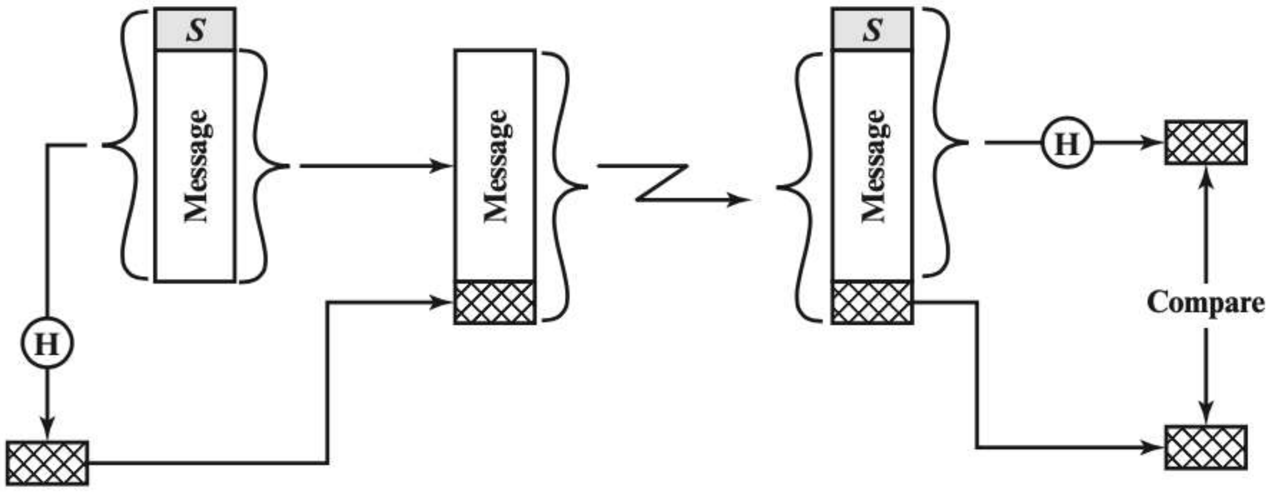
Lecture 16

Hash Function

Hash functions

- Hash function: h = H(M)
    - M can be of any size
    - h is always of fixed size
    - Typically, h << size(M)

One use case - using hash function



(c) Using secret value

- Initialization: A and B share a common secret, S_{AB}
- Message, M
- A calculates $MD_M = H (S_{AB} || M)$
- B recalculates MD'_M, and check
- $MD'_M = MD_M$

This scheme cannot provide authentication.

Requirements for secure hash functions

- 1. can be applied to any sized message M
- 2. produces fixed-length output h
- 3. is easy to compute $h=H(M)$ for any message M
- 4. given h is infeasible to find x s.t. $H(x)=h$
    - one-way property or preimage resistance
- 5. given x is infeasible to find x' s.t. $H(x')=H(x)$
    - weak collision resistance or second pre-image resistant
- 6. infeasible to find any pair of x, x' s.t. $H(x')=H(x)$
    - strong collision resistance

Hash Function: Collision Resistance

- **Collision**: Two different inputs with the same output
  - $x \neq x'$ and $H(x) = H(x')$
  - Can we design a hash function with no collisions?
    - No, because there are more inputs than outputs (pigeonhole principle)
  - However, we want to make finding collisions *infeasible* for an attacker
- **Collision resistance**: It is infeasible to (i.e. no polynomial time attacker can) find any pair of inputs $x' \neq x$ such that $H(x) = H(x')$

Secure hash function

- A hash function that satisfies the first five properties is referred to as a weak hash function
- **Security:** random/unpredictability, no predictable patterns for how changing the input affects the output
  - Changing 1 bit in the input causes the output to be completely different
  - Also called “random oracle” assumption
- A message digest
  - a fixed size numeric representation of the contents of a message, computed by a hash function
- Examples: SHA-1 (Secure Hash Algorithm 1), SHA-2, SHA-3, MD5

Hash Function: Examples

- MD5
    - Output: 128 bits
    - Security: Completely broken
- SHA-1
    - Output: 160 bits
    - Security: Completely broken in 2017
    - Was known to be weak before 2017, but still used sometimes
- SHA-2
    - Output: 256, 384, or 512 bits (sometimes labeled SHA-256, SHA-384, SHA-512)
    - Not currently broken, but some variants are vulnerable to a length extension attack
    - Current standard
- SHA-3 (Keccak)
    - Output: 256, 384, or 512 bits
    - Current standard (not meant to replace SHA-2, just a different construction)

Length Extension Attacks

- **Length extension attack**: Given *H*(*x*) and the length of *x*, but not *x*, an attacker can create *H*(*x* || *m*) for any *m* of the attacker's choosing
  - Length extension attack - Wikipedia
- SHA-256 (256-bit version of SHA-2) is vulnerable
- SHA-3 is not vulnerable