

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC - KỸ THUẬT MÁY TÍNH



Cryptography and Network Security

Assignment 1

TA: Nguyễn Hữu Hiếu
SV: Phạm Hồng Tiến
MSSV: 1713490

TP. HỒ CHÍ MINH, THÁNG 6/2020



Mục lục

1	Giới thiệu	2
1.1	Tổng quát dự án	2
1.2	Thông tin dự án	2
1.3	Giới hạn của dự án	2
1.4	Tính năng mở rộng	3
2	Chi tiết dự án	4
2.1	Tổng quan client	4
2.1.1	Đăng nhập	4
2.1.2	Đăng xuất	6
2.2	Công cụ mã hóa	7
2.2.1	notify box	7
2.2.2	Mã hóa file	8
2.2.3	Công cụ giải mã	11
2.2.4	Công cụ hash	13
2.2.5	Mã hóa và giải mã file khác	14
2.3	Các hàm mã hóa	17
2.3.1	Hàm openssl_encrypt(), openssl_decrypt()	17
2.3.2	base64_encode(), base64_decode()	17
2.3.3	md5()	18
2.4	Gọi hàm	18
3	Phân tích và kết luận	19
3.1	Kết quả đạt được	19
3.2	Đánh giá	19
4	Hướng phát triển	19

1 Giới thiệu

1.1 Tổng quát dự án

Đây là một trang web với tính năng upload file. Với những file được upload lên đây có thể được mã hóa với chuẩn AES (Advanced Encryption Standard) với cơ chế sau đây:

Input đầu vào là một file bất kì (.txt,.jpg,.docx,.zip,.png,.mp3,...). File input sẽ được hệ thống dùng công cụ đọc dưới dạng một chuỗi kèm theo một file txt chứa key(khóa). Sau đó sẽ được mã hóa bằng thuật toán AES tạo thành một file được mã hóa. File mã hóa sẽ được web lưu trữ trên server. Đồng thời trước khi file được mã hóa, web sẽ tạo một bản sao file gốc và lưu vào một thư mục lưu trữ khác, để sau này có thể so sánh tính toàn vẹn của việc mã hóa bằng cách dùng hàm hash. Ngoài ra, web còn có công cụ để giải mã một file đã được mã hóa và hash file để kiểm tra tính toàn vẹn dữ liệu. Cơ chế giải mã cũng tương tự như cơ chế mã hóa. Ta sẽ lấy một file đã được mã hóa trên server cùng với một key của file đã được mã hóa đó, dùng thuật toán AES để tìm ngược lại file gốc.

Trang web còn hỗ trợ tính năng đăng nhập, hiển thị trạng thái upload, hash,... trong thời gian thực. Với tính năng này nhiều người có thể tham gia vào server một lúc, điều này cũng giúp cho nhiều người khác có thể theo dõi trạng thái hoạt động của bạn.

1.2 Thông tin dự án

1.2.1 Application

- Nền tảng: WEB
- Ngôn ngữ: HTML,CSS,PHP,JavaScript,...
- IDE: Visual studio code

1.2.2 Database

- Sử dụng máy chủ ảo localhost -Công cụ xampp

1.2.3 Mã hóa

- Giải thuật: Advanced Encryption Standard (AES)
- Thư viện: OpenSSL

1.2.4 Đối tượng mã hóa

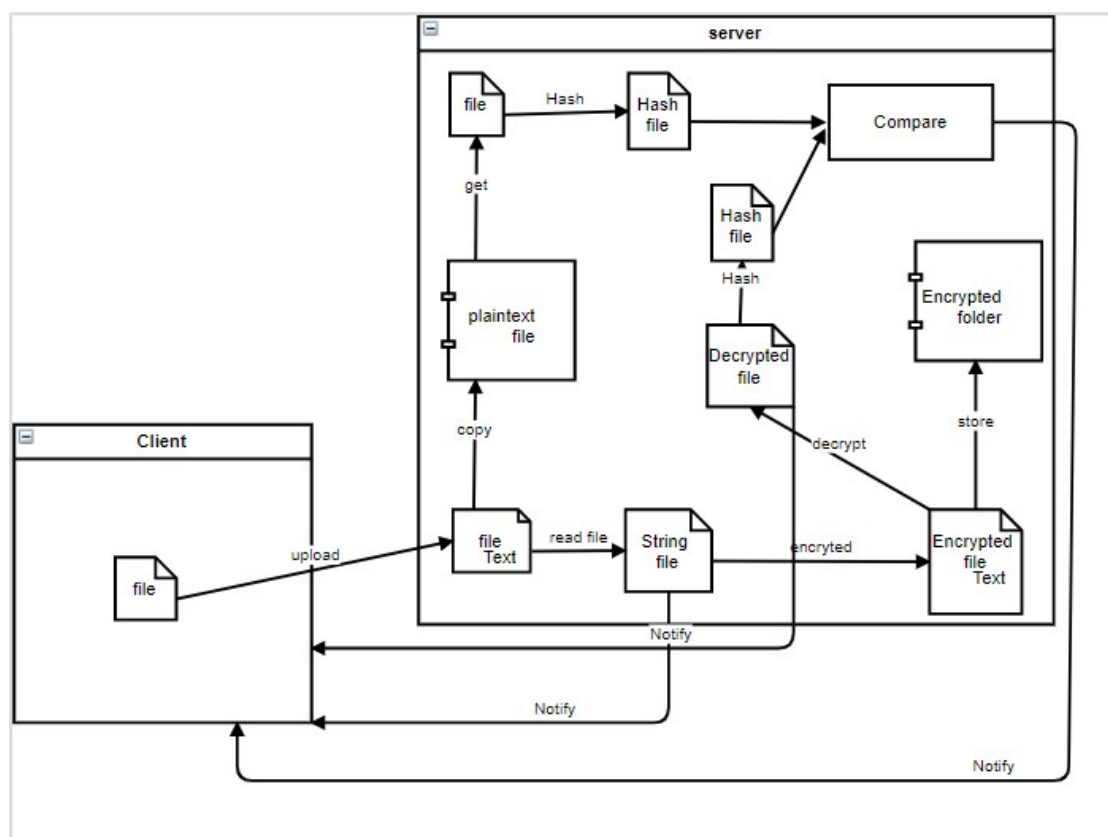
- Đa số các loại file thông dụng hiện tại(.txt,.jpg,.png,.mp3,...)

1.2.5 Kiểm tra tính toàn vẹn

- Chuẩn: MD5
- Hàm: md5(file) (với file là file được đọc dưới dạng một chuỗi)

1.3 Giới hạn của dự án

- Chỉ dùng thư viện để mã hóa. Dễ dàng bị tấn công vì phương thức mã hóa chưa đủ phức tạp, cộng với việc file key được upload lên nên dễ lộ.
- Web chưa thật sự thuận tiện khi dùng, chỉ áp dụng cho việc mã hóa một file một lần.



Hình 1: Cơ chế mã hoá tổng quát

-Thời gian web chạy chưa thật sự tối ưu.

1.4 Tính năng mở rộng

-Có chức năng đăng nhập, đăng xuất. Nhờ đó có thể hiển thị những file upload được upload lên bởi người dùng nào.

-Ngoài ra còn có tính năng thông báo thời gian thực của việc upload, mã hóa, đăng xuất, nhờ đó ta có thể theo dõi trạng thái của quá trình mã hóa. -Thuật toán tạo một khóa ngẫu nhiên mỗi lần mã hóa nên tạo nên tính ngẫu nhiên khó đoán, khó giải mã.

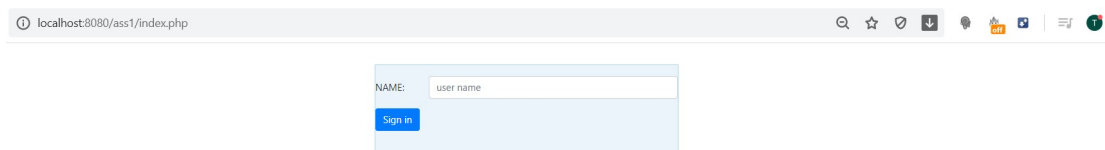


2 Chi tiết dự án

2.1 Tổng quan client

2.1.1 Đăng nhập

Khi bắt đầu vào trang web, ta sẽ phải đăng nhập bằng cách nhập username. Nhấn vào 'Sign in' sẽ đưa ta đến trang giao làm việc chính của web. (Hình 2,3)



Hình 2: Giao diện đăng nhập



localhost:8080/ass1/index.php

Welcome: **Phạm Hồng Tiến** [LOGOUT](#)

PLAINTEXT FILE Choose file

KEY FILE Choose file

ENCRYPTED FILE Choose file

KEY FILE Choose file

-----HASH CHECKING-----

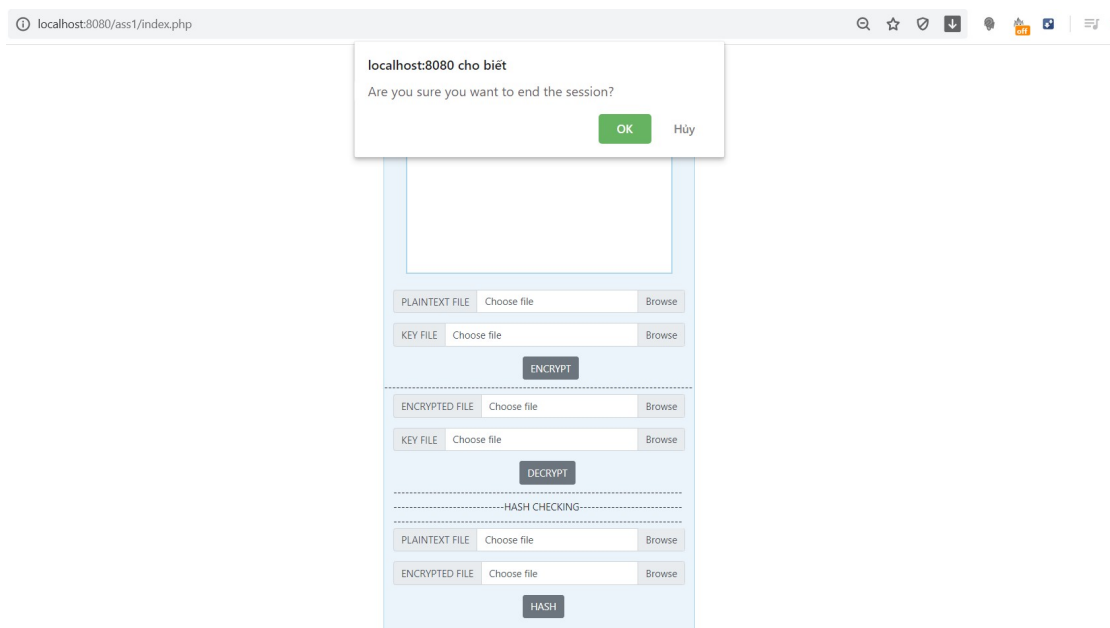
PLAINTEXT FILE Choose file

ENCRYPTED FILE Choose file

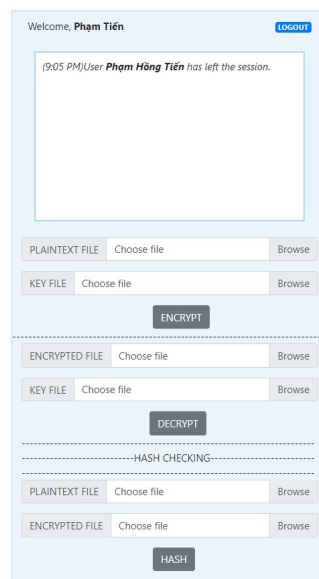
Hình 3: Giao diện làm việc chính

2.1.2 Đăng xuất

Sau khi đăng nhập ta có thể đăng xuất bằng việc nhấn vào nút 'logout' bên góc trên phải. Sau khi xác nhận, ta sẽ được kết thúc phiên làm việc và đưa về màn hình đăng nhập. Đồng thời, notifybox cũng hiện thị thông báo.



Hình 4: Xác nhận để kết thúc phiên làm việc



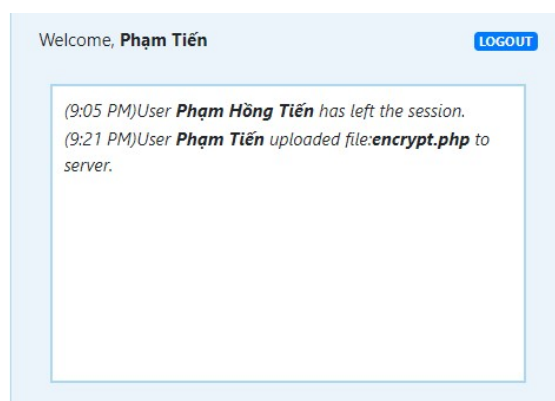
The screenshot shows a web application interface. At the top, it says "Welcome, **Phạm Tiến**" with a "LOGOUT" button. Below this is a notification box containing the message: "(9:05 PM) User **Phạm Hồng Tiến** has left the session." Below the notification box are three sections for file uploads. The first section has "PLAINTEXT FILE" and "KEY FILE" labels, each with a "Choose file" button and a "Browse" button, followed by an "ENCRYPT" button. The second section has "ENCRYPTED FILE" and "KEY FILE" labels, each with a "Choose file" button and a "Browse" button, followed by a "DECRYPT" button. The third section is labeled "HASH CHECKING" and has "PLAINTEXT FILE" and "ENCRYPTED FILE" labels, each with a "Choose file" button and a "Browse" button, followed by a "HASH" button.

Hình 5: notifybox thông báo về trạng thái đăng xuất của bạn

2.2 Công cụ mã hóa

2.2.1 notify box

Hộp thoại này có tác dụng thái đăng nhập của các user. Hộp thoại này còn hiện thời gian thực các hoạt động của user.

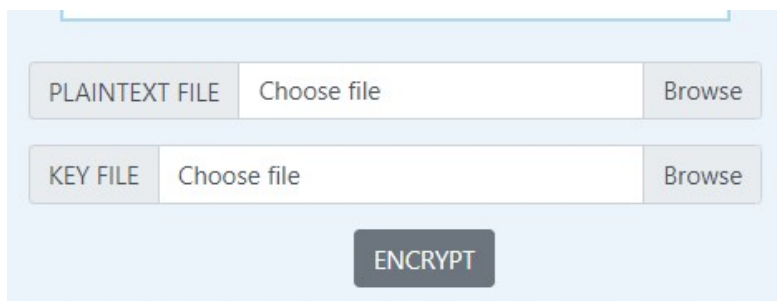


The screenshot shows a web application interface. At the top, it says "Welcome, **Phạm Tiến**" with a "LOGOUT" button. Below this is a notification box containing two messages: "(9:05 PM) User **Phạm Hồng Tiến** has left the session." and "(9:21 PM) User **Phạm Tiến** uploaded file: **encrypt.php** to server."

Hình 6: Hộp thoại

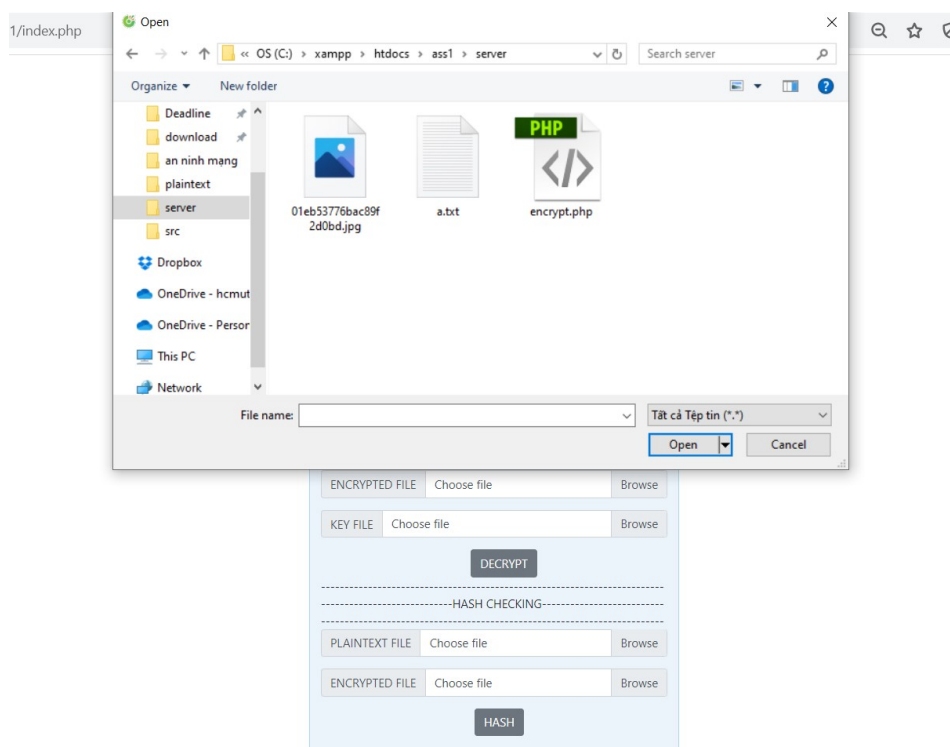
2.2.2 Mã hóa file

Công cụ này giúp ta upload file cần mã hóa kèm với một file txt chứa key được dùng để mã hóa.



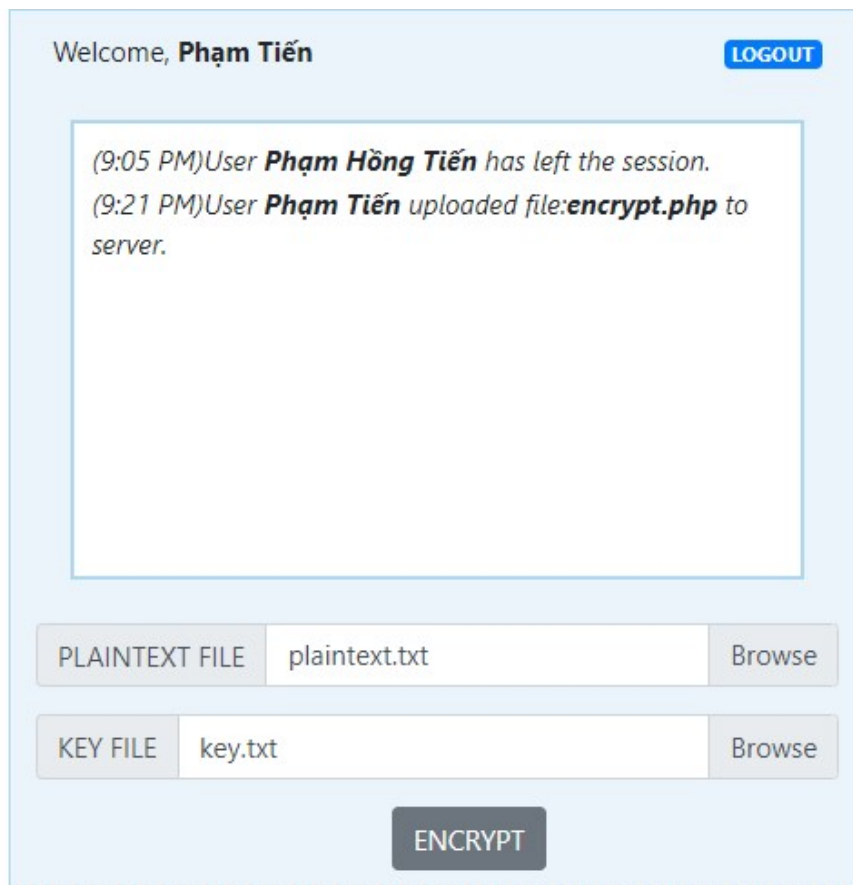
Hình 7: Công cụ mã hóa file

Ta chọn một file cần mã hóa (file nào cũng được, và chọn tiếp file txt chứa key ta cần mã hóa).



Hình 8: Chọn file cần mã hóa(Chọn bất kì file nào cũng được) và file txt chứa key

Sau đó ta nhấn 'ENCRYPT' để tiến hành mã hóa.



Welcome, **Phạm Tiến** [LOGOUT](#)

(9:05 PM) User **Phạm Hồng Tiến** has left the session.
(9:21 PM) User **Phạm Tiến** uploaded file: **encrypt.php** to server.

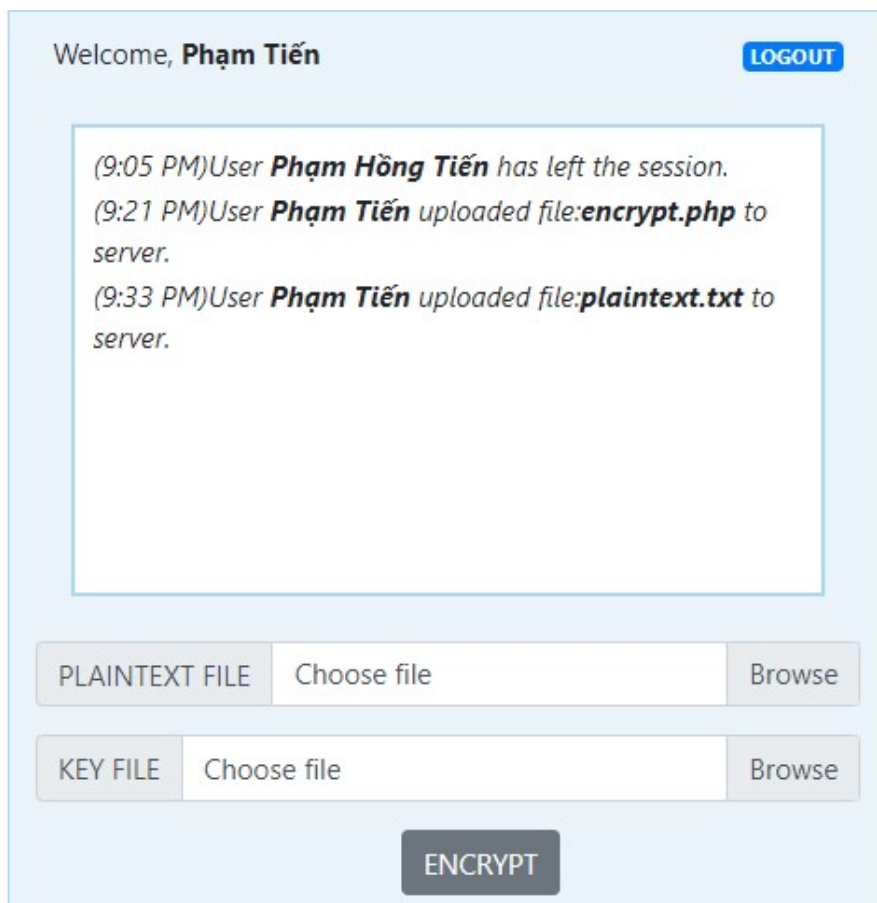
PLAINTEXT FILE [Browse](#)

KEY FILE [Browse](#)

[ENCRYPT](#)

Hình 9: Nhấn 'ENCRYPT' để mã hóa file

Hộp thoại sẽ thông báo cho ta biết khi đã mã hóa thành công.



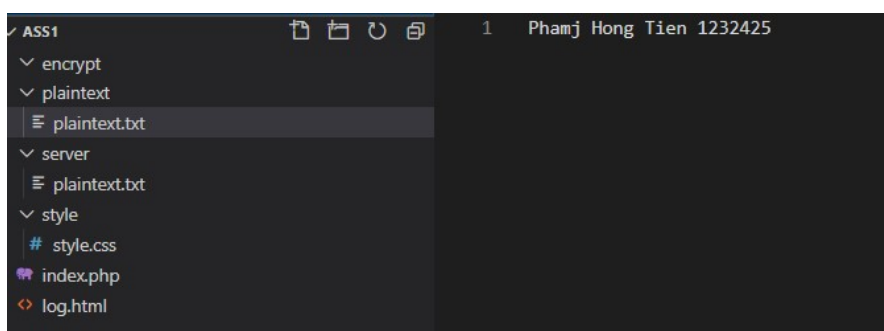
The screenshot shows a web application interface with a light blue background. At the top left, it says "Welcome, **Phạm Tiến**". At the top right, there is a blue button labeled "LOGOUT". Below the welcome message, there is a white box containing a session log with three entries: "(9:05 PM) User **Phạm Hồng Tiến** has left the session.", "(9:21 PM) User **Phạm Tiến** uploaded file: **encrypt.php** to server.", and "(9:33 PM) User **Phạm Tiến** uploaded file: **plaintext.txt** to server.". Below the log box, there are two file upload sections. The first section is labeled "PLAINTEXT FILE" and contains a text input field with "Choose file" and a "Browse" button. The second section is labeled "KEY FILE" and also contains a text input field with "Choose file" and a "Browse" button. At the bottom center, there is a dark blue button labeled "ENCRYPT".

Hình 10: Sau khi mã hóa thành công hộp thoại sẽ thông báo cho ta biết

Sau khi mã hóa thành công, file sẽ được lưu vào trên server.
Do đây trang web này chỉ sử dụng máy chủ ảo localhost, nên có thể coi thư mục server như mục nơi lưu trữ trên server.



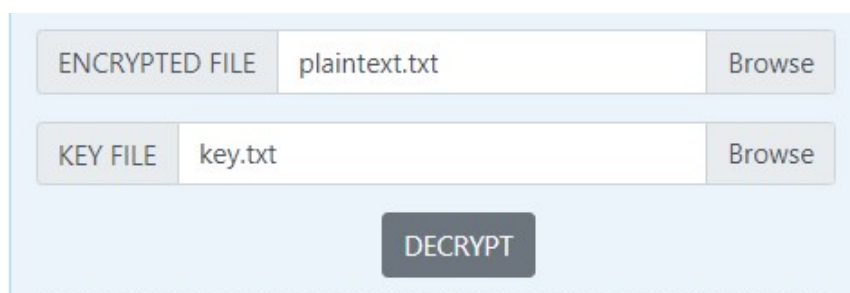
Hình 11: File plaintext.txt đã được mã hóa và lưu vào thư mục server kèm với nội dung bên cạnh



Hình 12: Web tạo một bản sao chứa nội dung ban đầu của file mã hóa và lưu vào thư mục plaintext

2.2.3 Công cụ giải mã

Tương tự đối với công cụ giải mã. Ta chọn một file đã được mã hóa, kèm với file txt chứa key mà nó đã mã hóa file đó.



Hình 13: Chọn file cần giải mã kèm key

Nhấn 'DECRYPT' để giải mã

Welcome, **Phạm Tiến**

LOGOUT

(9:05 PM)User **Phạm Hồng Tiến** has left the session.

(9:21 PM)User **Phạm Tiến** uploaded file:**encrypt.php** to server.

(9:33 PM)User **Phạm Tiến** uploaded file:**plaintext.txt** to server.

(9:50 PM)User **Phạm Tiến** uploaded file:**plaintext.txt** to encrypt.

PLAINTEXT FILE

Choose file

Browse

KEY FILE

Choose file

Browse

ENCRYPT

ENCRYPTED FILE

Choose file

Browse

KEY FILE

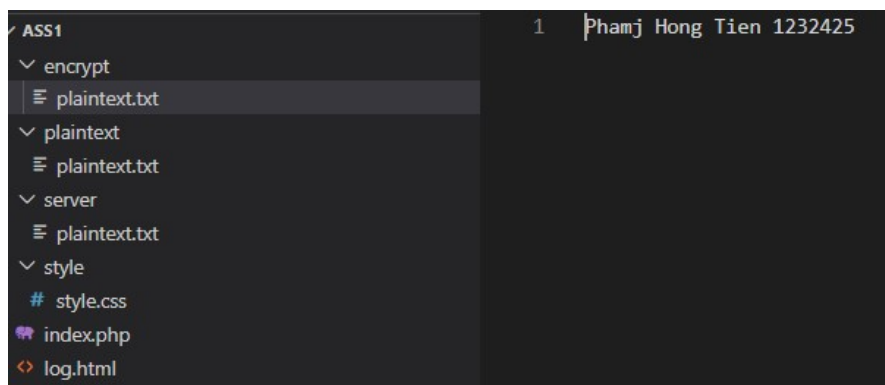
Choose file

Browse

DECRYPT

Hình 14: Sau khi mã hóa thành công hệ thoại sẽ thông báo

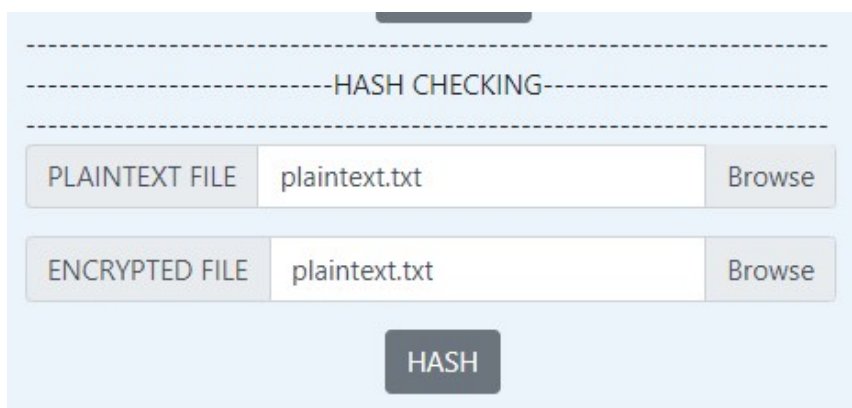
Sau khi giải mã thành công, sẽ tạo ra một file đã được giải mã và lưu nó vào thư mục encrypt



Hình 15: Nội dung file được giải mã

2.2.4 Công cụ hash

Với công cụ này ta có thể kiểm tra tính toàn vẹn của file trước và sau khi trải qua quá trình mã hóa, giải mã bằng cách hash nội dung của 2 file này bằng chuẩn md5 rồi sau đó so sánh giá trị. Nếu giá trị cho ra trùng nhau thì có nghĩa là file này không thay đổi tức dữ liệu được toàn vẹn sau quá trình mã hóa, giải mã.



Hình 16: Chọn file plaintext.txt ban đầu trong thư mục plaintext và plaintext.tx sau khi qua quá trình được lưu trong thư mục encrypt

Welcome, **Phạm Tiến** [LOGOUT](#)

(9:21 PM) User **Phạm Tiến** uploaded file: **encrypt.png** to server.
(9:33 PM) User **Phạm Tiến** uploaded file: **plaintext.txt** to server.
(9:50 PM) User **Phạm Tiến** uploaded file: **plaintext.txt** to encrypt.
(10:06 PM) User **Phạm Tiến**:
hash value of plaintext:
25c912c261dd987658ba39b2137cc571
hash value after decrypt:
25c912c261dd987658ba39b2137cc571

PLAINTEXT FILE [Browse](#)

KEY FILE [Browse](#)

[ENCRYPT](#)

ENCRYPTED FILE [Browse](#)

KEY FILE [Browse](#)

[DECRYPT](#)

PLAINTEXT FILE [Browse](#)

ENCRYPTED FILE [Browse](#)

[HASH](#)

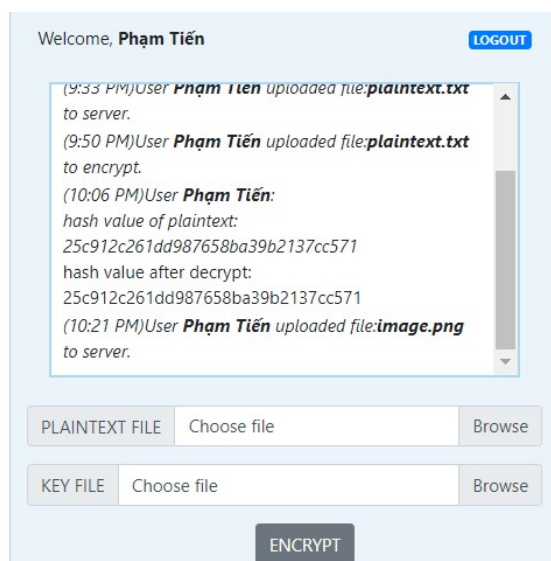
Hình 17: Sau khi hash giá trị sẽ hash của 2 file sẽ được hiển thị trên hộp thoại, ta thấy giá trị giống nhau nên tính toán vẹn được đảm bảo.

2.2.5 Mã hóa và giải mã file khác

Ngoài ra ta có thể mã hóa các file (.jpg,.png,...)

Cryptography and Network Security

Page 14/20



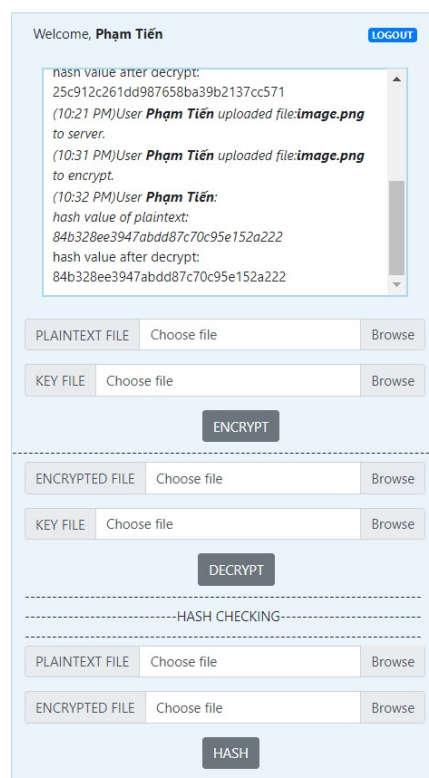
Hình 18: Mã hóa file định dạng .png



Hình 19: Ảnh ban đầu



Hình 20: Nội dung file sau khi được mã hóa được hiển thị dưới dạng text



Hình 21: Sau khi mã hóa giải mã, hash để kiểm tra tính toàn vẹn dữ liệu

2.3 Các hàm mã hóa

2.3.1 Hàm openssl_encrypt(), openssl_decrypt()

Với tham số đầu vào là 1 biến chứa chuỗi kí tự cần được mã hóa, 1 biến key chứa nội dung key và 1 biến vecto khởi tạo được tạo ngẫu nhiên bằng hàm `openssl_random_pseudo_bytes(openssl_cipher_iv_length('aes-256-cbc'))`;

Hàm `openssl_encrypt` này sẽ trả về một mảng đã được mã hóa. Hàm `openssl_decrypt` này sẽ giải mã chuỗi đã được mã hóa và trả về chuỗi đã được giải mã.

```
$iv = openssl_random_pseudo_bytes(openssl_cipher_iv_length('aes-256-cbc'));
// Encrypt the data using the $iv
$encrypted = openssl_encrypt($contents, 'aes-256-cbc', $key, 0, $iv);
// Write content to the destination file;
```

Hình 22: Hàm openssl_encrypt

2.3.2 base64_encode(), base64_decode()

`base64_decode()` là một hàm chuyển một chuỗi về một chuỗi gồm những kí thuộc bảng mã ASCII.

-base64_encode() dùng để chuyển ngược lại.

2.3.3 md5()

Hàm này để hash một chuỗi theo chuẩn MD5 với giá trị hash là 128bit. Ta dùng hàm này với mục đích để kiểm tra tính toàn vẹn dữ liệu.

2.4 Gọi hàm

Khi ta upload file cần được mã hóa kèm với file key và nhấn nút 'ENCRYPT'. Chương trình sẽ bắt sự kiện và lấy giá trị của file upload và chạy hàm encrypt_file(). Trong hàm encrypt_file(), nó sẽ dùng hàm file_get_contents() để đọc dữ liệu 1 file dưới dạng chuỗi. Tiếp tục chương trình tạo vecto khởi tạo ngẫu nhiên bằng hàm openssl_random_pseudo_bytes(openssl_cipher_iv_length('aes-256-cbc')). Sau đó gọi hàm openssl_encrypt() để mã hóa chuỗi vừa mới đọc ở trên. Giá trị trả về sẽ được mã hóa tiếp bằng base64_encode() với giá trị đầu vào dưới dạng: chuỗi mã hóa :: chuỗi vecto ngẫu nhiên.

Quá trình giải được thực hiện ngược lại. Chương trình gọi hàm base64_decode() để trả ngược về chuỗi trước đó. Sau dùng hàm explode() để tách chuỗi mã hóa ra khỏi chuỗi. Sau đó dùng hàm openssl_decrypt() để giải mã chuỗi. Cuối cùng ta sẽ được một chuỗi plaintext chứa nội dung ban đầu của file. Quá trình giải mã hoàn tất.

```
function encrypt_file($file, $key) {
    //get content of file need encrypt
    $contents = file_get_contents($file);
    //generate an initialization vector
    $iv = openssl_random_pseudo_bytes(openssl_cipher_iv_length('aes-256-cbc'));
    //encrypt the data using AES 256 encryption in CBC mode using our encryption key and initialization vector.
    $encrypted = openssl_encrypt($contents, 'aes-256-cbc', $key, 0, $iv);
    //write content in the destination file
    //Open the file and remove a file pointer resource
    $handle = fopen($file, 'w') or die("Could not open a file.");
    //write encrypted into file
    fwrite($handle, base64_encode($encrypted . ':' . $iv)) or die("Could not write to file.");
    //Close the opened file pointer
    fclose($handle);
}

function decrypt_file($file, $key){
    //get content of file need decrypt
    $contents = file_get_contents($file);
    //split into encrypted and iv
    list($encrypted, $iv) = explode(':', base64_decode($contents), 2);
    //decrypt
    $decrypted = openssl_decrypt($encrypted, 'aes-256-cbc', $key, 0, $iv);
    //open file to write
    $handle = fopen($file, 'w') or die("Could not open a file.");
    //write plaintext into file
    fwrite($handle, $decrypted) or die("Could not write to file.");
    //Close file
    fclose($handle);
}
```

Hình 23: Hàm encrypt_file và hàm decrypt_file

3 Phân tích và kết luận

3.1 Kết quả đạt được

Một ứng dụng với chức mã hóa, giải mã và kiểm tra tính toàn vẹn dữ liệu.

Các tính năng khác:

- Đăng nhập
 - Theo dõi tình trạng hoạt động
- Mã hóa:
- Sử dụng thuật toán AES

3.2 Đánh giá

- Trang web đáp ứng cơ bản đầy đủ yêu cầu của đề bài
- Có nhiều tính năng thêm hữu ích
- Thực hiện mã hóa gần như mọi file
- Tuy nhiên chỉ sử dụng thư viện có sẵn, chưa thực sự đi sâu vào giải thuật mã hóa -Chưa tối ưu

4 Hướng phát triển

- Thực hiện và điều chỉnh giải thuật mã hóa phù hợp hơn, tối ưu hơn
- Thực hiện xây dựng một server riêng biệt để tăng tính hiệu quả
- Quản lý, tăng cường tính năng đăng nhập
- Phát triển ứng thành ứng chat trực tuyến, chia sẻ file



Tài liệu

- [1] Advanced Encryption Standard
“<https://en.wikipedia.org/wiki/Advanced_Encryption_Standard>”
- [2] base64
“<<https://www.php.net/manual/en/function.base64-decode.php>>”
- [3] openssl_encrypt
“<<https://www.php.net/manual/en/function.openssl-encrypt.php>>”