T

TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI

**KHOA CÔNG NGHỆ THÔNG TIN**

------🙙🕮🙛-------

**BÁO CÁO THỰC NGHIỆM**

Học phần: An ninh mạng

**Chủ đề**:

.

Giáo viên hướng dẫn : TS. Phạm Văn Hiệp

Nhóm sinh viên thực hiện :

1.

2.

3.

4.

5.

Mã Lớp học phần:

Nhóm: 9

Hà Nội - Năm 2024

**MỤC LỤC**

Danh mục các từ viết tắt…

Danh mục hình ảnh…

Danh mục bảng biểu…

…Các chương, các đề mục trong báo cáo…

Bảng phân công công việc (theo file mẫu, in ra và kẹp vào báo cáo)..

**Lời nói đầu**

**Chương 1: Tổng quan**

* 1. Tổng quan về An ninh mạng

1.1.1 An ninh mạng

An ninh mạng là một lĩnh vực quan trọng trong bảo mật thông tin, tập trung vào việc bảo vệ hệ thống máy tính, mạng lưới và dữ liệu khỏi các mối đe dọa và tấn công nhằm được đạt được các mục tiêu đó là duy trì được:

Tính toàn vẹn.

Tính sẵn sàng ( tính khả dụng).

Tính bí mật trong tài nguyên hệ thống ( phần cứng, phần mềm, phần sụn – các chương trình được cài đặt trực tiếp trên các ứng dụng ).

An ninh mạng đề cập đến các biện pháp và công nghệ được sử dụng để bảo vệ hệ thống máy tính và thông tin khỏi các cuộc tấn công, truy cập trái phép, hư hỏng và phá hoại.

Bảo mật ( an toàn thông tin ) là mức độ bảo vệ thông tin trước các mối đe dọa về “thông tin lộ”,”thông tin không toàn vẹn” và “thông tin không sẵn sàng”. Mục đích là chống lại các nguy cơ về mất an toàn thông tin như “ nguy hiểm”,”thiệt hại”,”mất mát” và các tội phạm khác.

1.1.2 Các loại mối đe dọa

Malware: Phần mềm độc hại, bao gồm virus, worm, trojan, ransomware.

Phishing: Chiến dịch lừa đảo nhằm đánh cắp thông tin cá nhân.

DDoS (Distributed Denial of Service): Tấn công làm cho dịch vụ không khả dụng bằng cách làm ngập hệ thống bằng lưu lượng truy cập.

Man-in-the-Middle (MitM): Tấn công mà kẻ tấn công can thiệp vào giao tiếp giữa hai bên.

1.1.3 Tầm quan trọng

Với sự gia tăng của các tấn công mạng và lỗ hổng bảo mật, an ninh mạng trở thành một yếu tố thiết yếu cho mọi tổ chức, từ doanh nghiệp lớn đến các tổ chức nhỏ và cá nhân.

1.1.4 Thách thức

Tăng cường nhận thức: Nâng cao nhận thức của nhân viên về an ninh mạng.

Khó khăn trong việc cập nhật công nghệ: Luôn cần thiết phải cập nhật các biện pháp bảo mật mới nhất.

Tính phức tạp của các hệ thống: Quản lý và bảo vệ các hệ thống phức tạp ngày càng khó khăn.

An ninh mạng là một lĩnh vực luôn thay đổi và yêu cầu sự chú ý liên tục để bảo vệ thông tin và hệ thống khỏi các mối đe dọa ngày càng tinh vi.

* 1. Các kiến thức cơ sở (kiến thức cơ sở về toán học, thuật toán, ngôn ngữ lập trình…)

1.2.1 Cơ sở toán học

An toàn bảo mật thông tin là một lĩnh vực quan trọng và phức tạp, dựa trên nhiều khái niệm toán học để bảo vệ dữ liệu khỏi các cuộc tấn công và truy cập trái phép. Sau đây là một số cơ sở toán học chính được sử dụng trong an toàn bảo mật thông tin:

### Mật mã học (Cryptography)

Mật mã học là việc sử dụng các kỹ thuật toán học để bảo vệ thông tin. Có hai loại chính:

* **Mật mã đối xứng (Symmetric Cryptography)**: Cả hai bên sử dụng cùng một khóa để mã hóa và giải mã dữ liệu. Ví dụ: AES, DES.
* **Mật mã bất đối xứng (Asymmetric Cryptography)**: Sử dụng một cặp khóa công khai và khóa riêng tư. Một bên dùng khóa công khai để mã hóa và bên kia dùng khóa riêng tư để giải mã. Ví dụ: RSA, ECC.

### Lý thuyết số (Number Theory)

Lý thuyết số đóng vai trò quan trọng trong mật mã học, đặc biệt là trong các hệ mật mã bất đối xứng. Một số khái niệm cơ bản bao gồm:

* **Số nguyên tố**: Các số chỉ chia hết cho 1 và chính nó. Các thuật toán như RSA dựa vào tính chất khó phân tích số nguyên tố.
* **Hàm số phi Euler (Euler's Totient Function)**: Được sử dụng trong RSA để tạo khóa.
* **Logarit rời rạc (Discrete Logarithm)**: Khái niệm này được sử dụng trong các hệ mật mã như Diffie-Hellman và ECC.

### Đại số tuyến tính (Linear Algebra)

Đại số tuyến tính được sử dụng trong nhiều thuật toán mã hóa và mã hóa lỗi:

* **Ma trận và phép biến đổi ma trận**: Được sử dụng trong mã hóa khối như AES. Ví dụ, bước MixColumns trong AES sử dụng nhân ma trận để tạo sự khuếch đại giữa các byte.
* **Mã Hamming và mã BCH**: Sử dụng trong mã hóa lỗi để phát hiện và sửa lỗi trong truyền thông dữ liệu.

### Lý thuyết nhóm (Group Theory)

Lý thuyết nhóm là cơ sở của nhiều hệ thống mật mã, đặc biệt là các hệ mật mã bất đối xứng:

* **Nhóm cyclic**: Được sử dụng trong các thuật toán như Diffie-Hellman để trao đổi khóa an toàn.
* **Nhóm con**: Sử dụng trong ECC để mã hóa và giải mã dữ liệu dựa trên các điểm trên một đường cong elip.

### Hàm băm (Hash Function)

Hàm băm là một hàm toán học chuyển đổi dữ liệu đầu vào t hành một chuỗi bit cố định (bản băm), được sử dụng trong bảo mật mật khẩu, xác thực dữ liệu và chữ ký số:

* **Hàm băm mật mã**: Đảm bảo tính toàn vẹn và không thể đảo ngược. Ví dụ: SHA-256, MD5.
* **Hàm băm mở rộng**: Sử dụng trong các giao thức xác thực và chữ ký số.

1.2.2 Các thuật toán

Mã hóa (Cryptography): Thuật toán mã hóa là một tập hợp các quy tắc và phương pháp toán học được sử dụng để chuyển đổi thông tin từ dạng dễ đọc (dạng gốc) thành dạng khó đọc (dạng mã hóa) và ngược lại. Mục tiêu chính của mã hóa là bảo vệ thông tin khỏi sự truy cập và thay đổi trái phép.

Phân loại:

Mã hóa Đối Xứng (Symmetric Encryption)

* Sử dụng cùng một khóa để mã hóa và giải mã dữ liệu.
* Ưu điểm: Tốc độ cao, hiệu quả.
* Nhược điểm: Khó quản lý và phân phối khóa.
* Các thuật toán phổ biến:
  + DES (Data Encryption Standard): DES là một trong những tiêu chuẩn mã hóa đầu tiên được chính phủ Mỹ công nhận. Nó sử dụng một khóa 56-bit để mã hóa và giải mã dữ liệu.
  + 3DES (Triple DES): 3DES là phiên bản nâng cấp của DES, nhằm khắc phục nhược điểm về độ dài khóa của DES. Nó thực hiện thuật toán DES ba lần, sử dụng ba khóa khác nhau hoặc hai khóa giống nhau.
  + AES (Advanced Encryption Standard): Nó sử dụng các khóa có độ dài 128, 192 hoặc 256 bit, cung cấp mức độ bảo mật cao hơn nhiều so với DES và 3DES.

Mã hóa Bất Đối Xứng (Asymmetric Encryption)

* Sử dụng hai khóa khác nhau: khóa công khai và khóa riêng.
* Khóa công khai được phân phối rộng rãi để mã hóa dữ liệu.
* Khóa riêng được giữ bí mật để giải mã dữ liệu.
* Ưu điểm: An toàn hơn, dễ quản lý khóa.
* Nhược điểm: Tốc độ chậm hơn mã hóa đối xứng.
* Các thuật toán phổ biến:
  + RSA: RSA là một thuật toán mã hóa bất đối xứng, có nghĩa là nó sử dụng một cặp khóa: khóa công khai và khóa riêng. Khóa công khai được chia sẻ rộng rãi để mã hóa dữ liệu, trong khi khóa riêng được giữ bí mật để giải mã.

Hàm Băm (Hash Function)

* Hàm băm mật mã là một hàm toán học một chiều, biến một khối dữ liệu có độ dài bất kỳ thành một chuỗi ký tự có độ dài cố định gọi là giá trị băm (hash).
* Không thể đảo ngược quá trình mã hóa.
* Sử dụng để: kiểm tra tính toàn vẹn của dữ liệu, xác thực mật khẩu.
* Các hàm băm phổ biến:
  + MD5: Đã bị coi là không an toàn.
  + SHA-1: Cũng đã bị tấn công thành công.
  + SHA-256, SHA-512: An toàn hơn so với MD5 và SHA-1.

**Thám mã** **(cryptanalysis):** Thám mã hay còn gọi là phân tích mật mã, 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

* 1. Nội dung nghiên cứu (trình bày lý do và các nội dung nghiên cứu…)

**Lý do chọn đề tài**

Trong bối cảnh hiện nay, bảo mật thông tin trở thành một trong những ưu tiên hàng đầu trong lĩnh vực công nghệ thông tin, đặc biệt khi các mối đe dọa an ninh mạng ngày càng gia tăng cả về số lượng lẫn mức độ tinh vi. Chuẩn mã hóa AES (Advanced Encryption Standard) được xem là một trong những giải pháp mã hóa mạnh mẽ và đáng tin cậy nhất hiện nay, được công nhận rộng rãi và sử dụng trong nhiều lĩnh vực như ngân hàng, thương mại điện tử, và truyền thông. Lý do lựa chọn đề tài này gồm:

1. Tính thực tiễn cao: AES được áp dụng trong các hệ thống bảo mật thực tế như bảo vệ dữ liệu nhạy cảm, mã hóa tập tin, và truyền thông an toàn trên Internet. Việc nghiên cứu AES giúp chúng ta hiểu rõ hơn về một công cụ đang trực tiếp bảo vệ thông tin của người dùng và doanh nghiệp.

2. Tiêu chuẩn quốc tế: AES là một chuẩn mã hóa được Viện Tiêu chuẩn và Công nghệ Quốc gia Hoa Kỳ (NIST) phê duyệt và sử dụng làm chuẩn quốc gia. Điều này cho thấy sự uy tín, hiệu quả, và tính an toàn vượt trội của AES so với các thuật toán mã hóa khác.

3. Khả năng ứng dụng rộng rãi: Với đặc điểm an toàn, hiệu quả và tốc độ cao, AES đã trở thành thuật toán mã hóa mặc định trong nhiều giao thức như SSL/TLS, IPsec, và các phần mềm bảo mật dữ liệu. Do đó, việc nghiên cứu và hiểu rõ về AES mang lại giá trị thiết thực trong lĩnh vực an ninh mạng.

4. Ý nghĩa học thuật và ứng dụng: Tìm hiểu về AES không chỉ giúp nắm vững kiến thức lý thuyết mà còn hỗ trợ triển khai các ứng dụng mã hóa dữ liệu thực tế bằng các ngôn ngữ lập trình như C++ hoặc C#. Điều này phù hợp với yêu cầu thực hành và phát triển kỹ năng giải quyết vấn đề trong ngành công nghệ thông tin.

5. Thách thức và tiềm năng phát triển: Việc nghiên cứu và triển khai AES yêu cầu kiến thức về toán học, lý thuyết mã hóa, và lập trình. Đây là một cơ hội tốt để nâng cao năng lực tư duy, kỹ năng lập trình, cũng như khả năng xử lý các bài toán liên quan đến bảo mật dữ liệu.

**Các nội dung nghiên cứu:**

#### 1. Tổng quan về mã hóa và chuẩn AES

* Khái niệm mã hóa và vai trò của mã hóa trong bảo mật thông tin.
* Lịch sử phát triển của chuẩn AES: Từ cuộc thi do Viện Tiêu chuẩn và Công nghệ Quốc gia Hoa Kỳ (NIST) tổ chức đến khi trở thành chuẩn mã hóa quốc tế.
* Các ưu điểm của AES so với các thuật toán mã hóa khác như DES, Triple DES:
  + Tính an toàn cao (không dễ bị tấn công brute force).
  + Hiệu suất tốt trên cả phần cứng lẫn phần mềm.
  + Khả năng xử lý linh hoạt với các kích thước khối và khóa khác nhau.

#### 2. Nguyên lý hoạt động của thuật toán AES

* Cấu trúc thuật toán:
  + Kích thước khối cố định (128 bit) và các tùy chọn độ dài khóa (128, 192, 256 bit).
  + Số vòng lặp (10, 12, hoặc 14) tùy thuộc vào độ dài khóa.
* Các bước xử lý chính:
  + SubBytes: Thay thế byte thông qua bảng S-box.
  + ShiftRows: Xáo trộn hàng để tạo tính khuếch tán.
  + MixColumns: Kết hợp cột để gia tăng độ phức tạp.
  + AddRoundKey: Kết hợp dữ liệu với khóa con.
* Key Expansion: Quy trình mở rộng khóa từ khóa chính ban đầu

#### 3. Ứng dụng của thuật toán AES trong thực tiễn

* Mã hóa tập tin:
  + Bảo vệ các tệp dữ liệu nhạy cảm khỏi bị truy cập trái phép.
* Truyền thông an toàn:
  + Sử dụng trong các giao thức SSL/TLS để bảo vệ dữ liệu trên Internet.
* Bảo mật dữ liệu di động:
  + Áp dụng trong bảo mật dữ liệu trên các thiết bị di động và ứng dụng IoT.
* Ngân hàng và thương mại điện tử:
  + Mã hóa thông tin thanh toán và giao dịch trực tuyến.

#### 4. Phân tích và đánh giá bảo mật của AES

* Độ an toàn của AES trước các loại tấn công:
  + Brute force.
  + Tấn công phân tích (cryptanalysis).
* Ưu điểm và hạn chế:
  + AES hiện tại không có điểm yếu nghiêm trọng, nhưng cần tiếp tục theo dõi trước các nguy cơ từ máy tính lượng tử.

#### 5. Triển khai AES bằng ngôn ngữ C++, C#

* Cách thức triển khai thuật toán AES với các bước mã hóa và giải mã dữ liệu.
* Tạo một ứng dụng nhỏ:
  + Mã hóa một tệp văn bản hoặc một chuỗi ký tự bằng AES.
  + Giải mã dữ liệu đã mã hóa để kiểm tra tính toàn vẹn.

#### 6. Đề xuất ứng dụng và mở rộng

* Các hướng cải tiến thuật toán hoặc tích hợp AES vào các hệ thống hiện tại.
* Kết hợp AES với các thuật toán khác để tăng cường bảo mật, chẳng hạn như RSA để quản lý khóa.
* Ứng dụng AES trong bảo mật dữ liệu lớn (Big Data) và điện toán đám mây

**Chương 2: Kết quả nghiên cứu**

2.1 Nghiên cứu, tìm hiểu hệ mã hóa đối xứng

Mã hóa khóa đối xứng (hay còn gọi là mã hóa khóa đồng bộ) là một thuật toán mà trong đó cả hai quá trình mã hóa và giải mã đều dùng một khóa. Để đảm bảo tính an toàn, khóa này phải được giữ bí mật. Vì thế các thuật toán mã hóa khóa đồng bộ này còn có tên gọi khác là mã hóa với khóa bí mật (secret key cryptography). Một điều cần lưu ý là khi một người mã hóa một thông điệp gốc (plaintext) thành thông điệp mã hóa bằng một khóa K (thuật toán mã hóa) (ciphertext) rồi gửi ciphertext cho đối tác thì đối tác muốn giải mã cũng cần phải có khóa K, nghĩa là trước đó hai đối tác đã phải trao đổi cho nhau chia sẻ để cùng biết được khóa K.

2.1.1 Đặc điểm của hệ mã hóa đối xứng

Cả hai bên sử dụng cùng một khóa để mã hóa và giải mã.

So với các phương pháp mã hóa không đối xứng, mã hóa đối xứng thường nhanh hơn.

Khóa phải được bảo mật và không được chia sẻ công khai.

2.1.2 Các hệ mã hóa đối xứng phổ biến

a. AES (Advanced Encryption Standard)

Một trong những chuẩn mã hóa phổ biến nhất, sử dụng các chiều dài khóa 128, 192 hoặc 256 bit.Được sử dụng rộng rãi trong các ứng dụng bảo mật dữ liệu.

b. DES (Data Encryption Standard)

Một trong những hệ mã hóa đầu tiên được công nhận, sử dụng khóa 56 bit.Hiện nay đã không còn an toàn và thường được thay thế bằng AES.

c. 3DES (Triple DES)

Cải tiến của DES, mã hóa dữ liệu ba lần với ba khóa khác nhau.Mặc dù an toàn hơn DES, nhưng vẫn chậm hơn và ít được sử dụng trong thời đại hiện nay.

d. RC4

Một thuật toán mã hóa dòng, được sử dụng trong nhiều giao thức bảo mật như SSL/TLS.Tuy nhiên, RC4 đã bị phát hiện có nhiều lỗ hổng bảo mật, nên không còn được khuyến nghị.

e. Blowfish

Một thuật toán mã hóa khối, được thiết kế để thay thế DES.Hỗ trợ chiều dài khóa từ 32 đến 448 bit, nhanh và hiệu quả.

f. Twofish

Là phiên bản kế tiếp của Blowfish, hỗ trợ chiều dài khóa lên đến 256 bit và được đánh giá cao về tính an toàn và hiệu suất.

2.1.3 Ứng dụng của mã hóa đối xứng

### a. Bảo mật dữ liệu truyền thông

Truyền dữ liệu qua mạng: Mã hóa các gói tin dữ liệu trước khi truyền đi để đảm bảo tính bảo mật, ngăn chặn sự xâm nhập và đánh cắp dữ liệu.

Các ứng dụng nhắn tin: Nhiều ứng dụng nhắn tin như WhatsApp, Signal sử dụng mã hóa đầu cuối để bảo vệ nội dung tin nhắn của người dùng.

VPN: Mã hóa toàn bộ lưu lượng truy cập internet của người dùng để đảm bảo ẩn danh và bảo vệ thông tin cá nhân.

### b. Bảo mật dữ liệu lưu trữ

Mã hóa ổ cứng: Bảo vệ dữ liệu trên ổ cứng khỏi truy cập trái phép, đặc biệt là khi máy tính bị mất hoặc bị đánh cắp.

Mã hóa các tệp tin: Mã hóa các tệp tin quan trọng như tài liệu, hình ảnh, video để đảm bảo tính bảo mật.

Mã hóa cơ sở dữ liệu: Bảo vệ dữ liệu nhạy cảm trong cơ sở dữ liệu khỏi các cuộc tấn công.

### c. Các ứng dụng khác

Bảo mật hệ điều hành: Một số hệ điều hành sử dụng mã hóa để bảo vệ các phân vùng hệ thống và dữ liệu người dùng.

Bảo mật phần mềm: Mã hóa các phần mềm để ngăn chặn việc sao chép trái phép và bảo vệ bản quyền.

Bảo mật thiết bị IoT: Bảo vệ dữ liệu truyền đi và nhận về từ các thiết bị IoT như cảm biến, máy điều khiển thông minh.

Bảo vệ dữ liệu nhạy cảm: Bảo vệ thông tin tài chính, y tế, và dữ liệu cá nhân, được sử dụng trong các giao thức như SSL/TLS để bảo vệ dữ liệu truyền qua Internet.

Bảo vệ dữ liệu trên ổ đĩa bằng cách mã hóa các tệp và thư mục.

2.1.4 Ưu và Nhược điểm

1. Ưu điểm

Ưu điểm nổi bật của mã hóa đối xứng là tốc độ lập mã, giải mã khá nhanh chóng. Hiện nay có nhiều phần mềm thương mại hỗ trợ thuật toán mã hóa đối xứng hữu hiệu và rất phổ dụng.

Ưu điểm thứ hai là tuy có nhiều nghiên cứu thám mã đã thực hiện nhưng với các thuật toán được cải tiến gần đây như 3-DES và nhất là AES thì độ bảo mật khá cao, trong thực tế việc phá mã cũng không dễ dàng.

1. Nhược điểm

Tuy vậy nhược điểm lớn nhất của thuật toán mã hóa đối xứng là vấn đề chuyển giao chìa khóa giữa các đối tác, đặc biệt là trong môi trường mở.

Như trong ví dụ nói ở đầu chương về việc trao đổi thông điệp giữa An và Bình. Bình nhận được thông điệp đã mã hóa của An, muốn giải mã được thì Bình phải có chìa khóa mã của An. An không thể chuyển giao khóa mã đồng thời với thông điệp vì như vậy thì việc mã hóa trở thành vô tác dụng.

Vì vậy An phải dùng một phương pháp nào đó để chuyển giao khóa giải mã cho Bình trước khi gửi thông điệp. Mà dù dùng phương thức thông tin nào trong môi trường mở: gửi thư, E-mail, gọi điện thoại v.v. thì vẫn có nguy cơ có người thứ ba nắm bắt được khóa mã và kết quả vẫn như nhau.

So sánh lại với 5 nguyên lý bảo mật thông tin, xét trường hợp giao dịch của 2 đối tác An và Bình. Giả sử An và Bình “hoàn toàn tin tưởng vào nhau” và trao cho nhau mã khóa đối xứng bằng một phương pháp đáng tin cậy nào đó (trao tay trực tiếp hoặc có một phương pháp nào có thể thay thế cho trao tay trực tiếp mà cũng có giá trị tương đương như thế) và sau đó hai người sử dụng mã khóa truyền các thông điệp mã hóa cho nhau.

2.2 Nghiên cứu, tìm hiểu về chuẩn AES và ứng dụng trong mã hóa dữ liệu

2.2.1 Sự ra đời của AES

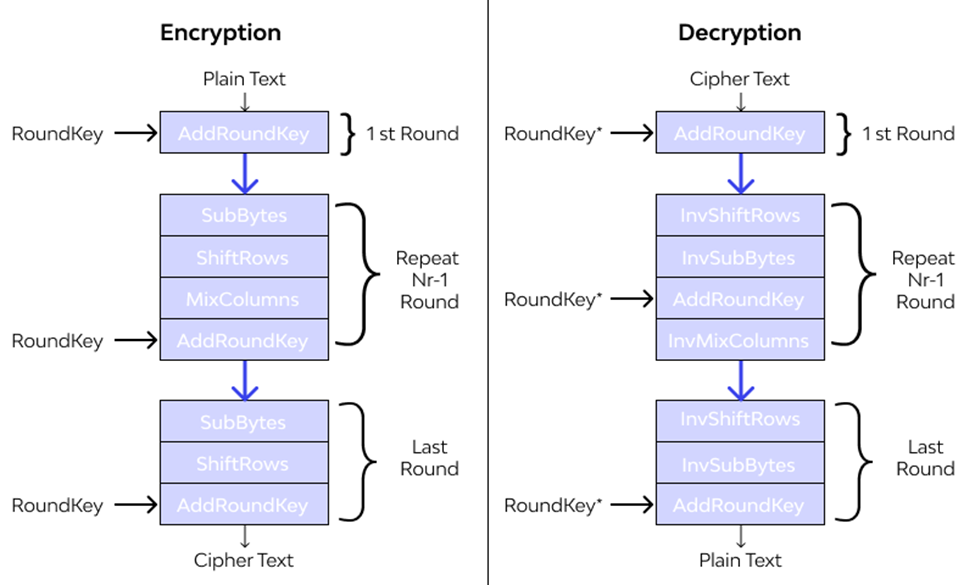
Từ cuối thập niên 1980, đầu thập niên 1990, xuất phát từ những lo ngại về độ an toàn và tốc độ thấp khi áp dụng bằng phần mềm, giới nghiên cứu đã đề xuất khá nhiều thuật toán mã hóa khối để thay thế DES. Những ví dụ tiêu biểu bao gồm: RC5, Blowwfish, IDEA (International Data Encryption Algorithm: Thuật toán mã hóa dữ liệu quốc tế), NewDES, SAFER và FEAL. Hầu hết những thuật toán này có thể sử dụng từ khóa 64 bit của DES mặc dù chúng thường được thiết kế hoạt động với từ khóa 64 bit hay 128 bit. Bản thân DES cũng cải tiến để có thể được sử dụng an toàn hơn.

Năm 2001, sau một cuộc thi quốc tế, NIST đã chọn ra một thuật toán mới là Tiêu chuẩn mã hóa tiên tiến AES (*Advanced* *Encryption* *Standard*) để thay thế cho DES. Thuật toán được trình diện dưới tên là *Rijndael*

*AES* là thuật toán mã hóa khối được chính phủ Hoa Kỳ áp dụng làm tiêu chuẩn mã hóa thay cho tiêu chuẩn DES trước đó. Giống như tiêu chuẩn DES, AES được kỳ vọng áp dụng trên phạm vi toàn thế giới và đã được nghiên cứu rất kỹ lưỡng. AES được chấp thuận làm tiêu chuẩn liên bang bởi Viện Tiêu chuẩn và Công nghệ Quốc gia Hoa Kỳ (NIST) trong một quá trình tiêu chuẩn hóa kéo dài 5 năm.

Thuật toán được thiết kế bởi hai nhà mật mã học người Bỉ: Joan Daemen và Vincent Rijmen. Thuật toán được đặt tên là "Rijndael" khi tham gia cuộc thi thiết kế AES theo cách ghép tên của hai đồng tác giả. Thuật toán được dffia trên bản thiết kế Square có trước đó của Daemen và Rijmen; còn Square lại được thiết kế dựa trên Shark. Khác với DES sử dụng mạng Feistel, Rijndael sử dụng mạng thay thế-chuyển vị. AES có thể dễ dàng thực hiện với tốc độ cao bằng phần mềm hoặc phần cứng và không đòi hỏi nhiều bộ nhớ. Do là một tiêu chuẩn mã hóa mới, AES đang được triển khai sử dụng rộng rãi hàng loạt.

2.2.1 Mô tả thuật toán



Thuật toán AES mã hóa được mô tả khái quát gồm 3 bước như sau:

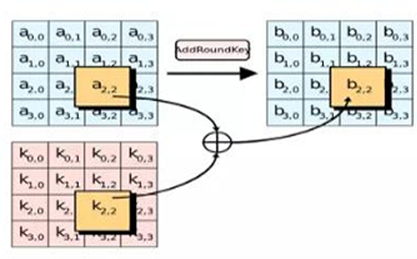
·1 Vòng khởi tạo chỉ gồm phép AddRoundKey

·Nr -1 Vòng lặp gồm 4 phép biển đổi lần lượt: SubBytes, ShiftRows, MixColumns, AddRoundKey.

·1 Vòng cuối gồm các phép biến đổi giống vòng lặp và không có phép MixColumns.

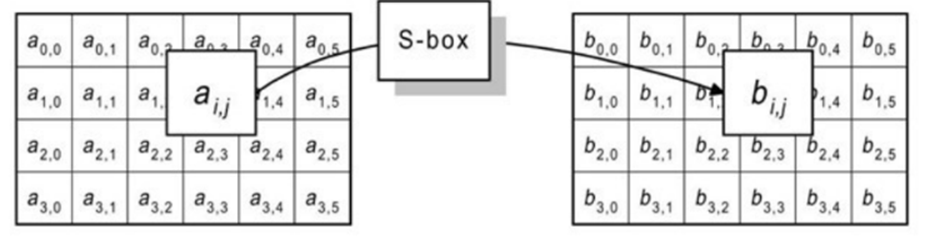
2.2.1.1 Phương thức AddRoundKey.

* Phép cộng với khóa là thực hiện phép XOR bit của 128bit của ma trận trạng thái và 128 bit của khóa tương ứng của vòng.

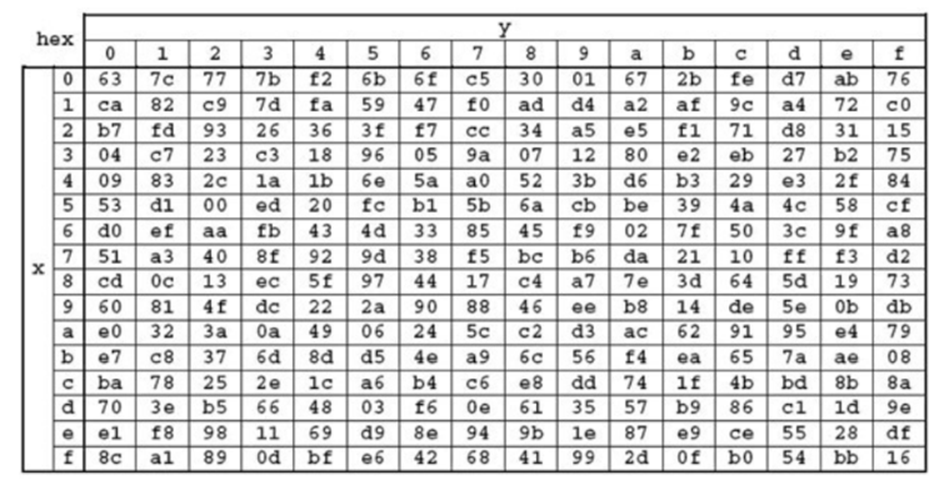


2.2.1.2 Phương thức SubBytes.

•Đây là một phép thay thế byte không tuyến tính. Phép biến đổi này thao tác trên mỗi byte của trạng thái một cách độc lập để tạo ra một giá trị byte mới bằng cách sử dụng một bảng thay thế S-box

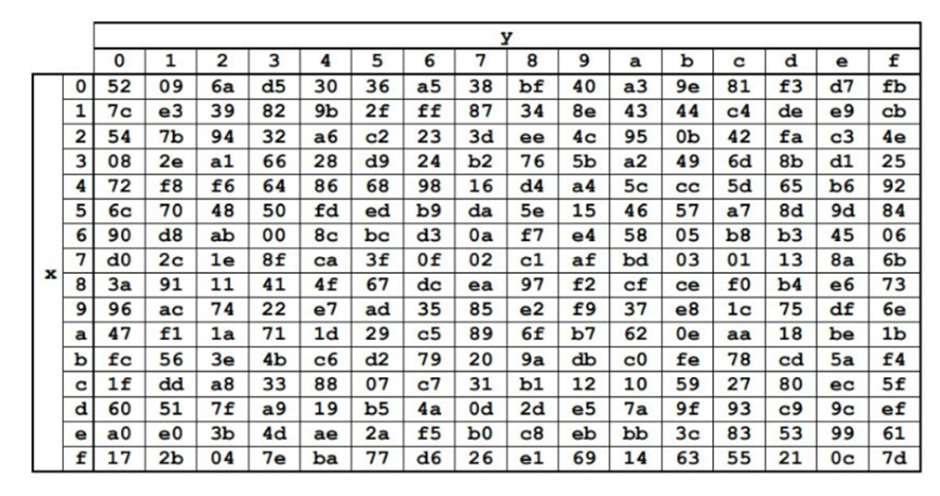


SubBytes thao tác trên mỗi byte trong trạng thái một cách độc lập



Bảng S-Box

Phép thay thế này có thể đảo ngược bằng cách sử dụng bảng Inverse Sbox, sử dụng hệt như bảng Sbox thường.



Bảng Inverse Sbox

2.2.1.3 Phương thức ShiftRows

•Trong biến đổi ShiftRows(), các byte trong ba hàng cuối cùng của trạng thái được dịch vòng đi các số byte khác nhau (độ lệch).

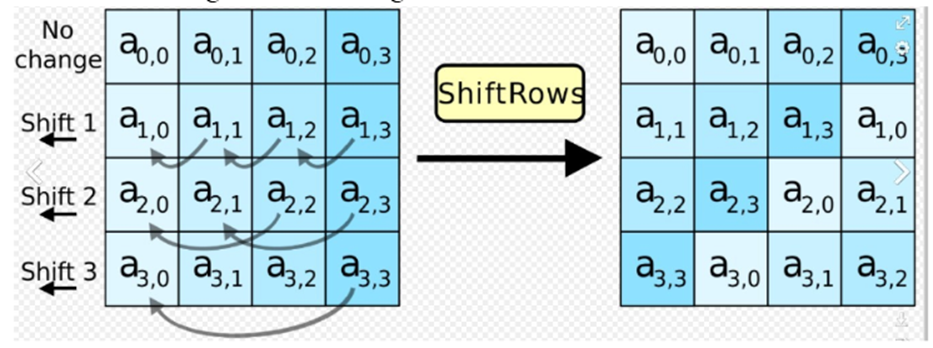
•Hàng đầu tiên không bị dịch, ba hàng còn lại bị dịch tương ứng:

Hàng thứ 1 giữ nguyên.

Hàng thứ 2 dịch vòng trái 1 lần.

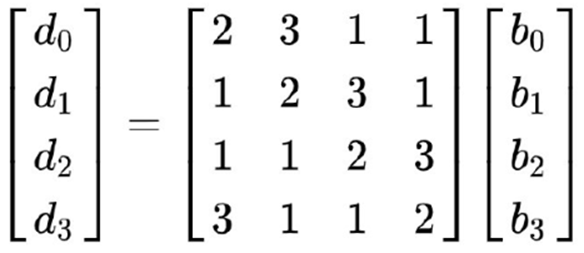
Hàng thứ 3 dịch vòng trái 2 lần.

Hàng thứ 4 dịch vòng trái 3 lần.



2.2.1.4 Phương thức MixColumns.

•Phép biến đổi này thao tác một cách độc lập trên mỗi cột của trạng thái và xem mỗi cột như một đa thức bậc 4.



Thuật toán AES giải mã được mô tả khái quát gồm 3 bước như sau:

* Để giải mã một bản mã được mã hóa AES, cần phải hoàn tác từng giai đoạn của hoạt động mã hóa theo thứ tự ngược lại mà chúng đã được áp dụng. Ba giai đoạn giải mã như sau:

Đảo ngược vòng cuối: AddRoundKey – InvShiftRows – InvSubBytes

Đảo ngược các vòng lặp: AddRoundKey – InvMixColumns – InvShiftRows –InvSubBytes

Đảo ngược vòng tiên quyết: AddRoundKey

* Trong số bốn hoạt động trong mã hóa AES, chỉ có phương thức AddRoundKey là nghịch đảo của chính nó (vì nó là exclusive-or). Để hoàn tác AddRoundKey, chỉ cần mở rộng toàn bộ lịch khóa AES (giống như mã hóa) và sau đó sử dụng khóa thích hợp theo chiều ngược với giai đoạn mã hóa.
* Hàm đảo ngược của SubBytes là invSubBytes, giống hệt như SubBytes, ngoại trừ việc sử dụng Sbox, ta sẽ sử dụng inverse Sbox

2.2.1.5 Phương thức invShiftRows

* Trong biến đổi invShiftRows(), các byte trong ba hàng cuối cùng của trạng thái được dịch vòng đi các số byte khác nhau (độ lệch).
* Hàng đầu tiên không bị dịch, ba hàng còn lại bị dịch tương ứng:

Hàng thứ 1 giữ nguyên.

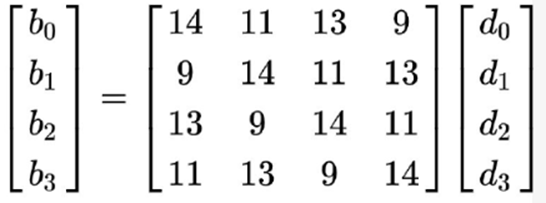
Hàng thứ 2 dịch vòng phải 1 lần.

Hàng thứ 3 dịch vòng phải 2 lần.

Hàng thứ 4 dịch vòng phải 3 lần

2.2.1.6 Phương thức InvMixColumns.

* Phép biến đổi này thao tác một cách độc lập trên mỗi cột của trạng thái và xem mỗi cột như một đa thức bậc 4.



2.3.Độ an toàn của AES

Vào thời điểm năm 2006, dạng tấn công lên AES duy nhất thành công là tấn công kênh bên (*side* *channel* *attack*). Vào tháng 6 năm 2003, Chính phủ Hoa Kỳ tuyên bố AES có thể được sử dụng cho thông tin mật.

*"Thiết* *kế và* *độ dài* *khóa* *của* *thuật* *toán* *AES* *(128,* *192* *và* *256* *bit)* *là* *đủ an* *toàn* *để bảo* *vệ các* *thông* *tin* *được* *xếp* *vào* *loại* *TỐI* *MĺT* *(secret).* *Các* *thông* *tin* *TUYŔT* *MĺT* *(top* *secret)* *sẽ phải* *dùng* *khóa* *192* *hoặc* *256* *bit.* *Các* *phiên* *bản* *thực* *hiện* *AES* *nhằm* *mục* *đích* *bảo* *vệ hệ thống* *an* *ninh* *hay* *thông* *tin* *quốc* *gia* *phải* *được* *NSA* *kiểm* *tra* *và* *chứng* *nhận* *trước* *khi* *sử dụng."*

Điều này đánh dấu lần đầu tiên công chúng có quyền tiếp xúc với thuật toán mật mã mà NSA phê chuẩn cho thông tin TUYNT MĺT.

Nhiều phần mềm thương mại hiện nay sử dụng mặc định khóa có độ dài 128 bit.

Phương pháp thường dùng nhất để tấn công các dạng mã hóa khối là thử các kiểu tấn công lên phiên bản có số chu trình thu gọn. Đối với khóa 128 bit, 192 bit và 256 bit, AES có tương ứng 10, 12 và 14 chu trình. Tại thời điểm năm 2006, những tấn công thành công được biết đến là 7 chu trình đối với khóa 128 bit, 8 chu trình với khóa 192 bit và 9 chu trình với khóa 256 bit.

Một số nhà khoa học trong lĩnh vffic mật mã lo ngại về an ninh của AES. Họ cho rằng ranh giới giữa số chu trình của thuật toán và số chu trình bị phá vỡ quá nhỏ. Nếu các kỹ thuật tấn công được cải thiện thì AES có thể bị phá vỡ. Ő đây, *phá* *vỡ* có nghĩa chỉ bất cứ phương pháp tấn công nào nhanh hơn tấn công kiểu duyệt toàn bộ (tấn công bạo lffic).

Vì thế một tấn công cần thffic hiện 2120 plaintexts cũng được coi là thành công mặc dù tấn công này chưa thể thffic hiện trong thffic tế. Tại thời điểm hiện nay, nguy cơ này không thffic sffi nguy hiểm và có thể bỏ qua.

Tấn công kiểu duyệt toàn bộ quy mô nhất đã từng thffic hiện là do *distributed.net* thffic hiện lên hệ thống 64 bit RC5 vào năm 2002 (Theo định luật Moore thì nó tương đương với việc tấn công vào hệ thống 66 bit hiện nay).

Một vấn đề khác nữa là cấu trúc toán học của AES. Không giống với các thuật toán mã hóa khác, AES có mô tả toán học khá đơn giản. Tuy điều này chưa dẫn đến mối nguy hiểm nào nhưng một số nhà nghiên cứu sợ rằng sẽ có người lợi dụng được cấu trúc này trong tương lai.

Vào năm 2002, Nicolas Courtois và Josef Pieprzyk phát hiện một tấn công trên lý thuyết gọi là tấn công XSL và chỉ ra điểm yếu tiềm tàng của AES. Tuy nhiên, một vài chuyên gia về mật mã học khác cũng chỉ ra một số vấn đề chưa rõ ràng trong cơ sở toán học của tấn công này và cho rằng các tác giả đã có thể có sai lầm trong tính toán.

Việc tấn công dạng này có thffic sffi trở thành hiện thffic hay không vẫn còn để ngỏ và cho tới nay thì tấn công XSL vẫn chỉ là suy đoán

2.3.Tấn công kênh bên (Side channel attacks)

Tấn công kênh bên không tấn công trffic tiếp vào thuật toán mã hóa mà thay vào đó, tấn công lên các hệ thống thffic hiện thuật toán có sơ hở làm lộ dữ liệu.

Tháng 4 năm 2005, Daniel J. Bernstein công bố một tấn công lên hệ thống mã hóa AES trong OpenSSL. Một máy chủ được thiết kế để đưa ra tối đa thông tin về thời gian có thể thu được và cuộc tấn công cần tới 200 triệu plaintexts lffia chọn. Một số người cho rằng tấn công không thể thffic hiện được trên Internet với khoảng cách vài điểm mạng.

Tháng 10 năm 2005, Adi Shamir và 2 nhà nghiên cứu khác có một bài nghiên cứu minh họa một vài dạng khác. Trong đó, một tấn công có thể lấy được khóa AES với 800 lần ghi trong 65 mili giây.

Tấn công này yêu cầu kẻ tấn công có khả năng chạy chương trình trên chính hệ thống thffic hiện mã hóa.

* 1. Thiết kế chương trình, cài đặt thuật toán.
* Thiết kế kịch bản chương trình
* Giới thiệu ngôn ngữ lập trình sử dụng để cài đặt thuật toán.
* Cài đặt thuật toán, giao diện chương trình (theo từng ngôn ngữ)

**Chương 3: Kết luận và bài học kinh nghiệm**

* 1. Kiến thức kỹ năng đã học được trong quá trình thực hiện đề tài.
  2. Bài học kinh nghiệm
  3. Đề xuất về tính khả thi của chủ đề nghiên cứu, những thuận lợi, khó khăn …

**Tài liệu tham khảo**

(Sắp xếp theo ngôn ngữ: Việt, Anh, Pháp, Đức, Nga…)

[1]. Nguyễn Xuân Dũng, *Bảo mật thông tin – Mô Hình và ứng dụng*, NXB thống kê, 2009.

[2]. Bùi Doãn Khanh, Nguyễn Đình Thúc, *Mã hóa thông tin – Lý thuyết và ứng dụng*, NXB Lao động xã hội, 2011.

[3]. William Stallings**,** *Cryptography and Network Security Principles and Practices***,** Fourth Edition, Prentice Hall, 2005.

…

[n]. <http://www.hhs.com>

(thời gian truy cập: 15h00, ngày 27/10/2024)

**Phụ lục 1: Chương trình Java**

**Phụ lục 2: Chương trình C#**

**Quy định trình bày báo cáo:**

Lề trên: 2,5 cm Lề dưới: 2 cm

Lề trái: 3,5 cm lề phải: 2 cm

Nội dung báo cáo trình bày: căn đều lề 2 bên

Phông chữ: Time New Roman

Cỡ chữ: 13 (hoặc 14)

Dãn dòng: Multiple 1.3

Các đoạn báo cáo căn đều 2 bên

Đánh số trang ở giữa, phía trên đầu mỗi trang giấy (phần Header)

Số lượng trang báo cáo: từ 40 - 60 trang