**Nhập môn ATTT**

**Câu 1:** Phân tích ý nghĩa của Luật Kirchoff để thấy tại sao mật mã hiện đại không chấp nhận quan điểm cần che giấu thuật toán mật mã.

* Có 3 lí do chính:
  + Trước hết, nó cho phép cộng đồng mật mã học kiểm tra và đánh giá tính bảo mật của thuật toán. Điều này đảm bảo rằng các chuyên gia có thể phát hiện và khắc phục các lỗ hổng bảo mật.
  + Thứ hai, bằng cách đặt sự bảo mật trên khóa, không phải thuật toán, ta tạo ra một hệ thống mật mã mạnh mà thuật toán có thể công khai mà không ảnh hưởng đến tính bảo mật.
  + Cuối cùng, việc công khai thuật toán khuyến khích sự đóng góp và phát triển từ cộng đồng mật mã học, giúp tìm ra các giải pháp bảo mật tiên tiến hơn.

**Câu 2:** Phân tích một số ưu điểm/nhược điểm trong nguyên lý của hệ mật mã khóa đối xứng (SKC) và mật mã khóa công khai (PKC).

* ***Ưu điểm của SKC:***

Hiệu suất cao: SKC thường nhanh hơn PKC trong việc thực hiện mã hóa và giải mã do sử dụng cùng một khóa cho cả hai quá trình.

Bảo mật tốt: Với việc sử dụng một khóa chia sẻ giữa người gửi và người nhận, SKC cung cấp mức độ bảo mật cao và kháng lại nhiều loại tấn công, như tấn công từ điển.

Xử lý dữ liệu lớn: SKC thích hợp cho việc xử lý dữ liệu lớn, như mã hóa và giải mã dữ liệu đại trà.

* ***Nhược điểm của SKC:***

Quản lý khóa: SKC yêu cầu một cơ chế đáng tin cậy để chia sẻ và quản lý các khóa bí mật giữa các bên liên quan. Điều này có thể gây ra khó khăn trong việc quản lý và phân phối khóa khi có nhiều bên tham gia.

* ***Ưu điểm của PKC:***

Quản lý khóa dễ dàng: PKC không yêu cầu việc chia sẻ khóa bí mật mà chỉ yêu cầu cặp khóa công khai và khóa bí mật riêng biệt. Điều này giúp việc quản lý khóa dễ dàng hơn trong môi trường có nhiều bên tham gia.

Xác thực và chữ ký điện tử: PKC cho phép xác thực người gửi và tính toàn vẹn của dữ liệu thông qua chữ ký điện tử. Điều này rất hữu ích trong các tình huống như giao dịch điện tử hay xác thực danh tính.

Truyền tin an toàn: PKC cung cấp khả năng truyền tin an toàn trên kênh bất an và mở rộng việc mã hóa cho nhiều bên tham gia một cách hiệu quả.Bí mật khóa: SKC đặt sự bảo mật chủ yếu trên khóa. Nếu khóa bị lộ hoặc bị tấn công, hệ thống mật mã có thể trở nên yếu đối với các cuộc tấn công brute-force hoặc tấn công từ điển.

* ***Nhược điểm của PKC:***

Hiệu suất thấp: PKC thường chậm hơn SKC trong việc thực hiện mã hóa và giải mã vì sử dụng phép tính toán phức tạp hơn, như phép tính lũy thừa.

Chi phí tính toán cao: PKC yêu cầu sự tính toán phức tạp và sử dụng các phép tính toán như phép lũy thừa modular và tính toán trên các đường cong elliptic, dẫn đến một chi phí tính toán cao hơn so với SKC.

**Câu 3:** Tính hỗn loạn (confusion) và tính khuếch tán (diffusion) là gì? Nguyên lý để tạo ra chúng.

* ***Tính hỗn loạn (Confusion):***

Tính hỗn loạn đề cập đến việc gây ra sự khó hiểu và không thể dự đoán được giữa khóa và dữ liệu đã được mã hóa. Nó đảm bảo rằng bất kỳ thay đổi nhỏ nào trong khóa hoặc dữ liệu đầu vào sẽ tạo ra sự biến đổi lớn và khó đoán trong đầu ra mã hóa. Tính hỗn loạn nhằm tránh việc tấn công dựa trên các mối quan hệ thống kê giữa khóa và dữ liệu.

Nguyên lý để tạo ra tính hỗn loạn là sử dụng các phép biến đổi phi tuyến (non-linear transformations) như hàm thay thế (substitution) hoặc hàm hộp (box function). Các phép biến đổi này sẽ làm mất đi sự tương quan đơn giản giữa khóa và đầu vào, tạo ra sự rối loạn và khó hiểu trong quá trình mã hóa.

* ***Tính khuếch tán (Diffusion):***

Tính khuếch tán đề cập đến việc phân tán thông tin của dữ liệu đầu vào ra khắp các phần tử của mã hóa đầu ra. Nó đảm bảo rằng mỗi bit hoặc phần tử dữ liệu đầu vào đều ảnh hưởng đến nhiều bit hoặc phần tử đầu ra. Tính khuếch tán nhằm ngăn chặn việc xác định thông tin ban đầu từ các thuộc tính thống kê của mã hóa.

Nguyên lý để tạo ra tính khuếch tán là sử dụng các phép biến đổi tuyến tính (linear transformations) như phép xor, phép dịch bit (bit shifting), hoặc phép hoán vị (permutation). Các phép biến đổi này sẽ đảm bảo rằng mỗi bit hoặc phần tử dữ liệu đầu vào sẽ ảnh hưởng đến nhiều bit hoặc phần tử đầu ra, làm lan truyền thông tin một cách đều và phân tán.

**Câu 4:** Hãy vẽ sơ đồ mã hóa/giải mã cho các chế độ mật mã khối ECB, CBC, CFB, CTR.

* ECB là chế độ mã hóa từng khối bit độc lập. Với cùng một khóa mã K, mỗi khối plaintext ứng với một giá trị ciphertext cố định và ngược lại.

A picture containing text, screenshot, font

Description automatically generated

* CBC là chế độ mã hóa chuỗi, kết quả mã hóa của khối dữ liệu trước (ciphertext) sẽ được tổ hợp với khối dữ liệu kế tiếp (plaintext) trước khi thực thi mã hóa.

A picture containing text, diagram, font, screenshot

Description automatically generated

* CFB là chế độ mã hóa mà ciphertext của lần mã hóa hiện tại sẽ được phản hồi (feedback) đến đầu vào của lần mã hóa tiếp theo. Nghĩa là, ciphertext của lần mã hóa hiện tại sẽ được sử dụng để tính toán ciphertext của lần mã hóa kế tiếp. Mô tả có vẻ giống CBC nhưng quá trình trực hiện lại khác.

A diagram of a block diagram

Description automatically generated with low confidence

* OFB là chế độ mã hóa mà giá trị ngõ ra của khối thực thi thuật toán mã hóa, **không phải ciphertext**, của lần mã hóa hiện tại sẽ được phản hồi (feedback) đến ngõ vào của lần mã hóa kế tiếp. A picture containing text, diagram, plan, technical drawing

  Description automatically generated

**Câu 5:** Hãy nêu nguyên lý xây dựng hệ mật mã khóa đối xứng PKC

* Hệ mật mã khóa công khai (PKC) được xây dựng dựa trên nguyên lý sử dụng cặp khóa công khai và khóa bí mật. Nguyên lý xây dựng PKC chủ yếu dựa trên hai phép tính toán toán học khó và ngược lại nhau: phép mã hóa và phép giải mã.
  + ***Phép mã hóa:***

Người gửi sử dụng khóa công khai của người nhận để mã hóa thông điệp.

Phép mã hóa sử dụng một thuật toán mã hóa công khai, chẳng hạn như RSA hoặc ECC, để thực hiện quá trình mã hóa.

Kết quả của phép mã hóa là một thông điệp được mã hóa hoặc mã hóa đối tượng.

* + ***Phép giải mã:***

Người nhận sử dụng khóa bí mật tương ứng với khóa công khai đã được sử dụng để mã hóa để giải mã thông điệp.

Phép giải mã sử dụng một thuật toán giải mã, tương ứng với thuật toán mã hóa công khai đã được sử dụng, để thực hiện quá trình giải mã.

Kết quả của phép giải mã là thông điệp ban đầu.

* Nguyên lý chính của PKC là tính không thể đảo ngược của các phép tính toán toán học phức tạp như phân tích nguyên tố, lũy thừa modul, hoặc tính toán trên các đường cong elliptic. Khóa công khai được công khai cho mọi người, trong khi khóa bí mật chỉ được giữ bí mật bởi người sở hữu. Điều này cho phép người gửi mã hóa thông điệp bằng khóa công khai của người nhận mà không cần trao đổi trước một khóa chia sẻ.

**Câu 6:** Tìm hiểu thuật toán GCD mở rộng tìm nghịch đảo theo mô-đun đồng dư.

* Thuật toán GCD mở rộng (Extended Euclidean Algorithm) được sử dụng để tìm nghịch đảo của một số theo mô-đun đồng dư. Nghịch đảo modulo là một số nguyên mà khi nhân với số cần tìm nghịch đảo và lấy modulo theo một số cho trước, ta được kết quả là 1.
* Thuật toán GCD mở rộng dựa trên thuật toán Euclid để tìm ước chung lớn nhất (Greatest Common Divisor - GCD) giữa hai số. Ngoài việc tìm GCD, nó cũng tính toán các hệ số mở rộng của hai số ban đầu.
* Dưới đây là mô tả thuật toán GCD mở rộng để tìm nghịch đảo theo mô-đun đồng dư:
  + Đầu vào: Hai số nguyên a và b, và một số nguyên dương m.
  + Bước 1: Gán các biến:

x₀ = 1, y₀ = 0, x₁ = 0, y₁ = 1.

a₀ = a, b₀ = b.

* Bước 2: Lặp cho đến khi b₀ khác 0:

Tính phần nguyên k và phần dư r khi chia a₀ cho b₀: a₀ = b₀ \* k + r.

Cập nhật các biến: a₀ = b₀, b₀ = r.

Tính các hệ số mới:

x\_new = x₀ - k \* x₁, y\_new = y₀ - k \* y₁.

Cập nhật các biến: x₀ = x₁, x₁ = x\_new, y₀ = y₁, y₁ = y\_new.

* Nếu b₀ = 1, nghĩa là a và m là cùng số nguyên tố, nghịch đảo của a theo m modulo là x₁ (nếu x₁ âm, ta cộng một lượng m để đảm bảo nằm trong khoảng 0 đến m-1).
* Nếu b₀ ≠ 1, không tồn tại nghịch đảo của a theo m modulo.
* Cuối cùng, nghịch đảo của a theo m modulo là x₁.

**Câu 7:** Tìm hiểu giải thuật nhân nhanh lũy thừa theo mô-đun đồng dư

* Giải thuật nhân nhanh lũy thừa theo mô-đun đồng dư, còn được gọi là "Square and Multiply" (Bình phương và Nhân), là một phương pháp hiệu quả để tính lũy thừa modulo của một số nguyên. Phương pháp này giúp giảm số lượng phép nhân cần thực hiện trong quá trình tính toán.
* Dưới đây là cách thực hiện giải thuật nhân nhanh lũy thừa theo mô-đun đồng dư:
* Dưới đây là cách thực hiện giải thuật nhân nhanh lũy thừa theo mô-đun đồng dư:
* Chuyển đổi số mũ sang dạng nhị phân: Để tính a^b mod n, ta chuyển đổi số mũ b sang dạng nhị phân.
* Khởi tạo kết quả: Đặt kết quả ban đầu bằng 1: result = 1.
* Lặp qua các bit của số mũ: Bắt đầu từ bit thấp nhất đến bit cao nhất của số mũ (từ phải sang trái).

a. Bước 3.1: Nhân kết quả hiện tại với chính nó (result = result \* result) và lấy phần dư modulo n.

b. Bước 3.2: Nếu bit tương ứng của số mũ bằng 1, nhân kết quả hiện tại với cơ số a (result = result \* a) và lấy phần dư modulo n.

Kết quả cuối cùng là giá trị result.

**Câu 8:** TÌm hiểu thuật toán RSA và xây dựng căp khóa bất đối xứng (KU, KR)

* Thuật toán RSA (Rivest-Shamir-Adleman) là một thuật toán mật mã khóa công khai phổ biến được sử dụng trong nhiều ứng dụng bảo mật. Nó dựa trên việc tính toán phân tích số nguyên tố và lũy thừa modulo.
* Dưới đây là mô tả của thuật toán RSA và cách xây dựng cặp khóa bất đối xứng (KU, KR):
  + Xây dựng cặp khóa:

a. Bước 1: Chọn hai số nguyên tố lớn ngẫu nhiên p và q.

b. Bước 2: Tính n = p \* q, n là modulo.

c. Bước 3: Tính φ(n) = (p-1) \* (q-1), trong đó φ(n) là hàm số Euler.

d. Bước 4: Chọn một số nguyên e thỏa mãn 1 < e < φ(n) và là số nguyên tố cùng nhau với φ(n).

e. Bước 5: Tìm số nguyên d sao cho (d \* e) % φ(n) = 1. Số d này là nghịch đảo của e modulo φ(n).

f. Cặp khóa bất đối xứng: KU = (e, n) là khóa công khai, KR = (d, n) là khóa bí mật.

* Mã hóa:

a. Cho một thông điệp m, mã hóa nó thành một số nguyên c sử dụng khóa công khai KU theo công thức: c = m^e mod n.

* Giải mã:

a. Cho một số nguyên c được mã hóa, giải mã nó thành một thông điệp m sử dụng khóa bí mật KR theo công thức: m = c^d mod n.

* Trong thuật toán RSA, khóa công khai KU được chia sẻ với mọi người trong khi khóa bí mật KR chỉ được giữ bí mật bởi người sở hữu. Việc mã hóa được thực hiện bằng cách sử dụng khóa công khai, trong khi giải mã chỉ có thể được thực hiện bằng khóa bí mật tương ứng.
* Đặc điểm quan trọng của RSA là tính khó phá vỡ nếu không có thông tin về khóa bí mật. Để đảm bảo tính bảo mật, số nguyên p và q phải được chọn ngẫu nhiên và đủ lớn, và việc tính toán dựa trên phân tích nguyên tố là khó khăn và tốn nhiều thời gian.
* Tuy nhiên, để đạt được hiệu suất tốt, trong thực tế, thường sử dụng phiên bản mở rộng của RSA như RSA-OAEP cho mã hóa và RSA-PSS cho chữ ký điện tử, cùng với các biện pháp bổ sung để bảo vệ khỏi các tấn công như tấn công kiểu chọn (chosen-ciphertext attack) và tấn công kiểu căn chỉnh (side-channel attack).

**Câu 9:** Trong pha thiết lập tham số thuật toán RSA, tại sao ta phải chon 2 số nguyên tố p và q có độ lớn xấp xỉ nhau (cùng độ dài)?

* ***Khó khăn trong việc phân tích nguyên tố:*** Độ lớn của số nguyên tố ảnh hưởng đến khả năng phân tích nguyên tố của nó. Khi p và q có độ lớn xấp xỉ nhau, việc phân tích nguyên tố để tìm ra hai số nguyên tố gốc trở nên khó khăn hơn. Nếu p và q có độ lớn khác biệt lớn, một số thuật toán phân tích nguyên tố có thể tận dụng sự khác biệt này để giảm đáng kể thời gian tính toán.
* ***Chống lại tấn công phân tích giai đoạn 1 (Phase 1 Factoring Attack):*** Trong tấn công này, kẻ tấn công cố gắng phân tích nguyên tố nhanh chóng bằng cách tìm kiếm số nguyên tố nhỏ hơn sqrt(n), nơi n là tích của hai số nguyên tố p và q. Nếu p và q có độ lớn xấp xỉ nhau, khả năng tìm kiếm những số nguyên tố nhỏ hơn sqrt(n) sẽ khó hơn và tăng thời gian cần thiết để thực hiện tấn công.
* ***Tăng tính đồng đều của khóa:*** Khi p và q có độ lớn xấp xỉ nhau, nằm trong khoảng cùng một độ dài bit, các phép tính và thao tác trên hai số nguyên tố này sẽ có cùng độ phức tạp. Điều này giúp tạo ra các khóa công khai và bí mật có cùng kích thước, đồng đều và thuận tiện cho việc triển khai và tính toán.

**Câu 10:** Cho p = 11, q = 17 trong hệ RSA. Chọn một số ngẫu nhiên M trong khoảng từ 5-20. Thực hiện các công việc sau:

A picture containing text, diagram, sketch, plan

Description automatically generated

1. Xây dựng cặp khóa bất đối xứng (KU, KR) dựa trên thuật toán GCD mở rộng.
2. Tính bản mã của bản tin M (lấy khóa Ku để mã hóa)
3. Nếu sử dụng hệ này làm chữ ký, xác định chữ ký cho M nói trên. (lấy Kr để làm chữ kí)
4. Nếu muốn gửi bản tin M vừa có tính xác thực, vừa có tính mật cần thực hiện như thế nào (Vẽ sơ đồ minh họa)

A screenshot of a computer

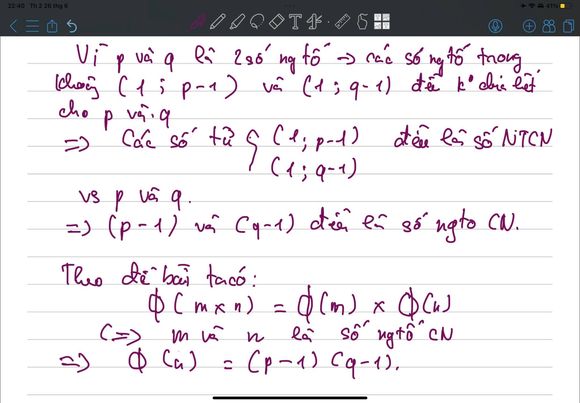
Description automatically generated

* Quy trình trên bao gồm các bước sau:
  + Mã **hóa** (Encryption): Sử dụng khóa mã hóa để mã hóa bản tin M thành một bản tin đã được mã hóa (Encrypted Message). Quá trình mã hóa này sử dụng thuật toán mã hóa đối xứng hoặc mã hóa không đối xứng.
  + **Chữ ký điện tử** (Digital Signature): Sử dụng khóa ký để tạo chữ ký điện tử cho bản tin đã mã hóa (Encrypted Message). Quá trình này sử dụng hàm băm và thuật toán ký điện tử để tạo ra chữ ký điện tử.
  + Kết hợp mã hóa và chữ ký: Kết hợp bản tin đã mã hóa và chữ ký điện tử thành một bản tin ký và mã hóa (Signed and Encrypted Message).
* Khi nhận được bản tin ký và mã hóa, người nhận thực hiện các bước ngược lại:
  + Xác thực chữ ký điện tử: Sử dụng khóa công khai tương ứng với khóa ký để xác thực tính hợp lệ của chữ ký điện tử. Nếu chữ ký điện tử hợp lệ, người nhận có thể tin tưởng tính xác thực của bản tin.
  + Giải mã (Decryption): Sử dụng khóa giải mã để giải mã bản tin đã được mã hóa. Quá trình này đảo ngược quá trình mã hóa và khôi phục bản tin gốc.
  + Kết quả là người nhận nhận được bản tin M ban đầu với tính xác thực và tính mật.

**Câu 11:** Biết rằng hàm (n) có nhân tính, có nghĩa là (m\*n) = (m) \* (n) với mọi m và n nguyên mà gcd(m,n)=1. Hãy chứng minh rằng (n)= (p-1)\* (q-1) khi n= p\*q với p, q là số nguyên tố

A screenshot of a computer

Description automatically generated with low confidence



**Câu 12:** Chứng tỏ rằng thuật toán RSA vẫn đúng, nghĩa là (Xe)d=X (mod n) ngay cả khi gcd(X,n)≠1

A screenshot of a computer

Description automatically generated with low confidence

A screenshot of a computer

Description automatically generated with low confidence

A screenshot of a computer

Description automatically generated with low confidence

**Câu 13:** Hàm băm là gì? Hãy nêu các yêu cầu cần thiết khi xây 0.dựng một hàm băm

* Hàm băm (hash function) là một phép ánh xạ một cách ngẫu nhiên từ một tập hợp lớn các đối tượng (như dữ liệu, chuỗi ký tự, hoặc tệp tin) thành một giá trị đầu ra có độ dài cố định. Mục đích chính của hàm băm là tạo ra một mã băm duy nhất (hash code) để đại diện cho dữ liệu đầu vào.
* Khi xây dựng một hàm băm, có một số yêu cầu cần thiết để đảm bảo tính chất và hiệu suất của hàm băm, bao gồm:
* Độ tương đồng: Hàm băm phải đảm bảo tính chất độ tương đồng (độ tin cậy cao) trong việc ánh xạ dữ liệu đầu vào. Điều này có nghĩa là hai dữ liệu khác nhau phải có khả năng tạo ra hai mã băm khác nhau hoặc có xác suất rất thấp để tạo ra mã băm giống nhau.
* Độc lập: Mã băm phải độc lập với dữ liệu đầu vào. Điều này có nghĩa là thay đổi nhỏ trong dữ liệu đầu vào sẽ dẫn đến sự thay đổi đáng kể trong giá trị mã băm.
* Tính đồng nhất: Hàm băm phải có khả năng tạo ra mã băm có độ dài cố định, bất kể kích thước của dữ liệu đầu vào. Điều này rất hữu ích trong việc lưu trữ và tìm kiếm dữ liệu.
* Hiệu suất: Hàm băm cần có hiệu suất tốt, tức là việc tính toán mã băm phải nhanh chóng và hiệu quả, ngay cả với dữ liệu lớn.
* Chiếu ngược khó: Hàm băm phải có tính chất chiếu ngược khó, tức là từ mã băm, không thể suy luận ngược lại được dữ liệu gốc.
* Phân phối đều: Mã băm phải phân phối đều trên không gian mã băm để tránh sự xung đột (collision) - tình huống mà hai dữ liệu khác nhau tạo ra cùng một mã băm.
* Các yêu cầu trên giúp đảm bảo tính chất an toàn và đáng tin cậy của hàm băm trong các ứng dụng như lưu trữ dữ liệu, tìm kiếm, mã hóa, xác thực và nhiều lĩnh vực khác.

**Câu 14:** Hãy chỉ ra ít nhất 3 ứng dụng của hàm băm trong thực tế.

* Mã hóa mật khẩu: Hàm băm được sử dụng rộng rãi trong việc mã hóa mật khẩu. Thay vì lưu trữ mật khẩu người dùng dưới dạng văn bản thông thường, các hệ thống thông thường sẽ lưu trữ một giá trị băm (hash value) của mật khẩu. Khi người dùng đăng nhập, hệ thống sẽ tính toán giá trị băm của mật khẩu được nhập vào và so sánh với giá trị băm đã lưu trữ. Điều này giúp bảo mật mật khẩu, vì ngay cả khi cơ sở dữ liệu bị xâm nhập, kẻ tấn công cũng không thể đọc được mật khẩu gốc từ giá trị băm.
* Kiểm tra tính toàn vẹn dữ liệu: Hàm băm được sử dụng để kiểm tra tính toàn vẹn dữ liệu trong các ứng dụng như gửi và nhận file, truyền tải dữ liệu qua mạng, hoặc lưu trữ dữ liệu trong cơ sở dữ liệu. Bằng cách tính toán giá trị băm của dữ liệu và so sánh với giá trị băm ban đầu, chúng ta có thể kiểm tra xem dữ liệu đã bị thay đổi hay không. Nếu giá trị băm khớp, tức là dữ liệu không bị thay đổi. Ngược lại, nếu giá trị băm không khớp, có khả năng dữ liệu đã bị sửa đổi hoặc bị hỏng.
* Lưu trữ tài liệu và tìm kiếm: Hàm băm được sử dụng trong các hệ thống lưu trữ và tìm kiếm để tăng tốc độ tìm kiếm và truy xuất dữ liệu. Thay vì tìm kiếm thông qua toàn bộ nội dung của tài liệu, một hàm băm được áp dụng cho nội dung tài liệu và tạo ra một giá trị băm duy nhất cho nó. Sau đó, chỉ cần tìm kiếm dựa trên giá trị băm thay vì so sánh nội dung tài liệu. Điều này giúp giảm thiểu thời gian và công sức cần thiết để tìm kiếm và truy xuất dữ liệu trong các hệ thống lưu trữ lớn.

**Câu 15:** Phân tích sự khác biệt giữa chữ ký truyền thống và chữ ký điện tử.

* Phương tiện truyền tải: Chữ ký truyền thống thường được thực hiện bằng việc ký tên hoặc ký hình chữ ký trên một tài liệu giấy. Chữ ký này là duy nhất cho mỗi cá nhân và có thể được kiểm tra bằng cách so sánh với chữ ký gốc. Trong khi đó, chữ ký điện tử được tạo ra và xác thực bằng các phương pháp mã hóa và công nghệ số hóa để tạo ra một chữ ký điện tử duy nhất cho mỗi tài liệu hoặc thông điệp. Chữ ký điện tử thường được gắn kết với tài liệu hoặc thông điệp để đảm bảo tính toàn vẹn và xác thực.
* Bảo mật: Chữ ký truyền thống có thể bị sao chép hoặc làm giả một cách dễ dàng. Bạn có thể chụp hình chữ ký truyền thống hoặc sao chép nó vào tài liệu khác mà không gặp khó khăn nhiều. Trong khi đó, chữ ký điện tử sử dụng các thuật toán mã hóa mạnh mẽ và công nghệ chống làm giả để tạo ra một chữ ký điện tử duy nhất và rất khó để sao chép hoặc làm giả. Do đó, tính bảo mật của chữ ký điện tử cao hơn chữ ký truyền thống.
* Xác thực: Chữ ký truyền thống thường yêu cầu việc xác thực trực tiếp từ người ký hoặc từ một bên thứ ba đáng tin cậy như một nhân viên của ngân hàng hoặc một nhà chứng thực. Trong khi đó, chữ ký điện tử có thể được xác thực tự động bằng cách sử dụng công nghệ mật mã và hệ thống quản lý chữ ký điện tử. Điều này giúp rút ngắn thời gian và công sức cần thiết để xác thực chữ ký và đảm bảo tính xác thực.

**Câu 16:** Tác sao nói chữ ký điện tử có hai công dụng: vừa xác thực văn bản vừa xác thực danh tính ngƣời ký

* Xác thực văn bản: Chữ ký điện tử được sử dụng để xác thực tính toàn vẹn và không thay đổi của văn bản. Khi một tài liệu hoặc thông điệp được ký bằng chữ ký điện tử, bất kỳ sự thay đổi nào đối với nội dung của tài liệu sẽ làm thay đổi giá trị băm hoặc chữ ký điện tử. Do đó, bằng cách xác thực chữ ký điện tử, chúng ta có thể biết chắc rằng văn bản không bị sửa đổi hoặc chỉnh sửa từ nguồn gốc của nó.
* Xác thực danh tính người ký: Chữ ký điện tử cũng được sử dụng để xác thực danh tính của người ký. Mỗi chữ ký điện tử được tạo ra liên kết với một khóa bí mật độc nhất của người ký. Khi chữ ký điện tử được xác thực, chúng ta có thể xác định rằng chỉ có người ký chính thức mới có khả năng tạo ra chữ ký điện tử đó. Điều này đảm bảo tính xác thực của người ký và đồng thời giúp ngăn chặn việc giả mạo danh tính.

**Câu 17:** Xây dựng sơ đồ ký số và xác nhận chữ ký dựa trên thuật toán RSA và hàm băm

* Sơ đồ ký số

A picture containing diagram, text, plan, technical drawing

Description automatically generated

* Sơ đồ xác thực chữ ký số:

A diagram of a flowchart

Description automatically generated with low confidence

**Câu 18:** Tấn công phát lại là gì? Hãy đề xuất ít nhất 2 cách để chống lại tấn công phát lại.

* Tấn công phát lại (replay attack) là một hình thức tấn công trong đó kẻ tấn công ghi lại một luồng dữ liệu truyền qua mạng và sau đó chuyển tiếp hoặc chèn lại dữ liệu đó vào mạng trong một thời gian sau đó. Mục tiêu của tấn công phát lại là lợi dụng lại dữ liệu đã ghi lại để gây ra những hậu quả không mong muốn hoặc chiếm đoạt thông tin nhạy cảm.
* ***Để chống lại tấn công phát lại, có thể áp dụng các biện pháp bảo mật sau đây:***

1. ***Sử dụng giao thức bảo mật:***

* Sử dụng giao thức bảo mật như SSL/TLS để mã hóa dữ liệu truyền qua mạng và xác thực hai chiều giữa các bên giao tiếp.
* Sử dụng giao thức bảo mật cấp trên như HTTPS để đảm bảo tính toàn vẹn và xác thực của trang web.

1. ***Sử dụng mã hóa thông điệp:***

* Sử dụng mã hóa thông điệp (message encryption) để đảm bảo tính bảo mật của dữ liệu truyền qua mạng. Khi dữ liệu được mã hóa, ngay cả khi bị ghi lại, kẻ tấn công cũng không thể đọc được nội dung gốc.
* Sử dụng mã hóa kết hợp với chữ ký số (digital signature) để đảm bảo tính toàn vẹn và xác thực của dữ liệu truyền.

1. ***Sử dụng hạn chế thời gian:***

* Sử dụng timestamp hoặc hạn chế thời gian sống của dữ liệu để đảm bảo rằng dữ liệu chỉ có hiệu lực trong một khoảng thời gian nhất định. Khi dữ liệu đã quá hạn, nó sẽ không được chấp nhận và tấn công phát lại không thể xảy ra.
* Sử dụng mã xác thực một lần (one-time password) hoặc mã thông qua thời gian (time-based one-time password) để tạo ra một mã xác thực duy nhất và thay đổi theo thời gian. Mỗi mã chỉ có thể sử dụng một lần, ngăn chặn việc sử dụng lại mã cũ và tấn công phát lại.

**Câu 19:** Tìm hiểu cơ chế xác lập và trao đổi khóa bí mật trong SKC

* Cơ chế xác lập và trao đổi khóa bí mật trong Mã hóa Khóa đối xứng (Symmetric Key Cryptography - SKC) có thể được thực hiện bằng hai phương pháp chính: trao đổi khóa bí mật trực tiếp (Direct Key Exchange) và sử dụng giao thức trao đổi khóa (Key Exchange Protocol).

1. ***Trao đổi khóa bí mật trực tiếp:***

* Phương pháp này yêu cầu hai bên muốn giao tiếp cùng nhau trao đổi khóa bí mật trực tiếp trước khi bắt đầu quá trình mã hóa và giải mã dữ liệu.
* Một khóa bí mật được tạo ra và truyền trực tiếp từ bên A đến bên B thông qua một kênh an toàn, ví dụ như gặp trực tiếp hoặc qua tin nhắn bảo mật.
* Việc truyền khóa bí mật trực tiếp yêu cầu sự tin tưởng và sử dụng các phương tiện bảo mật khác như mã hóa mạnh và kênh truyền tin cậy.

1. ***Sử dụng giao thức trao đổi khóa:***

* Giao thức trao đổi khóa là phương pháp cho phép hai bên tạo ra khóa chung bí mật mà không cần truyền trực tiếp khóa qua kênh không an toàn.
* Giao thức trao đổi khóa thông thường dựa trên các thuật toán mã hóa và hàm băm an toàn, như Diffie-Hellman (DH) và Elliptic Curve Diffie-Hellman (ECDH).
* Trong giao thức trao đổi khóa, hai bên thỏa thuận trên một số tham số và thực hiện các phép tính số học để tạo ra khóa chung bí mật.
* Quá trình trao đổi khóa thông thường bao gồm việc gửi các gói tin chứa thông tin tham số và kết quả tính toán giữa hai bên. Quá trình này xảy ra trên một kênh không an toàn, nhưng không cho phép kẻ thứ ba lấy được khóa chung bí mật.

**Câu 20:** Tìm hiểu về giao thức Needham-Schroeder (3 bước, 5 bước), chỉ ra các vấn đề bảo mật còn tồn tại và đề xuất giải pháp khắc phục.

* Giao thức Needham-Schroeder là một giao thức trao đổi khóa sử dụng mã thông qua tin nhắn trong môi trường bất đối xứng. Giao thức này đã được phát triển bởi Roger Needham và Michael Schroeder vào năm 1978 và có thể được thực hiện trong hai phiên bản: giao thức Needham-Schroeder 3 bước và giao thức Needham-Schroeder 5 bước.
* Giao thức Needham-Schroeder 3 bước:

Bước 1: A gửi yêu cầu phiên làm việc (request for session) cho B, kèm theo ID(A) và ID(B).

Bước 2: B gửi một số ngẫu nhiên NonceB cho A, và mã hóa NonceB bằng khóa riêng của B.

Bước 3: A gửi lại NonceB đã được mã hóa bằng khóa riêng của B cho B. B sau đó giải mã NonceB và kiểm tra tính đúng đắn.

* Giao thức Needham-Schroeder 5 bước:

Bước 1: A gửi yêu cầu phiên làm việc (request for session) cho B, kèm theo ID(A) và ID(B).

Bước 2: B gửi một số ngẫu nhiên NonceB cho A, và mã hóa NonceB bằng khóa riêng của B.

Bước 3: A gửi lại NonceB đã được mã hóa bằng khóa riêng của B cho B.

Bước 4: B gửi một số ngẫu nhiên NonceA cho A, và mã hóa NonceA bằng khóa riêng của A.

Bước 5: A gửi lại NonceA đã được mã hóa bằng khóa riêng của A cho B. B sau đó giải mã NonceA và kiểm tra tính đúng đắn.

* Một số vấn đề bảo mật, bao gồm:
* Tấn công Playback (Replay Attack): Một kẻ tấn công có thể ghi lại các thông điệp đã mã hóa và gửi lại sau đó để giả mạo danh tính và lợi dụng phiên làm việc đã xác thực
* Tấn công Reflection (Reflection Attack): Một kẻ tấn công có thể sử dụng cùng một thông điệp gửi lại cho A và B, nhưng giả mạo như mình là một bên khác trong giao thức trao đổi khóa.
* Tấn công Man-in-the-Middle (MITM): Một kẻ tấn công có thể chèn mình vào quá trình giao tiếp giữa A và B, giả mạo thông điệp và thu thập thông tin nhạy cảm.
* Để khắc phục các vấn đề bảo mật tồn tại trong giao thức Needham-Schroeder, có thể áp dụng các giải pháp sau:
* Sử dụng Nonce (Số ngẫu nhiên duy nhất): Các bên có thể sử dụng Nonce để đảm bảo tính duy nhất của các thông điệp và tránh bị tấn công lại.
* Sử dụng Mã hóa và Chữ ký số: Áp dụng mã hóa và chữ ký số cho các thông điệp để đảm bảo tính bí mật, toàn vẹn và xác thực của thông điệp trong quá trình trao đổi.
* Sử dụng Giao thức trao đổi khóa an toàn: Sử dụng các giao thức trao đổi khóa như Diffie-Hellman hay Elliptic Curve Diffie-Hellman để tạo ra khóa chung bí mật mà không bị tấn công bởi các kỹ thuật như MITM.

**Câu 21:** Tìm hiểu cơ chế sử dụng PKC để trao đổi khóa bí mật

* Cơ chế sử dụng Mã hóa Khóa công khai (Public Key Cryptography - PKC) để trao đổi khóa bí mật thường được thực hiện thông qua giao thức trao đổi khóa.

1. Gửi khóa công khai:

* Mỗi bên tạo một cặp khóa (khóa công khai và khóa bí mật). Khóa công khai được chia sẻ công khai trong hệ thống.
* Bên A gửi khóa công khai của mình cho bên B.

1. Tạo khóa chung bí mật:

* Bên B nhận khóa công khai của bên A và sử dụng nó để mã hóa một khóa chung bí mật.
* Bên B gửi khóa chung bí mật đã được mã hóa cho bên A.

1. Giải mã khóa chung bí mật:

* Bên A nhận khóa chung bí mật đã được mã hóa từ bên B.
* Bên A sử dụng khóa bí mật của mình để giải mã khóa chung bí mật và nhận được khóa chung bí mật ban đầu.

Khi quá trình trên được hoàn thành, cả hai bên sẽ có cùng một khóa chung bí mật mà không cần truyền trực tiếp khóa qua kênh không an toàn. Khóa chung bí mật này có thể được sử dụng trong các thuật toán mã hóa đối xứng để mã hóa và giải mã các tin nhắn.

* Điểm quan trọng trong cơ chế này là sử dụng tính chất của PKC, trong đó khóa công khai được sử dụng để mã hóa và chỉ khóa bí mật tương ứng mới có thể giải mã. Điều này đảm bảo tính bảo mật của khóa chung bí mật trong quá trình truyền tin.

**Câu 22:** Tìm hiểu về giao thức thống nhất khóa Diffie-Healman

* Giao thức thống nhất khóa Diffie-Hellman (Diffie-Hellman key exchange protocol) là một giao thức trao đổi khóa bí mật được phát triển bởi Whitfield Diffie và Martin Hellman vào năm 1976. Giao thức này cho phép hai bên, thường là một người dùng và một máy chủ, trao đổi khóa bí mật thông qua một kênh không an toàn mà không cần truyền trực tiếp khóa.
* Cơ chế hoạt động của giao thức Diffie-Hellman như sau:

1. Khởi tạo thông số chia sẻ:

* Hai bên đồng thuận về một số nguyên tố lớn p và một số nguyên g, được gọi là cơ số.
* Cả hai bên đều biết p và g.

1. Tạo khóa riêng:

* Mỗi bên tạo một số ngẫu nhiên riêng của mình. Gọi số ngẫu nhiên của bên A là a và của bên B là b.
* Bên A tính toán khóa riêng của mình là A = g^a mod p.
* Bên B tính toán khóa riêng của mình là B = g^b mod p.

1. Trao đổi khóa công khai:

* Bên A gửi A cho bên B thông qua kênh không an toàn.
* Bên B gửi B cho bên A thông qua kênh không an toàn.

1. Tính toán khóa chung:

* Bên A tính toán khóa chung là K = B^a mod p.
* Bên B tính toán khóa chung là K = A^b mod p.

Kết quả là cả hai bên A và B đều tính toán được cùng một khóa chung K mà không cần truyền trực tiếp khóa qua kênh không an toàn.

* Một điểm mạnh của giao thức Diffie-Hellman là tính khả thi của việc tính toán khóa chung, trong khi việc tìm ngược lại khóa bí mật từ khóa chung được coi là khó khăn, tùy thuộc vào sự bảo mật của số nguyên tố p và cơ số g.
* Tuy nhiên, giao thức Diffie-Hellman không cung cấp bảo mật toàn diện và có một số vấn đề bảo mật cần được xem xét:
* Tấn công Man-in-the-Middle (MITM): Một kẻ tấn công có thể chèn mình vào quá trình trao đổi và trao đổi khóa riêng giả mạo với cả hai bên, sau đó giữ các khóa riêng của mình và mã hóa/giải mã các thông điệp giữa hai bên.
* Tấn công Nghe trộm (Eavesdropping): Một kẻ tấn công có thể nghe lén quá trình trao đổi và thu thập thông tin khóa riêng của cả hai bên, từ đó suy ra khóa chung.
* Để khắc phục các vấn đề bảo mật trong giao thức Diffie-Hellman, có thể áp dụng các giải pháp như sử dụng chữ ký số (digital signatures) để xác thực khóa công khai của mỗi bên và sử dụng mã hóa bổ sung như mã hóa đối xứng hoặc mã hóa không đối xứng để đảm bảo tính bảo mật và xác thực trong quá trình trao đổi thông điệp.

**Câu 20:** TÌm hiểu các khái niệm và thành phần trong một hệ thống xác thực. Tìm hiểu các kỹ thuật xác thực phổ biến:

* Xác thực dựa trên mật khẩu
* Xác thực dựa trên kỹ thuật thách thức – đáp ứng
* Xác thực dựa trên kỹ thuật sinh trắc học
* Xác thực dựa trên địa điểm.

Lấy ví dụ minh họa trong thực tế và mô tả quá trình thực hiện các bước xác thực.

* Hệ thống xác thực là một phần quan trọng trong bảo mật thông tin, nhiệm vụ của nó là xác định và xác minh danh tính của người dùng hoặc thiết bị trước khi cung cấp quyền truy cập vào hệ thống, dịch vụ hoặc tài nguyên. Dưới đây là một số khái niệm và thành phần chính trong một hệ thống xác thực:
* Người dùng/Thiết bị: Đại diện cho đối tượng mà hệ thống đang cố gắng xác thực. Có thể là người dùng cá nhân, máy tính, điện thoại di động, hoặc bất kỳ thiết bị nào yêu cầu quyền truy cập vào hệ thống.
* Danh tính (Identity): Đại diện cho thông tin định danh duy nhất của người dùng hoặc thiết bị trong hệ thống. Danh tính thường được biểu diễn dưới dạng tên người dùng, địa chỉ email, số điện thoại, hoặc các thông tin đặc biệt khác.
* Phương pháp xác thực (Authentication Method): Là cách thức mà hệ thống sử dụng để xác minh danh tính của người dùng hoặc thiết bị. Có nhiều phương pháp xác thực phổ biến, bao gồm:

1. Xác thực dựa trên mật khẩu (Password-based Authentication): Người dùng cung cấp một mật khẩu (password) và hệ thống so sánh với mật khẩu đã lưu trữ để xác minh danh tính.
2. Xác thực dựa trên kỹ thuật thách thức - đáp ứng (Challenge-Response Authentication): Hệ thống gửi một thách thức (challenge) đến người dùng và người dùng phải cung cấp một đáp ứng (response) chính xác để xác minh danh tính.
3. Xác thực dựa trên kỹ thuật sinh trắc học (Biometric Authentication): Sử dụng các đặc điểm sinh trắc học duy nhất của người dùng như vân tay, mống mắt, hoặc giọng nói để xác minh danh tính.
4. Xác thực dựa trên địa điểm (Location-based Authentication): Sử dụng thông tin về vị trí địa lý của người dùng hoặc thiết bị để xác minh danh tính. Ví dụ, kiểm tra địa chỉ IP hoặc vị trí GPS.

* Ví dụ:

1. ***Xác thực dựa trên mật khẩu:***

Bước 1: Người dùng cung cấp tên người dùng và mật khẩu.

Bước 2: Hệ thống so sánh mật khẩu được cung cấp với mật khẩu đã lưu trữ.

Bước 3: Nếu mật khẩu khớp, người dùng được cho phép truy cập vào hệ thống. Nếu không, quá trình xác thực thất bại.

1. ***Xác thực dựa trên kỹ thuật thách thức - đáp ứng:***

Bước 1: Hệ thống gửi một thách thức ngẫu nhiên đến người dùng.

Bước 2: Người dùng tính toán một đáp ứng dựa trên thách thức và một khóa bí mật hoặc thuật toán xác định.

Bước 3: Người dùng gửi đáp ứng cho hệ thống.

Bước 4: Hệ thống kiểm tra tính chính xác của đáp ứng và xác minh danh tính.

1. ***Xác thực dựa trên kỹ thuật sinh trắc học:***

Bước 1: Người dùng cung cấp dữ liệu sinh trắc học, ví dụ: quét vân tay, quét mống mắt, hoặc nói vào microphone.

Bước 2: Hệ thống so sánh dữ liệu sinh trắc học với mẫu đã được lưu trữ.

Bước 3: Nếu dữ liệu sinh trắc học khớp, người dùng được xác minh và cho phép truy cập

1. ***Xác thực dựa trên địa điểm:***

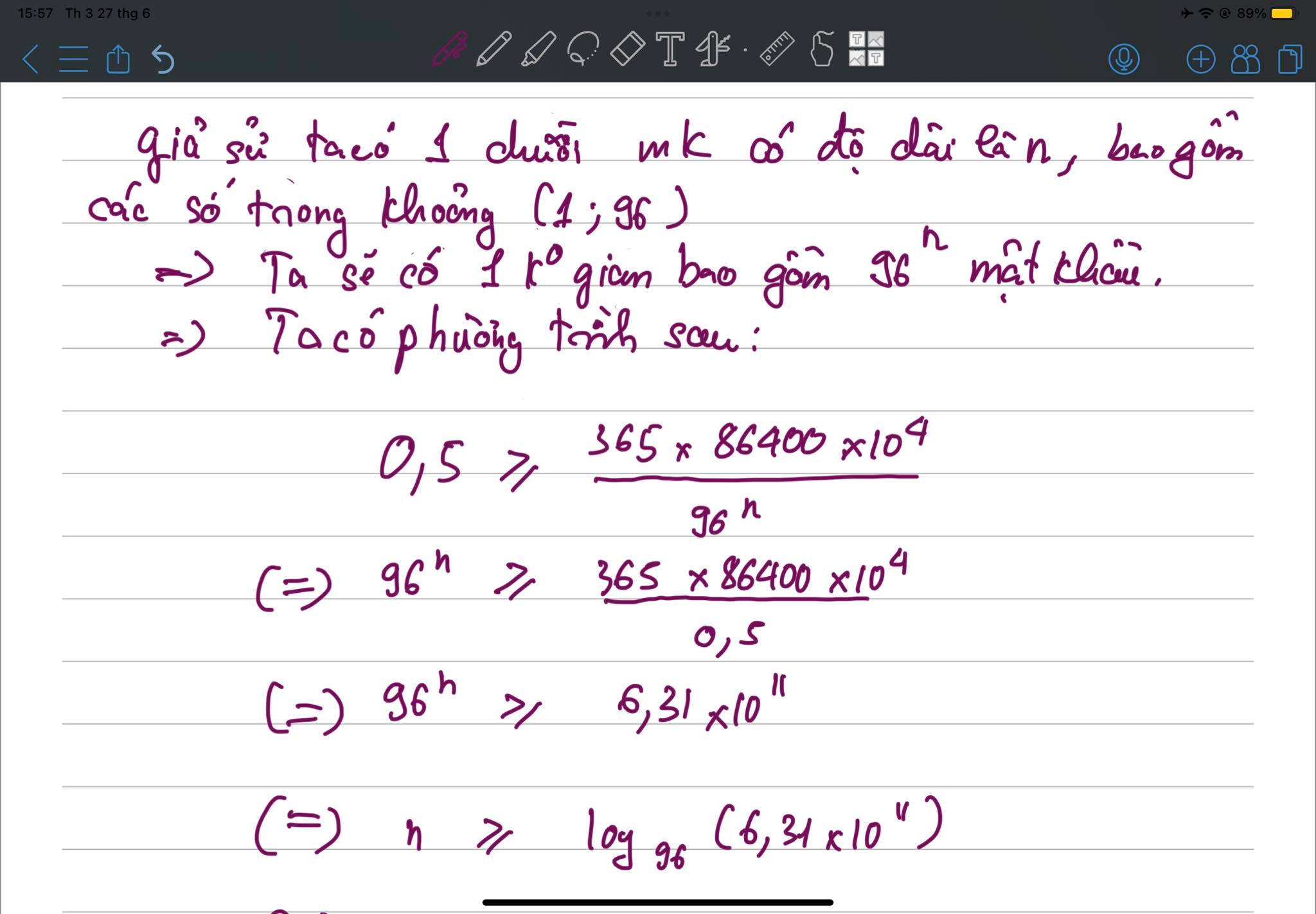
Bước 1: Hệ thống thu thập thông tin về vị trí địa lý của người dùng hoặc thiết bị, ví dụ: địa chỉ IP hoặc tọa độ GPS.

Bước 2: Hệ thống so sánh thông tin địa điểm với các quy tắc xác định trước để xác minh danh tính.

Bước 3: Nếu thông tin địa điểm hợp lệ, người dùng được xác minh và cho phép truy cập

**Câu 21:** Bài tập về tấn công mật khẩu.

Mật khẩu được tạo bởi các ký tự từ bảng chữ cái có kích thước 96. Giả thiết một kẻ tấn công có công cụ cho phép thử 104 mật khẩu trong một giây. Người thế kế hệ thống muốn đảm bảo kẻ địch không thể có cơ hội trên 50% trong tấn công vét cạn không gian mật khẩu trong vòng một năm. Hỏi độ dài mật khẩu cần quy định tối thiểu là bao nhiêu?

A screenshot of a tablet

Description automatically generated with low confidenceA screenshot of a computer

Description automatically generated with medium confidence

**Câu 22:** Tìm hiểu về giao thức keberos và ứng dụng trong hệ thống Active Directory.

* Giao thức Kerberos là một giao thức xác thực mạng phổ biến được sử dụng để xác thực và ủy quyền trong môi trường mạng. Nó được phát triển bởi MIT (Massachusetts Institute of Technology) và được sử dụng rộng rãi trong nhiều hệ thống và ứng dụng, trong đó có hệ thống Active Directory của Microsoft.
* Trong hệ thống Active Directory, giao thức Kerberos đóng vai trò quan trọng trong việc xác thực và ủy quyền người dùng để truy cập các tài nguyên trong mạng nội bộ. Dưới đây là mô tả sơ đồ hoạt động của giao thức Kerberos trong hệ thống Active Directory:

1. ***Yêu cầu Ticket TGT (Ticket Granting Ticket):***

* Người dùng đăng nhập vào hệ thống bằng tên người dùng và mật khẩu của mình.
* Máy trạm của người dùng gửi yêu cầu đăng nhập (AS\_REQ) tới máy chủ AS (Authentication Server) trong Active Directory.
* Máy chủ AS xác thực thông tin đăng nhập và trả về một Ticket TGT cho người dùng.

1. ***Xác thực Ticket TGT:***

* Người dùng gửi yêu cầu xác thực (TGS\_REQ) và Ticket TGT tới máy chủ TGS (Ticket Granting Server).
* Máy chủ TGS kiểm tra tính hợp lệ của Ticket TGT và phát hành một Ticket dịch vụ (Service Ticket) cho người dùng.

1. ***Sử dụng Ticket dịch vụ:***

* Người dùng gửi yêu cầu truy cập tài nguyên (AP\_REQ) cùng với Service Ticket tới máy chủ ứng dụng.
* Máy chủ ứng dụng xác thực tính hợp lệ của Service Ticket và cho phép người dùng truy cập tài nguyên.

Giao thức Kerberos trong hệ thống Active Directory giúp xác thực và ủy quyền truy cập cho người dùng một cách an toàn và bảo mật. Nó sử dụng mã hóa và chữ ký số để đảm bảo tính toàn vẹn và bảo mật của thông tin xác thực trong quá trình truyền tải.

* Các ứng dụng của giao thức Kerberos trong hệ thống Active Directory bao gồm:
* Xác thực người dùng: Kerberos xác thực danh tính của người dùng khi họ đăng nhập vào hệ thống, đảm bảo rằng chỉ những người dùng hợp lệ mới có thể truy cập vào tài nguyên.
* Ủy quyền truy cập: Kerberos cho phép người dùng nhận được Ticket dịch vụ để truy cập vào các tài nguyên khác mà họ được ủy quyền.
* SSO (Single Sign-On): Kerberos cho phép người dùng đăng nhập một lần và sử dụng phiên đăng nhập đó để truy cập vào nhiều ứng dụng trong hệ thống mà không cần phải đăng nhập lại.

**Câu 23:** Tìm hiểu về giao thức PGP và ứng dụng bảo mật trong email

* Giao thức PGP (Pretty Good Privacy) là một giao thức mã hóa thông tin được sử dụng để bảo mật trong gửi và nhận email. Nó được phát triển bởi Phil Zimmermann và trở thành một công cụ phổ biến để bảo vệ quyền riêng tư trong truyền thông điện tử.
* PGP sử dụng một hệ thống mã hóa khóa công khai (PKC) để đảm bảo tính bí mật, toàn vẹn và xác thực cho email. Dưới đây là một mô tả sơ đồ hoạt động của giao thức PGP trong việc bảo mật email:

1. Tạo cặp khóa:

* Người dùng tạo một cặp khóa công khai và khóa bí mật. Khóa công khai được chia sẻ với những người mà người dùng muốn gửi email bảo mật cho họ.

1. Mã hóa email:

* Người gửi sử dụng khóa công khai của người nhận để mã hóa email.
* Email được mã hóa sẽ không thể đọc được nếu không có khóa bí mật tương ứng.

1. Gửi email:

* Email đã mã hóa được gửi đi thông qua kênh truyền thông thường.

1. Giải mã email:

* Người nhận sử dụng khóa bí mật của mình để giải mã email.
* Email được hiển thị trong dạng thông thường cho người nhận đọc.
* Giao thức PGP trong ứng dụng bảo mật email có những ưu điểm sau:
* Bảo mật thông tin: PGP sử dụng mã hóa đáng tin cậy để đảm bảo rằng email chỉ có thể được đọc bởi người nhận mong muốn.
* Xác thực nguồn gốc: PGP sử dụng chữ ký số để xác thực danh tính của người gửi và đảm bảo rằng email không bị chỉnh sửa trên đường truyền.
* Quyền riêng tư: PGP cho phép người dùng giữ bí mật về nội dung email và chỉ chia sẻ thông tin cần thiết với người nhận.
* Ứng dụng của giao thức PGP trong bảo mật email là:
* Gửi email bảo mật: PGP cho phép người dùng gửi email được mã hóa để bảo vệ thông tin quan trọng khỏi sự xâm nhập.
* Bảo vệ quyền riêng tư: PGP giúp bảo vệ quyền riêng tư của người dùng, đặc biệt là trong việc chia sẻ thông tin nhạy cảm qua email.
* Xác thực nguồn gốc: PGP cho phép xác thực danh tính của người gửi, đảm bảo rằng email được gửi từ người dùng chính xác và không bị giả mạo.

**Câu 24:** Tìm hiểu về giao thức SSL/TLS và ứng dụng để bảo mật cho dịch vụ Web.

* Giao thức SSL (Secure Sockets Layer) và TLS (Transport Layer Security) là hai giao thức mã hóa mạnh mẽ được sử dụng để bảo mật việc truyền tải dữ liệu trên Internet, đặc biệt trong dịch vụ web. SSL đã được phát triển ban đầu bởi Netscape và sau đó đã được tiếp tục phát triển thành TLS.
* SSL/TLS hoạt động dựa trên hệ thống mã hóa khóa công khai (PKC) để đảm bảo tính bí mật, toàn vẹn và xác thực của dữ liệu được truyền tải giữa máy khách và máy chủ trong một phiên truyền thông an toàn. Dưới đây là mô tả sơ đồ hoạt động của giao thức SSL/TLS trong việc bảo mật dịch vụ web:

1. Giao tiếp khởi đầu:

* Máy khách (trình duyệt) gửi yêu cầu truy cập vào một trang web bảo mật qua HTTPS (HTTP Secure).
* Máy chủ nhận yêu cầu và gửi chứng chỉ số (Certificate) của mình cho máy khách.

1. Xác thực chứng chỉ:

* Máy khách xác thực tính hợp lệ của chứng chỉ bằng cách kiểm tra chứng chỉ với một cơ quan chứng thực tin cậy (Certificate Authority - CA) và kiểm tra xem chứng chỉ có hợp lệ không.
* Nếu chứng chỉ hợp lệ, máy khách tiếp tục quá trình trao đổi khóa bí mật.

1. Trao đổi khóa bí mật:

* Máy khách và máy chủ sử dụng giao thức Diffie-Hellman để thỏa thuận về một khóa bí mật chung.
* Khóa bí mật được sử dụng để mã hóa và giải mã dữ liệu trong quá trình truyền tải.

1. Mã hóa và giải mã dữ liệu:

* Dữ liệu truyền tải giữa máy khách và máy chủ được mã hóa bằng cách sử dụng khóa bí mật đã thỏa thuận.
* Máy chủ sẽ giải mã dữ liệu nhận được và xử lý yêu cầu từ máy khách.
* Giao thức SSL/TLS được sử dụng rộng rãi để bảo mật dịch vụ web trong nhiều ứng dụng và tình huống, bao gồm:
* Bảo mật trang web: SSL/TLS được sử dụng để bảo vệ truyền tải thông tin nhạy cảm như thông tin đăng nhập, thông tin tài chính và dữ liệu cá nhân trên các trang web bằng việc thiết lập kết nối an toàn giữa trình duyệt và máy chủ web.
* Thanh toán trực tuyến: Giao thức SSL/TLS được sử dụng trong các giao thức thanh toán trực tuyến như HTTPS để đảm bảo tính bảo mật và an toàn trong việc chuyển tiền và thông tin thanh toán qua mạng.
* Bảo mật ứng dụng di động: SSL/TLS được sử dụng để bảo mật việc truyền tải dữ liệu giữa ứng dụng di động và máy chủ, đảm bảo tính bí mật và toàn vẹn của thông tin như tin nhắn, tài liệu, hình ảnh trong quá trình truyền tải.
* Để đảm bảo an toàn và bảo mật cho dịch vụ web sử dụng SSL/TLS, cần chú ý các vấn đề sau:
* Sử dụng chứng chỉ SSL hợp lệ từ các cơ quan chứng thực tin cậy.
* Sử dụng các phiên bản SSL/TLS an toàn và loại bỏ các phiên bản cũ có lỗ hổng bảo mật.
* Thực hiện cấu hình chính sách bảo mật SSL/TLS phù hợp như sử dụng mã hóa mạnh, đảm bảo toàn vẹn và xác thực dữ liệu.
* Thực hiện kiểm tra định kỳ và cập nhật để bảo đảm rằng cơ sở hạ tầng SSL/TLS luôn được bảo mật và an toàn.

**Câu 25:** Tìm hiểu về mã độc mã hóa tống tiền (Ransomeware) và đề xuất một số giải pháp phòng chống/giảm thiểu ảnh hưởng của nó tới cá nhân/tổ chức.

* Ransomware là một loại phần mềm độc hại, có thể lây nhiễm vào máy tính nếu bạn nhấp vào một liên kết có chứa vi-rút.Khi liên kết này được mở ra, kẻ tấn công sẽ kiểm soát hệ thống máy tính của doanh nghiệp, từ đó truy cập và mã hóa các tập file quan trọng trong máy tính. Thủ phạm sau đó sẽ yêu cầu một khoản tiền đổi lấy các tập file đã bị đánh cắp. Một khi được tìm thấy,những dữ liệu quan trọng sẽ bị đánh cắp khỏi hệ thống. Các file bị đánh cắp được mã hóa, màn hình của nạn nhân sẽ bị khóa, đồng thời hiển thị thông báo máy bị nhiễm ransomware.
* Các bước lây nhiễm:

1. Phân phối:

Thủ phạm gửi email có đường link chứa vi-rút đến mục tiêu là các tổ chức và doanh nghiệp.

1. Lây nhiễm:

Khi người nhận mở đường link vi-rút sẽ được tải vào máy tính.

1. Kiểm soát:

Thủ phạm sau đó sẽ truy cập được vào thiết bị và kiểm soát hệ thống đã bị hack.

1. Phát hiện:

Sau khi kiểm soát được hệ thống, ransomware sẽ tìm kiếm dữ liệu quan trọng nhằm mục đích mã hóa.

1. Đánh cắp:

Một khi được tìm thấy,những dữ liệu quan trọng sẽ bị đánh cắp khỏi hệ thống.

1. Mã hóa:

Các file bị đánh cắp được mã hóa, màn hình của nạn nhân sẽ bị khóa, đồng thời hiển thị thông báo máy bị nhiễm ransomware.

1. Yêu cầu:

Kẻ tấn công sẽ yêu cầu một khoản tiền chuộc để nạn nhân đổi lấy mật mã và mở khóa hệ thống.

* Một số cách để phòng chống mã Ransomware:
* Tránh nhấp vào liên kết lạ, không rõ nguồn gốc
* Không chia sẻ thông tin cá nhân
* Không sử dụng USB không rõ nguồn gốc
* Không mở email và tập đính kèm lạ
* Thường xuyên nâng cấp, cập nhật hệ thống vận hành và chương trình
* Chỉ tải dữ liệu từ các nguồn đáng tin cậy
* Sử dụng VPN trên hệ thống wifi công cộng

**Câu 26:** Bài tập về giao thức mật mã

Với mỗi sinh viên, trước hết em hãy xác đinh cho mình giá trị X = (MSSV mod 2)

Y= (<Hàng chục của MSSV> mod 2). Ví dụ: MSSV 201432456 ➔ X=0, Y=1

Trong một giờ học về ATTT, thầy Lâm giới thiệu 1 giao thức đơn giản cơ bản PRT\_AU (xem dưới đây) và yêu cầu sinh viên bàn luận nhóm để phân tích mục đích & phê bình ưu-nhược; từ đó thử thiết kế một giao thức làm tốt hơn hoặc có lợi ích hơn về một phương diện nào đó. Kết quả cuối giờ có 4 sinh viên loại khá-giỏi là An, Bình, Chi, Dũng đã đưa ra các phiên bản mới dưới đây.

PRT\_AU: (thầy Lâm nêu đầu giờ)

1) A→S: IDA

2) S→A: Password Request

3) A→S: H(PWA)||T

|  |  |
| --- | --- |
| PROT\_00: (của An) 1) A→S: IDA 2) S→A: R 3) A→S: H(PWA||R) | PROT\_01: (của Chi) A→S: IDA || H(PWA||T) || T |
| PROT\_10: (của Bình) 1) A→S: IDA 2) S→A: R 3) A→S: H(PWA||R) 4) S→A: {ks}pA | PROT\_11: (của Dũng) 1) A→S: IDA 2) S→A: R 3) A→S: {R+1}pA 4) S→A: {ks}pA where pA= H(PWA) |

*Trong đó một số ký hiệu sử dụng ở trên có ý nghĩa như sau:*  
IDA – là ID (danh tính) của A; R là 1 số ngẫu nhiên; T là một nhãn thời gian (Timestamp) tại thời điểm hiện thời (có thể sử dụng computer clock)  
PWA là chuỗi ký tự mật khẩu của A; pA là giá trị băm của mật khẩu của A, tức là pA =H(PWA) và với H là một hàm băm xác định công bố trước.  
Câu hỏi:  
1. Phân tích rõ mục đích của PRT\_AU và từ đó so sánh với PROT\_0X để thấy rõ ưu nhược điểm

PRT\_AU:

***A→S: IDA***

***Người dùng (A) gửi thông tin định danh (IDA) của mình tới máy chủ (S).***

***S→A: Password Request***

***Máy chủ (S) nhận được thông tin định danh từ người dùng (A) và gửi yêu cầu mật khẩu (Password Request) cho người dùng.***

***A→S: H(PWA)||T***

***Người dùng (A) gửi lại một tin nhắn chứa mật khẩu đã được mã hóa (H(PWA)) kèm theo thông tin bổ sung (T) tới máy chủ (S). Ở đây, H() đại diện cho hàm băm (hash function) và PWA đại diện cho mật khẩu (password) của người dùng (A).***

PROT\_00:

***A→S: IDA***

***Người dùng (A) gửi thông tin định danh (IDA) của mình tới máy chủ (S).***

***S→A: R***

***Máy chủ (S) phản hồi bằng việc gửi một thông điệp ngẫu nhiên (R) cho người dùng (A). Thông điệp này có thể được sử dụng làm yêu cầu đáp ứng (challenge) trong quá trình xác thực.***

***A→S: H(PWA||R)***

***Người dùng (A) tính giá trị băm (hash) của sự kết hợp giữa mật khẩu (PWA) và thông điệp ngẫu nhiên (R) và gửi giá trị băm này (H(PWA||R)) tới máy chủ (S).***

***Thông qua quá trình trên, người dùng (A) xác thực bản thân bằng cách gửi thông tin định danh và mật khẩu đã được mã hóa kèm theo thông điệp ngẫu nhiên cho máy chủ (S). Máy chủ (S) nhận giá trị này và tiến hành xác thực bằng cách so sánh giá trị băm được gửi từ người dùng với giá trị băm của mật khẩu đã lưu trữ tương ứng trong hệ thống. Nếu hai giá trị băm khớp nhau, máy chủ (S) xác thực thành công và cho phép người dùng (A) truy cập vào hệ thống.***

PROT\_10: (của Bình)

***A→S: IDA***

***Người dùng (A) gửi thông tin định danh (IDA) của mình tới máy chủ (S).***

***S→A: R***

***Máy chủ (S) phản hồi bằng việc gửi một thông điệp ngẫu nhiên (R) cho người dùng (A). Thông điệp này có thể được sử dụng làm yêu cầu đáp ứng (challenge) trong quá trình xác thực.***

***A→S: H(PWA||R)***

***Người dùng (A) tính giá trị băm (hash) của sự kết hợp giữa mật khẩu (PWA) và thông điệp ngẫu nhiên (R) và gửi giá trị băm này (H(PWA||R)) tới máy chủ (S).***

***S→A: {ks}pA***

***Máy chủ (S) sử dụng khóa riêng (ks) của người dùng (A) để mã hóa một thông điệp (pA) và gửi lại cho người dùng (A).***

***Giao thức này là một giao thức xác thực dựa trên mật khẩu với sử dụng mã hóa đối xứng. Sau khi người dùng (A) gửi giá trị băm đến máy chủ (S), máy chủ (S) sử dụng khóa riêng của người dùng (A) để mã hóa một thông điệp và gửi lại cho người dùng (A). Người dùng (A) sau đó sẽ sử dụng khóa riêng của mình để giải mã thông điệp nhận được từ máy chủ (S).***

PROT\_01: (của Chi)

***A→S: IDA || H(PWA||T) || T***

***Người dùng (A) gửi thông tin định danh (IDA) của mình kèm theo giá trị băm (hash) của mật khẩu (PWA) và thông điệp ngẫu nhiên (T) cho máy chủ (S).***

***Thông qua quá trình trên, người dùng (A) xác thực bản thân bằng cách gửi thông tin định danh, giá trị băm của mật khẩu đã được mã hóa kèm theo thông điệp ngẫu nhiên cho máy chủ (S). Máy chủ (S) nhận giá trị này và tiến hành xác thực bằng cách kiểm tra tính hợp lệ của thông tin định danh và giá trị băm, cũng như sự tương thích với thông điệp ngẫu nhiên. Nếu thông tin định danh và giá trị băm được coi là hợp lệ, máy chủ (S) xác thực thành công và cho phép người dùng (A) truy cập vào hệ thống.***

PROT\_11: (của Dũng)

***A→S: IDA***

***Người dùng (A) gửi thông tin định danh (IDA) của mình tới máy chủ (S).***

***S→A: R***

***Máy chủ (S) phản hồi bằng việc gửi một thông điệp ngẫu nhiên (R) cho người dùng (A). Thông điệp này có thể được sử dụng làm yêu cầu đáp ứng (challenge) trong quá trình xác thực.***

***A→S: {R+1}pA***

***Người dùng (A) sử dụng khóa riêng của mình để mã hóa thông điệp R+1 và gửi cho máy chủ (S). Thông điệp này được mã hóa bằng cách sử dụng khóa riêng của người dùng (A) và có thể chứa một số mã băm (H(PWA)) của mật khẩu của người dùng (A).***

***S→A: {ks}pA where pA = H(PWA)***

***Máy chủ (S) sử dụng khóa riêng (ks) của người dùng (A) để mã hóa một thông điệp (pA) chứa giá trị băm (H(PWA)) của mật khẩu của người dùng (A) và gửi lại cho người dùng (A).***

***Giao thức này là một giao thức xác thực dựa trên mật khẩu với sử dụng mã hóa đối xứng. Sau khi người dùng (A) gửi thông điệp mã hóa đến máy chủ (S), máy chủ (S) sử dụng khóa riêng của người dùng (A) để giải mã thông điệp và kiểm tra tính hợp lệ của giá trị băm (H(PWA)). Nếu thông điệp nhận được và giá trị băm trùng khớp, máy chủ (S) xác thực thành công và cho phép người dùng (A) truy cập vào hệ thống.***

2. Phân tích chỉ rõ sự khác biệt và tính năng mới của PROT\_1Y so với hai giao thức ở câu trên (1)

3. Thầy Lâm sau đó trình bày một phiên bản nâng cao khác và thách thức lớp phân tích và tìm ra vấn đề  
còn tồn tại.

1) A→S: IDA || IDB || r1  
2) S→A: {IDA || IDB || r1 || ks || {IDA || ks || r1} pB } pA  
3) A→B: {IDA || ks || r1} pB  
4) B→A: {r2} ks  
5) A→B: {r1 + r2} ks

Sinh viên Đạt đã trả lời tốt câu hỏi của thầy, phân tích đúng vấn đề tồn tại và tìm ra giải pháp khắc phục  
thành công, trong đó chỉ cần thực hiện 1 sửa đổi ở bước 5 mà thôi.  
*Em hãy phân tích mục đích của giao thức và thử đoán xem vấn đề tồn tại mà Đạt chỉ ra? Nếu có thể hãy  
chỉ ra giải pháp khắc phục cũng như ý tưởng giải quyết của Đạt.*