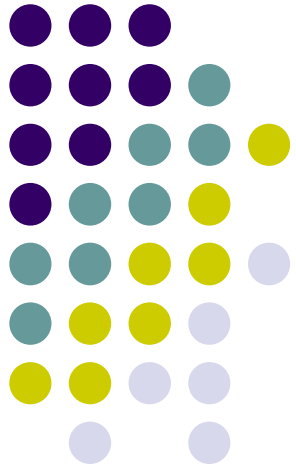
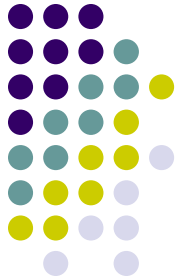
AN TOÀN VÀ BẢO MẬT THÔNG TIN

GVTH: ThS. Trần Phương Nhung

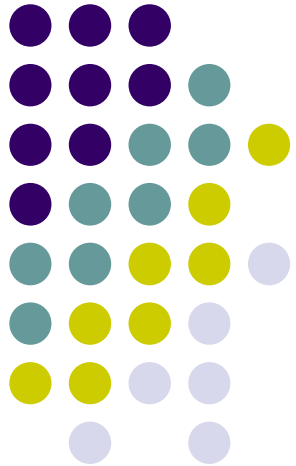


Nội dung



- Chương 1: Tổng quan về an toàn và bảo mật thông tin.
- Chương 2: Các phương pháp mã hóa cổ điển
- Chương 3: Chuẩn mã dữ liệu DES
- Chương 4: Mật mã công khai
- Chương 5: Các sơ đồ chữ ký số
- Chương 6: Hàm băm

Chương 1: Tổng quan về an toàn và bảo mật thông tin.



1. Tại sao phải bảo vệ thông tin

- Thông tin là một bộ phần quan trọng và là tài sản thuộc quyền sở hữu của các tổ chức
- Sự thiệt hại và lạm dụng thông tin không chỉ ảnh hưởng đến người sử dụng hoặc các ứng dụng mà nó còn gây ra các hậu quả tai hại cho toàn bộ tổ chức đó
- Thêm vào đó sự ra đời của Internet đã giúp cho việc truy cập thông tin ngày càng trở nên dễ dàng hơn

2. Khái niệm hệ thống và tài sản của hệ thống

- **Khái niệm hệ thống** :Hệ thống là một tập hợp các máy tính bao gồm các thành phần, phần cứng, phần mềm và dữ liệu làm việc được tích luỹ qua thời gian.
- **Tài sản của hệ thống bao gồm:**
  - ✓ Phần cứng
  - ✓ Phần mềm
  - ✓ Dữ liệu
  - ✓ Các truyền thông giữa các máy tính của hệ thống
  - ✓ Môi trường làm việc
  - ✓ Con người

3. Các mối đe doạ đối với một hệ thống và các biện pháp ngăn chặn

- **Có 3 hình thức chủ yếu đe dọa đối với hệ thống:**
  - ✓ Phá hoại: kẻ thù phá hỏng thiết bị phần cứng hoặc phần mềm hoạt động trên hệ thống.
  - ✓ Sửa đổi: Tài sản của hệ thống bị sửa đổi trái phép. Điều này thường làm cho hệ thống không làm đúng chức năng của nó. Chẳng hạn như thay đổi mật khẩu, quyền ngợời dùng trong hệ thống làm họ không thể truy cập vào hệ thống để làm việc.
  - ✓ Can thiệp: Tài sản bị truy cập bởi những ngươời không có thẩm quyền. Các truyền thông thực hiện trên hệ thống bị ngăn chặn, sửa đổi.

3. Các mối đe doạ đối với một hệ thống và các biện pháp ngăn chặn

- **Các đe dọa đối với một hệ thống thông tin có thể đến từ ba loại đối tượng như sau:**
  - ✓ Các đối tượng từ ngay bên trong hệ thống (insider), đây là những người có quyền truy cập hợp pháp đối với hệ thống.
  - ✓ Những đối tượng bên ngoài hệ thống (hacker, cracker), thường các đối tượng này tấn công qua những đường kết nối với hệ thống như Internet chẳng hạn.
  - ✓ Các phần mềm (chẳng hạn nhơ spyware, adware …) chạy trên hệ thống.

3. Các mối đe doạ đối với một hệ thống và các biện pháp ngăn chặn

Kiểm soát truy nhập

Chứng thực

Chống chối bỏ

Bảo mật số liệu

An toàn luồng tin

Nguyên vẹn số liệu

Khả dụng

Riêng tư

Nguy cơ

Phá hủy

Sửa đổi

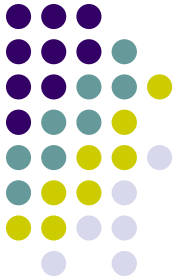
Cắt bỏ

Bóc, tiết lộ

Gián đoạn

Tấn công

3. Các mối đe doạ đối với một hệ thống và các biện pháp ngăn chặn



- **Các biện pháp ngăn chặn:**
  - ✓ **Điều khiển thông qua phần mềm**: dựa vào các cơ chế an toàn bảo mật của hệ thống nền (hệ điều hành), các thuật toán mật mã học
  - ✓ **Điều khiển thông qua phần cứng**: các cơ chế bảo mật, các thuật toán mật mã học được cứng hóa để sử dụng
  - ✓ **Điều khiển thông qua các chính sách của tổ chức:** ban hành các qui định của tổ chức nhằm đảm bảo tính an toàn bảo mật của hệ thống.

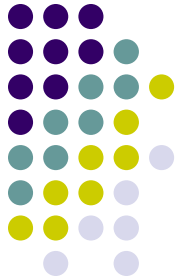
4. Mục tiêu chung của an toàn bảo mật thông tin

Ba mục tiêu chÌnh của an toàn bảo mật thông tin:

4. Mục tiêu chung của an toàn bảo mật thông tin

- **Tính bí mật** *(Confidentiality)*: - Đảm bảo rằng thông tin không bị truy cập bất hợp pháp
  - Thuật ngữ *privacy* thường được sử dụng khi dữ liệu được bảo vệ có liên quan tới các thông tin mang tÌnh cá nhân.
- **Tính toàn vẹn** *(Integrity):* - Đảm bảo rằng thông tin không bị sửa đổi bất hợp pháp.
- **Tính sẵn dùng** *(availability):* - Tài sản luôn sẵn sàng được sử dụng bởi nhưng ngợời có thẩm quyền.

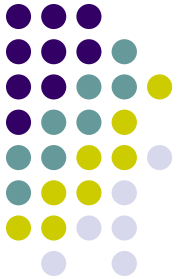
4. Mục tiêu chung của an toàn bảo mật thông tin



Thêm vào đó sự chính xác của thông tin còn được đánh giá bởi:

- **Tính xác thực** *(Authentication):* - Đảm bảo rằng dữ liệu nhận được chắc chắn là dữ liệu gốc ban đầu
- **Tính không thể chối bỏ** *(Non-repudation):* - Đảm bảo rằng người gửi hay người nhận dữ liệu không thể chối bỏ trách nhiệm sau khi đã gửi và nhận thông tin.

5. Các chiến lược an toàn hệ thống



- **Giới hạn quyền hạn tối thiểu** *(Last Privilege)*:theo nguyên tắc này bất kỳ một đối tượng nào cùng chỉ có những quyền hạn nhất định đối với tài nguyên mạng.
- **Bảo vệ theo chiều sâu** *(Defence In Depth)*:Không nên dựa vào một chế độ an toàn nào dù cho chúng rất mạnh, mà nên tạo nhiều cơ chế an toàn để tương hỗ lẫn nhau.
- **Nút thắt** *(Choke Point)*: Tạo ra một "cửa khẩu" hẹp, và chỉ cho phép thông tin đi vào hệ thống của mình bằng con đường duy nhất chỉnh là "cửa khẩu" này.

5. Các chiến lược an toàn hệ thống

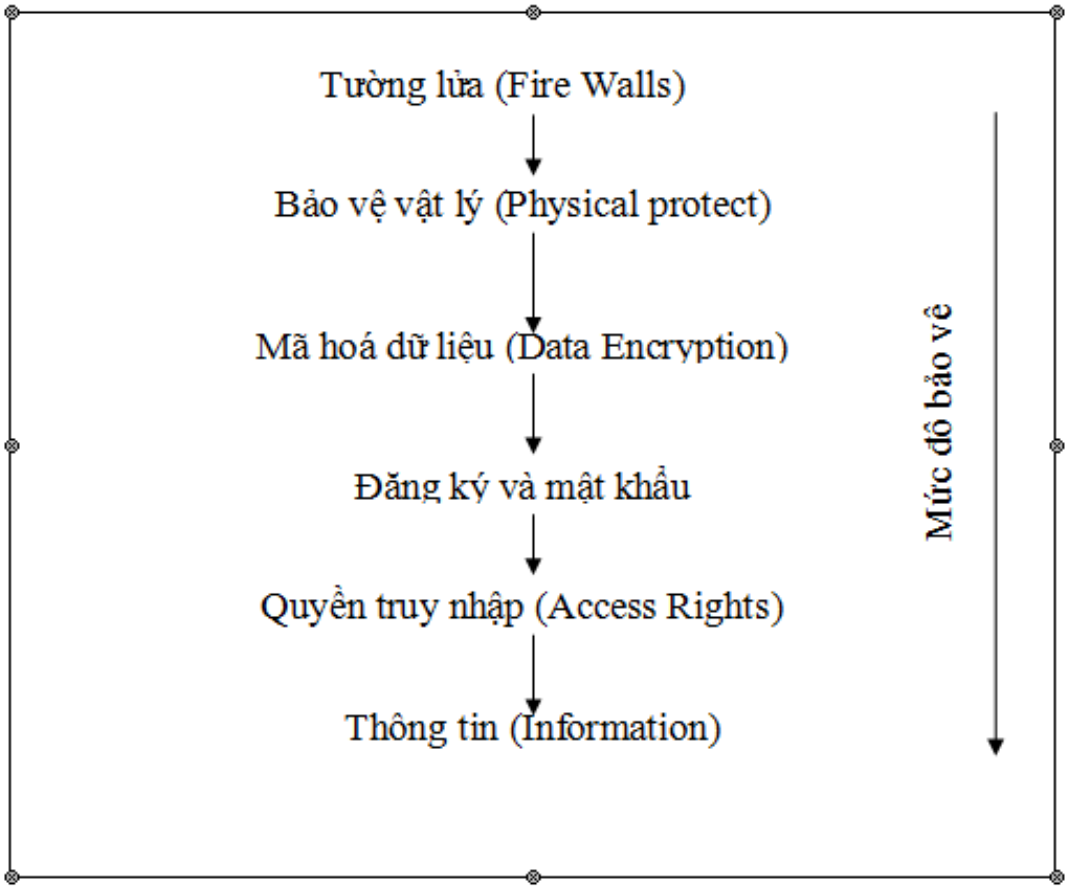
- **Điểm nối yếu nhất** *(Weakest Link):*Chiến lược này dựa trên nguyên tắc: " Một dây xÌch chỉ chắc tại mắt duy nhất, một bức tường chỉ cứng tại điểm yếu nhất".
- **Tính toàn cục***:* Các hệ thống an toàn đòi hỏi phải có tÌnh toàn cục của các hệ thống cục bộ.
- **Tính đa dạng bảo vệ***:* Cần phải sử dụng nhiều biện pháp bảo vệ khác nhau cho hệ thống khác nhau, nếu không có kẻ tấn công vào được một hệ thống thì chúng cũng dễ dàng tấn công vào các hệ thống khác.

6. Các mức bảo vệ trên mạng

- **Quyền truy nhập**: Là lớp bảo vệ trong cùng nhằm kiểm soát các tài nguyên của mạng và quyền hạn trên tài nguyên đó.
- **Đăng ký tên /mật khẩu**: Thực ra đây cũng là kiểm soát quyền truy nhập, nhưng không phải truy nhập ở mức thông tin mà ở mức hệ thống.
- **Mã hoá dữ liệu***:* Dữ liệu bị biến đổi từ dạng nhận thức được sang dạng không nhận thức được theo một thuật toán nào đó và sẽ được biến đổi ngược lại ở trạm nhận (giải mã).
- **Bảo vệ vật lý***:* Ngăn cản các truy nhập vật lý vào hệ thống.

6. Các mức bảo vệ trên mạng

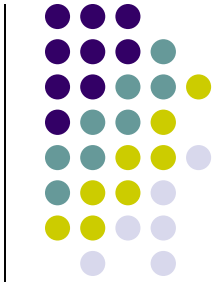
- **Tường lửa**: **Ngăn chặn thâm nhập trái phép và lọc bỏ các gói tin không muốn gửi hoặc nhận vì các lý do nào đó để bảo vệ một máy tính hoặc cả mạng nội bộ (intranet).**



6. Các mức bảo vệ trên mạng

- **Quản trị mạng**: Công tác quản trị mạng máy tÌnh phải được thực hiện một cách khoa học đảm bảo các yêu cầu sau :
  - ➔ Toàn bộ hệ thống hoạt động bình thường trong giờ làm việc.
  - ➔ Có hệ thống dự phòng khi có sự cố về phần cứng hoặc phần mềm xảy ra.
  - ➔ Backup dữ liệu quan trọng theo định kỳ.
  - ➔ Bảo dưỡng mạng theo định kỳ.
  - ➔ Bảo mật dữ liệu, phân quyền truy cập, tổ chức nhóm làm việc trên mạng.

7. Các phương pháp bảo mật



Các phương pháp quan trọng

- **Viết mật mã**: đảm bảo tÌnh bÌ mật của thông tin truyền thông
- **Xác thực quyền**: được sử dụng để xác minh, nhận dạng quyền hạn của các thành viên tham gia.

8. An toàn thông tin bằng mật mã

Mật mã là một ngành khoa học chuyên nghiên cứu các phương pháp truyền tin bÌ mật. Mật mã bao gồm : Lập mã và phá mã.

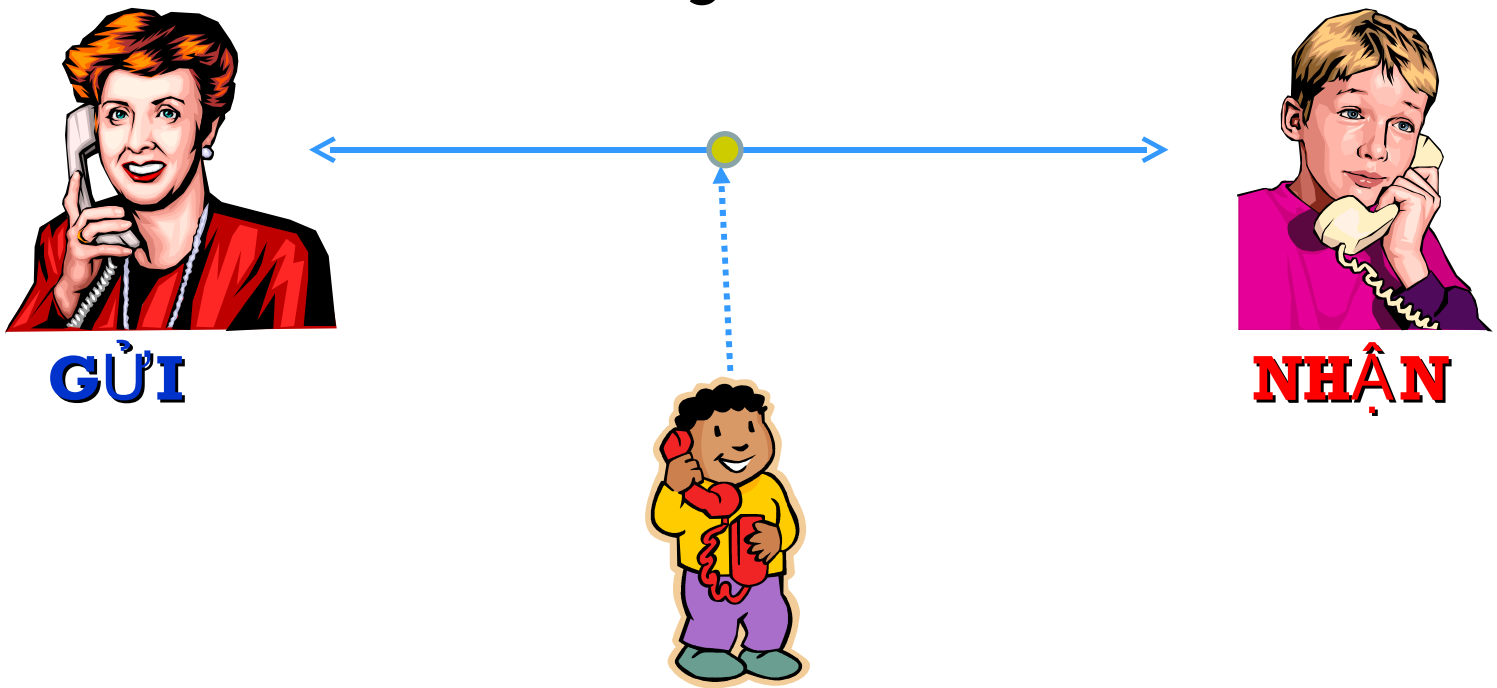
- **Lập mã bao gồm hai quá trình:** mã hóa và giải mã.Các sản phẩm của lĩnh vực này là các hệ mã mật , các hàm băm, các hệ chữ ký điện tử, các cơ chế phân phối, quản lý khóa và các giao thức mật mã.

- **Phá mã**: Nghiên cứu các phương pháp phá mã hoặc tạo mã giả. Sản phẩm của lĩnh vực này là các phương pháp phá mã , các phương pháp giả mạo chữ ký, các phương pháp tấn công các hàm băm và các giao thức mật mã

8. An toàn thông tin bằng mật mã

Cách hiểu truyền thống: giữ bí mật nội dung trao đổi

GỬI v‡ NHẬN trao đổi với nhau trong khi TRUNG GIAN tÏm c·ch "nghe lÈn"



8. An toàn thông tin bằng mật mã

- Một trong những nghệ thuật để bảo vệ thông tin là biến đổi nó thành một định dạng mới khó đọc.

- Viết mật mã có liên quan đến việc mã hoá các thông báo trước khi gửi chúng đi và tiến hành giải mã chúng lúc nhận được

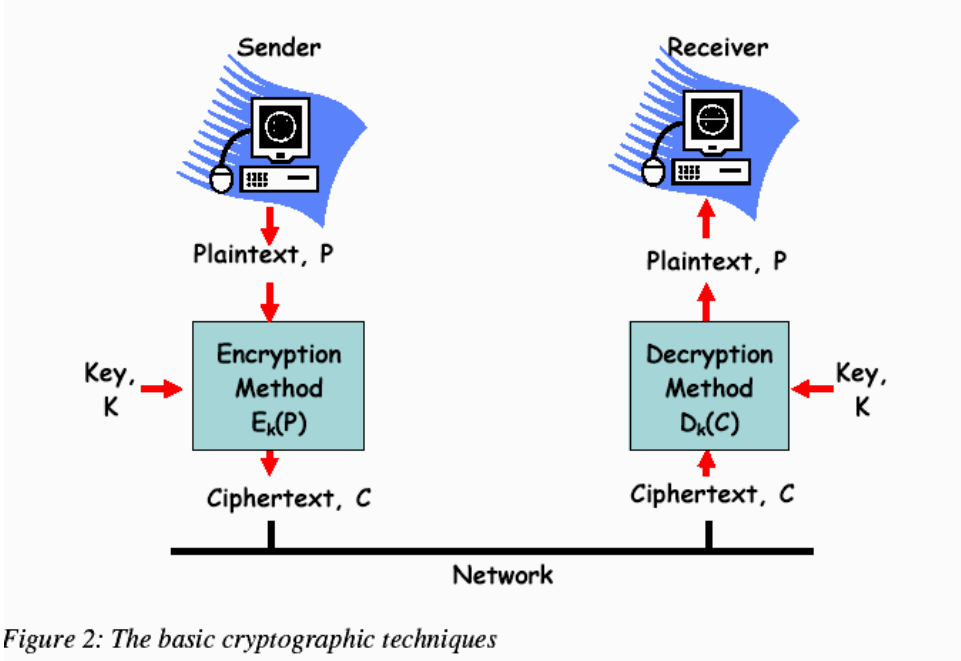


Figure 2: The basic cryptographic techniques

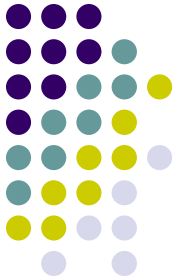
8. An toàn thông tin bằng mật mã

- **Có 2 phương thức mã hoá cơ bản: thay thế và hoán vị**:

  - ✓ **Phương thức mã hoá thay thế**: là phương thức mã hoá mà từng ký tự gốc hay một nhóm ký tự gốc của bản rõ được thay thế bởi các từ, các ký hiệu khác hay kết hợp với nhau cho phù hợp với một phương thức nhất định và khoá.

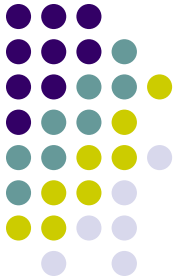
  - ✓ **Phương thức mã hoá hoán vị:** là phương thức mã hoá mà các từ mã của bản rõ được sắp xếp lại theo một phương thức nhất định.

9. Hệ mật mã



- **Vai trò của hệ mật mã:**
  - ✓ Hệ mật mã phải che dấu được nội dung của văn bản rõ (PlainText).
  - ✓ Tạo các yếu tố xác thực thông tin, đảm bảo thông tin lưu hành trong hệ thống đến người nhận hợp pháp là xác thực (Authenticity).
  - ✓ Tổ chức các sơ đồ chữ ký điện tử, đảm bảo không có hiện tượng giả mạo, mạo danh để gửi thông tin trên mạng.

9. Hệ mật mã



- **Khái niệm cơ bản**
  - ✓ **Bản rõ** X được gọi là là bản tin gốc. Bản rõ có thể được chia nhỏ có kÌch thước phù hợp.
  - ✓ **Bản mã** Y là bản tin gốc đã được mã hoá. Ở đây ta thường xét phương pháp mã hóa mà không làm thay đổi kÌch thước của bản rõ, tức là chúng có cùng độ dài.
  - ✓ **Mã** là thuật toán E chuyển bản rõ thành bản mã. Thông thường chúng ta cần thuật toán mã hóa mạnh, cho dù kẻ thù biết được thuật toán, nhưng không biết thông tin về khóa cũng không tìm được bản rõ.

9. Hệ mật mã

- **Khái niệm cơ bản**
  - ✓ **Khoá** K là thông tin tham số dùng để mã hoá, chỉ có người gửi và người nhận biết. Khóa là độc lập với bản rõ và có độ dài phù hợp với yêu cầu bảo mật.
  - ✓ **Mã hoá** là quá trình chuyển bản rõ thành bản mã, thông thường bao gồm việc áp dụng thuật toán mã hóa và một số quá trình xử lý thông tin kèm theo.
  - ✓ **Giải mã** chuyển bản mã thành bản rõ, đây là quá trình ngược lại của mã hóa.

9. Hệ mật mã

- **Các thành phần của một hệ mật mã** *:*

Một hệ mã mật là bộ 5 (P, C, K, E, D) thoả mãn các điều kiện sau:

- **P** là không gian bản rõ: là tập hữu hạn các bản rõ có thể có.
- **C** là không gian bản mã: là tập hữu hạn các bản mã có thể có.
- **K** là kkhông gian khoá: là tập hữu hạn các khoá có thể có.

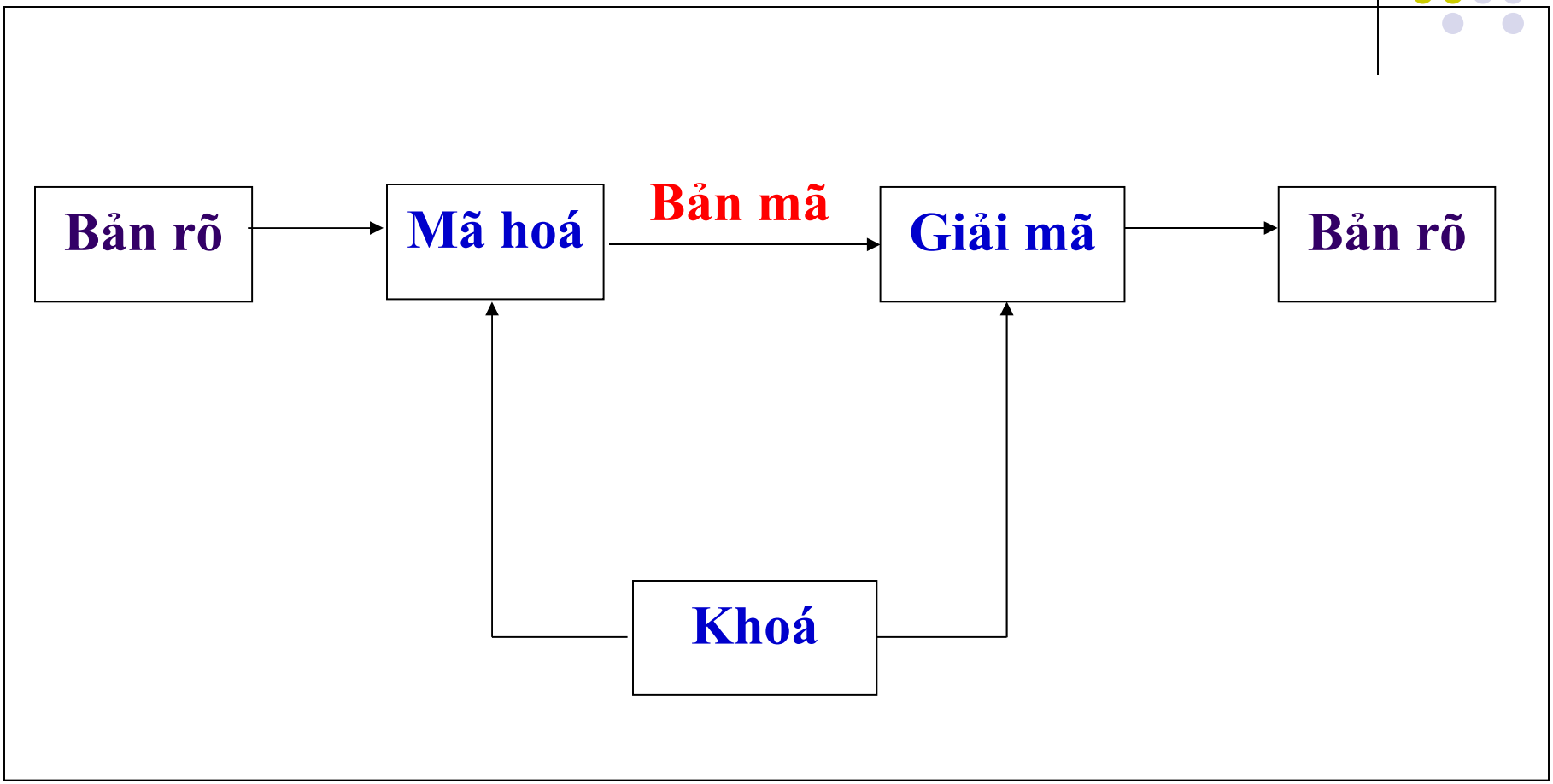
Đối với mỗi $k \in K$ có một quy tắc mã $e_K: P \rightarrow C$ và một quy tắc giải mã tương ứng $d_K \in D$.

Với mỗi $e_K: P \rightarrow C$ và $d_K: C \rightarrow P$ là những hàm mà

$d_K(e_K(x))=x$ với mọi bản rõ $x \in P$.

Hàm giải mã d_k chính là ánh xạ ngược của hàm mã hóa e_k

9. Hệ mật mã



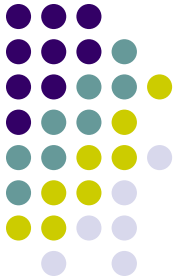
Quá trình mã hóa và giải mã thông tin

10. Phân loại hệ mật mã

- **Hệ mật đôi xứng** (hay còn gọi là mật mã khóa bí mật): là những hệ mật dùng chung một khoá cả trong quá trình mã hoá dữ liệu và giải mã dữ liệu. Do đó khoá phải được giữ bí mật tuyệt đối. Một số thuật toán nổi tiếng trong mã hoá đối xứng là: DES, Triple DES(3DES), RC4, AES…

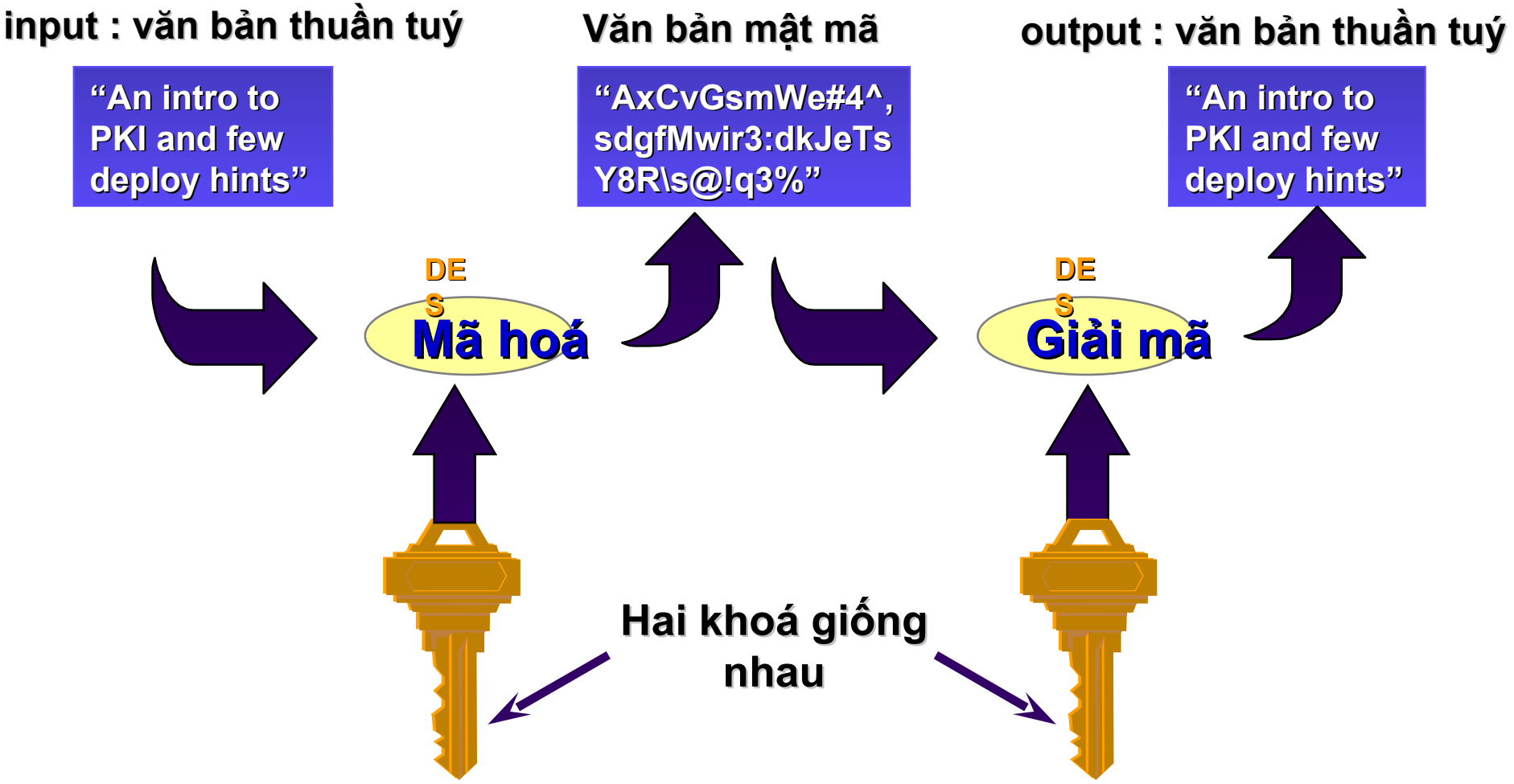
- **Hệ mật mã bất đối xứng** (hay còn gọi là mật mã khóa công khai): Các hệ mật này dùng một khoá để mã hoá sau đó dùng một khoá khác để giải mã, nghĩa là khoá để mã hoá và giải mã là khác nhau. Các khoá này tạo nên từng cặp chuyển đổi ngược nhau và không có khoá nào có thể suy được từ khoá kia. Khoá dùng để mã hoá có thể công khai nhưng khoá dùng để giải mã phải giữ bí mật. Do đó trong thuật toán này có 2 loại khoá: Khoá để mã hoá được gọi là khóa công khai-Public Key, khoá để giải mã được gọi là khóa bí mật - Private Key. Một số thuật toán mã hoá công khai nổi tiếng: Diffle-Hellman, RSA,…

10. Các phương pháp mã hoá



- **Có ba phương pháp chính cho việc mã hoá và giải mã**
  - Sử dụng khoá đối xứng
  - Sử dụng khoá bất đối xứng
  - Sử dụng hàm băm một chiều

10.1 Mã hoá đối xứng



10.1 Mã hoá đối xứng

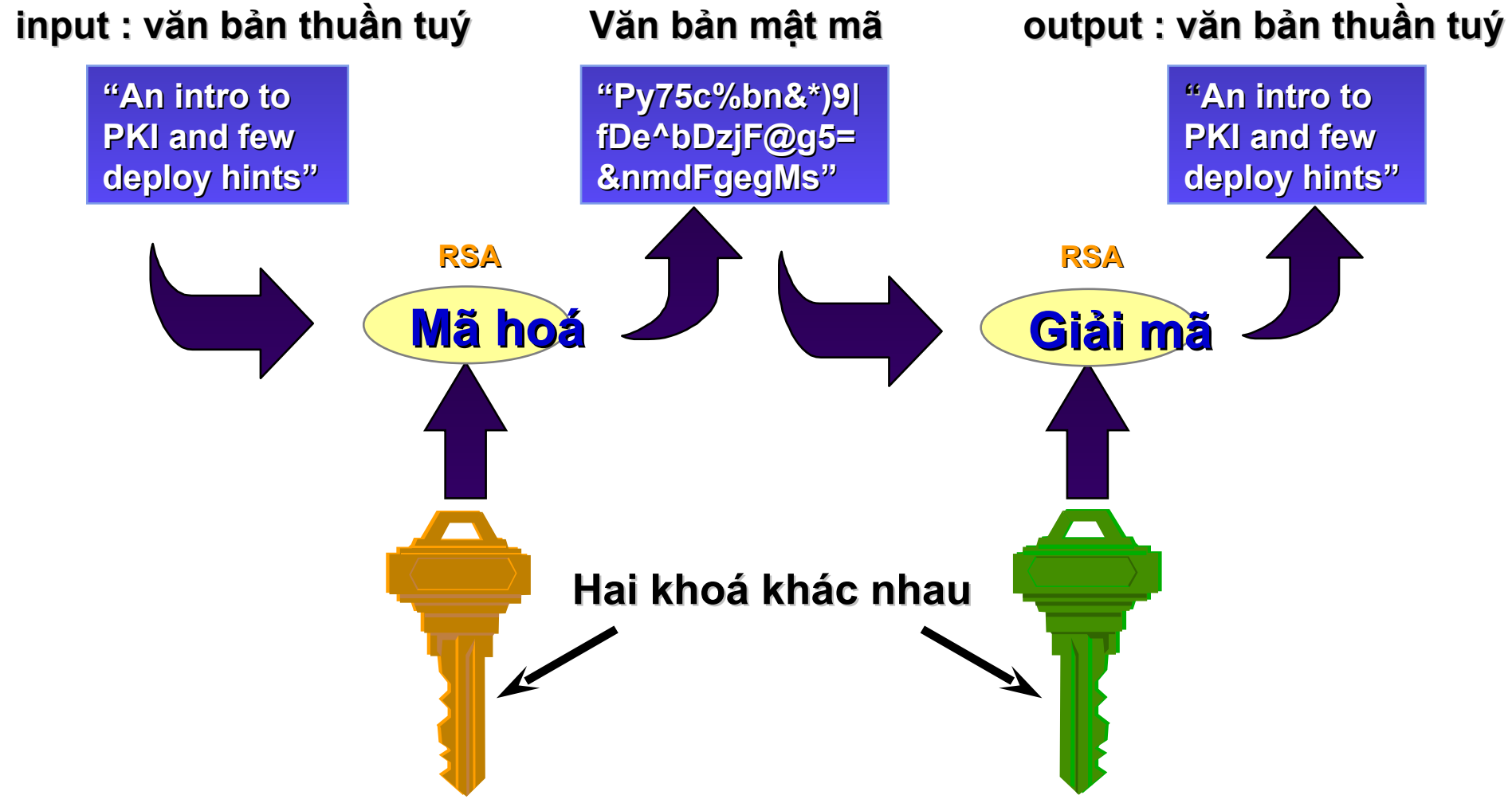
- Các khoá giống nhau được sử dụng cho việc mã hoá và giải mã
- Thuật toán mã hoá sử dụng khoá đối xứng thường được biết đến là DES (Data Encryption Standard)
- Các thuật toán mã hoá đối xứng khác được biết đến như:

  -Triple DES, DESX, GDES, RDES - 168 bit key

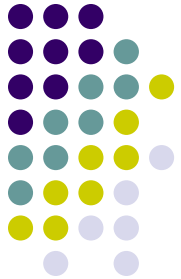
  -RC2, RC4, RC5 - variable length up to 2048 bits

  -IDEA - basis of PGP - 128 bit key

10.2 Mã hoá bất đối xứng

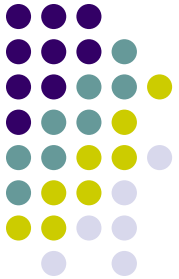


10.2 Mã hoá bất đối xứng



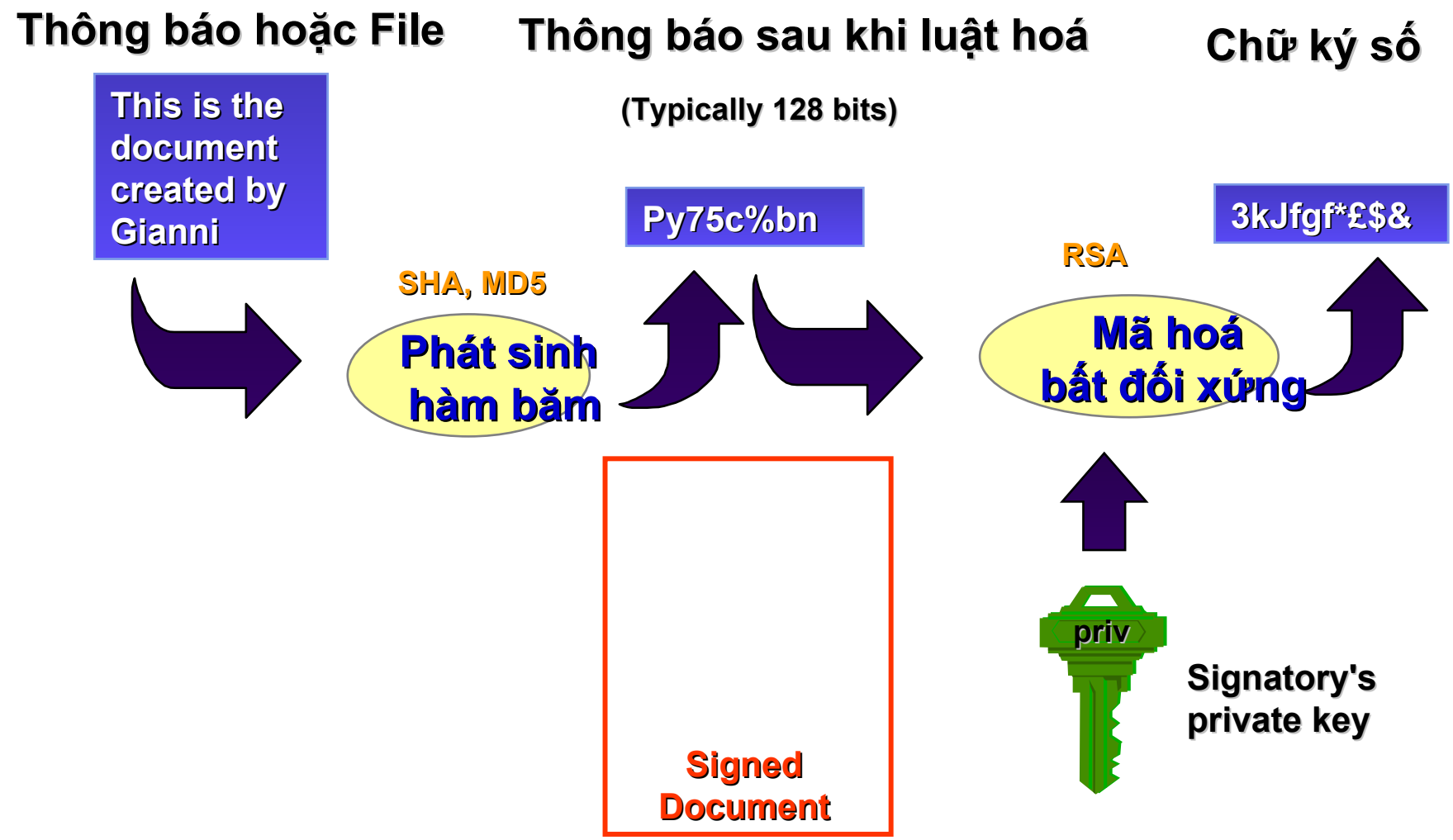
- Các khoá dùng cho mã hoá và giải mã khác nhau nhưng cùng một mẫu và là cặp đôi duy nhất(khoá private/public)
- Khoá private chỉ được biết đến bởi người gửi
- Khoá public được biết đến bởi nhiều người hơn nó được sử dụng bởi những nhóm người đáng tin cậy đã được xác thực
- Thuật toán mã hoá sử dụng khoá bất đối xứng thường được biết đến là RSA (Rivest,Shamir and Adleman 1978)

10.3 Hàm băm

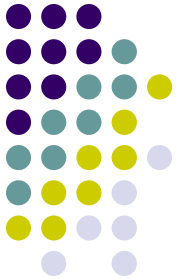


- Một hàm băm H nhận được một thông báo *m* với một độ dài bất kỳ từ đầu vào và đưa ra một xâu bÌt *h* có độ dài cố định ở đầu ra *h* = *H(m).*
- Hàm băm là một hàm một chiều, điều đó có nghĩa là ta không thể tÌnh toán được đầu vào *m* nếu biết đầu ra *h.*
- Thuật toán sử dụng hàm băm thường được biết đến là MD5

10.4 Tạo ra chữ ký số



11. Xác thực quyền



- Xác minh quyền hạn của các thành viên tham gia truyền thông
- Phương pháp phổ biến:
  - Sử dụng Password : để xác thực người sử dụng

11. Xác thực quyền

- Sử dụng Kerberos: phương thức mã hoá và xác thực trong AD của công nghệ Window
- Sử dụng Secure Remote Password (SRP): là một giao thức để xác thực đối với các truy cập từ xa
- Sử dụng Hardware Token
- Sử dụng SSL/TLS Certificate Based Client Authentication: sử dụng SSL/TLS để mã hoá, xác thực trong VPN, Web…
- Sử dụng X.509 Public Key
- Sử dụng PGP Public Key
- Sử dụng SPKI Public Key
- Sử dụng XKMS Public Key.
- Sử dụng XML Digital Signature

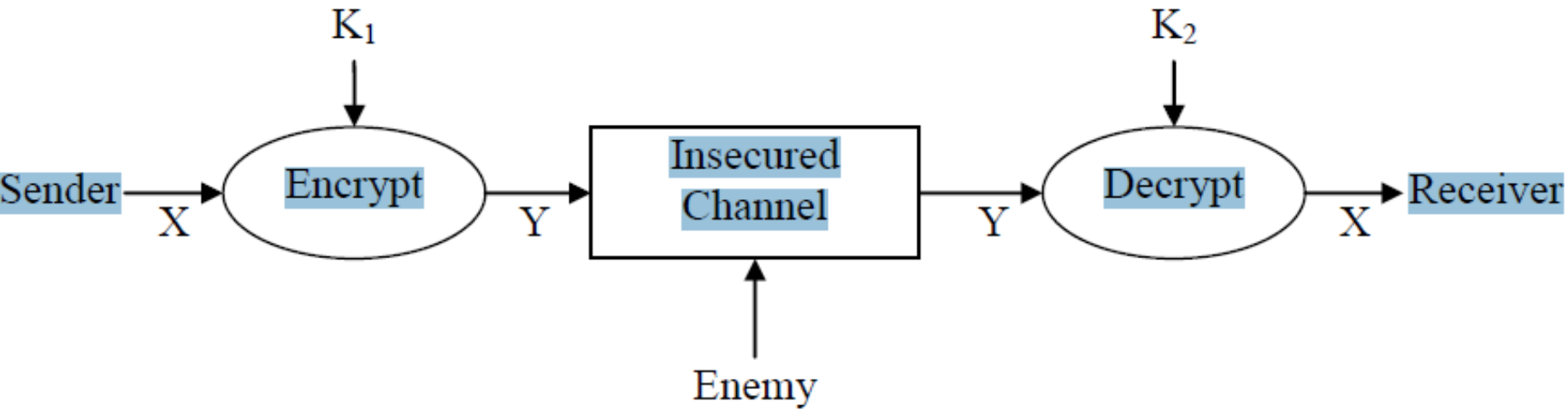
12.Tiêu chuẩn đánh giá hệ mật mã

- **Độ an toàn:** Một hệ mật được đưa vào sử dụng điều đầu tiên phải có độ an toàn cao.
  - Chúng phải có phương pháp bảo vệ mà chỉ dựa trên sự bÌ mật của các khoá, còn thuật toán thì công khai. Tại một thời điểm, độ an toàn của một thuật toán phụ thuộc:
    - ✓ Nếu chi phÌ hay phÌ tổn cần thiết để phá vỡ một thuật toán lớn hơn giá trị của thông tin đã mã hóa thuật toán thì thuật toán đó tạm thời được coi là an toàn.
    - ✓ Nếu thời gian cần thiết dùng để phá vỡ một thuật toán là quá lâu thì thuật toán đó tạm thời được coi là an toàn.
    - ✓ Nếu lượng dữ liệu cần thiết để phá vỡ một thuật toán quá lơn so với lượng dữ liệu đã được mã hoá thì thuật toán đó tạm thời được coi là an toàn
  - Bản mã C không được có các đặc điểm gây chú ý, nghi ngờ.

12.Tiêu chuẩn đánh giá hệ mật mã

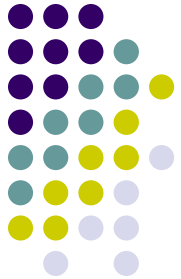
- **Tốc độ mã và giải mã:** Khi đánh giá hệ mật mã chúng ta phải chú ý đến tốc độ mã và giải mã. Hệ mật tốt thì thời gian mã và giải mã nhanh.
- **Phân phối khóa:** Một hệ mật mã phụ thuộc vào khóa, khóa này được truyền công khai hay truyền khóa bÌ mật. Phân phối khóa bÌ mật thì chi phÌ sẽ cao hơn so với các hệ mật có khóa công khai. Vì vậy đây cũng là một tiêu chÌ khi lựa chọn hệ mật mã.

13. Mô hình truyền tin cơ bản của mật mã học và luật Kirchoff



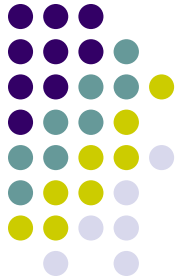
Hình 1.1: Mô hình cơ bản của truyền tin bảo mật

13. Mô hình truyền tin cơ bản của mật mã học và luật Kirchoff



- **Theo luật Kirchoff (1835 - 1903)** (một nguyên tắc cơ bản trong mã hoá) thì: *toàn bộ cơ chế mã/giải mã trừ khoá là không bí mật đối với kẻ địch.*
- **Ý nghĩa của luật Kirchoff**: sự an toàn của các hệ mã mật không phải dựa vào sự phức tạp của thuật toán mã hóa sử dụng.

14. Các loại tấn công



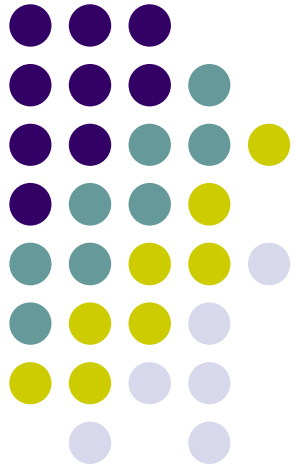
- **Các kiểu tấn công khác nhau**
  - *E biết được Y (ciphertext only attack).*
    - Eavesdropper: kẻ nghe trộm (Eve)
  - *E biết một số cặp plaintext-ciphertext* X-Y (known plaintext attack).
  - *E biết được cryptogram cho một số tin X do bản thân soạn ra* (chosen plaintext attack).

15. Một số ứng dụng của mã hóa trong security

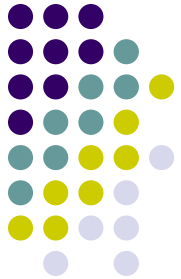
Một số ứng dụng của mã hoá trong đời sống hằng ngày nói chung và trong lĩnh vực bảo mật nói riêng. Đó là:

- Securing Email
- Authentication System
- Secure E-commerce
- Virtual Private Network
- Wireless Encryption

Chương 2: Các phương pháp mã hóa cổ điển

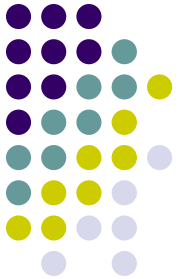


1. Modulo số học



- Ta có a ≡ b(mod n) nếu a = kn + btrong đó k là một số nguyên.
- Nếu a và b dương và a nhỏ hơn n, chúng ta có thể gọi a là phần dư của b khi chia cho n.
- Người ta còn gọi b là thặng dư của a theo modulo n, và a là đồng dư của b theo modulo n

1. Modulo số học



Ví dụ:

Ta có: 42=4.9+6 vậy 42 ≡6 (mod 9)

Ta có câu hỏi; -42 ≡? (mod9), ta thấy -42= -4.9-6

-42 ≡ -6 (mod 9) nhưng -6 ≡ -6+9 ≡ 3 (mod 9)

Vậy nên -42 ≡ 3 (mod 9)

1. Modulo số học

- Modulo số học cũng giống như số học bình thường, bao gồm các phép giao hoán, kết hợp và phân phối. Mặt khác giảm mỗi giá trị trung gian trong suốt quá trình tính toán.

$(a+b) \bmod n = ((a \bmod n) + (b \bmod n)) \bmod n$

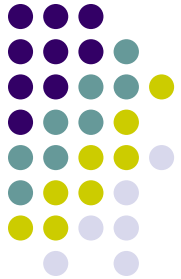
$(a- b) \bmod n = ((a \bmod n) - (b \bmod n)) \bmod n$

$(a \times b) \bmod n = ((a \bmod n) \times (b \bmod n)) \bmod n$

$(a \times (b + c)) \bmod n = (((a \times b) \bmod n) + ((a \times c) \bmod n)) \bmod n$

- Các phép tính trong các hệ mã mật hầu hết đều thực hiện đối với một modulo N nào đó.

2. Vành Z_N



- Tập các số nguyên $Z_N = \{0, 1, \ldots, N-1\}$ trong đó N là một số tự nhiên dương với hai phép toán cộng (+) và nhân (.) được định nghĩa như

Phép cộng:

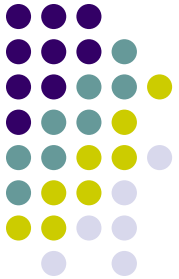
$$\forall\ a, b \in Z_N: a+b = (a+b) \bmod N.$$

Phép nhân:

$$\forall\ a, b \in Z_N: a \,.\, b = (a * b) \bmod N.$$

- Theo tính chất của modulo số học chúng ta dễ dàng nhận thấy Z_N là một vành giao hoán và kết hợp. Hầu hết các tính toán trong các hệ mã mật đều được thực hiện trên một vành Z_N nào đó.

2. Vành Z_N



- Trên vành Z_N

  số 0 là phần tử trung hòa vì $a + 0 = 0 + a = a, \forall a \in Z_N$

  số 1 được gọi là phần tử đơn vị vì $a \cdot 1 = 1 \cdot a = a \ \forall a \in Z_N$.

- Vì dụ N=9

$$Z_9 = \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$$

$$6 + 8 = 14 \equiv 5 \pmod 9$$

$$6 \times 8 = 48 \equiv 3 \pmod 9$$

3. Phần tử nghịch đảo trên vành Z_N

- Trên một vành số nguyên Z_N người ta đưa ra khái niệm về số nghịch đảo của một số như sau:

(GCD-Greatest Common Divisor) ước số chung lớn nhất

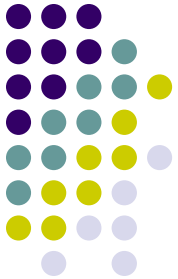
Giả sử a $\in Z_N$ và tồn tại b $\in Z_N$ sao cho a.b = (a*b) mod N = 1. Khi đó b được gọi là phần tử nghịch đảo của a trên Z_N và ký hiệu là $a^{-1} = b$.

Việc tìm phần tử nghịch đảo của một số a $\in Z_N$ cho trước thực chất tương đương với việc tìm hai số b và k sao cho: a.b = k.N + 1 trong đó b, k $\in Z_N$. Hay viết gọn lại là:

$$a^{-1} \equiv b \pmod{N}$$

Định lý về sự tồn tại của phần tử nghịch đảo : Nếu GCD(a, N) = 1 thì tồn tại duy nhất 1 số b $\in Z_N$ là phần tử nghịch đảo của a, nghĩa là thỏa mãn a.b = (a*b) mod N = 1.

4. Các hệ mật mã cổ điển – Hệ mã dịch vòng (shift cipher)



Shift Cipher:

- Một trong những phương pháp lâu đời nhất được sử dụng để mã hóa
- Thông điệp được mã hóa bằng cách dịch chuyển xoay vòng từng ký tự đi *k* vị trÌ trong bảng chữ cái
- Trường hợp với *k*=3 gọi là phương pháp *mã hóa Caesar*.

4. Các hệ mật mã cổ điển – Hệ mã dịch vòng (shift cipher)

Cho $P = C = K = \mathbf{Z}_n$

Với mỗi khóa $k \in K$, định nghĩa:

$e_k(x) = (x + k) \bmod n$ và $d_k(y) = (y - k) \bmod n$ với $x, y \in \mathbf{Z}_n$

$E = \{e_k, k \in K\}$ và $D = \{d_k, k \in K\}$

- Phương pháp đơn giản,
- Thao tác xử lý mã hóa và giải mã được thực hiện nhanh chóng
- Không gian khóa $K = \{0, 1, 2, \ldots, n\text{-}1\} = Z_n$
- Dễ bị phá vỡ bằng cách thử mọi khả năng khóa *k*

4. Các hệ mật mã cổ điển – Hệ mã dịch vòng (shift cipher)

- Vì dụ:
  - Mã hóa một thông điệp được biểu diễn bằng các chữ cái từ A đến Z (26 chữ cái), ta sử dụng Z_{26}.
  - Thông điệp được mã hóa sẽ không an toàn và có thể dễ dàng bị giải mã bằng cách thử lần lượt 26 giá trị khóa *k*.
  - TÌnh trung bình, thông điệp đã được mã hóa có thể bị giải mã sau khoảng 26/2 = 13 lần thử khóa

4. Các hệ mật mã cổ điển – Hệ mã dịch vòng (shift cipher)

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

Ta có sơ đồ mã như sau:

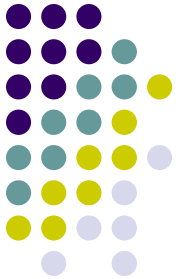
Giả sử $P = C = K = Z_{26}$ với $0 \le k \le 25$

Mã hóa: $e_k(x) = x + k \bmod 26$

Giải mã: $d_k(x) = y - k \bmod 26$

$(x, y \in Z_{26})$

4. Các hệ mật mã cổ điển – Hệ mã dịch vòng (shift cipher)



- Vì dụ K=17. Cho bản mã

  $X = x1; x2; : : : ; x6 = A\ T\ T\ A\ C\ K$.

  $X = x1; x2; : : : ; x6 = 0; 19; 19; 0; 2; 10.$
- Mã hóa

  $y1 = x1 + k \bmod 26 = 0 + 17 \bmod 26 = 17 = R.$

  $y2 = y3 = 19 + 17 \bmod 26 = 10 = K.$

  $y4 = 17 = R.$

  $y5 = 2 + 17 \bmod 26 = 19 = T.$

  $y6 = 10 + 17 \bmod 26 = 1 = B.$
- Giải mã

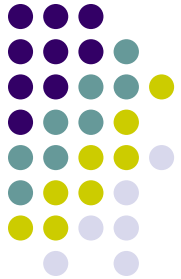
  $Y = y1; y2; : : : ; y6 = R\ K\ K\ R\ T\ B$.

5. Các hệ mật mã cổ điển- Hệ mã hóa thay thế(Substitution Cipher)

Substitution Cipher:

- Phương pháp mã hóa nổi tiếng
- Được sử dụng phổ biến hàng trăm năm nay
- Thực hiện việc mã hóa thông điệp bằng cách hoán vị các phần tử trong bảng chữ cái hay tổng quát hơn là hoán vị các phần tử trong tập nguồn *P*

5. Các hệ mật mã cổ điển- Hệ mã hóa thay thế(Substitution Cipher)



Cho $P = C = Z_n$

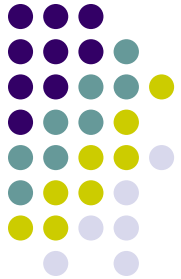
K là tập hợp tất cả các hoán vị của n phần tử $0, 1, \ldots, n-1$. Như vậy, mỗi khóa $\pi \in K$ là một hoán vị của n phần tử $0, 1, \ldots, n-1$.

Với mỗi khóa $\pi \in K$, định nghĩa:

$e_\pi(x) = \pi(x)$ và $d_\pi(y) = \pi^{-1}(y)$ với $x, y \in \mathbf{Z}_n$

$E = \{e_\pi, \pi \in K\}$ và $D = \{D_\pi, \pi \in K\}$

5. Các hệ mật mã cổ điển- Hệ mã hóa thay thế(Substitution Cipher)



- Đơn giản, thao tác mã hóa và giải mã được thực hiện nhanh chóng
- Không gian khóa K gồm $n!$ phần tử
- Khắc phục hạn chế của phương pháp Shift Cipher: việc tấn công bằng cách vét cạn các giá trị khóa $k \in K$ là không khả thi

Thật sự an toàn???

5. Các hệ mật mã cổ điển- Hệ mã hóa thay thế(Substitution Cipher)

AO VCO JO IBU RIBU

AO VCO JO IB

?A H?A ?A

MA HOA VA UN DUNG

Tấn công dựa trên tần số xuất hiện của ký tự trong ngôn ngữ

5. Các hệ mật mã cổ điển- Hệ mã hóa thay thế(Substitution Cipher)

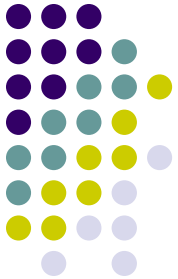
L FDPH L VDZ L FRQTXHUHG

L FDPH L VDZ L FRQTXHUHG

i ?a?e i ?a? i ?????e?e?

i came i saw i conquered

5. Các hệ mật mã cổ điển- Hệ mã hóa thay thế(Substitution Cipher)



- Chọn một hoán vị p: $\mathbf{Z}_{26} \rightarrow \mathbf{Z}_{26}$ làm khoá.
- VD:
  - Mã hoá

$e_p(a)=X$

a	b	c	d	e	f	g	h	i	j	k	l	m
X	N	Y	A	H	P	O	G	Z	Q	W	B	T

n	o	p	q	r	s	t	u	v	w	x	y	z
S	F	L	R	C	V	M	U	E	K	J	D	I

  - Giải mã

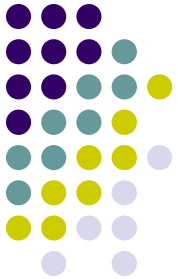
$d_p(A)=d$

A	B	C	D	E	F	G	H	I	J	K	L	M
d	l	r	y	v	o	h	e	z	x	w	p	t

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
b	g	f	j	q	n	m	u	s	k	a	c	i

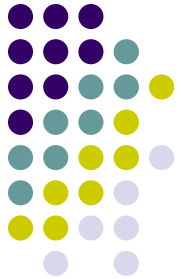
“nguyenthanhnhut” → “SOUDHSMGXSGSGUM”

Độ an toàn của mã thay thế



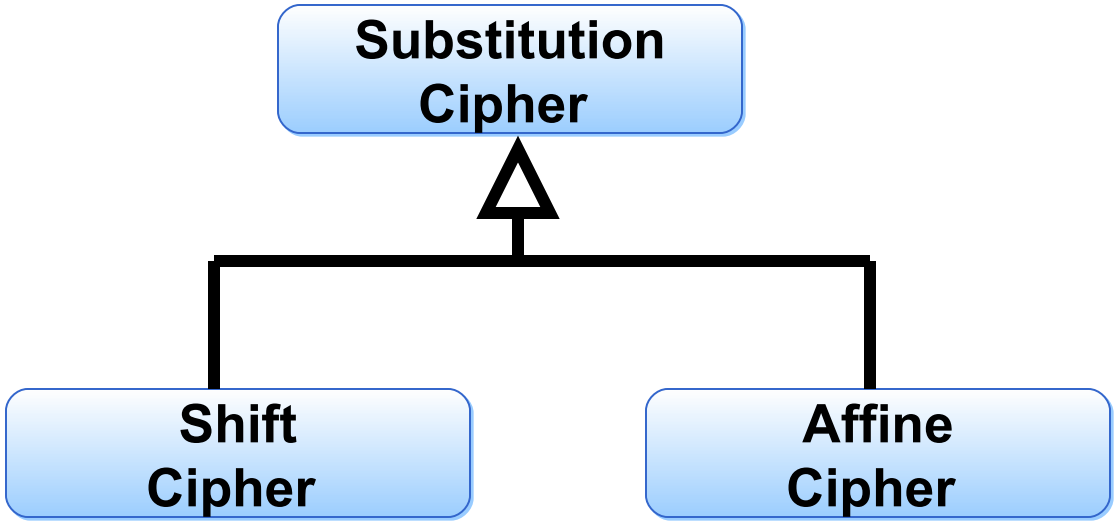
- Một khoá là một hoán vị của 26 chữ cái.
- Có 26! ($\approx 4.10^{26}$) hoán vị (khoá)
- Phá mã:
  - Không thể duyệt từng khoá một.
  - Cách khác?

5. Các hệ mật mã cổ điển- Hệ mã hóa thay thế(Substitution Cipher)



- Phân tÌch tần số
  - Ký tự: E > T > R > N > I > O > A > S
  - Nhóm 2 ký tự (digraph): TH > HE > IN > ER > RE > ON > AN > EN
  - Nhóm 3 ký tự (Trigraph): THE > AND > TIO > ATI > FOR > THA > TER > RES

6. Các hệ mật mã cổ điển - Hệ mã Affine



6. Các hệ mật mã cổ điển - Hệ mã Affine

Cho $P = C = Z_n$

$K = \{(a,b) \in \mathbf{Z}_n \times \mathbf{Z}_n : \gcd(a,n) = 1\}$

Với mỗi khóa $k = (a,b) \in K$, định nghĩa:

$e_k(x) = (ax+b) \bmod n$ và $d_k(x) = (a^{-1}(y-b)) \bmod n$ với $x, y \in \mathbf{Z}_n$

$E = \{e_k, k \in K\}$ và $D = \{D_k, k \in K\}$

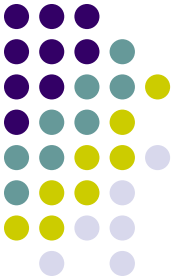
giải mã chÌnh xác thông tin ???

e_k phải là song ánh

$$\forall y \in Z_n, \exists! x \in Z_n, ax + b \equiv y \pmod{n}$$

a và n nguyên tố cùng nhau: gcd(a,n)=1

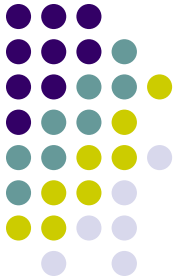
6. Các hệ mật mã cổ điển - Hệ mã Affine



- Vì dụ: Giả sử $P = C = Z_{26}$.
  - *encryption:* $e_k(x) = a \cdot x + b \bmod 26$.
  - *key:* $k = (a, b)$ where $a, b \in Z_{26}$.
  - *decryption:* $x = a^{-1}(y - b) \bmod 26$.

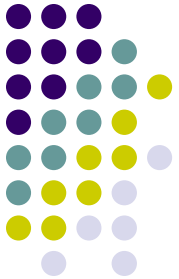
a và 26 nguyên tố cùng nhau: gcd(*a*,*n*)=1

6. Các hệ mật mã cổ điển - Hệ mã Affine



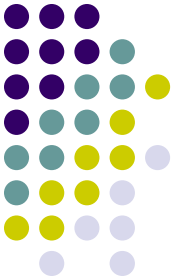
- Mã tuyến tình là một mã thay thế có dạng
  $e(x) = ax + b \pmod{26}$, trong đó a, b e $\mathbf{Z}_{26}$.
  Trường hợp a = 1 là mã dịch chuyển.
- Giải mã: Tìm x?
  $y = ax + b \pmod{26}$
  $ax = y - b \pmod{26}$
  $x = a^{-1}(y - b) \pmod{26}$.
- Vấn đề: TÌnh a^{-1}.
  Để có a^{-1}, đòi hỏi (a,26)=1.
  TÌnh a^{-1}: Thuật toán Euclide mở rộng.

VD: bài tập



- a = 5, b = 3: y = 5x + 3 (mod 26).
- Mã hoá: NGUYENTHANHNHUT → ?

6. Các hệ mật mã cổ điển - Hệ mã Affine

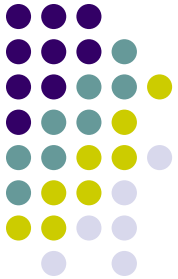


- **Ví dụ**

✓ Khóa
  - Plain(a): **abcdefghijklmnopqrstuvwxyz**
  - Cipher(b): **DKVQFIBJWPESCXHTMYAUOLRGZN**

✓ Mã hóa:
  - Plaintext: **ifwewishtoreplaceletters**
  - Ciphertext: **WIRFRWAJUHYFTSDVFSFUUFYA**

6. Các hệ mật mã cổ điển - Hệ mã Affine



Gọi $\phi(n)$ là số lượng phần tử thuộc $\mathbf{Z}_n$ và nguyên tố cùng nhau với n.

Nếu $n = \prod_{i=1}^{m} p_i^{e_i}$ *với* p_i *là các số nguyên tố khác nhau và* $e_i \in \mathbf{Z}^+$, $1 \le i \le m$

thì $\phi(n) = \prod_{i=1}^{m} \left(p_i^{e_i} - p_i^{e_i - 1} \right)$.

- n khả năng chọn giá trị b
- $\phi(n)$ khả năng chọn giá trị a
- $n \times \phi(n)$ khả năng chọn lựa khóa $k = (a, b)$

7. Thuật toán Euclide mở rộng

$$r_0 = q_1 r_1 + r_2, \ 0 < r_2 < r_1$$

$$r_1 = q_2 r_2 + r_3, \ 0 < r_3 < r_2$$

$$\ldots$$

$$r_{m-2} = q_{m-1} r_{m-1} + r_m, \ 0 < r_m < r_{m-1}$$

$$r_{m-1} = q_m r_m$$

$$\gcd(r_0, r_1) = \gcd(r_1, r_2) = \ldots = \gcd(r_{m-1}, r_m) = r_m$$

7. Thuật toán Euclide mở rộng

- Xây dựng dãy số:

$$t_0 = 0$$

$$t_1 = 1$$

$$t_j = (t_{j-2} - q_{j-1}t_{j-1}) \bmod r_0 \text{ với } j \geq 2$$

- Nhận xét:

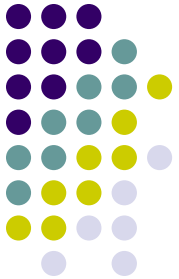
$$\text{Với mọi } j,\ 0 \leq j \leq m\ ,\ ta\ có\ r_j \equiv t_j r_1 (\bmod r_0)$$

$$\gcd(r_0, r_1) = 1 \Rightarrow t_m = r_1^{-1} \bmod r_0$$

8. Phương pháp Vigenere

- Trong phương pháp mã hóa bằng thay thế: với một khóa *k* được chọn, mỗi phần tử $x \in P$ được ánh xạ vào duy nhất một phần tử $y \in C$.
- Phương pháp Vigenere sử dụng khóa có độ dài *m*.
- Được đặt tên theo nhà khoa học Blaise de Vigenere (thế kỷ 16)
- Có thể xem phương pháp mã hóa Vigenere bao gồm *m* phép mã hóa bằng dịch chuyển được áp dụng luân phiên nhau theo chu kỳ
- Không gian khóa *K* của phương pháp Vigenere có số phần tử là n^m
- Vì dụ: n=26, m=5 thì không gian khóa ~1.1×10^7

8. Phương pháp Vigenere



Chọn số nguyên dương m. Định nghĩa $P = C = K = (\mathbf{Z}_n)^m$

$K = \{(k_1, k_2, \ldots, k_m) \in (Z_n)^m\}$

Với mỗi khóa $k = (k_1, k_2, \ldots, k_m) \in K$, định nghĩa:

$e_k(x_1, x_2, \ldots, x_m) = ((x_1 + k_1) \bmod n, (x_2 + k_2) \bmod n, \ldots, (x_m + k_m) \bmod n)$

$d_k(y_1, y_2, \ldots, y_m) = ((y_1 - k_1) \bmod n, (y_2 - k_2) \bmod n, \ldots, (y_m - k_m) \bmod n)$

với $x, y \in (\mathbf{Z}_n)^m$.

8. Phương pháp Vigenere

- Vì dụ: *m* = 6 và keyword là CIPHER
- Suy ra, khóa *k* = (2, 8, 15, 7, 4, 17)
- Cho bản rõ: **thiscryptosystemisnotsecure**

19	7	8	18	2	17
2	8	15	7	4	17
21	15	23	25	6	8

24	15	19	14	18	24
2	8	15	7	4	17
0	23	8	21	22	15

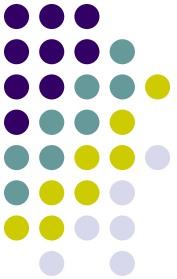
18	19	4	12	8	18
2	8	15	7	4	17
20	1	19	19	12	9

13	14	19	18	4	2
2	8	15	7	4	17
15	22	8	25	8	19

20	17	4
2	8	15
22	25	19

- **Vậy bản mã là: “vpxzgiaxivwoubttmjpwizitwzt”**

9. Phương pháp mã hóa Hill



- Phương pháp Hill (1929)
- Tác giả: Lester S. Hill
- Ý tưởng chÌnh:
  - Sử dụng *m* tổ hợp tuyến tÌnh của *m* ký tự trong plaintext để tạo ra *m* ký tự trong ciphertext
- VÌ dụ:

$$y_1 = 11x_1 + 3x_2$$
$$y_2 = 8x_1 + 7x_2.$$

$$(y_1, y_2) = (x_1, x_2) \begin{pmatrix} 11 & 8 \\ 3 & 7 \end{pmatrix}$$

9. Phương pháp mã hóa Hill

Chọn số nguyên dương m. Định nghĩa:

$P = C = (\mathbf{Z}_n)^m$ và K là tập hợp các ma trận $m \times m$ khả nghịch

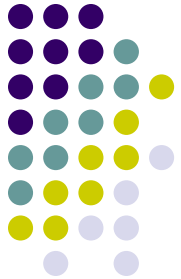
Với mỗi khóa $k = \begin{pmatrix} k_{1,1} & k_{1,2} & \cdots & k_{1,m} \\ k_{2,1} & \cdots & \cdots & k_{2,m} \\ \vdots & \vdots & & \vdots \\ k_{m,1} & k_{m,2} & \cdots & k_{m,m} \end{pmatrix} \in K$, định nghĩa:

$$e_k(x) = xk = (x_1, x_2, \ldots, x_m) \begin{pmatrix} k_{1,1} & k_{1,2} & \cdots & k_{1,m} \\ k_{2,1} & \cdots & \cdots & k_{2,m} \\ \vdots & \vdots & & \vdots \\ k_{m,1} & k_{m,2} & \cdots & k_{m,m} \end{pmatrix} \text{ với } x = (x_1, x_2, \ldots, x_m) \in P$$

và $d_k(y) = yk^{-1}$ với $y \in C$.

Mọi phép toán số học đều được thực hiện trên $\mathbf{Z}_n$.

9. Phương pháp mã hóa Hill

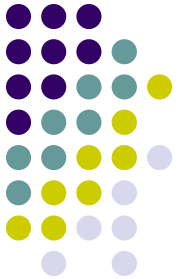


Ví dụ: cho hệ mã Hill có M = 2 (khóa là các ma trận vuông cấp 2) và bảng chữ cái là bảng chữ cái tiếng Anh, tức là N = 26. Cho khóa

$$K = \begin{pmatrix} 3 & 3 \\ 2 & 5 \end{pmatrix}$$

Hãy mã hóa xâu P = “HELP” và giải mã ngược lại bản mã thu được.

9. Phương pháp mã hóa Hill



Để mã hóa chúng ta chia xâu bản rõ thành hai vecto hàng 2 chiều "HE" (7 4) và "LP" (11 15) và tiến hành mã hóa lần lượt.

Với $P_1 = (7\ 4)$ ta có $C_1 = P_1 * K = (7\ 4)\begin{pmatrix} 3 & 3 \\ 2 & 5 \end{pmatrix} = (3\ 15) = (\mathrm{D\ P})$

Với $P_2 = (11\ 15)$ ta có $C_2 = P_2 * K = (11\ 15)\begin{pmatrix} 3 & 3 \\ 2 & 5 \end{pmatrix} = (11\ 4) = (\mathrm{L\ E})$

Vậy bản mã thu được là C = "DPLE".

9. Phương pháp mã hóa Hill

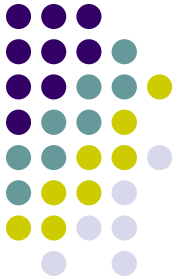
Để giải mã ta tính khóa giải mã là ma trận nghịch đảo của ma trận khóa trên Z_{26} theo công thức sau:

Với $K = \begin{pmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{pmatrix}$ và $\det(K) = (k_{11}*k_{22} - k_{21}*k_{12}) \bmod N$ là một phần tử có phần tử nghịch đảo trên Z_N (ký hiệu là $\det(K)^{-1}$) thì khóa giải mã sẽ là

$$K^{-1} = \det(K)^{-1} * \begin{pmatrix} k_{22} & -k_{12} \\ -k_{21} & k_{11} \end{pmatrix}$$

Áp dụng vào trường hợp trên ta có $\det(K) = (15 - 6) \bmod 26 = 9$. $GCD(9, 26) = 1$ nên áp dụng thuật toán Ơclit mở rộng tìm được $\det(K)^{-1} = 3$. Vậy $K^{-1} = 3 * \begin{pmatrix} 5 & 23 \\ 24 & 3 \end{pmatrix} = \begin{pmatrix} 15 & 17 \\ 20 & 9 \end{pmatrix}$.

9. Phương pháp mã hóa Hill

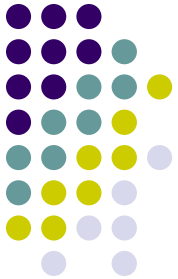


Giải mã C = “DP” = (3 15), $P = C * K^{-1} = (3\ 15) * \begin{pmatrix} 15 & 17 \\ 20 & 9 \end{pmatrix} = (3\ 15)$ = “HE”.

Tương tự giải mã xâu C = “LE” kết quả sẽ được bản rõ P = “LP”.

Chú ý là trong ví dụ trên chúng ta sử dụng khóa K có kích thước nhỏ nên dễ dàng tìm được khóa để giải mã còn trong trường hợp tổng quát điều này là không dễ dàng.

10. Các hệ mã dòng



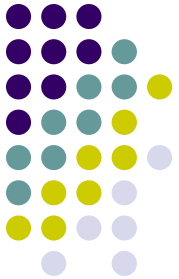
- ***Định nghĩa***

  Mật mã dòng là một bộ (P,C,K,L,F,E,D) thoả mãn dược các điều kiện sau:

  - *P là một tập hữu hạn các bản rõ có thể.*
  - *C là tập hữu hạn các bản mã có thể.*
  - *K là tập hữu hạn các khoá có thể (không gian khoá)*
  - *L là tập hữu hạn các bộ chữ của dòng khoá.*
  - *F = (f1 f2...) là bộ tạo dòng khoá. Với $i \geq 1$*

    $fi : K \times P\ i\text{-}1 \rightarrow L$
  - *Với mỗi $z \in L$ có một quy tắc mã $ez \in E$ và một quy tắc giải mã tương ứng $dz \in D$. $ez : P \rightarrow C$ và $dz : C \rightarrow P$ là các hàm thoả mãn $dz(ez(x)) = x$ với mọi bản rõ $x \in P$.*

10. Các hệ mã dòng

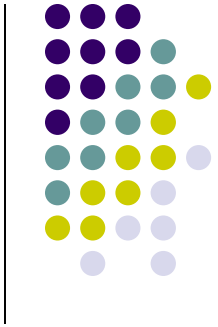


- Các mã dòng thường được mô tả trong các bộ chữ nhị phân tức là $P = C = L = Z_2$. Trong trường hợp này, các phép toán mã và giải mã là phép cộng theo *modulo 2*.

$$y_i = e_{z_i}(x_i) = x_i + z_i \bmod 2 \rightarrow \text{encryption}$$

$$x_i = e_{z_i}(y_i) = y_i + z_i \bmod 2 \rightarrow \text{decryption}$$

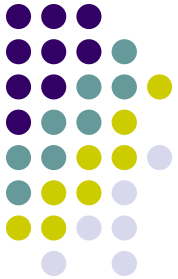
10. Các hệ mã dòng



- ***Chú ý:*** Nếu ta coi "*0*" biểu thị giá trị "sai" và "*1*" biểu thị giá trị "đúng" trong đại số Boolean thì phép cộng theo *moulo 2* sẽ ứng với phép hoặc loại trừ (XOR).
- Bảng chân lý phép cộng theo modul 2 giống như bảng chân lý của phép toán XOR

a	b	$c = a + b \bmod 2$
0	0	$0 + 0 = 0 \bmod 2$
0	1	$0 + 1 = 1 \bmod 2$
1	0	$1 + 0 = 1 \bmod 2$
1	1	$1 + 1 = 0 \bmod 2$

10. Các hệ mã dòng



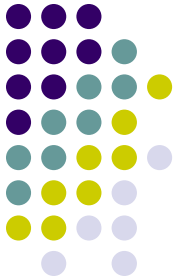
- Hàm mã hóa và giải mã được thực hiện bởi cùng một phép toán là phép cộng theo modulo 2(hay phép XOR)
- Vì:

$$\text{Decryption: } y_i + z_i = \underbrace{(x_i + z_i)}_{\text{encryption}} + z_i = x_i + (z_i + z_i) \equiv x_i \bmod 2.$$

- Trong đó với z_i=0 và z_i=1 thì

$$z_i + z_i \equiv 0 \bmod 2:$$

10. Các hệ mã dòng



- Vì dụ: mã hóa ký tự ‘A’ bởi Alice
- Ký tự ‘A’ trong bảng mã ASCII được tướng ứng với mã $65_{10}=1000001_2$ được mã hóa bởi hệ khóa $z_1,\ldots,z_7=0101101$
- Hàm mã hóa:

plaintext x_i:	1000001	= ‘A’	(ASCII symbol)
key stream z_i:	0101101		
ciphertext y_i:	1101100	= ‘l’	(ASCII symbol)

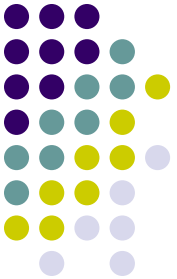
- Hàm giải mã:

ciphertext y_i:	1101100	= ‘l’	(ASCII symbol)
key stream z_i:	0101101		
plaintext x_i:	1000001	= ‘A’	(ASCII symbol)

11. Mã hóa One-time Pad(OTP)

- **Định nghĩa 1** :*Một hệ mật được coi là an toàn không điều kiện khi nó không thể bị phá ngay cả với khả năng tính toán không hạn chế.*
- OTP xuất hiện từ đầu thế kỉ 20 và còn có tên gọi khác là Vernam Cipher, OTP được mệnh danh là cái chén thánh của ngành mã hóa dữ liệu.
- OTP là thuật toán duy nhất chứng minh được về lý thuyết là không thể phá được ngay cả với tài nguyên vô tận (tức là có thể chống lại kiểu tấn công brute-force).
- Để có thể đạt được mức độ bảo mật của OTP, tất cả những điều kiện sau phải được thỏa mãn:
  - ✓ Độ dài của chìa khóa phải đúng bằng độ dài văn bản cần mã hóa.
  - ✓ Chìa khóa chỉ được dùng một lần.
  - ✓ Chìa khóa phải là một số ngẫu nhiên thực.

11. Mã hóa One-time Pad(OTP)



- **Định nghĩa 2:** Trong hệ mã hóa OTP ta có
  |P|=|C|=|K| với

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

$$\text{encrypt} \rightarrow e_{k_i}(x_i) = x_i + k_i \bmod 2.$$

$$\text{decrypt} \rightarrow d_{k_i}(y_i) = y_i + k_i \bmod 2.$$

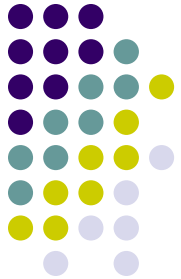
11. Mã hóa One-time Pad(OTP)

- Mới nghe qua có vẻ đơn giản nhưng trong thực tế những điều kiện này khó có thể thỏa mãn được. Giả sử Alice muốn mã hóa chỉ 10MB dữ liệu bằng OTP, cô ta phải cần một chìa khóa có độ dài 10MB. Để tạo ra một số ngẫu nhiên lớn như vậy Alice cần một bộ tạo số ngẫu nhiên thực (TRNG - True Random Number Generator). Các thiết bị này sử dụng nguồn ngẫu nhiên vật lý như sự phân rã hạt nhân hay bức xạ nền vũ trụ. Hơn nữa việc lưu trữ, chuyển giao và bảo vệ một chìa khóa như vậy cũng hết sức khó khăn.

- Dễ dàng hơn, Alice cũng có thể dùng một bộ tạo số ngẫu nhiên ảo (PRNG - Pseudo Random Number Generator) nhưng khi đó mức độ bảo mật giảm xuống gần bằng zero hay cùng lắm chỉ tương đương với một thuật toán dòng như RC4 mà thôi.

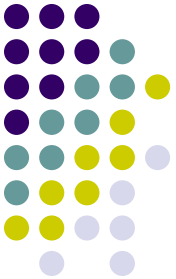
- Do có những khó khăn như vậy nên việc sử dụng OTP trong thực tế là không khả thi.

12. Lý thuyết thông tin



- **Kỹ thuật lộn xộn và rườm rà (Confusion and Diffusion)**
- Theo Shannon, có hai kỹ thuật cơ bản để che dấu sự dư thừa thông tin trong thông báo gốc, đó là: sự lộn xộn và sự rườm rà.

12. Lý thuyết thông tin



- **Kỹ thuật lộn xộn (Confusion)**: che dấu mối quan hệ giữa bản rõ và gốc. Kỹ thuật này làm thất bại các cố gắng nghiên cứu bản mã để tìm kiếm thông tin dư thừa và thống kê mẫu. Phương pháp dễ nhất để thực hiện điều này là thông qua **kỹ thuật thay thế**. Một hệ mã hoá thay thế đơn giản, chẳng hạn hệ mã dịch vòng Caesar, dựa trên nền tảng của sự thay thế các chữ cái của bản rõ, nghĩa là chữ cái này được thay thế bằng chữ cái khác

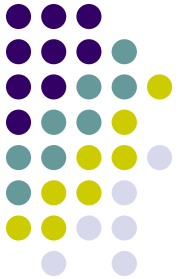
12. Lý thuyết thông tin

- **Kỹ thuật rườm rà (Diffusion)**: làm mất đi sự dư thừa của bản rõ bằng cách tăng sự phụ bản mã vào bản rõ (và khóa). Công việc tìm kiếm sự dư thừa của người thám mã sẽ rất mất thời gian và phức tạp. Cách đơn giản nhất tạo ra sự rườm rà là thông qua việc đổi chỗ (hay còn gọi là **kỹ thuật hoán vị**).
- Thông thợờng các hệ mã hiện đại thường kết hợp cả hai kỹ thuật thay thế và hoán vị để tạo ra các thuật toán mã hóa có độ an toàn cao hơn.

13. Lý thuyết độ phức tạp

Lý thuyết thông tin đã cho chúng ta biết rằng một thuật toán mã hoá có thể bị bại lộ. Còn lý thuyết độ phức tạp cho biết khả năng bị thám mã của một hệ mã mật.

- **Độ an toàn tính toán** :
- **Định nghĩa:**
- *Một hệ mật được gọi là an toàn về mặt tính toán nếu có một thuật toán tốt nhất để phá nó thì cần ít nhất N phép toán, với N là một số rất lớn nào đó.*
- **2.2. Độ an toàn không điều kiện**
- **Định nghĩa 1:**
- *Một hệ mật được coi là an toàn không điều kiện khi nó không thể bị phá ngay cả với khả năng tính toán không hạn chế.*



Chương 3: Chuẩn mã dữ liệu DES (Data Encryption Standard)

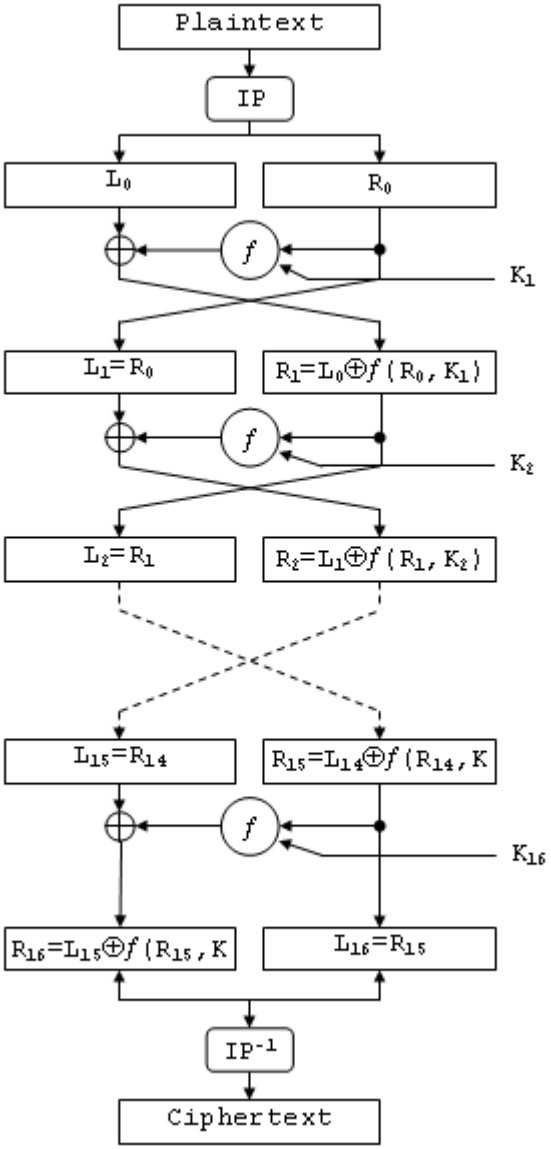
1.Giới thiệu chung về DES

- Ngày 13/5/1973 ủy ban quốc gia về tiêu chuẩn của Mỹ công bố yêu cầu về hệ mật mã áp dụng cho toàn quốc. Điều này đã đặt nền móng cho chuẩn mã hóa dữ liệu, hay là DES.
- Lúc đầu Des được công ty IBM phát triển từ hệ mã Lucifer, công bố vào năm 1975.
- Sau đó Des được xem như là chuẩn mã hóa dữ liệu cho các ứng dụng.

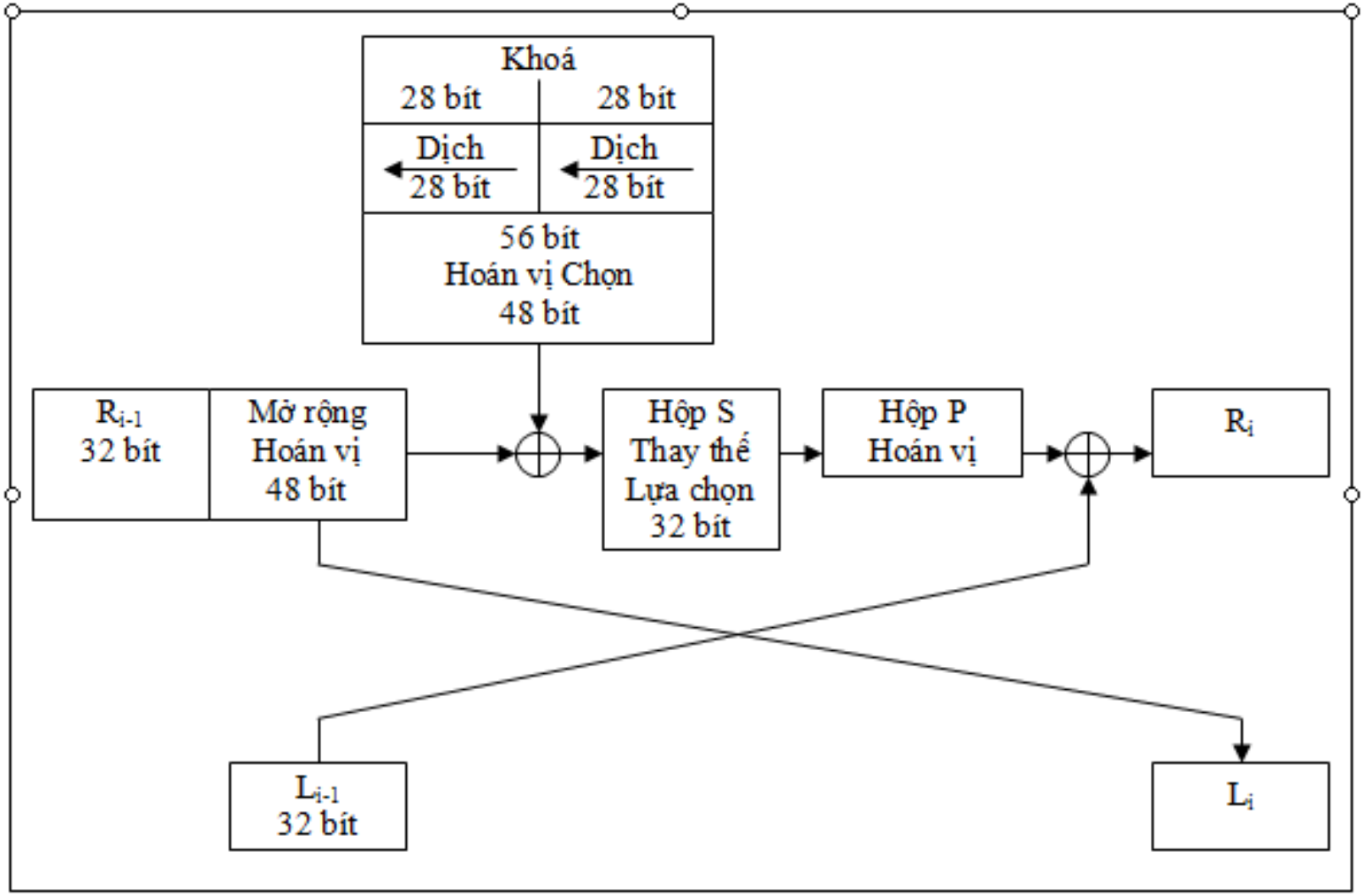
2. Đặc điểm của thuật toán DES

- DES là thuật toán mã hóa khối, độ dài mỗi khối là 64 bit .
- Khóa dùng trong DES có độ dài toàn bộ là 64 bit. Tuy nhiên chỉ có 56 bit thực sự được sử dụng; 8 bit còn lại chỉ dùng cho việc kiểm tra.
- Des xuất ra bãn mã 64 bit.
- Thuật toán thực hiện 16 vòng
- Mã hoá và giải mã được sử dụng cùng một khoá.
- DES được thiết kế để chạy trên phần cứng.

3. Mô tả thuật toán



3. Mô tả thuật toán



Một vòng lặp DES

3. Mô tả thuật toán

Thuật toán được thực hiện trong 3 giai đoạn:

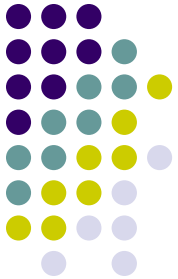
- Cho bản rõ x (64bit) được hoán vị khởi tạo IP (Initial Permutation) tạo nên xâu bit x_0.

  $x_0=IP(x)=L_0R_0$

  L_0 là 32 bit đầu tiên của x_0.

  R_0 là 32 bit cuối của x_0.

3. Mô tả thuật toán

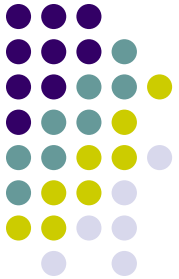


Bộ chuyển vị IP

Hoán vị khởi đầu nhằm đổi chỗ khối dữ liệu vào , thay đổi vị trÌ của các bÌt trong khối dữ liệu vào. VÌ dụ, hoán vị khởi đầu chuyển bÌt 1 thành bÌt 58, bÌt 2 thành bÌt 50, bÌt 3 thành bÌt 42,...

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

3. Mô tả thuật toán



- Từ L_0 và R_0 sẽ lặp 16 vòng, tại mỗi vòng tÌnh:

  $L_i=R_{i-1}$

  $R_i=L_{i-1}\oplus f(R_{i-1},K_i)$ với i= 1, 2,…,16

  với:

  $\oplus$ là phép XOR của hai xâu bit:

  $0 \oplus 0=0$, $1 \oplus 1=0$

  $1 \oplus 0=1$, $0 \oplus 1=1$

  f là hàm mà ta sẽ mô tả sau.

  K_i là các xâu có độ dài 48 bit được tÌnh như là các hàm của khóa K.

  K_1 đến K_{16} lập nên một lịch khóa.

3. Mô tả thuật toán

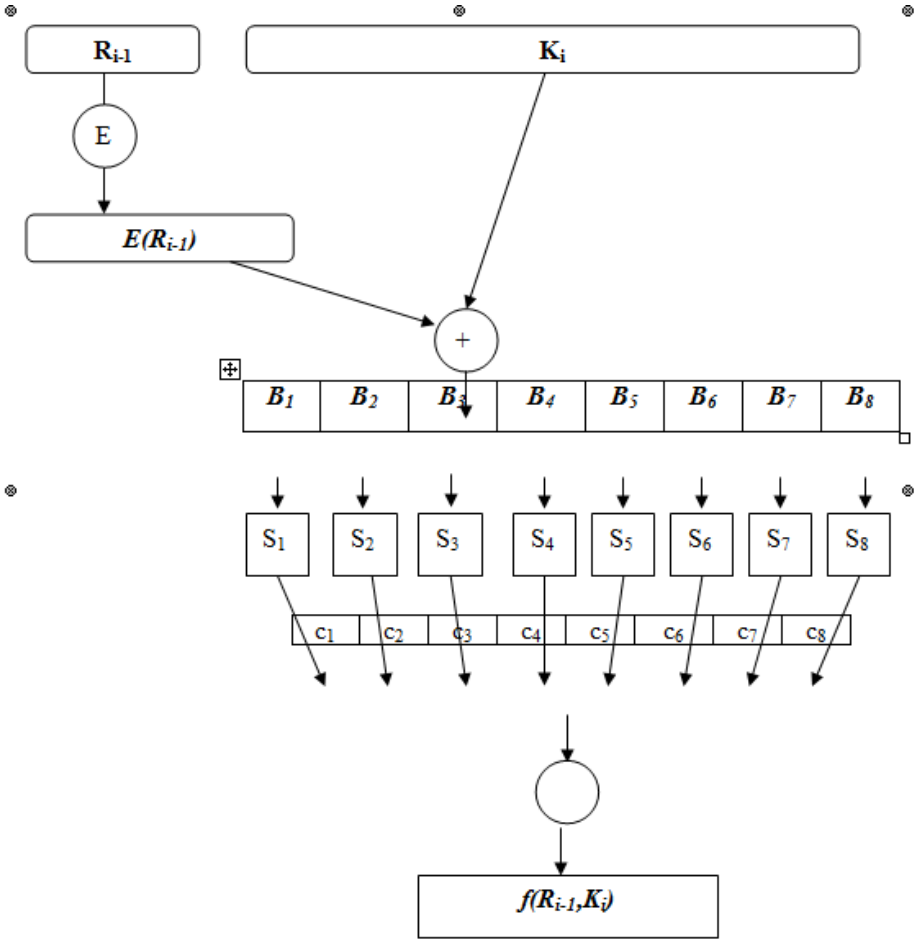
1. Tại vòng thứ 16, R16 đổi chỗ cho L16. Sau đó ghép 2 nửa R16, L16 cho đi qua hoàn vị nghịch đảo của hoàn vị IP sẽ tÌnh được bản mã. Bản mã cũng có độ dài 64 bÌt.

Hoán vị IP^{-1}

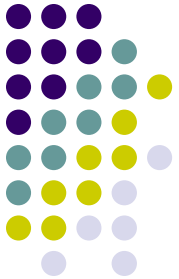
40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

3. Mô tả thuật toán

Hàm f



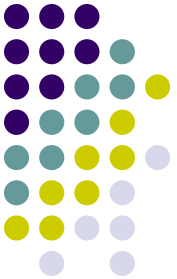
Hàm f



Hàm f lấy đối số đầu là xâu nhập R_{i-1} (32 bit) đối số thứ hai là K_i (48 bit) và tạo ra xâu xuất có độ dài 32 bit. Các bước sau được thực hiện.

- Đối số đầu R_{i-1} sẽ được "mở rộng" thành xâu có độ dài 48 bit tương ứng với hàm mở rộng E cố định. $E(R_i)$ bao gồm 32 bit từ R_i, được hoán vị theo một cách thức xác định, với 16 bit được tạo ra 2 lần.

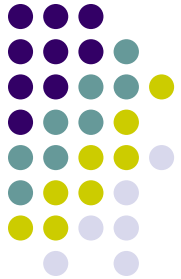
Hàm f



32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

Hàm mở rộng E

Hàm f



- Tình $E(R_{i-1}) \oplus K_i$ kết quả được một khối có độ dài 48 bit. Khối này sẽ được chia làm 8 khối $B=B_1B_2B_3B_4B_5B_6B_7B_8$. Mỗi khối này có độ dài là 6 bÌt.

- Bước kế tiếp là cho các khối Bi đi qua hộp S_i sẽ biến một khối có độ dài 6 bit thành một khối C_i có độ dài 4 bÌt.

S-box

- Mỗi hộp S-box là một bảng gồm 4 hàng và 16 cột được đánh số từ 0. Như vậy mỗi hộp S có hàng 0,1,2,3. Cột 0,1,2,…,15. Mỗi phần tử của hộp là một số 4 bÌt. Sáu bÌt vào hộp S sẽ xác định số hàng và số cột để tìm kết quả ra.

- Mỗi khối Bi có 6 bÌt kÌ hiệu là b1, b2, b3, b4, b5 và b6. BÌt b1 và b6 được kết hợp thành một số 2 bÌt, nhận giá trị từ 0 đến 3, tương ứng với một hàng trong bảng S. Bốn bÌt ở giữa, từ b2 tới b5, được kết hợp thành một số 4 bÌt, nhận giá trị từ 0 đến 15, tương ứng với một cột trong bảng S.

S-box

Hộp S1

14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

Hộp S2

15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9

S-box

Hộp S3

10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12

Hộp S4

7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14

S-box

Hộp S5

2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3

Hộp S6

12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13

S-box

Hộp S7

4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12

Hộp S8

13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

S-box

Ví dụ: Ta có B1=011000 thì b_1b_6=00 (xác định r=0), $b_2b_3b_4b_5$=1100 (xác định c=12), từ đó ta tìm được phần tử ở vị trÌ (0,12) --> S1(B1)=0101 (tương ứng với số 5).

$b_2b_3b_4b_5$=1100

b_1b_6=00

14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

Hộp S1

- Mỗi xâu xuất 4 bit của các hộp S được đưa vào các Cj tương ứng: Cj = Sj(Bj) (1<=j<=8).

Hàm f

- Xâu bit C = $C_1C_2C_3C_4C_5C_6C_7C_8$ có độ dài 32 bit được hoán vị tương ứng với hoán vị cố định P. Kết quả có P(C)= $f(R_i,K_i)$.

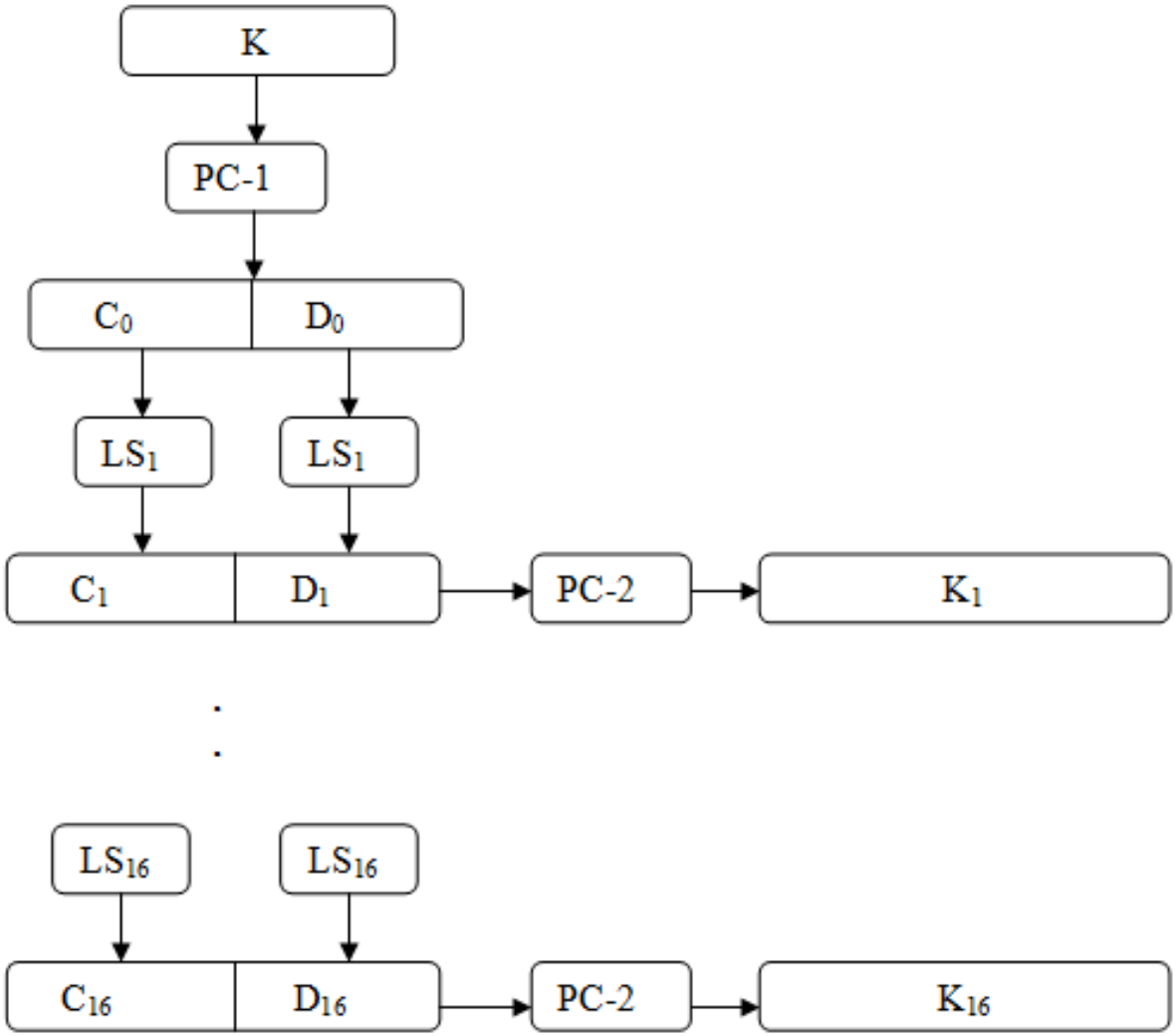
Hoán vị P

16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

Khóa K

- K là một xâu có độ dài 64 bit trong đó 56 bit dùng làm khóa và 8 bit dùng để kiểm tra sự bằng nhau (phát hiện lỗi).
- Các bit ở các vị trÌ 8, 16,…, 64 được xác định, sao cho mỗi byte chứa số lẻ các số 1, vì vậy từng lỗi có thể được phát hiện trong mỗi 8 bit.
- Các bit kiểm tra sự bằng nhau là được bỏ qua khi tÌnh lịch khóa.

Sơ đồ tính khóa K1, K2, …, K16



Khóa K

Quá trình tạo các khóa con (subkeys) từ khóa K được mô tả như sau:

Cho khóa K 64 bit, loại bỏ các bit kiểm tra và hoán vị các bit còn lại của K tương ứng với hoán vị cố định PC-1. Ta viết PC1(K) = C_0D_0, với C_0 bao gồm 28 bÌt đầu tiên của PC-1(k) và D_0 là 28 bit còn lại.

Trong đó bảng số bít dịch trái tại mỗi vòng là:

Vòng i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Số bít dịch	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

Khóa K

Các hoán vị cố định PC-1 và PC-2:

Bảng trật tự khoá (PC-1):

57	49	41	33	25	17	9	1	58	50	42	34	26	18
10	2	59	51	43	35	27	19	11	3	60	52	44	36
63	55	47	39	31	23	15	7	62	54	46	38	30	22
14	6	61	53	45	37	29	21	13	5	28	20	12	4

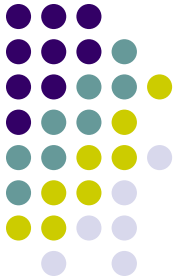
Bảng trật tự nén(PC-2):

14	17	11	24	1	5	3	28	15	6	21	10
23	19	12	4	26	8	16	7	27	20	13	2
41	52	31	37	47	55	30	40	51	45	33	48
44	49	39	56	34	53	46	42	50	36	29	32

Giải mã

- Việc giải mã dùng cùng một thuật toán như việc mã hoá.
- Để giải mã dữ liệu đã được mã hoá, quá trình giống như mã hoá được lặp lại nhưng các chìa khoá phụ được dùng theo thứ tự ngược lại từ K_{16} đến K_1, nghĩa là trong bước 2 của quá trình *mã hoá dữ liệu đầu vào ở* trên R_{i-1} sẽ được XOR với K_{17-i} chứ không phải với K_i.

Đặc điểm của mã DES



Tính chất bù của mã DES:

DES có tÌnh chất bù:

$$E_K(P) = C \Leftrightarrow E_{\overline{K}}(\overline{P}) = \overline{C}$$

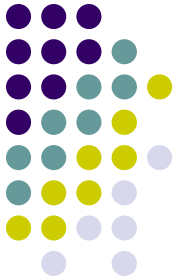
trong đó :

$\bar{A}$ là phần bù của A theo từng bÌt (1 thay bằng 0 và ngược lại).

E_K là bản mã hóa của E với khóa K. P và C là văn bản rõ (trước khi mã hóa) và văn bản mã (sau khi mã hóa).

Do tÌnh bù, ta có thể giảm độ phức tạp của tấn công duyệt toàn bộ xuống 2 lần (tương ứng với 1 bÌt) với điều kiện là ta có thể lựa chọn bản rõ.

Đặc điểm của mã DES



Các khóa yếu trong mã Des:

Ngoài ra DES còn có 4 khóa yếu (weak keys). Khi sử dụng khóa yếu thì mã hóa (E) và giải mã (D) sẽ cho ra cùng kết quả:

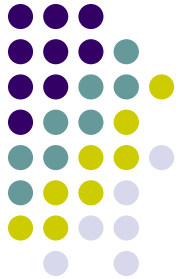
$$E_K(E_K(P)) = P \text{ or equivalently, } E_K = D_K$$

Bên cạnh đó, còn có 6 cặp *khóa nửa yếu* (semi-weak keys). Mã hóa với một khóa trong cặp, *K*1, tương đương với giải mã với khóa còn lại, *K*2:

$$E_{K1}(E_{K2}(P))=P \text{ or equivalently } E_{K1}=D_{K2}$$

Tuy nhiên có thể dễ dàng tránh được những khóa này khi thực hiện thuật toán, có thể bằng cách thử hoặc chọn khóa một cách ngẫu nhiên. Khi đó khả năng chọn phải khóa yếu là rất nhỏ.

Đặc điểm của mã DES



Triple DES:

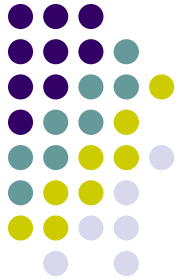
Triple-DES chỈnh là DES với hai chìa khoá 56 bit. Cho một bản tin cần mã hoá, chìa khoá đầu tiên được dùng để mã hoá DES bản tin đó.

Kết quả thu được lại được cho qua quá trình giải mã DES nhưng với chìa khoá là chìa khoá thứ hai.

Bản tin sau qua đã được biến đổi bằng thuật toán DES hai lần như vậy lại được mã hoá DES một lần nữa với chìa khoá đầu tiên để ra được bản tin mã hoá cuối cùng.

Quá trình mã hoá DES ba bước này được gọi là Triple-DES.

Đề Kiểm Tra



Môn: ATBMTT **Lớp: KHMT1K3** **Thời gian: 120’**

- Cho bản rõ mang nội dung: *x=“0123D56789ABCDE8”.*
- Cho khoá *K=183457799B3CDFF2*

Trong hệ cơ số 16, Thực hiện mã hóa văn bản rõ trên theo thuật toán DES

Xin chân thành cảm ơn!