Ứng dụng

Mã hóa hệ thống file

- Mã hóa nhiều file dùng AES với cùng khóa

IPSec

- Nhiều gói tin cùng được mã hóa bằng AES với cùng một khóa

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

Nhập môn An toàn thông tin

An toàn ngữ nghĩa cho khoá dùng nhiều lần

An toàn ngữ nghĩa cho khóa dùng nhiều lần

Khóa được dùng nhiều lần

- Kẻ tấn công thấy nhiều bản rõ được mã hóa bởi cùng một khóa

Khả năng của kẻ tấn công

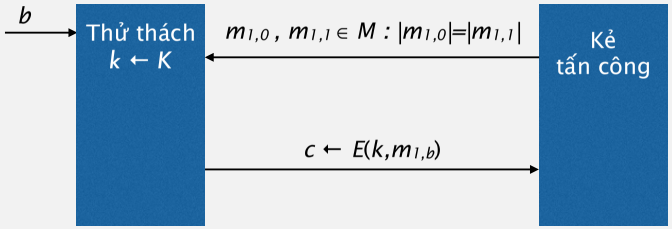
- Tấn công chọn bản rõ = chosen-plaintext attack (CPA)
  - Có thể lấy được mã hóa của một số thông điệp mà anh ta muốn
- Đây là mô hình thực tế

Mục đích của kẻ tấn công

- Phá được an toàn ngữ nghĩa

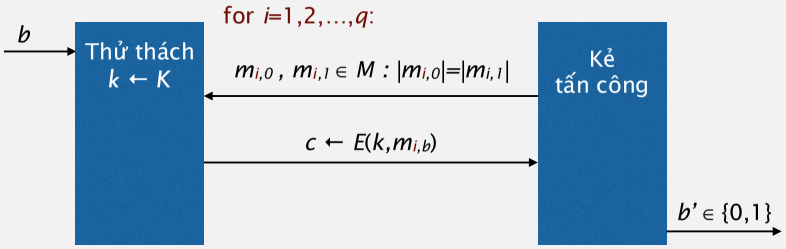
An toàn ngữ nghĩa cho khóa dùng nhiều lần

Xét $E = (E,D)$ là một hệ mà trên (K,M,C). Với b =0,1 ta định nghĩa EXP(b) như sau:



An toàn ngữ nghĩa cho khóa dùng nhiều lần

Xét $E = (E,D)$ là một hệ mã trên (K,M,C). Với b =0,1 ta định nghĩa EXP(b) như sau:



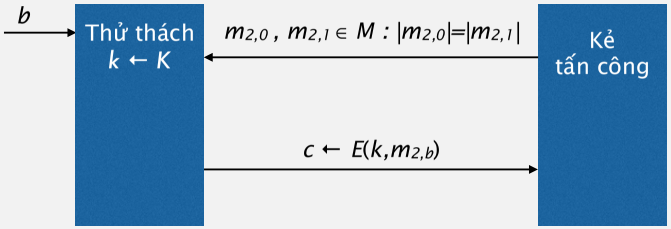
Nếu kẻ tấn công muốn $c \leftarrow E(k,m)$ anh ta có thể gửi $m_{j,0} = m_{j,1} = m$

Định nghĩa. E là *an toàn ngữ nghĩa* dưới CPA nếu với mọi thuật toán "hiệu quả" A:

$Adv_{CPA}[A,E] = |Pr[EXP(0)=1] - Pr[EXP(1)=1]|$ là "không đáng kể"

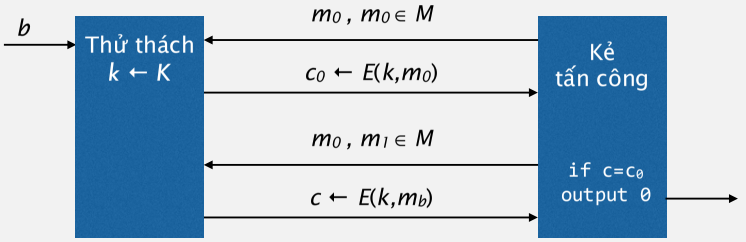
An toàn ngữ nghĩa cho khóa dùng nhiều lần

Xét $E = (E,D)$ là một hệ mà trên (K,M,C). Với b =0,1 ta định nghĩa EXP(b) như sau:



Mã hóa không an toàn dưới CPA

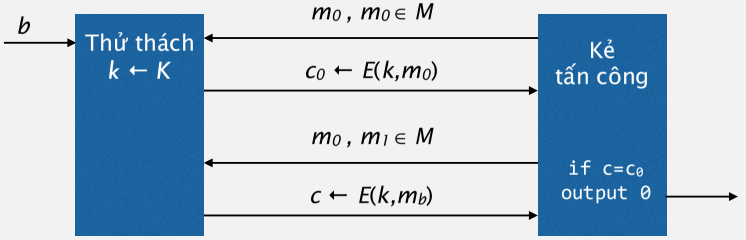
Giả sử E(k,m) luôn cho cùng một bản mã cho thông điệp m. Vậy thì



Nếu một khóa được dùng lại nhiều lần, vậy thì để an toàn, mã hóa cùng một bản mã hai lần khác nhau sẽ phải cho ra hai kết quả khác nhau.

Mã hóa không an toàn dưới CPA

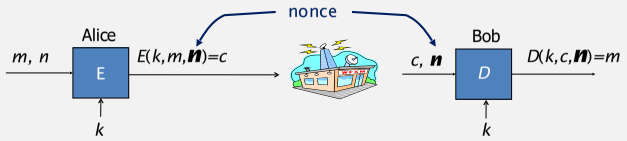
Giả sử E(k,m) luôn cho cùng một bản mã cho thông điệp m. Vậy thì



Kẻ tấn công có thể kiểm tra được hai bản mã có phải là của cùng một bản rõ.

Có thể tấn công hệ mã khi không gian thông điệp M nhỏ.

Giải pháp 2: Mã hóa dựa trên nonce



Nonce *n:* một giá trị thay đổi theo các thông điệp

- Cặp (k,n) sẽ được dùng tối đa một lần

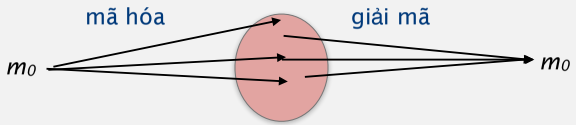
Phương pháp 1: nonce là một bộ đếm (ví dụ: đếm số gói tin)

- được dùng khi bộ mã hóa giữ trạng thái thay đổi theo thông điệp
- nếu bộ giải mã có cùng trạng thái, không cần gửi nonce cùng với bản mã

Phương pháp 2: bộ mã hóa chọn nonce ngẫu nhiên

Giải pháp 1: Mã hóa xác suất

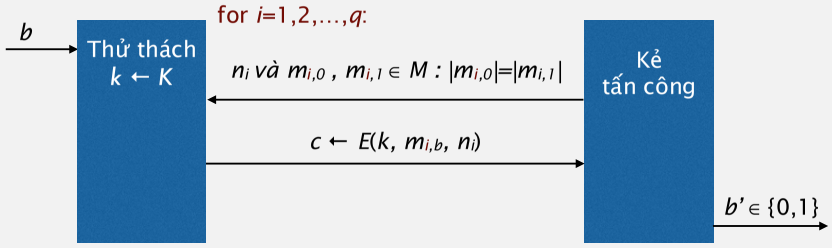
E(k,m) là thuật toán ngẫu nhiên



- Mã hóa hai lần của cùng một thông điệp sẽ cho hai bản mã khác nhau
- Bản mã phải dài hơn bản rõ
- Nói một cách nôm na:
  - Kích thước bản mã = Kích thước bản rõ + "số bit ngẫu nhiên"

An toàn ngữ nghĩa cho các hệ mã dựa trên nonce

Hệ mã vẫn phải an toàn khi nonce được chọn bởi kẻ tấn công



Các nonce $\{n_1,\ldots,n_q\}$ phải phân biệt

Định nghĩa. Hệ mã dựa trên nonce *E* là *an toàn ngữ nghĩa* dưới CPA nếu với mọi thuật toán "hiệu quả" *A:*

$$Adv_{nCPA}[A,E] = |Pr[EXP(0)=1] - Pr[EXP(1)=1]|$$

là "không đáng kể"

Câu hỏi

Xét PRF an toàn $F : K \times R \rightarrow M$. Ban đầu ta đặt $r = 0$.

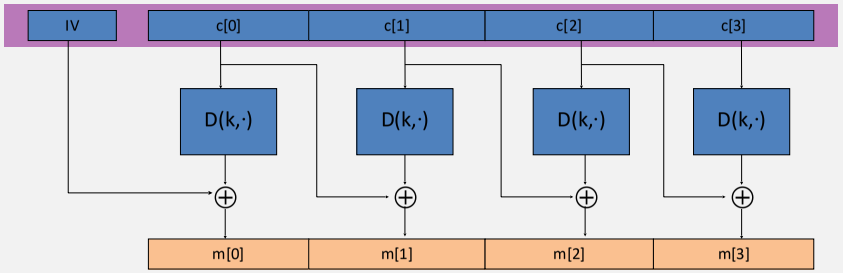
Với $m \in M$ ta định nghĩa

$$E(k,m) = [r++,\ \text{output}\ (r, F(k,r) \oplus m)]$$

Hệ mã *E* dựa trên nonce có phải an toàn ngữ nghĩa dưới CPA không?

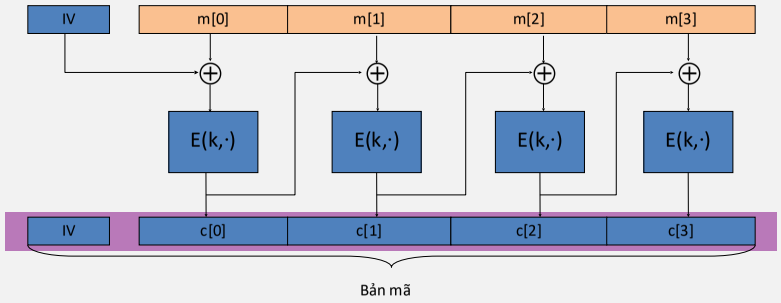
- Có, bất cứ khi nào *F* là PRF an toàn.
- Không, luôn có một cách tấn công CPA dựa trên nonce trên hệ này.
- Có, nhưng chỉ khi *R* đủ lớn sao cho *r* không bao giờ lặp lại.
- Phụ thuộc vào hàm *F* nào được dùng.

Mạch giải mã



Xây dựng 1: CBC với IV ngẫu nhiên

Xét (E,D) là một PRP. Chọn ngẫu nhiên IV ∈ X và tính toán theo sơ đồ sau:

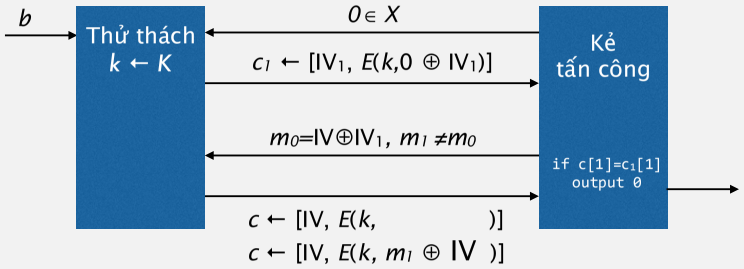


Bài tập. Hãy xây dựng mạch giải mã.

Chú ý: Tấn công CBC với IV ngẫu nhiên

Khi kẻ tấn công có thể dự đoán IV, vậy CBC không là CPA-an toàn !

Giả sử rằng biết $c \leftarrow E(k,m)$ ta có thể dự đoán IV cho thông điệp tiếp theo.



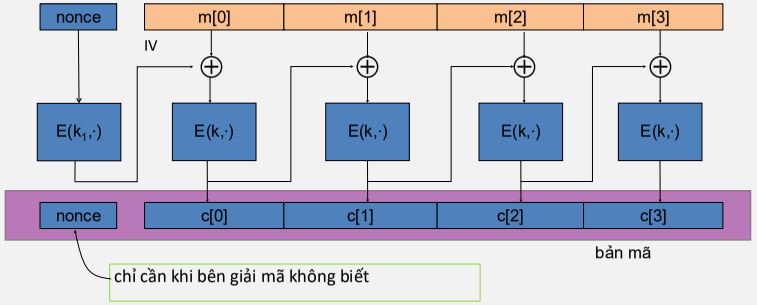
Lỗi trong SSL/TLS 1.0:

- IV cho bản ghi thứ i là block cuối của bản mã của bản ghi thứ i-1.

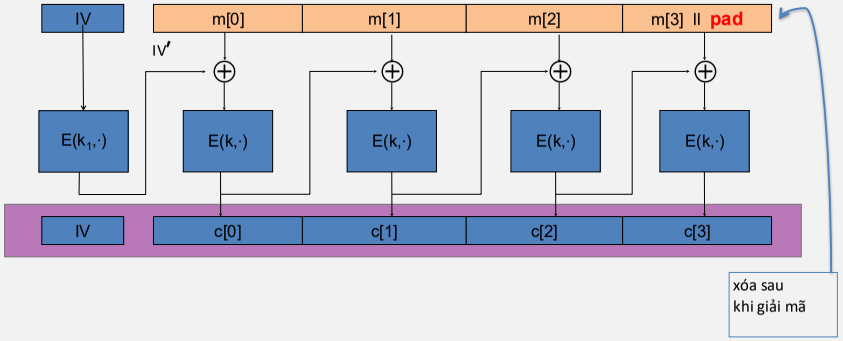
Xây dựng 1': CBC dựa trên nonce

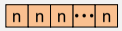
CBC với nonce duy nhất : key = (k,k_1)

- Nonce duy nhất: cặp (key,n) dùng cho chỉ một thông điệp



Một kỹ thuật padding cho CBC



TLS: với n>0, n byte pad là 

nếu không cần pad, ta thêm một block giả

Ví dụ về Crypto API

OpenSSL
Cryptography and SSL/TLS Toolkit

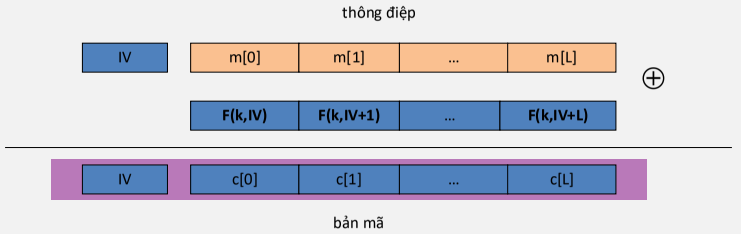
```
void AES_cbc_encrypt (
    const unsigned char * in,
    unsigned char *out,
    size_t length,
    const AES_KEY *key,
    unsigned char *ivec,
    AES_ENCRYPT  or AES_DECRYPT);
```

Chú ý. Nếu **ivec** không lấy ngẫu nhiên thì ta phải mã hóa nó trước khi dùng.

Xây dựng 2: Rand CTR-mode

Xét một hệ mã khối an toàn $F : K \times \{0,1\}^n \rightarrow \{0,1\}^n$

$E(k,m)$ được định nghĩa như sau: Chọn IV ngẫu nhiên và tính:



Chú ý. Khác với CBC-mode, CTR-mode cho phép song song hóa hiệu quả.

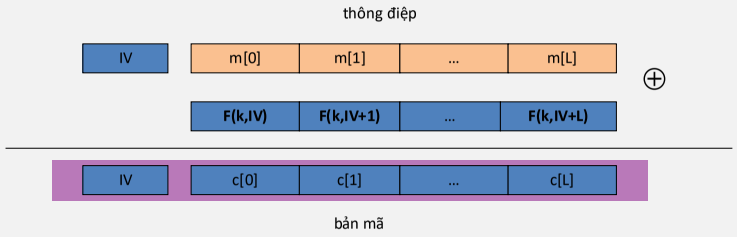
Bài tập. Xây dựng mạch giải mã.

https://class.coursera.org/crypto-preview/class/index

SỬ DỤNG MÃ KHỐI

- PRP và PRF an toàn
- Định nghĩa an toàn cho khóa dùng nhiều lần
- CBC mode
- Rand CTR mode

Xây dựng 2': nonce CTR-mode



Để đảm bảo rằng $F(k,x)$ không dùng khóa quá một lần, chọn IV như sau: