CHƯƠNG IV

XÁC THỰC THÔNG ĐIỆP VÀ CHỮ KÝ SỐ

ThS. Nguyễn Cao Đạt

E-mail: dat@hcmut.edu.vn

NỘI DUNG TRÌNH BÀY

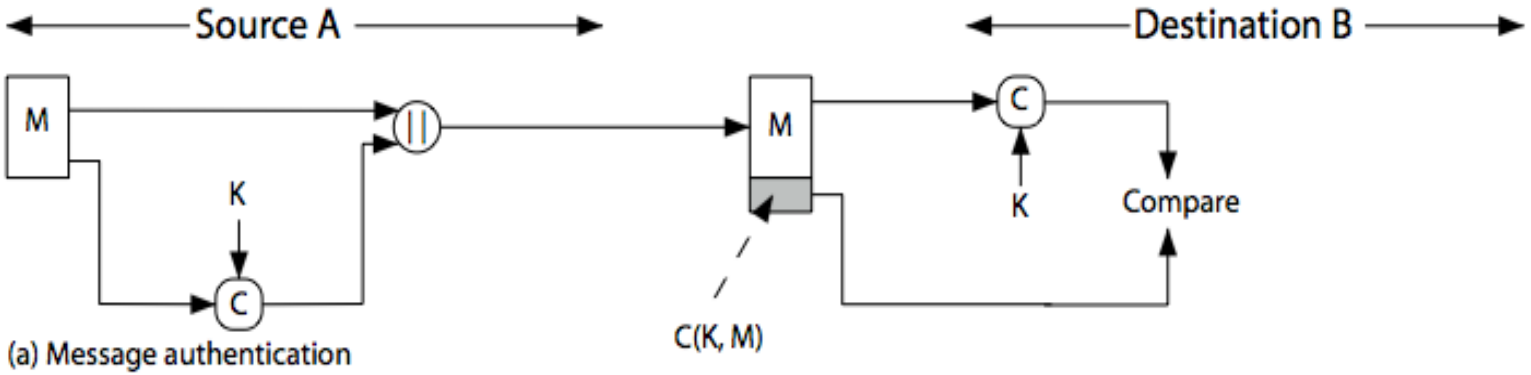
- **Xác thực thông điệp**
  - Khái niệm
  - Các hướng tiếp cận
- **Chữ ký số**
  - Khái niệm
  - Lược đồ chữ ký số

XÁC THỰC THÔNG ĐIỆP

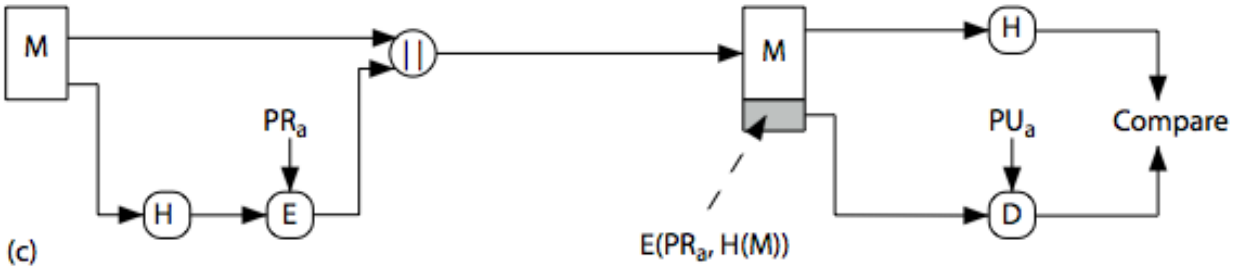
- **Xác thực thông điệp**
  - Bảo vệ tính toàn vẹn thông điệp
  - Xác nhận danh tính của người tạo ra thông điệp
  - Chống thoái thác về xuất xứ(giải quyết tranh chấp)
- **Chống lại các loại tấn công**
  - Giả mạo(masquerade)
  - Thay đổi nội dung(content modification)
  - Thay đổi trình tự(sequence modification)
  - Thay đổi thời gian(timing modification)

CÁC HƯỚNG TIẾP CẬN

- **Mã xác thực thông điệp**



- **Hàm băm và chữ ký số**



MÃ XÁC THỰC THÔNG ĐIỆP

- **Được sử dụng để đảm bảo tính xác thực và toàn vẹn.**
- **Cách tạo**

$$MAC = C_K(M)$$

- **Cô đọng một thông điệp M có chiều dài thay đổi dùng một khóa bí mật K thành một mã xác thực có chiều dài cố định.**
- **Là một hàm một chiều (nhiều – một)**
  - Nhiều thông điệp có cùng MAC
  - Nhưng tìm kiếm chúng rất khó khăn

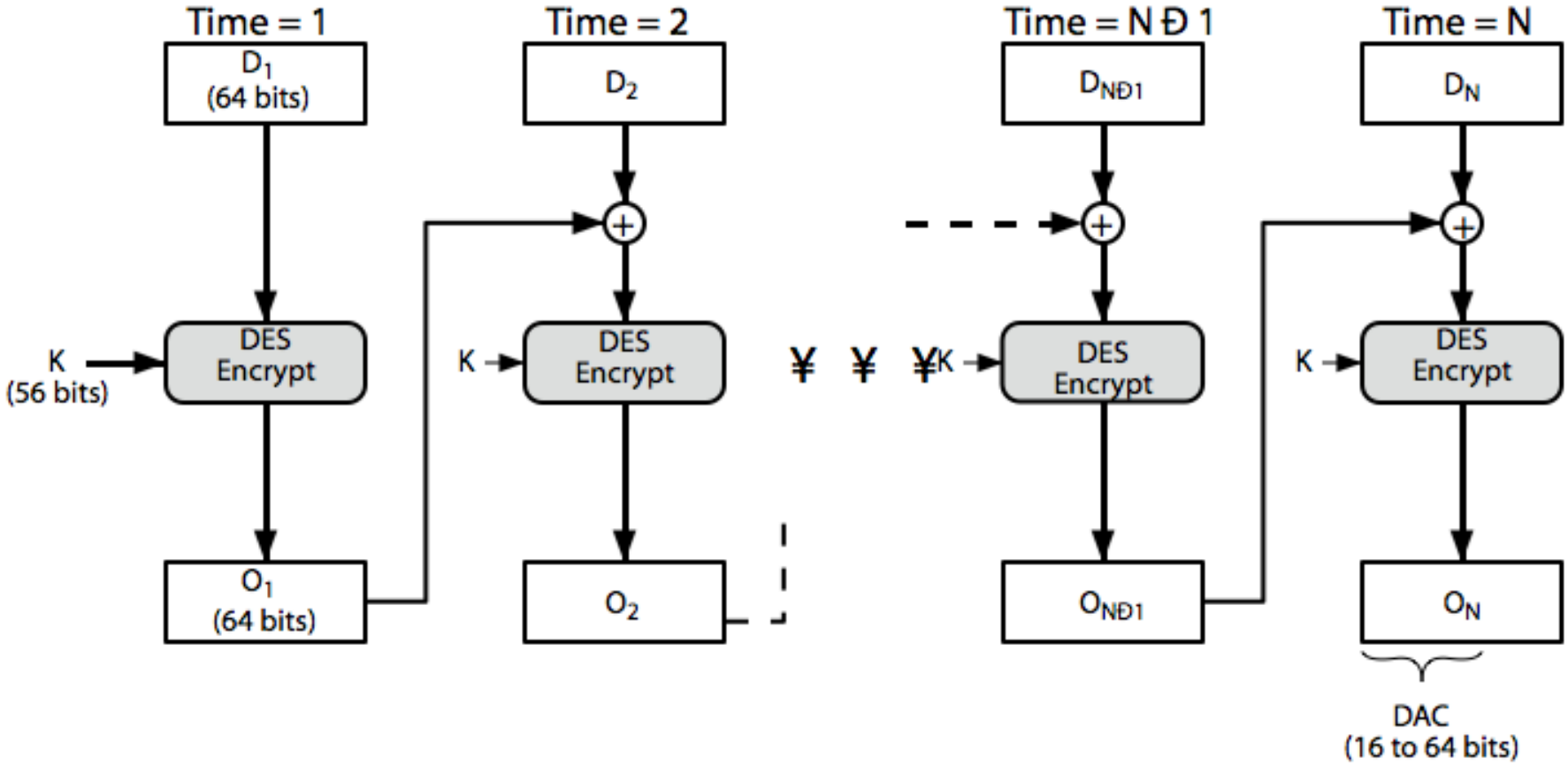
CÁC YÊU CẦU CỦA MÃ XÁC THỰC THÔNG ĐIỆP

1. Biết thông điệp và **mã xác thực thông điệp** của nó thì không khả thi để tìm ra một thông điệp khác có cùng **mã xác thực thông điệp**.
2. Các giá trị **mã xác thực thông điệp** phải phân bố đồng đều.
3. **Mã xác thực thông điệp** phải phụ thuộc như nhau trên tất cả các bit của thông điệp.

DÙNG MÃ HÓA ĐỐI XỨNG CHO MAC

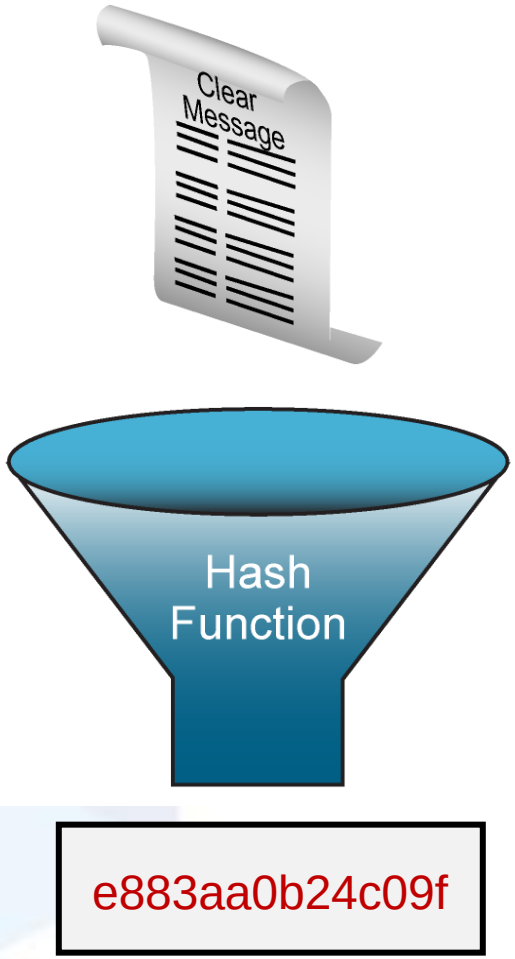
- **Mã hóa đối xứng dùng chế độ hoạt động CBC (cipher block chaining) và dùng đầu ra của khối cuối như là MAC**
- **Thuật toán xác thực dữ liệu (DAA) được dùng rộng rãi là MAC dựa trên DES-CBC**
  - Dùng IV=0 và thêm và các byte 0 ở khối cuối
  - Mã hóa thông điệp dùng DES trong chế độ CBC
  - Và dùng đầu ra hay M bits bên trái ($16 \leq M \leq 64$) của đầu ra khối cuối như là MAC.
- **Nhưng MAC dựa trên DES-CBC là quá nhỏ vì vậy hiện nay không an toàn.**

THUẬT TOÁN XÁC THỰC DỮ LIỆU



HÀM BĂM

- **Được sử dụng để đảm bảo tính toàn vẹn.**
- **Cách tạo `h = H(M)`**
- **Cô đọng một thông điệp M có chiều dài thay đổi thành một có chiều dài cố định.**
- **Là một hàm một chiều (nhiều – một)**
  - Nhiều thông điệp có cùng H
  - Nhưng tìm kiếm chúng rất khó khăn



CÁC YÊU CẦU CỦA HÀM BĂM

1. Cho h thì không khả thi để tìm x mà $H(x)=h$
2. Cho x thì không khả thi để tìm y mà $H(y)=H(x)$
3. Không khả thi để tìm x,y mà $H(y)=H(x)$

HÀM BĂM ĐƠN GIẢN

- **Nhiều đề xuất cho hàm băm đơn giản**
  - Dựa trên phép toán XOR của các khối thông điệp
  - Không an toàn vì có thể các thay đổi của thông điệp ảnh hưởng đến kết quả hàm băm.

- **Cần hàm băm dựa trên mật mã mạnh hơn**
  - Secure Hash Algorithm(SHA)
  - SHA-1, SHA-256, SHA-384.

DÙNG MÃ HÓA KHỐI NHƯ HÀM BĂM

- **Có thể dùng mã hóa khối như hàm băm**
  - Dùng H_0=0 và thêm các byte 0 vào khối cuối
  - Tính: $H_i = E_{M_i} [H_{i-1}]$
  - Và dùng kết quả của khối cuối làm giá trị băm.
  - Đơn giản như chế độ mã hóa CBC không dùng khóa
- **Kết quả là giá trị băm quá nhỏ(64-bit)**
  - Tấn công ngày sinh nhật
  - Tấn công "meet-in-the-middle"
- **Các biến thể khác cũng dễ bị tấn công**

TẤN CÔNG NGÀY SINH NHẬT

- Mọi người đều nghĩ rằng kết quả 64 bits của hàm băm là an toàn. Nhưng với tấn công ngày sinh nhật thì không.
- **Tấn công ngày sinh nhật thực hiện như sau:**
  - Kẻ tấn công tạo $2^{m/2}$ phiên bản của một thông điệp hợp lệ với cùng một nghĩa như nhau.
  - Kẽ tấn công cũng tạo ra $2^{m/2}$ phiên bản thông điệp giả mạo.
  - Hai tập thông điệp này được so sánh để tìm ra một cặp cùng giá trị băm(xác xuất là > 0.5)
  - Người dùng ký trên văn bản hợp lệ thì chữ ký cũng hợp lệ trên văn bản giả mạo.
- **Vì vậy giá trị MAC/băm phải có kích thước lớn hơn.**

AN TOÀN CỦA HÀM BĂM VÀ MAC

- **Tương tự như mã hóa khối**
- **Tấn công brute-force**
  - Có đề xuất phần cứng để bẻ gãy MD5
  - Giá trị băm 128 bits dễ bị tổn thương vì vậy nên dùng giá trị băm 160 bits thì tốt hơn.
  - MAC với các cặp (thông điệp, MAC) được biết
    - Có thể tấn công tìm kiếm khóa hay MAC
    - MAC phải có chiều dài ít nhất 128 bits mới an toàn

AN TOÀN CỦA HÀM BĂM VÀ MAC

- **Phân tích mã khai thác cấu trúc**
  - Có một số tấn công phân tích mã trên hàm băm lặp
    - $CV_i = f[CV_{i-1}, M_i]$; $H(M)=CV_N$
    - Như mã hóa khối cũng có các vòng
    - Tập trung vào các đụng độ trong hàm f
    - Khai thác các thuộc tính của hàm f

MAC DỰA TRÊN HÀM BĂM

- **Tạo MAC dựa trên hàm băm**
  - Hàm băm thông thường nhanh hơn.
  - Thư viện các hàm băm phổ biến rộng rãi
- **Thực hiện hàm băm trên thông điệp kết hợp với khóa**
- **Đề xuất nguyên thủy**

```
KeyedHash = Hash(Key|Message)
```

  - Đề xuất không an toàn
- **Dẫn đến sự phát triển của HMAC**

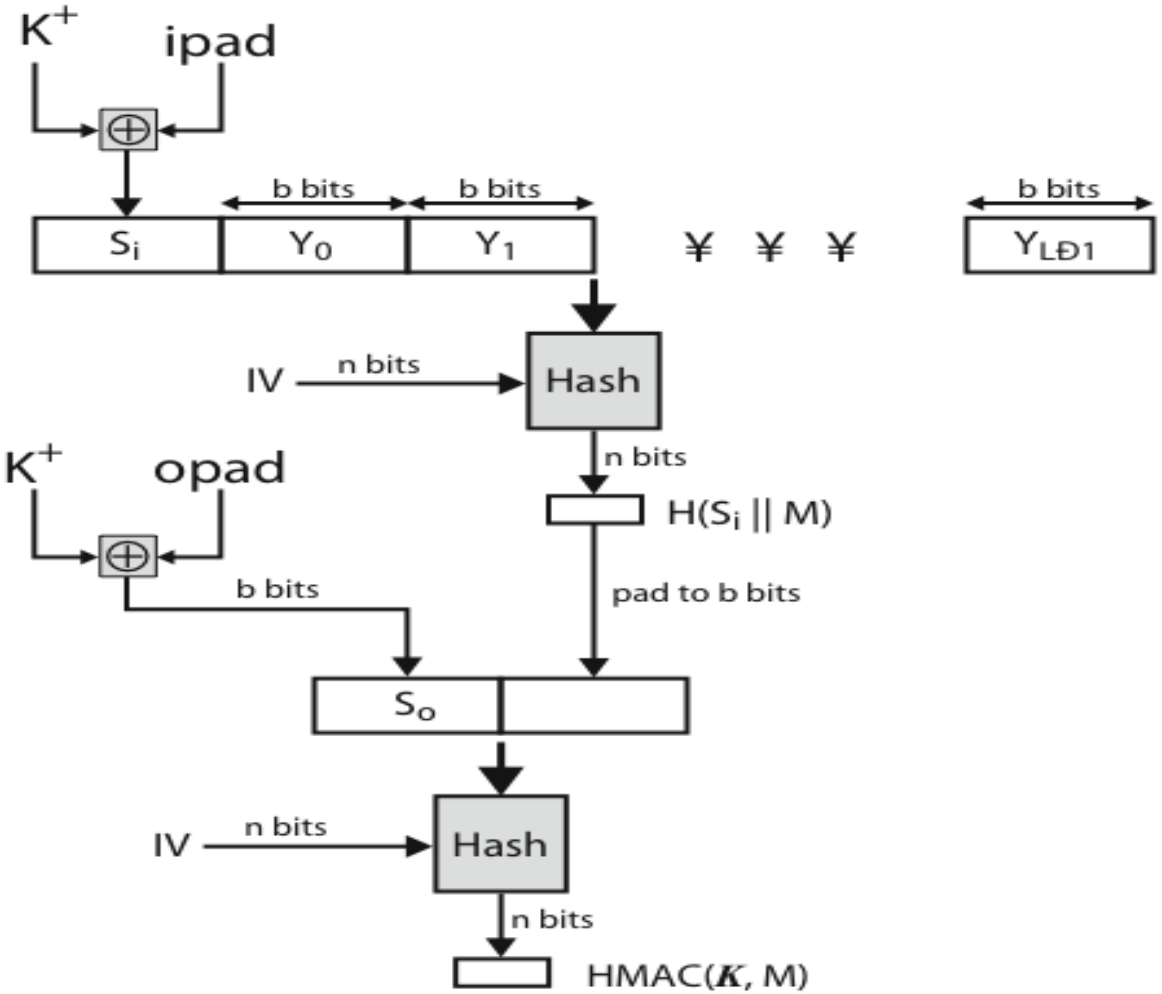
HMAC

- **Đặc tả như một chuẩn Internet với RFC 2104**
- **Dùng hàm băm trên thông điệp M và khóa K**

$$HMAC_K = Hash[(K^+ \text{ XOR opad}) \,||\, Hash[(K^+ \text{ XOR ipad})||M)]]$$

- K^+ là khóa được thêm vào một số byte
- opad, ipad được đặc tả là các hằng số
- Nhiều hàm băm được dùng
  - MD5, SHA-1, RIPEMD-160, Whirlpool

HMAC


K+
ipad
b bits
b bits
Si
Y0
Y1
¥ ¥ ¥
b bits
YLĐ1
IV
n bits
Hash
K+
opad
n bits
H(Si || M)
b bits
pad to b bits
So
IV
n bits
Hash
n bits
HMAC(K, M)

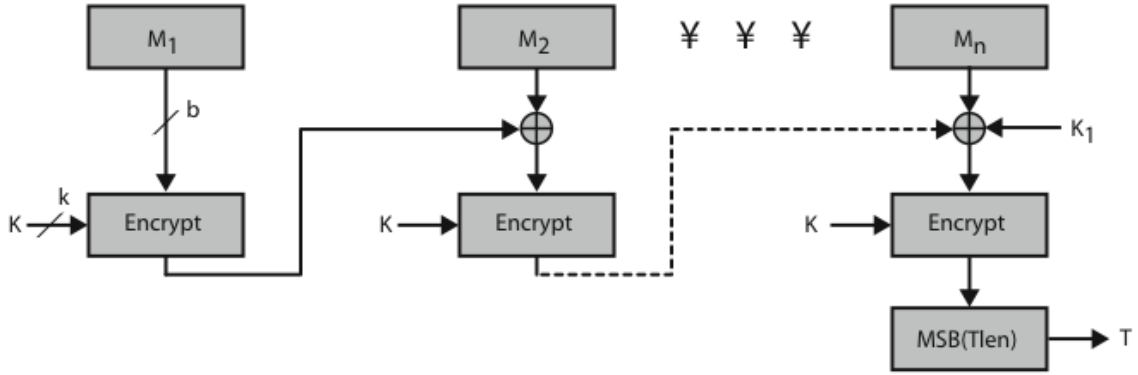
AN TOÀN CỦA HMAC

- **Đã chứng minh an toàn của HMAC liên quan đến thuật toán băm**
- **Các tấn công trên HMAC gồm:**
  - Tấn công brute force trên khóa
  - Tấn công ngày sinh nhật
- Hàm băm được chọn dựa trên các ràng buộc về tốc độ và tính an toàn.

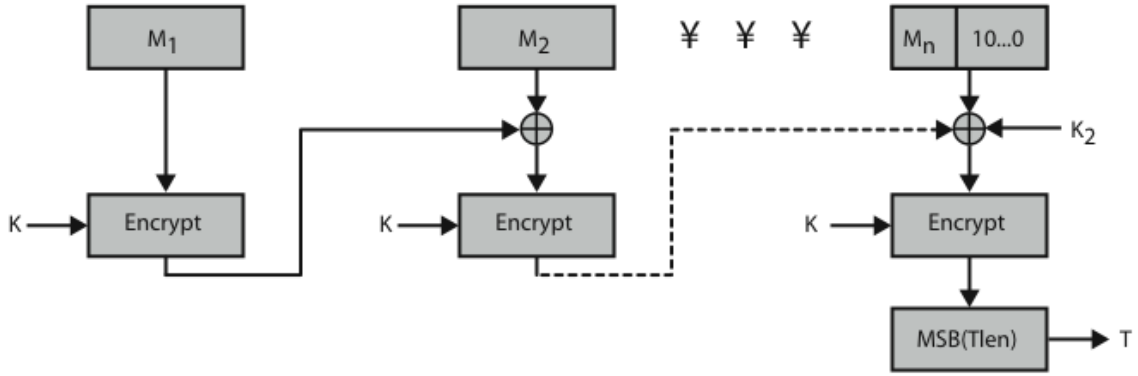
CMAC

- **Được mô tả như DAA (CBC-MAC)**
- Được dùng rộng rãi trong chính phủ và công nghiệp
- Nhưng kích thước MAC có giới hạn
- Sự hạn chế này có thể được khắc phục bằng cách sử dụng nhiều khóa được dẫn xuất từ một khóa.
- Được áp dụng bởi NIST với các thuật toán mã hóa AES và Triple-DES.
- Được đặc tả ở Special Publication NIST 800-38B.

CMAC



(a) Message length is integer multiple of block size



(b) Message length is not integer multiple of block size

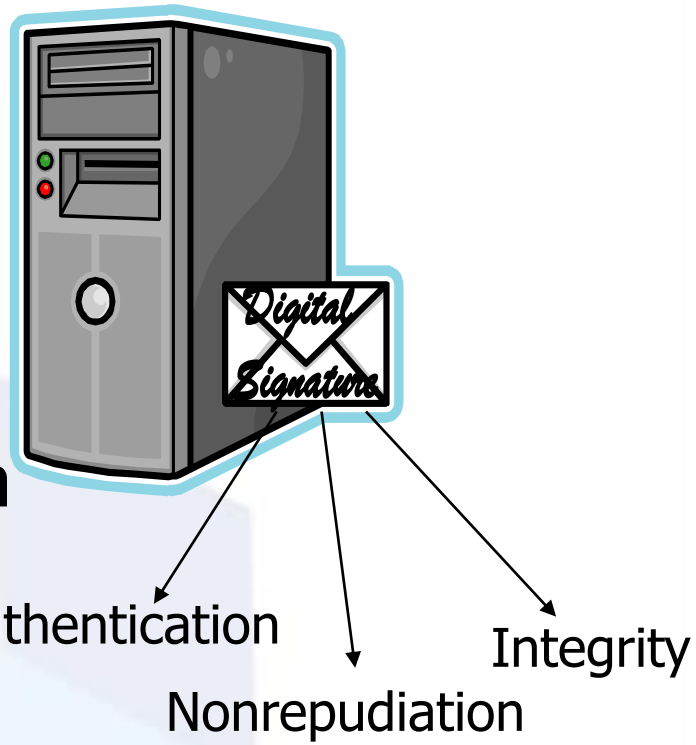
Figure 12.12 Cipher-Based Message Authentication Code (CMAC)

NỘI DUNG TRÌNH BÀY

- **Xác thực thông điệp**
  - Khái niệm
  - Các hướng tiếp cận
- **Chữ ký số**
  - Khái niệm
  - Lược đồ chữ ký số

CHỮ KÝ SỐ

- **Chữ ký số cung cấp khả năng**
  - Xác minh tác giả, ngày giờ đã ký.
  - Toàn vẹn thông điệp.
  - Được xác minh bởi một tổ chức thứ 3 để giải quyết tranh chấp.
- **Chữ ký số bao gồm cả chức năng xác thực thông điệp và một số chức năng bổ sung.**



CHỮ KÝ SỐ

- Chữ ký là xác thực và không thể giả: Chữ ký là bằng chứng cho thấy người ký, và không ai khác đã ký tài liệu.
- Chữ ký không thể tái sử dụng: Chữ ký là một phần của tài liệu và không thể được chuyển sang một tài liệu khác.
- Chữ ký là không thể thay đổi: Sau khi một tài liệu được ký, nó không thể được thay đổi.
- Chữ ký không thể phủ nhận: Đối với các mục đích pháp lý, chữ ký và tài liệu được coi là những minh chứng. Người ký không thể nói rằng họ đã không ký.

CÁC YÊU CẦU CỦA CHỮ KÝ SỐ

- **Phải phụ thuộc trên thông điệp được ký.**
- **Phải sử dụng thông tin duy nhất từ người gửi để tránh giả mạo và từ chối.**
- **Phải tương đối dễ dàng để tạo.**
- **Phải tương đối dễ dàng để nhận biết và xác minh.**
- **Không khả thi trong tính toán để giả mạo**
  - Một thông điệp mới với chữ ký số đang tồn tại.
  - Chữ ký số cho một thông điệp đã cho.
- **Lưu trữ chữ ký số trong thực tế.**

LƯỢC ĐỒ CHỮ KÝ SỐ

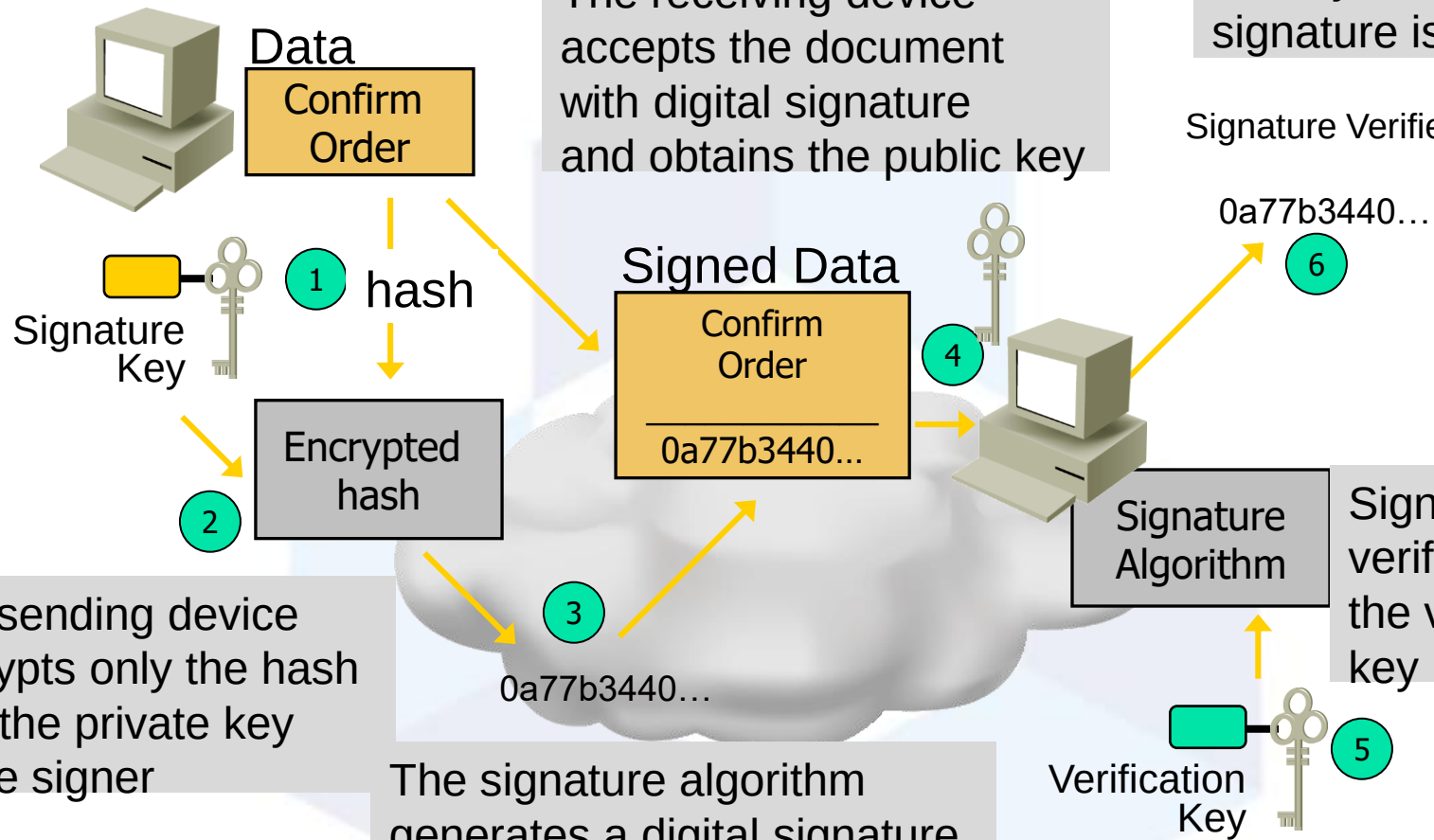
- ***Bên gửi***
  - Tính toán h = H(M) (Ví dụ dùng hàm băm SHA-1)
  - h được mã hóa với khóa riêng của người gửi để có chữ ký S
  - Bên gửi sẽ gửi M || S
- ***Bên nhận***
  - Lấy ra M. Tính toán h = H(M).
  - S được giải mã với khóa công khai của người gửi để có h'.
  - Xác minh h' = h hay không ?

LƯỢC ĐỒ CHỮ KÝ SỐ

The sending device creates a hash of the document

The receiving device accepts the document with digital signature and obtains the public key

Validity of the digital signature is verified

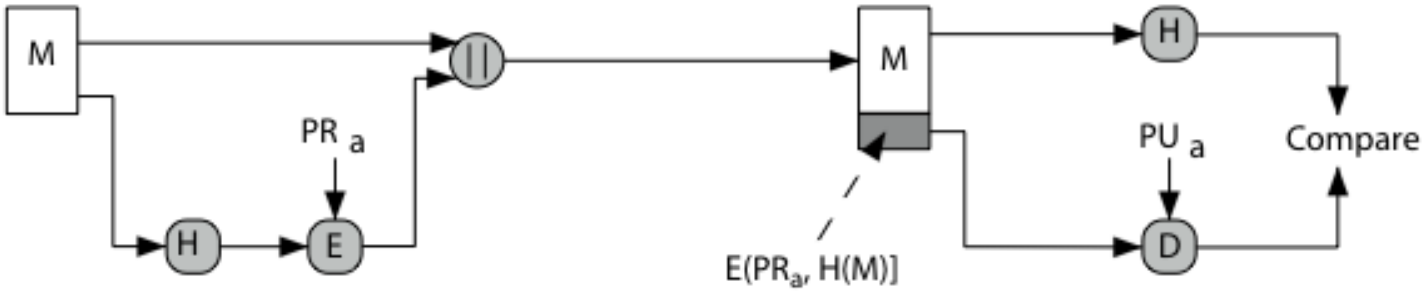


The sending device encrypts only the hash with the private key of the signer

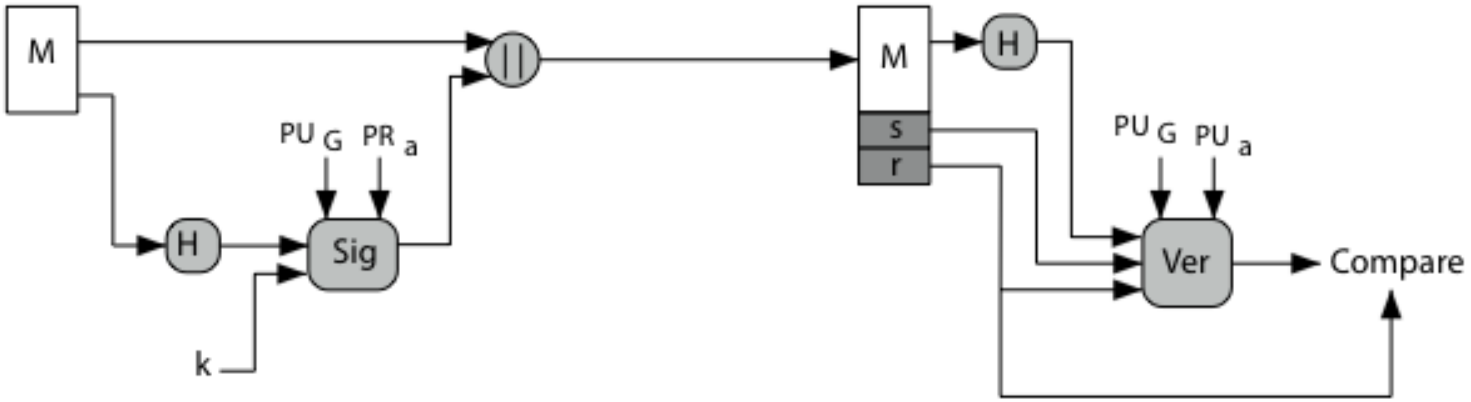
The signature algorithm generates a digital signature and obtains the public key

Signature is verified with the verification key

CÁC HƯỚNG TIẾP CẬN CHỮ KÝ SỐ



(a) RSA Approach



(b) DSS Approach

LƯỢC ĐỒ CHỮ KÝ VỚI RSA

- **Khóa công khai của A là (n_a, e_a)**
- **Bên A ký và gửi**
  - Tính h = H(M)
  - S được tính: $S = h^{d_a} \bmod n_a$.
  - A gửi M||S
- **Bên nhận xác minh**

$$h' = S^{e_a} \equiv h^{e_a d_a} \equiv h^{k\phi(n_a)+1} \equiv h \bmod n_a.$$

CHUẨN CHỮ KÝ SỐ

- **Digital Signature Standard (DSS)**
- Lược đồ chữ ký được chính phủ Mỹ phê duyệt
- Được thiết kế bởi NIST và NSA vào đầu thập kỷ 90
- Công bố như FIPS-186 vào năm 1991
- Được sửa đổi vào năm 1993, 1996 và sau đó năm 2000
- **DSS là chuẩn, DSA là thuật toán**
- **Dùng thuật toán băm SHA**
- **FIPS 186-2 (2000) bao gồm cả chữ ký dựa trên RSA và ECC**

THUẬT TOÁN CHỮ KÝ SỐ

- Digital Signature Algorithm (DSA)
- Tạo chữ ký chiều dài 320 bits
- Chỉ là một lược đồ chữ ký số
- An toàn dựa trên bài toán logarit rời rạc
- Một biến thể của các lược đồ được ElGamal & Schnorr trình bày

TẠO KHÓA TRONG DSA

- **Dùng chung các giá trị toàn cục(p,q,g)**
  - Chọn một số nguyên tố lớn p sao cho $2^{L-1} < p < 2^{L}$
    - L= 512 đến 1024 bits và là bội của 64
  - Chọn q với $2^{159} < q < 2^{160}$
    - q là một ước nguyên tố của $(p-1)$
  - Chọn $g = h^{(p-1)/q}$
    - $1<h<p-1$ và $h^{(p-1)/q} \bmod p > 1$
- **Người dùng chọn khóa riêng và tính toán khóa công khai**
  - Chọn $x<q$
  - Tính toán $y = g^{x} \bmod p$

TẠO CHỮ KÝ TRONG DSA

- **Để ký một thông điệp M, bên gửi**
  - Tạo ra khóa ngẫu nhiên k, $k<q$
  - k phải ngẫu nhiên, được hủy sau khi dùng và không bao giờ được tái sử dụng.
- **Tính toán cặp chữ ký**

$$r = (g^k \bmod p) \bmod q$$

$$s = [k^{-1}(H(M) + xr)] \bmod q$$

- **Gửi chữ ký (r,s) với thông điệp M**

XÁC MINH CHỮ KÝ TRONG DSA

- **Bên nhận có thông điệp M và chữ ký (r,s)**
- **Để xác minh chữ ký, bên nhận tính toán**

$$w = s^{-1} \bmod q$$

$$u1 = [H(M)w] \bmod q$$

$$u2 = (rw) \bmod q$$

$$v = [(g^{u1}\ y^{u2}) \bmod p] \bmod q$$

- **Nếu v=r thì chữ ký đã được xác minh**

ƯU ĐIỂM CỦA DSA

- **Kích thước chữ ký nhỏ hơn RSA**
  - 320 bits
  - Tương đương khoảng 2 lần giá trị băm SHA-1
- **Tất cả các tính toán cho việc ký và xác minh là nhỏ hơn so với RSA**
  - Thực hiện phép toán modulo q
  - Q có kích thước 160 bits

TÓM TẮT

- Xác thực thông điệp là một cơ chế hay dịch vụ được dùng để xác minh tính toàn vẹn của thông điệp. Hai kỹ thuật thường dùng cho xác thực thông điệp là MAC(HMAC, CMAC) và hàm băm an toàn.
- Chữ ký số là một cơ chế xác thực nhằm xác nhận danh tính của người tạo ra thông điệp và chống thoái thác về xuất xứ(giải quyết tranh chấp). Chữ ký số được tính toán trên giá trị băm của thông điệp và mã hóa bằng khóa riêng của người gửi.
- Có hai lược đồ chữ ký số dựa trên RSA và DSA.