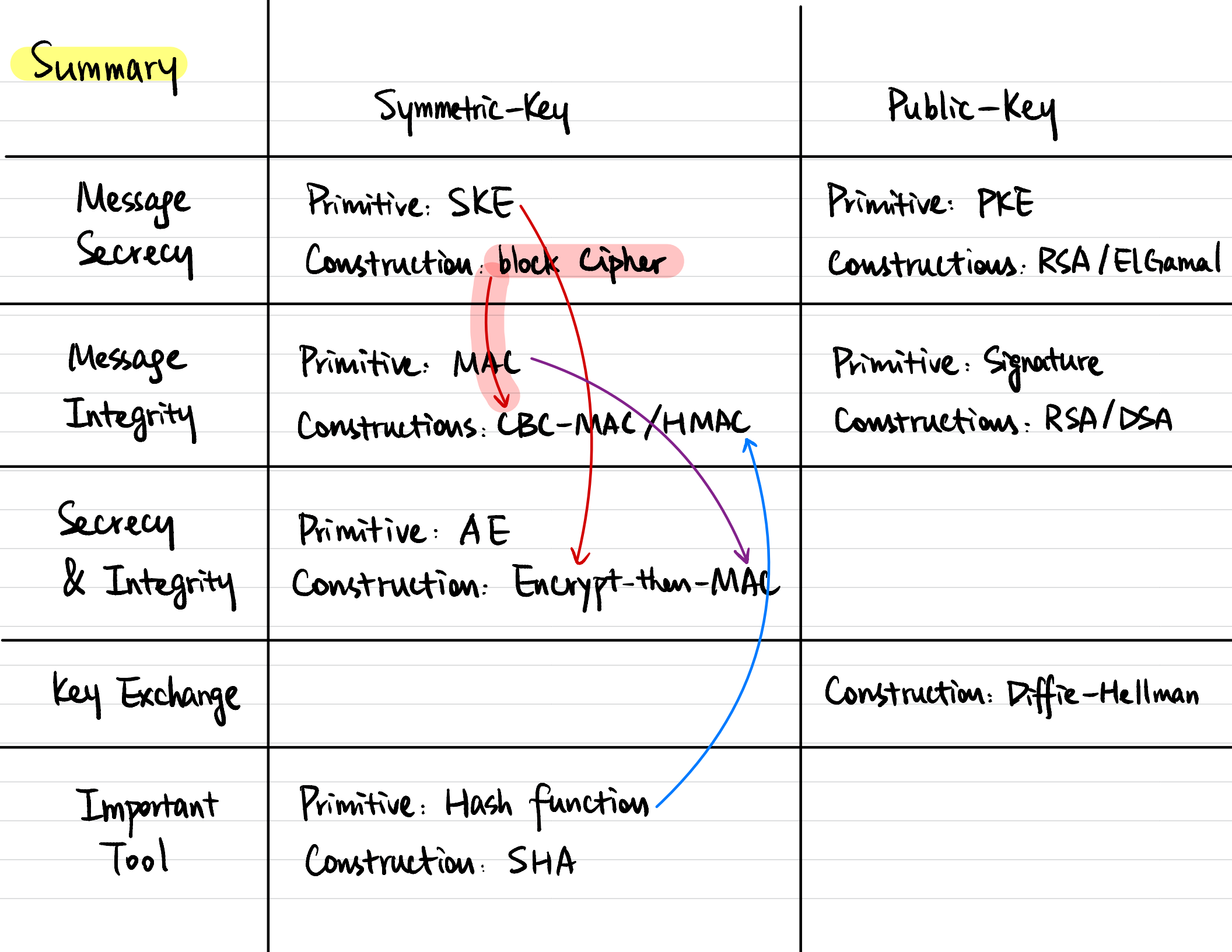
CSCI 1515 Applied Cryptography

This Lecture:

- Block Cipher and Modes of Operation (Continued)
- CBC-MAC
- Case Study: Secure Shell (SSH)

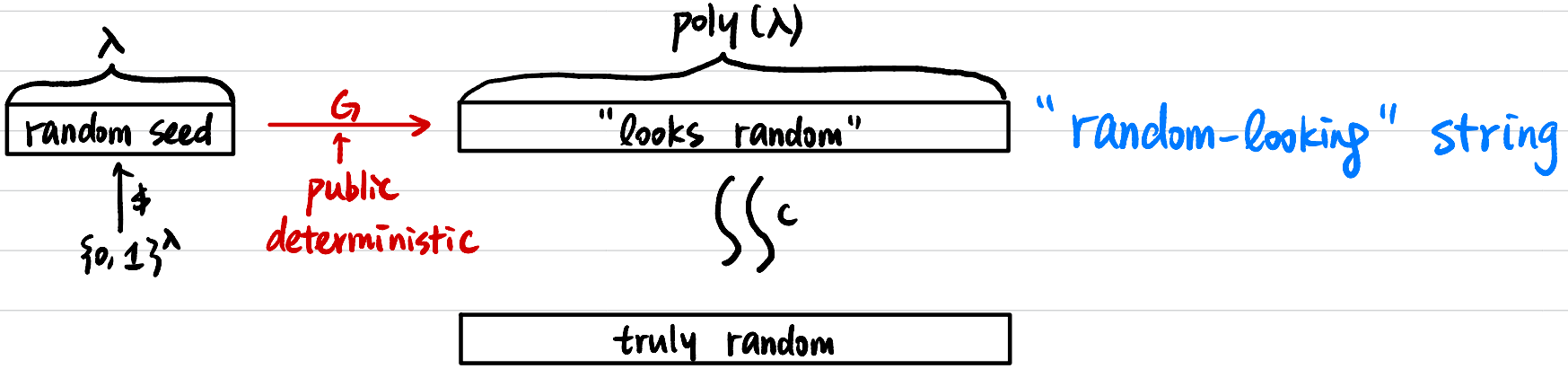
Summary

	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	



Pseudorandom Function (PRF)

Pseudorandom Generator (PRG)



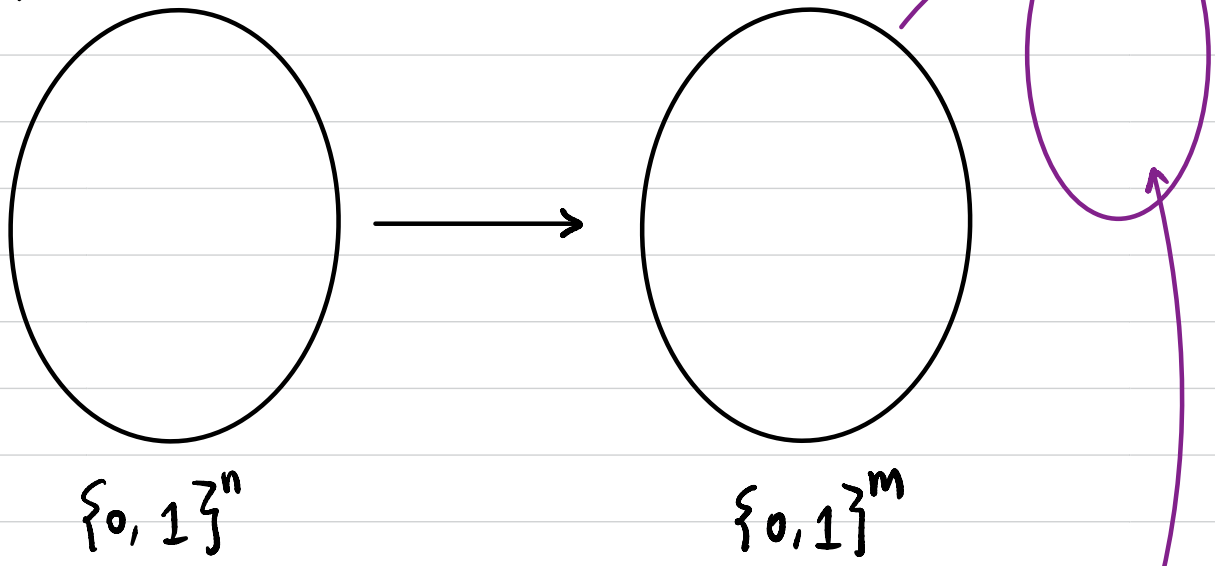
"random-looking" string

Pseudorandom Function (PRF):

"random-looking" function

Pseudorandom Function (PRF)

$k \xleftarrow{\$} \{0,1\}^\lambda$ $\quad F_k :$



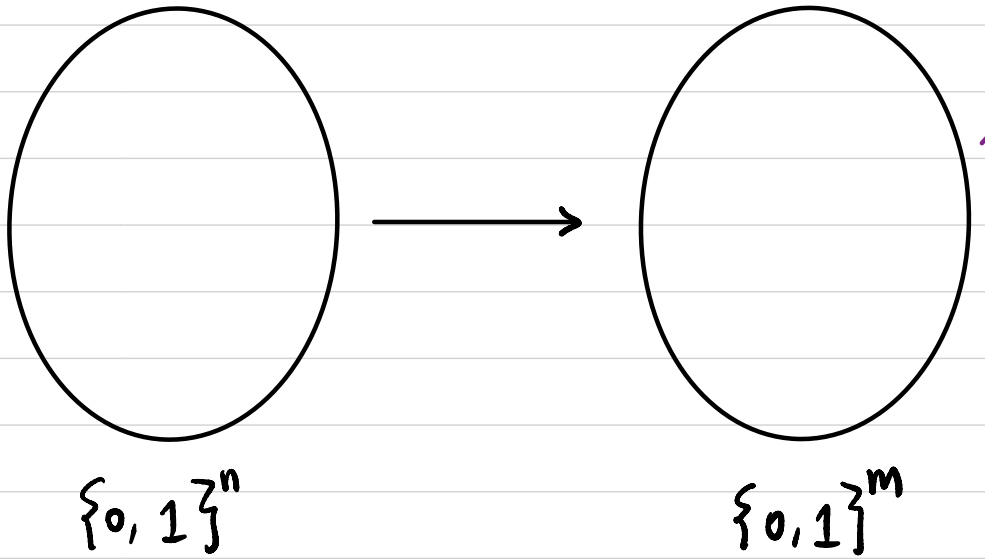
How many possible F_k's ?

2^λ

$\S\S$ C (not knowing k)

$f \xleftarrow{\$} \{ F \mid F: \{0,1\}^n \to \{0,1\}^m \}$

$f :$

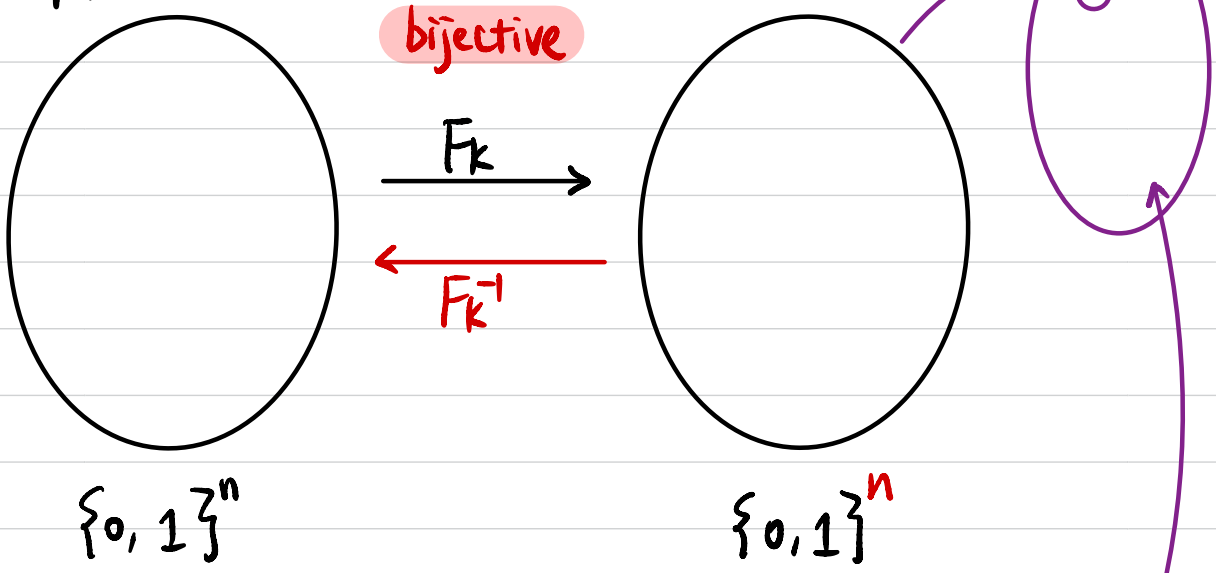


How many possible f's ?

$(2^m)^{2^n}$

Pseudorandom Permutation (PRP)

$k \xleftarrow{\$} \{0,1\}^\lambda$ $\quad F_k:$



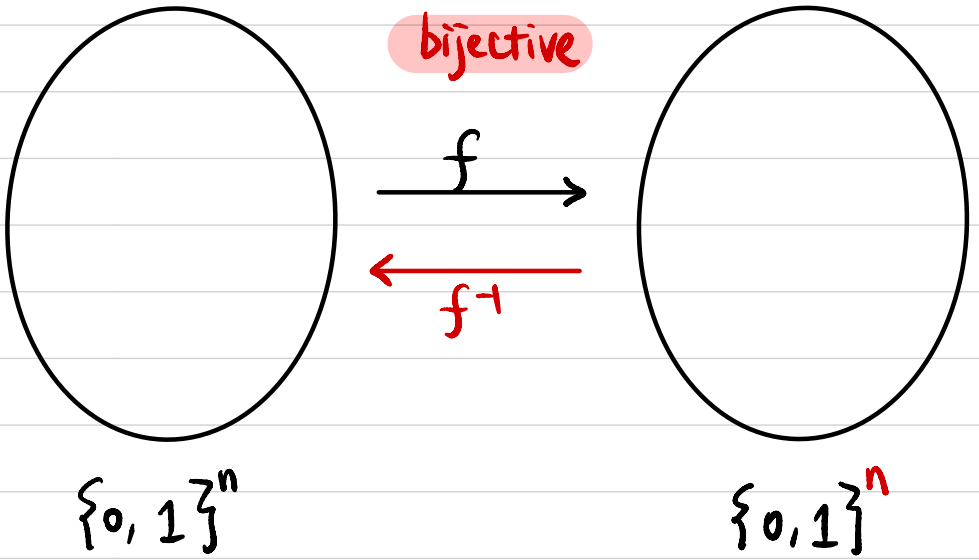
How many possible F_k's ?

2^λ

$\mathcal{C}$ (not knowing k)

$f \xleftarrow{\$} \{ F \mid F: \{0,1\}^n \to \{0,1\}^n,$ F is bijective $\}$

$f:$



How many possible f's ?

$2^n!$

Block Cipher

$$F: \{0,1\}^\lambda \times \{0,1\}^n \to \{0,1\}^n$$

λ: key length — AES: $n = 128$

n: block length — $\lambda = 128/192/256$

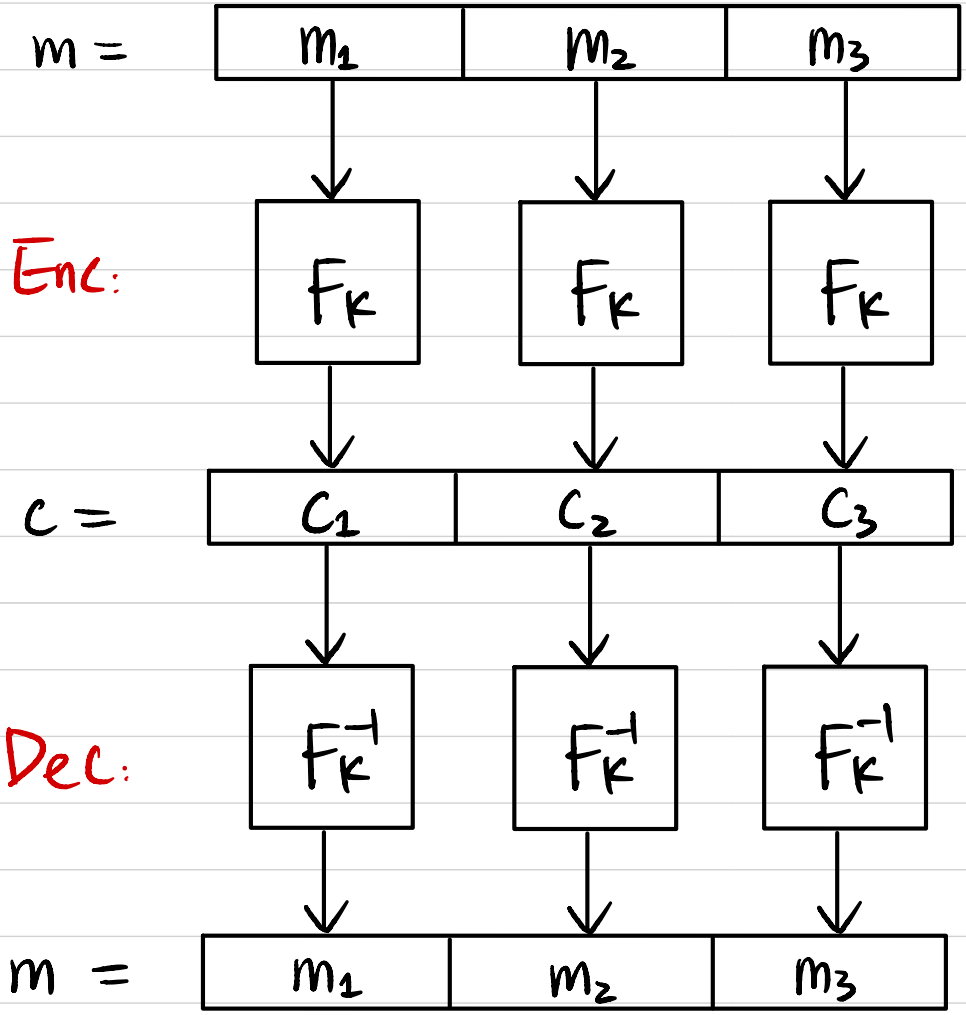
It is assumed to be a pseudorandom permutation (PRP).

Construct an SKE scheme from F for arbitrary-length messages.

- $k \xleftarrow{\$} \{0,1\}^\lambda$
- $Enc_k(m)$ — $|m| = \alpha \cdot \lambda$ (If not, pad it to a multiple of λ)
- $Dec_k(c)$

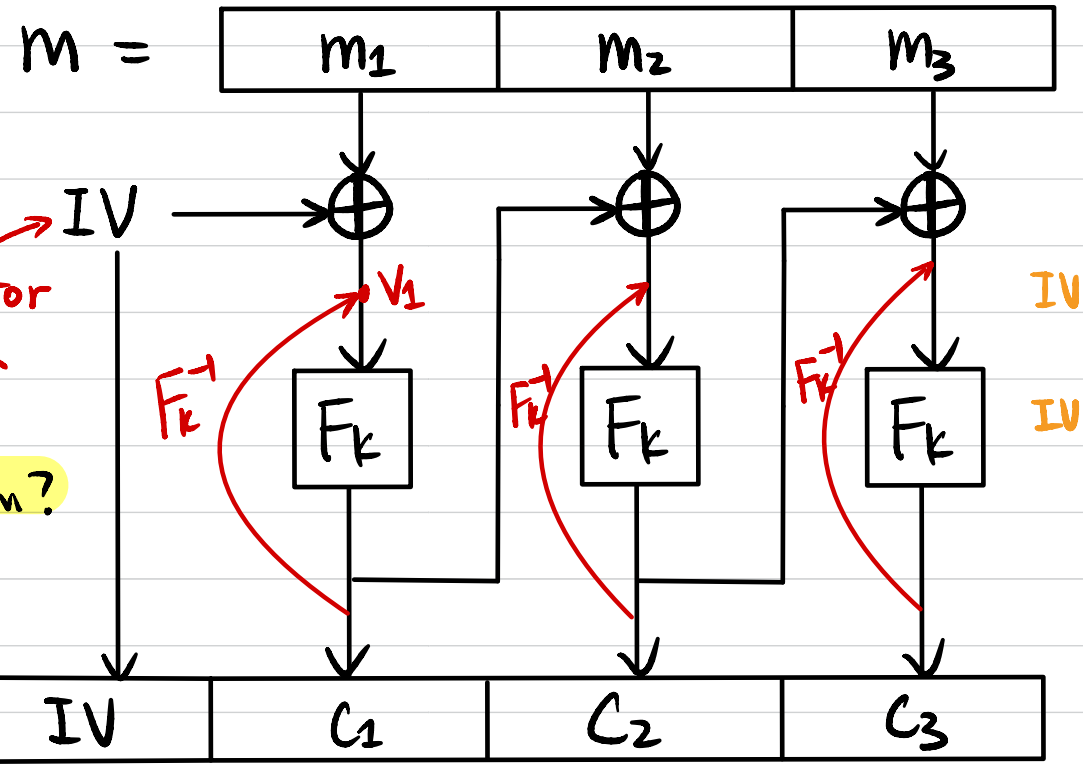
Goal: CPA (Chosen Plaintext Attack) Security

Electronic Code Book (ECB) Mode



CPA Secure? No! Deterministic!

Cipher Block Chaining (CBC) Mode



$m_1 \oplus IV \to V_1$

$IV \oplus V_1 \to m_1$

Initialization Vector

$IV \xleftarrow{\$} \{0,1\}^\lambda$

What if not random?

$IV = 0 \cdots 0 \quad V_1 = m_1$

flip ↓ last bit

$IV' = 0 \cdots 1 \quad V_1' = m_1' \oplus 0 \cdots 1$

How to decrypt?

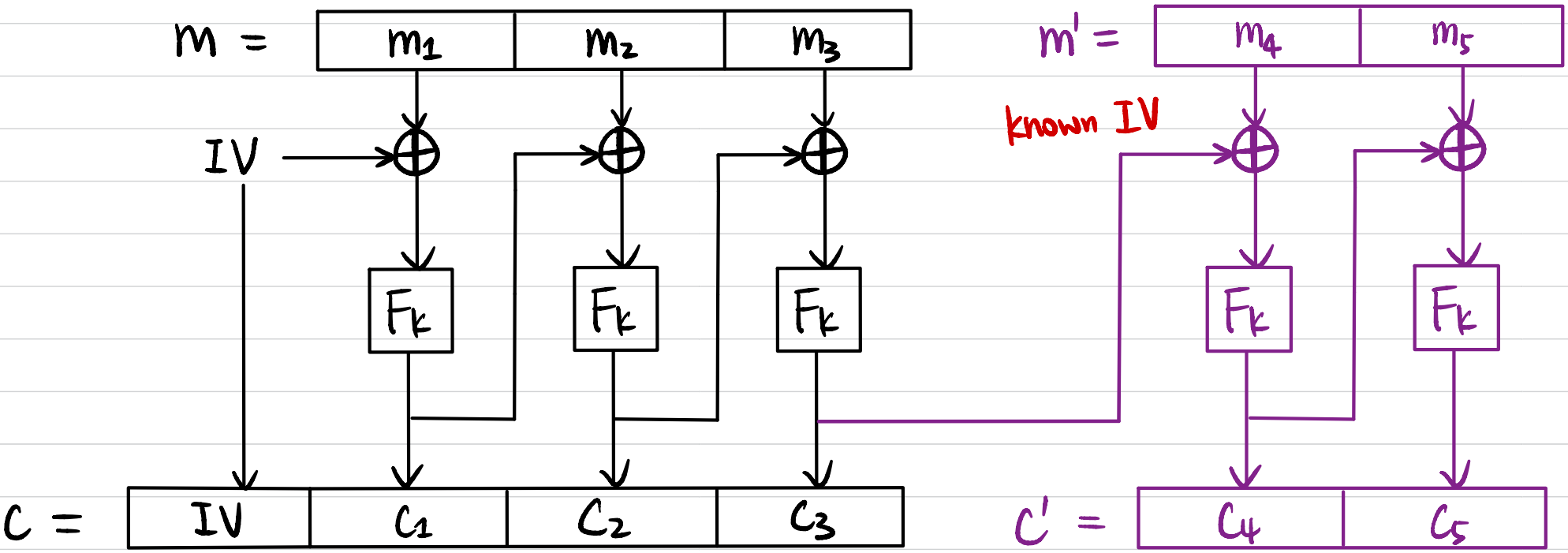
CPA Secure? YES!

Can we parallelize the computation?

NO for Enc

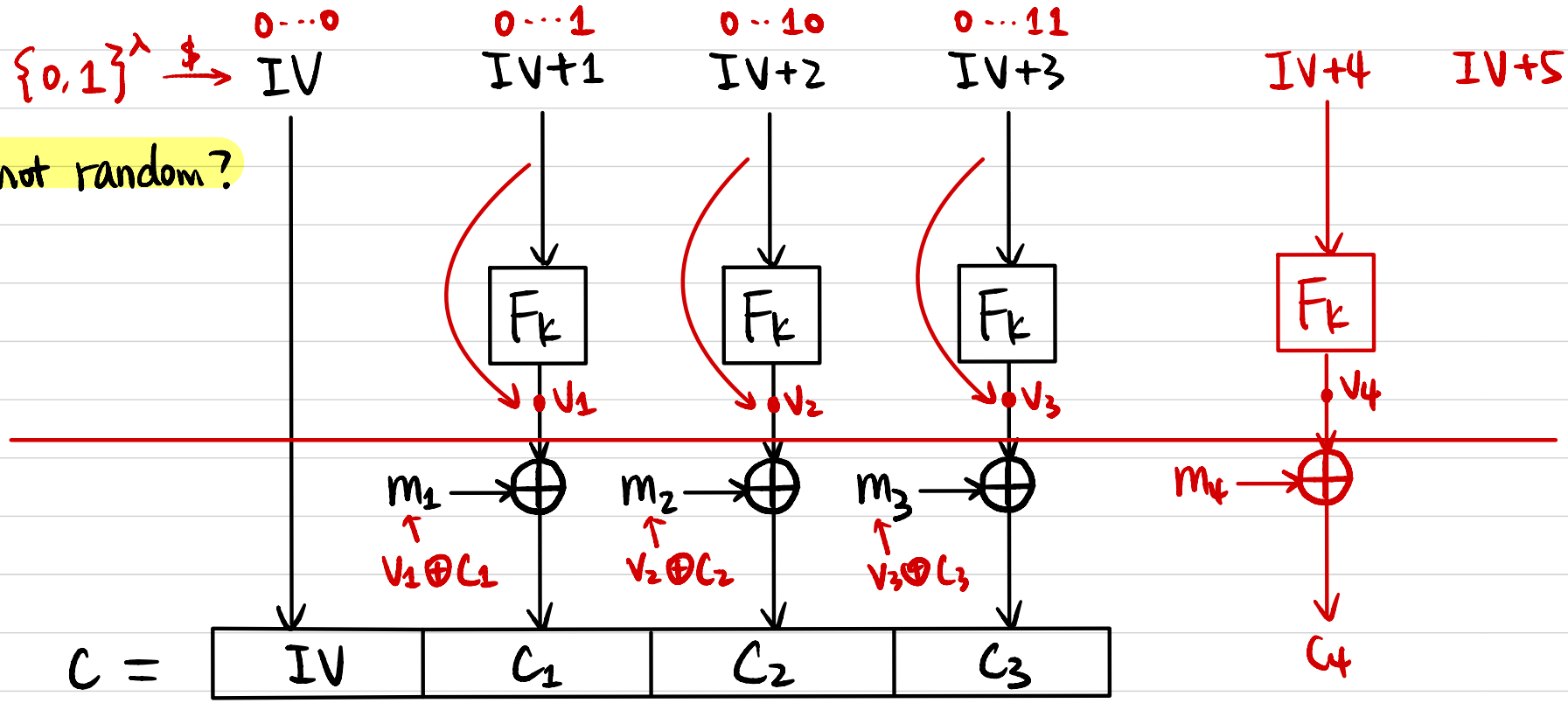
YES for Dec

Chained Cipher Block Chaining (CBC) Mode



CPA secure ?

Counter (CTR) Mode



What if not random?

How to decrypt?

CPA secure?

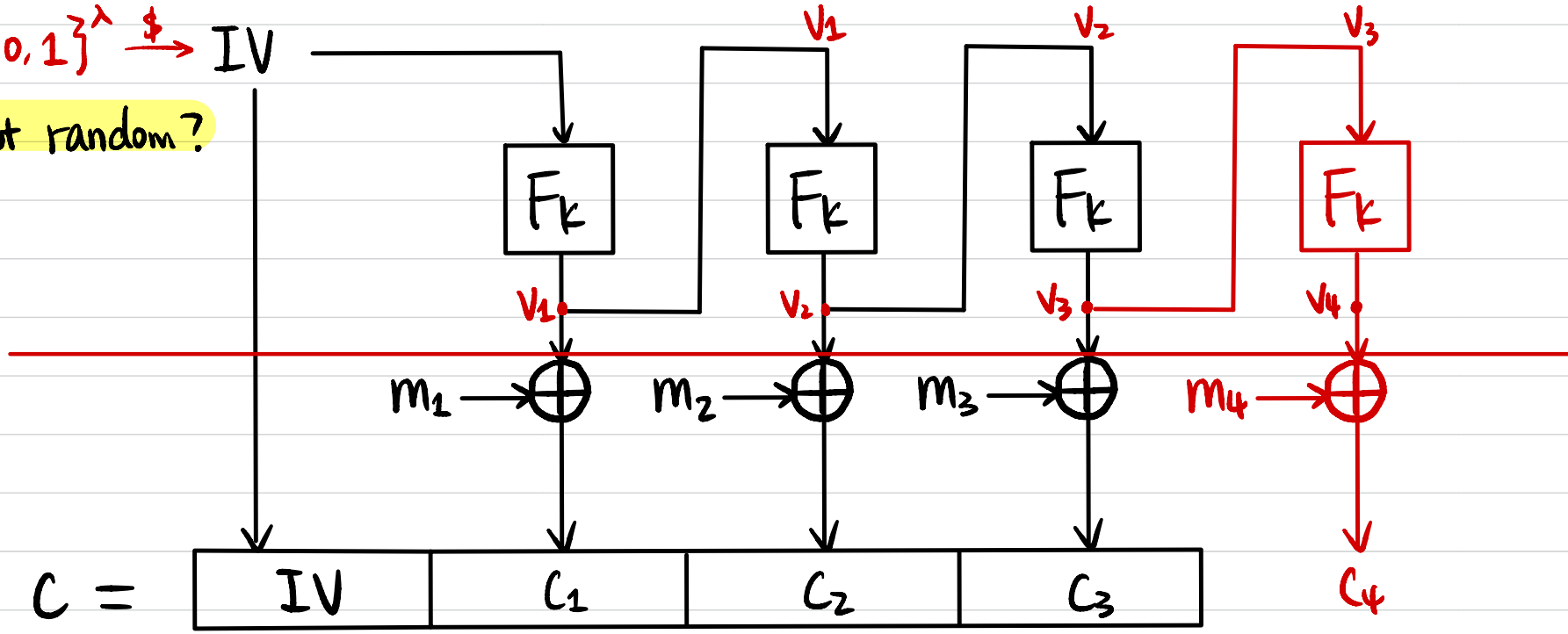
"Stateful" CTR Mode?

Can we parallelize the computation?

PRG from PRF

$G(k) = F_k(0) \| F_k(1) \| F_k(2) \| \cdots$

Output Feedback (OFB) Mode



How to decrypt?

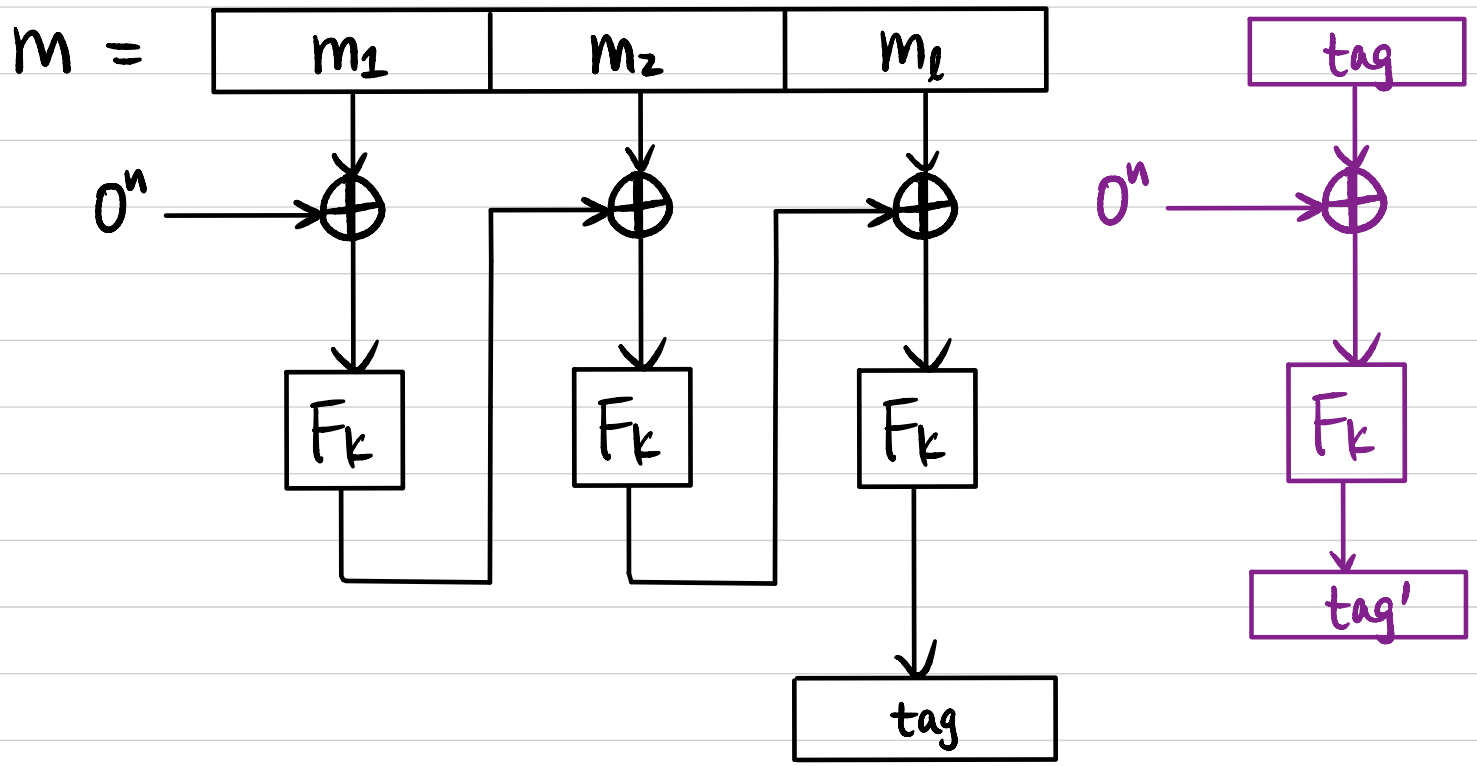
CPA secure?

"Stateful" OFB Mode?

Can we parallelize the computation?

PRG from PRF

CBC-MAC



How to verify ?

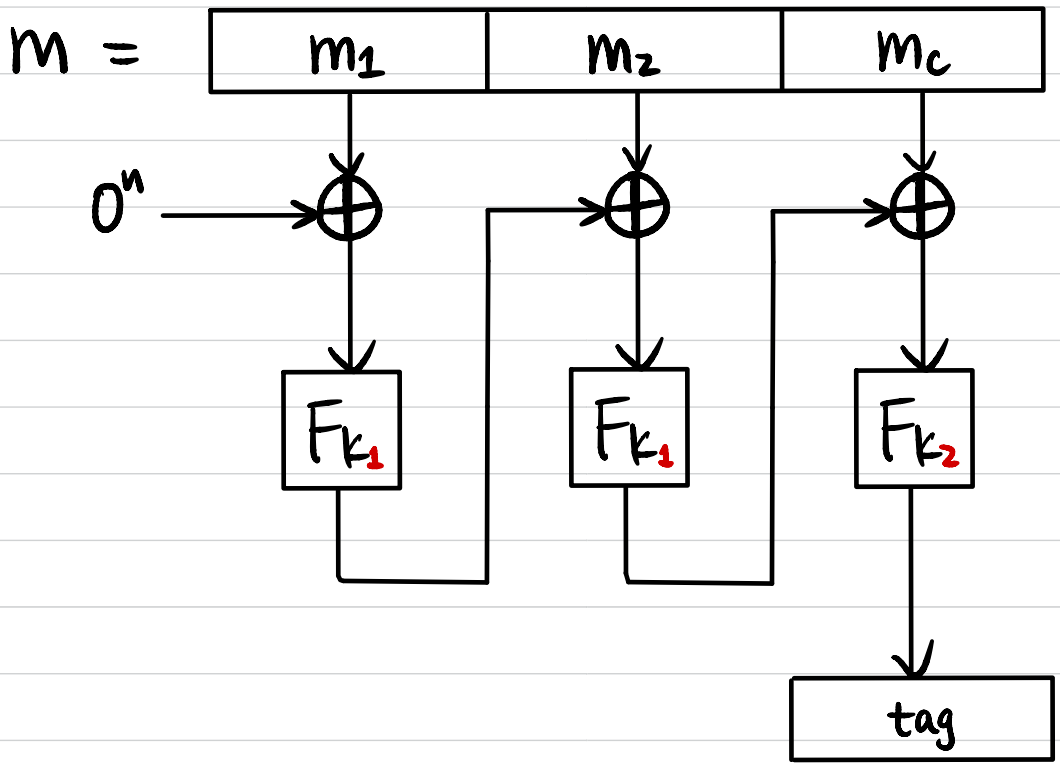
$m^* =$ | m_1 | m_2 | m_e | 0 |

CMA (Chosen Message Attack) Secure ?

$t^* =$ | tag' |

- Fixed-length messages of length $l \cdot n$
- Arbitrary-length messages

Encrypt-last-block CBC-MAC (ECBC-MAC)



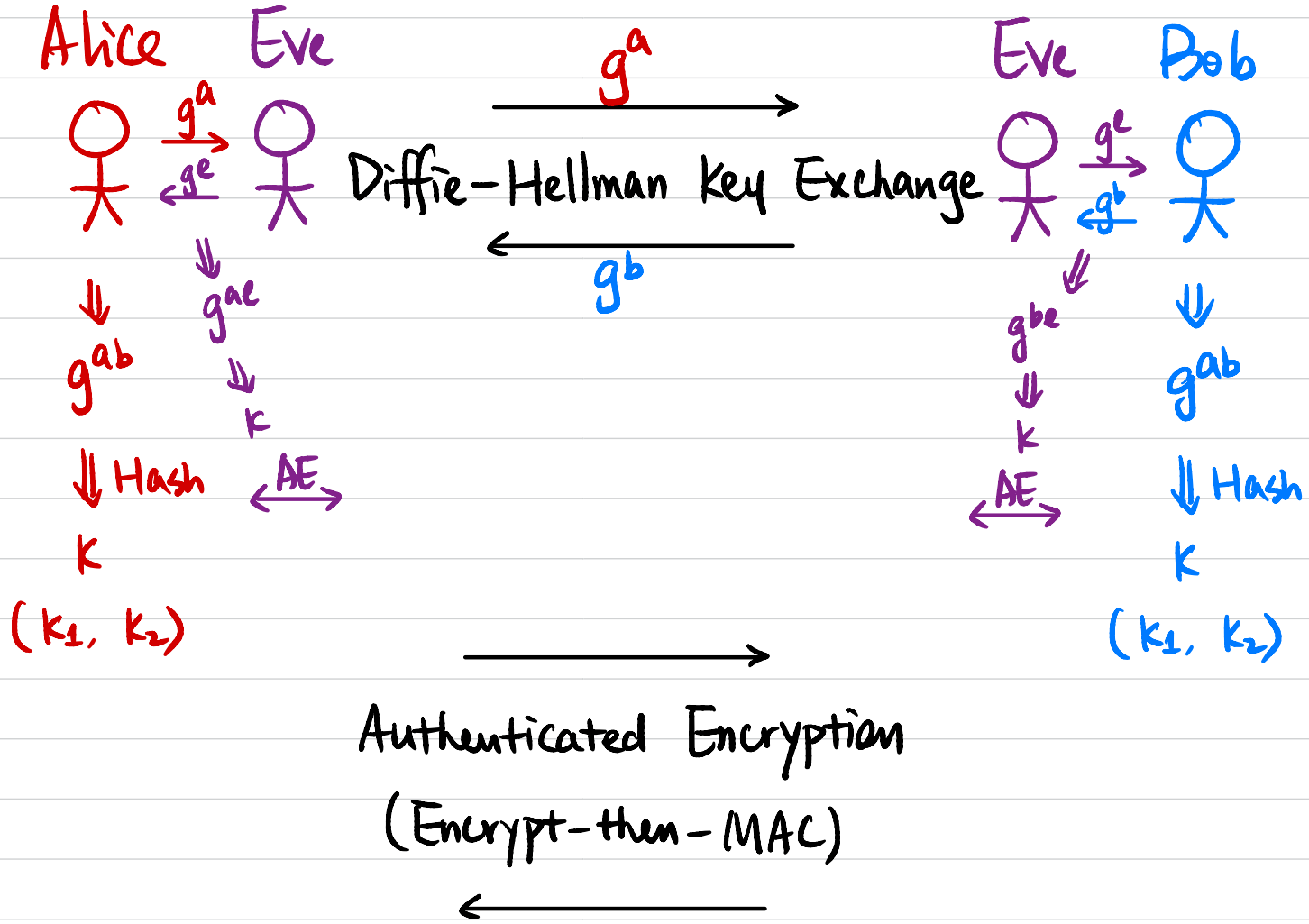
CMA (Chosen Message Attack) Secure?

- Arbitrary-length messages of length c·n
- Arbitrary-length messages

Summary

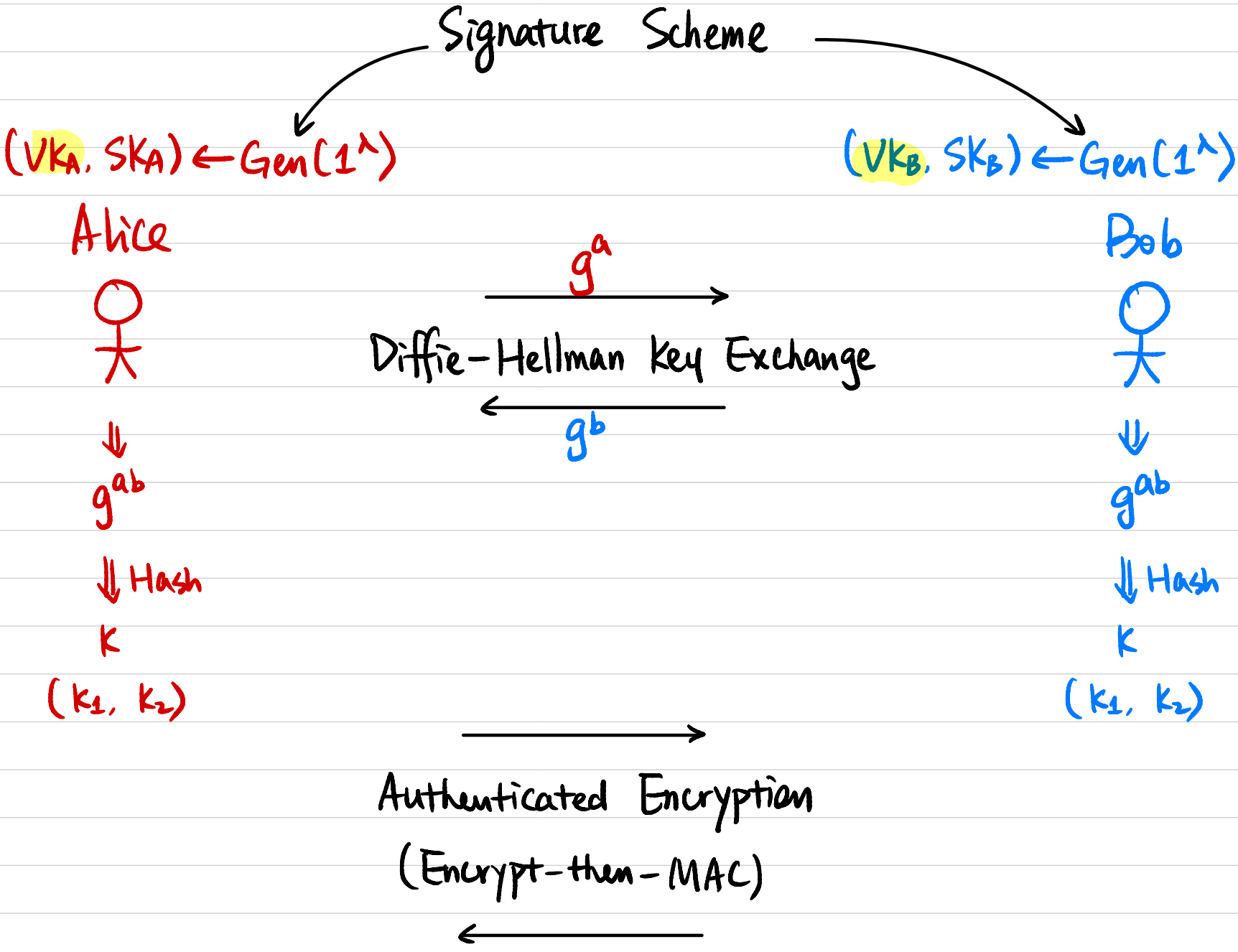
	Symmetric-Key	Public-Key
Message Secrecy	Primitive: SKE Construction: block Cipher	Primitive: PKE Constructions: RSA / ElGamal
Message Integrity	Primitive: MAC Constructions: CBC-MAC / HMAC	Primitive: Signature Constructions: RSA / DSA
Secrecy & Integrity	Primitive: AE Construction: Encrypt-then-MAC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

Putting it Together: Secure Communication



Any security issue?

"Man-in-the-Middle" Attack


Signature Scheme
$(VK_A, SK_A) \leftarrow Gen(1^\lambda)$
$(VK_B, SK_B) \leftarrow Gen(1^\lambda)$
Alice
Bob
g^a
Diffie-Hellman Key Exchange
g^b
g^{ab}
Hash
K
(k_1, k_2)
g^{ab}
Hash
K
(k_1, k_2)
Authenticated Encryption
(Encrypt-then-MAC)

Secure Shell Protocol (SSH)