CSCI 1515 Applied Cryptography

This Lecture:

- CBC-MAC
- Password-Based Authentication
- Two-Factor Authentication (2FA)

Summary

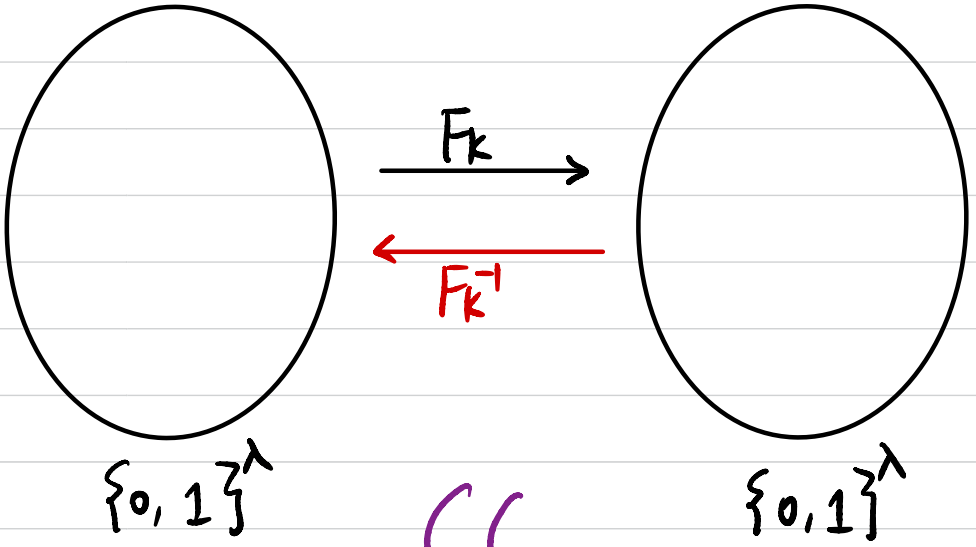
	Symmetric-Key	Public-Key
Message Secrecy	Primitive: SKE Construction: block cipher	Primitive: PKE Constructions: RSA / ElGamal
Message Integrity	Primitive: MAC Constructions: CBC-MAC / HMAC	Primitive: Signature Constructions: RSA / DSA
Secrecy & Integrity	Primitive: AE Construction: Encrypt-then-MAC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

Block Cipher

$F: \{0,1\}^\lambda \times \{0,1\}^\lambda \to \{0,1\}^\lambda$ ← 128

pseudorandom permutation (PRP).

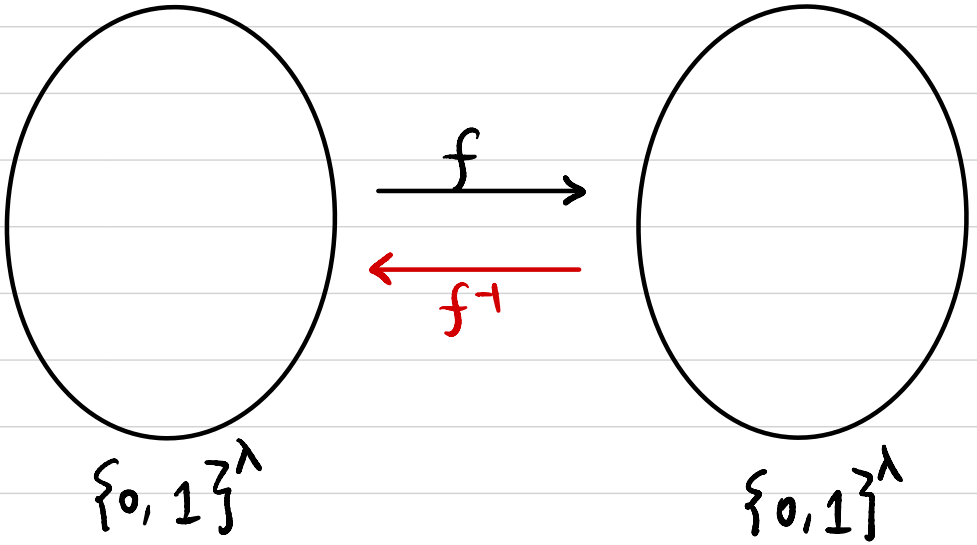
$k \xleftarrow{\$} \{0,1\}^\lambda$ F_k :



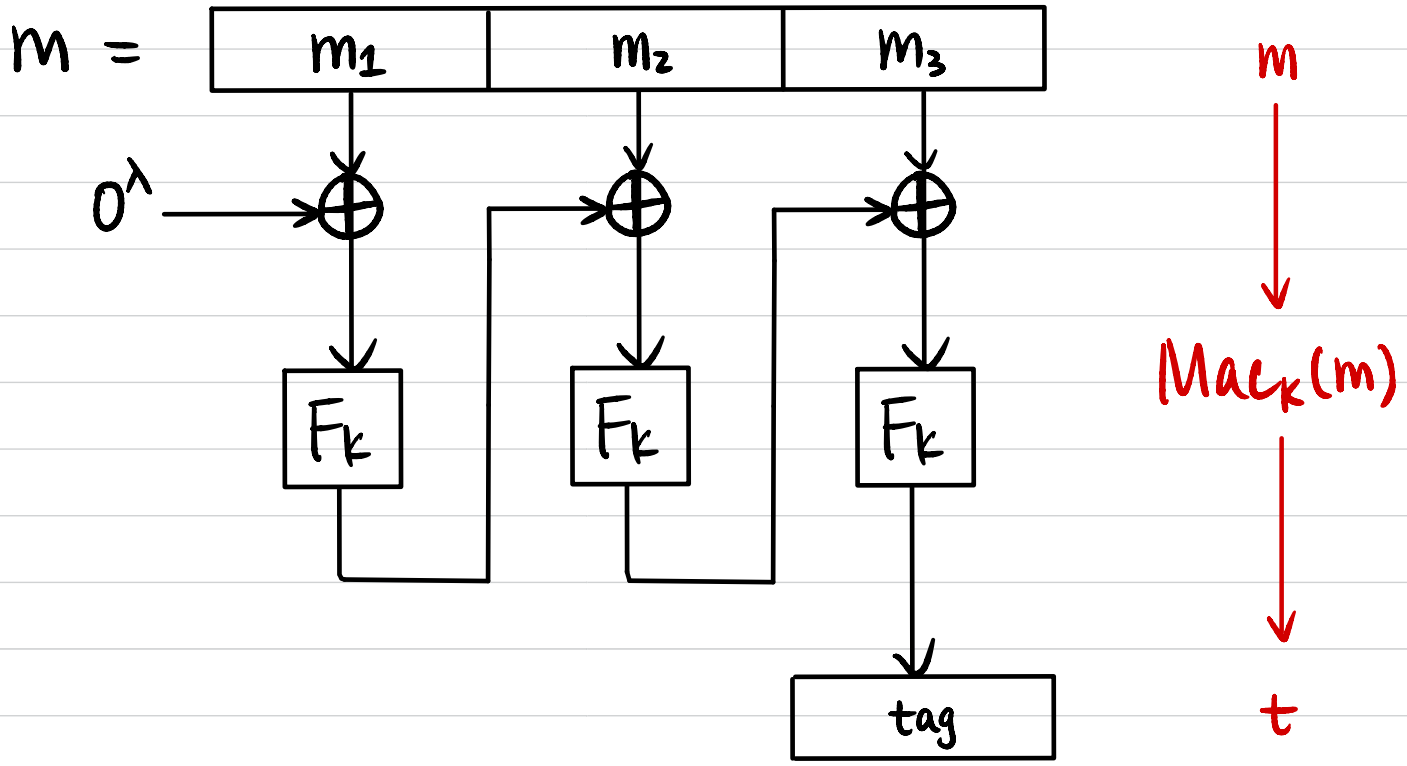
$\approx_c$

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

f :



CBC-MAC



m

$\downarrow$

$Mac_k(m)$

$\downarrow$

t

How to verify? $Vrfy_k(m,t)$: $Mac_k(m) \stackrel{?}{=} t$

CMA (Chosen Message Attack) Secure?

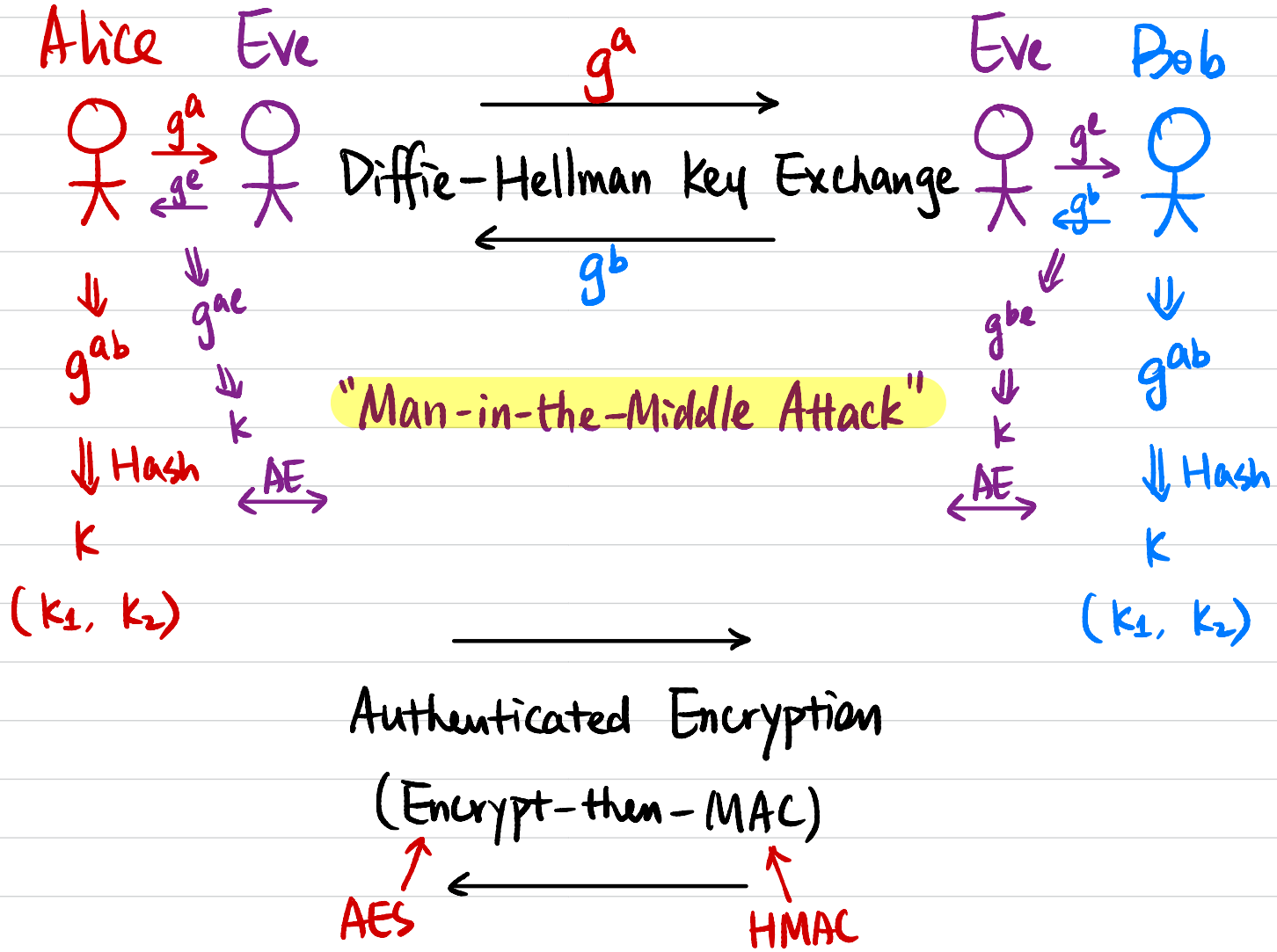
- Fixed-length messages of length $3 \cdot \lambda$ Yes!
- Arbitrary-length messages No!

Query $m_1 \rightarrow$ Get t_1

Query $t_1 \rightarrow$ Get t_2

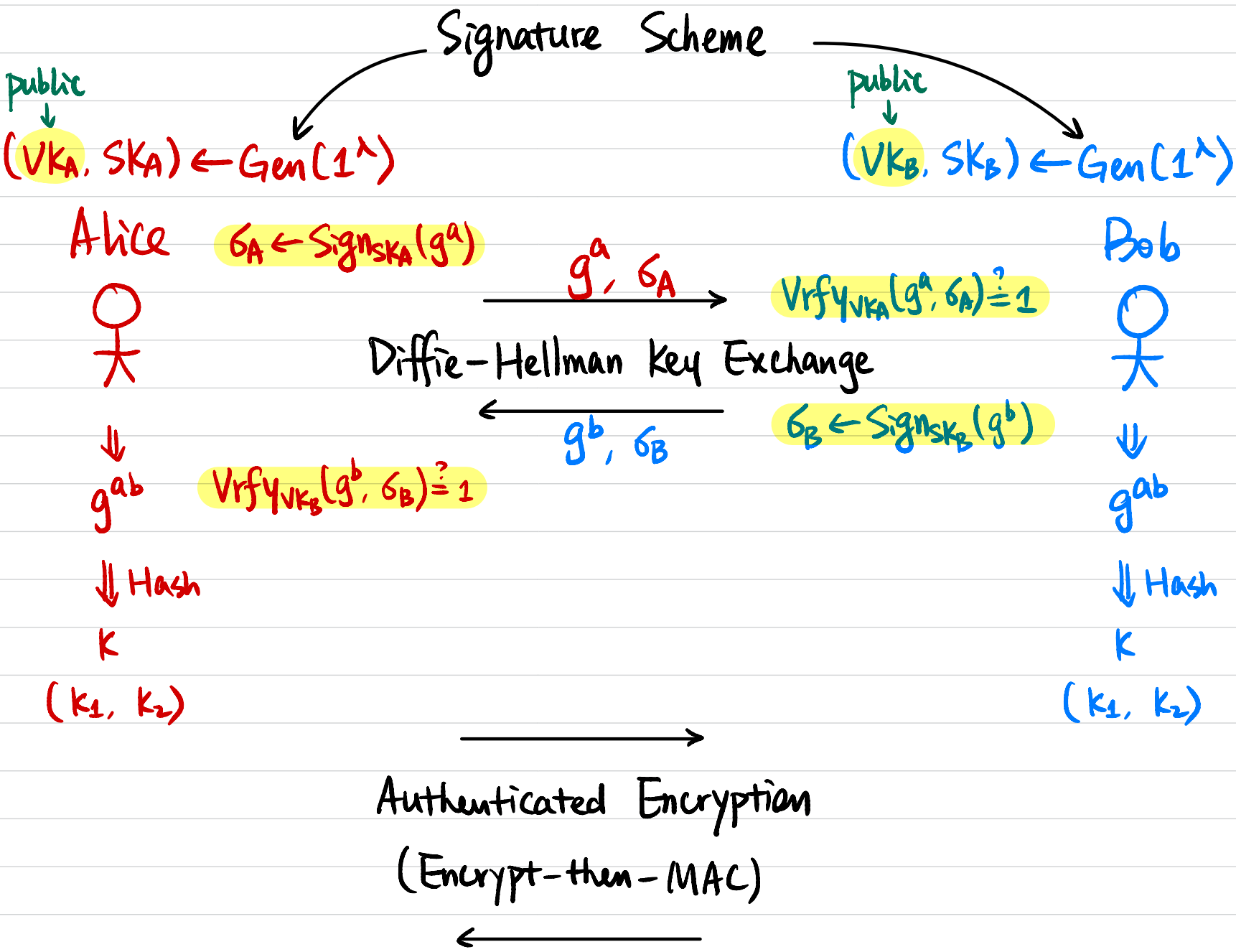
Output $m^* = m_1 \| 0^\lambda$, $t^* = t_2$

Putting it All Together: Secure Communication



Any security issue?

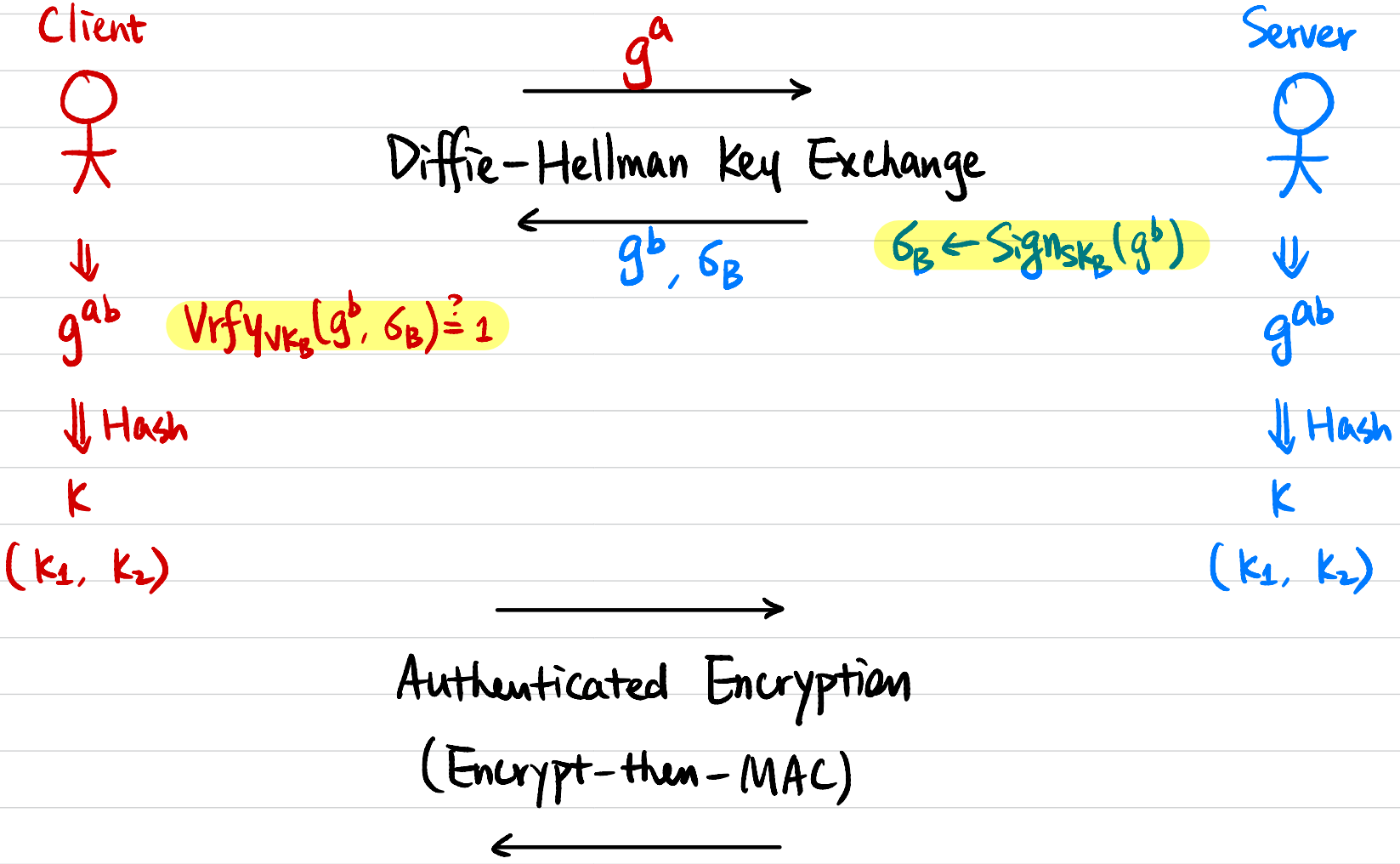
Authenticated Key Exchange



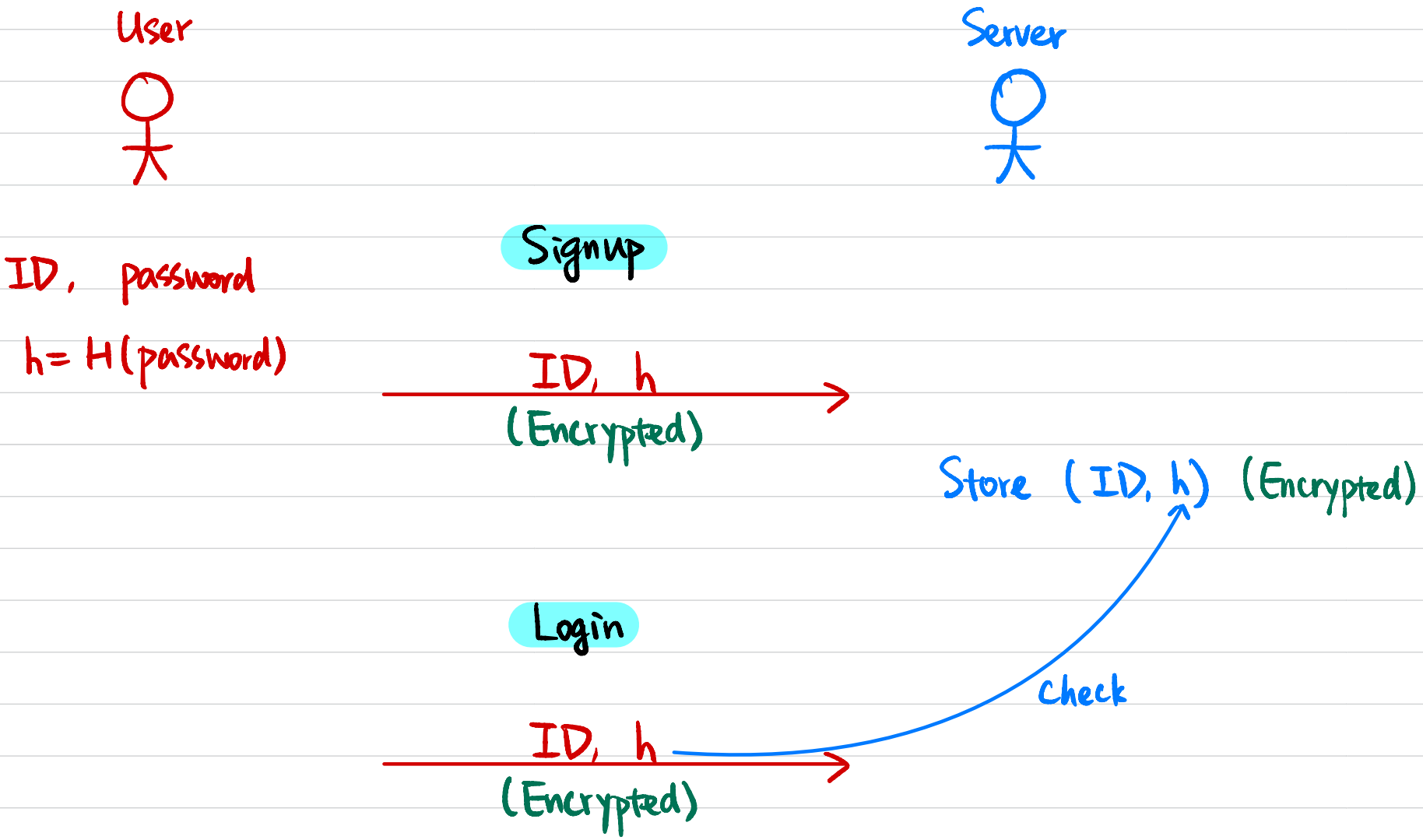
One-Sided Secure Authentication

public

$(VK_B, SK_B) \leftarrow Gen(1^\lambda)$



Password-Based Authentication



Password Security ?

Online Dictionary Attack

User

Server

ID, password

$h = H(\text{password})$

Signup

ID, h

(Encrypted)

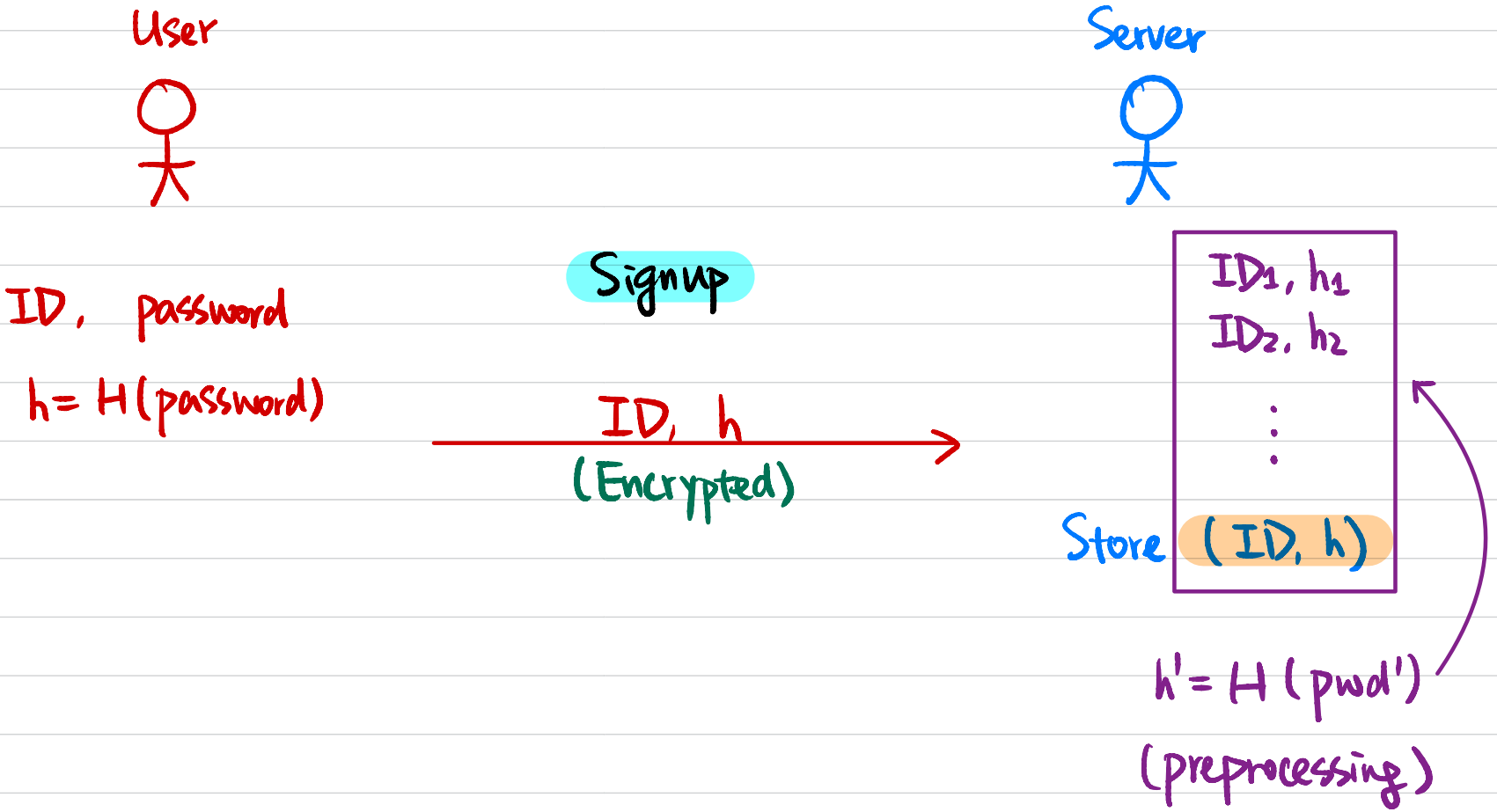
Store (ID, h) (Encrypted)

Login

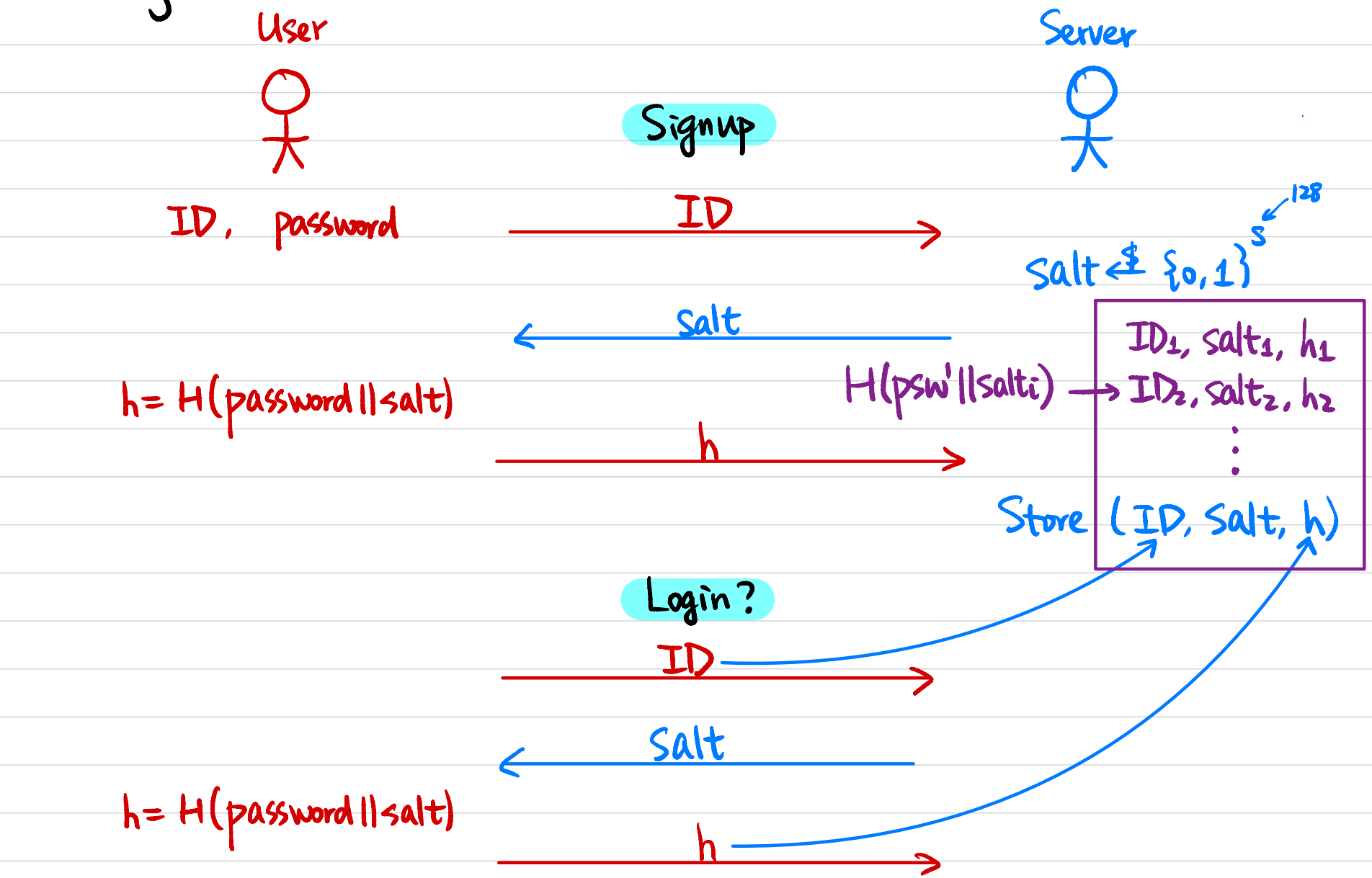
ID, $h' = H(\text{pwd}')$

(Encrypted)

Offline Dictionary Attack

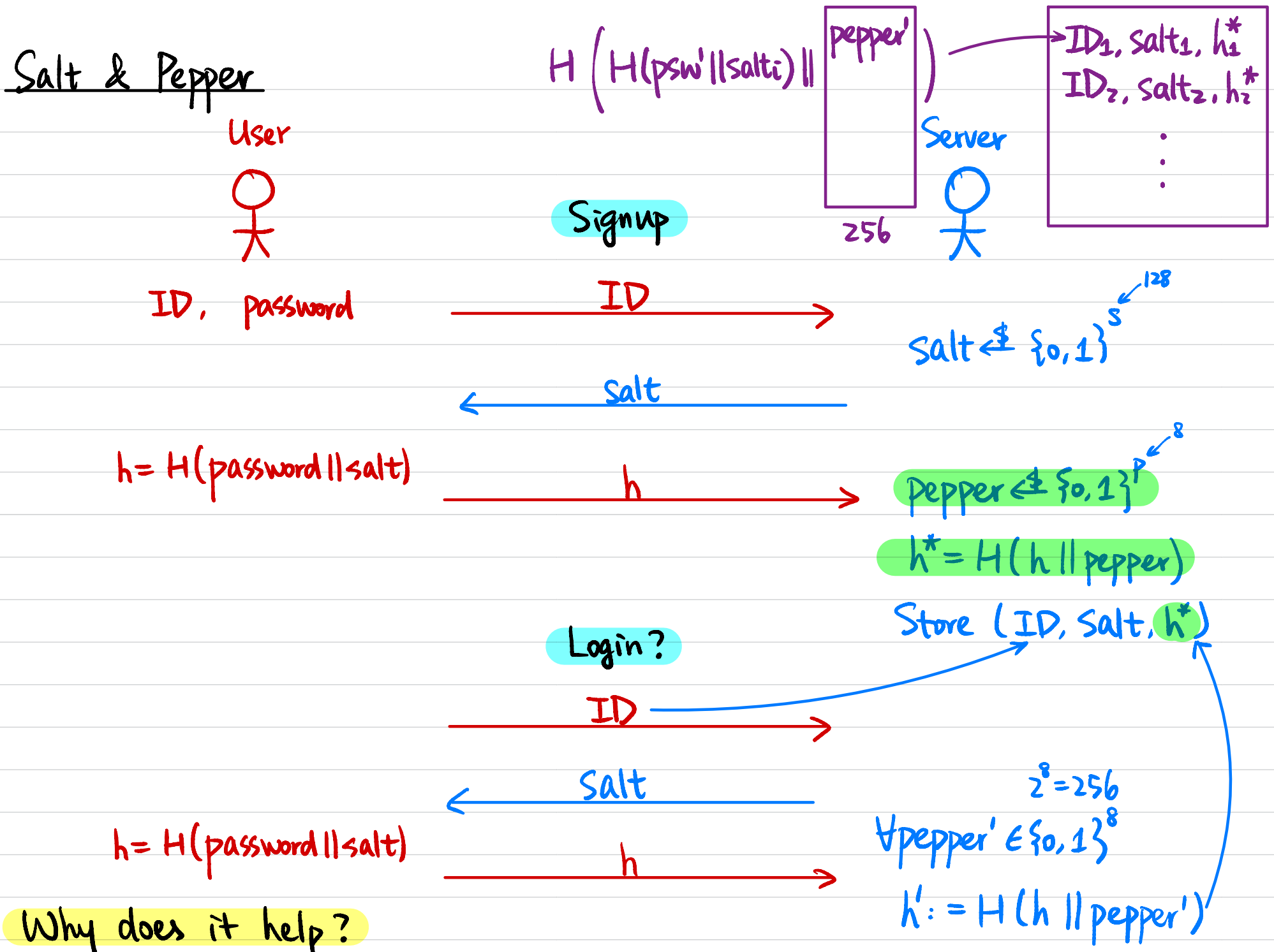


Salting



Why does it help?

Salt & Pepper



Why does it help?

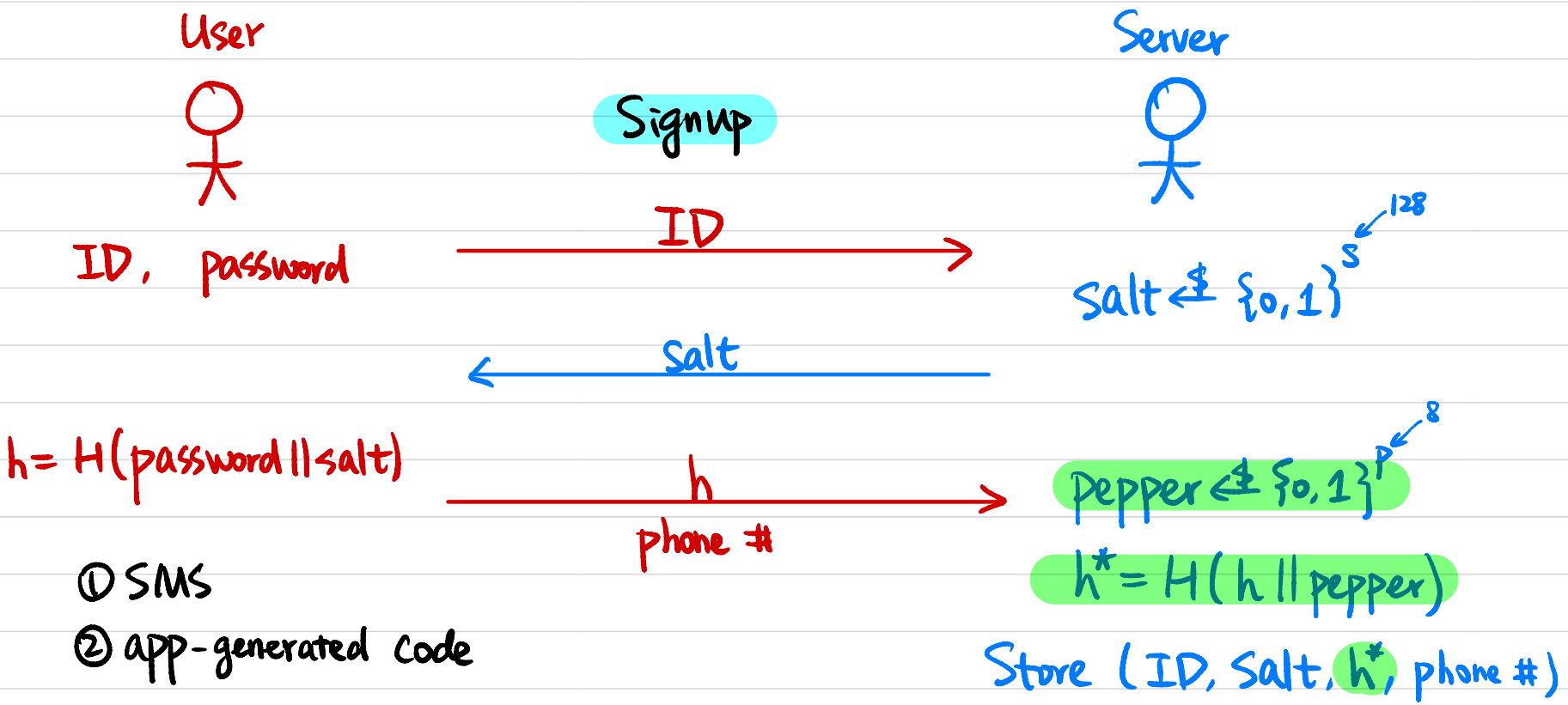
Slow Hash Functions

- Computation-heavy hash function
  - ↳ Compose SHA256 in a certain way.

Application-Specific Integrated Circuit (ASIC) → blockchain mining

- Memory-hard hash functions
  - ↳ Scrypt

Two-Factor Authentication (2FA)



How would you design it ?