Hash Functions, Message Authentication Codes

Ahmet Burak Can
Hacettepe University
abc@hacettepe.edu.tr

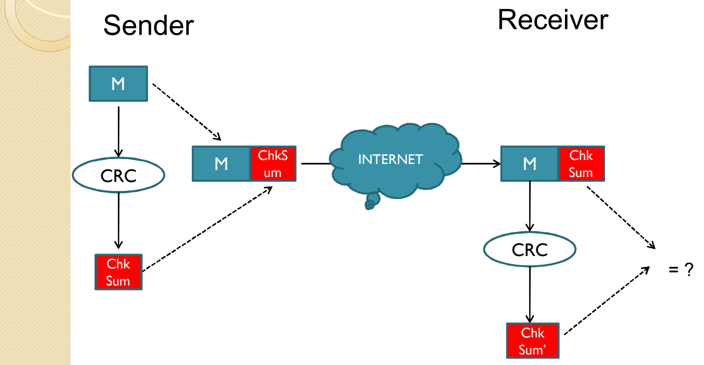
Security Services

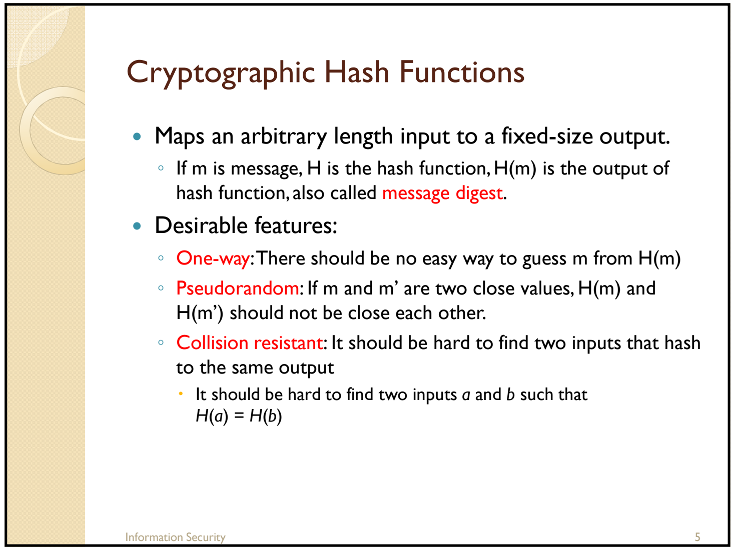
- ✓ Confidentiality : Symmetric encryption solves
- Integrity
- Authentication
- Non-repudiation
- Access control
- Availability

Integrity in Networking

- Sender computes a CRC for the message
- Sender appends the CRC code to the message and sends them to the receiver
- The receiver computes the CRC of the message.
  - If the CRC appended to the message is equal to the computed one, the message is unchanged with a high probability.
  - If the CRCs do no match, the message is changed during the transmission.

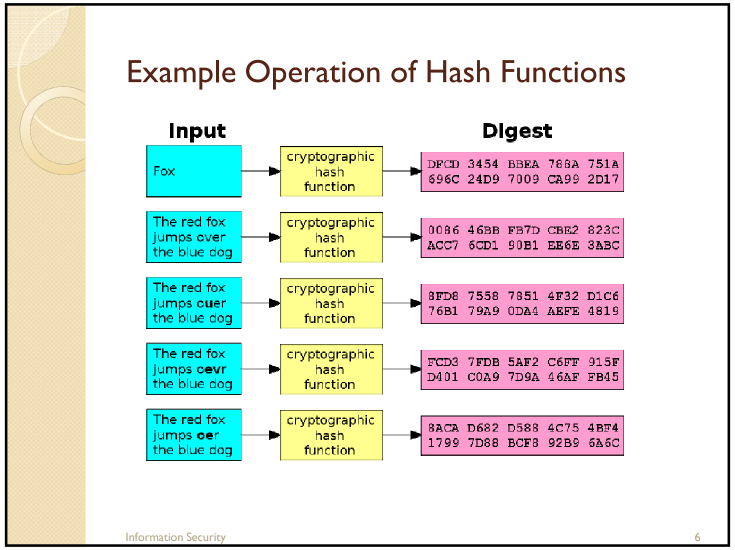
CRC Checksum in Networking





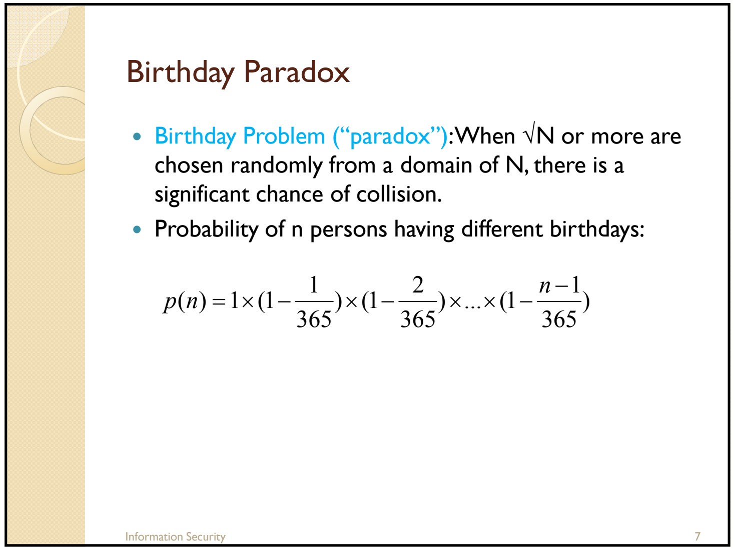
Cryptographic Hash Functions

- Maps an arbitrary length input to a fixed-size output.
  - If m is message, H is the hash function, H(m) is the output of hash function, also called message digest.
- Desirable features:
  - One-way: There should be no easy way to guess m from H(m)
  - Pseudorandom: If m and m' are two close values, H(m) and H(m') should not be close each other.
  - Collision resistant: It should be hard to find two inputs that hash to the same output
    - It should be hard to find two inputs *a* and *b* such that *H*(*a*) = *H*(*b*)



Example Operation of Hash Functions

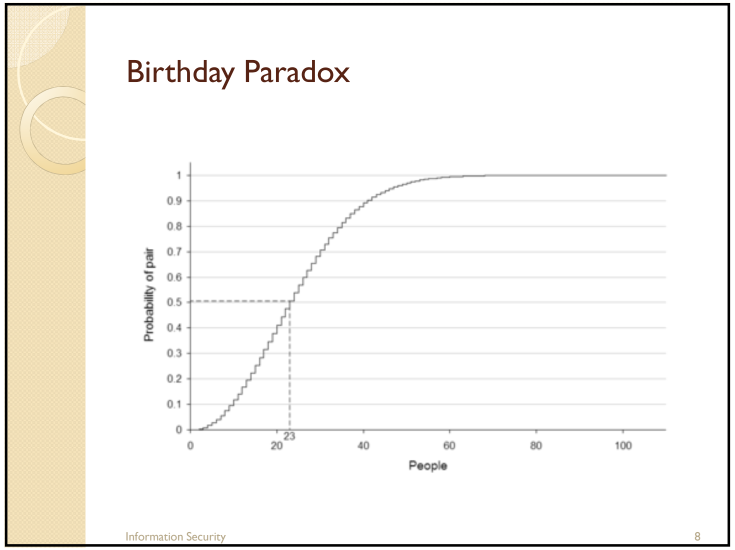
Input		Digest
Fox	cryptographic hash function	DFCD 3454 BBEA 788A 751A 696C 24D9 7009 CA99 2D17
The red fox jumps over the blue dog	cryptographic hash function	0086 46BB FB7D CBE2 823C ACC7 6CD1 90B1 EE6E 3ABC
The red fox jumps ouer the blue dog	cryptographic hash function	8FD8 7558 7851 4F32 D1C6 76B1 79A9 0DA4 AEFE 4819
The red fox jumps oevr the blue dog	cryptographic hash function	FCD3 7FDB 5AF2 C6FF 915F D401 C0A9 7D9A 46AF FB45
The red fox jumps oer the blue dog	cryptographic hash function	8ACA D682 D588 4C75 4BF4 1799 7D88 BCF8 92B9 6A6C



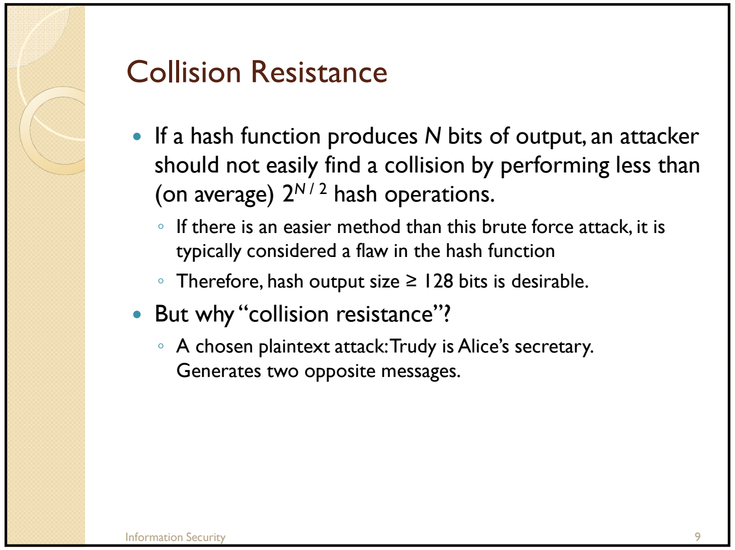
Birthday Paradox

- Birthday Problem ("paradox"): When √N or more are chosen randomly from a domain of N, there is a significant chance of collision.
- Probability of n persons having different birthdays:

$$p(n) = 1 \times (1 - \frac{1}{365}) \times (1 - \frac{2}{365}) \times \ldots \times (1 - \frac{n-1}{365})$$

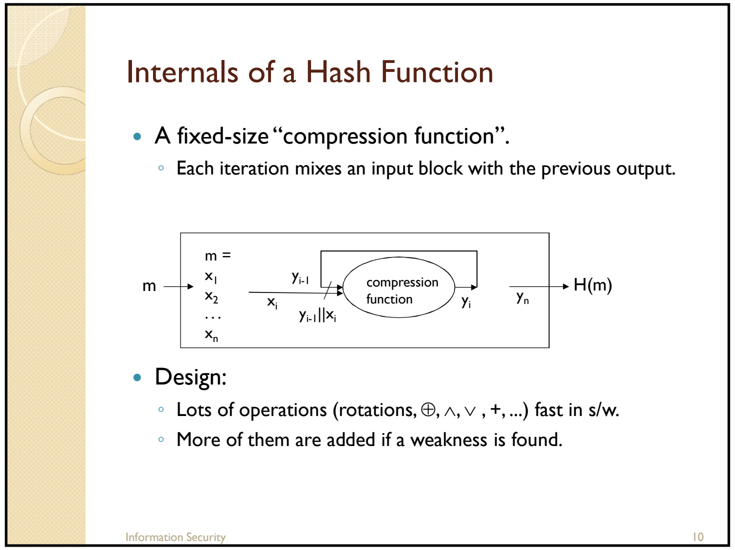


Birthday Paradox



Collision Resistance

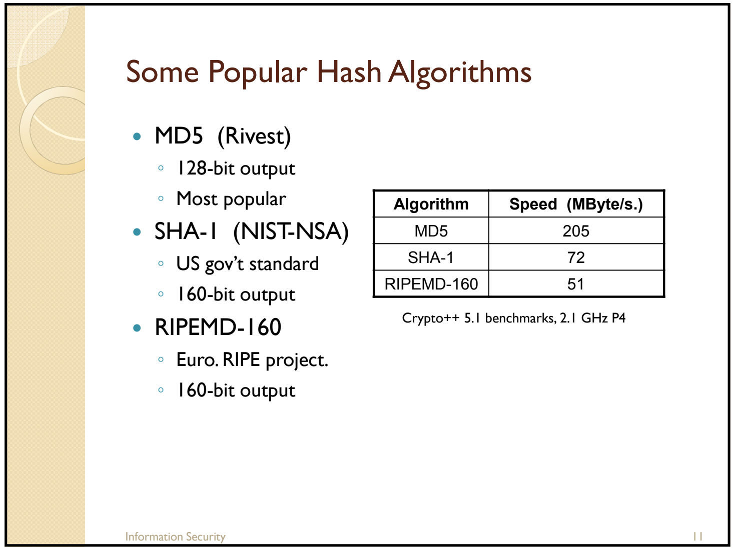
- If a hash function produces *N* bits of output, an attacker should not easily find a collision by performing less than (on average) $2^{N/2}$ hash operations.
  - If there is an easier method than this brute force attack, it is typically considered a flaw in the hash function
  - Therefore, hash output size ≥ 128 bits is desirable.
- But why "collision resistance"?
  - A chosen plaintext attack: Trudy is Alice's secretary. Generates two opposite messages.



Internals of a Hash Function

- A fixed-size "compression function".
  - Each iteration mixes an input block with the previous output.

- Design:
  - Lots of operations (rotations, ⊕, ∧, ∨ , +, ...) fast in s/w.
  - More of them are added if a weakness is found.

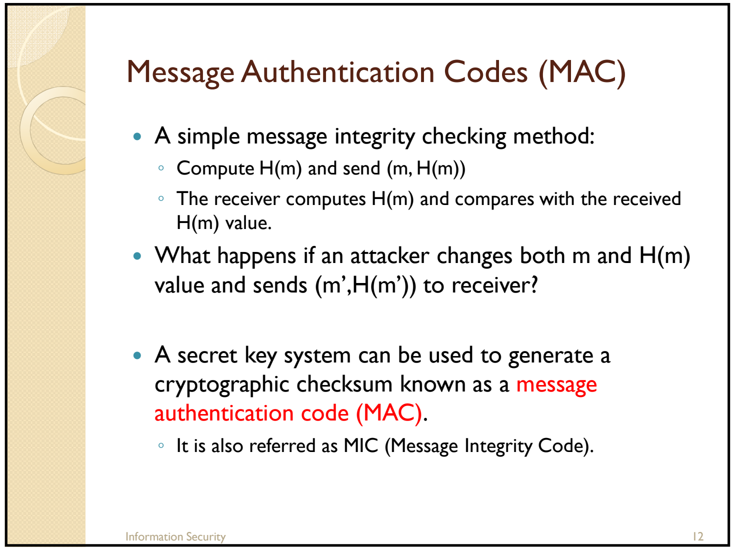


Some Popular Hash Algorithms

- MD5 (Rivest)
  - 128-bit output
  - Most popular
- SHA-1 (NIST-NSA)
  - US gov't standard
  - 160-bit output
- RIPEMD-160
  - Euro. RIPE project.
  - 160-bit output

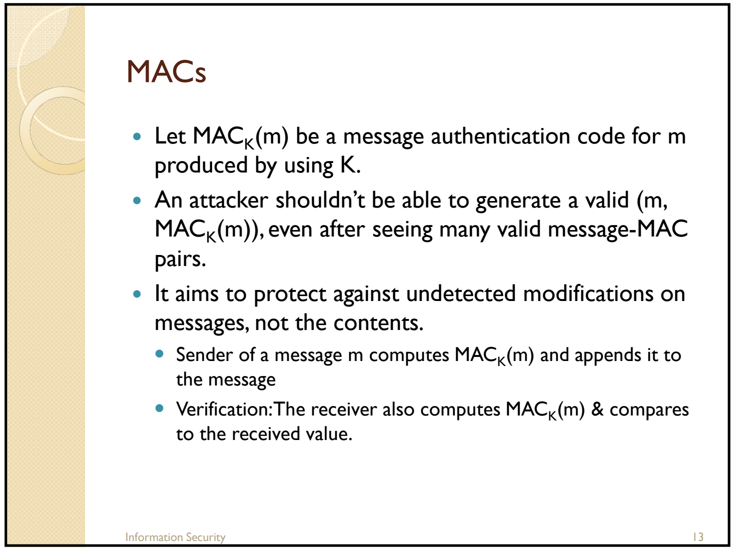
Algorithm	Speed (MByte/s.)
MD5	205
SHA-1	72
RIPEMD-160	51

Crypto++ 5.1 benchmarks, 2.1 GHz P4



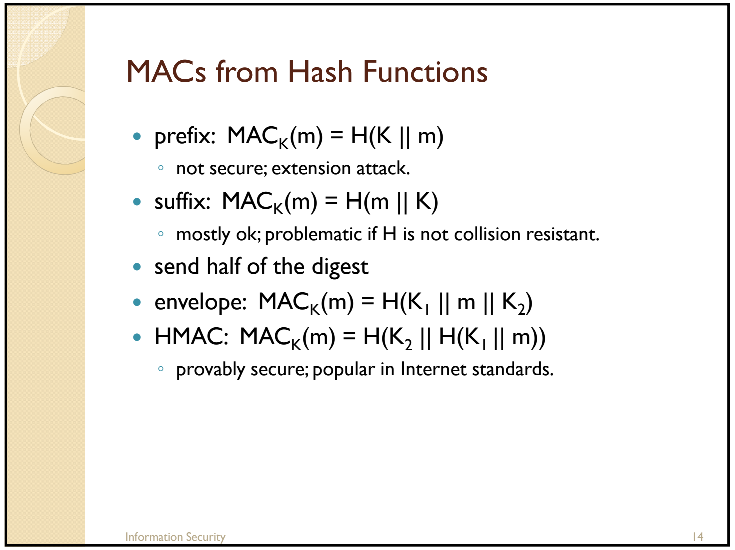
Message Authentication Codes (MAC)

- A simple message integrity checking method:
  - Compute H(m) and send (m, H(m))
  - The receiver computes H(m) and compares with the received H(m) value.
- What happens if an attacker changes both m and H(m) value and sends (m',H(m')) to receiver?
- A secret key system can be used to generate a cryptographic checksum known as a message authentication code (MAC).
  - It is also referred as MIC (Message Integrity Code).



MACs

- Let $MAC_K(m)$ be a message authentication code for m produced by using K.
- An attacker shouldn't be able to generate a valid (m, $MAC_K(m)$), even after seeing many valid message-MAC pairs.
- It aims to protect against undetected modifications on messages, not the contents.
  - Sender of a message m computes $MAC_K(m)$ and appends it to the message
  - Verification: The receiver also computes $MAC_K(m)$ & compares to the received value.



MACs from Hash Functions

- prefix: $MAC_K(m) = H(K \| m)$
  - not secure; extension attack.
- suffix: $MAC_K(m) = H(m \| K)$
  - mostly ok; problematic if H is not collision resistant.
- send half of the digest
- envelope: $MAC_K(m) = H(K_1 \| m \| K_2)$
- HMAC: $MAC_K(m) = H(K_2 \| H(K_1 \| m))$
  - provably secure; popular in Internet standards.