Full name: Ngô Minh Khôi

**FINAL TEST – NSC**

**Câu 1: Giới thiệu về mã hóa đối xứng và mã hóa bất đối xứng**

1.1 Khái niệm:

Mã hóa đối xứng và mã hóa bất đối xứng là hai phương pháp trong lĩnh vực bảo mật thông tin để bảo vệ dữ liệu trước khi truyền hoặc lưu trữ.

Mã hóa đối xứng: Trong mã hóa đối xứng, cùng một khóa được sử dụng cho cả quá trình mã hóa và giải mã. Khóa này phải được bảo mật và phải được chia sẻ giữa người gửi và người nhận. Dữ liệu được mã hóa và sau đó giải mã bằng cùng một khóa. Đây là phương pháp nhanh chóng và hiệu quả, nhưng yêu cầu quản lý và bảo vệ khóa một cách an toàn.

Mã hóa bất đối xứng: Trong mã hóa bất đối xứng, có hai khóa: một khóa công khai (public key) và một khóa bí mật (private key). Dữ liệu được mã hóa bằng khóa công khai và chỉ có thể được giải mã bằng khóa bí mật tương ứng. Khóa công khai có thể được công bố rộng rãi, trong khi khóa bí mật phải được bảo mật một cách an toàn. Phương pháp này cung cấp tính bảo mật cao hơn và không yêu cầu chia sẻ khóa bí mật.

1.2 Vẽ sơ đồ

a) Mã hóa đối xứng



b) Mã hóa bất đối xứng



1.3 So sánh:

Mã hóa đối xứng nhanh hơn và hiệu quả hơn đối với lượng dữ liệu lớn, nhưng yêu cầu quản lý khóa an toàn.

Mã hóa bất đối xứng cung cấp tính bảo mật cao hơn vì không cần chia sẻ khóa bí mật, nhưng tốn thời gian hơn trong quá trình mã hóa và giải mã.

1.4 Viết chương trình mã hóa và giải mã bất đối xứng bằng ngôn ngữ C#

**Câu 2: Lỗ hổng bảo mật**

2.1 Khái niệm

Lỗ hổng bảo mật là các điểm yếu trong hệ thống hoặc ứng dụng mà kẻ tấn công có thể sử dụng để xâm nhập, thực hiện các hành vi không được ủy quyền hoặc gây hại đến hệ thống, dữ liệu hoặc tài sản của một tổ chức hoặc cá nhân. Lỗ hổng bảo mật có thể xuất hiện trong cách thức thiết kế, triển khai hoặc quản lý hệ thống, và chúng có thể được sử dụng để tạo cơ hội cho việc tấn công và xâm nhập.

2.2 SQL Injection và cách phòng tránh

SQL Injection là một loại lỗ hổng bảo mật xảy ra khi kẻ tấn công chèn các truy vấn SQL độc hại vào các trường đầu vào, cho phép chúng thao túng cơ sở dữ liệu và có khả năng truy cập trái phép hoặc đánh cắp dữ liệu.

Để ngăn chặn SQL Injection, hãy làm theo các phương pháp sau:

* Sử dụng truy vấn được tham số hóa: Thay vì nhúng trực tiếp thông tin đầu vào của người dùng vào câu lệnh SQL, hãy sử dụng truy vấn được tham số hóa hoặc câu lệnh được chuẩn bị sẵn do thư viện cơ sở dữ liệu cung cấp. Chúng tự động thoát và vệ sinh đầu vào của người dùng
* Xác thực đầu vào: Xác thực và vệ sinh đầu vào của người dùng ở cả phía máy khách và máy chủ để đảm bảo chúng tuân thủ các mẫu và định dạng dự kiến. Từ chối đầu vào không đáp ứng tiêu chí xác thực.
* Nguyên tắc đặc quyền tối thiểu: Giới hạn quyền được cấp cho người dùng cơ sở dữ liệu. Đảm bảo họ chỉ có những đặc quyền cần thiết tối thiểu để thực hiện nhiệm vụ của mình.
* Thoát khỏi đầu vào của người dùng: Nếu bạn phải nhúng đầu vào của người dùng vào các truy vấn SQL, hãy thoát và loại bỏ nó để ngăn chặn việc tiêm SQL. Nhiều ngôn ngữ và khung công tác cung cấp các chức năng cho mục đích này.

**Câu 3: Mã hóa**