TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO CUỐI KỲ MÔN LẬP TRÌNH MẠNG CĂN BẢN**

**XÂY DỰNG CHƯƠNG TRÌNH QUẢN LÝ SINH VIÊN TRÊN GIAO THỨC UDP**

*Người hướng dẫn*: **TS TỐNG THANH VĂN**

*Người thực hiện*: **LÊ TUẤN THÔNG - 51900562**

**PHẠM NGUYỄN – 52000092**

Lớp **: 19050401-20050261**

Khoá  **: 23-24**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO CUỐI KỲ MÔN LẬP TRÌNH MẠNG CĂN BẢN**

**XÂY DỰNG CHƯƠNG TRÌNH QUẢN LÝ SINH VIÊN TRÊN GIAO THỨC UDP**

Người hướng dẫn: **TS TỐNG THANH VĂN**

Người thực hiện: **LÊ TUẤN THÔNG - 51900562**

**PHẠM NGUYỄN – 52000092**

Lớp **: 19050401-20050261**

Khoá  **: 23-24**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023**

LỜI CẢM ƠN

Đầu tiên, tập thể nhóm chúng em xin gửi lời cảm ơn đến Khoa Công nghệ thông tin trường đại học Tôn Đức Thắng đã tạo điều kiện cho chúng em được tham gia học tập môn học này. Ngoài ra khoa còn tạo điều kiện để chúng em có cơ hội hoàn thành môn học một cách tốt nhất.

Tiếp theo, chúng em xin gửi lời cảm ơn đến thầy Tống Thanh Văn, giảng viên bộ môn Lập trình mạng căn bản đã hướng dẫn chỉ dạy chúng em trong suốt quá trình học và hoàn thành bài báo cáo này. Và bên cạnh đó, chúng em cũng xin cảm ơn đến thầy cô giảng viên tại trường, đặt biệt là  các giảng viên trong trong khoa Công Nghệ Thông Tin đã dạy chúng em nhiều kiến thức cơ bản, thiết yếu và trang bị cho chúng em một lượng tri thức trong suốt quá trình học cũng như trong quá trình làm bài báo cáo.

Cuối cùng em xin chúc khoa Công nghệ thông tin trường đại học Tôn Đức Thắng và thầy Tống Thanh Văn có thật nhiều sức khỏe để có thể góp phần cống hiến cho sự phát triển khoa Công nghệ thông tin trường đại học Tôn Đức Thắng cũng như truyền đạt thật nhiều kiến thức đến sinh viên chúng em.

Tập thể nhóm chúng em xin chân thành cảm ơn!

**CÔNG TRÌNH ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi và được sự hướng dẫn khoa học của TS Tống Thanh Văn;. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong luận văn còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung luận văn của mình.** Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày tháng năm*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Lê Tuấn Thông*

*Phạm Nguyễn*

MỤC LỤC

[LỜI CẢM ƠN i](#_Toc131633451)

[MỤC LỤC 1](#_Toc131633452)

[DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT 2](#_Toc131633453)

[DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ 3](#_Toc131633454)

[CHƯƠNG 1 – GIỚI THIỆU 4](#_Toc131633455)

[1.1 Giới thiệu đề tài nghiên cứu 4](#_Toc131633456)

[1.2 Mục đích nghiên cứu và phương pháp nghiên cứu 4](#_Toc131633457)

[CHƯƠNG 2 – CƠ SỞ LÝ THUYẾT 5](#_Toc131633458)

[2.1 Mô hình Client – Server 5](#_Toc131633459)

[2.2 Giao thức UDP 5](#_Toc131633460)

[2.3 Mã hóa dữ liệu 6](#_Toc131633461)

[2.3.1 Thuật toán mã hóa DES 6](#_Toc131633462)

[2.3.2 Thuật toán Vigenere 8](#_Toc131633463)

[2.4 Kết nối cơ sở dữ liệu MySQL bằng JDBC 9](#_Toc131633464)

[CHƯƠNG 3 –YÊU CẦU HỆ THỐNG VÀ XÂY DỰNG CHƯƠNG TRÌNH 11](#_Toc131633465)

[3.1 Yêu cầu hệ thống : 11](#_Toc131633466)

[3.2 Mô tả các chức năng chính của hệ thống: 11](#_Toc131633467)

[3.2.1 Yêu cầu chức năng hệ thống 12](#_Toc131633468)

[3.2.2 Xây dựng module hệ thống 12](#_Toc131633469)

[3.2.3 Xây dựng giao diện và quy trình người dùng 14](#_Toc131633470)

[CHƯƠNG 4 – KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 19](#_Toc131633471)

DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT

**CÁC KÝ HIỆU**

**CÁC CHỮ VIẾT TẮT**

DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ

**DANH MỤC HÌNH**

[Hình 1. (1.3.1)Mã hóa DES 6](#_Toc131551418)

[Hình 2. (1.3.1)Một vòng lặp DES 7](#_Toc131551419)

[Hình 3. (1.3.2)Bảng mã Vigenère 8](#_Toc131551420)

[Hình 4.(3.2.3) frmClient.java 15](#_Toc131551421)

[Hình 5.(3.2.3)frmMenu.java 15](#_Toc131551422)

[Hình 6.(3.2.3) frmInput.java 16](#_Toc131551423)

[Hình 7.(3.2.3) frmShow.java 17](#_Toc131551424)

[Hình 8.(3.2.3) frmText.java 17](#_Toc131551425)

[Hình 9.(3.2.3) FileInformation.java 18](#_Toc131551426)

**DANH MỤC BẢNG**

CHƯƠNG 1 – GIỚI THIỆU

* 1. Giới thiệu đề tài nghiên cứu

Đề tài này tập trung vào xây dựng một chương trình quản lý sinh viên sử dụng mô hình Client – Server bằng Java với giao thức UDP. Chương trình sẽ cho phép người dùng nhập thông tin sinh viên và gửi nó đến Server để lưu trữ và quản lý. Client cũng có thể gửi một đoạn text đến Server để lưu trữ hoặc yêu cầu Server lấy nội dung của một tập tin text mà Client đã lưu trên Server. Chương trình cũng sử dụng mã hóa Vigenere và thuật toán DES để bảo vệ dữ liệu truyền tải giữa Client và Server.

* 1. Mục đích nghiên cứu và phương pháp nghiên cứu

Mục đích của nghiên cứu này là xây dựng một chương trình quản lý sinh viên sử dụng mô hình Client-Server bằng Java với giao thức UDP. Nghiên cứu này sẽ giúp cải thiện quá trình quản lý thông tin sinh viên và bảo mật thông tin của sinh viên trong quá trình truyền tải và lưu trữ.

Cụ thể, nghiên cứu này sẽ đạt được các mục tiêu sau:

* Phát triển phần mềm Client-Server bằng Java với giao thức UDP để quản lý thông tin sinh viên.
* Áp dụng các thuật toán mã hóa Vigenere và DES để bảo vệ thông tin của sinh viên trước khi lưu trữ và truyền tải giữa Client và Server.
* Lưu trữ thông tin sinh viên vào cơ sở dữ liệu SQL và truy vấn dữ liệu để hiển thị kết quả cho người dùng.
* Đặt nền móng để có thể mở rộng và phát triển lĩnh vực này trong tương lai.
* Phương pháp nghiên cứu:
* Nghiên cứu và áp dụng các kỹ thuật lập trình Client-Server và UDP.
* Sử dụng Java để thiết kế và triển khai chương trình.
* Sử dụng Vigenere và DES để bảo vệ dữ liệu truyền tải dữ liệu giữa Client-Server-Database.

CHƯƠNG 2 – CƠ SỞ LÝ THUYẾT

2.1 Mô hình Client – Server

Client – Server là mô hình mạng máy tính bao gồm 2 thành phần chính là máy khách (client) và máy chủ (server). Trong mô hình này, máy chủ là nơi lưu trữ tài nguyên, cài đặt các chương trình dịch vụ và thực hiện các yêu cầu của khách hàng. Máy khách nhận vai trò gửi yêu cầu đến máy chủ. Máy khách bao gồm máy tính và thiết bị điện tử nói chung.

Mô hình Client – Server cho phép mạng tập tin trung tâm của các ứng dụng và chức năng tại một hoặc nhiều tập tin dịch vụ mạng chuyên dụng. Các máy này trở thành trung tâm của hệ thống. Hệ điều hành của Client server cho phép người dùng chia sẻ đồng thời cùng một tài nguyên, không quan trọng vị trí địa lý.

Mô hình Client – Server như sau: Client – Server là mô hình tổng quát nhất, trên thực tế thì một Server có thể được nối tới nhiều Server khác nhằm làm việc hiệu quả và nhanh hơn. Khi nhận được 1 yêu cầu từ Client, Server này có thể gửi tiếp yêu cầu vừa nhận được cho Server khác ví dụ như Database Server vì bản thân nó không thể xử lý yêu cầu này được. Máy Server có thể thi hành các nhiệm vụ đơn giản hoặc phức tạp.

2.2 Giao thức UDP

UDP (User Datagram Protocol) là giao thức truyền tải hướng không kết nối.

Giao thức UDP không có quá trình bắt tay 3 bước như giao thức TCP, mà thay vào đó, nó sẽ truyền dữ liệu ngay lập tức, bỏ qua cơ chế xác thực cũng như đánh số thứ tự cho các gói tin.

Các gói tin liên tục được gửi cho bên nhận, bên gửi không cần phải chờ xem bên nhận đã nhận được hay chưa, mà cứ tiếp tục gửi tiếp.

Chính vì vậy, nếu bên nhận có bị mất một vài gói tin thì coi như sẽ bị mất (bên gửi sẽ không gửi lại), đó cũng là lý do tại sao những ứng dụng dùng giao thức UDP sẽ rất nhanh, vì nó bỏ qua bước kiểm tra lỗi khi truyền gói tin.

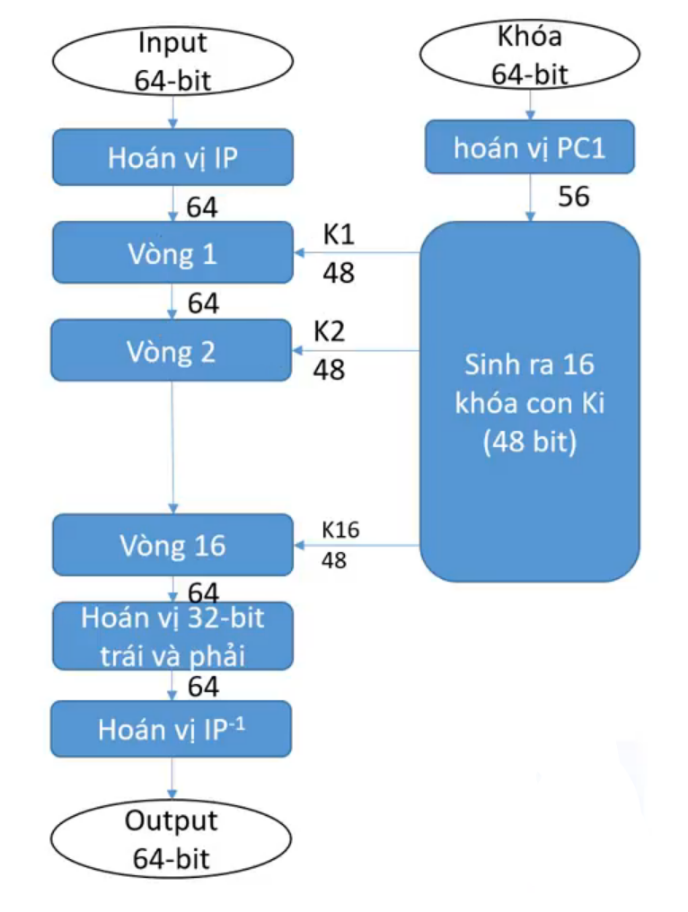
Việc bỏ qua cơ chế sửa lỗi giúp UDP tăng tốc độ kết nối và giảm độ trễ thời gian thực, một điều rất quan trọng trong các ứng dụng cần xử lý liên tục.

2.3 Mã hóa dữ liệu

2.3.1 Thuật toán mã hóa DES

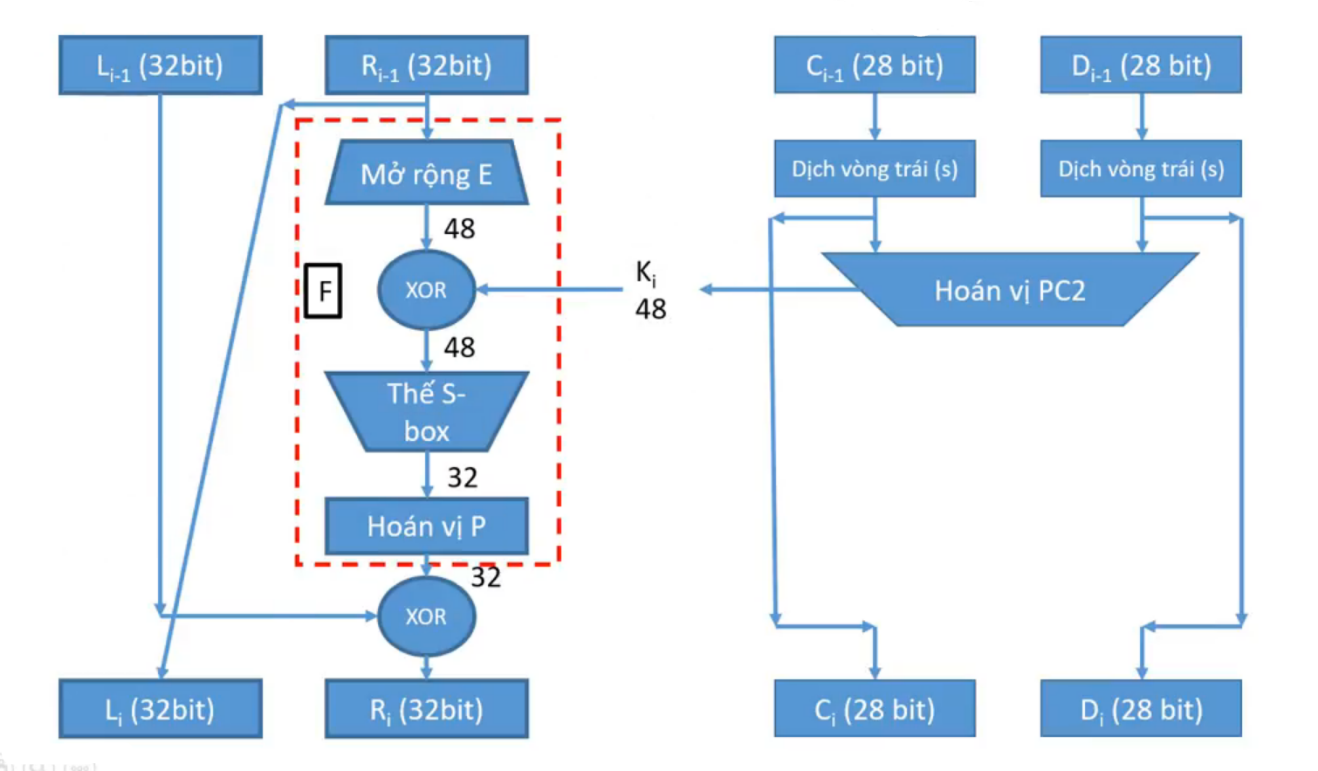
DES – viết tắt của cụm từ Data Encryption Standard là một thuật toán mã hoá khối ra đời giữa những năm 1970 và được sử dụng phổ biến trong vòng 30 năm sau đó.Vào những năm 1973, 1974 máy tính đã bắt đầu được sử dụng nhiều trong các ngân hàng, công ty tài chính lớn. Nhu cầu giữ bí mật thông tin đã trở nên cấp thiết nhưng các thuật toán cổ điển lại không đáp ứng được. Viện chuẩn và công nghệ Hoa Kỳ NIST (National Institute of Standards and Technology) đã kêu gọi xây dựng các thuật toán mã hóa mới và an toàn hơn. Sau đó các ngân hàng Mỹ đã đặt hàng cho công ty IBM – người khổng lồ trong lĩnh vực điện toán thời đó, thiết kế một thuật toán mã hoá hoàn toàn mới. Lúc đầu, IBM đưa ra thuật toán Lucifer và sau đó thuật toán này được thiết kế, chỉnh sửa lại thành thuật toán DES. Năm 1976 DES đã được NIST công nhận là chuẩn mã hoá dữ liệu, và đến năm 1981 DES trở thành chuẩn mật mã của ANSI (American National Standards Institute). DES được sử dụng rộng rãi trên thế giới, đặc biệt là trong các giao dịch ngân hàng, thông tin liên lạc. Định kỳ, ANSI công nhận lại chuẩn mật mã cho thời gian 5 năm tiếp theo. DES là thuật toán mã hóa khối được sử dụng phổ biến trong suốt gần 30 năm, từ năm 1976 đến những năm gần đây.

Quá trình mã hóa một khối 64-bit đầu vào bằng DES thông qua các bước: (1) Hoán vị ban đầu IP (Initial Permutation); (2) 16 vòng tính toán phức tạp có sử dụng khóa; (3) Hoán vị kết thúc, là nghịch đảo  của IP. Quy trình mã hoá và tạo khoá của thuật toán DES có thể được mô tả  trong sơ đồ sau:



Hình 1. (1.3.1)Mã hóa DES

Phần bên trái của hình vẽ là quy trình mã hoá dữ liệu. Phần bên phải là quy trình tạo các khoá con (subkey).



Hình 2. (1.3.1)Một vòng lặp DES

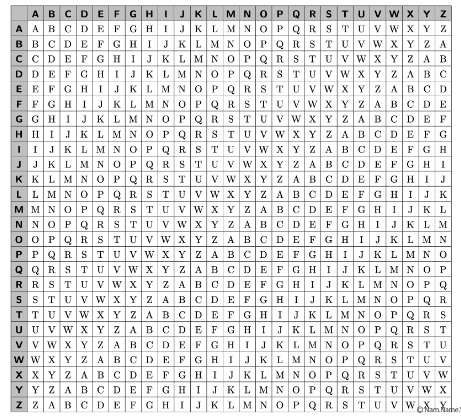
2.3.2 Thuật toán Vigenere

Trong các loại mã hóa cổ điển: mã hóa Caesar, mã hóa Affine, được gọi chung là **mã thay thế dùng một bảng chữ cái** (monoalphabetic substitution cipher). Nghĩa là ta dùng một ánh xạ các chữ cái trong bảng mã thành bản rõ và tất cả các chữ cái trong bản mã đều dùng chung một ánh xạ.

Yếu điểm của loại mã này là bản mã sẽ giữ lại các đặc điểm mẫu từ và tần xuất của văn bản gốc. Để loại bỏ yếu điểm trên, các nhà lập mã đã tạo ra một hệ mã khác gọi là **mã thay thế dùng nhiều bảng chữ cái** (polyalphabetic substitution cipher).

Hệ mã này được đặt theo tên của một nhà mật mã học người Pháp Blaise de Vigenère (1523-1596). **Mật mã Vigenère** là một dạng đơn giản của mật mã thay thế dùng nhiều bảng chữ cái.

Để mã hóa, ta dùng một hình vuông Vigenère (hình dưới).Gồm 26 hàng, mỗi hàng dịch về bên trái một bước so với hàng phía trên, tạo thành 26 bảng mã Caesar. Trong quá trình mã hóa, tùy theo từ khóa mà mỗi thời điểm ta dùng một dòng khác nhau để mã hóa văn bản.



Hình 3. (1.3.2)Bảng mã Vigenère

2.4 Kết nối cơ sở dữ liệu MySQL bằng JDBC

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở được phát triển bởi công ty Oracle. Nó là một trong những hệ quản trị cơ sở dữ liệu phổ biến nhất trên thế giới và được sử dụng rộng rãi trong các ứng dụng web.

MySQL cung cấp các tính năng như:

* Hỗ trợ nhiều ngôn ngữ lập trình như C++, Java, Python, Perl, PHP, và Ruby.
* Hỗ trợ đa nền tảng, cho phép chạy trên nhiều hệ điều hành như Windows, Linux, macOS, và FreeBSD.
* Hỗ trợ các loại dữ liệu phổ biến như số, văn bản, ngày tháng, và hình ảnh.
* Cung cấp khả năng lưu trữ và truy xuất dữ liệu hiệu quả và đáng tin cậy.
* Hỗ trợ các tính năng bảo mật như mã hóa mật khẩu, phân quyền truy cập, và SSL.
* Hỗ trợ kết nối đồng thời của nhiều người dùng.

Áp dụng vào hệ thống:

* Đầu tiên ta cần phải tải cơ sở dữ liệu MySQL, chúng em quyết định sử dụng chương trình XAMPP để chạy cơ sở dữ liệu trên máy server vì XAMPP hỗ trợ nhiều nền tảng hệ điều hành khác nhau, cho phép người dùng sử dụng MySQL trên các hệ điều hành khác nhau một cách dễ dàng, bên cạnh đó XAMPP cũng là XAMPP là một phần mềm miễn phí và mã nguồn mở, cho phép người dùng sử dụng một cách tự do và linh hoạt.
* Đồng thời để kết nối MySQL với chương trình Java, ta phải sử dụng JDBC (Java Database Connectivity). Đây là một API chuẩn trong Java cho phép truy cập và thao tác với cơ sở dữ liệu từ chương trình Java.
* Sau đó chúng em sử dụng các phương thức mà thư viện này cung cấp để thực hiện kết nối và các truy vấn với cơ sở dữ liệu đã Host

CHƯƠNG 3 –YÊU CẦU HỆ THỐNG VÀ XÂY DỰNG CHƯƠNG TRÌNH

3.1 Yêu cầu hệ thống :

Xây dựng chương trình Quản lý Sinh viên theo mô hình Client – Server bằng Java với giao thức UDP

* Client:
* + Nhập vào tên địa chỉ, cổng kết nối với Server, nếu không thành công thì thông báo nhập lại còn thành công thì kết nối.
* + Sau đó vào giao diện gửi Text cho server hoặc thêm thông tin của sinh viên
* + Gửi Text: gửi một đoạn text, tên tập tin cần lưu gửi lên Server.
* + Hiển thị nội dung file Text đã gửi cho Server.
* + Thêm sinh viên: nhập các dòng dữ liệu gửi lên server gồm họ và tên mã số sinh viên, điểm toán, văn và tiếng anh.
* + Hiển thị kết quả Server trả về thông tin của sinh viên gồm họ tên mã số và điểm trung bình.
* Server:
* + Nhận text từ client gửi lên nội dung thì mã hóa bằng Vigenere rồi lưu file với tên mà Client gửi lên.
* + Mở File mà Client đã lưu trên hệ thông tiến hành giải mã bằng Vigenere rồi sau đó trả về cho Client.
* + Nhận thông tin sinh viên từ Client và mã hóa bằng thuật toán DES sau đó lưu vào SQL
* + Nhận dữ liệu từ SQL và giải mã sau đó trả về Client thông tin họ tên mã số và điểm trung bình

3.2 Mô tả các chức năng chính của hệ thống:

3.2.1 Yêu cầu chức năng hệ thống

* Kết nối với server bằng thông tin địa chỉ và cổng kết nối được nhập từ người dùng.
* Thêm thông tin sinh viên và lưu trên cơ sở dữ liệu.
* Hiển thị danh sách sinh viên từ CSDL về máy client.
* Gửi file text từ máy client đến máy server và lưu trữ tại máy server.
* Hiển thị danh sách file có trong thư mục lưu trữ của máy server.
* Mã hóa thông tin khi truyền tải và lưu trữ.

3.2.2 Xây dựng module hệ thống

Nhóm bọn em đã quyết định xây dựng 2 project riêng biệt để có thể dễ dàng phân công chức năng và phân biệt hoạt động giữa client và server. Project client sẽ chứa các form và giao diện để người dùng có thể thuận tiện tương tác với server, project server sẽ bao gồm các file code cung cấp các phương thức hỗ trợ kết nối, xử lý, truyền tải thông tin giữa CSDL-server và server-client.

A.Cơ sở dữ liệu:

* Một hệ thống thông tin sinh viên sẽ chứa nhiều thông tin của sinh viên bao gồm: họ và tên sinh viên, mã số sinh viên, điểm số các môn học (toán, văn, anh).
* Đối tượng: chứa class Sinhvien.java tương ứng với các thực thể sinh viên đã được khởi tạo trong cơ sở dữ liệu.
* Kết nối: DBAccess.java và MyConnection.java đóng vai trò cung cấp những phương thức trung gian hỗ trợ kết nối và truyền tải thông tin giữa hệ thống và CSDL.

B.Kết nối Client-Server:

Để tạo sự kết nối giữa các client tới server, hệ thống phải được bảo đảm có thể duy trì kết nối giữa các client để chúng có thể hoạt động song song độc lập với nhau.

Cụ thể với phần kết nối này sẽ có 2 file code thuộc package source trong project server và 1 file code thuộc package source trong project client được thực hiện:

* Server.java
* ServerCtr.java
* Client.java

Trong đó:

* Server.java có nhiệm vụ liên tục lắng nghe tạo sự kết nối và đường truyền dữ liệu từ phía các Clients. Nhưng bản thân Server không thể nhận định được cụ thể các client riêng biệt với nhau nên chúng ta cần có sự giúp đỡ của ServerCtr.java, có nhiệm vụ chủ chốt như một tiểu trình đảm bảo các Client hoạt động song song và cùng lúc đồng thời xử lý các dữ liệu nhận về hoặc truyền đi. Đồng thời ServerCtr.java cũng sẽ chịu trách nhiệm xử lý các thông tin khi nhận được yêu cầu từ người dùng như gửi thông tin sinh viên và file, hiển thị thông tin sinh viên và file và trả về kết quả tương ứng.
* Client.java thì đại diện cho các giao diện GUI nhằm lấy dữ liệu từ phía người dùng để truyền lên Server, tùy vào dạng dữ liệu mà server sẽ có các cách xử lý khác nhau. Đồng thời cổng và địa chỉ server cũng sẽ được nhập từ client.

C. Mã hóa dữ liệu theo thuật toán DES và Vigenere:

Ngoài thuật toán Vigenere theo yêu cầu của đề bài, nhóm chúng em còn sử dụng thuật toán DES để đa dạng mã hóa thông tin nhằm vào việc khi một dạng mã hóa bị tấn công thì một dạng khác vẫn có thể hoạt động bình thường.

File code Encode.java sẽ hỗ trợ các phương thức mã hóa và giải mã cho cả 2 thuật toán mã hóa DES và Vigenere.

D. Thêm và hiển thị danh sách sinh viên từ CSDL

Thêm thông tin sinh viên: Khi bắt được yêu cầu thêm sinh viên từ phía client thông qua Socket , việc này có nghĩa là tín hiệu này sẽ đi kèm với các thông tin liên quan đến sinh viên, ServerCtr sẽ bắt đầu thực hiện việc tạo một kết nối mới với CSDL, mã hóa các thông tin đó theo thuật toán DES, sau đó sẽ truyền tải dữ liệu lên CSDL và cuối cùng là thông báo kết quả việc tải thông tin sinh viên lên cơ sở dữ liệu có thành công hay không về phía client.

Hiển thị danh sách sinh viên: Khi server bắt được tín hiệu này, ServerCtr sẽ tạo một kết nối mới với CSDL, lấy toàn bộ thông tin sinh viên từ CSDL truyền về cho client thông qua Socket, dữ liệu được truyền về này sẽ được một lần nữa xử lý tại frmShow.java và hiển thị theo dạng JList.

E. Gửi, lưu trữ và hiển thị file

Trong chức năng gửi, lưu trữ và hiển thị File, ta cũng sẽ sử dụng cơ chế giao tiếp giữa Client và Server thông qua Socket. Hệ thống hiện tại chỉ có thể thực hiện 1 chức năng trên 1 client trong 1 thời điểm nên sau khi thực hiện xong 1 chức năng thì Socket trên client đó sẽ “trống” và có thể thực hiện chức năng tiếp theo. Tuy nhiên, điểm hạn chế của việc này là sẽ giảm hiệu suất và độ an toàn của hệ thống

Để lưu trữ file trên máy server:

* Thư viện FileWriter sẽ hỗ trợ phương thức ghi file với tên và nội dung đã nhập từ client vào thư mục mặc định trên máy server.
* Đường dẫn mặc định khi tải File lên Server sẽ là D:\\dev\\mangcanban\\db\\ trên máy server.

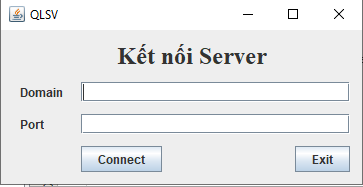
Để truyền thông tin file về client:

* File fileWorker.java sẽ hỗ trợ các phương thức để hệ thống có thể tương tác với các file trên máy server bao gồm: lấy tên file, tìm kiếm file, tìm kiếm nội dung trong file (sau khi đã giải mã), kiểm tra file có tồn tại hay không

3.2.3 Xây dựng giao diện và quy trình người dùng

Bao gồm các giao diện xây dựng bằng thư viện Swing java, khi từng giao diện thực hiện các chức năng khác nhau hiện ra, sẽ có một tín hiệu được gửi về cho ServerCtr thông qua Socket để thực hiện các chức năng này:

* Giao diện kết nối server (nhập thông tin server và nhấn connect):



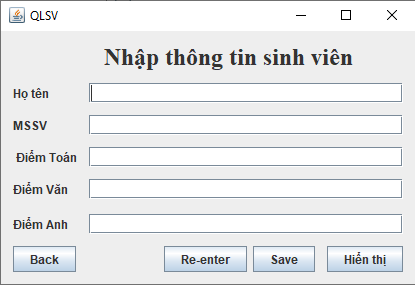
Hình 4.(3.2.3) frmClient.java

* Giao diện lựa chọn chức năng hệ thống:



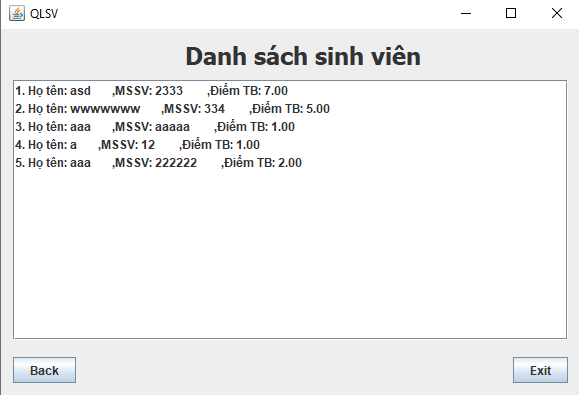
Hình 5.(3.2.3)frmMenu.java

* Giao diện nhập thông tin sinh viên (nhập thông tin sinh viên, nhấn nút save để lưu, re-enter để nhập lại toàn bộ, hiển thị để hiển thị danh sách sinh viên đã được lưu thành công trên CSDL):



Hình 6.(3.2.3) frmInput.java

* Giao diện hiển thị thông tin sinh viên:



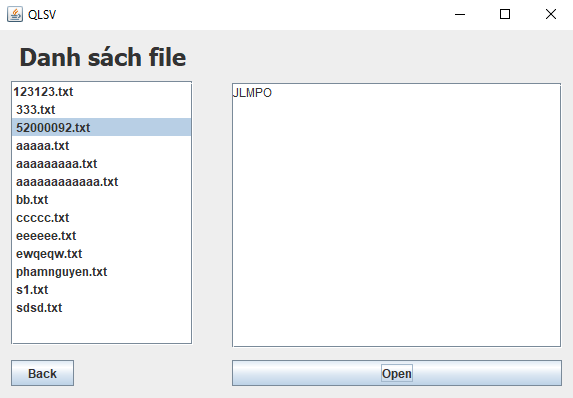
Hình 7.(3.2.3) frmShow.java

* Giao diện nhập tên và nội dung một file text (nhập nội dung file và tên file, nhấn nút send để có thể gửi và lưu trữ file trên máy chủ, nhấn nút open file để hiển thị giao diện danh sách file đã được lưu trữ trên máy chủ):



Hình 8.(3.2.3) frmText.java

* Giao diện hiển thị danh sách file đã được lưu trữ tại máy server (chọn file người dùng muốn và nhấn open để xem nội dung được lưu trữ sau khi đã mã hóa):



Hình 9.(3.2.3) FileInformation.java

CHƯƠNG 4 – KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Thông qua việc thực hiện bài đồ án này, chúng em đã bổ sung và hoàn thiện hơn các kiến thức liên quan đến:

* Giao thức UDP
* Thuật toán mã hóa DES
* Thuật toán Vigenere
* Mô hình client-server
* Áp dụng Java Swing và Java Socket

Đồng thời chúng em cũng nhận ra được phương hướng phát triển của hệ thống đã được thực hiện trong bài đồ án:

* Đảm bảo tính bảo mật: Để đảm bảo tính bảo mật của hệ thống, sử dụng mã hóa và chứng thực cho các gói tin được truyền qua mạng. Các gói tin không được mã hóa và chứng thực có thể dẫn đến các lỗ hổng bảo mật như tin tặc có thể đánh cắp thông tin.
* Tăng tính tin cậy và độ ổn định: Nếu hệ thống sử dụng UDP, gói tin có thể bị mất hoặc trùng lặp khi truyền qua mạng. Sử dụng các thuật toán để đảm bảo rằng gói tin được truyền đến đúng địa chỉ đích và không bị mất. Đảm bảo tính ổn định của hệ thống bằng cách xử lý các lỗi và ngoại lệ một cách thích hợp.
* Tối ưu hóa hiệu suất: Tối ưu hóa hiệu suất của hệ thống bằng cách sử dụng các thuật toán tối ưu và giảm thiểu độ trễ mạng.
* Phát triển các tính năng mới: Bổ sung các tính năng mới cho hệ thống của mình, chẳng hạn như chức năng trò chuyện trực tuyến, truyền file, hoặc đồng bộ hóa dữ liệu giữa các máy chủ khác nhau.
* Nâng cao khả năng mở rộng: Thiết kế hệ thống của mình để mở rộng dễ dàng trong tương lai.

**TÀI LIỆU THAM KHẢO**

**Tiếng Việt**

1. <https://www.bkns.vn/client-server-la-gi.html>
2. <https://blogchiasekienthuc.com/thu-thuat-internet/tcp-la-gi-va-udp-la-gi.html#2-giao-thuc-udp-la-gi>
3. <https://websitehcm.com/thuat-toan-ma-hoa-khoi-block-cipher/>

**Tiếng Anh**

1. Anderson J.E. (1985), The Relative Inefficiency of Quota, The Cheese Case, *American* *Economic Review*, 75(1), pp. 178-90.
2. Borkakati R. P.,Virmani S. S. (1997), Genetics of thermosensitive genic male sterility in Rice, *Euphytica* 88, pp. 1-7.
3. Boulding K.E. (1955), *Economics Analysis*, Hamish Hamilton, London.
4. Burton G. W. (1988), “Cytoplasmic male-sterility in pearl millet (penni-setum glaucum L.)”, *Agronomic Journal* 50, pp. 230-231.
5. Central Statistical Oraganisation (1995), *Statistical Year Book*, Beijing.
6. FAO (1971), *Agricultural Commodity Projections (1970-1980)*, Vol. II. Rome.
7. Institute of Economics (1988), *Analysis of Expenditure Pattern of Urban Households in* *Vietnam,* Departement pf Economics, Economic Research Report, Hanoi.