Chương 6

CHỮ KÝ ĐIỆN TỬ (CHỮ KÝ SỐ) - HÀM BĂM

Giáo viên: Lê Quốc Anh

Nội dung

1. Khái niệm về chữ ký số (Digital Signature)
2. Chữ ký điện tử RSA
3. Chuẩn chữ ký điện tử DSS
4. Hàm băm
5. Ứng dụng của hàm băm
6. Kiến trúc hàm băm
7. Hàm băm MD5 và SHA1

Khái niệm về chữ ký số (Digital Signature)

- Khái niệm về Digital Signature được để xuất bởi Diffie & Hellman (1976)
- "Chữ ký điện tử (còn gọi là chữ ký số) là thông tin được mã hoá bằng Khoá riêng của người gửi, được gửi kèm theo văn bản nhằm đảm bảo cho người nhận định danh, xác thực đúng nguồn gốc và tính toàn vẹn của tài liệu nhận.
- Chữ ký điện tử thể hiện văn bản gửi đi là đã được ký bởi chính người sở hữu một Khoá riêng tương ứng với một Chứng chỉ điện tử nào đó."
- Chữ ký điện tử và chữ ký tay đều có chung đặc điểm là rất khó có thể tìm được hai người có cùng một chữ ký. Chữ ký điện tử được người ký tạo ra bằng Khoá riêng và phần có đặc tính duy nhất của văn bản được ký."

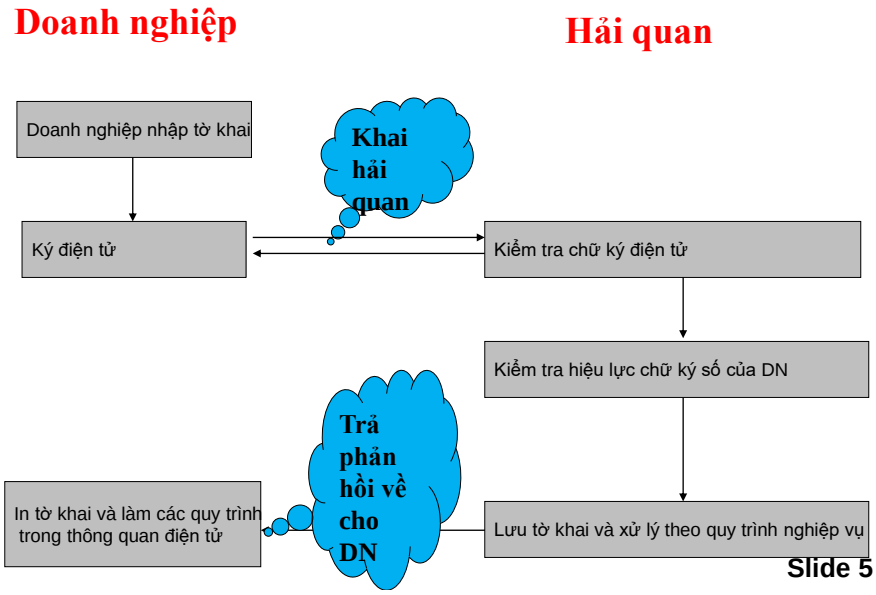
Slide 3

Khái niệm về chữ ký số (Digital Signature)

- Chữ ký điện tử là một trong ứng dụng quan trọng nhất của mã hóa khóa công khai.
- Message Authentication chỉ bảo vệ thông điệp trao đổi giữa hai bên tham gia không bị hiệu chỉnh hay giả mạo từ bên thứ 3, nhưng nó không bảo vệ thông điệp bị hiệu chỉnh hay giả mạo từ một trong 2 bên tham gia, nghĩa là:
  - Bên nhận giả mạo thông điệp của bên gửi
  - Bên gửi chối là đã gửi thông điệp đến bên nhận
- Chữ ký điện tử không những giúp xác thực thông điệp mà còn bảo vệ mỗi bên khỏi bên kia

Slide 4

VÍ DỤ CHỮ KÝ SỐ TRONG THỦ TỤC HQĐT



Slide 5

Khái niệm về chữ ký số (Digital Signature)

Ví dụ cơ bản:

- Mike có hai khóa, một khóa công khai và một khóa riêng.
- Mike đưa khóa công khai của mình cho Amanda, nhưng giữ lại khóa riêng cho mình.
- Khi muốn chuyển tài liệu cho Amanda, Mike có thể xác nhận (ký) các tài liệu này dùng chính khóa riêng của mình và gửi chúng đến Amanda.
- Amanda sau đó sẽ dùng khóa công khai của Mike, để có thể kiểm tra tài liệu mà cô ấy nhận được, thực sự được gửi bởi Mike.

Slide 6

Mục tiêu của chữ ký điện tử

Mục tiêu an toàn

- Xác thực (Authentication)
- Chống phủ nhận (Non-repudiation)

Slide 7

Đặc điểm chữ ký số

1. Đảm bảo tính xác thực

- Chứng minh tính hợp pháp của người gửi
- Chứng minh tính toàn vẹn của dữ liệu

2. Chữ ký số là hàm của các tham số

- Thông báo giao dịch (văn bản gốc)
- Thông tin bí mật của người gửi (Khóa riêng của sender)
- Thông tin công khai trên mạng (Khóa công khai)
- Mã xác thực : Đảm bảo tính toàn vẹn của thông điệp

Slide 8

Đặc điểm chữ ký số

Khóa bí mật	Tính bí mật của Khóa bí mật	Chỉ có người chủ mới biết
Khóa công khai	Tính sẵn sàng truy cập của khóa công khai	Có thể truy cập thông qua phương tiện thông dụng vào bất cứ thời điểm: Chứa trong một thư mục công cộng Đảm bảo tính chính xác và không giả mạo
Chứng thư số	Công bố khóa công khai	Được cấp phát bởi tổ chức có thẩm quyền
Độ dài khóa	Tương ứng tính an toàn của khóa	Khóa có thể có độ dài (thông dụng là) 512, 1024, 2048, 4096 Khóa càng dài mã càng chậm
Tính pháp lý	Với công nghệ đảm bảo sẽ tương đương chữ ký tay	Được cấp phát theo quy trình an toàn với các thông số kỹ thuật đảm bảo Được lưu trữ an toàn
Tính khả dụng	Ngày càng dễ sử dụng	Được lưu trong các thiết bị cá nhân như USB-token, smart card Ngày càng nhiều ứng dụng hỗ trợ

Slide 9

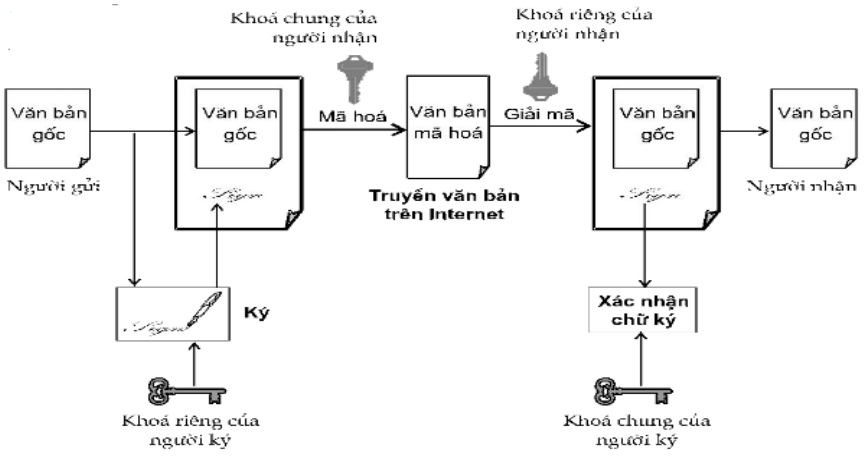
Nguyên lý ký điện tử trong hệ mật mã công khai

1. Người gửi (chủ nhân văn bản): ký văn bản bằng cách mã hóa nó với khóa bí mật của mình, rồi gửi cho bên nhận.

2. Người nhận tiến hành kiểm tra chữ ký bằng cách sử dụng khóa công khai của người gởi để giải mã văn bản. Nếu giải mã thành công thì văn bản ký là đúng người gửi

Slide 10

Sơ đồ sử dụng chữ ký điện tử



Slide 11

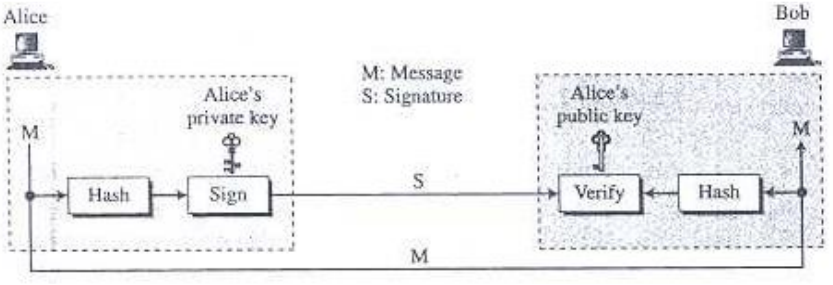
Hàm băm mật mã

- Dùng để chiết xuất đặc trưng của văn bản, đầu ra là một dãy số xác định gọi là mã băm.
- Rất "nhạy" đối với các thay đổi trong văn bản.
- Có tính kháng xung đột, tính một chiều và tốc độ nhanh.

Slide 12

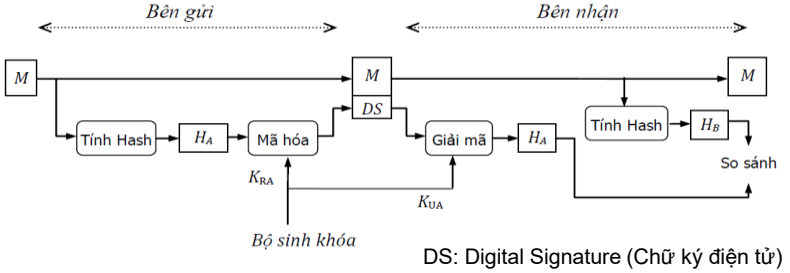
Hàm băm mật mã - Signing the Digest

- Ký trên digest của thông đệp sẽ ngắn hơn ký trên thông điệp
- Người gửi có thể ký trên cốt thông điệp và người nhận có thể kiểm tra trên cốt thông điệp



Slide 13

Digital Signature Process
Tạo và Kiểm tra chữ ký số



DS: Digital Signature (Chữ ký điện tử)

- Trong mô hình này, Alice sau khi tính giá trị hash *HA* cho thông điệp *M* thì sẽ mã hóa *HA* bằng khóa riêng của Alice để tạo thành chữ ký điện tử *DS*. Alice gửi kèm *DS* theo *M* cho Bob.
- Bob dùng khóa công khai của Alice để giải mã chữ ký điện tử *DS* và có được giá trị hash *HA* của Alice.
- Vì Trudy không có *K*RA nên không thể sửa được HA.
- Ngoài ra, vì Alice là người duy nhất có *KRA*, nên chỉ có Alice mới có thể tạo *DS* từ *M*. Do đó Alice không thể từ chối là đã gửi bản tin.

Slide 14

Nhận xét

- Chữ ký không phải là nét vẽ ngoằn ngoèo khó bắt chước mà là một dãy số trích từ đặc trưng văn bản đã được mã hóa.
- So với chữ ký thông thường, chữ ký số có ưu thế vượt trội hơn chữ ký tay.
  - Chính xác tuyệt đối
  - Kiểm định dễ dàng và chính xác

"Chữ ký điện tử mở đường cho các dịch vụ có độ tin cậy cao"

Slide 15

Nhược điểm mô hình chữ ký số

- Mô hình CKS ở trên chỉ đạt được nếu như mỗi người sở hữu đúng cặp chìa khóa của chính mình.
- Có thể xẩy ra hiện tượng *"mạo danh"* người gửi. Do đó, ta cần có cơ chế để xác định *"ai là ai"* trên toàn hệ thống.
- Giải pháp: chứng minh thư số

Slide 16

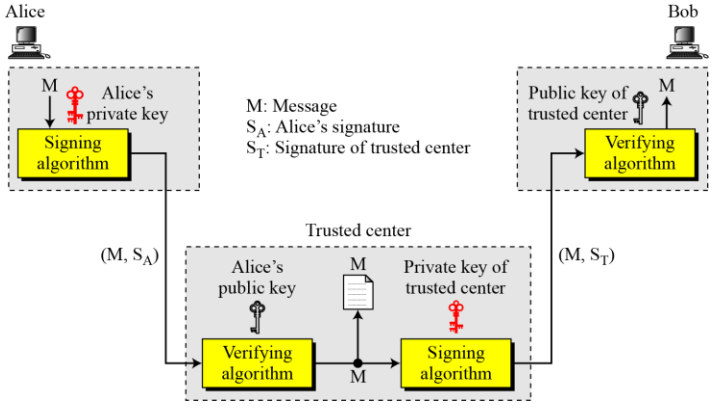
Ứng dụng của chữ ký số

1. Xác thực thông điệp (Message Authentication): Người ký được xác nhận là chủ chữ ký.
2. Toàn vẹn thông điệp (Message Integrity): Nội dung chưa bị thay đổi hoặc xáo trộn kể từ khi nó được ký điện tử.
3. Chống từ chối (Non-repudiation): Chứng minh với tất cả các bên về nguồn gốc của nội dung đã ký. Từ "thoái thác" dùng để chỉ hành động của một người ký từ chối bất kỳ mối liên kết nào với nội dung đã ký.
4. Bảo mật (Confidentiality)

Slide 17

Chống từ chối (Non-repudiation)

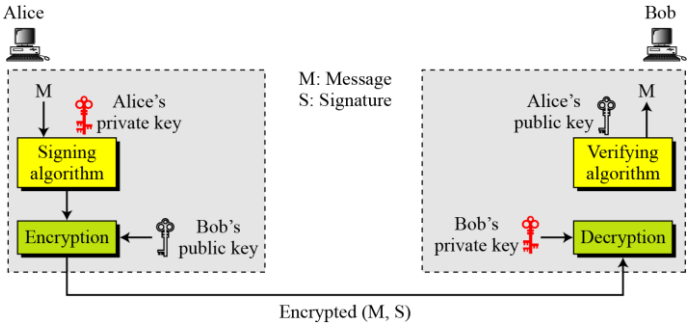
- Nonrepudiation có thể được cung cấp bằng cách dùng một trusted Center



Slide 18

Bảo mật (Confidentiality)

- Thêm confidentiality vào cơ chế digital signature



- Một Digital Signature không cung cấp tính bí mật. Nếu cần bảo mật, encryption/decryption được áp dụng

Slide 19

Public Notary (công chứng)

- Trong trường hợp bên Alice cố cãi rằng cô ta chẳng may làm thất lạc hay vô tình đánh lộ zA và bị một kẻ thứ ba lợi dụng chứ không có ý định tạo ra văn bản có chữ ký như thế→ khi đó có thể thêm trọng tài vào hệ thống.
- Người trọng tài này cũng còn gọi là công chứng viên (public notary) sẽ ký đè lên chữ ký của Alice để chứng thực, Alice không thể nào chối cãi.

Slide 20

Proof of delivery (xác nhận giao hàng - hoá đơn)

- Ngược lại, bên gửi cũng cần được bảo vệ để chống lại hiện tượng người nhận có nhận được thông báo nhưng chối là chưa nhận được.
- Điều này có thể thực hiện được qua những giao thức có phân xử (adjudicated protocol), tức là những giao thức mà sau đó cho phép người thứ ba có thể kiểm định lại được.

Slide 21

Tấn công trên Digital Signature

- Các loại tấn công trên Digital Signature
  (Attack Types)

- Giả mạo chữ ký (Forgery)

Slide 22

Attack Types

- Key-Only Attack
- Known-Message Attack
- Chosen-Message Attack

Slide 23

Forgery Types

- Existential Forgery
- Selective Forgery

Slide 24

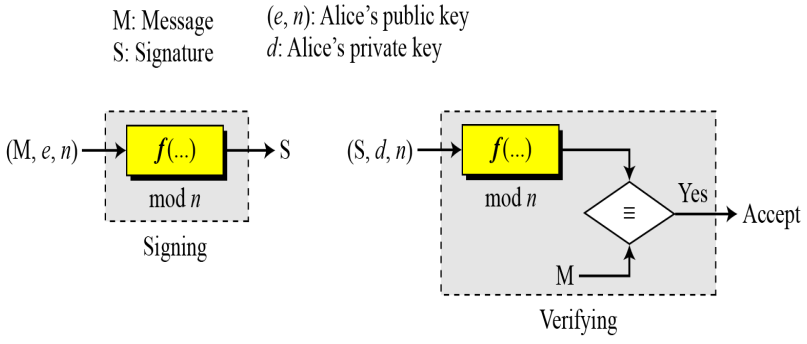
2. Chữ ký điện tử RSA

- Ý tưởng mật mã RSA được dùng cho việc ký và kiểm tra chữ ký, nó được gọi là cơ chế chữ ký số RSA
- Người gửi dùng private key của chính mình để ký vào tài liệu; người nhận dùng public key của người gửi để kiểm tra chữ ký
- So với chữ ký truyền thống, private key đóng vai trò là chữ ký của chính người gửi, public key của người gửi đóng vài trò là bản sao của chữ ký mà có thể được công khai

Slide 25

RSA Digital Signature Scheme

- Ý tưởng tổng quát của cơ chế chữ ký RSA



Slide 26

RSA Digital Signature Scheme

Phát sinh khóa (Key Generation)

- Phát sinh khóa trong cơ chế chữ ký RSA là hoàn toàn giống như phát sinh khóa RSA
- Trong đó **d** là bí mật, **e** và **n** là công khai

RSA Digital Signature Scheme

1) Tạo cặp khóa (***bí mật, công khai***) (**a**, **b**):

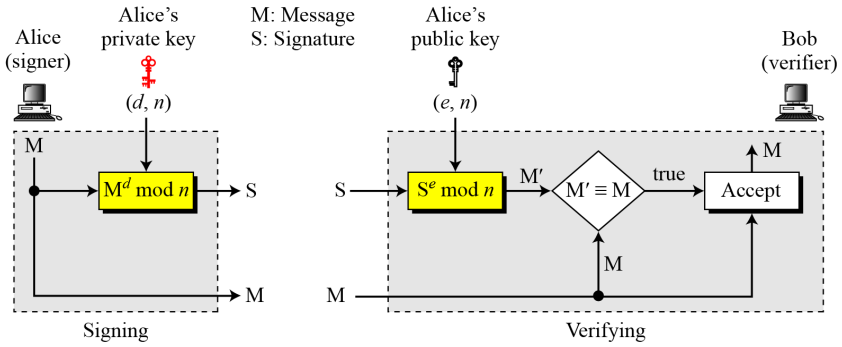
- Chọn 2 số nguyên tố **p**, **q**, xác định **n** = **p** * **q**, n là công khai, đặt P = A = Zn và định nghĩa:
- Tính $\phi(\mathbf{n}) = (\mathbf{p-1}).(\mathbf{q-1})$.
- Chọn khóa công khai **e** < $\phi(\mathbf{n})$, nguyên tố cùng nhau với $\phi(\mathbf{n})$.
- Khóa bí mật **d** là phần tử nghịch đảo của e theo mod ϕ(n): $\mathbf{e}*d \equiv \mathbf{1} \pmod{\phi(n)}$.
- Tập cặp khóa (bí mật, công khai) **K** = {(**e**, d)/ e, d ∈ Zn , $e*\mathbf{d} \equiv 1 \pmod{\phi(n)}$}.

2) ***Ký số***: Chữ ký trên $\mathbf{M} \in \mathbf{P}$ là $\mathbf{S} = \mathbf{sig_k}(\mathbf{M}) = \mathbf{M^d} \pmod{\mathbf{n}}$, $S \in \mathbf{A}$

3) .***Kiểm tra chữ ký***: $\mathbf{ver_k}(\mathbf{M}, S)$ = TRUE $\Leftrightarrow \mathbf{M} \equiv \mathbf{S^e} \pmod{\mathbf{n}}$ với $M,S \in Z_n$.

RSA Digital Signature Scheme

Tạo và Thẩm tra chữ ký



Slide 29

RSA Digital Signature Scheme

Ví dụ:

Phát sinh khóa:

- Alice chọn p = 823 và q = 953, và tính n = 784319.
- Giá trị ϕ(n) là 782544
- Alice chọn e = 313 và tính d = 160009

Alice muốn ký và gửi thông điệp M=19070 cho Bob

- Tạo chữ ký:

  $S=(19070^{160009}) \bmod 784319 = 210625 \bmod 784319$

- Gửi (M, S) cho Bob

- Bob nhận (M, S) và tính M'

  $M'=210625^{313} \bmod 784319 = 19070 \bmod 784319$

- → $M \equiv M' \bmod n$

Slide 30

EXAMPLE

Chữ ký trên m = 2

**Tạo cặp khóa (bí mật, công khai) :*

Chọn 2 số nguyên tố: **p**=3, **q**=5 → **n** = **p***q** = 3*5 = 15(công khai)

Đặt **P** = **A** = $\mathbf{Z_n}$. Tính $\phi(\mathbf{n}) = (\mathbf{p-1}).(\mathbf{q-1}) = 2 * 4 = 8$.

Chọn khóa công khai **e** = **3** < $\phi(\mathbf{n})$, nguyên tố với $\phi(\mathbf{n}) = \mathbf{8}$.

=>Tìm d: là phần tử nghịch đảo của **e** theo mod ϕ(n): e*$\mathbf{d} \equiv \mathbf{1} \pmod{\phi(n)}$.

Vậy khóa bí mật **d = 3**

Slide 31

EXAMPLE

* ***Ký số***: Chữ ký trên **M = 2**$\in$**P** là:

$$\mathbf{S} = \mathbf{sig_k}(M) = M^{\mathbf{d}} \pmod{\mathbf{n}}$$
$$= \mathbf{2^3} \pmod{\mathbf{15}}$$
$$= \mathbf{8}, \mathbf{S} \in \mathbf{A}.$$

* ***Kiểm tra chữ ký***:

$$\mathbf{ver_k}(\mathbf{M}, S) = \text{TRUE} \Leftrightarrow \mathbf{M} \equiv \mathbf{S^e} \pmod{\mathbf{n}}$$
$$\Leftrightarrow \mathbf{2} \equiv \mathbf{8^3} \pmod{\mathbf{15}}.$$

Slide 32

BÀI TẬP

Sử dụng cơ chế chữ ký RSA cho các bài toán sau:

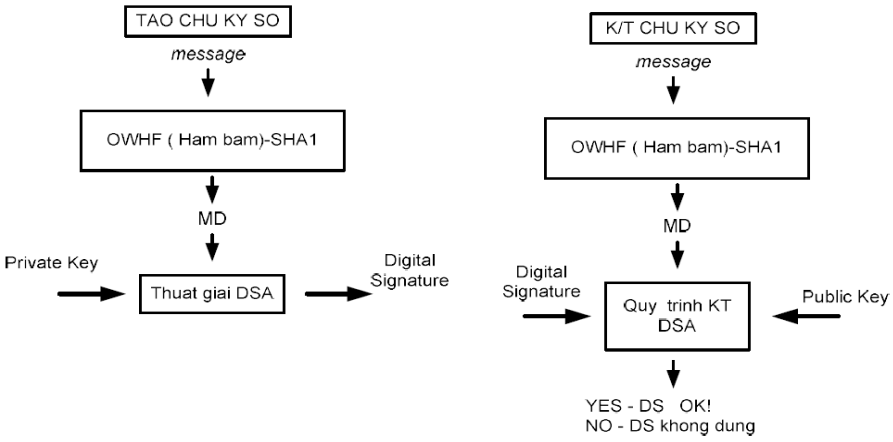
1) Cho p=5, q=11, e=7. Tạo và kiểm tra chữ ký điện tử với bản tin M=5

2) Cho p=7, q=11, e=17. Tạo và kiểm tra chữ ký điện tử với bản tin M=3

3. Chuẩn chữ ký điện tử DSS

- Thuật toán CKĐT DSA là thuật toán được đề nghị trong chuẩn chữ ký điện tử DSS (Digital Signature Standard) của NIST.
- Nó cung cấp một trình tự để tạo và xác nhận CKĐT
- DSA sử dụng 1 cặp khóa công khai – khóa riêng. Trong cả giai đoạn ký và xác nhận văn bản được ký thể hiện dưới dạng thông điệp rút gọn là kết quả của việc áp dụng hàm băm SHA-1 lên văn bản cần ký.

Digital Signature Process
Tạo và Kiểm tra chữ ký số



a. Tao chu ky so

b. Kiem tra chu ky so

Thuật giải DSA – Chuẩn chữ ký số

1,Tạo khoá

- Chọn p là số nguyên tố L bit, sao cho 2L-1<p<2L , 512 ≤ L ≤ 1024, L chia hết cho 64.
- Chọn q là một số nguyên tố 160 bít và là ước số của p-1 với 2159<q<2160
- Chọn h, với 1 < h < p - 1 sao cho g = h^z mod p > 1. (z = (p-1) / q.)
- Chọn x ngẫu nhiên, thoả mãn 0 < x < q.
- Tính giá trị y = g^x mod p.
- Khoá công là (p, q, g, y). Khoá riêng là x.

Thuật giải DSA – Chuẩn chữ ký số

- **b. Tạo chữ kí số**
- Tạo 1 số ngẫu nhiên với mỗi thông điệp, giá trị k thỏa mãn $0 < k < q$
- Tính $r = (g^k \bmod p) \bmod q$
- Tính $s = (k^{-1}(\text{SHA-1}(m) + x*r)) \bmod q$, ở đây SHA-1($m$) là hàm băm mã hoá SHA-1 áp dụng cho thông điệp m
- Tính toán lại chữ ký trong trường hợp không chắc chắn khi r=0 hoặc s=0
- Dữ liệu được gửi đi là Văn bản M, chữ ký (r, s).

Slide 37

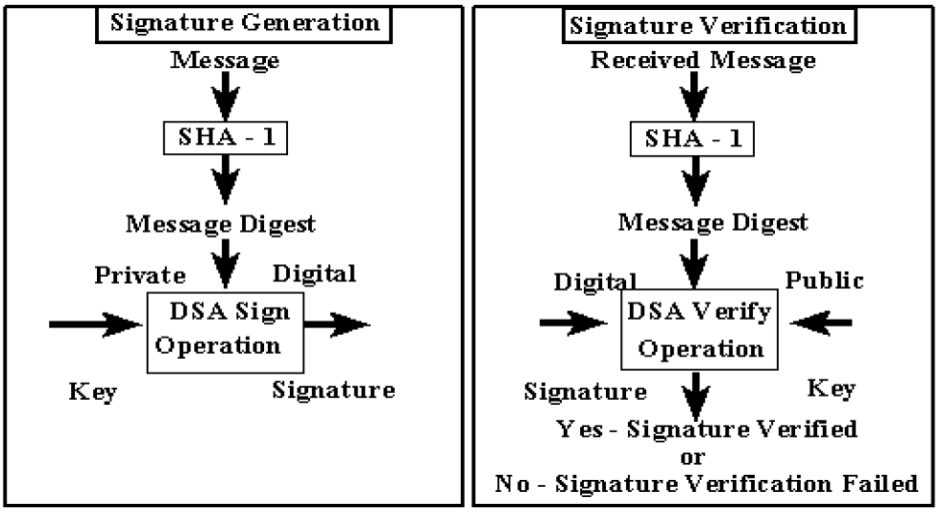
Thuật giải DSA – Chuẩn chữ ký số

3, Kiểm tra chữ ký

- Loại bỏ chữ ký nếu hoặc 0< r <q hoặc 0< s <q không thỏa mãn.
  - Tính $w = (s)^{-1} \bmod q$
  - Tính $u1 = (\text{SHA-1}(m)*w) \bmod q$
  - Tính $u2 = (r*w) \bmod q$
  - Tính $v = ((g^{u1}*y^{u2}) \bmod p) \bmod q$
- Chữ ký là có hiệu lực nếu $v = r$
- Nếu v != r' văn bản có thể đã được sửa đổi trên đường truyền hoặc khóa cá nhân mã hóa văn bản không khớp với khóa công khai mà người nhận đang giữ (người gửi mạo danh)

Slide 38

Thuật giải DSA – Chuẩn chữ ký số



Slide 39

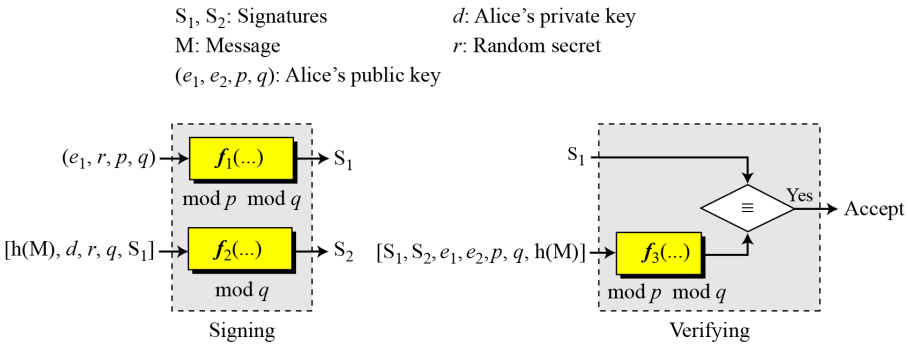
Digital Signature Standard (DSS)

- NIST đã công bố chuẩn xử lý thông tin liên ban FIPS 186, được biết như là Digital Signature Standard (DSS)
- DSS được đề xuất năm 1991 và hiểu chỉnh lại 1993, 1996 có một hiệu chỉnh nhỏ, năm 2000, một phiên bản mở rộng của chuẩn được phát hành như FIPS 186-2, 2009 cập nhật FIPS 186-3. Phiên bản cuối cùng hợp nhất các thuật tóan chữ ký số dựa trên mật mã RSA và đường cong Eliptic

Slide 40

Digital Signature Standard (DSS)

Ý tưởng tổng quát của chữ ký DSS



Slide 41

Digital Signature Standard (DSS)

1) **Tạo cặp khóa:** Cho p là 1 số nguyên tố 512 bit trong trường logarit rời rạc Z_p ; q là 1 số nguyên tố 160 bit và q chia hết cho (p-1). Cho $\alpha \in Z_p^*$; $P=Z_p^*$; $A=Z_p^* Z_q$:

$K=\{(p, q, \alpha, a, \beta): \beta \equiv \alpha^a \pmod p\}$

với p, q, α, β công khai, a là bí mật

Chọn 1 số ngẫu nhiên k(1<=k<=q-1)

2) Ký số: $sig_k(\gamma , \delta)$

trong đó: $\gamma=(a^k \bmod p) \bmod q$ & $\delta=(x+a*\gamma)k^{-1} \bmod q$

Với $x \in Z_p^*$ và $\gamma, \delta \in Z_q$

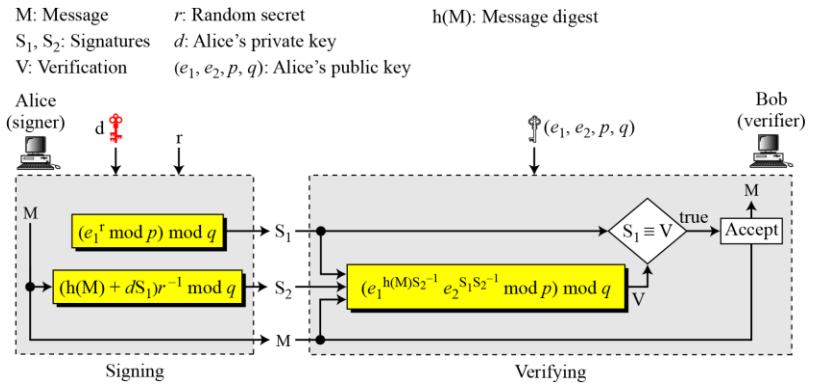
3) Kiểm tra chữ ký: Tính $e_1=x*\delta^{-1}$ và $e_2=\gamma*\delta^{-1}$

$Ver(x, \gamma, \delta)$= TRUE $\Leftrightarrow (\alpha^{e1}\beta^{e2} \bmod p) \bmod q = \gamma$

Slide 42

Digital Signature Standard (DSS)

Tạo và Thẩm tra chữ ký



Slide 43

Digital Signature Standard (DSS)

- Giả sử: q=101, p=78q + 1=7879
- 3 là phần tử nguyên tử trong Z_{7879} nên ta có thể lấy:

  $a = 3^{78} \bmod 7879 = 170$
- Giả sử a=75, khi đó
- $B=a^a \bmod 7879 = 4567$
- Bây giờ giả sử Bob muốn kí bức điện x = 1234 anh ta chọn số ngẫu nhiên k = 50, vì thế: $k^{-1} \bmod 101 = 99$
- Khi đó: $\gamma = (170^{50} \bmod 7879) \bmod 101$

  $= 2518 \bmod 101 = 94$
- Và $\delta = (1234 + 75 * 94) \bmod 101 = 97$

=> chữ ký: (94, 97)

Slide 44

Digital Signature Standard (DSS)

Chứng minh:

- $\delta^{-1} = 97^{-1} \bmod 101 = 25$
- $e_1 = 1234 * 25 \bmod 101 = 45$
- $e_2 = 94 * 25 \bmod 101 = 27$
- $(170^{45} * 4567^{27} \bmod 7879) \bmod 101 = 2518 \bmod 101 = 94$
- Vì thế chữ kí hợp lệ

Slide 45

Những biến thể chữ ký

- **Time Stamped Signatures**
  - Một tài liệu được ký cần được gắn nhãn thời gian (Timestamped) để ngăn chặn tài liệu bị phát lại (replay) bởi đối phương
  - Ví dụ: Alice ký một yêu cầu đối với ngân hàng của cô ta, Bob chuyển tiền cho Eve. Tài liệu yêu cầu này có thể bị chặn và phát lại bởi Eve nếu không có nhãn thời gian gắn trên tài liệu

Slide 46

Những biến thể chữ ký

Blind Signatures

- Giả sử có một tài liệu mà chúng ta muốn có chữ ký mà không muốn tiết lộ nội dung của tài liệu đối với người ký.
- Ví dụ: Nhà khoa học phát minh ra một lý thuyết rất quan trọng mà cần được ký bởi công chứng viên, công chứng viên sẽ ký nhưng sẽ không biết gì về nội dung của phát minh.

Slide 47

Những biến thể chữ ký

Blind Signatures

- Các bước thực hiện:
  - Bob tạo một thông điệp, ẩn (Blind) nó, và gửi thông điệp ẩn này cho Alice
  - Alice ký thông điệp ẩn và trả về chữ ký trên thông điệp ẩn.
  - Bob bỏ ẩn chữ ký để thu về chữ ký trên thông điệp gốc

Slide 48

4. Hàm băm (Hash Function)

Các định nghĩa cơ bản:

- Hàm f là hàm một chiều (one-way function) nếu:
  - Cho x dễ dàng tính được f(x)
  - Cho f(x) khó tìm được x
- Hàm f là hàm cửa lật một chiều (one-way trapdoor) nếu:
  - Cho x dễ dàng tính được f(x)
  - Cho f(x) khó tìm được x
  - Nếu có thêm thông tin (trapdoor" information) thì dễ dàng tính được x từ f(x)

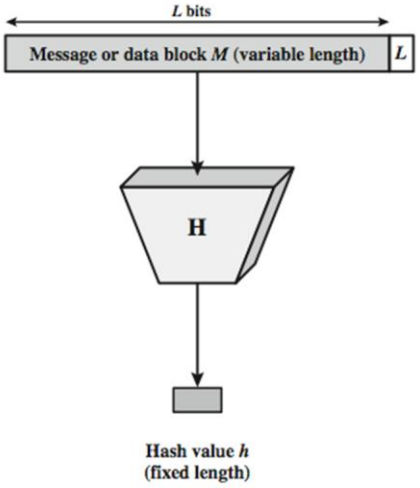
Slide 49

4. Hàm băm (Hash Function)

- **Hàm băm** là các thuật toán không sử dụng khóa để mã hóa, nó có nhiệm vụ băm thông điệp được đưa vào theo một thuật toán ***h* một chiều nào đó**, rồi đưa ra một **bản băm – văn bản đại diện – có kích thước cố định**. Do đó người nhận không biết được nội dung hay độ dài ban đầu của thông điệp đã được băm bằng hàm băm.
- Giá trị của hàm băm là duy nhất, và **không thể suy ngược** lại được nội dung thông điệp từ giá trị băm này.

Slide 50

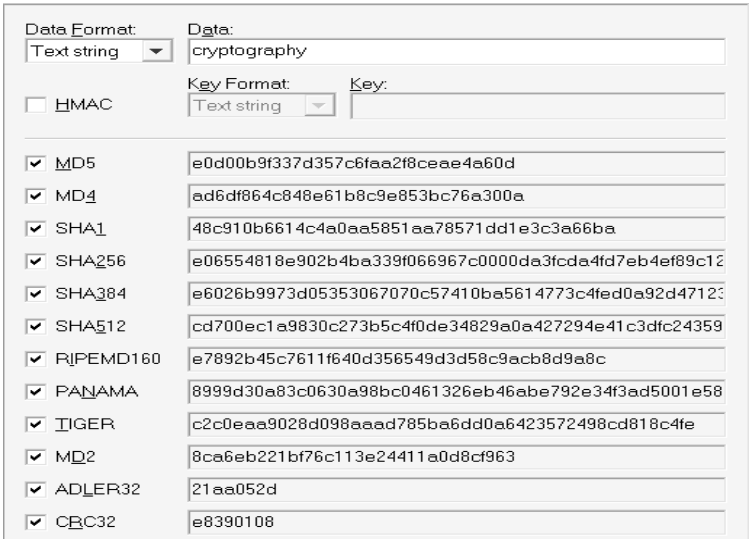
4. Hàm băm (Hash Function)



- Input: ***M có kích thước bất kỳ***
- Output – ***giá trị h có kích thước cố định, ngắn.***
- **H(x)** – ***hàm một chiều ("Khó để tính nghịch đảo")***

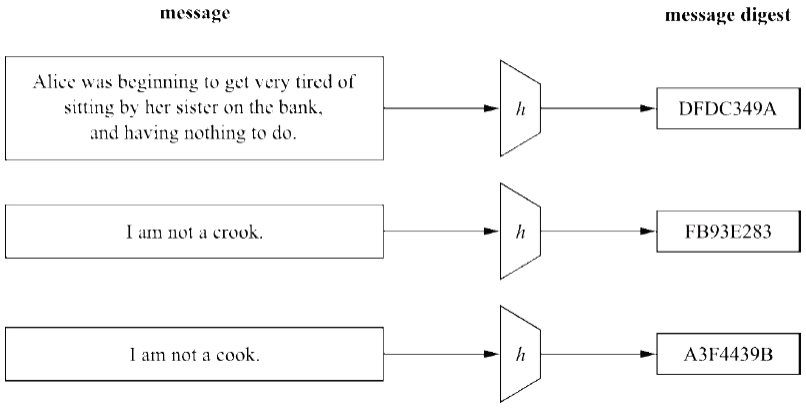
Slide 51

4. Hàm băm (Hash Function)



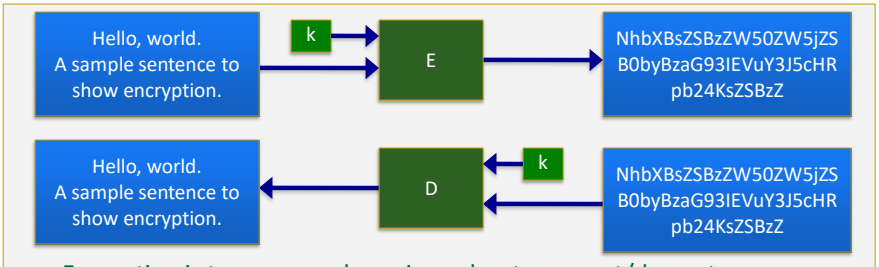
Slide 52

4. Hàm băm (Hash Function)

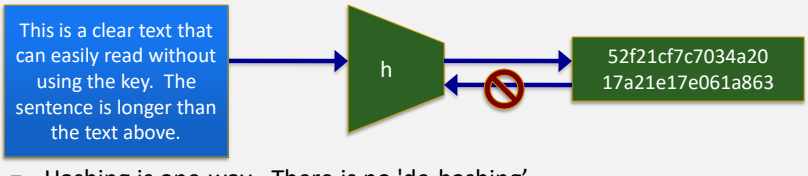


Slide 53

4. Hàm băm (Hash Function)



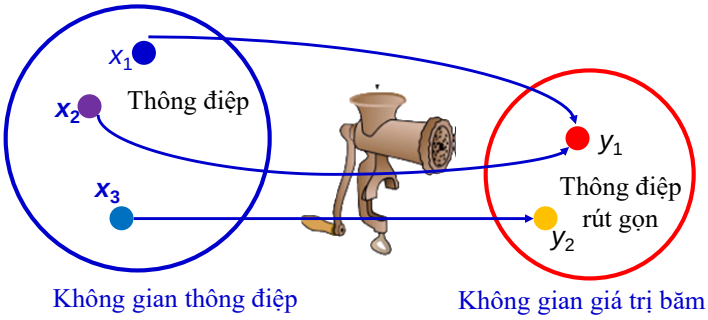
- Encryption is two way, and requires a key to encrypt/decrypt



- Hashing is one-way. There is no 'de-hashing'

Slide 54

4. Hàm băm (Hash Function)



→Không gian giá trị Băm nhỏ hơn rất nhiều so với Không gian thông điệp về mặt kích thước

→chắc chắn sẽ tồn tại đụng độ (trùng), nghĩa là **có hai tin *x* và *x"* mà giá trị Băm của chúng là giống nhau, tức là *h(x) = h(x")***

Slide 55

Tính chất hàm băm

1. Tính 1 chiều (Preimage resistant – one-way property):

Cho trước giá trị băm h việc tìm x sao cho H(x)=h là rất khó

2. Tính kháng đụng độ yếu (Second preimage resistant – weak collision resistance – Tính chống trùng yếu):

Cho thông điệp đầu vào x, việc tìm một thông điệp x' với (x'≠ x) sao cho h(x')=h(x) là rất khó

3. Tính kháng đụng độ mạnh-tính chống trùng mạnh (Strong Collision resistance):

Không thể tính toán để tìm được hai thông điệp đầu vào $x_1 \neq x_2$ sao cho chúng có cùng giá trị băm

(Nghịch lý ngày sinh – Birthday paradox)

Slide 56

Tính chất hàm băm

1. Tính 1 chiều (Preimage resistant – one-way property):

Cho trước giá trị băm h việc tìm x sao cho H(x)=h là rất khó

- Dạng tấn công thứ nhất là người C bắt đầu với một bức điện được ký có giá trị (x, y), trong đó y = sigK(h(x)) (cặp (x, y) có thể là bất kỳ bức điện trước đó mà B đã ký). Sau đó, C tính z = h(x) và cố gắng tìm x‟ # x để h(x‟) = h(x). Nếu C làm được điều này thì cặp (x‟, y) sẽ là một bức điện được ký có giá trị (một bức điện giả mạo có giá trị). Để ngăn cản việc này, hàm Băm h phải thoả mãn tính chất 1

Slide 57

Tính chất hàm băm

2. Tính kháng đụng độ yếu (Second preimage resistant – weak collision resistance – Tính chống trùng yếu):

Cho thông điệp đầu vào x, việc tìm một thông điệp x' với (x'≠ x) sao cho h(x')=h(x) là rất khó

- Một dạng tấn công khác mà người C có thể làm là: đầu tiên anh ta tìm 2 bức điện x ≠ x‟ sao cho h(x) = h(x‟). Sau đó C đưa bức điện x cho B và thuyết phục B ký vào cốt bức điện h(x); và vì vậy, anh ta tìm được y. Như vậy, cặp (x‟, y) là một cặp chữ ký giả có giá trị. Điều này là nguyên nhân mà việc thiết kế hàm Băm phải thoả mãn tính chất 2.

Slide 58

Tính chất hàm băm

3. Tính kháng đụng độ mạnh-tính chống trùng mạnh (Strong Collision resistance):

Không thể tính toán để tìm được hai thông điệp đầu vào $x_1 \neq x_2$ sao cho chúng có cùng giá trị băm

(Nghịch lý ngày sinh – Birthday paradox)

Dạng tấn công thứ 3 là chọn một giá trị cốt z ngẫu nhiên. Người C sẽ tính một chữ ký với một giá trị ngẫu nhiên z, sau đó anh ta tìm một bức điện x sao cho z = h(x). Nếu anh ta làm được điều này thì cặp (x, y) là cặp chữ ký giả có giá trị. Như vậy một tính chất nữa mà h cần thoả mãn là tính một chiều.

Slide 59

Nghịch lý ngày sinh (birthday paradox)

- Nếu hàm Băm có không gian Băm 64 -bit thì số lượng văn bản phải ít nhất 2^{64} (với một máy tính có thể thực hiện việc Băm 1 triệu bức điện trong 1 giây, thì phải mất 6000.000 năm tính toán).
- Nếu thám mã thử với lượng văn bản ít hơn nhiều, trong phạm vi có thể tính được thì xác suất để tìm được đụng độ sẽ như thế nào?
- Bản chất của hiện tượng này được minh hoạ rõ thông qua phát biểu sau: thường được gọi là nghịch lý ngày sinh.
  - Trong một nhóm có 23 người bất kỳ , xác suất để có hai người có cùng ngày sinh nhật ít nhất là ½.

Slide 60

Nghịch lý ngày sinh (birthday paradox)

- Một cách tổng quát, giả sử một hàm Băm có n giá trị Băm khác nhau, nếu chúng ta có k giá trị Băm từ k thông tin khác nhau được chọn ngẫu nhiên, thì xác suất để không xảy ra đụng độ là:

$$(1-\frac{1}{n})(1-\frac{2}{n}) \ldots (1-\frac{k-1}{n}) = \prod_{i=1}^{k-1}(1-\frac{i}{n}).$$

Với $\frac{i}{n} << 1$, thì $\prod_{i=1}^{k-1}(1-\frac{i}{n}) \approx \prod_{i=1}^{k-1} e^{\frac{-i}{n}} = e^{\frac{-k(k-1)}{2n}}$

- Do đó xác xuất để xảy ra đụng độ là: $1-e^{\frac{-k(k-1)}{2n}}$

Slide 61

Nghịch lý ngày sinh (birthday paradox)

- Gọi xác suất trên là ε : $1-e^{\frac{-k(k-1)}{2n}} \approx \varepsilon$
- Suy ra: $k^2 - k \approx 2n \ln(\frac{1}{1-\varepsilon})$
- Khi k lớn: $k \approx \sqrt{2n \ln(\frac{1}{1-\varepsilon})}$
- Với $\varepsilon = 0{,}5$ ta $có$: $k \approx 1{,}1774\sqrt{n}$
- Với k = 23 là số người , n = 365 là số ngày trong năm thì xác xuất tồn tại hai người có cùng sinh nhật sẽ là: $\varepsilon = 1 - 2{,}7^{-07} \approx 0{,}507$

Slide 62

Nghịch lý ngày sinh (birthday paradox)

- Công thức nghịch lý ngày sinh nhật cho phép chúng ta có thể xác định được giới hạn dưới.
- Một hàm băm 40 bít sẽ không an toàn vì chỉ cần thử 2^{20} phép tính (gần 1 tỷ phép tính) thì xác suất đụng độ là 50%.
- Với hàm băm 64 bít xác suất đụng độ là 2^{32}, với máy tính ngày nay chỉ cần mất 1 giờ để tính toán.
- Một số hàm băm hiện nay:
  - MD5
  - SHA-1, SHA-2
  - ….

Nghịch lý ngày sinh (birthday paradox)

Bài toán 1: *Giả sử trong phòng có M sinh viên. Vậy xác suất để có hai SV có cùng ngày sinh là bao nhiêu phần trăm? (1 năm 365 ngày khác nhau)*

- *Theo nguyên lý chuồng bồ câu Dirichlet: cần có 365+1 = 366 người để tìm thấy 2 người có cùng ngày sinh với xác suất 100%. Vì vậy với 30 người thì xác xuất này rất nhỏ.*
- *Tính theo xác suất thống kê toán học thì*

$$M(M-1) >= 2 \times 365 \times \log_e 2 \quad (*)$$

chỉ cần 23 người là đủ để xác suất hơn 50%. Vì vậy bài toán này gọi là ***nghịch lý ngày sinh***

Nghịch lý ngày sinh (birthday paradox)

Điều này muốn nói lên rằng, *trong nhiều trường hợp xác suất để hai mẩu tin có cùng bản Hash là không nhỏ như chúng ta nghĩ.*

→Tính chống trùng mạnh

Slide 65

Nghịch lý ngày sinh (birthday paradox)

Bài toán 2: Giả sử bạn đang ở trong một lớp học với **M** sinh viên. Hỏi M tối thiểu là bao nhiêu để tồn tại một bạn khác có cùng ngày sinh với bạn với xác suất (XS) lớn hơn 50%?

- XS để 1 người khác ngày sinh với bạn là 364/365.
- →XS để M người đều khác ngày sinh với bạn là $(364/365)^M$.
- →XS để tồn tại ít nhất một người có cùng ngày sinh với bạn là: $1-(364/365)^M$
- Để XS này >50% ➔ **M>=253 người**

→Tính chống trùng yếu

Slide 66

Nghịch lý ngày sinh (birthday paradox)

- Áp dụng cho hàm băm, ta thấy **tính chống trùng mạnh giống bài toán 1**; **tính chống trùng yếu giống bài toán 2**.
  - Gọi **n** là số bit của giá trị băm h, có **$N=2^n$** giá trị băm khác nhau. Giả sử 2^n giá trị băm này là ngẫu nhiên, có khả năng xuất hiện như nhau.
  - Thay 365 của bất phương trình (*) bằng 2^n

$$M(M-1)>=2 \times 2^n \times \log_n 2$$

Giải bất phương trình trên, ta có xấp xỉ

$$M > \sqrt[2]{2^n} = 2^{n/2}$$

Slide 67

Nghịch lý ngày sinh (birthday paradox)

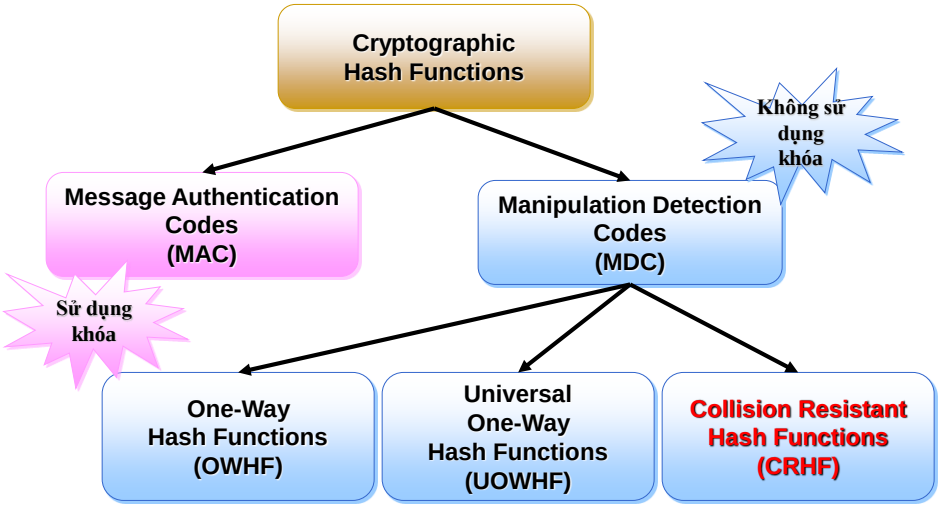
- Để tìm ra **hai thông điệp có cùng giá trị băm** (vét cạn) thì phải thử bao nhiêu thông điệp khác nhau?

  →Phải thử khoảng **$2^{n/2}$** thông điệp khác nhau (xác suất > 50%)

- Ví dụ: Nếu **n=128** thì phải thử **2^{64}** thông điệp (khá lớn), nghĩa là hàm hăm đạt được tính chống trùng mạnh (tương đương tấn công vét cạn khóa của DES)

Slide 68

Phân Loại hàm băm mật mã



Slide 69

5. Ứng dụng hàm băm

- **Lưu trữ mật khẩu**
- **Chứng thực thông điệp**
  (Message Authentication)
- **Chữ ký số**
  (Digital Signatures)
- **Các ứng dụng khác**
  (Other Applications)

Slide 70

5.1 Lưu trữ mật khẩu

- Mật khẩu bao gồm chuỗi các chữ cái (hoa, thường), chữ số và các ký tự đặc biệt (@, # …).
- Do tính chất của hàm toán học một chiều, mật khẩu của tài khoản được bảo vệ ngay cả trong trường hợp file lưu trữ mật khẩu hệ thống bị sao chép.

Username	Password
admin	@123!FitIuh
trandung	@123!Tran

Lưu trữ không mã hóa mật khẩu

Username	Password
admin	69c919ee4881666e4c90d51d4a2ed505
trandung	168838fe639bd5d8b660d5b008978759

Lưu trữ mã hóa mật khẩu với MD5

Slide 71

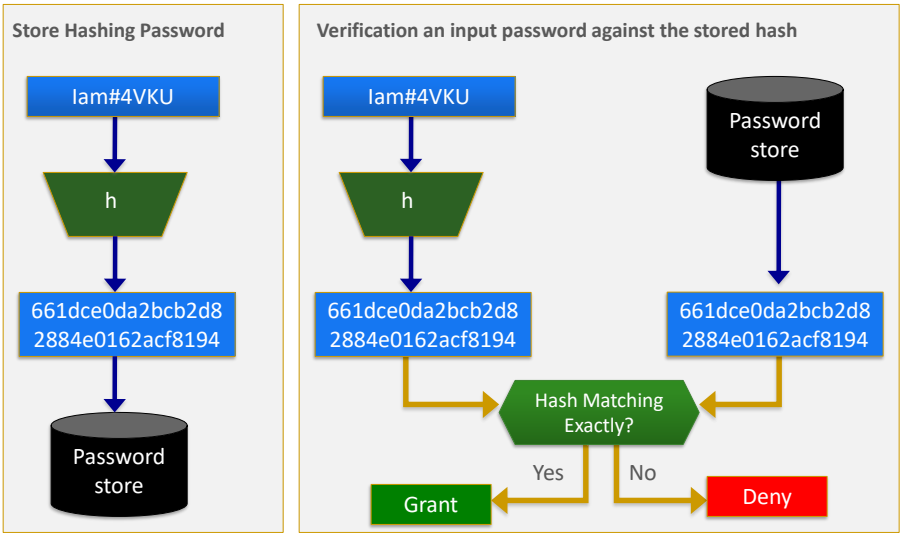
5.1 Lưu trữ mật khẩu

Dùng lưu trữ mật khẩu (băm password):

- Hàm băm được dùng để tạo **one-way password file**, trong cơ chế này giá trị băm của password được lưu, điều này tốt hơn là lưu chính bản rõ password. → password không bị truy xuất bởi kẻ tấn công nơi chứa password.
- Khi user nhập vào một password, thì giá trị băm của password được so với giá trị băm được lưu để kiểm tra.

Slide 72

Password Verification



Slide 73

5.2 Message Authentication

- Xác thực thông điệp liên quan đến các khía cạnh sau khi truyền tin trên mạng
  - **Bảo vệ tính toàn vẹn của thông điệp:** bảo vệ thông điệp không bị thay đổi hoặc có các biện pháp phát hiện nếu thông điệp bị thay đổi trên đường truyền.
  - **Kiểm chứng danh tính, nguồn gốc**: xem xét thông điệp có đúng do người xưng tên gửi không hay một kẻ mạo danh nào khác gửi.
  - **Không chối từ bản gốc:** trong trường hợp cần thiết, bản thân thông điệp chứa các thông tin chứng tỏ chỉ có người xưng danh gửi, không một ai khác có thể làm điều đó => Người gửi không thể từ chối hành động gửi, thời gian gửi và nội dung của thông điệp.
- Hàm băm dạng này, giá trị băm (h) được gọi là **tóm tắt thông điệp (*message digest*)**

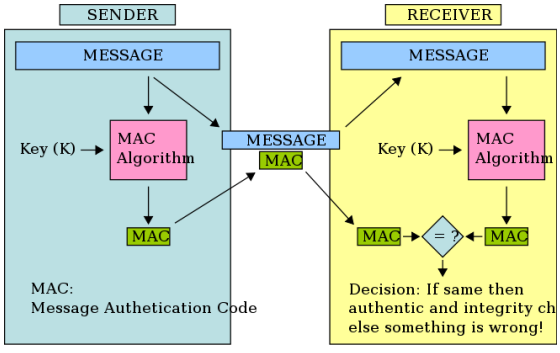
Slide 74

5.2 Message Authentication

- **Các yêu cầu bảo mật khi truyền mẩu tin:**
  - **Để lộ bí mật**: giữ bí mật nội dung mẩu tin, chỉ cho người có quyền biết.
  - **Thám mã đường truyền**: không cho theo dõi hoặc làm trì hoãn việc truyền tin.
  - **Giả mạo**: lấy danh nghĩa người khác để gửi tin.
  - **Sửa đổi nội dung**: thay đổi, cắt xén, thêm bớt thông tin.
  - **Thay đổi trình tự** các gói tin nhỏ của mẩu tin truyền.
  - **Sửa đổi thời gian**: làm trì hoãn mẩu tin.
  - **Từ chối gốc**: không cho phép người gửi từ chối trách nhiệm của tác giả mẩu tin.
  - **Từ chối đích**: không cho phép người nhận phủ định sự tồn tại và đến đích của mẩu tin đã gửi.

Slide 75

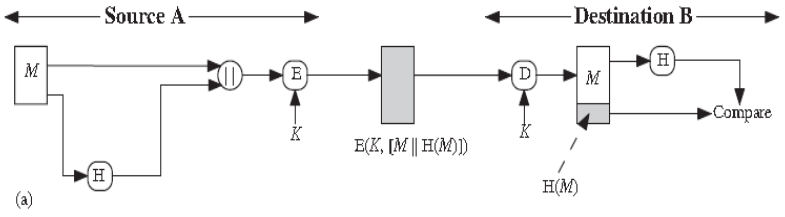
5.2 Message Authentication

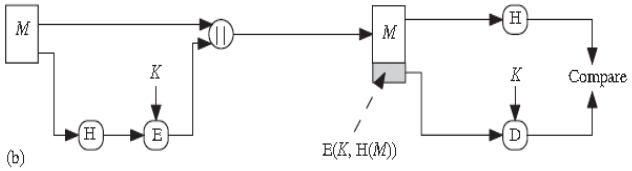


- bảo vệ toàn vẹn thông điệp và cả tính xác thực thông điệp bằng cách cho phép kiểm định (cũng là người sở hữu khóa bí mật) phát hiện bất kỳ thay đổi trong nội dung thông điệp

Slide 76

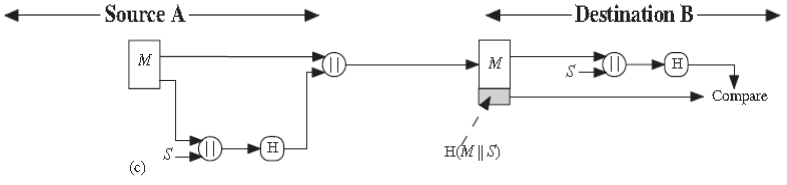
5.2 Message Authentication

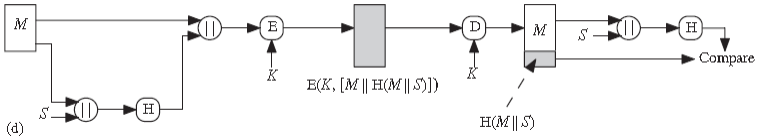




Slide 77

5.2 Message Authentication





Slide 78

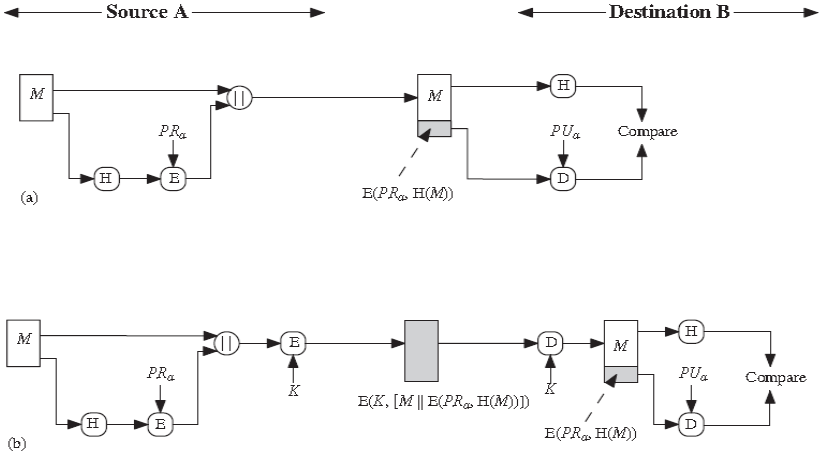
5.3. Chữ ký điện tử(Chữ ký số)

- Giá trị băm của thông điệp được mã hóa bằng **private key** của user, bất kỳ ai biết **public key** của user thì có thể thẩm tra thông điệp mà được gắn kết với chữ ký số.

- Kẻ tấn công muốn hiệu chỉnh thông điệp thì sẽ cần phải biết private key của user.

Slide 79

5.3. Chữ ký điện tử(Chữ ký số)



Slide 80

5.4 Các ứng dụng khác

Dùng nhận diện xâm hại (*intrusion detection*) và nhận diện virus (*virus detection*).

- Tính, lưu và bảo mật giá trị băm H(F) của các tập tin trong hệ thống (có thể lưu trên CD-R)
- Kẻ xâm hại cần phải hiệu chỉnh F mà không thay đổi H(F)

Slide 81

5.4 Các ứng dụng khác

- **Dùng:**

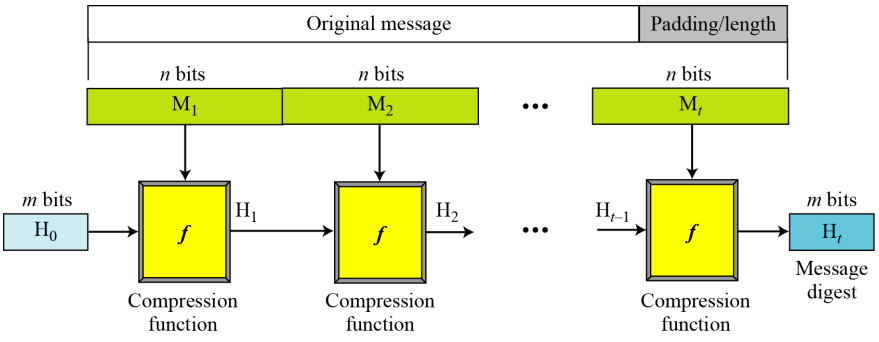
  Xây dựng hàm ngẫu nhiên giả (*pseudorandom function - PRF*)

  hoặc

  Phát sinh số ngẫu nhiên giả (*pseudorandom number generator - PRNG*)

Slide 82

6. Kiến trúc hàm băm an toàn



- Tác giả: Ralph Merkle, Ivan Damgård
- Hầu hết các hàm băm đều sử dụng cấu trúc này
- Ví dụ: SHA-1, MD5

Slide 83

7. Hàm băm MD5, SHA1

- MD5 (Message Digest)
  - Phát minh bởi Ron Rivest (RSA)
  - Phát triển từ MD4, trước đó MD2 (không an toàn)
  - Kích thước giá trị băm là 128-bit
  - 1994 và 1998: một phương pháp tấn công MD5 và một số thông điệp có cùng giá trị băm MD5 được chỉ ra (vi phạm tính chống trùng mạnh). Tuy nhiên MD5 vẫn còn sử dụng phổ biến

Slide 84

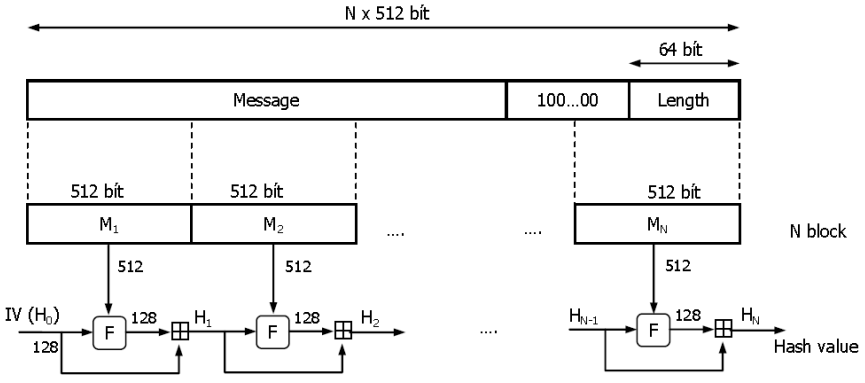
7. Hàm băm MD5, SHA1

- SHA (Secure Hash Algorithm)
  - Được phát triển bởi NIST 1993 (SHA-0)
  - 1995: SHA-1 - Chính phủ Mỹ chọn làm chuẩn quốc gia. Kích thước giá trị băm 160-bit
  - Hiện nay còn có SHA-224, SHA-256, SHA-384, SHA-512

	SHA-1	SHA-224	SHA-256	SHA-384	SHA-512
Message Digest Size	160	224	256	384	512
Message Size	$< 2^{64}$	$< 2^{64}$	$< 2^{64}$	$< 2^{128}$	$< 2^{128}$
Block Size	512	512	512	1024	1024
Word Size	32	32	32	64	64
Number of Steps	80	64	64	80	80

Slide 85

7.1 Hàm băm MD5 (128-bit, $\leq 2^{64}$-bit)

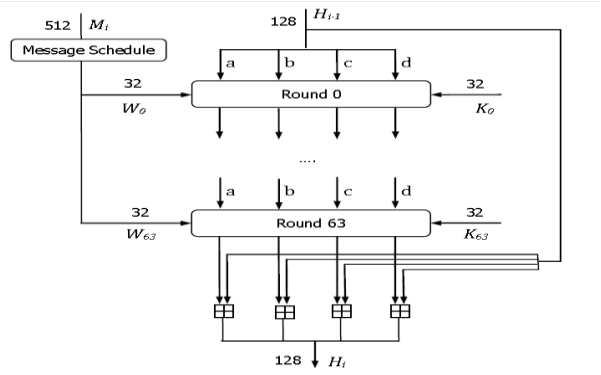


H_0 – 128-bit, chia thành 4 từ 32-bit, ký hiệu a,b,c,d – hằng số (thập lục phân)
a=01234567; b=89abcdef; c=fedbca98; d=76543210

Slide 86

7.1 Hàm băm MD5 (128-bit, ≤2^{64}-bit)

- **Cấu trúc hàm F tại mỗi bước lũy tiến**



K_j: phần nguyên của 2^{32}abs(sin(i)) với *i* biểu diễn radian
M_i được biến đổi qua hàm **message schedule** cho ra W_0, W_1, .., W_{63}
mỗi giá trị 32-bit

Slide 87

7.1 Hàm băm MD5 (128-bit, ≤2^{64}-bit)

Cấu trúc của một vòng trong F

Ở đây: b→c, c→d, d→a, b được tính qua hàm

$t=a+f(b,c,d)+W_i+K_i$

$b=b+ROTL(t,s)$

Hàm f(x,y,z):

f(x,y,z)=(x∧y)∨(¬x∧z) nếu vòng 0 – 15

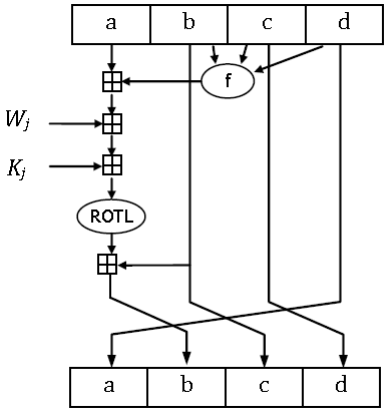
f(x,y,z)=(z∧x)∨(¬z∧y) nếu vòng 16 – 31

f(x,y,z)=x⊕y ⊕x nếu vòng 32 – 48

f(x,y,z)=y⊕ (x∨¬z) nếu vòng 49 – 63

Hàm ROTL(t,s): t được dịch vòng trái s-bit, với s là các hằng số cho vòng thứ i

Phép + (hay ⊞) là phép cộng modulo 2^{32}



Slide 88

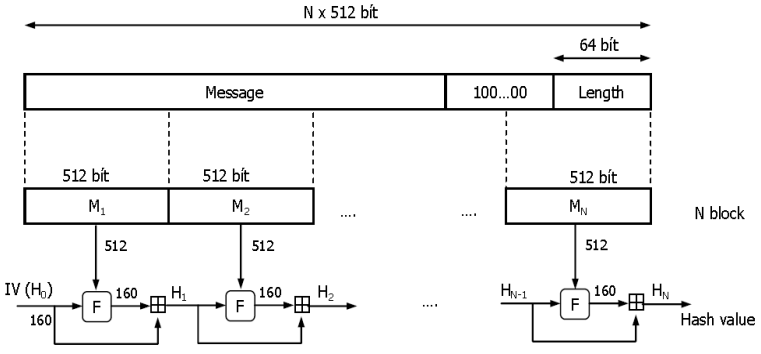
7.1 Hàm băm MD5 (128-bit, $\leq 2^{64}$-bit)

- **Hàm ROTL(t,s):** t được dịch vòng trái s-bit, với s là các hằng số cho vòng thứ i

i	*s*
0, 4, 8, 12	7
1, 5, 9, 13	12
2, 6,10, 14	17
3, 7, 11, 15	22
16, 20, 24, 28	5
17, 21, 25, 29	9
18, 22, 26, 30	14
19, 23, 27, 31	20
32, 36, 40, 44	4
33, 37, 41, 45	11
34, 38, 42, 46	16
35, 39, 43, 47	23
48, 52, 56, 60	6
49, 53, 57, 61	10
50, 54, 58, 62	15
51, 55, 59, 63	21

Slide 89

7.2 Hàm băm SHA-1 (160-bit, 2^{64}-bit)

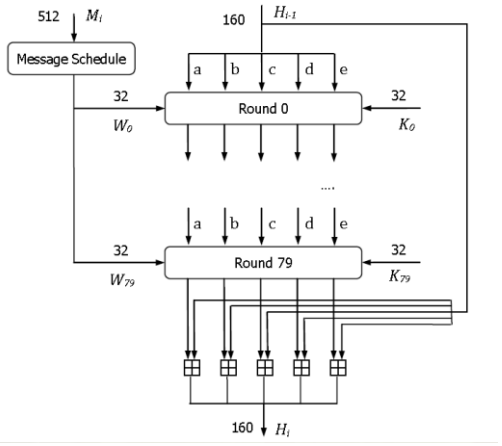


H_0 – 160-bit, chia thành 5 từ 32-bit, ký hiệu a,b,c,d,e – hằng số
a=67452301; b=efcdab89; c=98badcfe; d=10325476; e=c3d2e1f0

Slide 90

7.2 Hàm băm SHA-1 (160-bit, 2^{64}-bit)

- Cấu trúc hàm F cũng tương tự MD5, nhưng được thực hiện **80 vòng**



K_i là hằng số
K_i=5A827999 với 0≤i ≤19
K_i=6ED9EBA1 với 20≤i ≤39
K_i=8F1BBCDC với 40≤i ≤59
K_i=CA62C1D6 với 60≤i ≤79

Giá trị M_i – biến đổi qua message schedule cho ra 80 giá trị như sau:
- M_i chia thành 16 block 32-bit ứng với W_0, W_1,…, W_{15}.
- Các W_t (16≤t ≤79) được tính: W_t=ROTL(W_{t-3}+W_{t-8}+W_{t-14}+W_{t-16},1)

với phép cộng modulo 2^{32}

Slide 91

7.2 Hàm băm SHA-1 (160-bit, 2^{64}-bit)

Cấu trúc của một vòng trong F

Ở đây: a→b, c→d, d→e. Giá trị a và c được tính:

a=ROTL(a,5)+f(b,c,d)+e+W_i+K_i

c=ROTL(b,30)

Hàm f(x,y,z):

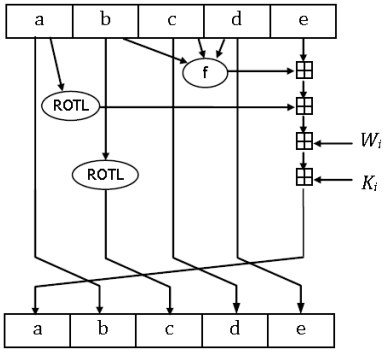
f(x,y,z)=Cf(x,y,z)=(x∧y) ⊕(–|x∧z) nếu vòng 0 – 19

f(x,y,z)=Parity(x,y,z)= x⊕y ⊕z nếu vòng 20 – 39

f(x,y,z)=Maj(x,y,z)=(x∧y) ⊕(y∧z)⊕(z∧x) nếu vòng 40 – 59

f(x,y,z)=Party(x,y,z)= x⊕y ⊕z nếu vòng 60 – 79

- Hàm Maj: giải sử x_i, y_i, z_i là bit thứ i của x,y,z thì bit thứ i của hàm Maj là giá trị nào chiếm đa số, 0 hoặc 1
- Hàm Ch: bit thứ i của hàm Ch là phép chọn: if x_i then y_i else x_i



Slide 92

So sánh giữa MD5 và SHA-1

- Khả năng chống lại tấn công brute-force:
  - Để tạo ra thông điệp có giá trị băm cho trước, cần 2^{128} thao tác với MD5 và 2^{160} với SHA-1
  - Để tìm 2 thông điệp có cùng giá trị băm, cần 2^{64} thao tác với MD5 và 2^{80} với SHA-1
- Khả năng chống lại thám mã: cả 2 đều có cấu trúc tốt
- Tốc độ:
  - Cả hai dựa trên phép toán 32 bit, thực hiện tốt trên các kiến trúc 32 bit
  - SHA-1 thực hiện nhiều hơn 16 bước và thao tác trên thanh ghi 160 bit nên tốt độ thực hiện chậm hơn
- Tính đơn giản: cả hai đều được mô tả đơn giản và dễ dàng cài đặt trên phần cứng và phần mềm

Slide 93

Hàm băm - Ứng dụng

- Hàm băm trên Java
- Key Stretching (tạo khóa bí mật từ mật khẩu)
- Lưu trữ mật khẩu
- Integrity checking (kiểm tra tính toàn vẹn dữ liệu)
- HMAC - Hashed Message Authentication Code(mã chứng thực thông điệp sử dụng hàm băm)
- Chữ ký điện tử

Slide 94

Hàm băm - Ứng dụng

Hàm băm trên JAVA

- Tạo hàm băm:
  - MessageDigest md;
  - md = MessageDigest.getInstance("MD5");
- Băm ("Digestion") dữ liệu:
  - byte[] data1, data2, result;
  - md.update(data1);
  - result = md.digest(data2);

Hàm băm - Ứng dụng

Mật mã hóa dựa trên mật khẩu (PBE)

Khóa của DES:

- Chiều dài 56 bits (trong thực tế cài đặt cần 64 bits)
- Phức tạp, Khó nhớ

→Sử dụng mật khẩu (password)

- Chiều dài thay đổi, không phải lúc nào cũng có 64 bits (hay 8 ký tự)

Mật mã hóa dựa trên mật khẩu

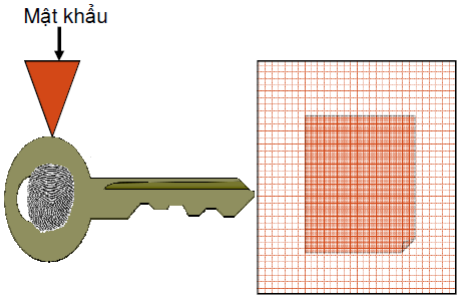
- Băm mật khẩu có kích thước bất kỳ thành khóa có đúng 64 bits.

Hàm băm - Ứng dụng

Mật mã hóa dựa trên mật khẩu (PBE)

Password Based Encryption

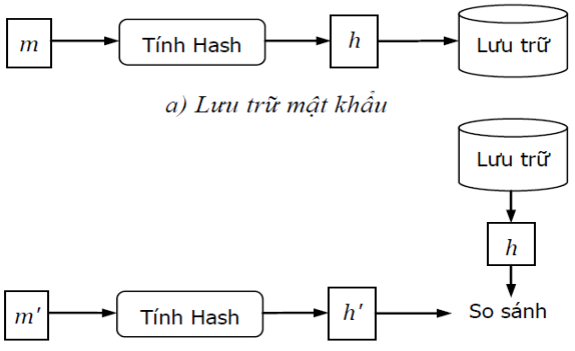
- Kết hợp một **hàm băm** và một giải thuật **mã hóa đối xứng** để mật hóa dữ liệu.



Slide 97

Một số ứng dụng MD5 và SHA-1

- Lưu trữ mật khẩu



a) Lưu trữ mật khẩu

b) Chứng thực mật khẩu, theo tính chống trùng, nếu h'=h thì m'=m

Slide 98

Một số ứng dụng MD5 và SHA-1

	username	password	email
1	admin	nhx64312	nguyen@yahoo.com
2	devil	kin32xz	nam@hotmail.com
3	vampire	62ntt34	hung@gmail.com

Lưu trữ password không mã hóa

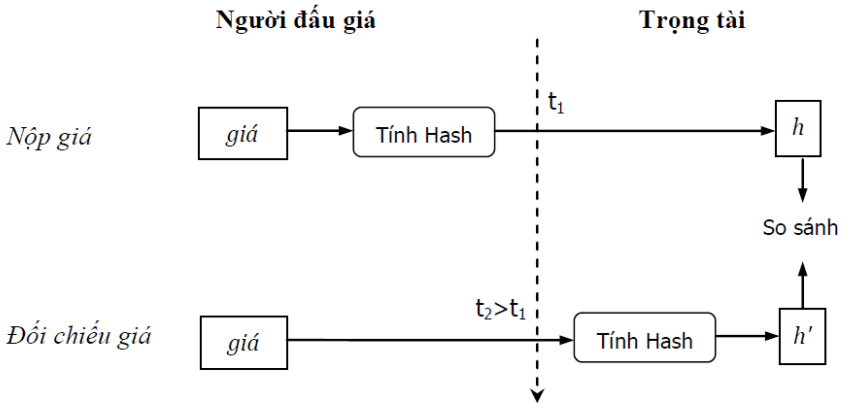
	username	password	Salt	email
1	admin	23dacd8cd768c95ad2e63cdc399c0535	64f84a9bdec1999e43c97ab12f8d9a36	nguyen@yahoo.com
2	devil	d400c472ab7a09ba87bf5c9715bbe118	66436db921558ae452f1e76a44e40aac	nam@hotmail.com
3	vampire	0d7b12ce9cf7ef3534aa5fee204eb0f5	1c798886f04910bad5a85fbec1c1bfea	hung@gmail.com

Lưu trữ password mã hóa bằng hàm hash MD5

Slide 99

Một số ứng dụng MD5 và SHA-1

- Đấu giá trực tuyến



Hình 5-6. Đấu giá bí mật

Slide 100

Một số ứng dụng MD5 và SHA-1

Đấu giá trực tuyến: Giả sử Alice, Bob và Trudy cùng tham gia đấu giá, họ sẽ cung cấp mức giá của mình cho trọng tài.

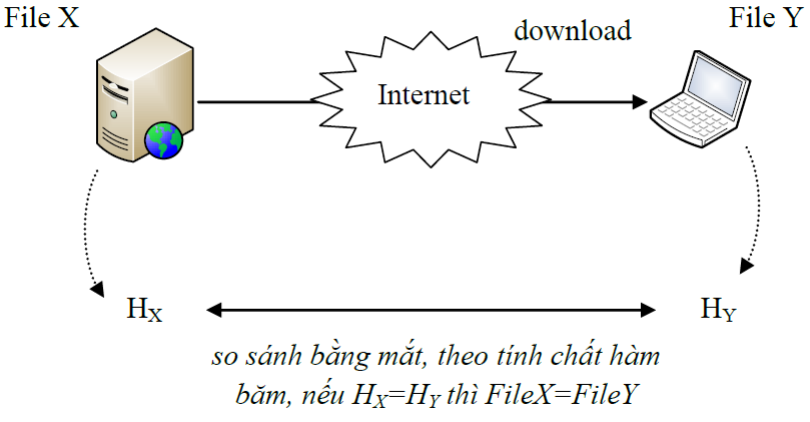
- Giả sử mức giá của Alice là 100, mức giá của Bob là 110, nếu Trudy thông đồng với trọng tài và biết được giá của Alice và Bob, Trudy có thể đưa ra mức giá 111 và thắng thầu.
- Có thể tránh những hình thức lừa đảo như vậy bằng cách sử dụng hàm băm. Từ mức giá bỏ thầu, Alice và Bob sẽ tính các giá trị băm tương ứng và chỉ cung cấp cho trọng tài các giá trị băm này.
- Vì hàm băm là một chiều, nếu trọng tài và Trudy bắt tay nhau thì cũng

không thể biết được giá của Alice và Bob là bao nhiêu. Đến khi công bố, Alice, Bob và Trudy sẽ đưa ra mức giá của mình. Trọng tài sẽ tính các giá trị băm tương ứng và so sánh với các giá trị băm đã nộp để bảo đảm rằng mức giá mà Alice, Bob và Trudy là đúng với ý định ban đầu của họ. Vì tính chống trùng của hàm băm nên Alice, Bob và Trudy không thể thay đổi giá so với ý định ban đầu.

Slide 101

Một số ứng dụng MD5 và SHA-1

- DownLoad File



Slide 102

Một số ứng dụng MD5 và SHA-1

Chữ ký điện tử (Digital signature) đảm bảo

- Authentication
- Integrity
- Non-repudiation

Slide 103

Tấn Công Hàm Băm

- **Kỹ thuật tấn công vét cạn**: kẻ tấn công tạo ra một lượng lớn các văn bản và lần lượt tính toán, so sánh giá trị băm của chúng để tìm ra đụng độ. Trên thực tế, kích thước bảng băm theo các thuật toán thông dụng là 64 bit, 128 bit …, do đó thời gian tính toán để tìm được đụng độ theo phương pháp là rất lớn.

Slide 104

Tấn Công Hàm Băm

Kỹ thuật phân tích mã:

- Hạn chế của kỹ thuật tấn công vét cạn là số phép tính lớn, khi độ dài của bảng băm tương đối lớn (>=128 bit) thì việc tìm va chạm mất rất nhiều thời gian. Hiện nay có nhiều phương pháp cho ta kết quả tốt hơn kỹ thuật vét cạn, có thể kể đến như:
- Tấn công theo kiểu gặp nhau ở giữa (meet – in – the – middle attack)
- Tấn công khác biệt giữa các module (The modular differential attack)
- Boomerang Attacks …

Slide 105

Xin chân thành cảm ơn!

Slide 106