Hanoi University of Science and Technology

School of Information and Communications Technology

——————— o0o ———————

Project 2

Hệ thống file lưu trữ an toàn (mức độ sơ khai)

Supervisor:
Tran Hai Anh

Group of students:

Full Name	**Student ID**
Tran Minh Tuan	20214978

June 20, 2024

Contents

Abstract

Mục tiêu của báo cáo này là giới thiệu một hệ thống lưu trữ ảnh chia sẻ an toàn, giúp người dùng chia sẻ và truy cập ảnh một cách an toàn và hiệu quả. Hệ thống này được thiết kế để đáp ứng nhu cầu chia sẻ ảnh của người dùng, đồng thời đảm bảo an toàn cho dữ liệu ảnh.

Hệ thống lưu trữ ảnh chia sẻ sử dụng các công nghệ mã hóa và xác thực để bảo vệ dữ liệu ảnh khỏi các mối đe dọa. Hệ thống cũng được thiết kế để đảm bảo tính khả dụng và độ tin cậy, giúp người dùng truy cập ảnh một cách nhanh chóng và hiệu quả.

Các điểm nổi bật của hệ thống:

- Mã hóa ảnh: Hệ thống sử dụng các thuật toán mã hóa để bảo vệ dữ liệu ảnh khỏi các mối đe dọa.
- Xác thực người dùng: Hệ thống sử dụng các phương pháp xác thực người dùng để đảm bảo rằng chỉ người dùng được phép truy cập vào ảnh.
- Tính khả dụng: Hệ thống được thiết kế để đảm bảo tính khả dụng và độ tin cậy, giúp người dùng truy cập ảnh một cách nhanh chóng và hiệu quả.
- Quản lý quyền truy cập: Hệ thống cho phép người dùng quản lý quyền truy cập vào ảnh, bao gồm quyền đọc, quyền ghi và quyền xóa.

Báo cáo này sẽ giới thiệu chi tiết về hệ thống lưu trữ ảnh chia sẻ, bao gồm các công nghệ và kỹ thuật được sử dụng để đảm bảo an toàn và hiệu quả. Mục tiêu của báo cáo là cung cấp một hệ thống lưu trữ ảnh chia sẻ an toàn và hiệu quả cho người dùng.

1 Thông tin kỹ thuật

Project được thực hiện trên ngôn ngữ Python, và sử dụng một số thư viện sau đây:

- Flask Framework và Flask RESTful: xây dựng RESTful API với Flask webframework
- Thư viện AES nằm trong mã nguồn mở của PyCryptodome
- Thư viện secrets: tạo ngẫu nhiên các chuỗi với độ dài xác định
- Thư viện hashlib: hỗ trợ các thuật toán SHA256, MD5,...
- Thư viện PIL, OpenCV: chuyển đổi định dạng ảnh và thao tác với ảnh Các yêu cầu kĩ thuật để cài đặt và chạy thử sẽ được miêu tả rõ hơn ở phần 5

2 Giới thiệu về Protocol

2.1 Đăng ký

Quá trình đăng ký được miêu tả như sau:

1. User nhập tên của mình vào client
2. Client tiến hành tạo ngẫu nhiên cặp khóa public key, private key
3. User chọn cặp khóa tùy ý, rồi submit lên server để đăng ký

Ưu điểm của cách làm này là đảm bảo được user là người sở hữu khóa public key. Sau đó user sẽ tiến hành ghi nhớ ID và khóa d, đây sẽ là lần lượt là ID đăng nhập và mật khẩu đăng nhập cho tài khoản

2.2 Đăng nhập

Quá trình đăng nhập được miêu tả như sau:

1. User tiến hành nhập ID và mật khẩu private key
2. Client sẽ gửi yêu cầu đăng nhập tới server, kèm theo ID của user.
3. Server sẽ tạo một token ngẫu nhiên và lưu một bản sao trong database, sau đó encrypt bằng public key của user và gửi trả token đã được encrypt về client.
4. Client sẽ nhận token dùng khóa private key để giải mã token bằng thuật toán RSA

5. Nếu token gửi lên giống với token trong database, server sẽ cấp cho user một API token sử dụng trong suốt session của mình.

Cần lưu ý là khóa bí mật private key không bao giờ được gửi đi khỏi client. Như vậy, giả sử người dùng có bị lộ khóa công khai public key thì cũng không ảnh hưởng đến quá trình đăng nhập; ngược lại, giả sử khóa public key bị kẻ tấn công thay thế từ pha đăng ký, khi đó tài khoản sẽ không thể đăng nhập từ lần đầu tiên - hạn chế khả năng bị đánh tráo khóa công khai

2.3 Upload, Download và Chia sẻ Ảnh

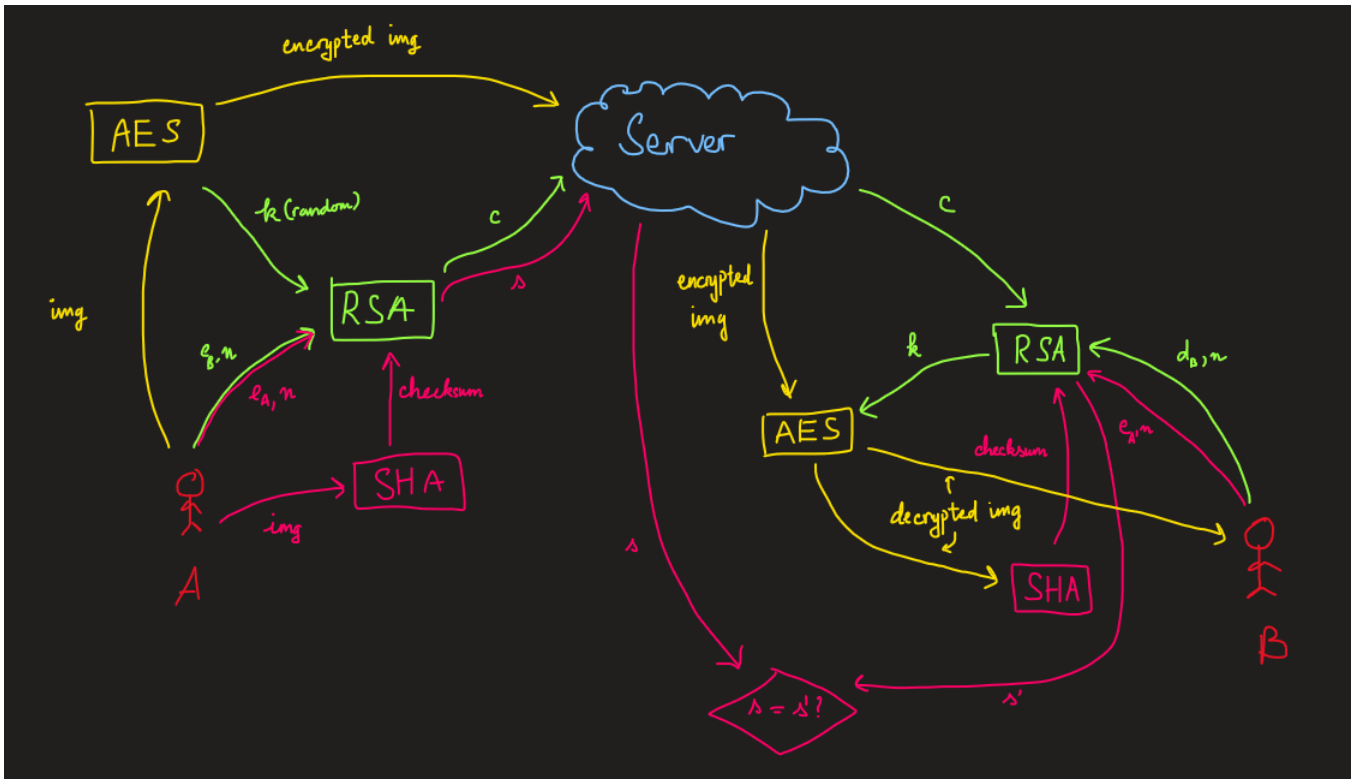

Figure 1: Ảnh minh họa các giao thức upload ảnh và download ảnh

2.3.1 Upload ảnh

Giao thức upload ảnh được mô tả như sau: Giả sử A là người upload ảnh có file đuôi là *.png, A có mã khóa công khai public key A

1. Client sinh khóa ngẫu nhiên k
2. Client dùng khóa k để mã hóa ảnh a.png bằng thuật toán AES
3. Client dùng khóa công khai A để mã hóa k thành C
4. Client gửi ảnh mã hóa và c lên server

Trong giao thức này có một bước nữa: tạo checksum để kiểm tra tính toàn vẹn của ảnh:

1. Client hash ảnh bằng thuật toán SHA256 thành một chuỗi checksum h

2. Client tiến hành mã hóa h bằng thuật toán RSA với khóa công khai public key, tạo chữ ký s

3. Client gửi chữ ký s lên server

2.3.2 Download ảnh

Giao thức download ảnh được miêu tả như sau: giả sử A có quyền download ảnh (được chia sẻ ảnh) và có khóa bí mật private key A; B là người upload ảnh có khóa công khai là public key B. Ảnh được upload lên server có chữ ký s.

1. client download ảnh đã được mã hóa từ server.

2. Client yêu cầu khóa c từ server

3. Client dùng khóa bí mật private key A để giải mã c trở thành k bằng thuật toán RSA

4. Client dùng k để giải mã ảnh đã được mã hóa

Giao thức này còn một bước, là kiểm tra checksum của ảnh:

1. Client băm ảnh vừa được giải mã bên trên bằng thuật toán SHA256, tạo chuỗi checksum h

2. Client dùng khóa công khai public key B của B mã hóa h thành chữ ký s', sử dụng thuật toán RSA

3. Client sau đó so sánh s' với s, nếu 2 chuỗi này trùng khớp thì ảnh được download nguyên vẹn, giống với ảnh gốc mà B đã upload

2.3.3 Chia sẻ ảnh

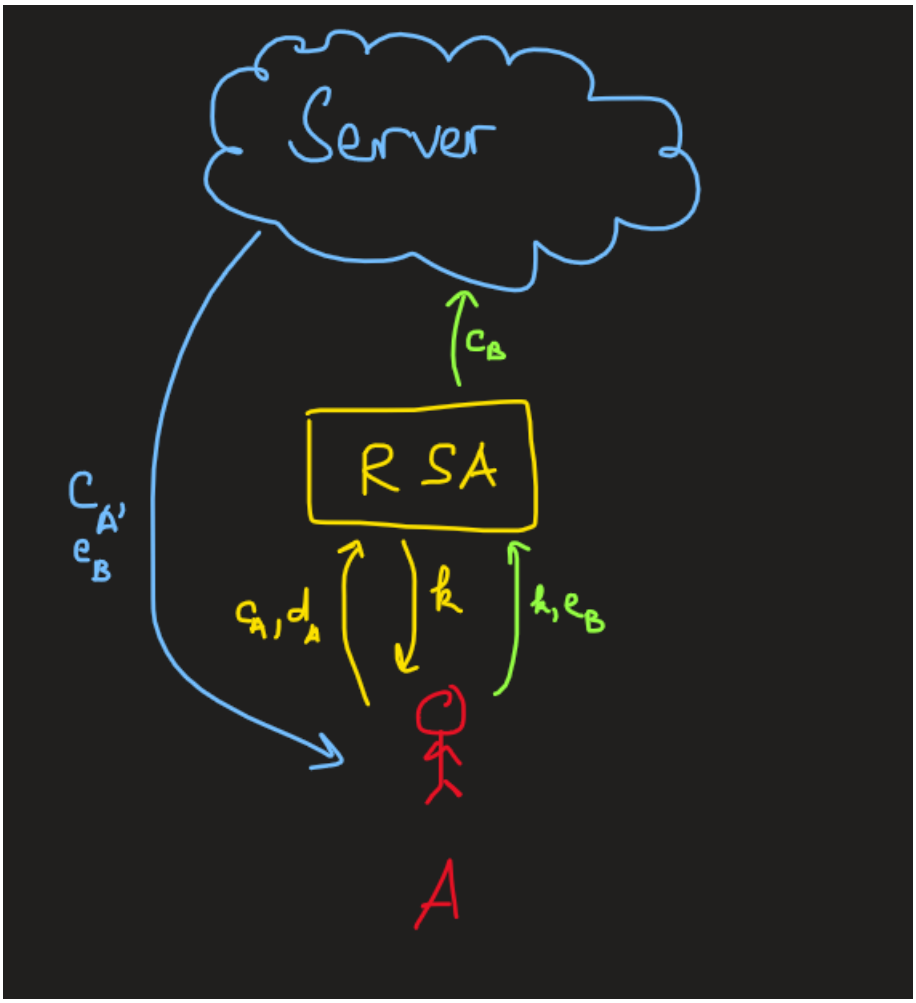

Figure 2: Ảnh minh họa giao thức chia sẻ ảnh

- Giao thức chia sẻ ảnh được miêu tả như sau: giả sử A là chủ sở hữu của ảnh muốn chia sẻ ảnh x.png được mã hóa bởi A và khóa công khai của B .
- A tiến hành giải mã khóa C_A thành k bằng khóa bí mật d_A mình đang giữ, sử dụng thuật toán ECIES
- A mã khóa k thành c_B bằng khóa công khai e_B của B, sau đó gửi c_B trở lại server

Khi đó B chỉ cần yêu cầu download ảnh từ server

3 API

Tài liệu API xem ở link này:
https://documenter.getpostman.com/view/34877729/2sA3JNazzm

4 Vận dụng

4.1 Các thuật toán sử dụng

4.1.1 RSA

Nhóm cài đặt lại RSA protocol theo nội dung được giảng dạy trên lớp.:

1. Chọn p, q là 2 số nguyên tố lớn.
2. Tính $n = pq, \phi = (p-1)(q-1)$
3. Chọn d, e sao cho $e \equiv d^{-1} \pmod{\phi}$
4. Bản mã $c \equiv m^e \pmod{n}$
5. Bản rõ $m \equiv c^d \pmod{n}$

Các phép nhân, phép mũ sử dụng thuật toán nhân nhanh và mũ nhanh theo modulo.

4.1.2 Thuật toán Euclide mở rộng

Dựa trên bổ đề Bézout: với hai số nguyên không âm a và b, $g = (a, b)$ thì:

- Tồn tại hai số nguyên x, y sao cho $ax + by = g$
- g là số nguyên nhỏ nhất có thể viết dưới dạng $ax + by$
- Mỗi số e có dạng $ax + by$ đều là bội của d.

Thuật toán nhận vào a, b, đưa ra $x, y; g = (a, b)$ sao cho $ax + by = g$

Algorithm 1 Thuật toán Euclide mở rộng (Extended Euclidean Algorithm)

function XEUCLIDEAN(a, b)
  if $a = 0$ **then return** (b, 0, 1)
  else
    $(g, y, x) \leftarrow XEuclidean(b\%a, a)$
    $x \leftarrow x - \left\lfloor \frac{b}{a} \right\rfloor \times y$
  return (g, x, y)
  end if
end function

4.1.3 Tìm modulo nghịch đảo

Vẫn dựa trên bổ để Bézout bên trên: giả sử ta có $ax + by = g \iff ax \equiv g \pmod{y}$. Khi đó x là modulo nghịch đảo của a khi $g = 1$.

Algorithm 2 Thuật toán tìm modulo nghịch đảo

```
function INVERSE_MODULO(a, n)
    (g, x) ← XEuclidean(a, n)
    if g ≠ 1 then return 'DnE'
    else
      return x % n
    end if
end function
```

4.1.4 Thuật toán sinh khóa RSA ngẫu nhiên

Sinh khóa RSA sử dụng 2 thuật toán chính:

- Sinh số nguyên tố lớn (sử dụng để tạo p, q)
- Kiểm tra số nguyên tố

Để sinh d, e ta chỉ cần sinh một số ngẫu nhiên trong $(1, \phi]$ và tìm khả nghịch của số đó trong ϕ.

4.2 AES - Mã hóa ảnh và giải mã file ảnh

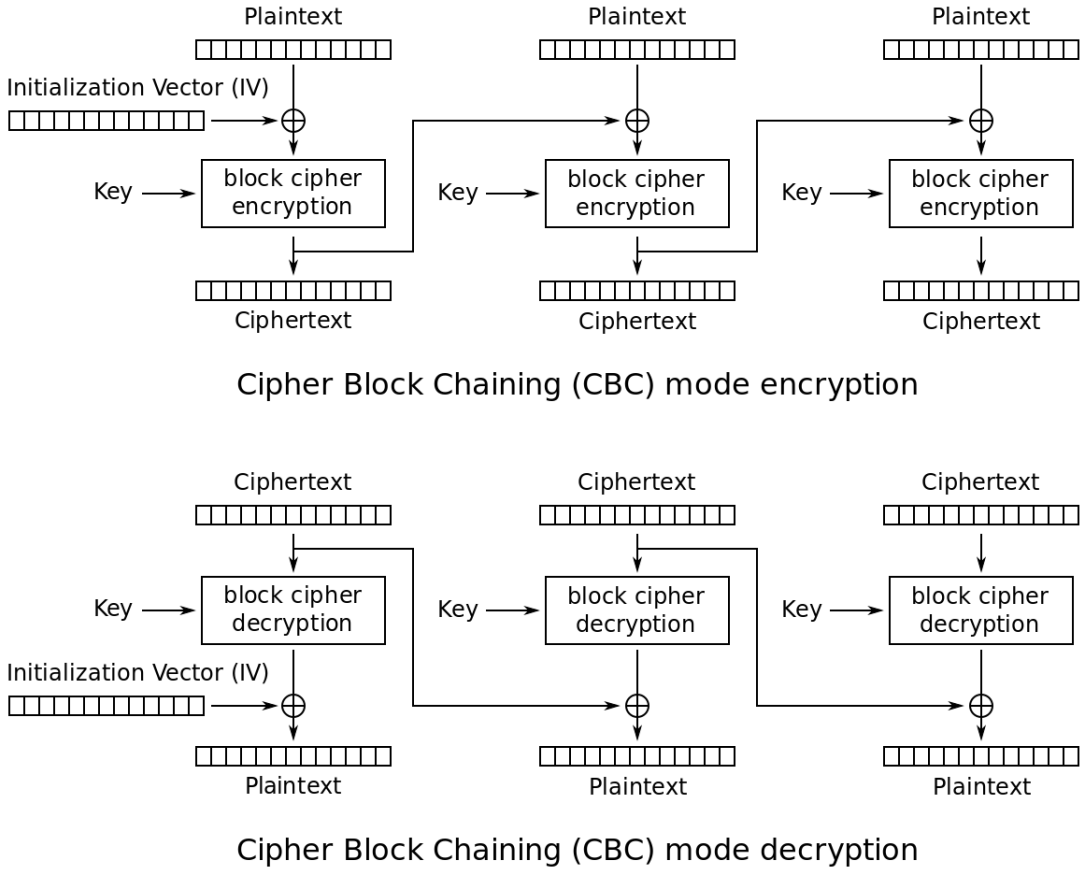

Figure 3: Tóm tắt mode CBC.[1] Thuật toán block encryption sử dụng ở đây là AES.

Sử dụng thuật toán AES trong thư viện PyCryptodome với mode CBC. Với cách làm này ảnh vẫn giữ nguyên format nhưng nội dung được làm nhiễu. Quá trình mã hóa như sau:

- Vì ta muốn mã hóa nhiều format ảnh khác nhau, nhóm ưu tiên chuyển các ảnh về định dạng PNG trong thư viện PIL. Việc chuyển format về dạng PNG sẽ giúp đảm bảo chất lượng của ảnh
- Khởi tạo khóa k là một chuỗi ngẫu nhiên có độ dài là 16 bytes. Khởi tạo block IV = 0000000000000000 cũng có cùng độ dài
- Tiến hành thêm padding cho file để file có kích thước là bội của 16. Byte cuối cùng của phần padding sẽ lưu số lượng dòng padding đã thêm. Chú ý là nếu file đã có kích thước là bội của 16, ta vẫn tiến hành thêm 16 bytes. Lúc này byte cuối cùng sẽ lưu số 16.

- Tiến hành mã hóa với AES và lưu vào file có định dạng PNG.

Với quá trình giải mã:

- Từ file và key ban đầu, dùng thuật toán AES để giải mã và cho vào một ma trận
- Phần tử cuối của ma trận lưu số dòng padding. Xóa các dòng padding đi
- Lưu ma trận vào file với định dạng gốc, sử dụng OpenCV

Cách làm này được tham khảo từ câu trả lời trên StackOverFlow của tác giả Rotem

5 Cài đặt & chạy thử

5.1 Server

Sử dụng Terminal (Linux) hoặc Command Line (Window)

1. Tạo virtual environment cho server:

   Browse đến thư mục `server`, sau đó:

   - Đối với Linux:

     ```
     python3 –m venv venv
     ```

   - Đối với Windows:

     ```
     python –m venv venv
     ```

2. Kích hoạt virtual environment

   - Đối với Linux:

     ```
     . venv/bin/activate
     ```

   - Đối với Windows:

     ```
     venv\Scripts\activate
     ```

3. Cài các package cần thiết:

   ```
   pip install –r requirements.txt
   ```

4. Tạo mới database

- Đối với Linux:

```
export FLASK_APP=serverside
export FLASK_ENV=development
flask init-db
```

- Đối với Windows:

  Trường hợp dùng Commmand Prompt

```
set FLASK_APP=serverside
set FLASK_ENV=development
flask init-db
```

  Trường hợp dùng Powershell

```
$env:FLASK_APP="serverside"
$env:FLASK_ENV="development"
flask init-db
```

5. Chạy server

```
flask run
```

Notes: Nếu như bạn gặp lỗi *An attempt was made to access a socket in a way forbidden by its access permissions* thì chúng ta vào link này để giải quyết:

5.2 Client

1. Trong folder client, chạy câu lệnh sau để cài các package cần thiết:
pip install -r requirements.txt
2. Chạy file main.py để khởi động client
python main.py

5.3 Hướng dẫn sử dụng

Sau khi khởi động cả server và client, nhập IP của server và connection port để kết nối tới server. Vì đây là một console application, chủ yếu người dùng sẽ nhập từ bàn phím để chọn các menu item / nhập liệu

Đăng kí

- Người dùng chọn menu Register new accoutn
- Người dùng nhập tên

- Hệ thống sẽ random ra cặp khóa e và d ngẫu nhiên. Nếu đồng ý thì bấm y, ngược lại bấm n sẽ tạo khóa mới
- Nếu chọn khóa xong, người dùng sẽ thấy hiển thị thông báo đăng ký thành công.

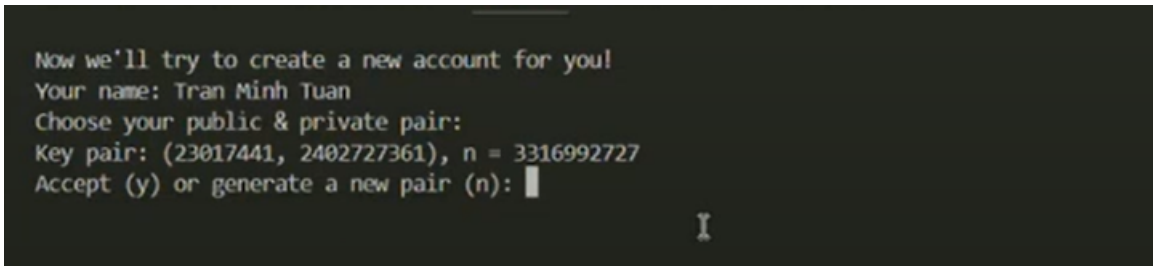

Figure 4: Giao diện đăng ký người dùng

Đăng nhập

- Người dùng chọn menu Login
- Người dùng nhập ID.
- Người dùng nhập khóa bí mật và nhấn Enter.
- Thông báo đăng nhập thành công, người dùng sẽ được chuyển hướng đến menu chính

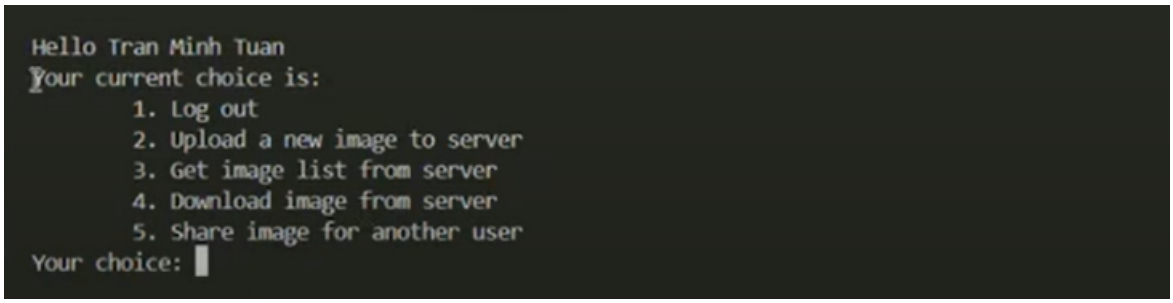

Figure 5: Menu chính của client

Đăng xuất : Trong menu chính, người dùng chọn LogOut

Upload ảnh :

- Trong menu chính, người dùng chọn Upload a new image to server
- Người dùng nhập đường dẫn đến file ảnh. Chú ý: Hệ thống chỉ hỗ trợ các extension .png, .jpg/jpeg, .bmp
- Nhấn enter, người dùng đã upload ảnh thành công

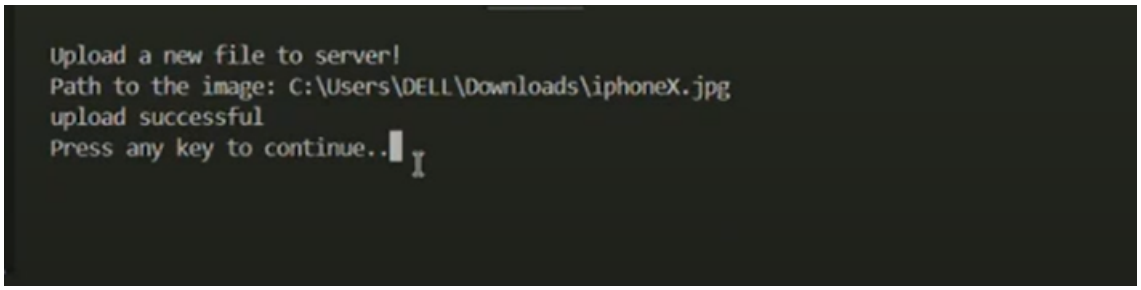

Figure 6: Menu upload

Download ảnh

- Trong menu chính, người dùng chọn Download image from server
- Có 2 lựa chọn: download tất cả hoặc download 1 ảnh
- Nếu download tất cả: Hệ thống sẽ download tất cả những ảnh bạn đã upload / được chia sẻ
- Nếu download 1 ảnh: nhập ID ảnh và download

Figure 7: Menu download

Xem danh sách ảnh

- Trong menu chính, người dùng chọn Get image list from server
- Tất cả các ảnh của user (upload, được share) sẽ được liệt kê ở đây (ID - tên ảnh)

Figure 8: Danh sách ảnh của một User

Chia sẻ ảnh

- Trong menu chính, người dùng chọn Share image for another user
- Người dùng nhập ID người cần share. Chú ý: người dùng phải là chủ (người upload) ảnh mới có thể share ảnh

Figure 9: Chia sẻ ảnh cho một User

Chi tiết hướng dẫn sử dụng, mời bạn xem demo

5.4 Demo

Video demo quá trình cài đặt & chạy thử ở link này: Demo

6 Kết luận

6.1 Ưu điểm

- Hệ thống có thể chia sẻ hình ảnh một cách bảo mật, không làm lộ thông tin cho bên thứ 3.
- File được lưu trữ nguyên vẹn, không bị tổn hại
- Hệ thống đảm bảo chỉ những user có quyền (được chia sẻ) mới có thể xem ảnh; user có quyền (chủ sở hữu) mới có thể chia sẻ ảnh

6.2 Nhược điểm

Vì đây là hệ thống ở mức độ sơ khai và không có hệ thống bảo mật nào là an toàn tuyệt đối, nên còn nhiều khuyết điểm cần chỉ rõ:

- Hệ thống vẫn có thể làm lộ access_token qua quá trình truyền tải nếu không có SSL encryption
- Vì hệ thống chỉ hỗ trợ đăng nhập bằng khóa bí mật d, nên user chưa thay đổi password.
- Hệ thống chưa hỗ trợ tính năng xóa ảnh
- Nếu nhiều user chưa sử dụng cùng 1 máy thì rất dễ lộ Download folder. Hiện tại hệ thống chỉ hỗ trợ 1 user / 1 máy

6.3 Cải tiến

- Có thể thay thế CBC mode bằng một số mode khác phức tạp hơn để đảm bảo an toàn cho ảnh đã encrypt
- Có thể tăng độ dài key RSA và AES để đảm bảo khóa an toàn

References

[1] Wikimedia Commons, the free media repository. *File:Cbc encryption.png*. [Online; accessed January 10, 2022]. 2006. URL: https://commons.wikimedia.org/wiki/File:Cbc_encryption.png.

Appendix: Class diagram

Server

The server component handles various functionalities related to user authentication, image uploading, and cryptographic operations. It includes classes for login, registration, uploading images, utility functions, and implementing cryptographic algorithms like RSA and extended Euclidean algorithm.

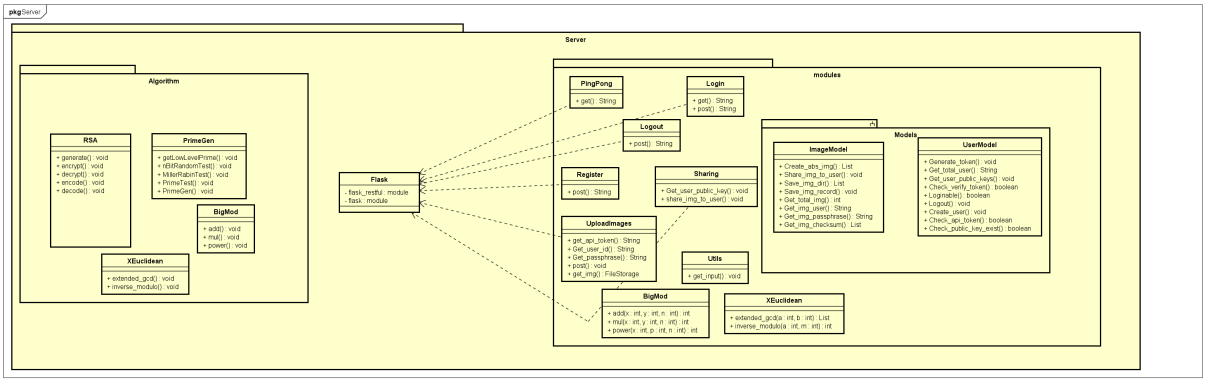

Figure 10: Class Diagram Server

Client

The client component focuses on the user interface and cryptographic operations. It includes classes for generating UI elements like download, upload, and image list views. It also has classes for implementing cryptographic algorithms like AES, RSA, big integer modular arithmetic, and prime number generation algorithms.

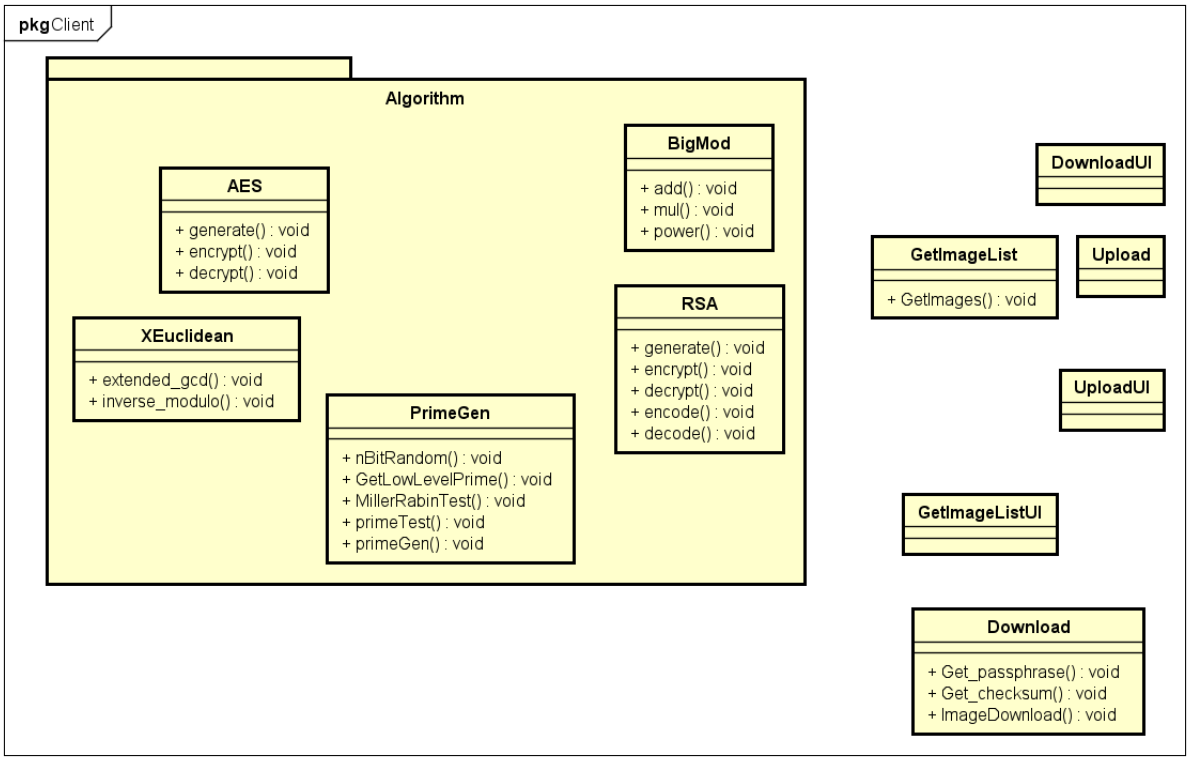

Figure 11: Class Diagram Client