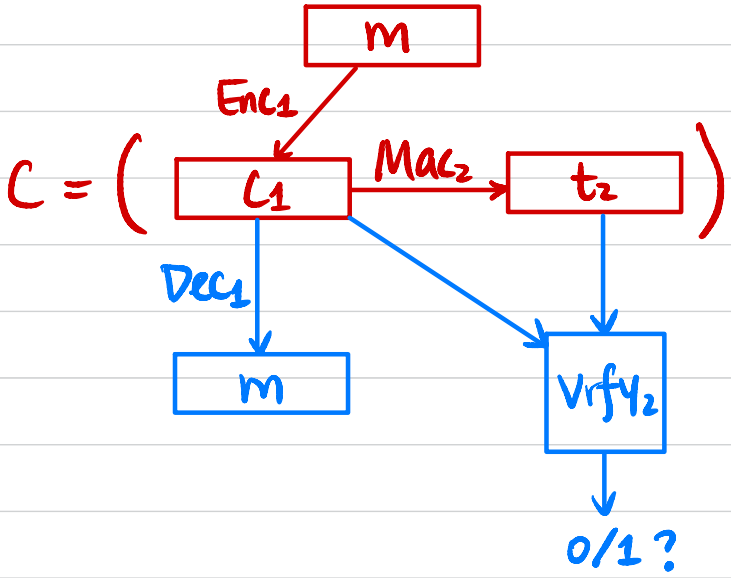
Output $k=(k_1, k_2)$

$Enc_k(m)$:

$C_1 \leftarrow Enc_1(k_1, m)$

$t_2 \leftarrow Mac_2(k_2, C_1)$

output $C=(C_1, t_2)$



$Dec_k(C)$: $C = (C_1, t_2)$

$m := Dec_1(k_1, C_1)$

$b := Vrfy_2(k_2, (C_1, t_2))$

If $b=1$, output m

Otherwise output $\perp$

Summary

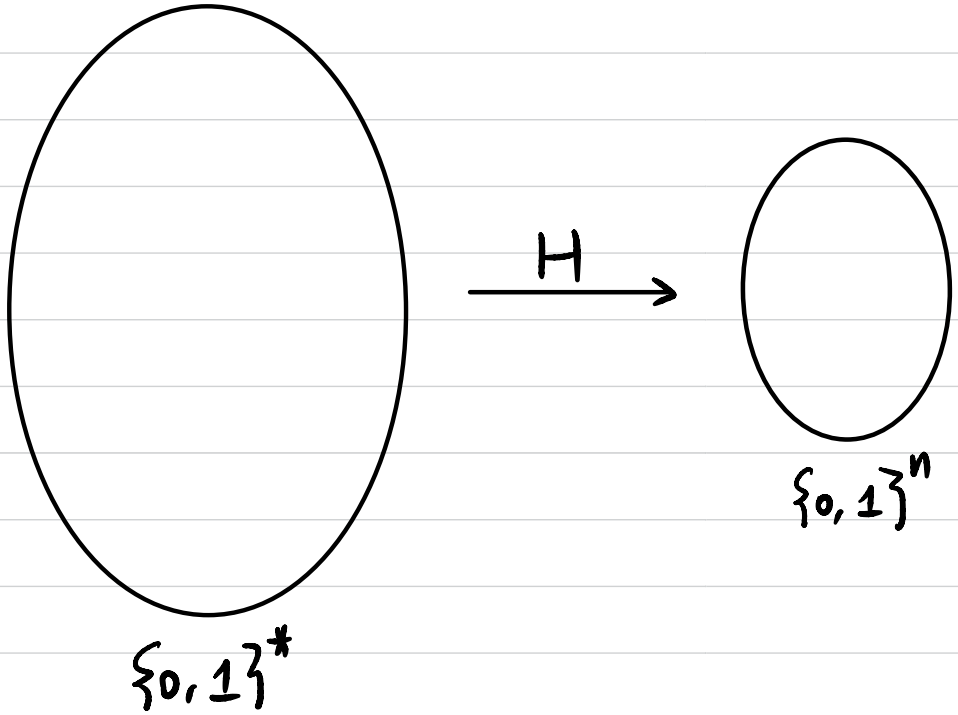
	Symmetric-Key	Public-Key
Message Secrecy	Primitive: SKE Construction: block Cipher	Primitive: PKE Constructions: RSA / ElGamal
Message Integrity	Primitive: MAC Constructions: CBC-MAC / HMAC	Primitive: Signature Constructions: RSA / DSA
Secrecy & Integrity	Primitive: AE Construction: Encrypt-then-MAC	
Key Exchange		Construction: Diffie-Hellman
Important Tool	Primitive: Hash function Construction: SHA	

Hash Function

$H: \{0,1\}^* \longrightarrow \{0,1\}^n$ (n: $\theta(\lambda)$)

$\lambda = 128$

$n = 256$



Collision-Resistant Hash Function (CRHF):

It's computationally hard to find $x, y \in \{0,1\}^*$ s.t.

$x \neq y$, $H(x) = H(y)$ (collision)

Birthday Problem / Paradox

There are q students in a class.

Assume each student's birthday is a random $y_i \xleftarrow{\$} [365]$

What's the probability of a collision?

$q = 366 \Rightarrow$ prob. $= 1$

$q = 23 \Rightarrow$ prob. $\simeq 50\%$

$q = 70 \Rightarrow$ prob. $\simeq 99.9\%$

$y_i \xleftarrow{\$} [N]$

$q = N+1 \Rightarrow$ prob. $= 1$

$q = \sqrt{N} \Rightarrow$ prob. $\simeq 50\%$

Random Oracle Model

$H: \{0,1\}^* \longrightarrow \{0,1\}^n$

