ĐẠI HỌC QUỐC GIA HÀ NỘI

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ



BÁO CÁO

Xây dựng ứng dụng mã hóa dữ liệu ổ cứng sử dụng hệ mật Vigenère

|  |  |
| --- | --- |
| Môn học: Mật mã và an toàn dữ liệu | |
|  | Học viên: Trần Quang Linh – 2102540 |
|  |  |
|  |  |

Hà Nội – 2022

Mục lục

[Phần I. Hệ mật Vigenère 4](#_Toc121260355)

[1. Công thức mã hóa và giải mã trong hệ mật Vigenere 4](#_Toc121260356)

[2. Ứng dụng hệ mật Vigenere vào mã hóa và giải mã tệp tin 5](#_Toc121260357)

[Phần II. Cài đặt 6](#_Toc121260358)

[1. Xử lý mật khẩu 6](#_Toc121260359)

[2. Mã hóa/ giải mã file dựa trên hệ mật Vigenere 6](#_Toc121260360)

[3. Xây dựng giao diện ứng dụng 7](#_Toc121260361)

[Phần III. Kết quả 8](#_Toc121260362)

[Tài liệu tham khảo 9](#_Toc121260363)

Tóm tắt

Báo cáo này được thực hiện sau quá trình tìm hiểu về kỹ thuật giải quyết vấn đề bằng SAT và các phương pháp SAT encoding nổi tiếng. Để có được cái nhìn tổng quan về phương pháp cũng như giúp những người nghiên cứu sau dễ dàng tiếp cận, báo cáo này chọn bài toán giải Slitherlink làm ví dụ để trình bày về quy trình giải quyết một bài toán bằng kỹ thuật SAT. Báo cáo gồm các phần sau:

**Phần I. Hệ mật Vigenere –** Giới thiệu về hệ mật Vigenère và ứng dụng trong mã hóa tệp tin.

**Phần II. Cài đặt –** Mô tả quy trình xây dựng ứng dụng mã hóa tệp tin sử dụng hệ mật Vigenère.

**Phần III. Kết quả** – Demo kết quả xây dựng phần mềm

# Hệ mật Vigenère

Hệ mật Vigenere được thiết kế bởi Blasic de Vigenere – một nhà toán học người Pháp ở thế kỷ 16. Đây là một phương pháp dùng để mã hóa văn bản sử dụng phương pháp thay thế bảng chữ cái đơn giản.

## Công thức mã hóa và giải mã trong hệ mật Vigenere

Mật mã Vigenère sử dụng bảng 26×26 với A đến Z làm tiêu đề hàng và tiêu đề cột Bảng này thường được gọi là Vigenère Tableau, Vigenère Table hoặc Vigenère Square. Chúng ta sẽ sử dụng Bảng Vigenère. Hàng đầu tiên của bảng này có 26 chữ cái tiếng Anh. Bắt đầu từ hàng thứ hai, mỗi hàng có các chữ cái dịch chuyển sang trái một vị trí theo chu kỳ. Ví dụ, khi B được chuyển đến vị trí đầu tiên trên hàng thứ hai, chữ A sẽ di chuyển đến cuối cùng.

Calendar

Description automatically generated with medium confidence

Ví dụ: hàng của B có được bằng cách dịch chuyển hàng của A sang trái một vị trí. Điều này tương đương với việc chuyển bảng chữ cái sang bên phải một vị trí. Đối với hàng của B, A được chuyển sang B và B được chuyển sang C và do đó, A được mã hóa thành B và B được mã hóa thành C. Tương tự, đối với hàng của D cách A ba vị trí, A được dịch chuyển ba vị trí sang D, B được dịch ba vị trí sang E, và C được dịch ba vị trí sang F. Do đó, A, B và C được mã hóa sang D, E và F bằng cách dịch sang ba vị trí bên phải. Nói chung, nếu một chữ cái P rõ ràng được mã hóa bởi một chữ cái từ khóa K nằm cách A d vị trí, thì P được mã hóa bởi K thành chữ C nằm d vị trí bên phải của P. Chúng ta phải xem xét sự dịch chuyển theo chu kỳ. Nếu các chữ cái A, B, C, ..., Z được gán các giá trị 0, 1, 2, ..., 25 thì mỗi chữ cái từ khóa chỉ đơn giản là khoảng cách từ chữ cái đó đến A. Kết quả là bản mã chữ C thu được như sau, trong đó "mod" là phép chia lấy dư modulo:

**C = (P + d) mod 26**

Tóm lại, nếu từ khóa được lặp lại đủ số lần sao cho tổng độ dài bằng độ dài của bản rõ, đối với bản rõ p1p2...pn, từ khóa k1k2...kn và bản mã c1c2...cn, chúng ta có

***ci* = (*pi* + *ki*) mod 26**

Giải mã là thủ tục đảo ngược bằng cách dịch chuyển bản mã sang trái. Vì dịch chuyển sang trái là một phép trừ nên quy trình giải mã đơn giản là:

***pi* = (*ci* - *ki*) mod 26**

## Ứng dụng hệ mật Vigenere vào mã hóa và giải mã tệp tin

Đối với mã hóa tệp tin, thay vì mã hóa từng ký tự như mã hóa văn bản, ta sẽ mã hóa từng byte (8 bit) của tập tin đó. Với khóa có độ dài m ta sẽ lặp lại khóa cho đến khi độ dài khóa bằng tổng số byte của 1 tệp tin.

Với P = C = (Z256)n (256 = 2­8)

* Công thức mã hóa:

ek(p1, p2,......pm, pm+1,pm+2,pm+3,.........) = (p1+ k1, p2 + k2,...., pm+km, pm+1+k1, pm+2+k2,........) (mod 256)

* Công thức giải mã:

dk(c1, c2,......cm, cm+1, cm+2, cm+3,.........) = (c1- k1, c2 - k2, ...., cm-km, cm+1-k1,, cm+2-k2) (mod 256)

# Cài đặt

## Xử lý mật khẩu

Để đảm bảo tính bảo mật cho hệ mật Vigenere, ta cần chọn khóa có độ dài đủ lớn, tuy nhiên người dùng thường không thể nhớ được các mật khẩu dài và phức tạp. Phần mềm trong báo cáo giải quyết vấn đề này bằng cách sử dụng nhiều thuật toán băm kết hợp với nhau để tạo mật khẩu được dùng trong mã hóa và giải mã từ mật khẩu ngắn của người dùng.

Hiện tại, chương trình sử dụng 4 thuật toán băm có sẵn là SHA224, SHA256, SHA384, SHA512. Đặc điểm của các hàm băm SHA là đảm bảo độ dài của giá trị băm luôn cố định. VD SHA256 cho đầu ra là một chuỗi 256 bit. Do đó độ dài sử dụng để mã hóa và giải mã dùng cho ứng dụng luôn đạt độ dài tối thiểu là 1376 bit.

## Mã hóa/ giải mã file dựa trên hệ mật Vigenere

Cài đặt mã hóa:

def encrypt(self, inputFile, outputFile, password):

        byte = inputFile.read(1)

        kidx = 0

        while byte != b"":

            outputFile.write(((ord(byte) + password[kidx]) % 256).to\_bytes())

            kidx = (kidx+1) % len(password)

            byte = inputFile.read(1)

Cài đặt giải mã:

def decrypt(self, inputFile, outputFile, password):

        byte = inputFile.read(1)

        kidx = 0

        while byte != b"":

            outputFile.write(((ord(byte) - password[kidx]) % 256).to\_bytes())

            kidx = (kidx+1) % len(password)

            byte = inputFile.read(1)

## Xây dựng giao diện ứng dụng

Ứng dụng được xây dựng giao diện thông qua thư viện CustomTkinter, là một bản nâng cấp của Tkinter dành cho ngôn ngữ lập trình Python. Giao diện ứng dụng cung cấp hai chức năng chính là mã hóa và giải mã thư mục. Sau khi chọn một tính năng, người dùng có thể:

* Xem tổng số file và kích thước của tất cả các tệp tin trong thư mục.
* Nhập mật khẩu dạng chuỗi ký tự để mã hóa hoặc giải mã thư mục.
* Tùy chọn xóa tệp tin gốc sau khi mã hóa hoặc giải mã thành công.
* Cho phép người dùng mã hóa/ giải mã nhiều thư mục cùng lúc.
* Xem tiến trình trong khi thực hiện mã hóa và giải mã bao gồm:
  + Thời gian chạy đã qua
  + Tiến trình chạy mã hóa/ giải mã tệp tin hiện tại (kích thước đã mã hóa / tổng kích thước tệp tin)
  + Tiến trình chạy mã hóa/ giải mã thư mục (tổng số tệp tin đã mã hóa / tổng số tệp tin trong thư mục)

# Kết quả

Sau quá trình xây dựng, sản phẩm thu được là một phần mềm hoàn thiện có giao diện đơn giản và trực quan giúp người dùng có thể mã hóa thư mục hoặc

Graphical user interface

Description automatically generated

Hình III‑1. Giao diện ứng dụng mã hóa thư mục sử dụng hệ mật Vigenere chạy trên hệ điều hành Windows.

# Tài liệu tham khảo

[1] https://pages.mtu.edu/~shene/NSF-4/Tutorial/VIG/Vig-Algebraic.html