Trường Đại học Bách Khoa
Đại học Quốc gia TP.HCM

Khoa Điện - Điện tử

Bộ môn điện tử

ĐỒ ÁN 1

Encoding

Bùi Thị Huyền Như - 2212464

Gmail: nhu.buiiud@hcmut.edu.vn
Tel: 0706.229.026

GVHD: Trần Hoàng Linh

Ngày 16 tháng 2 năm 2025

Mục lục

Danh sách hình vẽ

1 Đặt vấn đề

2 Một số kĩ thuật mã hóa cơ bản

2.1 Hàm băm

Hàm băm một chiều (one-way hash function) là một công cụ mật mã được sử dụng trong nhiều ứng dụng. Chúng được sử dụng cùng với các thuật toán khóa công khai cho cả mã hóa và chữ ký số. Chúng được sử dụng trong kiểm tra tính toàn vẹn. Chúng được sử dụng trong xác thực. Chúng có rất nhiều ứng dụng trong rất nhiều giao thức khác nhau. Hơn cả các thuật toán mã hóa, hàm băm một chiều là công cụ làm việc chính của mật mã hiện đại.

Một hàm băm một chiều cần có 2 tính chất: Tính một chiều và tính chống xung đột. Thứ nhất, Tính một chiều tức là từ dữ liệu ban đầu ta có thể dễ dàng tính toán được giá trị băm của nó, nhưng sau khi băm sẽ không thể khôi phục được ("không thể"có nghĩa là "không thể thực hiện trong bất kỳ khoảng thời gian hợp lý nào."). Và thứ hai, tính chống xung đột (collision-free) tức là những dữ liệu khác nhau sau khi băm không có đầu ra là giống nhau.

Phá vỡ một hàm băm có nghĩa là chứng minh rằng một trong hai hoặc cả hai tính chất trên không còn đúng.

Năm 1990, Ron Rivest đã phát minh ra hàm băm MD4. Năm 1992, ông cải tiến MD4 và phát triển một hàm băm khác: MD5. Năm 1993, Cơ quan An ninh Quốc gia (NSA) đã công bố một hàm băm rất giống với MD5, được gọi là SHA (Secure Hash Algorithm). Sau đó, vào năm 1995, NSA đã thay đổi SHA, viện dẫn một điểm yếu mới được phát hiện mà họ từ chối giải thích chi tiết, được gọi là SHA-1. Nhưng kể từ năm 2005, SHA-1 đã không được coi là an toàn.

Vào năm 2010, nhiều tổ chức đã khuyến nghị thay thế nó. NIST chính thức không còn sử dụng SHA-1 vào năm 2011 và không cho phép sử dụng chữ ký số vào năm 2013. Tất cả các nhà cung cấp trình duyệt web lớn đã ngừng chấp nhận chứng chỉ SSL SHA-1 vào năm 2017. Vào tháng 2 năm 2017, CWI Amsterdam và Google tuyên bố họ đã thực hiện một cuộc tấn công xung đột chống lại SHA-1, xuất bản hai tệp PDF không giống nhau tạo ra hàm băm SHA-1 giống nhau.

Năm 2001, SHA-2 cũng được tạo ra bởi Cơ quan An ninh Quốc gia (NSA). Mã hàm băm này gần như thay thế hoàn toàn phiên bản trước đó SHA-1. SHA-2 bao gồm 2 mã hàm băm bảo mật, SHA-256 và SHA-512. Cả 2 mã hàm băm này khá tương đồng nhưng có kích cỡ khác nhau.

SHA-3, Mã này trước đây được gọi là Keecak, được lựa chọn bởi các nhà chức trách sau một cuộc thi được tổ chức công khai giữa các nhà sáng tạo không thuộc NSA. Chiều dài của mã này tương đương với SHA-2. Tuy nhiên, chiều dài bên trong thì hoàn toàn khác biệt so với các hàm băm họ SHA và chưa có bất kỳ thông tin gì về việc nó sẽ được phát hành rộng rãi. Tính đến năm 2022, NIST không có kế hoạch loại bỏ SHA-2. Mục đích của SHA-3 là nó có thể được thay thế trực tiếp cho SHA-2 trong các ứng dụng hiện tại nếu cần thiết, và để cải thiện đáng kể độ bền vững của bộ công cụ thuật toán băm tổng thể của NIST.

2.2 SHA-1 (Secure Hash Algorithm 1)

2.2.1 Giới thiệu về SHA-1

2.2.2 Các bước thực hiện SHA-1

Bước 1: Nhận văn bản đầu vào, chia nó thành một mảng các mã ASCII của các ký tự và chuyển sang mã nhị phân 8-bit.

Ví dụ:

$$\text{'Do an 1'} \rightarrow \text{[d,o, ,a,n, ,1]} \xrightarrow{ASCII} \text{[100, 111, 32, 97, 110, 32, 49]}$$
$$\rightarrow 01100100\ 01101111\ 00100000\ 01100001\ 01101110\ 00100000\ 00110001$$

Bước 2: Thêm một bit 1 kế tiếp và các bit 0 để đạt độ dài là bội số của 512 - 64 (trong ví dụ này là 512 - 64 = 448 bit).

$$\underbrace{01100100\ 01101111\ 00100000\ 01100001\ 01101110\ 00100000\ 00110001}_{56\text{ bit}}\ \underbrace{1}_{\text{thêm 1 bit 1}}\ \underbrace{000...000}_{391\text{ bit 0}}$$

$\rightarrow$ Tổng các bit là 448.

Bước 3: Thêm độ dài của thông điệp gốc (64-bit).
Qua bước này, tổng độ dài của chuỗi hiện tại là một bội số của 512.

Trong ví dụ, độ dài của thông điệp ban đầu là 56 bit $\rightarrow$ Biểu diễn số 56 trong hệ nhị phân 64-bit như sau:

$\rightarrow$ 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00111000

Khi đó, với 448 bit ban đầu, ghép tiếp 64 bit, ta được độ dài của chuỗi là 512-bit hoàn chỉnh.

*Lưu ý: Nếu độ dài thông điệp lớn hơn, ta sẽ chia nhỏ thành nhiều khối 512-bit để xử lý.
Bước 4: Chia nhỏ khối 512-bit thành 16 từ (word), mỗi từ 32-bit.

$$W_0 = 01100100\ 01101111\ 00100000\ 01100001$$

$$W_1 = 01101110\ 00100000\ 00110001\ 10000000$$

$$W_2 = 00000000\ 00000000\ 00000000\ 00000000$$

$$...$$

$$W_{14} = 00000000\ 00000000\ 00000000\ 00000000$$

$$W_{15} = 00000000\ 00000000\ 00000000\ 00111000$$

Bước 5: Mở rộng từ 16 từ 32-bit thành 80 từ 32-bit
Với W_0 đến W_{15} lấy trực tiếp từ thông điệp được tính toán như ở trên, W_{16} đến W_{79} được tính như sau
Mỗi từ thứ i (với i từ 16 đến 79) được tính bằng công thức:

$$W_i = (W_{i-3} \oplus W_{i-8} \oplus W_{i-14} \oplus W_{i-16}) \lll 1$$

Trong đó:

- $\oplus$ là phép XOR bit.
- W_i là từ thứ i trong dãy mở rộng 80 từ.

Ví dụ:

$$W_{16} = (W_{13} \oplus W_8 \oplus W_2 \oplus W_0) \lll 1$$

$$= (0 \oplus 0 \oplus 0 \oplus 1685004385) \lll 1 = 1685004385 \lll 1$$

$$W_{16} = 3370008770$$

Tương tự với W_i khác, ta sẽ được khối 80 word 32-bit.

Bước 6: Khởi tạo giá trị hash. SHA-1 sử dụng 5 giá trị ban đầu (hằng số 32-bit):
$H_0 = 0x67452301$ $H_1 = 0xEFCDAB89$ $H_2 = 0x98BADCFE$
$H_3 = 0x10325476$ $H_4 = 0xC3D2E1F0$

Bước 7: 80 vòng lặp xử lý với quy trình như sau:

$$(A, B, C, D, E) \leftarrow (H_0, H_1, H_2, H_3, H_4)$$

Mỗi vòng lặp thực hiện:
1. **Tính toán giá trị tạm thời:**

$$T = (A \lll 5) + f(B, C, D) + E + W_t + K_t$$

Trong đó:

- $f(B, C, D)$ là hàm phi tuyến, thay đổi theo từng giai đoạn (*).
- W_t là từ mở rộng thứ t.
- K_t là hằng số vòng lặp (4 hằng số khác nhau)(**).
- T là giá trị tạm thời.

(*) Hàm $f(B, C, D)$ được định nghĩa khác nhau cho ba vòng lặp của thuật toán, như sau:

Vòng lặp 1 ($0 \leq t < 20$): $f(B, C, D) = (B \wedge C) \vee (\neg B \wedge D)$

Vòng lặp 2 ($20 \leq t < 40$): $f(B, C, D) = B \oplus C \oplus D$

Vòng lặp 3 ($40 \leq t < 60$): $f(B, C, D) = (B \wedge C) \vee (B \wedge D) \vee (C \wedge D)$

Vòng lặp 4 ($60 \leq t < 80$): $f(B, C, D) = B \oplus C \oplus D$

(**) Hằng số K_t có các giá trị sau đây:

Vòng lặp 1 ($0 \leq t < 20$): $K_t = 0x5A827999$

Vòng lặp 2 ($20 \leq t < 40$): $K_t = 0x6ED9EBA1$

Vòng lặp 3 ($40 \leq t < 60$): $K_t = 0x8F1BBCDC$

Vòng lặp 4 ($60 \leq t < 80$): $K_t = 0xCA62C1D6$

2. **Cập nhật các biến:**

$$E = D, \quad D = C, \quad C = B \lll 30, \quad B = A, \quad A = T$$

3. **Lặp lại cho đến vòng 80.**

Bước 8: Cập nhật giá trị H sau 80 vòng lặp:
$H_0 = H_0 + A \qquad H_1 = H_1 + B \qquad H_2 = H_2 + C \qquad H_3 = H_3 + D \qquad H_4 = H_4 + E$

*Lưu ý: Với thông điệp có nhiều khối 512-bit, thực hiện như sau:
Sau khi xử lý Block[1], các giá trị H0, H1, H2, H3, H4 được cập nhật. Các giá trị H0, H1, H2, H3, H4 mới này sẽ được sử dụng làm đầu vào cho việc xử lý Block[2]. Quá trình này lặp lại cho đến khi tất cả các khối 512 bit được xử lý.

Bước 9: Xuất ra giá trị băm SHA-1:
Sau khi xử lý toàn bộ dữ liệu, ta nối 5 giá trị H_0, H_1, H_2, H_3, H_4 lại để tạo thành chuỗi băm 160-bit (40 ký tự hex):

$$\text{SHA-1 Hash} = H_0 \| H_1 \| H_2 \| H_3 \| H_4$$

Với các bước làm trên, 'do an 1' sau khi băm sẽ là:

$$190a04f297856bff18618feb759f33bac2d8f3df$$

2.3 SHA-256

2.3.1 Giới thiệu về SHA-256

2.3.2 Các bước thực hiện SHA-256

3 Tài liệu tham khảo

1. Descriptions of SHA-256, SHA-384, and SHA-512 from NIST.

2. Cryptanalysis of SHA-1.

3. Youtube: How Does SHA-1 Work - Intro to Cryptographic Hash Functions and SHA-1.

4. SHA-1 wikipedia.

5. Hash function wikipedia

ref1

ref2

ref3