MÃ DÒNG

Phan Quốc Tín – tinpq@uit.edu.vn

Mã đối xứng

- Định nghĩa: Một hệ mã đối xứng $(\mathcal{K},\mathcal{M},\mathcal{C})$ là hệ mã bao gồm 2 thuật toán "hiệu quả" (E,D) sao cho
  - E:$\mathcal{K} \times \mathcal{M} \to \mathcal{C}$
  - D:$\mathcal{K} \times \mathcal{C} \to \mathcal{M}$
  - $\forall m \in \mathcal{M}, k \in \mathcal{K}: D\big(k, E(k,m)\big) = m$
  - E thường là một thuật toán ngẫu nhiên
  - D phải là một thuật toán tất định

One-Time-Pad (Vernam 1917)

- Là ví dụ đầu tiên về hệ mã an toàn
  - $\mathcal{M} = \mathcal{C} = \{0,1\}^n$
  - $\mathcal{K} = \{0,1\}^n$
  - Khóa k có chiều dài bits bằng với thông điệp m

3

One-Time-Pad (Vernam 1917)

- c = E(k,m) = k $\oplus$ m
- D (k,c) = k $\oplus$ c

msg:	0	1	1	0	1	1	1
key:	1	0	1	1	0	1	0
CT:							

$\oplus$

- D(k,E(k,m)) = D(k, k $\oplus$ m) = k $\oplus$ (k $\oplus$ m) = (k $\oplus$ k) $\oplus$ m = 0 $\oplus$m = m
- Có dùng được trong thực tế hay không?

4

One-Time-Pad (Vernam 1917)

- Cho trước thông điệp (m) và từ mã tương ứng (c). Có thể tính toán được khóa k từ m và c hay không?
  - Không
  - Có thể, $k = m \oplus c$
  - Có thể tính được một nữa chiều dài bit của k
  - Có thể, $k = m \oplus m$

5

One-Time-Pad (Vernam 1917)

- Mã hóa / giải mã nhanh nhưng khóa dài
- OTP có an toàn không?
- Một mã an toàn là như thế nào?

6

Định lý an toàn của Shannon (1949)

- Ý tưởng cơ bản: bản mã không để lộ bất kỳ thông tin nào về bản rõ.
- Một hệ mã gồm 2 thuật toán (E,D) được định nghĩa trên $(\mathcal{K},\mathcal{M},\mathcal{C})$ an toàn hoàn hảo nếu:

  $\forall m_0, m_1 \in \mathcal{M}: \ len(m_0) = len(m_1)$ và $\forall c \in \mathcal{C}$

$$\Pr[E(k, m_0) = c] = \Pr[E(k, m_1) = c]$$

Trong đó, k được phân bố đều trong $\mathcal{K}$ ($k \leftarrow_{uniform} \mathcal{K}$)

7

Định lý an toàn của Shannon (1949)

$$\Pr[E(k, m_0) = c] = \Pr[E(k, m_1) = c]$$

Cho trước bản mã, không thể biết được bản rõ là m_0 hay m_1 (với mọi m_0, m_1).

➔không thể tấn công chỉ dựa vào bản mã

8

Tính an toàn của OTP

- Mệnh đề: OTP có chiều dài khóa bằng chiều dài thông điệp thì an toàn hoàn hảo
- Chứng minh

$\forall m, c$:

$$\Pr(E(k,m) = c) = \frac{|\#keys\ k \in \mathcal{K}: E(k,m) = c|}{|\mathcal{K}|}$$

$|k \in \mathcal{K}: E(k,m) = c|$ là hằng số

Ghi chú: || là số lượng phần tử

9

Tính an toàn của OTP

- Cho $m \in \mathcal{M}$ và $c \in \mathcal{C}$. Có bao nhiêu khóa k để mã hóa m thành c?
  - Không có
  - 1
  - 2
  - Phụ thuộc vào m
- Điều kiện $|\mathcal{K}| \geq |\mathcal{M}|$ khó chấp nhận trong thực tế

10

Mã dòng

- Ý tưởng: để làm cho OTP an toàn trong thực tế
  - Thay thế khóa k "ngẫu nhiên" bằng khóa "giả ngẫu nhiên"
  - PRG: bộ phát sinh giả ngẫu nhiên (PseudoRandom Generator)
  - PRG là một thuật toán tất định

$$G: \{0,1\}^s \rightarrow \{0,1\}^n \text{ với } n \gg s$$

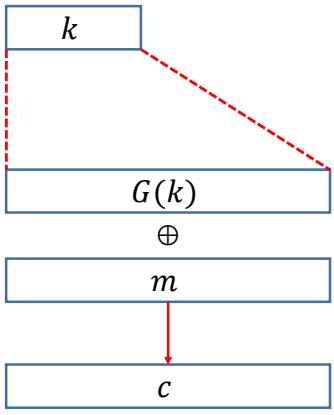
seed

- Định nghĩa: mã dòng là một phương pháp mã hóa trong đó khóa và thuật toán được áp dụng đối với từng bit một trong dòng dữ liệu

11

Mã dòng

- $c = E(k, m) = m \oplus G(k)$
- $D(k, c) = c \oplus G(k)$



12

Mã dòng: hiện thực OTP

- Cần một định nghĩa khác về an toàn
- An toàn sẽ phụ thuộc vào PRG

13

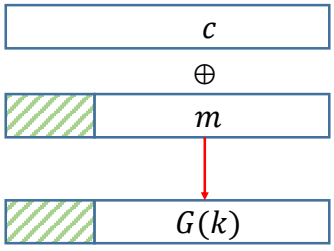
PRG phải có tính "không thể tiên đoán"

- Giả sử có thể tiên đoán được một phần kết quả của PRG

$$\exists i, G(k)_{1,\ldots,i} \rightarrow G(k)_{i+1,\ldots,n}$$

(nếu biết được i bits thì có thể đoán được n-i bits còn lại)

- Thì



Nếu $G(k)_{1,\ldots,i} \rightarrow G(k)_{i+1}$
Có nguy hiểm không???

14

PRG phải có tính "không thể tiên đoán"

- PRG có thể tiên đoán nếu

∃ thuật toán A và ∃ $1 \leq i \leq n-1$ sao cho

Pr[A (G(k) | $_{1,\ldots,i}$)] = G(k) | $_{i+1}$] ≥ ½ + ϵ (non-neg $\epsilon \geq \frac{1}{2^{30}}$)

- Định nghĩa: PRG là không thể tiên đoán nếu ∀i, không tồn tại thuật toán A nào để tìm ra bit (i+1) với non-neg ϵ

15

Các PRGs yếu

- Bộ phát sinh giả ngẫu nhiên của *Ling. Cong* với các tham số a, b, p:

$r[i] \leftarrow [a \cdot r[i-1] + b] \; mod \; p$
Trả về các bits của $r[i]$
i++

Seed = r[0]

- Không được sử dụng hàm random() trong mật mã học (Kerperos V4)

glibc random():
r[i] ← (r[i-3] + r[i-31]) % 2^{32}
output r[i] >> 1

16

Tấn công nhằm vào OTP

- Two-Time-Pad không an toàn
  - Không bao giờ sử dụng khóa của OTP hơn 1 lần

$$C_1 \leftarrow m_1 \oplus PRG(k)$$
$$C_2 \leftarrow m_2 \oplus PRG(k)$$

Nếu tin tặc dùng $c_1 \oplus c_2 \rightarrow \ \boldsymbol{m_1 \oplus m_2}$
Suy đoán bằng cách sử dụng các từ tiếng Anh phổ biến như "_the_"

17

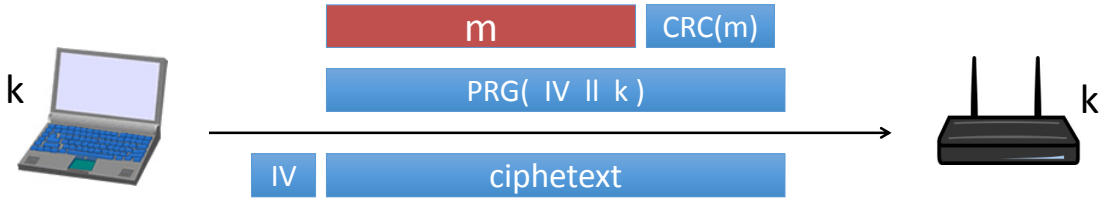
Các ví dụ trong thực tế

- Dự án Venona (1941-1946)
  - Dự án của quân đội Mỹ dùng để giải mã các thông điệp của Liên Xô
  - Liên Xô đã gửi khoảng 35.000 trang được mã hóa bằng 1 khóa
- MS-PPTP (Windows NT)
  - Client và server sử dụng cùng 1 key để chia sẻ dữ liệu

18

Các ví dụ trong thực tế

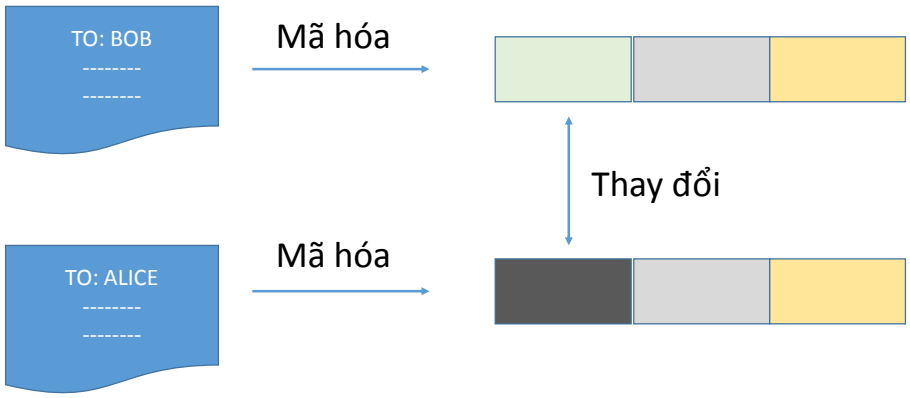
❑802.11b WEP:



- Chiều dài của IV: 24 bits
- Lặp lại IV sau $2^{24} \approx$ 16M frames
- Trong một số 802.11 cards: IV bị reset về 0 sau khi tắt và mở nguồn

Các ví dụ trong thực tế

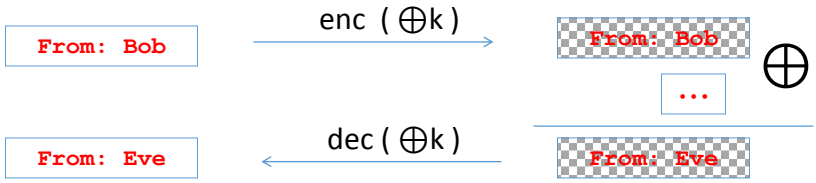
❑Mã hóa và lưu vào đĩa cứng



20

Tính toàn vẹn của OTP không được đảm bảo

- Tính toàn vẹn không được đảm bảo



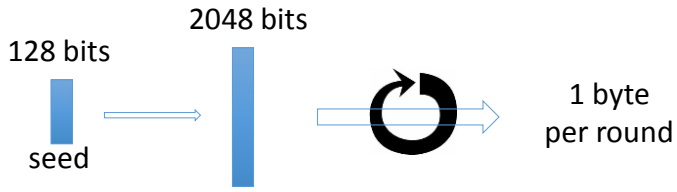
- Việc thay đổi bản mã sẽ làm thay đổi bản rõ khi giải mã.
- Không thể kiểm tra xem bản mã có bị thay đổi hay không.

21

MÃ DÒNG TRÊN THỰC TẾ

Phan Quốc Tín – tinpq@uit.edu.vn

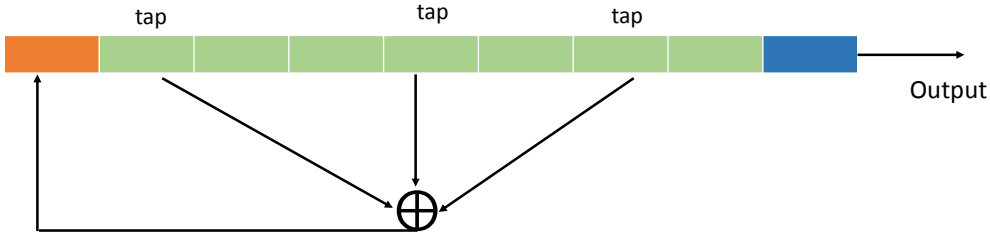
(Software) RC4 (1987)



- Sử dụng trong HTTPS và WEP
- Điểm yếu:
  1. Xác suất không đồng đều với output ban đầu:
     Pr[2nd byte = 0] = 2/256
  1. Xác suất của cặp (0,0) là $1/256^2 + 1/256^3$
  2. Keys có thể bị tấn công

(Hardware): CSS (badly broken)

Linear feedback shift register (LFSR):
Seed = giá trị ban đầu của LFSR



DVD encryption (CSS): 2 LFSRs
GSM encryption (A5/1,2): 3 LFSRs
Bluetooth (E0): 4 LFSRs

} Tất cả đã bị phá

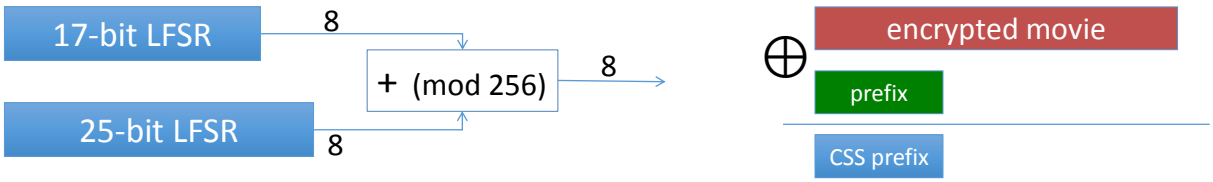
LFSR

- Cho số lượng cells là 4, chuỗi tap là (1001) với trạng thái đầu $s_0 = (0001) = (k_3k_2k_1k_0)$

m	k_{m+3}	k_{m+2}	k_{m+1}	k_m	n	k_{m+3}	k_{m+2}	k_{m+1}	k_m
0	0	0	0	1	8	1	0	1	0
1	1	0	0	0	9	1	1	0	1
2	1	1	0	0	10	0	1	1	0
3	1	1	1	0	11	0	0	1	1
4	1	1	1	1	12	1	0	0	1
5	0	1	1	1	13	0	1	0	0
6	1	0	1	1	14	0	0	1	0
7	0	1	0	1	15	0	0	0	1

- Sau 15 vòng, giá trị output của LFSR là

Thám mã CSS (2^{17} lần)



Vét cạn 2^{17} giá trị có thể có của thanh ghi 17-bit LFSR :

- "Clock out" 17-bit LFSR để thu được 20 bytes giá trị đầu ra
- Tính CSS prefix – 20 bytes ở bước trước ⇒ 20 bytes output của thanh ghi 25-bit LFSR
- Dựa vào 20 bytes output của 25-bit LFSR, suy luận tìm ra toàn bộ trạng thái của nó.
- Tiếp tục "clock out", nếu không khớp với bản mã thì chuyển sang giá trị khác của 17-bit LFSR.

Mã dòng hiện đại: eStream

PRG: $\{0,1\}^s \times R \longrightarrow \{0,1\}^n$

Nonce: Một giá trị không lặp lại đối với cùng một khóa

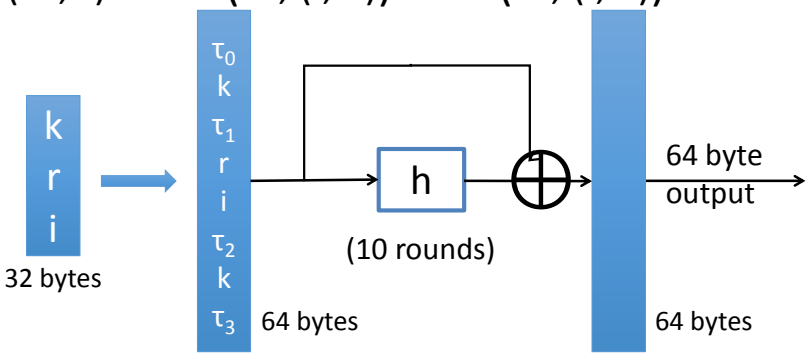
$E(k, m ; r) = m \oplus PRG(k ; r)$

Cặp (k,r) không bao giờ được dùng lại

eStream: Salsa 20 (Software + Hardware)

Salsa20: $\{0,1\}^{128 \text{ or } 256} \times \{0,1\}^{64} \longrightarrow \{0,1\}^n$ (max $n = 2^{73}$ bits)

Salsa20(k ; r) := H(k , (r, 0)) ll H(k , (r, 1)) ll …



h: hàm khả nghịch. Được thiết kết thực hiện nhanh trên x86 (SSE2)

Salsa20 có an toàn không?

- Không biết: chưa có PRG nào được chứng minh là không thể tiên đoán
- Thực tế: chưa có phương pháp tấn công nào tốt hơn vét cạn

Hiệu suất: Crypto++ 5.6.0 [Wei Dai]

http://www.cryptopp.com/

AMD Opteron, 2.2 GHz (Linux)

	PRG	Speed (MB/sec)
	RC4	126
eStream	Salsa20/12	643
eStream	Sosemanuk	727

HẾT CHƯƠNG 2