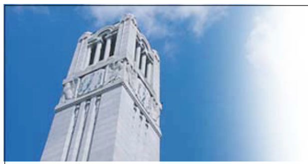
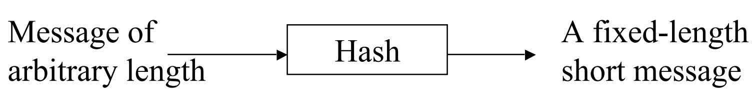


NC STATE UNIVERSITY Computer Science

CSC 474/574 Information Systems Security

Topic 2.3 Hash Functions

Hash Function



- Also known as
  - Message digest
  - One-way transformation
  - One-way function
  - Hash
- Length of $H(m)$ much shorter then length of m
- Usually fixed lengths: 128 or 160 bits

Requirements for a Hash Function

- Consider a hash function H
  - Flexibility: Can be applied to a block of data of any size
  - Convenience (for check): produce a fixed-length short output.
  - Performance: Easy to compute H(*m*)
  - One-way property: Given H(*m*) but not *m*, it's difficult to find *m*
  - Weak collision resistance (free): Given H(*m*), it's difficult to find *m'* such that H(*m'*) = H(*m*).
  - Strong collision resistance (free): Computationally infeasible to find m_1, m_2 such that $H(m_1) = H(m_2)$

Birthday Paradox

- Question:
  - What is the minimum value of *k* such that the probability is greater than 0.5 that at least two people in a group of *k* people have the same birthday?
    - Ignore February 29 and assume each birthday is equally likely.
  - Probability of *k* people having *k* different birthdays: $Q(365,k)=365!/(365-k)!365^k$
  - Probability that at least two people have the same birthday: $P(365,k)=1-Q(365,k)$
  - *K* is about 23.

Generalization of Birthday Paradox

- Given a random variable that is an integer with uniform distribution between 1 and n and a selection of k instances of the random variables, what is the least value of k such that the probability $P(n,k)$ is greater than 0.5 that there is at least one duplicate?
  - $P(n,k) > 1 - e^{-k*(k-1)/2n}$
  - For large n and k, we have

$$k = \sqrt{2(\ln 2)n} = 1.18\sqrt{n} \approx \sqrt{n}$$

  - Intuition: How many k do we need to have a collision with P=0.5?
- Implication
  - For a hash function H with 2^m possible outputs, if we apply H to $k=(2^m)^{1/2}=2^{m/2}$ random inputs, the probability that there is at least one duplicate is greater than 0.5.

Birthday Attack

- The source, A, is prepared to sign a message
- The opponent generates $2^{m/2}$ variations on the message, and prepares $2^{m/2}$ variations on the fraudulent message.
- The opponent compares the two sets of messages to find a pair of messages that produces the same hash value. The probability of success is greater than 0.5. The opponent repeats generating variations until a match is found.
- The opponent offers the valid variation to A for signature, but attaches the signature to the fraudulent variation.

How Many Bits for Hash?

- *m* bits, takes $2^{m/2}$ to find two with the same hash at the probability 0.5
- 64 bits, takes 2^{32} messages to search duplicate
- Need at least 128 bits

Building Hash Using Block Chaining Techniques

- Divide M into fixed-size blocks $M_1, M_2, \ldots, M_n$
- Compute the hash as follows
  - H_0=Initial value
  - $H_i=E_{Mi}(H_{i-1})$
  - Hash value $G=H_n$
- Weakness
  - Birthday attack (reason: hash value is too short)
  - Meet-in-the-middle attack

Building Hash Using Block Chaining Techniques (Cont'd)

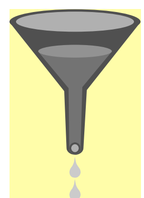
- Meet-in-the-middle attack
  - Get the correct hash value G
  - Construct any message in the form $Q_1, Q_2, \ldots, Q_{n-2}$
  - Compute $H_i = E_{Qi}(H_{i-1})$ for $1 \leq i \leq (n-2)$.
  - Generate $2^{m/2}$ random blocks; for each block X, compute $E_X(H_{n-2})$.
  - Generate $2^{m/2}$ random blocks; for each block Y, compute $D_Y(G)$.
  - With high probability there will be an X and Y such that $E_X(H_{n-2}) = D_Y(G)$.
  - Form the message $Q_1, Q_2, \ldots, Q_{n-2}$, X, Y. It has the hash value G.

Modern Hash Functions

- MD5
  - Previous versions (i.e., MD2, MD4) have weaknesses.
- SHA (Secure Hash Algorithm)
- SHA-1
- RIPEMD-160

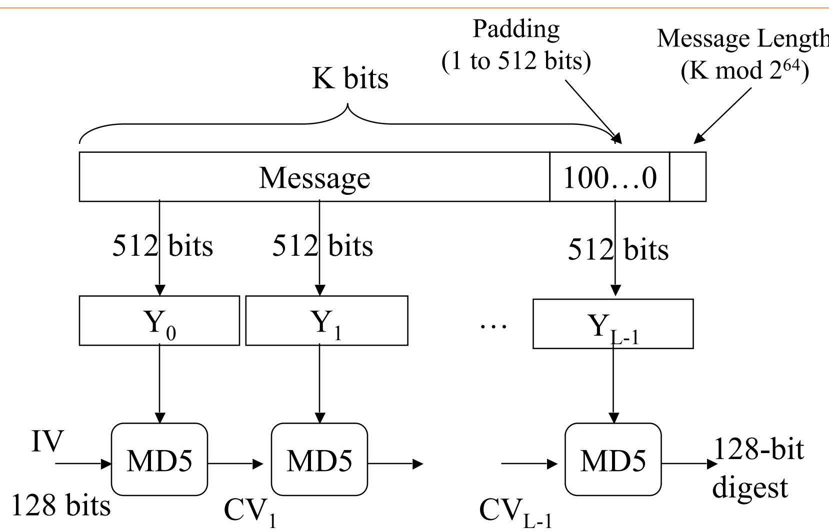
MD5: Message Digest Version 5

input Message



Output 128 bits Digest

MD5: A High-Level View



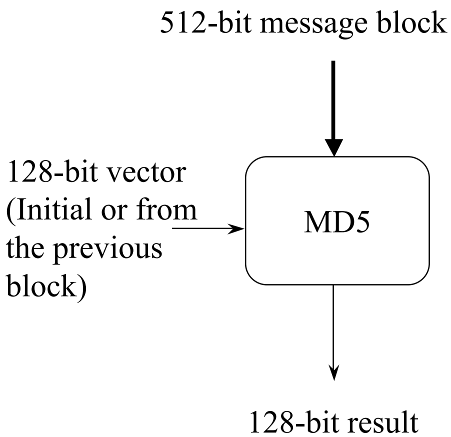
Padding

- Given original message M, add padding bits "10*" such that resulting length is 64 bits less than a multiple of 512 bits.
- Append (*original length in bits* mod 2^{64}), represented in 64 bits to the padded message
- Final message is chopped 512 bits a block
- Exercise:
  - How to add padding bits to a message that is already a multiple of 512 bits?

MD5 (Intermediate) Buffer

- Used to hold intermediate and final result of MD5.
- 128 bits
- Represented as four 32-bit words
  - (A,B,C,D)
  - Initially, A=0x67452301, B=0xEFCDAB89, C=0x98BADCFE, D=0x10325476
  - Stored in little-endian format, A=0x01234567, B=0x89ABCDEF, C=0xFEDCBA98, D=0x76543210.

Processing of A Single Block



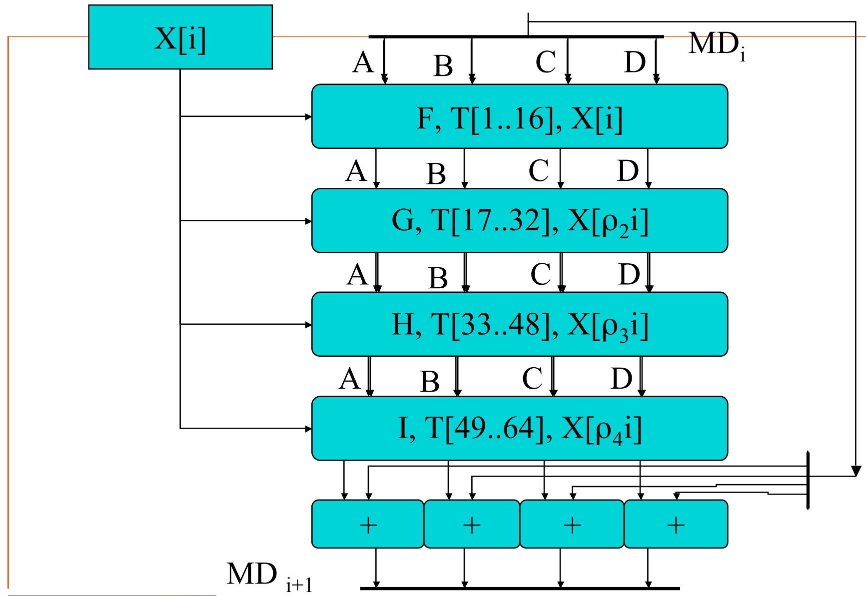
Primitive operations used in MD5:

$F(x,y,z) = (x \wedge y) \vee (\sim x \wedge z)$
$G(x,y,z) = (x \wedge z) \vee (y \wedge \sim z)$
$H(x,y,z) = x \oplus y \oplus z$
$I(x,y,z) = y \oplus (x \wedge \sim z)$
+: addition mod 2^{32}
$x \lll y$: x left rotate y bits

Processing of A Single Block (Cont'd)

- Every message block contains 16 *32-bit words*:
  - X[0] X[1] X[2] … X[15]
- Every stage consists of 4 rounds over the message block, each modifying the MD5 buffer (A,B,C,D).
  - The four rounds use functions F, G, H, I, respectively.
- Each round uses one-fourth of a 64-element table T[1…64].
  - T[i] = 2^{32}*abs(sin(*i*)) represented in 32 bits.
- The output of the fourth round is added to the input to the first round.

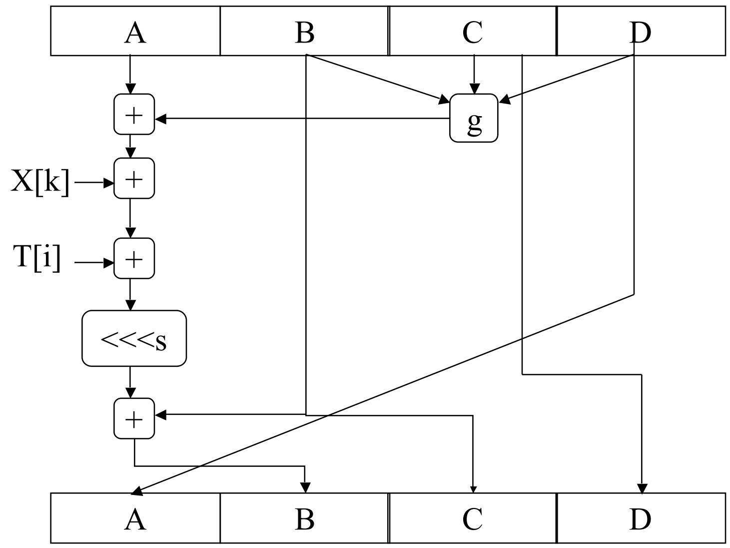
Processing of Block m_i : 4 Rounds



Logic of Each Round

- Each round consists of 16 steps
- Each step is of the form
  – A←B+((A+g(B,C,D)+X[k]+T[i])<<<s)
    - Function g is one of F, G, H, I
    - X[k] is the word in the input
    - T[i] is the ith word in T
    - <<<s: circular left shift by s bits.
  – Followed by a word level circular right shift of one word.

Logic of Each Step



Logic of Each Step

- Within a round, each of the 16 words of X[i] is used exactly
  - First round, X[i] are used in the order of I
  - Round 2, in the order of ρ2(i), where ρ2(i)=(1+5i)mod 16;
  - Round 3, in the order or ρ3(i), where ρ3(i)=(5+3i)mod 16;
  - Round 4, in the order or ρ4(i), where ρ2(i)=7imod 16;
- Each word of T[i] is used exactly once.

Security of MD5

- A recently discovered method can find a collision in a few hours
  - A few collisions were published on 8/17/04
  - Exact method has not been published yet
  - Can find many collisions for two 1024-bit messages
- Birthday attack
  - 2^{64}