TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

KHOA MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG

BÁO CÁO BÀI TẬP THỰC HÀNH

LAB 3 OFF CLASS

MÔN HỌC: MẬT MÃ HỌC

Họ và tên: LẠI QUAN THIÊN

Mã số sinh viên: 22521385

Lớp: NT219.O21.ANTT

TP. HỒ CHÍ MINH, THÁNG 06 NĂM 2024

MỤC LỤC

PHẦN I: TỔNG QUAN VÀ MÔ TẢ

1.1. Thông tin cá nhân:

Họ và tên: Lại Quan Thiên

Mã số sinh viên: 22521385

Lớp thực hành: NT219.O21.ANTT.1

Link github: Cryptography-Course/Labs/OffClass/Lab_3 at main · WanThinnn/Cryptography-Course (github.com)

1.2. Thông tin thiết bị:

- Thiết bị: Macbook Air 2019 – RAM 8GB (LPDDR3 2133MHz) – SSD 128GB

- Hệ điều hành:

+ Windows 11 Pro (cài đặt thông qua Bootcamp của Apple)

+ Ubuntu 22.04 Jammy Jellyfish (cài đặt thông qua t2linux.org)

- Bộ xử lý:

+ Intel Core i5 8210Y – 1.60GHz – Turbo Boost 3.60Ghz

+ Intel UHD Graphics 617

+ Apple T2 Security Chip

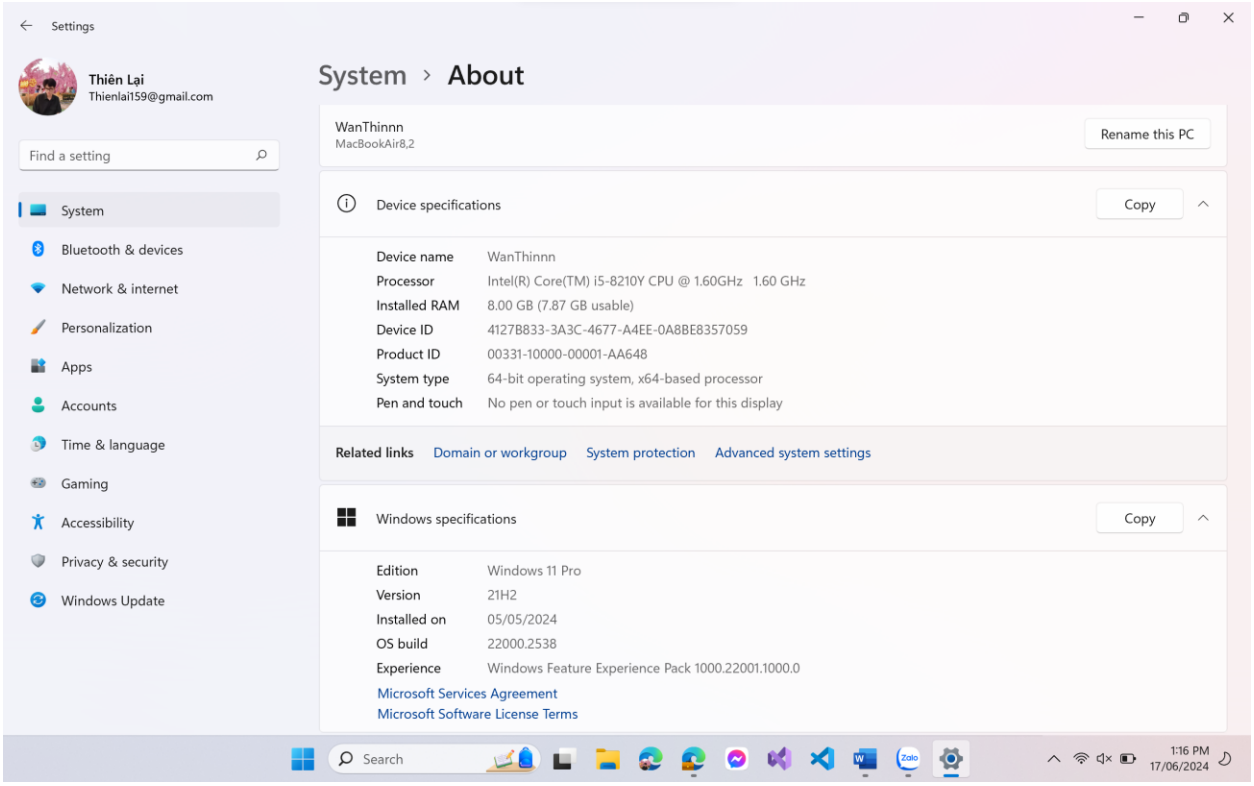
- Thông tin về bộ xử lý:

+ Intel Core i58210Y Processor 4M Cache up to 3.60 GHz Thông số kỹ thuật sản phẩm

+ Hỗ trợ Intel® UHD Graphics 617

+ Apple T2 Security Chip: Security Overview

- Các thực nghiệm được thực thi trong quá trình laptop cắm điện, nhiệt độ phòng.


Settings
Thiên Lại
Thienlai159@gmail.com
Find a setting
System
Bluetooth & devices
Network & internet
Personalization
Apps
Accounts
Time & language
Gaming
Accessibility
Privacy & security
Windows Update
System › About
WanThinnn
MacBookAir8,2
Rename this PC
Device specifications
Copy
Device name WanThinnn
Processor Intel(R) Core(TM) i5-8210Y CPU @ 1.60GHz 1.60 GHz
Installed RAM 8.00 GB (7.87 GB usable)
Device ID 4127B833-3A3C-4677-A4EE-0A8BE8357059
Product ID 00331-10000-00001-AA648
System type 64-bit operating system, x64-based processor
Pen and touch No pen or touch input is available for this display
Related links Domain or workgroup System protection Advanced system settings
Windows specifications
Copy
Edition Windows 11 Pro
Version 21H2
Installed on 05/05/2024
OS build 22000.2538
Experience Windows Feature Experience Pack 1000.22001.1000.0
Microsoft Services Agreement
Microsoft Software License Terms
Search
1:16 PM
17/06/2024

1.3. Tổng quan:

Bài báo cáo bao gồm việc triển khai mã nguồn để thực hiện thuật toán RSA-OAEP bằng ngôn ngữ C++, sử dụng thư viện CryptoPP để hỗ trợ mã hóa và giải mã các file đầu vào.

Sau khi xây dựng mã nguồn, em đã tạo 3 tệp tin với các kích thước khác nhau và tiến hành đo thời gian thực hiện 10000 lần mã hóa/giải mã trên cả hai hệ điều hành Windows và Linux. Cuối cùng, kết quả được thống kê và biểu diễn dưới dạng biểu đồ để phân tích và so sánh. Chi tiết cụ thể sẽ được trình bày trong các mục sau.

Dưới đây là hình ảnh minh hoạ phần code chạy 10 000 lần của RSA-OAEP (bao gồm toàn bộ quá trình load file, check format, thực hiện mã hoá/giải mã và lưu/xuất file):

```
else if (mode == "encrypt" && argc == 8)
{
    auto start = std::chrono::high_resolution_clock::now();
    for (int i = 0; i < 10000; i++){
        RSAencrypt(argv[2], argv[3], argv[4], argv[5],argv[6], argv[7]);

    }
    auto stop = std::chrono::high_resolution_clock::now();
    auto duration = std::chrono::duration_cast<std::chrono::milliseconds>(stop - start);

    cout << "Overview RSA Encryption Test" << endl;
    cout << "Encryption time: " << duration.count() << " milliseconds" << endl;
    cout << "------------------------------------------------------------" << endl;

    cout.flush(); // Đảm bảo dữ liệu được ghi vào cout sẽ được xuất ra màn hình ngay lập tức

}
else if (mode == "decrypt" && argc == 8)
{



    auto start = std::chrono::high_resolution_clock::now();
    for (int i = 0; i < 10000; i++){
        RSAdecrypt(argv[2], argv[3], argv[4], argv[5],argv[6], argv[7]);

    }
    auto stop = std::chrono::high_resolution_clock::now();
    auto duration = std::chrono::duration_cast<std::chrono::milliseconds>(stop - start);

    cout << "Overview RSA Decryption Test" << endl;
    cout << "Decryption time: " << duration.count() << " milliseconds" << endl;
    cout << "------------------------------------------------------------" << endl;

    cout.flush(); // Đảm bảo dữ liệu được ghi vào cout sẽ được xuất ra màn hình ngay lập tức
}
```

PHẦN II: NỘI DUNG THỰC HÀNH

2.1. RSA-OAEP: RSA - Optimal asymmetric encryption padding

2.2.1. Bảng thống kê số liệu thực nghiệm:

Dưới đây là bảng thống kê chi tiết thời gian encrypt/decrypt của từng mode (đơn vị thời gian: ms):

2.2.1.1. Hệ điều hành Windows 11:

- **Bảng thời gian trung bình 10 000 lần Encrypt:**

Encrypt			
Key Size	**8912**	**10240**	**12288**
File 1 (0.4KB)	2.68		
File 2 (1KB)		2.6869	
File 3 (1.4KB)			2.595

- **Bảng thời gian trung bình 10 000 lần Decrypt:**

Decrypt			
Key Size	**8912**	**10240**	**12288**
File 1 (0.4KB)	141.5941		
File 2 (1KB)		140.9495	
File 3 (1.4KB)			225.5939

2.2.1.2. Hệ điều hành Ubuntu 22.04 (Linux):

- **Bảng thời gian trung bình 10 000 lần Encrypt:**

Encrypt			
Key Size	**8912**	**10240**	**12288**
File 1 (0.4KB)	0.6481		
File 2 (1KB)		1.2296	
File 3 (1.4KB)			1.1138

- **Bảng thời gian trung bình 10 000 lần Decrypt:**

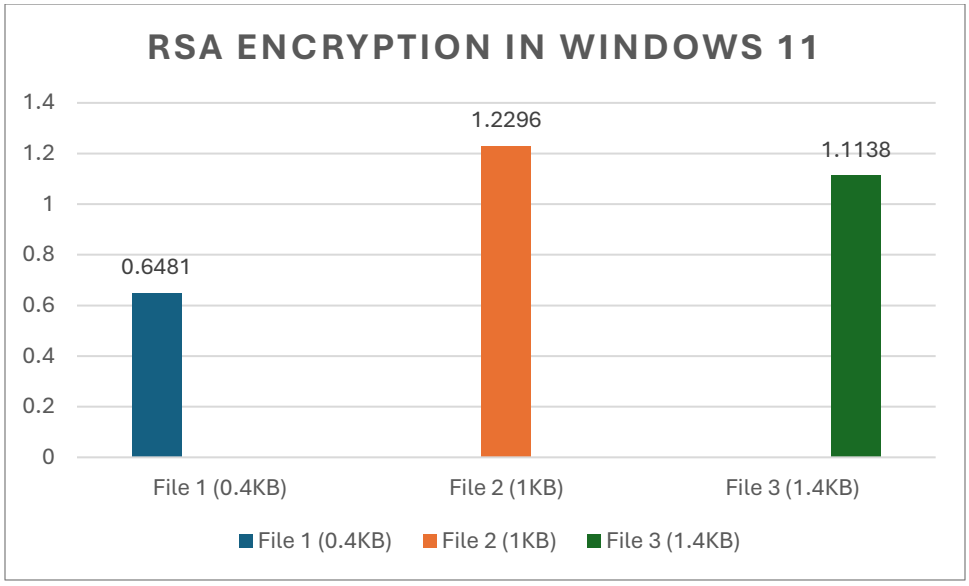
Decrypt			
Key Size	**8912**	**10240**	**12288**
File 1 (0.4KB)	87.0504		
File 2 (1KB)		136.235	
File 3 (1.4KB)			163.821

2.2.2. Biểu đồ cột so sánh các mode:

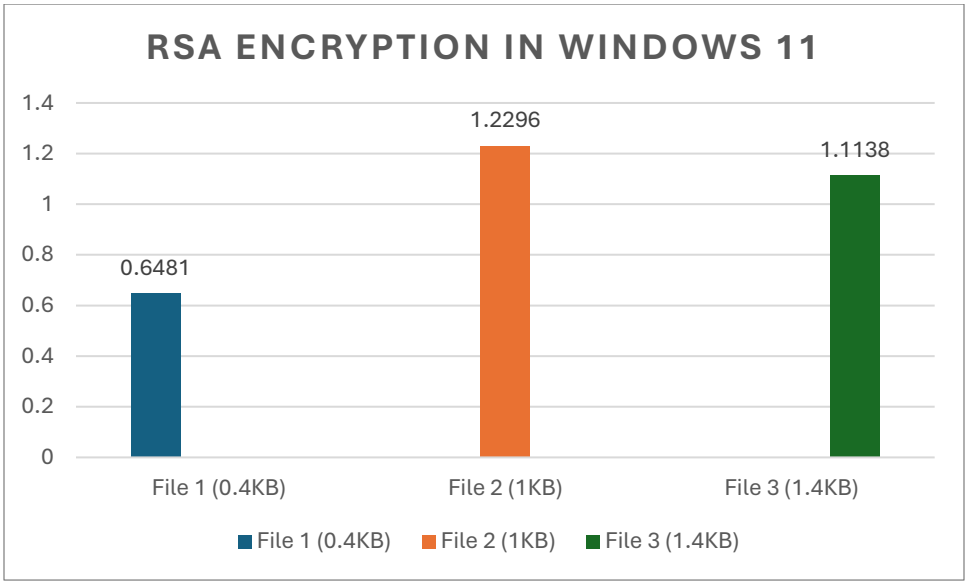
Dựa theo số liệu từ các bảng ở trên, vẽ biểu đồ để dễ dàng so sánh một cách trực quang:

2.2.2.1. Hệ điều hành Windows 11:

- **Biểu đồ thời gian trung bình 10 000 lần Encrypt:**

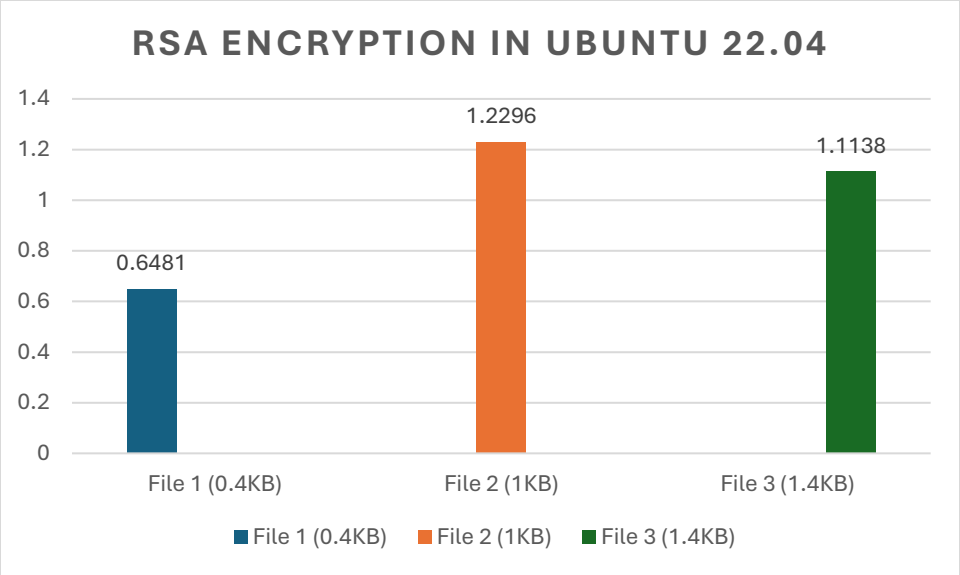


- **Biểu đồ thời gian trung bình 10 000 lần Decrypt:**

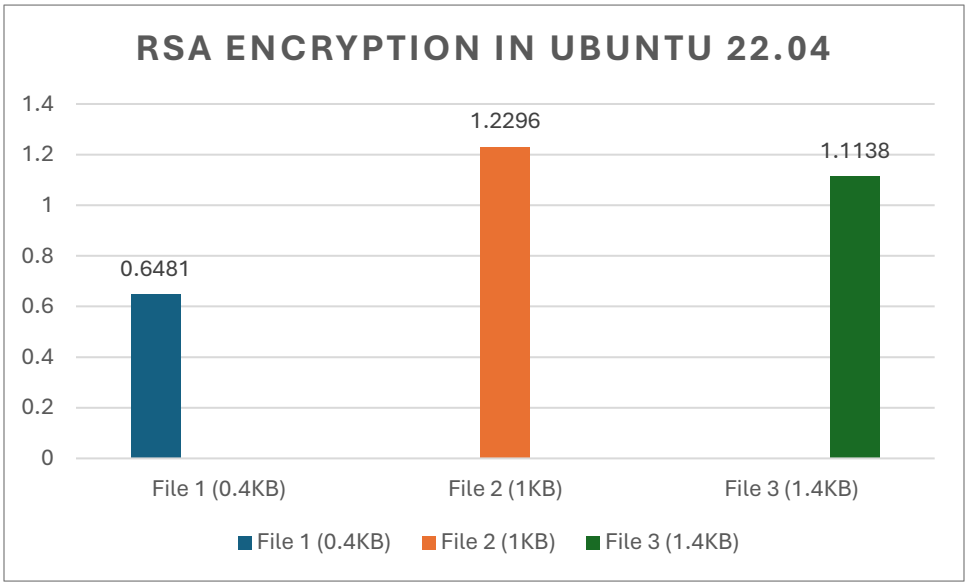


2.2.2.1. Hệ điều hành Ubuntu 22.04 (Linux):

- **Biểu đồ thời gian trung bình 10 000 lần Encrypt:**

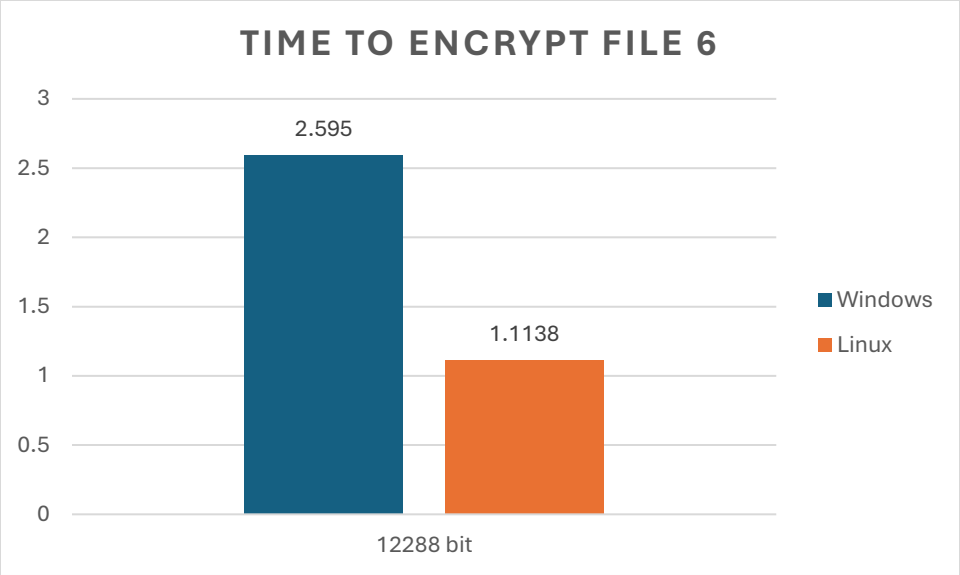


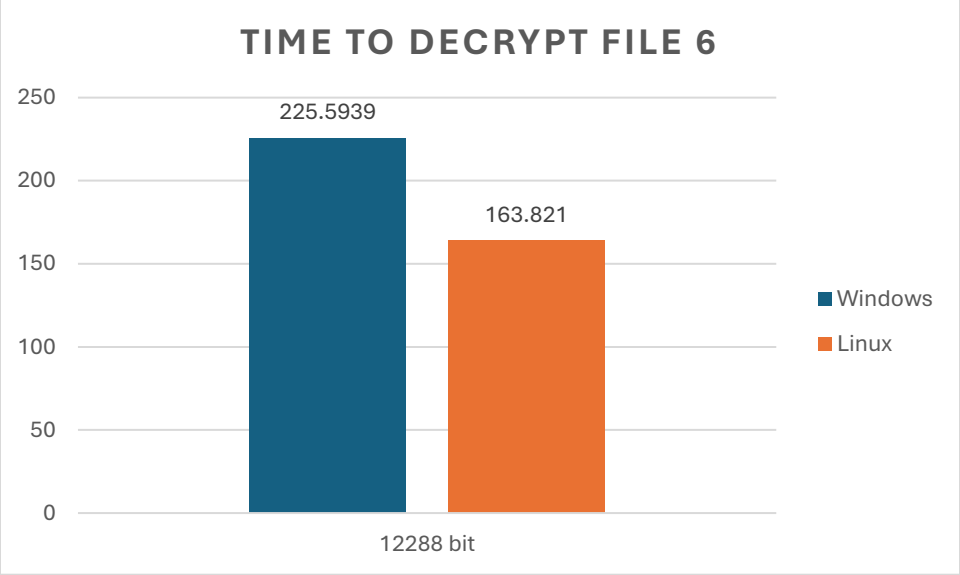
- **Biểu đồ thời gian trung bình 10 000 lần Decrypt:**



2.2.3. Biểu đồ cột so sánh hai hệ điều hành:

Dưới đây là biểu đồ so sánh thời gian mã hoá và giải mã file 5.2 MB trên 2 hệ điều hành:





2.2.4. Phân tích và so sánh:

- Số liệu cho ta thấy việc giải mã RSA tốn thời gian nhiều hơn mã hóa RSA, điều này là do khóa riêng tư lớn hơn rất nhiều so với khóa công khai. Việc giải mã yêu cầu thực hiện nhiều phép toán phức tạp hơn, do khóa công khai thường là một số nhỏ trong khi khóa bí mật là một số lớn. Quá trình mã hóa chỉ cần tính lũy thừa mô-đun với số mũ nhỏ, trong khi giải mã phải tính với số mũ lớn, làm cho các phép toán trở nên phức tạp và tốn thời gian hơn.

- Kết quả này còn cho thấy Linux có hiệu suất mã hóa tốt hơn so với Windows. Việc lựa chọn hệ điều hành và chế độ mã hóa phù hợp là yếu tố quan trọng để đạt được hiệu suất tối ưu trong các ứng dụng đòi hỏi mã hóa nhanh chóng.

2.3. Tổng kết:

Sau bài lab này, em đã học được cách sử dụng thư viện CryptoPP, biết cách triển khai mã hóa và giải mã RSA bằng thư viện này, và có cái nhìn tổng quát về thời gian thực thi của từng loại. Em cũng nhận thấy sự khác biệt đáng kể về hiệu suất khi thực thi trên Linux và Windows. Việc hiểu rõ và lựa chọn hệ điều hành, cùng với chế độ mã hóa phù hợp, là rất quan trọng để đạt được hiệu suất tối ưu trong các ứng dụng yêu cầu mã hóa nhanh chóng.