Mật mã khối và mật mã khóa đối xứng

Văn Nguyễn

Đại học Bách Khoa Hà nội

9/13/2008

Khái niệm mã khối

- So sánh với mã đã học: stream cipher vs. block cipher

key	000	001	010	011	100	101	110	111
0	001	111	110	000	100	010	101	011
1	001	110	111	100	011	010	000	101
2	001	000	100	101	110	111	010	011
3	100	101	110	111	000	001	010	011
4	101	110	100	010	011	001	011	111

- TIN= 010100110111= (010)(100)(110)(111)
  - ➔ MÃ= 111 011 000 101 theo key=1
  - ➔ MÃ= 100 011 011 111 theo key=4
- Có 5 khóa, $2^2 < 5 < 2^3$ nên cần 3 bit để biểu diễn ➔ kich thước khóa (và kích thước khối cùng) là 3.
- Nếu Eve tóm đc khối MÃ=001 sẽ suy ra TIN là 000 hoặc 1

Điều kiện cho an toàn block ciphers

- 1. Kích thước khối phải đủ lớn để chống lại các loại tấn công phá hoại bằng phương pháp thống kê.
  - Tuy nhiên cần lưu ý rằng kích thước khối lớn sẽ làm thời gian trễ lớn.
- 2. Không gian khóa phải đủ lớn (tức là chiều dài khóa phải đủ lớn) để chống lại tìm kiếm vét cạn.
  - Tuy nhiên mặt khác, khóa cần phải đủ ngắn để việc làm khóa, phân phối và lưu trữ được hiệu quả.

Nguyên tắc thiết kế cho block ciphers

- *Confusion.* (Hỗn loạn) Sự phụ thuộc của MÃ đối với TIN phải thật phức tạp để gây rắc rối hỗn loạn đối với kẻ thù có ý định tìm qui luật để phá mã.
  - Quan hệ hàm số của Mã với TIN nên là phi tuyến (non-linear).
- *Diffusion.* (khuếch tán) Làm khuếch tán những mẫu văn bản mang đặc tính thống kê (gây ra do dư thừa của ngôn ngữ) lẫn vào toàn bộ văn bản.
  - Nhờ đó tạo ra khó khăn cho kẻ thù trong việc dò phá mã trên cơ sở thống kê các mẫu lặp lại cao.
- Trong khi *confusion* được thực hiện bằng phép thay thế (substitution) thì *diffusion* được tạo ra bằng các phép chuyển đổi chỗ (tranposition) hay hoán vị.

Ví du: Phép hoán vị cột

- Để mã hóa TIN="computer security", viết lại thành nhiều hàng 5 cột

c o m p u

t e r s e

c u r i t

y.

- MÃ tạo ra bằng cách viết lại theo cột:

C T C Y O E U M R R P S I U E T

Cài đặt

- Software: mềm dẻo, giá thành thấp.
- Hardware: nhanh.

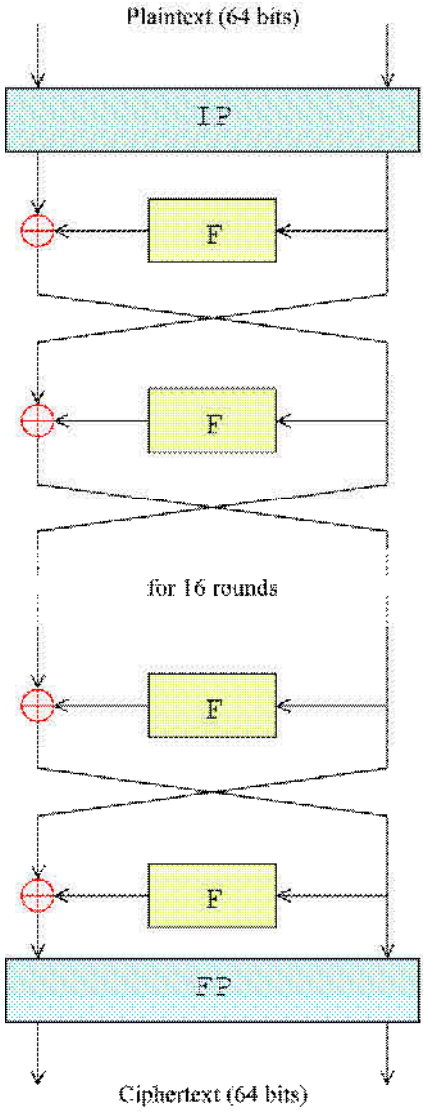
- Study case:

Data Encryption Standard (DES) - 1977

  - Hiện tại đã có chuẩn mới AES ➔ topic bài tập lớn

Khái niệm vòng lặp

- Các mã khối thường được xây dựng nhiều vòng lặp với mỗi vòng lặp cơ sở = việc thực hiện một hàm f.
  - đầu vào của một vòng lặp là đầu ra của vòng lặp trước và một khóa con phát sinh từ khóa đầy đủ dựa trên một thuật toán key-schedule.
- Giải mã sẽ là một quá trình ngược với các khóa con cho mỗi vòng sẽ được phát sinh theo thứ tự ngược.



The overall Feistel struct

Involution (đối hợp)

- Đặc biệt, hàm cơ sở vòng lặp f thông thường là một hàm có đặc tính đối hợp (involution), tức là nó bằng hàm ngược của nó: $f = f^{-1}$ hay là f(f(x)) = x
  - Ví dụ:

  $x \in$ {tập các chuỗi nhị phân độ dài 3}

  (bit thứ nhất và thứ hai đổi chỗ cho nhau, bit thứ ba giữ nguyên).
  - Như thế ta có f là một hàm xoáy ốc, chẳng hạn cụ thể là

  f(101 = 011)

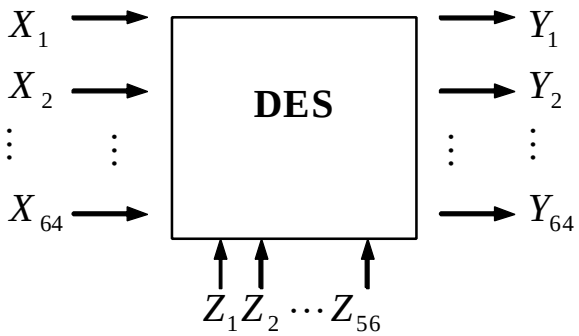
  f(f(101)) = 101

Lịch sử của DES

- Vào những năm đầu thập kỷ 70, nhu cầu có một chuẩn chung về thuật toán mã hóa đã trở nên rõ ràng:
  1. Sự phát triển của công nghệ thông tin và của nhu cầu an toàn & bảo mật thông tin.
  2. Các thuật toán ‘cây nhà lá vườn’ (ad hoc) không thể đảm bảo được tính tin cậy đòi hỏi.
  3. Các thiết bị khác nhau đòi hỏi sự trao đổi thông tin mã hóa.

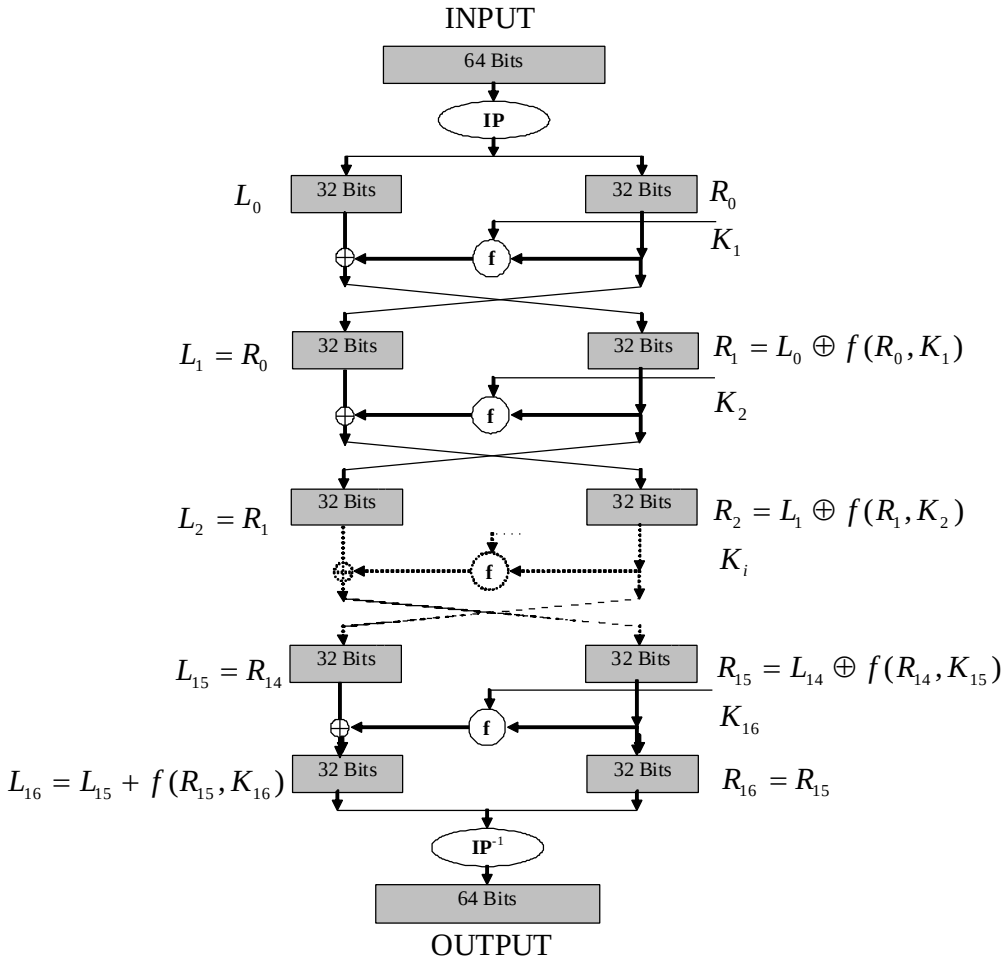
- Một chuẩn chung cần thiết phải có với các thuộc tính như:
  1. Bảo mật ở mức cao
  2. Thuật toán được đặc tả và công khai hoàn toàn, tức là tính bảo mật không được phép dựa trên những phần che giấu đặc biệt của thuật toán.
  3. Việc cài đặt phải dễ dang để đem lại tính kinh tế
  4. Phải mềm dẻo để áp dụng được cho muôn vàn nhu cầu ứng dụng

- Năm 1973, Cục quản lý các chuẩn quốc gia của Mỹ đã có văn bản cổ động cho các hệ thống mã hóa ở cơ quan đăng ký liên bang của Mỹ. Điều đó cuối cùng đã dẫn đến sự phát triển của Data Encryption Standard, viết tắt là DES.
  - DES, IBM, Lucipher.
  - dùng rộng rãi nhất, tranh cãi nhiều nhất
- Sơ đồ chung



- Đầu vào là khối độ dài 64 bits, đầu ra 64 bits và khóa là 56

Sơ đồ cấu trúc: 16 vòng lặp của DES

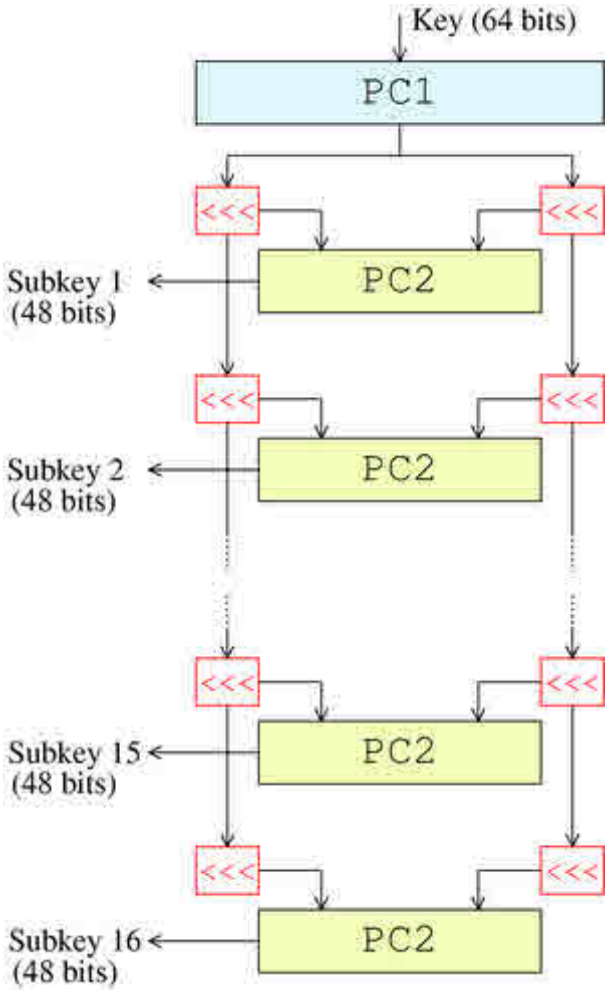


Van K Nguyen --Dai hoc Bach khoa Ha noi

- DES được cấu tạo bởi 16 bước lặp có cơ sở là hàm chuyển đổi phi tuyến f;
- 16 bước lặp này được kẹp vào giữa hai tác tử giao hoán IP và IP-1.
  - Hai tác từ này không có ý nghĩa gì về mặt mật mã mà hoàn toàn nhằm tạo điều kiện cho việc 'chip hóa' thuật toán DES.
- Hàm f là nguồn gốc của sức mạnh trong thuật toán DES này.
  - Sự lặp lại nhiều lần các bước lặp với tác dụng của f là nhằm tăng cường thêm mãnh lực của f về mặt lượng.

Thuật toán sinh khóa con

- 16 vòng lặp của DES chạy cùng thuật toán như nhau nhưng với các khóa khác nhau, được gọi là các khóa con
  - sinh ra từ khóa chính của DES bằng một thuật toán sinh khóa con.
- Khóa chính K, 64 bit, qua 16 bước biến đổi, mỗi bước sinh 1 khóa con 48 bit.
- Thực sự chỉ có 56 bit của khóa chính được sử dụng
  - 8 parity bits, lọc ra qua PC1.
  - Các bộ biến đổi PC1 và PC2 là các bộ vừa chọn lọc vừa hoán vị.
  - R1 và R2 là các phép đẩy bit trái 1 và hai vị trí.



Cấu trúc vòng lặp DES

- Mỗi vòng lặp của DES thực hiện trên cơ sở công thức sau:
- $(L_i,R_i) = (R_{i-1}, L_{i-1} (F(R_{i-1},K_i))$
- Ta cũng có thể viết lại

$$(L_i,R_i) = T \bullet F (L_{i-1}, R_{i-1})$$

  - Trong đó F là phép thay thế L_{i-1} bằng $L_{i-1} \oplus F(R_{i-1},K_i)$
  - T là phép đổi chỗ hai thành phần L và R.
  - Tức là mỗi biến đổi vòng lặp của DES có thể coi là một tích hàm số của F và T (trừ vòng cuối cùng không có T).
- Viết lại toàn bộ **thuật toán sinh mã DES** dưới dạng công thức:

$$DES = (IP)^{-1} \bullet F_{16} \bullet T \bullet F_{15} \bullet T \bullet \ ... \ \bullet F_2 \bullet T \bullet F_1 \bullet (IP)$$

Thuật toán giải mã DES

- Giống hệt như thuật toán sinh mã nhưng có các khóa con được sử dụng theo thứ tự ngược lại
  - Vì vậy, thuật toán giải mã có thể được viết lại dưới dạng công thức sau:

$$DES^{-1} = (IP)^{-1} \bullet F_1 \bullet T \bullet F_2 \bullet T \bullet \ldots \bullet F_{15} \bullet T \bullet F_{16} \bullet (IP)$$

- Chú ý rằng mỗi hàm T hoặc F đều là các hàm có tính chất đối hợp ($f=f^{-1}$, hay $f(f(x)=x$) ➔ thực hiện $DES \bullet DES^{-1}$ sẽ thu được phép đồng nhất.
  - Điều đó giải thích tại sao thuật toán giải mã lại giống hệt như sinh mã chỉ có khác thự từ dùng khóa con.

Cấu trúc cụ thể hàm f

- 32 bit của R_{i-1} được mở rộng thành 48 bit thông qua E rồi đem XOR với 48 bit của K_i.
- 48 bit kết quả sẽ được phân thành 8 nhóm 6 bit; mỗi nhóm này sẽ qua một biến đổi đặc biệt, S-box, và biến thành 4 bit.
  - có 8 S-box khác nhau ứng với mỗi nhóm 6 bit
- 32 bit hợp thành từ 8 nhóm 4 bit (sau khi qua các S-box) sẽ được hoán vị lại theo P rồi đưa ra kết quả cuối cùng của hàm f (Fi).

Cấu trúc của các S-Box

- Mỗi S-box như một bộ biến đổi gồm 4 bảng biến đổi, mỗi bảng biến đổi 1 đầu vào 4 bit thành đầu ra cũng 4 bit (bảng 16 dòng).
  - Đầu vào 4 bit chính là lấy từ các bit 2-5 của nhóm 6 bit.
  - Các bit 1 và 6 sẽ dùng để xác định 1 trong 4 bảng biến đổi của S-box. Vì thế chúng được gọi là các bit điều khiển (CL và CR: left control và right control bit).

S_5		Middle 4 bits of input															
		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
Outer bits	00	0010	1100	0100	0001	0111	1010	1011	0110	1000	0101	0011	1111	1101	0000	1110	1001
	01	1110	1011	0010	1100	0100	0111	1101	0001	0101	0000	1111	1010	0011	1001	1000	0110
	10	0100	0010	0001	1011	1010	1101	0111	1000	1111	1001	1100	0101	0110	0011	0000	1110
	11	1011	1000	1100	0111	0001	1110	0010	1101	0110	1111	0000	1001	1010	0100	01	

Các thuộc tính của S-Box

- Các nguyên tắc thiết kế của 8 S-boxes được đưa vào lớp 'Classified information' ở Mỹ.
- NSA đã tiết lộ 3 thuộc tính của S-boxes, những thuộc tính này bảo đảm tính confusion & diffusion của thuật toán.
  1. Các bít vào (output bit) luôn phụ thuộc không tuyến tính vào các bít ra (input bit).
  2. Sửa đổi ở một bit vào làm thay đổi ít nhất là hai bit ra.
  3. Khi một bit vào được giữ cố định và 5 bit con lại cho thay đổi thì S-boxes thể hiện một tính chất được gọi là 'phân bố đồng nhất ' (uniform distribution): so sánh số lượng bit số 0 và 1 ở các đầu ra luôn ở mức cân bằng.
     - Tính chất này khiến cho việc áp dụng phân tích theo lý thuyết thông kê để tìm cách phá S-boxes là vô ích.

- 3 tính chất này đảm bảo tốt confusion & diffusion.
  - Sau 8 vòng lặp tất cả các bit ra của DES sẽ chịu ảnh hưởng của tất cả các bit vào và tất cả các bit của khóa.
- Tuy nhiên cấu tạo của S-box đã gây tranh luận mạnh mẽ từ hàng thập kỷ qua về khả năng cơ quan NSA (National Security Agency), Mỹ, vẫn còn che dấu các một số đặc tính của S-box hay cài bên trong những cửa bẫy (trapdoor) mà qua đó họ có thể dễ dàng phá giải mã hơn người bình thường.

Các điểm yếu của DES

- Tính bù

Ký hiệu $\bar{u}$ là phần bù của u (e.g. 0100101 và 1011010 là bù của nhau) thì DES có tính chất sau:

$$y = DES_z(x) \Rightarrow \bar{y} = DES_{\bar{z}}(\bar{x})$$

Cho nên nếu biết MÃ y được mã hóa từ TIN x với khóa z thì ta suy ra $\bar{y}$ được mã hóa từ TIN $\bar{x}$ với khóa $\bar{z}$.

  - Tính chất này chính là một điểm yếu của DES bởi vì nhờ đó kẻ địch có thể loại trừ một nửa số khóa cần phải thử khi tiến hành phép thử-giải mã theo kiểu vét cạn (tiếp)

Khóa yếu

- Các khóa yếu là các khóa mà theo thuật toán KS sinh khóa con thì tất cả 16 khóa con đều như nhau

$$Z_1 = Z_2 = Z_3 = ... = Z_{15} = Z_{16}$$

điều đó khiến cho phép sinh mã và giải mã đối với các khóa yếu này là giống hệt nhau

$$DES_z = DES^{-1}_z$$

- Có tất cả 4 khóa yếu như sau:

1) [00000001 00000001 ... ... 00000001]

2) [11111110 11111110 ... ... 11111110]

3) [11100000 11100000 11100000 11100000 11110001 11110001 11110001 11110001]

4) [00011111 00011111 00011111 00011111 00001110 00001110 00001110 00001110]

Tấn công bằng phương pháp vét cạn (brute-force attack)

- DES có $2^{56}=10^{17}$ khóa. Nếu như biết một cặp TIN/Mã thì chúng ta có thể thử tất cả 10^{17} khả năng này để tìm ra khóa cho kết quả khớp.
  - Giả sử như một phép thử mất quãng 10^{-6}s, thì chúng ta sẽ thử mất 10^{11}s tức là 7300 năm!
  - Xử lý song song: một thiết bị với 10^{7} con chip mật mã DES chạy song song,mỗi con chip chỉ thực hiện 10^{10} phép thử.
  - Chip mã DES ngày nay có thể xử lý tới tốc độ là 4.5 x 10^{7}bits/s tức là có thể làm được hơn 105 phép mã DES trong một giây.
- Diffie và Hellman (1977) ước lượng: máy tính chuyên dụng vét cạn không gian khóa DES trong1/2 ngày với giá 20 triệu đô la. Giảm xuống $200,000 vào năm 1987.
  - Thực tế DES có thể bị phá trong vòng mấy chục giờ, với giá chỉ $10,000