

Cryptographic hash functions

$$H: \{0, 1\}^* \rightarrow \{0, 1\}^L$$

(SHA256 $L=256$)

deterministic

$H(x)$ = hash of x

digest

fingerprint

$$\{0, 1\}^{1000} \rightarrow \{0, 1\}^{256}$$

$$2^{1000}$$

$$\rightarrow 2^{256}$$

Correctness: deterministic

Efficiency: computing H should be easy

Security:

1) One-way "preimage resistance"

$$\forall \text{Adv}, \Pr [x \leftarrow \$\text{rand}; y = H(x); \text{Adv}(y) \rightarrow x] \\ = \text{negl} \left(\frac{1}{2^{\text{size of output of } H; L}} \right)$$

~~$$\forall \text{Adv}, \forall x, \Pr [\text{Adv}(H(x)) \rightarrow x] = \text{negl}$$~~

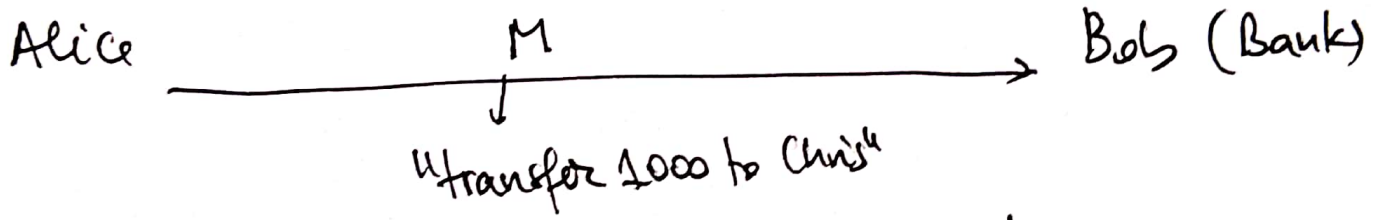
~~$$"2" \rightarrow H(2)$$~~

2) Collision-resistance (CR)

It is infeasible to find (x, x') s.t. $(x \neq x')$
and $H(x) = H(x')$

SHA256 is assumed currently to be CR

	Symmetric-Key	Asym-Key
Confidentiality	symmetric-key encryption (AES-CBC)	public-key enc (ElGamal)
Integrity & auth	MAC (AES-EMAC)	Digital Signatures (RSA)



authenticity: M comes from the expected user
 integrity: M was not modified

↑ El Gamal ($g^r \pmod p$; $M \cdot g^k \pmod p$)

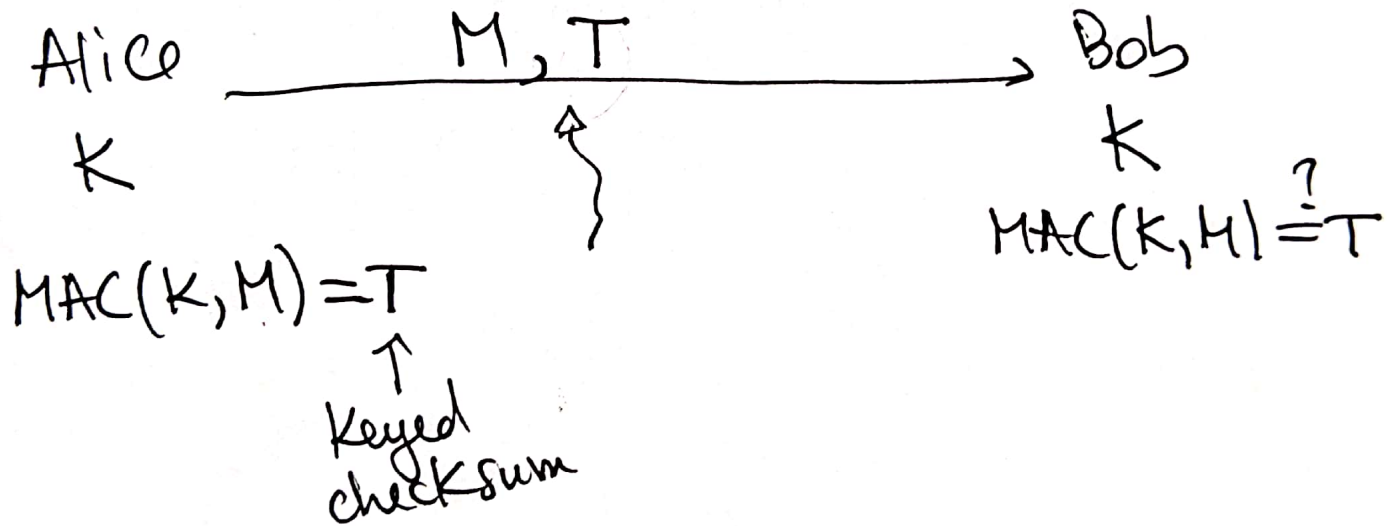
$\times 2$

$= 2M \cdot g^k \pmod p$

↓
easy to modify

Encryption does not provide these properties

MAC (Message authentication code)



Correctness: determinism

Efficiency: compute MAC should be poly time

Security: EU-CPA existentially-unforgeable
under chosen plaintext
attack

Adv cannot forge a MAC for a new message

Challenger

K

$\leftarrow M_i$

$\xrightarrow{\text{MAC}(K, M_i)}$

$\leftarrow M \neq \{M_i\}$

T

Adv



$\forall \text{Adv}, \Pr [K \leftarrow \text{Keygen}(); \text{Adv}() \rightarrow (M, T)$
s.t. $M \neq M_i, \text{MAC}(K, M) = T]$
 $= \text{negl}$

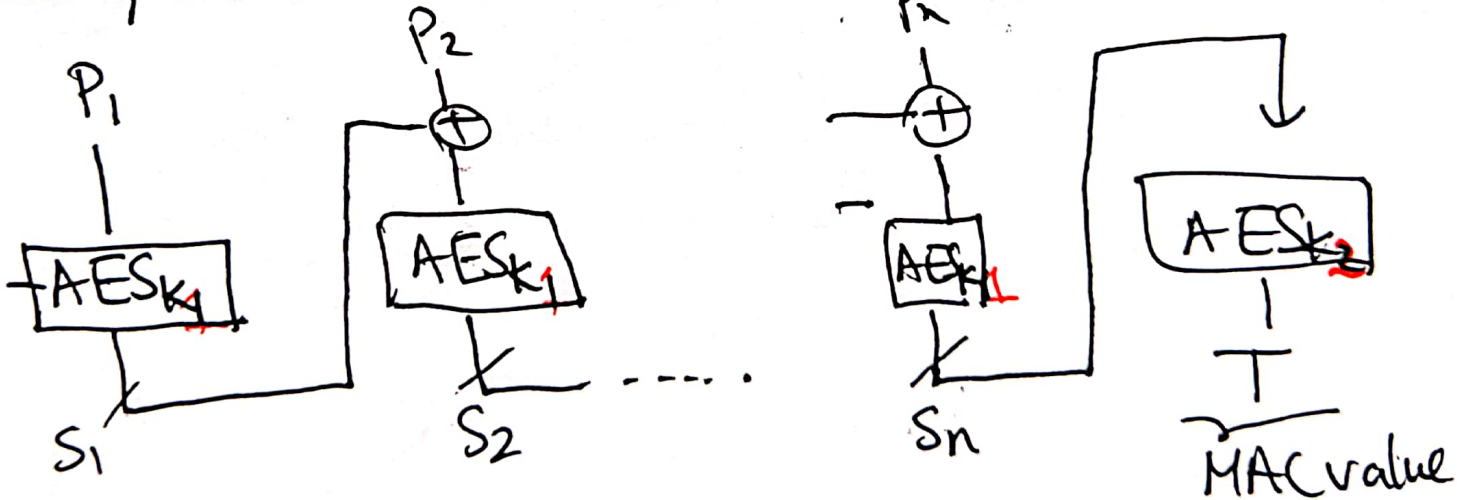
AES-EMAC

Keygen() \rightarrow

Choose 2 keys $(K_1, K_2) = K$
 $\underbrace{\hspace{1.5cm}}_{256}$

MAC(K, M): $\overset{128}{\text{---}}$

Split M into $P_1 || P_2 \dots || P_n$



~~Propose MAC'(K, M) = (S_1, \dots, S_n) *insecure*~~

~~can forge MAC' ; Adv asks MAC M = (P_1 || P_2) \rightarrow (S_1, S_2)
M* = P_1 ; MAC is S_1~~

Propose $\text{MAC}''(K, M) = S_n$ (insecure MAC for exercise)

Adv asks for P_1 receives S_1

$P_1' \rightarrow S_1'$

$P_1 \parallel P_2 \rightarrow T$

forgeable
 \Rightarrow points to needing K_2 at the end

Adv forges MAC for $(P_1'; S_1 \oplus P_2 \oplus S_1')$ is T

Alice

K

M

$$C = \text{Enc}(K_E, M);$$

$$\text{MAC}(K_M, C)$$

both confidentiality
& integrity/auth

Bob

K =

$$= (K_E, K_M)$$

1. checks MAC
for C using K_M
?

$$\text{MAC}(K_M, C) = T \checkmark$$

2. Decrypt to get M

$$\text{Dec}(K_E, C) = M$$

K_E, K_M