

Tấn công vét cạn

Trần Vĩnh Đức

HUST

Ngày 6 tháng 4 năm 2020

Hệ mã lý tưởng

Xét hệ mã khối $E(k, x) = y$. Nếu giữ bí mật k , ta xác định hoán vị f như sau:

$$f(x) = E(k, x) = y$$

Nếu hệ mã E là **an toàn** thì f giống như một hoán vị ngẫu nhiên.

Cài đặt hoán vị ngẫu nhiên

Khi nhận truy vấn $x_i \in X$ từ kẻ tấn công \mathcal{A} :

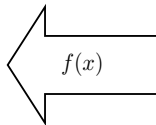
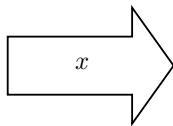
if $x_i == x_j$ với $j < i$

then $y_i = y_j$

else $y_i \leftarrow X \setminus \{y_1, \dots, y_{i-1}\}$

Gửi y_i cho \mathcal{A} .

Hoán vị ngẫu nhiên



x	$f(x)$
00101	10101
11111	01110
10111	01011
00011	10001

Tấn công vét cạn để tìm khóa của mã khối

Kẻ tấn công A biết

$$E: K \times X \rightarrow Y$$

và k là khóa cần tìm.

Bài toán

- ▶ Cho một số cặp input/output $(m_i, c_i = E(k, m_i))$ với $i = 1, 2, \dots, q$.
- ▶ Hãy tìm khóa k .

Câu hỏi

Rõ ràng A có thể biết c_1, \dots, c_q . Nhưng tại sao A lại biết m_1, \dots, m_q ?

- ▶ Do tiết lộ (về sau) của dữ liệu
- ▶ Do kiến thức có từ trước về ngữ cảnh.

Bài toán càng chặt chẽ sẽ càng phù hợp cho nhiều tình huống thực tế!

Tiết lộ dữ liệu

- ▶ S và R chia sẻ khóa chung
- ▶ Vào ngày 10 tháng 1, S mã hóa thông điệp

m = Hẹn gặp ngày mai lúc 5 giờ chiều

và gửi bản mã c đến cho R .

- ▶ Kẻ tấn công lấy được c
- ▶ Vào ngày 11 tháng 1, kẻ tấn công quan sát thấy có cuộc họp giữa S và R lúc 5 giờ chiều và do đó biết thông điệp m .

Biết về ngữ cảnh

- ▶ S và R chia sẻ khóa với nhau.
- ▶ Biết rằng Email luôn bắt đầu bằng From
- ▶ S mã hóa email
- ▶ Kẻ tấn công biết bản mã c
- ▶ Và anh ta biết từ From là một phần của bản rõ.

Kiểu tấn công

Cho $(m_1, c_1), \dots, (m_q, c_q)$ với $c_i = E(k, m_i)$.

Tấn công chọn thông điệp: Kẻ tấn công A có thể lấy m_1, \dots, m_q , một cách thích nghi, tức là chọn m_i như một hàm theo

$$(m_1, c_1), \dots, (m_{i-1}, c_{i-1}).$$

Alice

Kẻ tấn công

m_1
←

$c_1 = E(k, m_1)$
→

m_2
←

$c_2 = E(k, m_2)$
→

.....

Tấn công vét cạn để tìm khóa

tấn-công-vét-cạn (m_1, c_1)

for $k = 1, 2, \dots, 2^{|K|}$

if $E(k, m_1) == c_1$ then return k

Câu hỏi: Thuật toán trên liệu có giúp tìm khóa k đúng?

Bổ đề

Giả sử DES là một hệ mã lý tưởng

(2^{56} hoán vị ngẫu nhiên $\pi_i : \{0, 1\}^{64} \rightarrow \{0, 1\}^{64}$)

Vậy thì với mỗi cặp x, y có nhiều nhất **một** khóa k thỏa mãn

$$y = DES(k, x)$$

với xác suất $\geq 1 - 1/256 \approx 99.5\%$.

$$\begin{aligned} & \Pr[\exists k' \neq k \text{ thỏa mãn } c = DES(k, m) = DES(k', m)] \\ & \leq \sum_{k' \in \{0,1\}^{56}} \Pr[DES(k, m) = DES(k', m)] \\ & \leq 2^{56} \cdot \frac{1}{2^{64}} = \frac{1}{2^8}. \end{aligned}$$

Tìm kiếm vét cạn để tìm khóa cho mã khối

- ▶ Với hai cặp DES $(m_1, c_1 = DES(k, m_1))$ và $(m_2, c_2 = DES(k, m_2))$
xác suất để có k có duy nhất là $\approx 1 - 1/2^{71}$.
- ▶ Với AES-128: cho hai cặp input/output, xác suất có k duy nhất $\approx 1 - 1/2^{128}$
- ▶ Với hai cặp input/output là đủ thông tin để tìm kiếm vét cạn cho khóa.

Thử thách DES

Cho các cặp bản rõ và bản mã

msg =	"The unknown messages is : XXXX ... "			
CT =	c1	c2	c3	c4

Hãy tìm khóa $k \in \{0,1\}^{56}$ thỏa mãn $DES(k, m_i) = c_i$ với $i = 1, 2, 3$.

- ▶ 1997: DESCHALL project với internet search – 96 ngày
- ▶ 1998: EFF dùng máy DeepCrack – 3 ngày (250K \$)
- ▶ 1999: Kết hợp cả DeepCrack và internet search – 22 giờ
- ▶ 2006: COPACOBANA (120 FPGA) – 7 ngày (10K \$).

Không nên dùng mã khối 56 bit khóa !! (128-bit khóa $\Rightarrow 2^{72}$ ngày)

Cải tiến DES chống tấn công vét cạn: Triple-DES

Phương pháp 1: Triple-DES

- ▶ Xét $E: K \times X \rightarrow X$ là một hệ mã khối.
- ▶ Ta định nghĩa hệ mã khối

$$3E: K^3 \times X \rightarrow X$$

bởi:

$$3E((k_1, k_2, k_3), m) := E(k_3, D(k_2, E(k_1, m)))$$

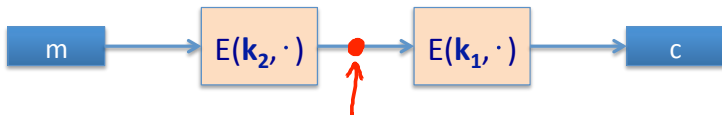
Nhận xét

Nếu $k_1 = k_2 = k_3$ thì $3E = E$

Triple-DES: Một số nhận xét

- ▶ Kích thước khóa là $3 \times 56 = 168$ bit,
- ▶ Chậm gấp 3 lần DES.
- ▶ Có thể tấn công trong thời gian $\approx 2^{118}$.

Tại sao không dùng double-DES?



Hình: $2E((k_1, k_2), m) := E(k_1, E(k_2, m))$

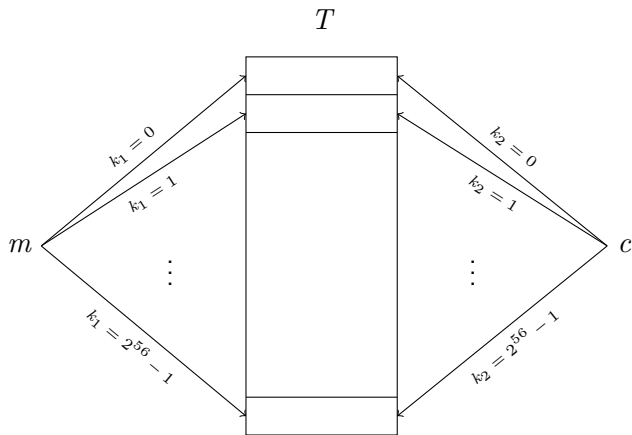
Ý tưởng tấn công double-DES: Tìm (k_1, k_2) thỏa mãn

$$E(k_1, E(k_2, m)) = c$$

tương đương với

$$E(k_2, m) = D(k_1, c).$$

Ý tưởng tấn công



Hình: $E(k_1, E(k_2, m)) = c \Leftrightarrow E(k_2, m) = D(k_1, c).$

Thuật toán

1. Xây dựng bảng

2^{56}

$k_0 = 00 \dots 0$	$E(k_0, m)$
$k_1 = 00 \dots 1$	$E(k_1, m)$
\dots	
$k_N = 11 \dots 1$	$E(k_N, m)$

2. Sắp xếp các phần tử của bảng theo cột thứ hai $E(k, m)$

56×2^{56}

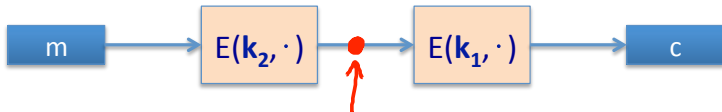
3. for $k \in \{0, 1\}^{56}$:

2^{56}

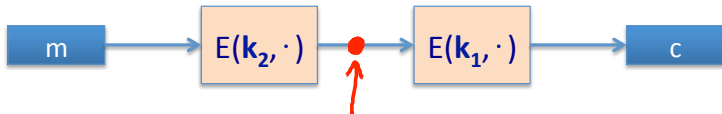
kiểm tra liệu $D(k, c)$ có nằm trong cột thứ hai của bảng

$\times 56$

nếu có thì $E(k_i, m) = D(k, c) \Rightarrow (k_i, k) = (k_1, k_2)$.



Chi phí tính toán



$$\text{Thời gian} = \underbrace{\text{Xây dựng bảng và sắp xếp}}_{56 \times 2^{56}} + \underbrace{\text{Tìm kiếm nhị phân trong bảng}}_{56 \times 2^{56}} \\ < 2^{63}$$

$$\text{Không gian} \approx 2^{56}$$

Phương pháp 2: DESX

- ▶ Xét hệ mã khối $E : K \times \{0, 1\}^n \rightarrow \{0, 1\}^n$
- ▶ Định nghĩa EX bởi

$$EX((k_1, k_2, k_3), m) = k_1 \oplus E(k_2, m \oplus k_3)$$

- ▶ Với DESX: Độ dài khóa = $64 + 56 + 64 = 184$ bit

Bài tập

Liệu cách xây dựng $E2((k_1, k_2), m)$ như dưới đây có hiệu quả hay không?

- ▶ $k_1 \oplus E(k_2, m)$
- ▶ $E(k_2, m \oplus k_1)$

Giới hạn của tính an toàn chống lại vét cạn khóa

Xét hệ mã

$$E : \{0, 1\}^{128} \times \{0, 1\}^{128} \rightarrow \{0, 1\}^{128}$$

định nghĩa bởi

$$E(k, m) = m.$$

- ▶ Không thể tìm khóa nếu cho một số cặp bản rõ/bản mã.
- ▶ Nhưng hệ này là không an toàn!

Thế nào là hệ mã khối “tốt”?

Tính chất	Cần?	Đủ?
an toàn chống lại vét cạn khóa	Có	Không!
khó tìm m khi cho $c = E(k, m)$	Có	Không!
\vdots		

Ta không thể nào định nghĩa hoặc hiểu tính an toàn nếu đưa ra một danh sách không xác định

Ta muốn một tính chất “chủ đạo” của hệ mã khối để đảm bảo an toàn cho việc sử dụng mã khối!