CRYPTOGRAPHIC HASH FNs & MACs

ONE-WAY FNs (OWF)

→ easy to compute but hard to invert

→ $\forall y$, there is no efficient algo. which can compute x such that $f(x) = y$

e.g. Constant fns $f(x) = c$ are NOT OWF

Multiplication of large primes is an OWF

COLLISION-RESISTANT FNs (CRF)

→ no efficient algo. tht. can find two messages m_1 and m_2 s.t. $f(m_1) = f(m_2)$

e.g. Constant fns are not CRF; $\forall m_1, m_2 . f(m_1) = f(m_2)$

Multiplication of large primes is a CRF

CHF

→ A CHF $H: M \to T$ is a fn tht satisfies:

1) $|M| >> |T|$ ⇒ collisions are unavoidable!
2) it is easy to compute the hash value for any given message
3) it is hard to retrieve a message frm its hashed value → OWF
4) it is hard to find 2 diff. messages w/ same hash value → CRF

APPLICATIONS

1) Digital signature generation & verification
2) File integrity
3) Password verification
4) Key derivation
5) Used to build other crypto primitives (e.g. block cipher, MAC...)

BIRTHDAY ATTACK

→ is a type of cryptographic attack

→ Let $H : M \to \{0,1\}^n$ be a CHF

An algo. to find a collision in time $O(\sqrt{2^n}) = O(2^{n/2})$ hashes:

1) Choose $2^{n/2}$ random messages in M : $m_1, \ldots, m_{2^{n/2}}$

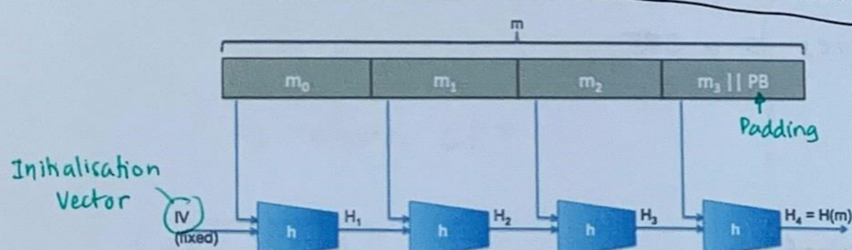
2) For $i = 1, \ldots, 2^{n/2}$, compute $t_i = H(m_i)$

3) If there exists a collision, return (m_i, m_j), else go to 1)

↳ $\exists i, j.\ t_i = t_j$

→ CHF shld. have output length $n \geq 256$!

THE MERKLE-DAMGARD (MD) CONSTRUCTION



- Compression function: $h : \mathcal{T} \times \mathcal{X} \to \mathcal{T}$
- PB: 1000…0||mes-len (add extra block if needed)

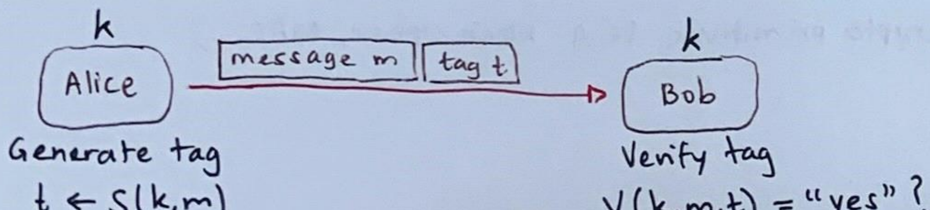
Theorem
Let H be built using the MD construction to the compression function h. If H admits a collision, so does h.

MAC [Message Authentication Code]

GOAL: MESSAGE INTEGRITY + Authentication

→ is a pair of algos (S, V) defined over (K, M, T)

- $S : K \times M \to T$
- $V : K \times M \times T \to \{\bot, \top\}$
- Consistency: $V(k, m, S(k,m)) = \top$

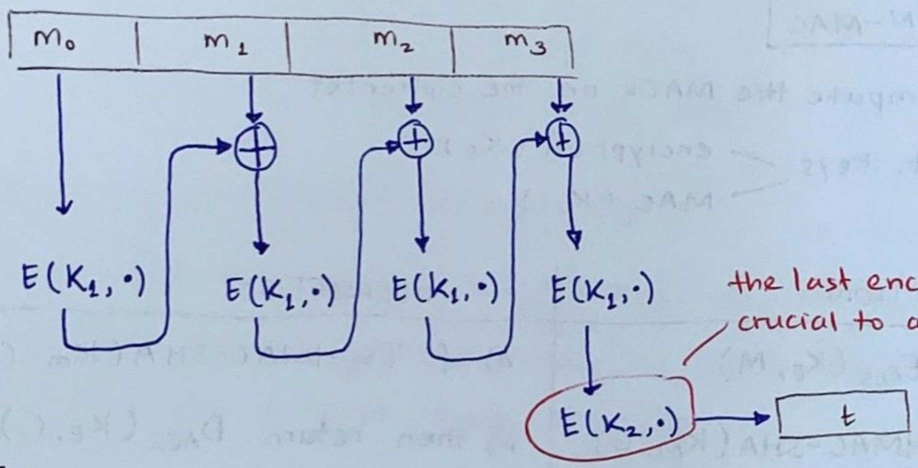


→ It's hard to compute a valid pair $(m, S(k,m))$ w/o knowing key k

BLOCK CIPHER & MAC

- → MAC algos. can be constructed from block cipher algo.
- → However, block ciphers can only process 128 / 256 bits
- → So, we need to construct MACs for long messages.

ECBC-MAC



the last encryption is crucial to avoid forgeries!

PMAC [Parallelizable MAC]

- → can evaluate block ciphers in parallel

HMAC

- → MAC built from CHFs

$$HMAC(k, m) = H(k \oplus OP \,||\, H(k \oplus IP \,||\, m))$$

Publicly known padding constants

AUTHENTICATED ENCRYPTION

→ Plain encryption is malleable; the decryption algo. never fails

→ Decryption shld. fail if a ciphertxt was not computed using the key

→ GOAL: — Provide data confidentiality, integrity & authenticity silmutaneously

ENCRYPT-THEN-MAC

1) Always compute the MACs on the ciphertxt

2) Use 2 diff. keys
   - encryption (K_E)
   - MAC (K_M)

ENCRYPTION	DECRYPTION
a) $C \leftarrow E_{AES}(K_E, M)$	a) if $T = \text{HMAC-SHA}(K_m, C)$
b) $T \leftarrow \text{HMAC-SHA}(K_M, C)$	b) then return $D_{AES}(K_E, C)$
c) return $C \| T$	c) else return $\perp$

AES-GCM

→ combines
   - Galois field based <u>one-time MAC</u> for authentication
   - AES based <u>counter mode</u> for encryption

→ One-time MAC is encrypted too ⇒ Secure for many messages