

Mật mã

Tài liệu

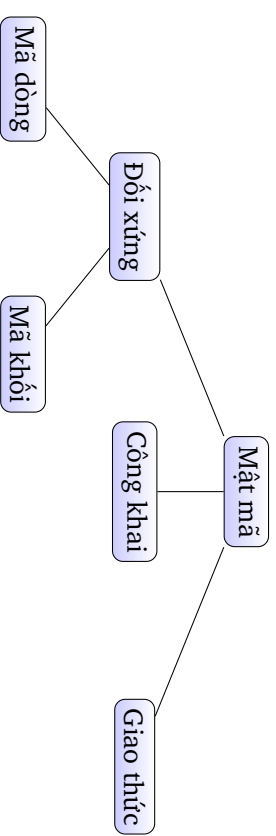
<https://www.crypto-textbook.com>



2 / 30



VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG



4 / 30



VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

Nội dung

- 1 Giới thiệu
- 2 Mã hóa và giải mã với hệ mã dòng
- 3 Bộ sinh số ngẫu nhiên
- 4 One Time Pad (OTP)
- 5 Hệ mã dòng trong thực tế



ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

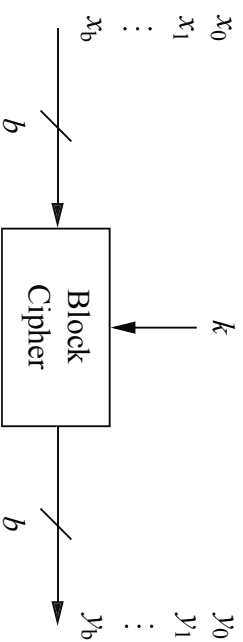
Nhập môn An toàn Thông tin Mã dòng

1 / 30



VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

Mã khối



Các hệ **mã khối** mã hoá mỗi lần cả một khối của bản rõ.

Ví dụ

Kích thước khối của

- AES (Advanced encryption standard) là 128 bit;
- DES (Data encryption standard) hoặc 3DES là 64 bit.

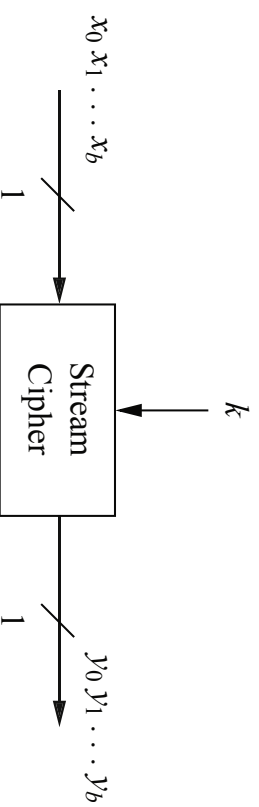
Định nghĩa (Mã hoá và giải mã)

Dòng bản rõ, bản mã và khóa là các bit riêng rẽ. Cụ thể

$$x_i, y_i, s_i \in \{0, 1\}.$$

- Mã hoá: $y_i = \text{Enc}(s_i, x_i) = x_i \oplus s_i$.
- Giải mã: $x_i = \text{Dec}(s_i, y_i) = y_i \oplus s_i$.

Mã dòng



Các hệ **mã dòng** mã hoá từng bit riêng rẽ.

Nội dung

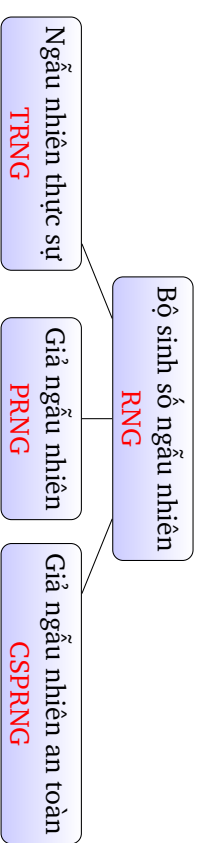
- 1 Giới thiệu
- 2 Mã hóa và giải mã với hệ mã dòng
- 3 Bộ sinh số ngẫu nhiên
- 4 One Time Pad (OTP)
- 5 Hệ mã dòng trong thực tế

Khoá của mã dòng

- **Hỏi:** Làm thế nào để sinh dòng bit khoá s_i ?
- **Trả lời:** Liên quan đến việc sinh dãy số **ngẫu nhiên**.



Ba kiểu số ngẫu nhiên



Thuật ngữ:

- **R**andom **N**umber **G**enerator
- **T** rue **R**NG
- **P**seudo RNG
- **C**ryptographically **S**ecure **P**RRNG



Ví dụ

Alice muốn mã hoá ký tự A, biểu diễn bởi mã ASCII. Mã ASCII của 'A' là $65_{10} = 1000001_2$. Giả sử các bit đầu tiên của dòng khoá là $(s_0, \dots, s_6) = 0101100$.

Alice	Oscar	Bob
$x_0, \dots, x_6 = 1000001 = A$		
\oplus		
$s_0, \dots, s_6 = 0101100$		
$y_0, \dots, y_6 = 1101101 = m$		
	$m = 1101101$	
	\rightarrow	
		$y_0, \dots, y_6 = 1101101 = m$
		\oplus
		$s_0, \dots, s_6 = 0101100$
		$x_0, \dots, x_6 = 1000001 = A$

Nội dung

- 1 Giới thiệu
- 2 Mã hóa và giải mã với hệ mã dòng
- 3 Bộ sinh số ngẫu nhiên
- 4 One Time Pad (OTP)
- 5 Hệ mã dòng trong thực tế



Sinh số giả ngẫu nhiên

- Bộ sinh số giả ngẫu nhiên sinh dãy bằng cách tính toán từ một giá trị seed ban đầu.

- Thông thường chúng được tính theo công thức truy hồi:

$$s_0 = \text{seed}$$

$$s_{i+1} = f(s_i), \quad i = 0, 1, \dots$$

- hoặc tổng quát hơn

$$s_{i+1} = f(s_i, s_{i-1}, \dots, s_{i-t})$$

với t là một giá trị cố định.

Ví dụ dùng rand()

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <time.h>
4
5 int main(int argc, char **argv)
6 {
7     srand(time(0));
8     printf("%d\n", rand());
9     return 0;
10 }
```

Hàm time(0) trả về số giây tính từ thời điểm ban đầu kỷ nguyên máy tính (00:00:00 UTC, ngày 1 tháng 1 năm 1970); và dùng nó làm giá trị seed.

Sinh số ngẫu nhiên thực sự

- Các số ngẫu nhiên thực sự phát sinh từ quá trình vật lý.
- Ví dụ:** tung đồng xu, tung xúc xắc, di chuyển chuột, thời gian ấn phím, Microphone, camera, sự thay đổi tốc độ của ổ cứng,...

- Trên Linux, nguồn ngẫu nhiên có thể lấy từ file /dev/random.

```
1 > hexdump -C -n 8 /dev/random
2
3 00000000  10 59 69 d4 dd 1e ad 66  |.Y1....f|
4 00000008
5
```

Ví dụ

Hàm rand() trong ANSI C:

$$s_0 = 12345$$

$$s_{i+1} = 1103515245 \cdot s_i + 12345 \pmod{2^{31}}, \quad i = 0, 1, \dots$$

Nội dung

- 1 Giới thiệu
- 2 Mã hóa và giải mã với hệ mã dòng
- 3 Bộ sinh số ngẫu nhiên
- 4 One Time Pad (OTP)
- 5 Hệ mã dòng trong thực tế



Định nghĩa (One Time Pad)

One Time Pad (OTP) là hệ mã dòng trong đó

- 1 dòng bit khoá s_i được sinh bằng bộ sinh số ngẫu nhiên thực sự.
- 2 mỗi dòng bit khoá chỉ được sử dụng **một lần**.



Mục đích

Xây dựng hệ mã “hoàn hảo”.

Định nghĩa (An toàn không điều kiện)

Một hệ mật là *an toàn không điều kiện* hoặc *an toàn theo lý thuyết thông tin* nếu nó không thể bị phá ngay cả khi kẻ tấn công có nguồn tài nguyên tính toán vô hạn.

Sinh số giả ngẫu nhiên an toàn

Bộ sinh số giả ngẫu nhiên an toàn là bộ sinh số giả ngẫu nhiên với tính chất: **Không thể dự đoán được**.

Định nghĩa (Không dự đoán được)

Cho n bit

$$s_i, s_{i+1}, \dots, s_{i+n-1}.$$

Không có thuật toán chạy trong thời gian đa thức để dự đoán bit tiếp theo s_{i+n} với xác suất thành công lớn hơn 50%.



Tính xác thực của OTP

- OTP có thể bị giả mạo.
Thay đổi các bit của bản mã sẽ làm cho các bit tương ứng của bản rõ bị thay đổi.
- OTP không cho phép xác thực nội dung thông điệp hay cho phép chống lại việc sửa đổi thông điệp.

Nội dung

- 1 Giới thiệu
- 2 Mã hóa và giải mã với hệ mã dòng
- 3 Bộ sinh số ngẫu nhiên
- 4 One Time Pad (OTP)
- 5 Hệ mã dòng trong thực tế

Nhược điểm của OTP

Người dùng phải

- sinh ra các khóa bí mật lớn,
- chia sẻ chúng an toàn,
- giữ chúng bí mật,
- tránh sử dụng lại khóa:

$$Y_1 \oplus Y_2 = (x_1 \oplus s) \oplus (x_2 \oplus s) \\ = x_1 \oplus x_2$$

có thể rút ra nhiều thông tin về x_1, x_2 .

Bài tập

Giả sử bạn biết mã hóa của thông điệp “attack at dawn” dùng mã hoá OTP là

6c73d5240a948c86981bc294814d

(Bản rõ ở dạng mã ASCII 8-bit và bản mã được viết ở dạng hexa).
Bản mã của thông điệp “attack at dusk” với cùng khóa OTP là gì?

Mã dòng với PRNG

Ví dụ (Đồng dư tuyến tính LCG)

$S_0 = \text{seed}$

$$S_i = A \cdot S_i + B \mod m, \quad i = 0, 1, \dots$$

với m có kích thước khoảng 100 bit; và $S_i, A, B \in \{0, 1, \dots, m-1\}$.

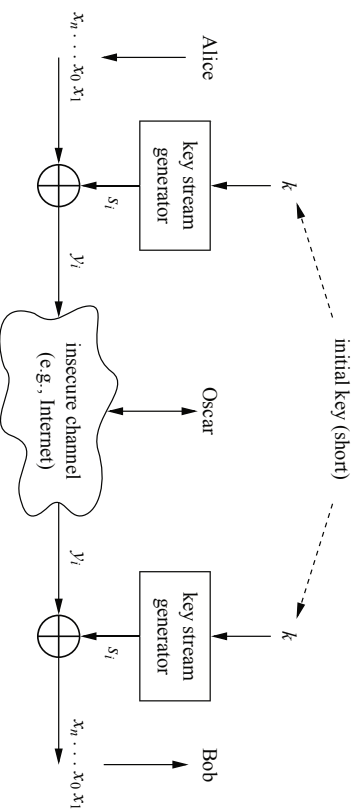
Khoá của hệ mã dòng là $k = (A, B)$.

Tấn công hệ mã dựa trên LCG 2

$$A = (S_2 - S_3)(S_1 - S_2)^{-1} \mod m$$

$$B = S_2 - S_1(S_2 - S_3)(S_1 - S_2)^{-1} \mod m$$

Mã dòng thực tế



Tấn công hệ mã dựa trên LCG

Giả sử Oscar biết x_1, x_2, x_3 .

- 1 Oscar tính S_1, S_2, S_3 .
- 2 Ta có

$$S_2 = A \cdot S_1 + B \mod m$$

$$S_3 = A \cdot S_2 + B \mod m$$

Đây là hệ phương trình đồng dư tuyến tính với hai biến!



Không dùng LCG để sinh dòng bit cho khoá!