BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC ĐẠI NAM

KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO BÀI TẬP LỚN

NHẬP MÔN AN TOÀN BẢO MẬT THÔNG TIN

GỬI BÀI TẬP CHIA THÀNH NHIỀU PHẦN

Sinh viên thực hiện : Nguyễn Quý Trung - 1771020690
Nguyễn Hùng Dũng - 1771020183
Lê Quốc Nhật - 1451020169

Ngành : Công nghệ thông tin

Giảng viên hướng dẫn : ThS. Lê Thị Thùy Trang

Lời cảm ơn

Trong bối cảnh công nghệ thông tin phát triển mạnh mẽ, việc bảo mật dữ liệu khi truyền tải qua mạng đã trở thành một yêu cầu cấp thiết. Đặc biệt đối với các tổ chức giáo dục, nơi thường xuyên cần trao đổi các tài liệu quan trọng như bài tập, đề thi hay nghiên cứu khoa học, việc đảm bảo tính bảo mật và toàn vẹn của dữ liệu càng trở nên quan trọng. Xuất phát từ nhu cầu thực tế đó, chúng em đã quyết định thực hiện đề tài "Xây dựng hệ thống truyền file bảo mật sử dụng DES và RSA" như một giải pháp khả thi cho bài toán này.

Trong quá trình nghiên cứu và phát triển hệ thống, chúng em đã gặp phải nhiều thách thức đáng kể. Từ việc lựa chọn thuật toán mã hóa phù hợp, triển khai cơ chế trao đổi khóa an toàn, cho đến xử lý các vấn đề về hiệu năng khi làm việc với các file có kích thước lớn. Mỗi khó khăn đều đòi hỏi sự tìm tòi, phân tích kỹ lưỡng và thử nghiệm nhiều phương án khác nhau.

Chúng em xin chân thành cảm ơn sự hướng dẫn tận tình của giáo viên bộ môn cùng những góp ý quý báu từ các bạn sinh viên trong suốt quá trình thực hiện đề tài. Những đóng góp đó đã giúp chúng em hoàn thiện hệ thống một cách tốt nhất có thể.

Mặc dù đã cố gắng hết sức, nhưng do hạn chế về thời gian và kinh nghiệm, hệ thống chắc chắn không tránh khỏi những thiếu sót. Chúng em rất mong nhận được những ý kiến đóng góp quý báu từ thầy cô và các bạn để có thể tiếp tục hoàn thiện đề tài trong tương lai.

Hy vọng rằng kết quả của đề tài này sẽ là một tài liệu tham khảo hữu ích cho những ai quan tâm đến lĩnh vực bảo mật dữ liệu và an toàn thông tin.

Mục lục

Danh sách bảng

Danh sách giải thuật

Chương 1

Giới thiệu

Trong bối cảnh công nghệ thông tin phát triển nhanh chóng, nhu cầu truyền tải dữ liệu một cách an toàn và bảo mật ngày càng trở nên cấp thiết, đặc biệt trong các tổ chức giáo dục. Việc chia sẻ các tệp tin quan trọng như bài tập, đề thi, hoặc tài liệu nghiên cứu trên môi trường mạng công cộng tiềm ẩn nhiều rủi ro như bị nghe lén, thay đổi hoặc giả mạo dữ liệu.

Nhằm giải quyết vấn đề trên, nhóm chúng em lựa chọn đề tài **"Xây dựng hệ thống truyền file bảo mật sử dụng DES và RSA"** làm bài tập lớn cho học phần Nhập môn An toàn bảo mật thông tin.

1.1 Đặt vấn đề

Hệ thống được đề xuất nhằm bảo vệ tệp tin được truyền giữa người gửi (Sender) và người nhận (Receiver), với các yêu cầu bảo mật như sau:

- **Bảo mật dữ liệu:** Nội dung file cần được mã hóa để tránh bị lộ khi truyền qua Internet.
- **Xác thực nguồn gốc:** Đảm bảo rằng file đến từ đúng người gửi, ngăn chặn giả mạo.
- **Toàn vẹn dữ liệu:** Phát hiện nếu file bị sửa đổi trong quá trình truyền.

Hệ thống cần đảm bảo hiệu quả ngay cả khi hoạt động trên đường truyền có **băng thông thấp** và không an toàn.

1.1.1 Phân tích bài toán

Từ yêu cầu thực tế, bài toán được phân tích thành các chức năng bảo mật chính:

1. **Mã hoá dữ liệu:** Sử dụng thuật toán mã hóa đối xứng (DES) để bảo vệ nội dung file.

2. **Xác thực người dùng:** Ứng dụng chữ ký số (RSA + SHA-512) để xác minh danh tính Sender.

3. **Kiểm tra toàn vẹn:** Sử dụng hàm băm SHA-512 để phát hiện thay đổi dữ liệu trong quá trình truyền tải.

Ngoài ra, hệ thống phải dễ triển khai, hiệu quả trong môi trường giáo dục, và có khả năng phát hiện tấn công phổ biến như MITM hoặc Replay Attack.

Cách trích dẫn tài liệu tham khảo trong LaTeX được thực hiện như sau: [1].

1.2 Mục tiêu của đề tài

Mục tiêu tổng quát là xây dựng một hệ thống bảo mật truyền tệp tin qua mạng. Các mục tiêu cụ thể:

- Xây dựng quy trình truyền file chia thành nhiều phần để dễ kiểm soát và bảo vệ.
- Kết hợp các thuật toán mã hóa (DES), chữ ký số (RSA-SHA512) và kiểm tra toàn vẹn (SHA-512).
- Thử nghiệm hệ thống trong môi trường mạng không an toàn, đo lường hiệu năng.
- Đề xuất giải pháp cải tiến nhằm nâng cao bảo mật và hiệu suất.

1.3 Cấu trúc của báo cáo

Báo cáo gồm các phần chính sau:

- **Chương 1:** Giới thiệu – Đặt vấn đề, phân tích bài toán và mục tiêu đề tài.
- **Chương 2:** Phân tích yêu cầu và thiết kế ứng dụng – Mô tả quy trình, mã nguồn và thử nghiệm.
- **Chương 3:** Kết luận và hướng phát triển – Đánh giá hiệu quả hệ thống và đề xuất mở rộng.

Chương 2

Phân tích yêu cầu và thiết kế ứng dụng

Chương này trình bày chi tiết các yêu cầu của hệ thống, mô tả thuật toán được sử dụng trong bài toán truyền file bảo mật, phân tích mã nguồn đã triển khai và thử nghiệm thực tế.

2.1 Mô tả thuật toán

Hệ thống sử dụng kết hợp các thuật toán mật mã để đảm bảo tính bảo mật, toàn vẹn và xác thực:

- **DES (CBC mode)**: Mã hóa từng phần file sau khi chia nhỏ.
- **RSA 1024-bit**: Mã hóa khóa phiên (session key) và thực hiện chữ ký số.
- **SHA-512**: Tạo chuỗi hash kiểm tra toàn vẹn dữ liệu.
- **RSA + SHA-512**: Ký số metadata và từng phần dữ liệu.

Quy trình tổng thể:

1. Sender chia file thành 3 phần.
2. Tạo `session key` ngẫu nhiên cho DES.
3. Ký metadata (tên file, thời gian, số phần) bằng private key.
4. Mã hóa session key bằng public key của Receiver.
5. Mỗi phần dữ liệu được mã hóa bằng DES (CBC) + IV, hash SHA-512, ký số hash.
6. Receiver giải mã session key, kiểm tra chữ ký và ghép file.

2.2 Phân tích mã nguồn

2.2.1 Chia file

```
def split_file(file_path, num_parts=3):
    with open(file_path, 'rb') as f:
        data = f.read()
    part_size = len(data) // num_parts
    return [data[i*part_size:(i+1)*part_size] for i in range(num_parts-1)] + [data[
```

2.2.2 Mã hóa bằng DES

```
def encrypt_part(data, session_key):
    iv = get_random_bytes(8)
    cipher = DES.new(session_key, DES.MODE_CBC, iv)
    padded_data = pad(data, DES.block_size)
    ciphertext = cipher.encrypt(padded_data)
    return iv, ciphertext
```

2.2.3 Xác thực tại Receiver

```
def verify_packet(iv, ciphertext, received_hash, signature):
    computed_hash = hashlib.sha512(iv + ciphertext).hexdigest()
    if computed_hash != received_hash:
        return False
    try:
        pkcs1_15.new(sender_public_key).verify(SHA512.new(iv + ciphertext), signatu
        return True
    except:
        return False
```

2.3 Thử nghiệm

2.3.1 Test case 1: Truyền file thành công

- Handshake: thành công (Sender nhận được "Ready!").
- Gửi 3 packet dữ liệu → Receiver nhận đúng thứ tự.
- Ghép file → hash khớp với bản gốc.

2.3.2 Test case 2: Giả mạo packet

Giả lập sửa trường `cipher` trong packet:

- Receiver phát hiện sai lệch hash.
- Hệ thống từ chối packet → báo lỗi.

2.3.3 So sánh hiệu suất mã hóa

Bảng 2.1: Thời gian mã hóa DES vs AES-256

Dung lượng	DES (s)	AES-256 (s)
10MB	1.2	0.8
100MB	12.8	8.2

2.3.4 Mô phỏng luồng dữ liệu

Giải thuật 1: Xử lý từng phần dữ liệu tại Sender

```
data_parts ← split_file("assignment.txt")
for each part in data_parts do
    iv, ciphertext ← encrypt_part(part, session_key)
    hash ← SHA512(iv + ciphertext)
    sig ← RSA_sign(hash)
    packet ← {iv, ciphertext, hash, sig}
    send(packet)
end for
```

2.3.5 Kết luận thử nghiệm

Hệ thống hoạt động ổn định trên mạng băng thông thấp. Tốc độ xử lý và khả năng phát hiện giả mạo dữ liệu đạt yêu cầu.

Chương 3

Kết luận và hướng phát triển

3.1 Phân tích hiệu quả

Qua quá trình triển khai và thử nghiệm hệ thống truyền file bảo mật sử dụng các thuật toán mã hóa và chữ ký số, nhóm nhận thấy rằng hệ thống đáp ứng đầy đủ các tiêu chí về bảo mật, toàn vẹn dữ liệu và xác thực nguồn gốc. Cụ thể:

- Hệ thống hoạt động ổn định trong môi trường mạng có băng thông thấp.
- Dữ liệu được chia nhỏ, mã hóa và ký số giúp đảm bảo an toàn từng phần.
- Phát hiện chính xác các gói tin bị giả mạo nhờ kiểm tra hash và chữ ký số.
- Thử nghiệm cho thấy các gói tin hợp lệ được xử lý và ghép lại chính xác thành file gốc.

3.1.1 Phân tích và nhận xét đặc điểm của các thuật toán sử dụng

- **DES (CBC mode)**: Dễ triển khai, đủ dùng cho bài toán nhỏ, nhưng độ an toàn không cao do khóa 56-bit, dễ bị brute-force nếu dùng lâu dài.
- **RSA 1024-bit**: Bảo mật ở mức cơ bản, tuy nhiên tốc độ chậm, không phù hợp với dữ liệu lớn hoặc xử lý đồng thời.
- **SHA-512**: Có độ an toàn cao, chống va chạm tốt, phù hợp cho kiểm tra toàn vẹn và tạo đầu vào cho ký số.
- **Kết hợp RSA + SHA-512**: Đảm bảo xác thực nguồn gốc dữ liệu, chống giả mạo hiệu quả.

3.2 Đề xuất cải tiến

Dựa trên kết quả thực tế và giới hạn hiện tại, nhóm đề xuất một số hướng cải tiến trong tương lai như sau:

- **Nâng cấp thuật toán mã hóa:** Thay thế DES bằng AES-256-GCM giúp tăng cả tính bảo mật và hiệu suất.
- **Tối ưu hiệu suất:**
  - Áp dụng nén dữ liệu trước khi mã hóa (`zlib`).
  - Sử dụng đa luồng để mã hóa và gửi các phần dữ liệu đồng thời.
- **Mở rộng chức năng:**
  - Tích hợp giao diện đồ họa (GUI) để người dùng dễ thao tác hơn.
  - Xác thực hai chiều: cho phép cả Receiver ký ACK xác nhận.
  - Kết nối với lưu trữ đám mây (VD: AWS S3) để gửi/nhận file.
- **Tăng độ an toàn hệ thống:**
  - Áp dụng xác thực ECDH hoặc kết hợp với OTP để chống MITM.
  - Kiểm tra gói tin kết hợp thời gian thực và blacklist nếu phát hiện bất thường.

Tổng kết lại, hệ thống hiện tại phù hợp cho các ứng dụng giáo dục hoặc môi trường truyền file nhỏ. Tuy nhiên, để mở rộng cho mục đích thực tế hoặc thương mại, cần có thêm các cải tiến về hiệu năng, độ tin cậy và bảo mật.

Tài liệu tham khảo

[1] Timothy Van Zandt and Timothy VAN. Documentation for fancybox. sty: Box tips and tricks for latex, 2010.