Cryptographic Hash Functions

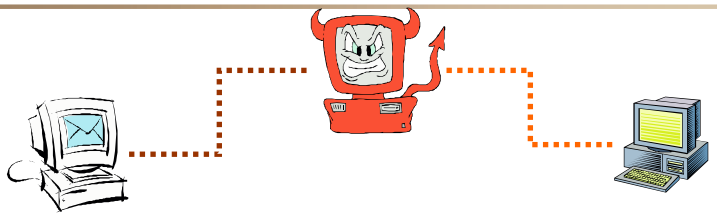
Murat Kantarcioglu

Based on Prof. Ninghui Li's Slides

Lecture Outline

- Hash functions
- Security properties
- Iterative Hash Functions
- Merkle-Damgard construction
- SHA1

Data Integrity and Source Authentication



- Encryption does not protect data from modification by another party.
- Need a way to ensure that data arrives at destination in its original form as sent by the sender and it is coming from an authenticated source.

Cryptographic Hash Functions

- A hash function maps a message of an arbitrary length to a m-bit output
  - output known as the fingerprint or the message digest
  - if the message digest is transmitted securely, then changes to the message can be detected
- A hash is a many-to-one function, so collisions can happen.

Requirements for Cryptographic Hash Functions

Given a function h:X $\rightarrow$Y, then we say that h is:

- preimage resistant (one-way):
  if given y $\in$ Y it is computationally infeasible to find a value x $\in$ X s.t. h(x) = y
- 2-nd preimage resistant (weak collision resistant):
  if given x $\in$ X it is computationally infeasible to find a value x' $\in$ X, s.t. x'$\neq$x and h(x') = h(x)
- collision resistant (strong collision resistant):
  if it is computationally infeasible to find two distinct values x',x $\in$ X, s.t. h(x') = h(x)

Uses of hash functions

- Message authentication
- Software integrity
- One-time Passwords
- Digital signature
- Timestamping
- Certificate revocation management

Brute-force Attacks on Hash Functions

- Attacking one-wayness
  - Goal: given $h:X \rightarrow Y$, $y \in Y$, find x such that h(x)=y
  - Algorithm: pick a random set X_0 of q values in X,
    for each $x \in X_0$, return x if h(x)=y,
    after all q values have been evaluated, return fail
  - A Las Vegas randomized algorithm
  - When h is a random instance of all functions mapping X to Y, the average-case success probability is

$$\varepsilon = 1 - \left(1 - 1/|Y|\right)^q \approx \frac{q}{|Y|}$$

  - Let $|Y|=2^m$, to get ε to be close to 0.5, $q \approx 2^{m-1}$

Las Vegas Randomized Algorithms

- An Las Vegas randomized algorithm may fail to give an answer, but when it does give an answer, the answer is always correct
- Such an algorithm has worst-case success probability ε if the algorithm returns a correct answer with probability at least ε
- Such an algorithm has average-case success probability ε if the probability that the algorithm returns a correct answer, averaged over all problem instances, is at least ε

Brute Force Attacks on Hash Functions

- Attacking collision resistance
  - Goal: given h, find x, x' such that h(x)=h(x')
  - Algorithm: pick a random set X_0 of q values in X
    for each $x \in X_0$, computes $y_x=h(x)$
    if $y_x=y_{x'}$ for some $x' \neq x$ then return (x,x') else fail
  - The average success probability is

$$1-e^{-\frac{q(q-1)}{2|Y|}}$$

  - Let $|Y|=2^m$, to get ε to be close to 0.5, $q \approx 2^{m/2}$
  - This is the birthday attack.

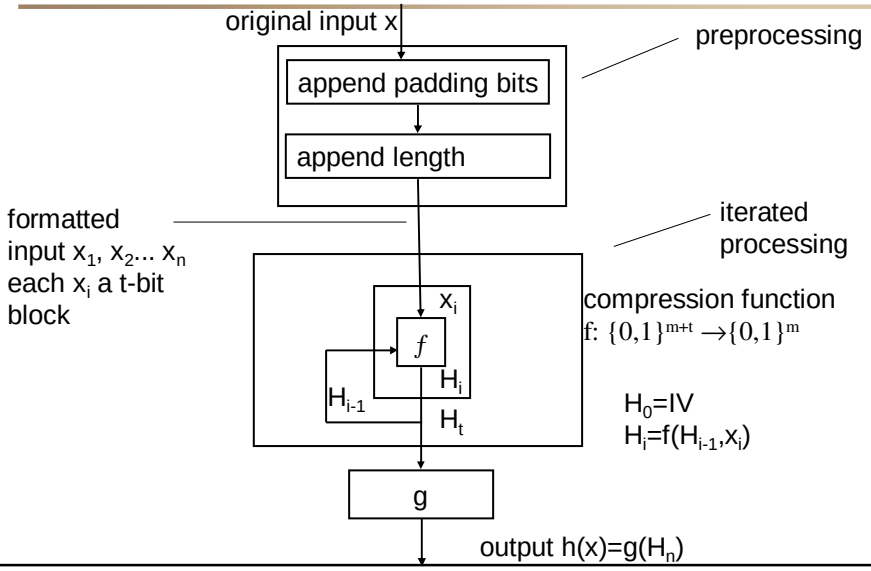
Choosing the length of Hash outputs

- Because of the birthday attack, the length of hash outputs in general should double the key length of block ciphers
  - SHA-256, SHA-384, SHA-512 to match the new key lengths (128,192,256) in AES

Constructing Hash Function From Compression Functions

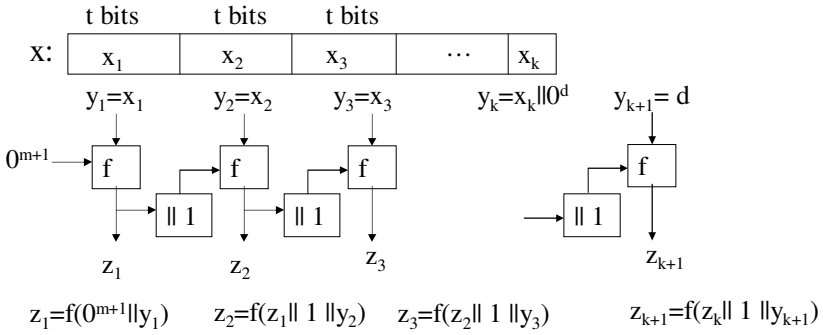
- Goal: A hash function h that maps a message of an arbitrary length to a m-bit output
  - h: $\{0,1\}^* \rightarrow \{0,1\}^m$
- Input: a compression function that takes a fixed-length input string and output a shorter string
  - f:$\{0,1\}^{m+t} \rightarrow \{0,1\}^m$
- The following properties can be defined for compression functions similar to hash functions
  - preimage resistance (one-way):
  - 2-nd preimage resistance (weak collision resistance):
  - collision resistance (strong collision resistance):

Model for Iterated Hash Functions



The Merkle-Damgard Construction of Hash Functions

- Goal: construct a hash function h: $\{0,1\}^* \rightarrow \{0,1\}^m$ from a compression function f: $\{0,1\}^{m+t+1} \rightarrow \{0,1\}^m$
- Given message x of arbitrary length



Example:

- Compression function: f: $\{0,1\}^{128+512+1} \rightarrow \{0,1\}^{128}$
- Message x has 1000 bits:
  - y_1 is first 512 bits of x
  - y_2 is last 488 bits of x || 0^{24}
  - y_3 is 0^{480} || 32-bit binary representation of 24
  - $z_1 = f(0^{129} \| y_1)$ z_1 has 128 bits
  - $z_2 = f(z_1 \| 1 \| y_2)$
  - $z_3 = f(z_2 \| 1 \| y_3)$ z_3 is the message digest h(x)

Example:

- Suppose that message x' has 488 bits and h(x)=h(x'):
  - y_1' is x' || 0^{24}
  - y_2' is 0^{480} || 32-bit binary representation of 24
  - z_1' = f(0^{129} || y_1') z_1 has 128 bits
  - z_2' = f(z_1' || 1 || y_2') z_2' is h(x)
- Then f(z_1' || 1 || y_2') = f(z_2 || 1 || y_3) and y_3=y_2'
  - if z_1' ≠z_2 then a collision is found for f
  - if z_1' =z_2 then f(0^{129} || y_1') =f(z_1 || 1 || y_2), there is also a collision for f

Security of the Merkle-Damgard Construction

- If f: $\{0,1\}^{m+t+1} \rightarrow \{0,1\}^m$ is collision resistant, then the Merkle-Damgard construction h: $\{0,1\}^* \rightarrow \{0,1\}^m$ is collision resistant.
- Proof:
  - suppose that we can find x≠x' such that h(x)=h(x'), we show that we can find collision on f
  - let y(x) = y_1 || y_2 || … || y_{k+1}
  - let z_1, z_2, …, z_{k+1} be the intermediate results of h(x), then h(x) = z_{k+1}=f(z_k || 1 || y_{k+1})
  - let y(x') = y_1' || y_2' || … || y_{n+1}' and z_1', z_2', …, z_{n+1}' be the intermediate results of h(x'), then
    f(z_k || 1 || y_{k+1}) = h(x') = z_{n+1}' = f(z_n' || 1 || y_{n+1}')

Security of the Merkle-Damgard Construction (Proof continued)

$f(z_k \| 1 \| y_{k+1}) = f(z_n' \| 1 \| y_{n+1}')$

- Case 1: $|x| \neq |x'| \bmod t$ (the number of padding bits are different), then $y_{k+1} \neq y_{n+1}'$, a collision has been found
- Case 2a: $|x| = |x'|$, then k=n, either $z_k \neq z_k'$, in which case a collision has been found, or $z_k = z_k'$, in which case
  $f(z_{k-1} \| 1 \| y_k) = z_k = z_k' = f(z_{k-1}' \| 1 \| y_k')$
  if $y_k \neq y_k'$, then a collision has been found; otherwise consider z_{k-1} and z_{k-1}', if they are different, a collision has been found, otherwise go backwards. There must exists an number j such that $y_j \neq y_j'$.

Security of the Merkle-Damgard Construction (Proof continued)

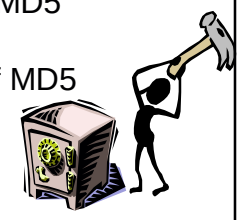
- Case 2b: $|x| \neq |x'|$. Similar to case (2a), except that we may go all the way back to the beginning of one of the strings and have
  $f(0^{m+1} \| y_1) = f(z_j' \| 1 \| y_{j+1}')$
  A collision has been found.

MD2, MD4 and MD5

- Family of cryptographic hash functions designed by Ron Rivest
- MD2: produces a 128-bit hash value, perceived as slower and less secure than MD4 and MD5
- MD4: produces a 128-bit hash of the message, using bit operations on 32-bit operands for fast implementation, specified as Internet standard RFC1320
- MD5: produces a 128-bit output, specified as Internet standard in RFC1321; till relatively recently was widely used.

MD5 Cryptanalysis

- Known attacks:
  - Berson (1992): for a single-round MD5, he used differential cryptanalysis to find two messages producing the same hash. Attack does not work for 4-round MD5.
  - Boer & Bosselaers(1993): found a pseudo collision (same message, two different IV's)
  - Dobbertin (1996) created collisions on MD5 compression function with a chosen IV
  - Wang, Feng, Lai, Yu found collisions of MD5
    - works on any IV
    - easy to find multiple collisions



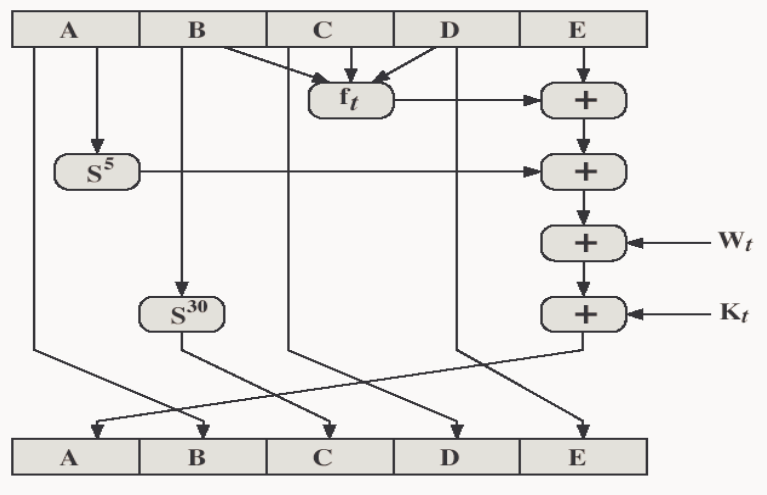
SHA1 (Secure Hash Algorithm)

- SHA was designed by NIST and is the US federal standard for hash functions, specified in FIPS-180 (1993).
- SHA-1, revised version of SHA, specified in FIPS-180-1 (1995) use with Secure Hash Algorithm).
- It produces 160-bit hash values.
- NIST have issued a revision FIPS 180-2 that adds 3 additional hash algorithms: SHA-256, SHA-384, SHA-512, designed for compatibility with increased security provided by AES.

SHA1 Overview

- As in MD5 message is padded such as its length is a multiple of 512 bits
- Initialize a 5-word (160-bit) buffer
  - Word A: 67 45 23 01
  - Word B: EF CD AB 89
  - Word C: 98 BA DC FE
  - Word D: 10 32 54 76
  - Word E: C3 D2 E1 F0
- Message is processed in 16-word (512-bit) chunks:
  - expand 16 words into 80 words by mixing & shifting
  - use 4 rounds of 20 operations on message block and buffer

SHA-1 Compression Function (Single Step)



SHA-1 Compression Function

- Each round consists of 20 steps, updates the buffer as follows:

  (A,B,C,D,E) <-
  (E+f(t,B,C,D)+(A<<5)+W_t+K_t),A,(B<<30),C,D)
- t is the step number
- f(t,B,C,D) is a non-linear function for round
- W_t is derived from the message block
  - $W_t=S^1(W_{t-16} \oplus W_{t-14} \oplus W_{t-8} \oplus W_{t-3})$
- K_t is a constant value derived from the sin function
- S^k is circular left shift by k bits

SHA-1 Cryptanalysis

- SHA1 shuffles and mixes them using rotates & XOR's to form a more complex input that makes finding collisions more difficult.
- Brute force attack is harder (160 vs 128 bits for MD5)
- Various attacks against simplified versions of SHA-1
- SHA-1 is still secure as today, but it may fall soon

More Precise Definitions

- A hash function h is (t,ε) one-way if there exists no t-time probabilistic algorithm A where $h(A(y)) = y$ with probability $> \varepsilon$
  - probability taken over random y and internal random
- A hash function h is (t,ε) weak collision resistant if there exists no t-time probabilistic algorithm A such that when given x, with probability $> \varepsilon$, it outputs x' such that $x' \neq x$ and $h(x')=h(x)$

More Precise Definitions

- A hash function h is (t,ε) collision resistant if there exists no t-time probabilistic algorithm that outputs two messages x_1 and x_2 such that $h(x_1)=h(x_2)$ with probability > ε

Reduction among the security properties

- (*t*+*c*, ε) collision resistant implies (*t*,ε) weak-collision resistant, where c is a small constant
- Proof idea:
  - Suppose that h is not (*t*,ε) weak-collision resistant, then there exists algorithm A, when given x, outputs x'=A(x) such that h(x')=y.
  - Construct B as follows, B picks a random x, feeds it to A, and then outputs (A(x),x).

Reduction among the security properties

- Collision-resistant implies one-way
  - main idea: given an algorithm A that (t,ε) breaks one-wayness, construct algorithm B, which picks a random x and gives h(x) to A, then A outputs x'. If x'≠x, then a collision is found.
  - Overall, the probability that B succeeds is close to ε assuming that the domain of h is significantly larger than the range of h
- Similarly, weak collision-resistant implies one-way

Summary

- Hash functions produce a fixed-length digest of any message
- Hash functions requirements are being one-way, weak-collision resistant and strong collision resistant
- Brute force attacks, finds a collision in $O(2^{m/2})$