An toàn và bảo mật thông tin

Giáo viên: TS. Lê Thị Anh

Sdt/zalo: 0976621138

Tổng quan môn học

- **Mã học phần: IT6001**
- **Số tín chỉ: 3(2.5;0.5;0)**
- **Bộ môn phụ trách: Kỹ thuật và mạng máy tính**
- **Đánh giá: 02 bài kiểm tra thường xuyên 1, 2; 01 bài tập lớn thi hết môn.**
- **Tài liệu học tập:**
  - **Tài liệu chính: Giáo trình bảo mật an toàn thông tin – Khoa CNTT, Đại học Công nghiệp Hà Nội**

I. Tổng quan về an toàn thông tin

Một số thống kê về tình hình an toàn thông tin

Một số thống kê về An ninh mạng trong báo cáo “Security Report 01/24/22” của hãng bảo mật Checkpoint.

Năm 2021, tổng các cuộc tấn công vào các mạng doanh nghiệp tăng 50% mỗi tuần so với năm 2020.

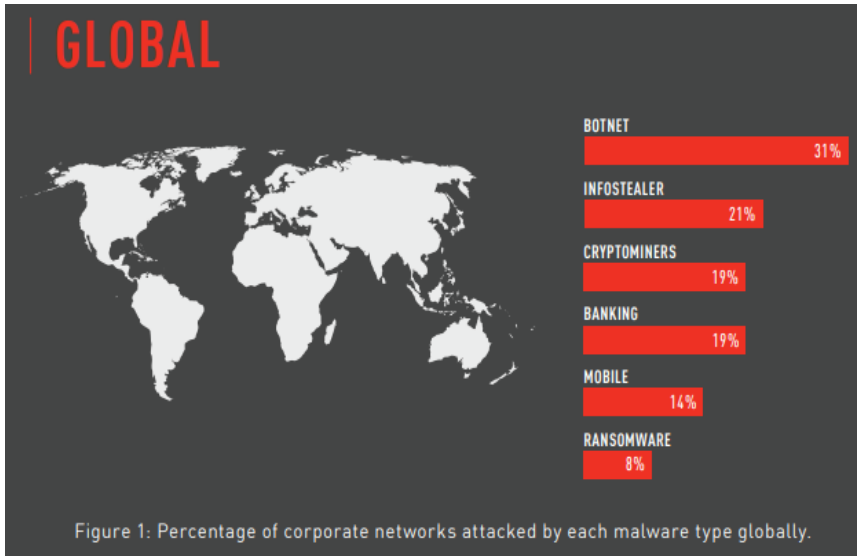


Figure 1: Percentage of corporate networks attacked by each malware type globally.

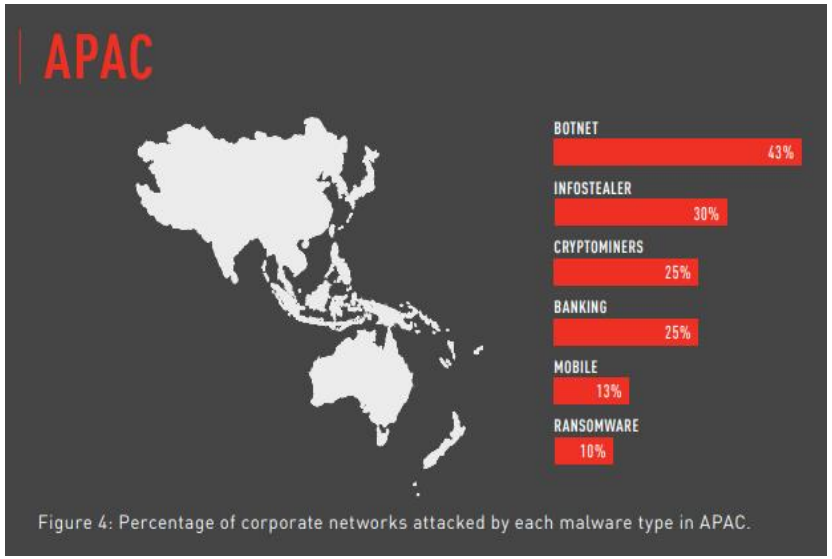


Figure 4: Percentage of corporate networks attacked by each malware type in APAC.

0.1. Một số thống kê về tình hình an toàn thông tin

Một số thống kê về An ninh mạng trong báo cáo “Security Report 01/24/22” của hãng bảo mật Checkpoint.

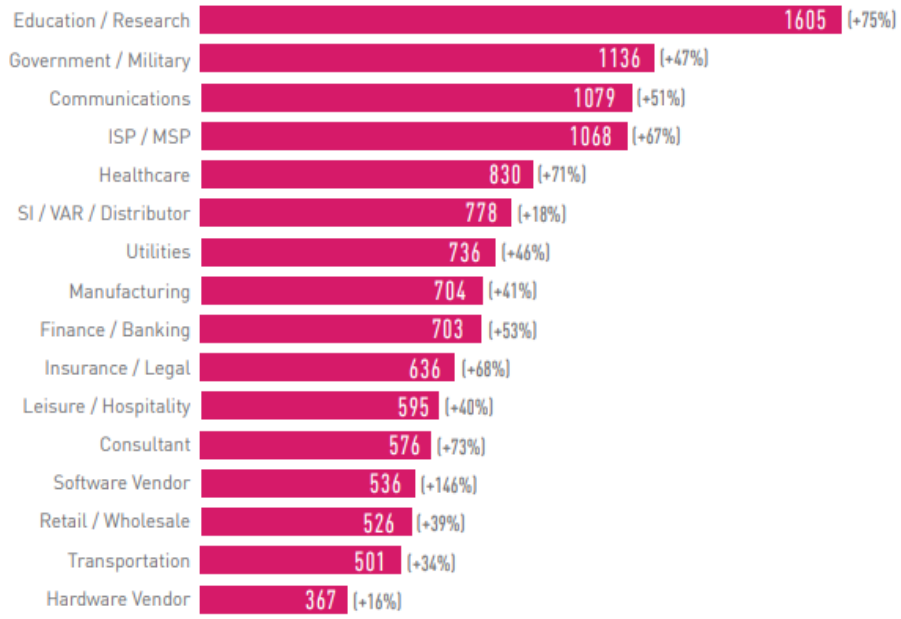
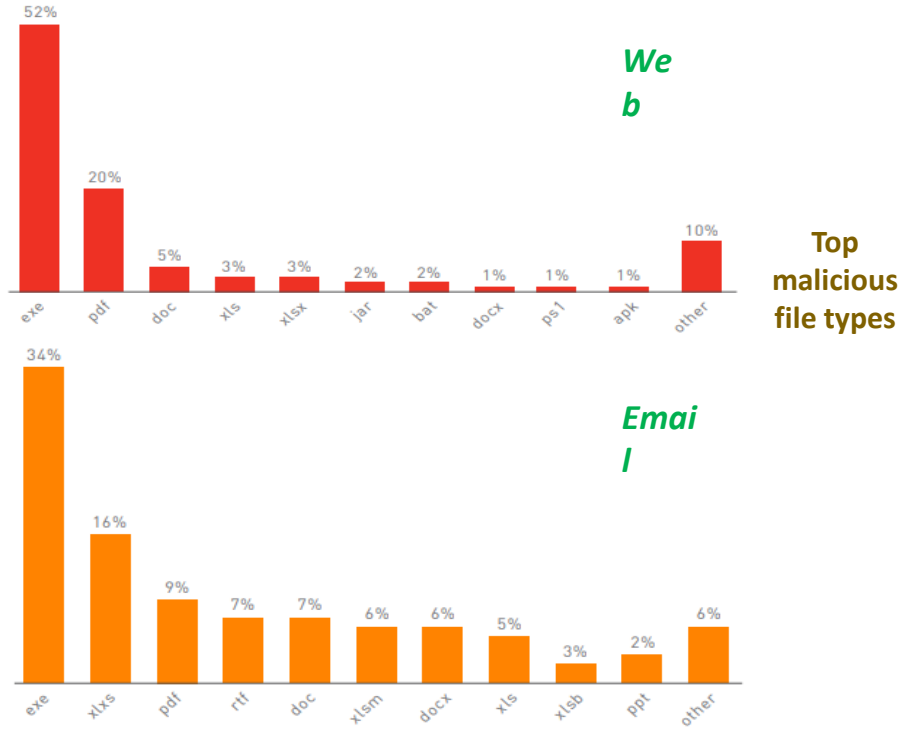


Figure 6: Average weekly attacks per organization by Industry 2021, compared to 2020.



Top malicious file types

0.1. Một số thống kê về tình hình an toàn thông tin

Việt Nam: *Luật An ninh mạng được Quốc hội thông qua năm 2018 và chính thức có hiệu lực từ 01/01/2019, với 7 chương, 43 điều quy định về hoạt động bảo vệ an ninh quốc gia, bảo đảm trật tư, an toàn xã hội trên không gian mạng, bên cạnh đó là trách nhiệm của cơ quan, tổ chức, cá nhân có liên quan. (**tập trung điều 2, 8, 19, 41, 42**)*

Số máy tính Việt Nam bị nhiễm 5 dòng mã độc phổ biến năm 2022

Năm 2022, thiệt hại do mã độc máy tính gây ra đối với người dùng Việt Nam ở mức 21,2 nghìn tỷ (tương đương 883 triệu USD) → Mức thiệt hại nhóm thấp so với thế giới (toàn cầu 1000 tỷ USD).

Lần đầu tiên sau hơn 10 năm Bkav thực hiện thống kê, con số thiệt hại ghi nhận giảm so với các năm trước đó.

Việt Nam tăng 25 bậc về chỉ số an toàn an ninh mạng GCI, cho thấy nỗ lực của Chính phủ và giới an ninh mạng trong nước.

Một số thống kê về tình hình toàn thông tin

Dữ liệu về mã độc được lấy từ bản đồ mối đe dọa trên mạng toàn cầu của Checkpoin từ tháng 1 đến tháng 12 năm 2021 của hãng: https://threatmap.checkpoint.com/

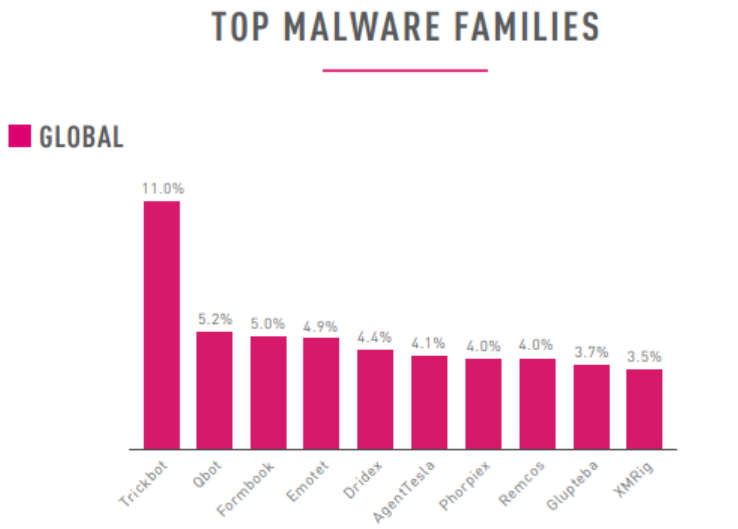


Figure 10: Most prevalent malware globally.
Percentage of corporate networks attacked by each malware family.

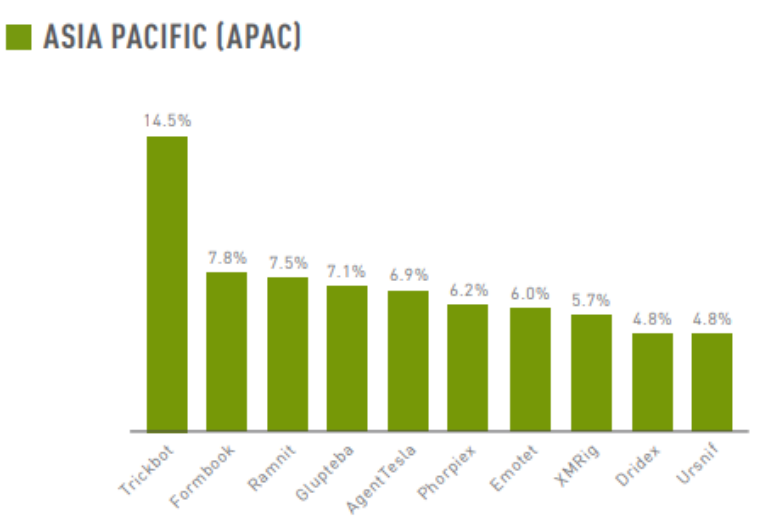


Figure 13: Most prevalent malware in APAC.

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 ngơời 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 người 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.

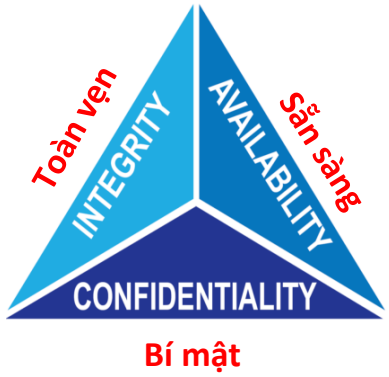
3. Các mối đe doạ đối với một hệ thống và các biện pháp ngăn chặn

- **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 của an toàn thông tin

- ***Computer Security*:** The protection afforded to an automated information system in order to attain the applicable objectives of preserving the integrity, availability, and confidentiality of information system resources (includes hardware, software, firmware, information/data, and telecommunications).

- ***An toàn mạng máy tính***: Sự bảo vệ dành cho hệ thống thông tin tự động nhằm đạt được các mục tiêu đó là duy trì tính toàn vẹn, tính sẵn sàng (tính khả dụng) và tính bí mật của tài nguyên hệ thống thông tin (bao gồm phần cứng, mềm, phần sụn, thông tin/dữ liệu và viễn thông).



Hình 1. Tam giác CIA

Ba nguyên tắc cốt lõi này phải dẫn đường cho tất cả các hệ thống an ninh mạng

4. Mục tiêu An toàn thông tin

Tính bí mật: là sự ngăn ngừa việc tiết lộ trái phép những thông tin quan trọng, nhạy cảm. Gồm 2 nội dung là Bí mật về dữ liệu và Quyền riêng tư.

→ Đối với an ninh mạng thì tính bí mật rõ ràng là điều đầu tiên được nói đến và nó thường xuyên bị tấn công nhất.

Tính toàn vẹn: Là sự phát hiện và ngăn ngừa việc sửa đổi trái phép về dữ liệu, thông tin và hệ thống, do đó Bảo đảm sự chính xác về dữ liệu vaf hệ thống. Gồm có toàn vẹn về dữ liệu và toàn vẹn của hệ thống:

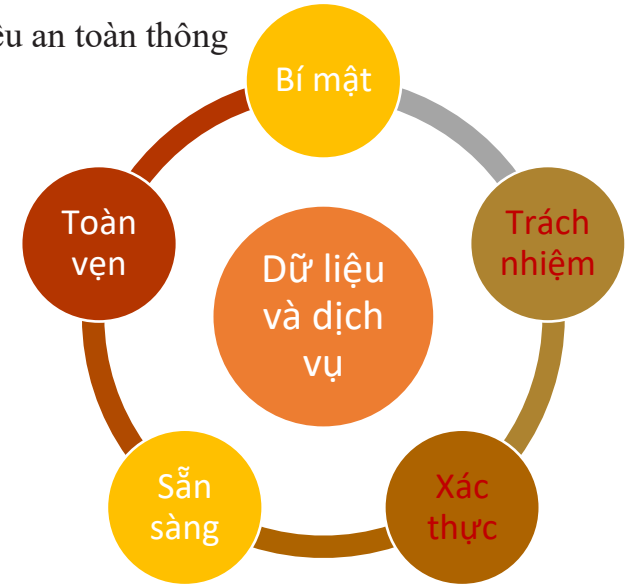
→ Toàn vẹn dữ liệu: Đảm bảo rằng dữ liệu và các chương trình chỉ được thay đổi theo bởi người được cấp quyền.

→ Tính toàn vẹn của hệ thống: Đảm bảo rằng một hệ thống thực hiện chức năng dự kiến của nó một cách nguyên vẹn, không bị thao túng trái phép một cách có chủ ý hoặc vô ý.

Tính sẵn sàng: Đảm bảo truy cập và sử dụng thông tin kịp thời và đáng tin cậy. Mất tính sẵn sàng là sự gián đoạn truy cập hoặc gián đoán sử dụng thông tin hoặc gián đoạn sử dụng hệ thống thông tin.

4. Mục tiêu của an toàn thông tin

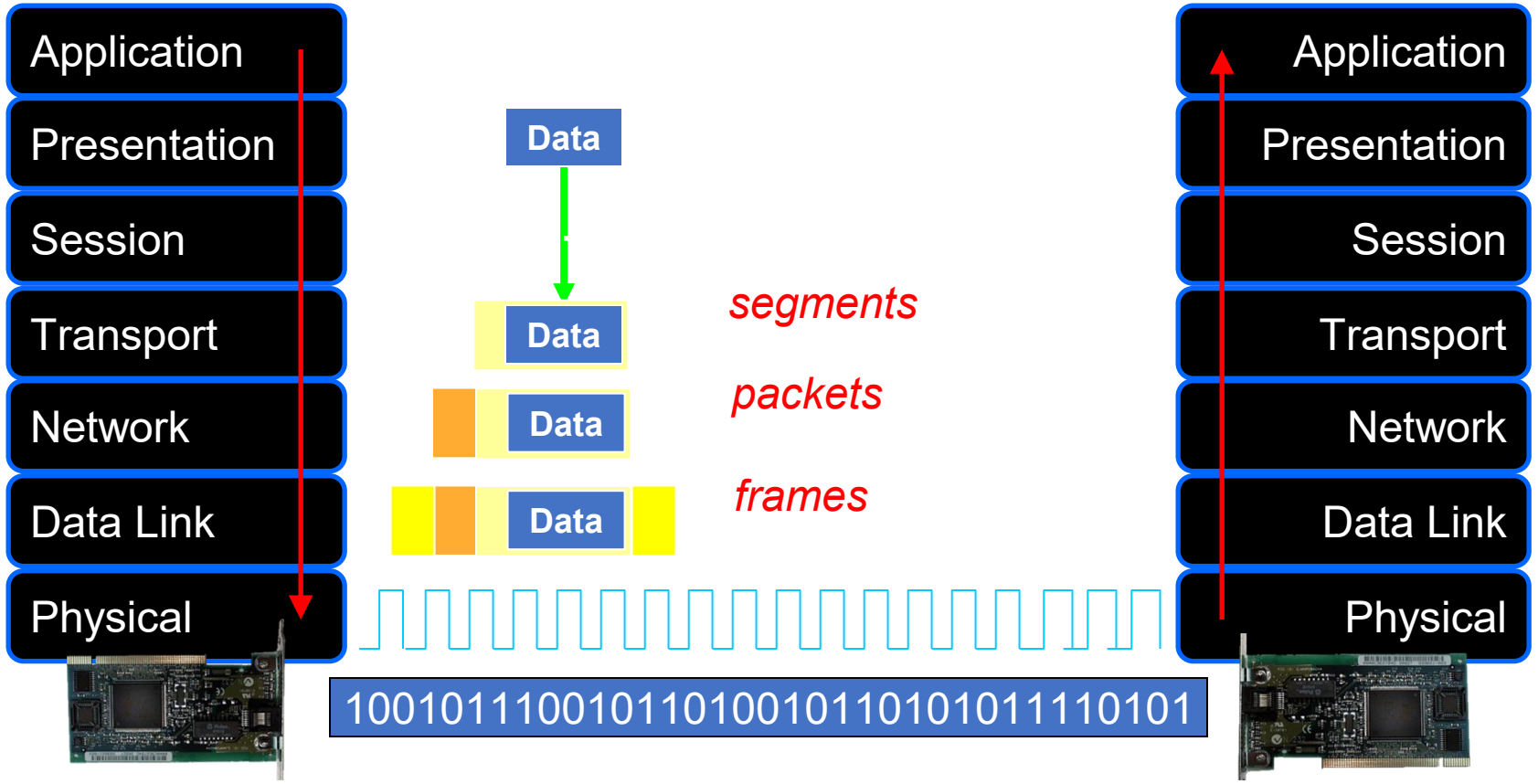
Một bức tranh hoàn chỉnh để xác định mục tiêu an toàn thông tin được đề xuất gồm 5 yếu tố:



Trách nhiệm: Mục tiêu an ninh quy định các hành động của một thực thể phải được quy một cách duy nhất về thực thể đó. Điều này hỗ trợ chống từ chối, ngăn chặn, cách ly lỗi, phát hiện và ngăn chặn xâm nhập, phục hồi sau hành động và hành động pháp lý.

Tính xác thực: Thể hiện thuộc tính được xác minh và có độ tin cậy; độ tin cậy vào tính hợp lệ của việc truyền thông, tin nhắn hoặc người khởi tạo tin nhắn

5. Mô hình OSI


Application
Presentation
Session
Transport
Network
Data Link
Physical
Data
Data
Data
Data
segments
packets
frames
10010111001011010010110101011110101
Application
Presentation
Session
Transport
Network
Data Link
Physical

6. Các loại tấn công an toàn thông tin

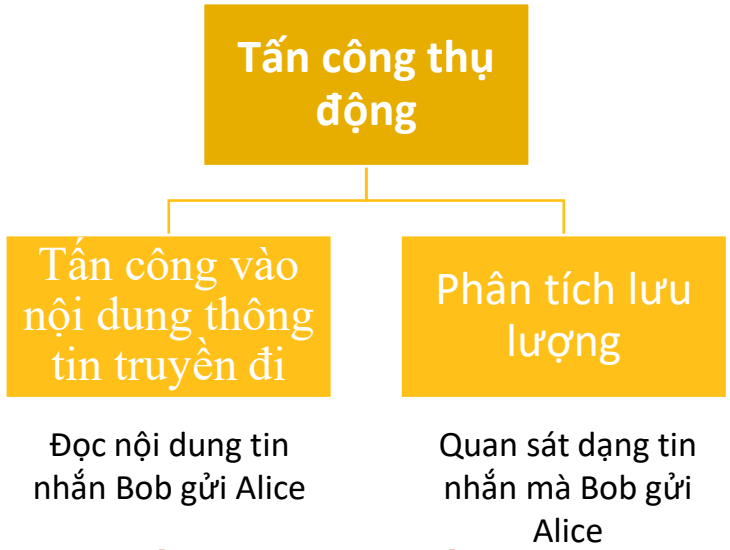
Tấn công an toàn

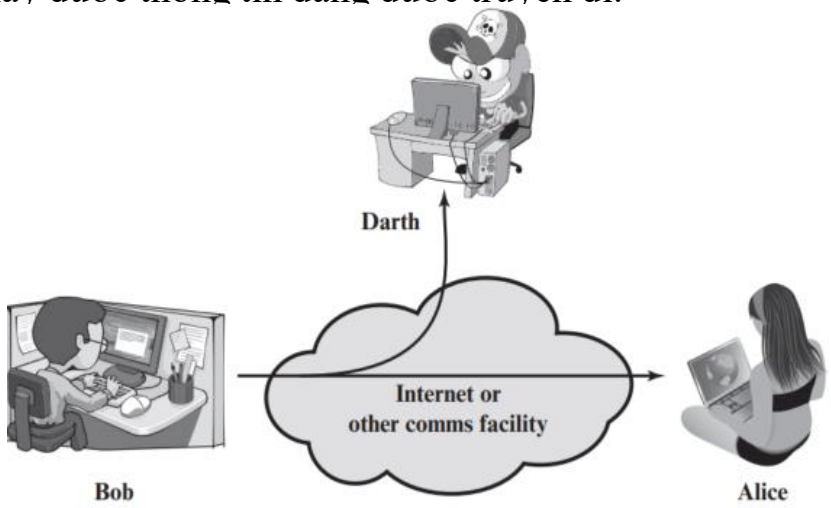
Có 2 loại hình tấn công an ninh chính được sử dụng trong cả X.800 (đây là kiến trúc bảo mật cho hệ thống OSI đươc ITU quy định), tiêu chuẩn RFC 4949 (RFC viết tắt của Request for comment, bao gồm các thuật ngữ bảo mật Internet).

- Tấn công thụ động: là cuộc tấn cống cố gắng tìm hiểu hoặc sử dụng thông tin từ hệ thống nhưng không ảnh hưởng đến tài nguyên của hệ thống.
- Tấn công chủ động: là cuộc tấn mà attacker cố gắng thay đổi tài nguyên hệ thống hoặc ảnh hưởng đến hoạt động của các hệ thống đó

6. Các loại tấn công an toàn thông tin

Tấn công thụ động: Các cuộc tấn công bị động có bản chất là nghe lén hoặc giám sát đường truyền dữ liệu. Mục tiêu là lấy được thông tin đang được truyền đi.



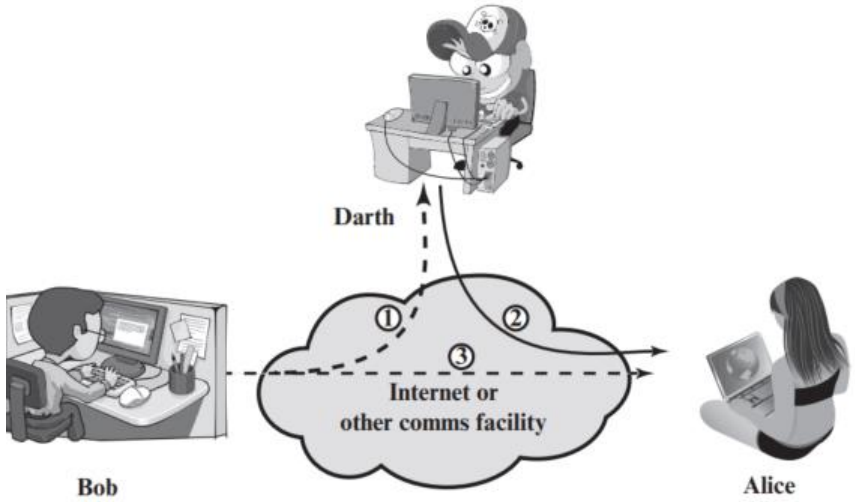


Các cuộc tấn công thụ động rất khó phát hiện do không tạo ra sự thay đổi gì về dữ liệu

Để đối phó với các cuộc tấn công này, chúng ta phải có các kỹ thuật phòng ngừa (như mã hóa) hơn là phát hiện.

6. Các loại tấn công an toàn thông tin

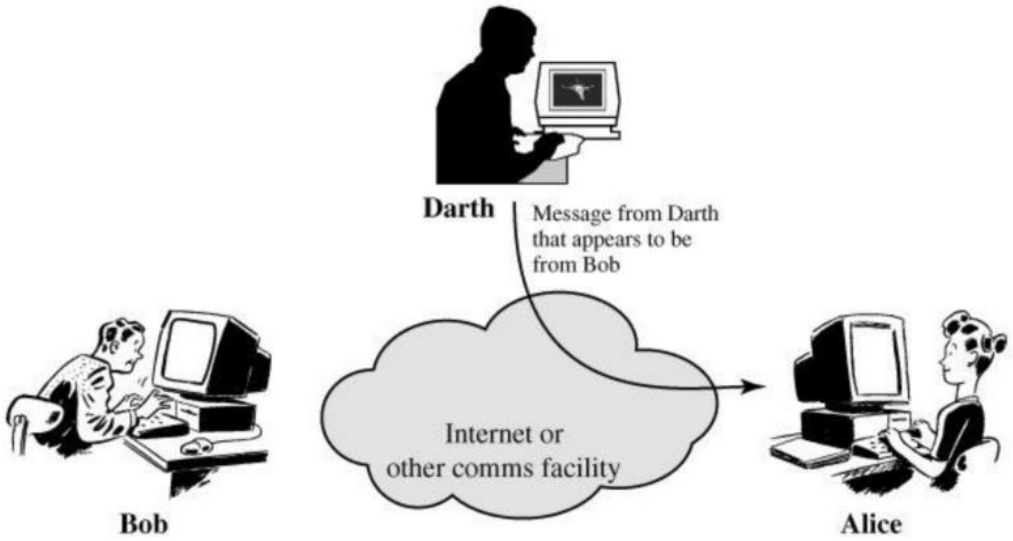
Tấn công chủ động: liên quan đến việc sửa đổi luồng dữ liệu hoặc tạo luồng già vả có thể được chia thành 4 loại: giả mạo, phát lại, sửa đổi thông tin, và từ chối dịch vụ.



6. Các loại tấn công an toàn thông tin

Tấn công chủ động

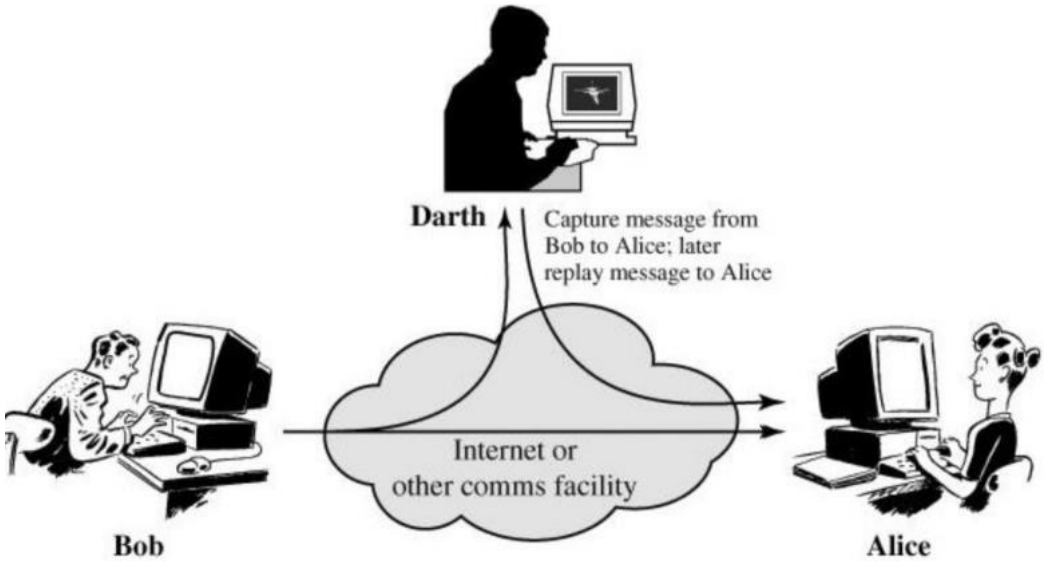
Giả mạo: Diễn ra khi một thực thể giả vờ một thực thể khác (2) – tin nhắn từ Darth tới Alice nhưng lại giả vờ là từ Bob



6. Các loại tấn công an toàn thông tin

Tấn công chủ động

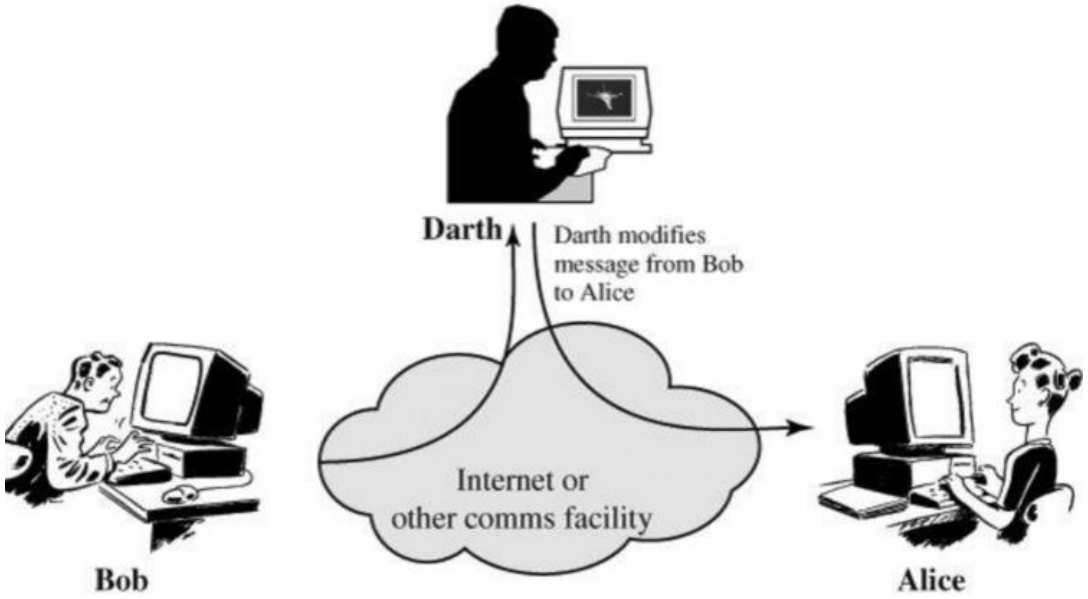
- **Phát lại**: Liên quan đến việc nắm bắt thụ động dữ liệu và truyền lại sau đó tạo ra hiệu ứng không xác thực (1,2,3) – Darth bắt gói tin từ Bob tới Alice; sau đó phát lại tin nhắn tới Alice



6. Các loại tấn công an toàn thông tin

Tấn công chủ động

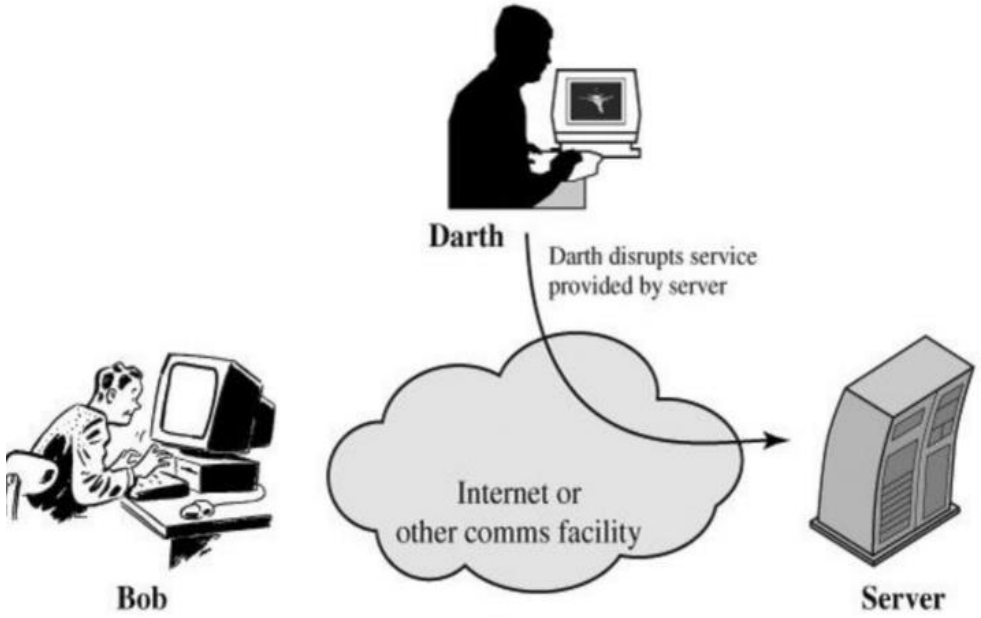
- **Sửa đổi**: tin nhắn bị sửa lại một phần hoặc tin nhắn gửi đi bị trễ để tạo ra hiệu ứng không xác thực (1, 2) – Darth sửa tin nhắn mà Bob gửi cho Alice.



6. Các loại tấn công an toàn thông tin

Tấn công chủ động

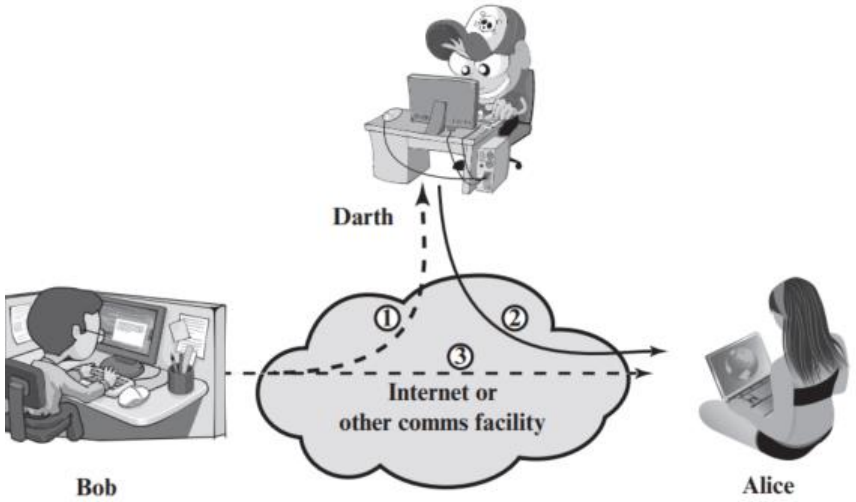
- **Từ chối dịch vụ**: là ngăn chặn hoặc cản trở việc sử dụng hoặc quản lý các phương tiện truyền thông (3 – Darth sẽ ngắt dịch vụ được cung cấp bởi máy chủ).



6. Các loại tấn công an toàn thông tin

Tấn công chủ động:

- Các cuộc tấn công chủ động thể hiện các đặc điểm ngược lại của các tấn công bị động.
- Có rất nhiều lỗ hổng vật lý, phần mềm và mạng tiềm ẩn → rất khó để ngăn chặn hoàn toàn tấn công chủ động
- Mục tiêu là phát hiện các cuộc tấn công chủ động và khắc phục sự cố



7. An toàn thông tin bằng mật mã

Mật mã là một ngành khoa học chuyên nghiên cứu các phương pháp truyền tin bí mật.

Mật mã bao gồm : Lập mã và phá mã.

- **Lập mã hay** 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

7. 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

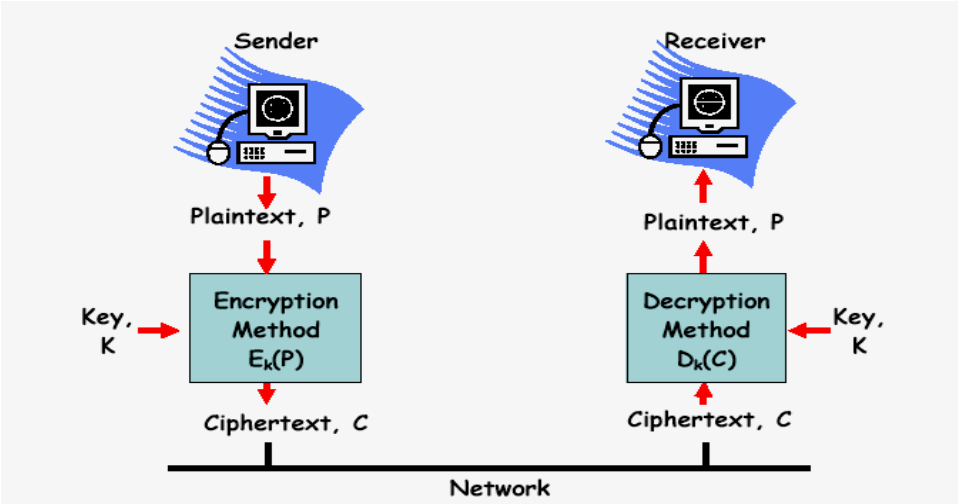


Figure 2: The basic cryptographic techniques

7. 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.

8. 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.

8. 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 kích 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 kích 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.

8. 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 không 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

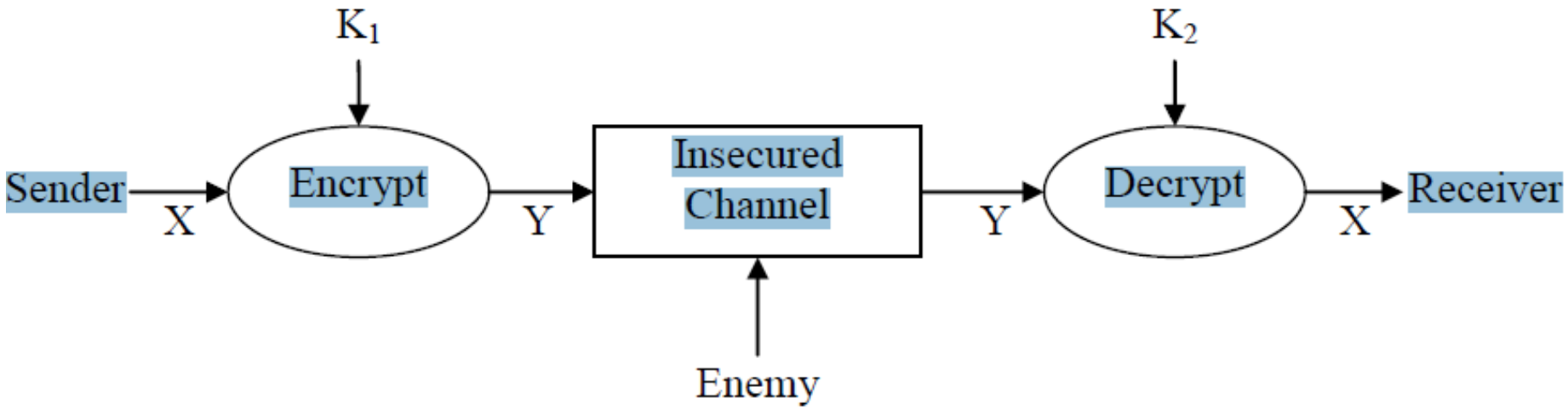
9.Tiêu chuẩn đánh giá hệ mật mã

- **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 bí 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 phí hay phí 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.

9.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 bí mật. Phân phối khóa bí mật thì chi phí sẽ cao hơn so với các hệ mật có khóa công khai. Vì vậy đây cũng là một tiêu chí khi lựa chọn hệ mật mã.

10. 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

10. 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.

11. 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

Câu hỏi ôn tập: (20 phút)

1. Tìm hiểu luật An ninh mạng 2018: tập trung điều 2, 8, 19, 41, 42
2. Lấy ví dụ về các tấn công thụ động và chủ động?
3. Kể tên các ứng dụng của mã hóa?

CƠ SỞ TOÁN HỌC CHO MẬT MÃ

Chương 2: Cơ sở toán học

- **Số học đồng dư (modulo):**
  - Cho một số nguyên ***a*** và số nguyên dương ***n*** bất kỳ, thực hiện phép chia ***a*** cho ***n*** thì thu được thương số ***q*** và phần dư ***r*** thỏa [illegible] sau:

a= q*n +r, 0<=r<n

Bảng 2. 1 Minh họa thương số và phần dư khi thực hiện phép chia a cho n

$a = 13$	$n = 4$	$13 = 3 \times 4 + 1$	$q = 3$	$r = 1$
$a = -13$	$n = 4$	$-13 = (-4) \times 4 + 3$	$q = -4$	$r = 3$

Tóm lại, cho một số nguyên *a* và số nguyên dương *n* thì ta định nghĩa *a mod n* là phần dư của phép chia *a* cho *n*. *Ví dụ: 13 mod 4 =1*

Hai số nguyên a và b được gọi là đồng dư modulo với n nếu (a mod n) = (b mod n) và được ký hiệu như sau: a ≡b (mod n). Ví dụ 13 ≡5 (mod 4).

Chương 2: Cơ sở toán học

- **Số học đồng dư (modulo): Các tính chất của đồng dư trên Z**

Tính chất	*Biểu thức*
Giao hoán	$(x + y)\ mod\ n = (y + x)\ mod\ n$ $(x \times y)\ mod\ n = (y \times x)\ mod\ n$
Kết hợp	$[(x + y) + z]\ mod\ n = [x + (y + z)]\ mod\ n$ $[(x \times y) \times z]\ mod\ n = [x \times (y \times z)]\ mod\ n$
Phân phối	$[x \times (y + z)]\ mod\ n = [(x \times y) + (x \times z)]\ mod\ n$
Số đối (-x)	Với mỗi số nguyên $x \in Z_n$ tồn tại số y sao cho $x + y \equiv 0\ (mod\ n)$
Identities	$(0 + x)\ mod\ n = x\ mod\ n$ $(1 \times x)\ mod\ n = x\ mod\ n$

Chương 2: Cơ sở toán học

- **Ước số chung lớn nhất:**
  - Ước số chung lớn nhất của 2 số nguyên *a* và *b* là số nguyên dương lớn nhất vừa là ước của a và của b, được ký hiệu là *gcd(a,b)*
  - Hai số a và b được gọi là nguyên tố cùng nhau nếu *gcd(a,b) = 1*
  - Thuật toán Ơclit tìm ước số chung lớn nhất dựa vào định lý sau: Ứng với số nguyên không âm *a* và số nguyên dương *b* bất kỳ thì:

$$\gcd(a, b) = \gcd(b, a \bmod b)$$

Ví dụ: *gcd(55,22)?*

Chương 2: Cơ sở toán học

Đoạn chương trình sau minh họa cài đặt thuật toán Ơclit để tìm ước số chung lớn nhất bằng ngôn ngữ lập trình Java.

```
int euclid(int a, int b){
 int r;
 while(true){
      if(b==0) return a;
      r = a%b;
      a = b;
      b = r;
 }
}
```

Số nguyên tố: Số nguyên $p > 1$ được gọi là số nguyên tố nếu nó chỉ có ước số là ±1 và ±\. Ví dụ 2 là số nguyên tố vì nó chỉ có các ước số là ±1 và ±2.

2	101	211	307	401
3	103	223	311	409
5	107	227	313	419
7	109	229	317	421
11	113	233	331	431
13	127	239	337	433
17	131	241	347	439
19	137	251	349	443
23	149	257	353	449
29	151	263	359	457
31	157	269	367	461
37	163	271	373	463
41	167	277	379	467
43	173	281	383	479
47	179	283	389	487
53	181	293	397	491
59	191			499
61	193			
67	197			
71	199			

73
79
83
89
97

Một số thuật toán trên Zn

- ***Tìm phần tử nghịch đảo***
  Phần tử nghịch đảo của số nguyên $a \in Z_n$ là số nguyên $x \in Z_n$ sao cho:

$$a \times x \equiv 1 (mod\ n)$$

Nếu tồn tại *x* thì nó là duy nhất và *a* được gọi là khả nghịch

Để tìm phần tử nghịch đảo của *a* với *n* nhỏ thì ta có thể sử dụng bảng nhân để tìm trực tiếp. Tuy nhiên, với *n* lớn thì phương pháp này không khả thi

Nếu *gcd(a,n) = 1* thì *a* là khả nghịch modulo *n* có nghĩa là $a \times a^{-1} \equiv 1 (\text{mod } n)$
Như vậy, ta có thể mở rộng thuật toán Ơclit để tìm ước số chung lớn nhất của *a* và *n*.
Nếu *gcd(a,n) = 1* thì thuật toán sẽ trả về phần tử nghịch đảo của *a*

Một số thuật toán trên Zn

- ***Tìm phần tử nghịch đảo***
  Phần tử nghịch đảo của số nguyên a∈ Z_n là số nguyên x ∈ Z_n sao cho:

$$a \times x \equiv 1 (mod\ n)$$

x là nghịch đảo của a, ký hiệu là $a^{-1} \bmod n$

- Ví dụ: Tính 6^{-1} mod 11=?

Phải đi tìm phần tử nghịch đảo của a là x? **a.x = 1(mod 11)**

Lập bảng:

x	x*6	x*6 mod 11
1	6	6
2	12	1

Vậy x=2= 6^{-1} mod 11

Một số thuật toán trên Zn

- ***Tìm phần tử nghịch đảo***
  Ví dụ: Tính 6^{-1} mod 13=?

Vậy x=??? = 6^{-1} mod 13

x	x*6	x*6 mod 13
1	6	6
2	12	12
3	18	5
4	24	11
5	30	4
6	36	10
7	42	3
8	48	9
9	54	2
10	60	8
11	66	1
12	72	7

Một số thuật toán trên Zn

- ***Tìm phần tử nghịch đảo***
  Ví dụ: Tính 6^{-1} mod 8=?

Vậy x=??? = 6^{-1} mod 8

x	x*6	x*6 mod 8
1	6	6
2	12	4
3	18	2
4	24	0
5	30	6
6	36	4
7	42	2

Kết luận: Không có nghịch đảo của 6 mod 8.

Để tồn tại nghịch đảo thì a và n phải là 2 số nguyên tố cùng nhau

Tính 550^{-1} mod 1759=?

Một số thuật toán trên Zn

Ta có: 550^{-1} mod 1759 =? Tức là tìm x sao cho: 550x mod 1759 = 1
Hay 550x = 1759*k + 1 hay 550 x + 1759 y =1
Tìm x và y sao cho: 550x + 1759y =1

Thuật toán Oclit tìm USCLN

*1) 1759 = 3*550 + 109*
*2) 550 = 5*109 + 5*
*3) 109 = 21 * 5 + 4*
*4) 5 = 1*4 + 1*
*5) 4 = 4*1 + 0*

Mở rộng để tìm x, y

1) 109 = 550(-3) + 1759*
2) 550 = 5(109 = 550*(-3) + 1759) + 5*
*→ 5 = 550 *16 + 1759*(-5)*
3) 550(-3) + 1759 = 21 * (550 *16 + 1759*(-5)) + 4*
→ 4 = 550(-339) + 1759 * 106*
*4) 550 *16 + 1759*(-5) = 1* 550*(-339) + 1759 * 106 + 1*
*→ 1 = 550 * 355 + 1759 * (-111)*

Như vậy: nghịch đảo của 550 mod 1759 là 355

Các bước **mở rộng** luôn thỏa

$$r_i = 550x_i + 1759y_i$$

Thuật toán Oclit mở rộng

Ta có: 550^{-1} mod 1759 =? Tức là tìm x sao cho: 550x mod 1759 = 1
Hay 550x = 1759*k + 1 hay 550 x + 1759 y =1
Tìm x và y sao cho: 550x + 1759y =1

Các bước **mở rộng** luôn thỏa
$$r_i = 550x_i + 1759y_i$$

→ Lập bảng để tìm r_i, x_i và y_i
→ i được khởi gán từ giá trị -1, …

$$r_i = r_{i-2} \; mod \; r_{i-1}$$
$$q_i = \lfloor r_{i-2} / r_{i-1} \rfloor$$
$$x_i = x_{i-2} - q_i x_{i-1}$$
$$y_i = y_{i-2} - q_i y_{i-1}$$

i	r_i	q_i	x_i	y_i
-1	1759		0	1
0	550		1	0
1	109	3	-3	1
2	5	5	16	-5
3	4	21	-339	106
4	1	1	355	-111
5	0	4	-1759	550

Một số thuật toán trên Zn

- ***Tính a^b mod n***
- Để tính a^b với a và b là các số nguyên dương. Nếu ta biểu diễn b thành số nhị phân $b_k b_{k-1} \ldots b_0$ thì b= $\sum 2^i$ (với $b_i \neq 0$)nên ta có:

$$a^b = a^{\sum_{b_i \neq 0} 2^i} = \prod_{b_i \neq 0} a^{2^i}$$

$$a^b mod\ n = \left[\prod_{b_i \neq 0} a^{2^i}\right] mod\ n = \left(\left[\prod_{b_i \neq 0} a^{2^i}\ mod\ n\right]\right) mod\ n$$

-

Một số thuật toán trên Zn

Tính a^b mod n : Do đó, ta có thể phát triển thuật toán bình phương và nhân để tính giá trị của a^b *mod n* như đoạn mã giả sau đây:

```
MODULE calcExponent(a,b,n)
 Biểu diễn b dưới dạng nhị phân: bkbk-1…b0
 f = 1
 FOR i = k DOWNTO 0 DO
      f  = (f*f) mod n
      IF bi = 1 THEN
           f = (f*a) mod n
      END_IF
 END_FOR
 RETURN f
END MODULE
```

Một số thuật toán trên Zn

- ***Tính a^b mod n***
- Ví dụ: minh họa các bước trong thuật toán bình phương và nhân để tính a^b với $a = 7$, $b = 560 = 1000110000$, $n = 561$

i	9	8	7	6	5	4	3	2	1	0
b_i	1	0	0	0	1	1	0	0	0	0
f	7	49	157	526	160	241	298	166	67	**1**

Số nguyên tố: Số nguyên $p > 1$ được gọi là số nguyên tố nếu nó chỉ có ước số là ± 1 và $\pm p$

- ***Phân tích một số ra thừa số nguyên tố***
  Phân tích một số ra thừa số nguyên tố tức là viết nó dưới dạng tích lũy thừa của các số nguyên tố. Với mọi số nguyên $a > 1$, ta có thể phân tích nó thành tích sau:

$$a = {p_1}^{a_1} \times {p_2}^{a_2} \times \ldots \times {p_k}^{a_k}$$

- Trong đó, $p_1 < p_2 < \ldots < p_k$ là các số nguyên tố dương.
  Ví dụ: $91 = 3 \times 17$; $3600 = 2^4 \times 3^2 \times 5^2$
  Ta có thể dễ dàng tìm được ước số chung lớn nhất của 2 số nguyên dương bằng cách phân tích chúng ra thừa số nguyên tố, bởi vì nếu $k = gcd(a,b)$ thì
  $k_p = min(a_p, b_p)$ với mọi giá trị của p.
- Ví dụ: Tìm $gcd(300,18)$?

Một số thuật toán trên Zn

- ***Định lý Fermat***
  Định lý Fermat phát biểu như sau: Nếu p là số nguyên tố và a là số nguyên dương không chia hết cho p thì:

$$a^{p-1} \equiv 1(mod\ p)$$

- Ví dụ p = 3 là số nguyên tố, a = 5 không chia hết cho 3. Ta có a^{p-1} mod p = 5^2 mod 3 = 25 mod 3 =1.
- Định lý Fermat có thể phát biểu cách khác như sau: Nếu p là số nguyên tố và a là số nguyên dương thì:

$$a^p \equiv a\ (mod\ p)$$

- Ví dụ: p = 5, a = 3 khi đó: a^p mod p = 3^5 mod 5 = 243 mod 5 = 3 mod 5 = a mod p

Một số thuật toán trên Zn

Định lý phần dư Trung Hoa

- Trong nhiều trường hợp ta muốn tìm cách để tăng tốc độ tính toán Modulo → Các phép toán trên modulo các số nhỏ tính nhanh nhiều so với các số lớn.
- Chính vì vậy nếu số lớn phân tích được thành tích của các số nhỏ, từng cặp là nguyên tố cùng nhau .
- Giả sử ta cần tính *A mod M*, trong đó *M* là một số lớn và có thể phân tích thành tích các số nhỏ như công thức sau:

$$M = \prod_{i=1}^{k} m_i$$

- Trong đó m_i là cặp các số nguyên tố cùng nhau, tức là $gcd(m_i,m_j) = 1$, với mọi $i \neq j$ và $1 \leq i, j \leq k$.

Một số thuật toán trên Zn

Định lý phần dư Trung Hoa

- Ta có thể biểu diễn số nguyên A bất kỳ trong Z_M bởi một bộ k thành phần và các thành phần nằm trong Z_{Mi} như sau:

$$A \leftrightarrow (a_1, a_2, a_3, \dots, a_k)$$

- Trong đó: $A \in Z_M, a_i \in Z_{m_i}$ và $a_i = A \bmod m_i$ với $1 \le i \le k$.

- Đặt $M_i = \frac{M}{m_i}$ và $c_i = M_i \times (M_i^{-1} \bmod m_i)$ với $1 \le i \le k$.

- Định lý phần dư trung hoa xác định:

$$A \bmod M = \left(\sum_{i=1}^{k} a_i \times c_i\right) \bmod M.$$

.

Một số thuật toán trên Zn

Định lý phần dư Trung Hoa

- Định lý phần dư trung hoa áp dụng cho các phép toán số học. Nếu ta có:

$$A \leftrightarrow (a_1, a_2, a_3, \dots, a_k)$$

$$B \leftrightarrow (b_1, b_2, b_3, \dots, b_k)$$

$$(A + B)\ mod\ M \leftrightarrow \big((a_1 + b_1)\ mod\ m_1, (a_2 + b_2)\ mod\ m_2, \dots, (a_k + b_k)\ mod\ m_k\big)$$

$$(A - B)\ mod\ M \leftrightarrow \big((a_1 - b_1)\ mod\ m_1, (a_2 - b_2)\ mod\ m_2, \dots, (a_k - b_k)\ mod\ m_k\big)$$

$$(A \times B)\ mod\ M \leftrightarrow \big((a_1 \times b_1)\ mod\ m_1, (a_2 \times b_2)\ mod\ m_2, \dots, (a_k \times b_k)\ mod\ m_k\big)$$

m1 = 5 m2 = 7	M	Mod 7						
		0	1	2	3	4	5	6
Mod 5	0	0	15	30	10	25	5	20
	1	21	1	16	31	11	26	6
	2	7	22	2	17	32	12	27
	3	28	8	23	3	18	33	13
	4	14	29	9	24	4	19	34

A = 16 ⇔ (1, 2)

B = 29 ⇔ (4, 1)

(A + B) MOD 35 ⇔ (1+4, 2 + 1)

- (A + B) MOD 35 = (16 + 29) mod 35 = 10
- (1+4, 2 + 1) = (0, 3)

10 ⇔ (0, 3)

Một số thuật toán trên Zn

Định lý phần dư Trung Hoa

- Áp dụng định lý phần dư trung hoa để tính 101^{59} mod 323

Đặt

A = 101^{59},

M = 323 = 17*19;

➔ m_1 = 17, m_2 = 19.

Tính theo các modul thành phần

a_1 = 101^{59} mod 17 = 16

a_2 = 101^{59} mod 19 = 5

Tính kết quả A mod M;

Khi đó M_1 = 19, M_2 = 17 và

$M_1^{-1} \, mod \, m_1$ = 19^{-1} mod 17 = 9,

$M_2^{-1} \, mod \, m_2$ = 17^{-1} mod 19 = 9,

Tính $c_i = M_i \times (M_i^{-1} \, mod \, m_i)$

c_1 = 19.9 = 171

c_2 = 17.9 = 153

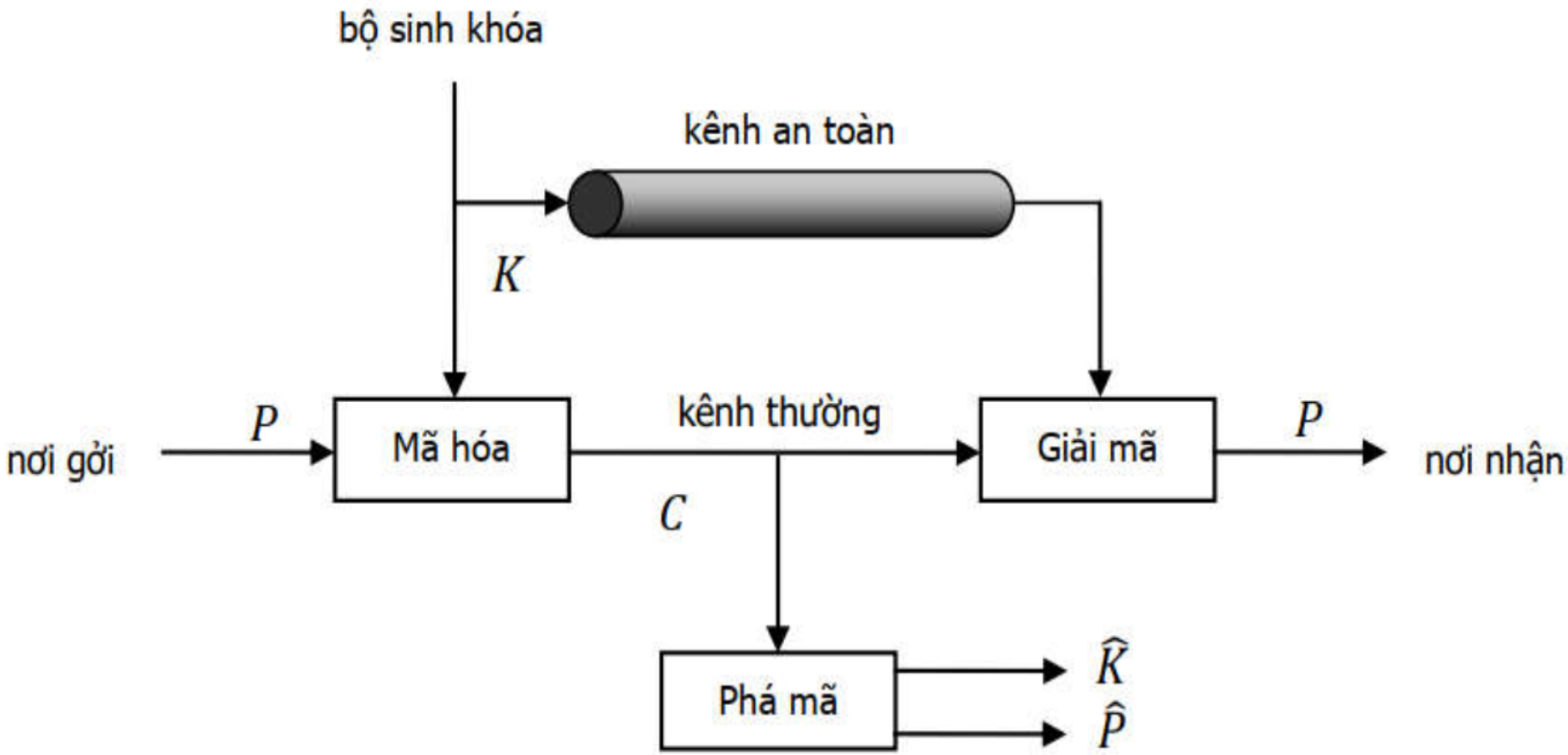
A = $(a_1c_1 + a_2c_2)$ mod M

= (16*171 + 5*153) mod 323

= 3501 mod 323 = 271

Chương 3: Các hệ mã khóa bí mật

Mô hình



Chương 3: Các hệ mã khóa bí mật

I. Hệ mã hóa cổ điển:

1. **Hệ mã hoá thay thế :**

Hệ mã hoá thay thế là hệ mã hoá trong đó mỗi ký tự của bản rõ được thay thế bằng ký tự khác trong bản mã (có thể là một chữ cái, một số hoặc một ký hiệu).

Có **4** kỹ thuật thay thế sau đây:

Thay thế đơn

Thay thế đồng âm

Thay thế đa mẫu tự

Thay thế đa sơ đồ

I. Hệ mã hóa cổ điển:

a. ***Thay thế đơn***: là hệ trong đó một ký tự của bản rõ được thay bằng một ký tự tương ứng trong bản mã. Một ánh xạ 1-1 từ bản rõ tới bản mã được sử dụng để mã hoá toàn bộ thông điệp.

b. ***Thay thế đồng âm***: giống như hệ thống mã hoá thay thế đơn, ngoại trừ một ký tự của bản rõ có thể được ánh xạ tới một trong số một vài ký tự của bản mã: sơ đồ ánh xạ 1-n (one-to-many). Ví dụ, “A” có thể tương ứng với 5, 13, 25, hoặc 56, “B” có thể tương ứng với 7, 19, 31, hoặc 42, v.v.

I. Hệ mã hóa cổ điển:

c.***Thay thế đa mẫu tự***: được tạo nên từ nhiều thuật toán mã hoá thay thế đơn. Ánh xạ 1-1 như trong trường hợp thay thế đơn, nhưng có thể thay đổi trong phạm vi một thông điệp. Ví dụ, có thể có năm thuật toán mã hoá đơn khác nhau được sử dụng; đặc biệt thuật toán mã hoá đơn được sử dụng thay đổi theo vị trí củ

d. ***Thay thế đa sơ đồ*:** là thuật toán trong đó các khối ký tự được mã hoá theo nhóm. Đây là thuật toán tổng quát nhất, cho phép thay thế các nhóm ký tự của văn bản gốc. Ví dụ, “ABA” có thể tương ứng với “RTQ”, “ABB” có thể tương ứng với “SLL”, v.v

I. Hệ mã hóa cổ điển:

2. Hệ mã Caesar:

-Hệ mã Caesar là một hệ mã hoá thay thế đơn âm làm việc trên bảng chữ cái tiếng Anh 26 ký tự (A, B, ... , Z).

-Không gian các bản rõ *P là các thông điệp được tạo từ bảng chữ cái A,* không gian các bản mã $C \equiv P$. Giả sử số phần tử của bảng chữ cái $|A| = N$.

- Để mã hóa người ta đánh số các chữ cái từ 0 tới N-1.

- Không gian khóa $k = Z_N$. *Với mỗi khóa* $K \in k$ *hàm mã hóa và giải mã một ký tự có số thứ tự là i sẽ được thực hiện như sau:*

I. Hệ mã hóa cổ điển:

2. Hệ mã Caesar:

Mã hóa: $E_K(i) = (i + k) \bmod N.$

Giải mã: $D_K(i) = (i - k) \bmod N.$

- Hệ mã Caesar với bảng chữ cái tiếng Anh sẽ có N =

Bảng 3. 1 Bảng chữ cái tiếng Anh

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

I. Hệ mã hóa cổ điển:

2. Hệ mã Caesar:

Ví dụ: Với k=3 (trường hợp đã được hoàng đế Caesar sử dụng), ký tự A được thay bằng D, B được thay bằng E, ... , W được thay bằng Z, ... , X được thay bằng A, Y được thay bằng B, và Z được thay bằng C.

Bảng chữ cái gốc:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Bảng chữ cái dùng để mã hoá:

D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

- Do đó chẳng hạn xâu "ANGLES" sẽ được mã hoá thành "DQJOHV".

I. Hệ mã hóa cổ điển:

2. Hệ mã Caesar:

- Hệ mã Caesar sử dụng phương pháp thay thế đơn âm nên có hiện tượng gọi là phụ thuộc tần suất xuất hiện của ngôn ngữ tự nhiên.
- Trên thực tế hệ mã Caesar có số khóa ít nên hoàn toàn có thể thám mã bằng cách thử tất cả các khóa có thể (kiểu tấn công Brute force).

I. Hệ mã hóa cổ điển:

3. Hệ mã Affine: cũng là hệ mã thay thế

$P = C = Z_{26}$, $K = \{(a,b) \in Z_{26} \times Z_{26},$ ước chung lớn nhất của a và 26 bằng $1\}$.

Với mỗi $k \in K$ ta có:

Hàm mã hóa $e_k(x) = ax + b \pmod{26}$

Hàm giải mã $d_k(y) = a^{-1}(y-b) \pmod{26}$.

Để việc giải mã có thể thực hiện được, yêu cầu cần thiết là hàm Affine phải là đơn ánh, tức là với bất kỳ $y \in Z_{26}$, ta muốn có đồng nhất thức sau

$ax + b \equiv y \pmod{26}$ phải có nghiệm x duy nhất

Ví dụ: Mã hóa cụm từ “HOT”

I. Hệ mã hóa cổ điển:

Ví dụ: Giả sử $P = C = Z$

- $encryption:\ e_k(x) = a \cdot x + b \bmod 26\ .$
- $key:\ k = (a, b)$ 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

I. Hệ mã hóa cổ điển:

- Mã tuyến tính là một mã thay thế có dạng

$e(x) = ax + b \pmod{26}$, trong đó $a, b \in \mathbf{Z}_{26}$.

- Giải mã: Tìm x?

$y = ax + b \pmod{26}$

$ax = y - b \pmod{26}$

$x = a^{-1}(y - b) \pmod{26}$.

- Vấn đề: Tính a^{-1}.

Để có a^{-1}, đòi hỏi (a,26)=1.

Tính a^{-1}: Thuật toán Euclide mở rộng (lập trình để tính)

I. Hệ mã hóa cổ điển:

4. Hệ mã 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

-Ví dụ: n=26, m=5 thì không gian khóa ~1.1×10^7

I. Hệ mã hóa cổ điển:

5. Hệ mã Hill:

-Phương pháp Hill (1929)

-Tác giả: Lester S. Hill

-Ý tưởng chính:

Sử dụng *m* tổ hợp tuyến tính của *m* ký tự trong plaintext để tạo ra *m* ký tự trong ciphertext

-Ví dụ:

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

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

I. Hệ mã hóa cổ điển:

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 $\mathbf{Z}_n$.

I. Hệ mã hóa cổ điển:

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.

I. Hệ mã hóa cổ điển:

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”.

I. Hệ mã hóa cổ điển:

Để giải mã ta tính khóa giải mã là ma trận ngh ịch 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 $K^{-1} = 3 * \begin{pmatrix} 5 & 23 \\ 24 & 3 \end{pmatrix} = \begin{pmatrix} 15 & 17 \\ 20 & 9 \end{pmatrix}$.

I. Hệ mã hóa cổ điển:

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.

I. Hệ mã hóa cổ điển:

6. **Hệ mã đổi chỗ (transposition cipher)**

Một hệ mã hoá đổi chỗ là hệ mã hoá trong đó các ký tự của bản rõ vẫn được giữ nguyên, nhưng thứ tự của chúng được đổi chỗ cho nhau.

Ví dụ: một hệ mã hoá đổi chỗ cột đơn giản

Bản rõ: COMPUTER GRAPHICS MAY BE SLOW BUT AT LEAST IT'S EXPENSIVE		
	COMPUTERGR	
	APHICSMAYB	
	ESLOWBUTAT	
	LEASTITSEX	
	PENSIVE	
Bản mã: CAELPOPSEEMHLANPIOSSUCWTITSBIUEMUTERATSGYAERBTX		

I. Hệ mã hóa cổ điển:

Các kỹ thuật đổi chổ:

1. *Đảo ngược toàn bộ bản rỏ*: nghĩa là bản rõ được viết theo thứ tự ngược lại để tạo ra bản mã.

   Ví dụ: bản rõ “TRANSPOSITION CIPHER” được mã hoá thành “REHPICNOITISOPSNART”.

   Nhận xét: Đây là phương pháp mã hoá đơn giản nhất vì vậy không đảm bảo an toàn.

2. *Mã hóa theo mẫu hình học*: Bản rõ được sắp xếp lại theo một mẫu hình học nào đó, thường là một mảng hoặc một ma trận hai chiều.

I. Hệ mã hóa cổ điển:

Ví dụ: bản rõ "LIECHTENSTEINER" được viết thành ma trận 3×5 theo hàng như sau:

Cột	1	2	3	4	5
Bản rõ	L	I	E	C	H
	T	E	N	S	T
	E	I	N	E	R

Nếu lấy các ký tự ra theo số thứ tự cột 2, 4, 1, 3, 5 thì sẽ có bản mã "IEICSELTEENNHTR".

Nhận xét: Hạn chế của phương pháp này là toàn bộ các ma trận ký tự phải được sinh để mã hoá và giải mã.

I. Hệ mã hóa cổ điển:

3. *Hoán vị các ký tự của bản rõ theo chu kỳ cố định d*:
   Nếu hàm f là một hàm hoán vị của một khối gồm d ký tự được biểu diễn bởi K(d,f)

   Bản rõ:

$$M = m_1 m_2 \ldots m_d m_{d+1} \ldots m_{2d}$$

   Với m_i là các ký tự , và bản rõ sẽ được mã hoá thành

$$Ek(M) = m_{f(1)} m_{f(2)} \ldots m_{f(d)} m_{f(d)+1} \ldots m_{d+f(d)}$$

   Trong đó $m_{f(1)} m_{f(2)} \ldots m_{f(d)}$ là một hoán vị của $m_1 m_2 \ldots m_d$.

I. Hệ mã hóa cổ điển:

Ví dụ: giả sử d=5 và f hoán vị dãy i=12345 thành f(i)=35142

Vị trí đầu	Vị trí hoán vị	Từ	Mã hoá
1	3	G	O
2	5	R	P
3	1	O	G
4	4	U	U
5	2	P	R

- **Mật mã Playfair** là một hệ mã hóa nhiều chữ, giảm bớt tương quan giữa văn bản mã hóa và nguyên bản bằng cách mã hóa đồng thời nhiều chữ cái của nguyên bản.
- Cơ chế hoạt động như sau: sử dụng một ma trận chữ cái 5x5 trên cơ sở một từ khóa: điền các chữ cái của từ khóa (bỏ các chữ trùng), điền những vị trí còn lại của ma trận với các chữ cái khác của bảng chữ cái; I, J có thể ở trên cùng một ô của ma trận.

- Ví dụ ma trận với từ khóa
- MONARCHY
- M O N A R C H Y B D E F G I/J K L P Q S T U V W X Z
- • Mã hóa 2 chữ cái một lúc

– Nếu 2 chữ giống nhau, tách ra bởi 1 chữ điền thêm thường là X hoặc Q Ví dụ: EE sẽ dược thay bởi EX

– Nếu 2 chữ nằm cùng hàng, thay bởi các chữ bên phải Ví dụ: EF sẽ thay bằng FG

– Nếu 2 chữ nằm cùng cột, thay bởi các chữ bên dưới Ví dụ: OF thay bằng HP

– Các trường hợp khác, mỗi chữ cái được thay bởi

MÃ HÓA DES

I. Mã hóa (Nhắc lại)

1. Giới thiệu chung về mật mã học (Cryptography)

- Mật mã học là một lĩnh vực liên quan đến các kỹ thuật ngôn ngữ và toán học để đảm bảo an toàn thông tin, cụ thể là trong thông tin liên lạc.
- Mật mã học gắn liền với quá trình mã hóa tức là chuyển đổi thông tin từ dạng "có thể hiểu được" thành dạng "không thể hiểu được" hay chuyển đổi thông tin từ "bản rõ – plain text" sang "bản mã – cipher text" và ngược lại là quá trình giải mã

I. Mã hóa (nhắc lại)

1. Giới thiệu chung về mật mã học (Cryptography)

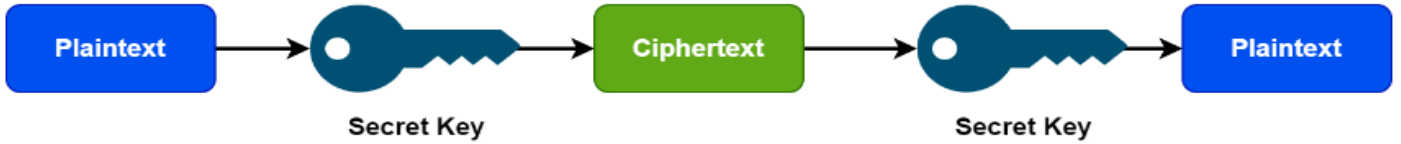
Mật mã học giúp bảo đảm các yếu tố sau cho dữ liệu:

- **Tính bí mật (*confidentiality*):** thông tin chỉ được tiết lộ cho những ai được phép
- ***Tính toàn vẹn (integrity):*** thông tin không thể bị thay đổi mà không bị phát hiện.
- ***Tính xác thực (authentication):*** người gửi (hoặc người nhận) có thể chứng minh đúng họ.
- ***Tính chống chối bỏ (non-repudiation):*** người gửi hoặc nhận sau này không thể chối bỏ việc đã gửi hoặc nhận thông tin.

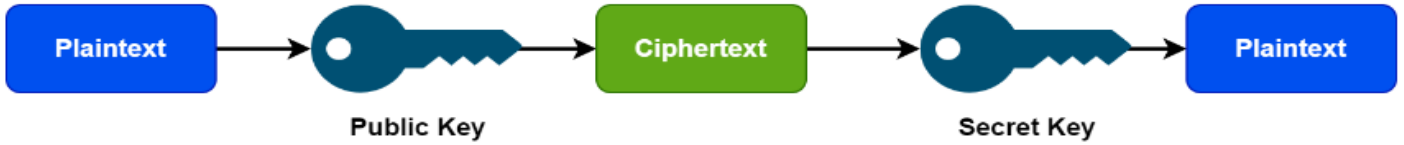
I. Mã hóa (nhắc lại)

1. Giới thiệu chung về mật mã học (Cryptography)

- **Phân loại:**
  - Loại thao tác dùng để chuyển bản rõ thành bản mã: thay thế, chuyển vị
  - Số khóa sử dụng: Khóa đơn –khóa bí mật (Mã hóa đối xứng); ứng)



Mã hóa đối xứng



Mã hóa bất đối xứng

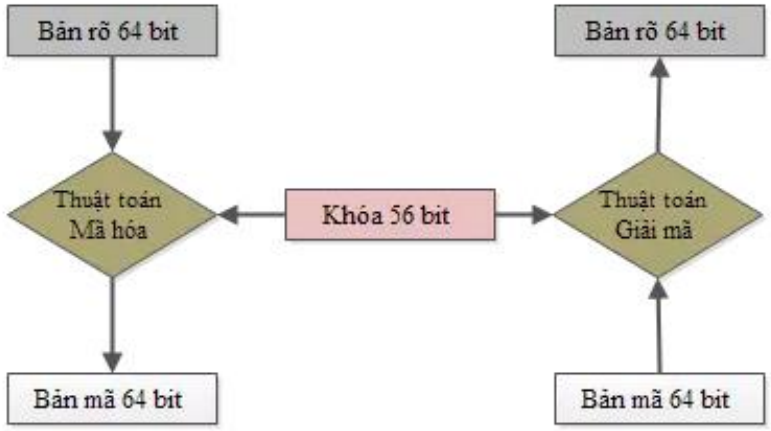
I. Mã hóa (Nhắc lại)

2. Thám mã (cryptanalysis)

- Thám mã hay còn gọi là phân tích mật mã – đây là ngành học nghiên cứu các phương thức để thu được ý nghĩa của thông tin đã được mã hóa
- Các phương pháp tấn công thám mã:
  - Tìm khóa vét cạn
  - Phân tích thống kê
  - Phân tích toán học

2.1. Mật mã DES (Data Encryption Standard)

- Ngày 13/5/1973 ủy ban quốc gia về tiêu chuẩn của Mỹ công bố yêu cầu về mật mã áp dụng cho toàn quốc → sự ra đời của DES
- Ban đầu DES được phát triển từ hệ mã Lucifer bởi công ty IBM, năm 1975
- Sau đó DES được xem như là chuẩn mã hóa dữ liệu cho các ứng dụng



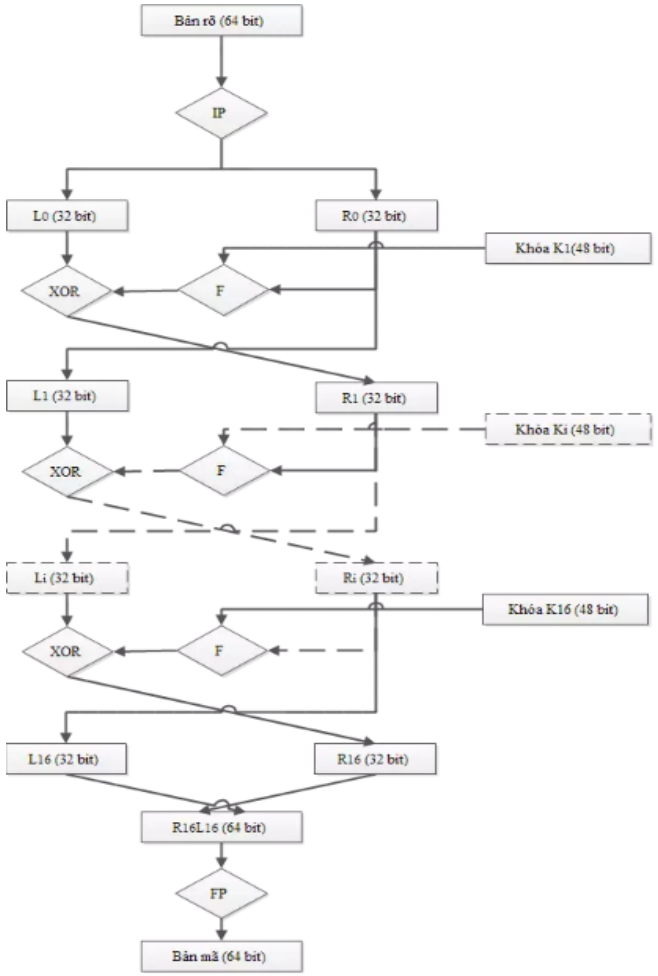
2.3. Mật mã DES (Data Encryption Standard)

- **Đặc điểm của thuật toán DES như sau:**
- DES là một thuật toán mã hóa khối, độ dài mỗi khối là 64 bít
- Khóa dùng trong DES có độ dài toàn bộ 64 bít. Tuy nhiên chỉ có 56 bít thực sự được sử dụng, 8 bít còn lại chỉ dùng cho việc kiểm tra
- DES xuất ra bản mã 64 bít
- Thuật toán thực hiện 16 vòng lặp, chỉ khác nhau về khóa trong mỗi vòng lặp đó
- Mã hóa và giải mã được sử dụng cùng một khóa

2.1. Mật mã DES (Data Encryption Standard)

- **Sơ đồ khái quát thuật toán DES**
- Với mỗi khóa K và bản rõ x, quá trình lập mã diễn ra như sau:
- Ban đầu, dùng một phép hoán vị IP (Initial Permutation), từ x với 64 bít sẽ biến thành một từ mới IP(x), từ này được chia thành 2 nửa L_0 và R_0, mỗi nửa là một từ 32 bít
- Từ cặp (L_0, R_0) sẽ dùng 15 lần những phép toán giống nhau để liên tiếp được các cặp (L_1, R_1),… (L_{15},R_{15}), sau đó dùng phép hoán vị nghịch đảo IP^{-1} cho từ đảo ngược $R_{15}L_{15}$ ta sẽ được bản mã y tương ứng.

2.1. Mật mã DES (Data Encryption Standard)



- Thông tin đầu vào là 64 bít, được chia thành 2 khối trái (L) và phải (R)
- Từ khóa 56 bít tạo ra các khóa con (subkey) gọi là Ki.
- Hàm f là một hàm hoán vị
- Trong quá trình mã hóa, dữ liệu đầu vào phải thực hiện quá trình hoán vị đầu IP (initial permutation) và hoán vị cuối (final permutation) sau vòng thứ 16
- Hàm cơ sở f cho phép đảm bảo tính bảo mật trong DES
- Cấu trúc vòng lặp DES thực hiện theo công thức sau:

$$(Li,Ri) = (R_{i-1}, L_{i-1} \text{ XOR } f(R_{i-1},K_i))$$

- Trong đó (L_i,R_i) là nửa trái và nửa phải lấy được của phép biến đổi vòng lặp thứ i

2.1. Mật mã DES (Data Encryption Standard)

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

$L_i=R_{i-1}$

$R_i=L_{i-1}\oplus f(R_{i-1},K_i)$ 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 tính như là các hàm của khóa K.

2.1. Mật mã DES (Data Encryption Standard)

- IP là một phép hoán vị vị trí của các ký tự trong mỗi từ 64 bít, từ vị trí thứ nhất đến vị trí thứ 64.
- Bảng dưới đây cho ta phép hoán vị IP, với cách biểu diễn là bít thứ nhất của IP(x) là bít thứ 58 của từ x (có 64 bít), bít thứ hai của IP(x) là bít thứ 50 của x,…
- Bảng của phép hoán vị IP^{-1} cũng được hiểu tương tự

IP

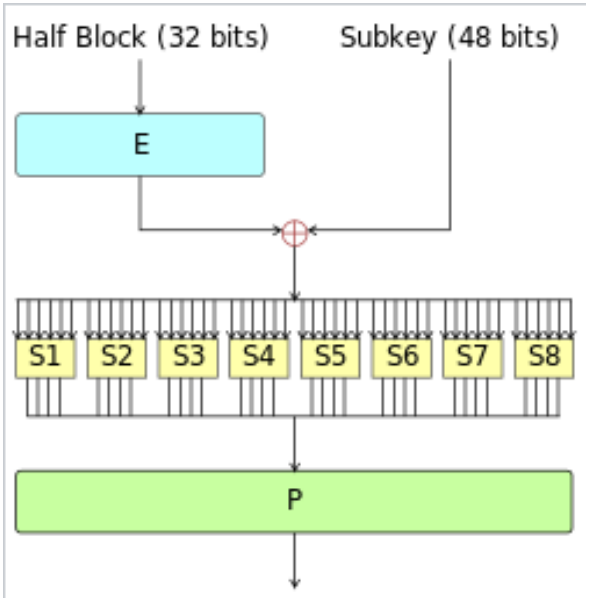
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

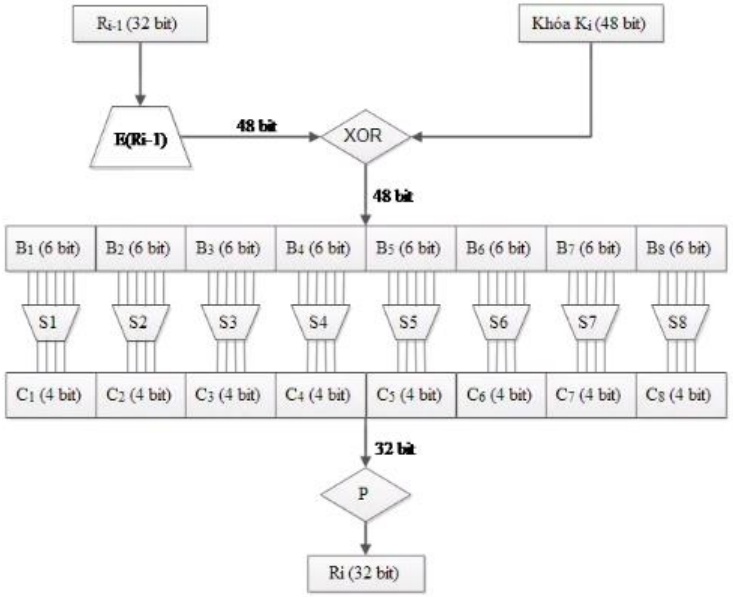
IP^{-1}

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

2.1. Mật mã DES (Data Encryption Standard)

- Sơ đồ hàm f (Feistel function):
- Hàm f lấy đầu vào là hai từ: R có 32 bít và K có 48 bít và có kết quả ở đầu ra là từ f(R,K) có 32 bít, được xác định bởi sơ đồ sau:

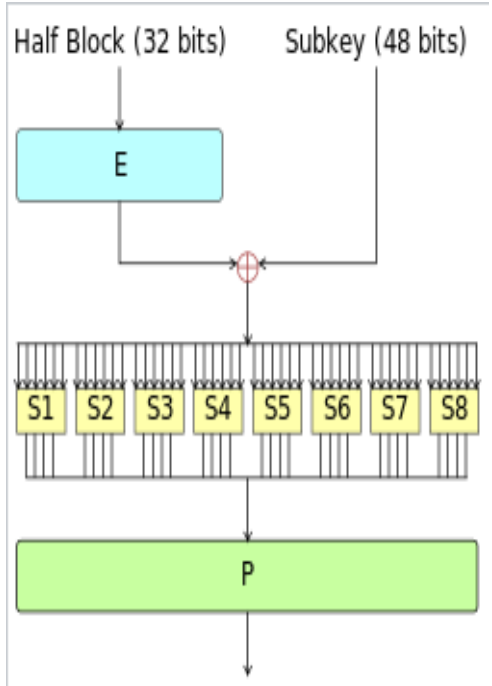




2.1. Mật mã DES (Data Encryption Standard)

- **Hàm E (Extension):** Là một phép hoán vị "mở rộng" theo nghĩa là nó biến mỗi từ R 32 bít thành từ E(R) bằng các hoán vị 32 bít của R nhưng có một số cặp bít được lặp lại để E(R) thành một từ có 48 bít.
- Cụ thể phép hoán vị "mở rộng" đó được cho bởi bảng sau:

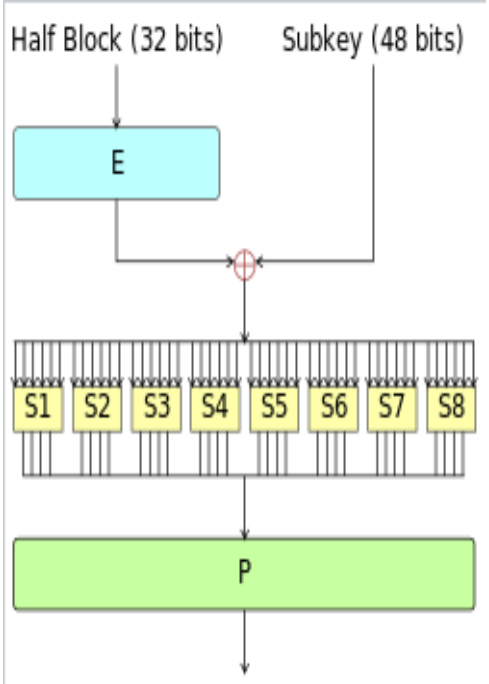
Phép hoán vị "mở rộng" E					
32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1



- Như vậy mỗi từ $R=a_1a_2....a_{32}$ sẽ biến thành $E(R)=a_{32}a_1a_2a_3a_4a_5a_4a_5a_6....a_{30}a_{31}a_{32}a_1$

2.1. Mật mã DES (Data Encryption Standard)

- Sau khi thực hiện E, E(R) sẽ được cộng (từng bít theo mod2) với K, được một từ 48 bít, chia thành 8 khối (6 bít)
- Mỗi hộp S_i (i=1,..8) là một phép thay thế, biến mỗi từ Bj 6 bít thành một từ C_j 4 bít; các hộp S_i được cho bởi bảng dưới đây với cách biểu diễn như sau:
- Cụ thể phép hoán vị "mở rộng" đó được cho bởi bảng sau:
- Mỗi từ $B_j=b_1b_2b_3b_4b_5b_6$ ứng với một vị trí (r,s) ở hàng thứ r và cột thứ s trong bảng, các hàng được đánh số thứ tự từ 0 đến 3 với biểu diễn nhị phân b_1b_6 và các cột được đánh số thứ tự từ 0 đến thứ 15 ứng với biểu diễn nhị phân $b_2b_3b_4b_5$.
- Nghĩa là r= b_1b_6; s=$b_2b_3b_4b_5$ (từ nhị phân chuyển sang thập phân)



2.1. Mật mã DES (Data Encryption Standard)

Ví dụ:

S_1(101110) = 11_d=1011_b (hàng r=10_b+1=3, cột s=0111_b+1 =8)

S_2(011000) = 12_d=1100_b (hàng r=00_b+1=1, cột s=1100_b+1 =13)

S_3(100110) =?

S là bí mật rất quan trọng trong bảo đảm tính bí mật của DES

S1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

S2	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
1	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
2	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
3	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9

S3	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
1	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
2	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
3	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12

S4	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
1	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
2	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
3	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14

S5	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
1	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
2	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
3	11	8	12	7	0	14	2	13	6	15	0	9	10	4	5	3

S6	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
1	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
2	9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
3	4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13

S7	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	4	11	2													
1	13	0	11													
2	1	4	11													
3	6	11	13													

S8	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
1	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
2	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
3	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

2.1. Mật mã DES (Data Encryption Standard)

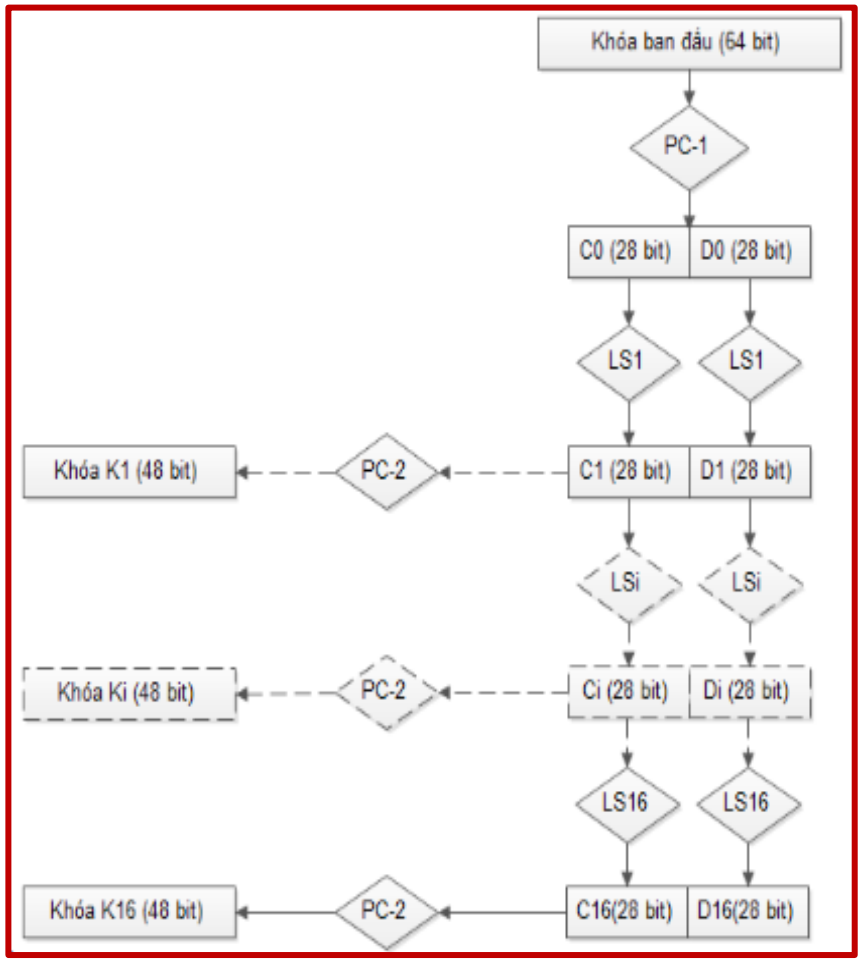
Phép hoán vị P trong sơ đồ của hàm f được cho ở bảng dưới đây:
Mỗi 4 bít đầu ra của các hộp S-box sẽ được ghép lại, theo thứ tự các hộp và được đưa vào hộp P-box. P đơn giản chỉ là phép hoán vị các bít với nhau.

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

Như vậy hàm f được xác định hoàn toàn

2.1. Mật mã DES (Data Encryption Standard)

- **Thuật toán sinh khóa K_i**: $K_1, K_2, \ldots, K_{16}$
- Các khóa con đều được sinh ra từ khóa chính của DES bằng thuật toán sinh khóa con (thuật toán G)
- LS: left shilft
- Khóa mật mã K là một từ 56 bít, ta chia thành 8 khối, mỗi khối 7 bít, ta cho thêm mỗi khối 7 bít đó một bít kiểm tra tính chẵn lẻ vào vị trí cuối để được một từ 64 bít, ta vẫn ký hiệu là K.



2.3. Mật mã DES (Data Encryption Standard)

- Trước tiên, thuật toán PC-1 biến K thành một từ 56 bít, ta chia thành 2 nửa C0, D0.
- Phép hoán vị PC-1 được xác định bởi bảng sau đây

57	49	41	33	25	17	9	1
58	50	42	34	25	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

Chú ý: trong bảng không có các số 8,16,24,32,40,48,56,64 là vị trí của những bít được thêm vào khi hình thành từ mới K

2.1. Mật mã DES (Data Encryption Standard)

- Ls_i, i=1,2,…,16 là phép chuyển dịch vòng sang trái; VD: 00000100 dịch trái 2 bít thành 00010000
- Chuyển dịch một vị trí nếu i=1,2,9,16
- Chuyển dịch hai vị trí với giá trị i còn lại

Vòng lặp	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Số lần dịch trái	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

- Phép hoán vị PC2 biến [illegible] t Ki theo bảng dưới đây

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