

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

BÁO CÁO BÀI THỰC HÀNH THỰC TẬP CƠ SỞ

**Bài 15: Lập trình client/server để trao đổi thông tin an toàn**

# Họ và tên: Nguyễn Huy Quang

# Mã sinh viên: B20DCAT144

**Giảng viên: Nguyễn Hoa Cương**

***Hà Nội – 2023***

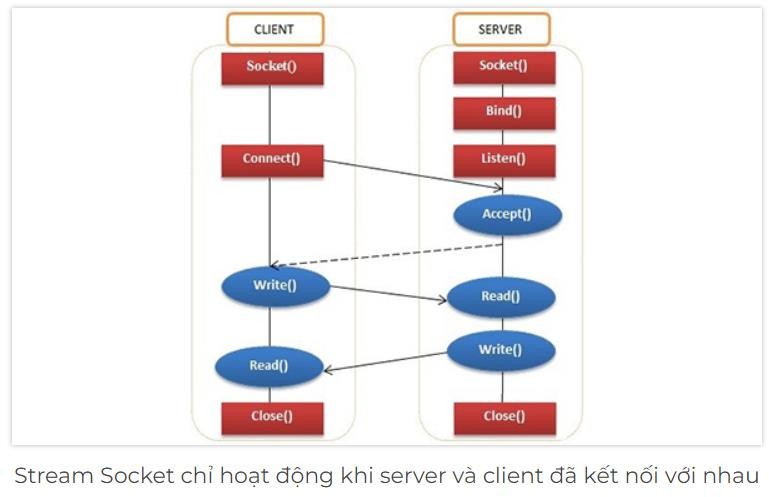
1. **Mục đích**

- Sinh viên hiểu về cơ chế client/server và có thể tự lập trình client/server dựa trên socket, sau đó thực hiện ca đặt giao thức đơn giản để trao đổi thông tin an toàn.

# Nội dung thực hành 2.1.Tìm hiểu lý thuyết

*2.1.1. Tìm hiểu về các khái niệm liên quan tới lập trình socket với TCP*

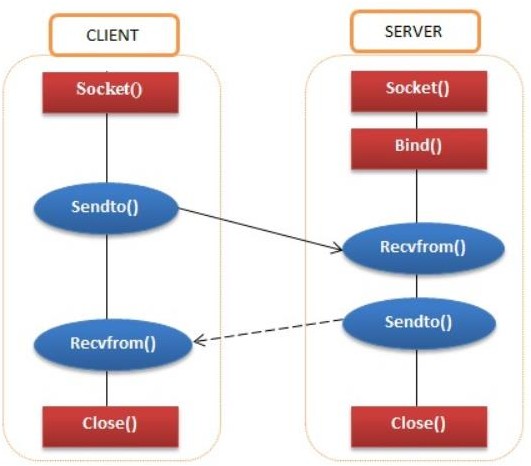
* Socket là giao diện lập trình ứng dụng mạng được dùng để truyền và nhận dữ liệu trên internet. Giữa hai chương trình chạy trên mạng cần có một liên kết giao tiếp hai chiều, hay còn gọi là two-way communication để kết nối 2 process trò chuyện với nhau. Điểm cuối (endpoint) của liên kết này được gọi là socket.
* Một chức năng khác của socket là giúp các tầng **TCP** hoặc **TCP Layer** định danh ứng dụng mà dữ liệu sẽ được gửi tới thông qua sự ràng buộc với một cổng port (thể hiện là một con số cụ thể), từ đó tiến hành kết nối giữa client và server.
* Dựa trên giao thức TCP( Tranmission Control Protocol), stream socket thiết lập giao tiếp 2 chiều theo mô hình client và server. Được gọi là socket hướng kết nối.



* **Stream Socket** hay còn gọi là socket hướng kết nối, là socket hoạt động thông qua **giao thức TCP (Transmission Control Protocol)**. Stream Socket chỉ hoạt động khi server và

client đã kết nối với nhau. Giao thức này đảm bảo dữ liệu được truyền đến nơi nhận một cách đáng tin cậy, đúng tuần tự nhờ vào cơ chế quản lý luồng lưu thông trên mạng và cơ chế chống tắc nghẽn.

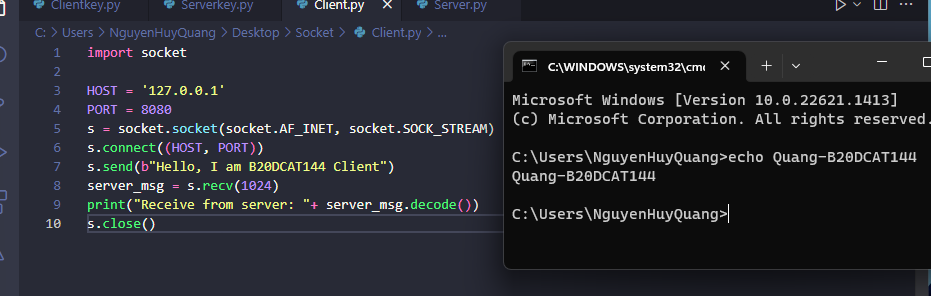
* Dựa trên giao thức UDP( User Datagram Protocol) việc truyền dữ liệu không yêu cầu có sự thiết lập kết nối giữa 2 process. Tức là nó cung cấp connection-less point cho việc gửi và nhận packets. Gọi là socket không hướng kết nối.



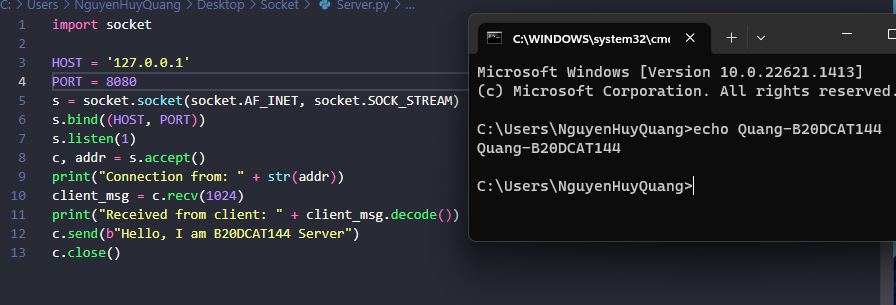
-Do không yêu cầu thiết lập kết nối, không phải có những cơ chế phức tạp. Nên tốc độ giao thức khá nhanh, thuận tiện cho các ứng dụng truyền dữ liệu nhanh như chat, game online…

# Thực hành

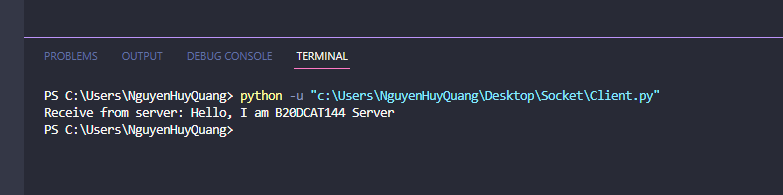
* + 1. *Lập trình client và server với TCP socket*
* Lập trình Client



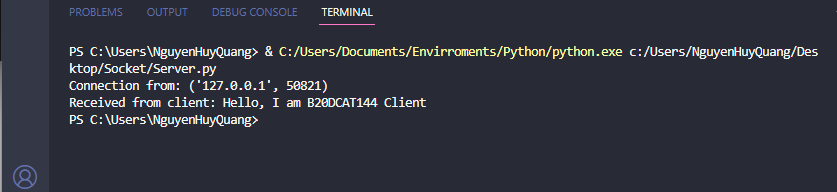
* Lập trình Server



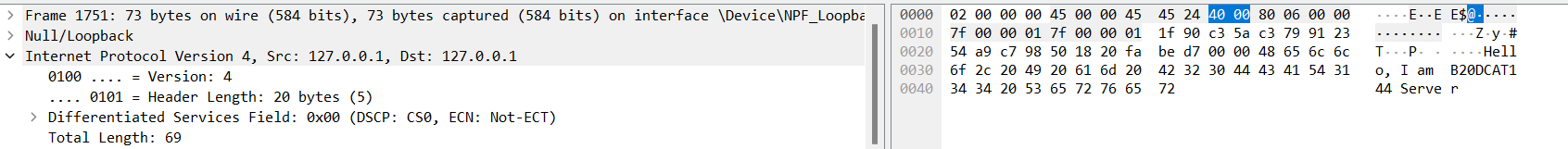
* Chạy Server sau đó chạy Client
* Client gửi thông điệp cá nhân hóa cho Server

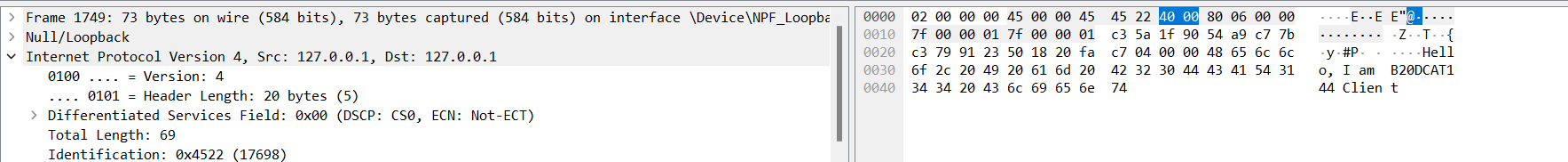


* Server nhận và gửi lại thông điệp cho Client



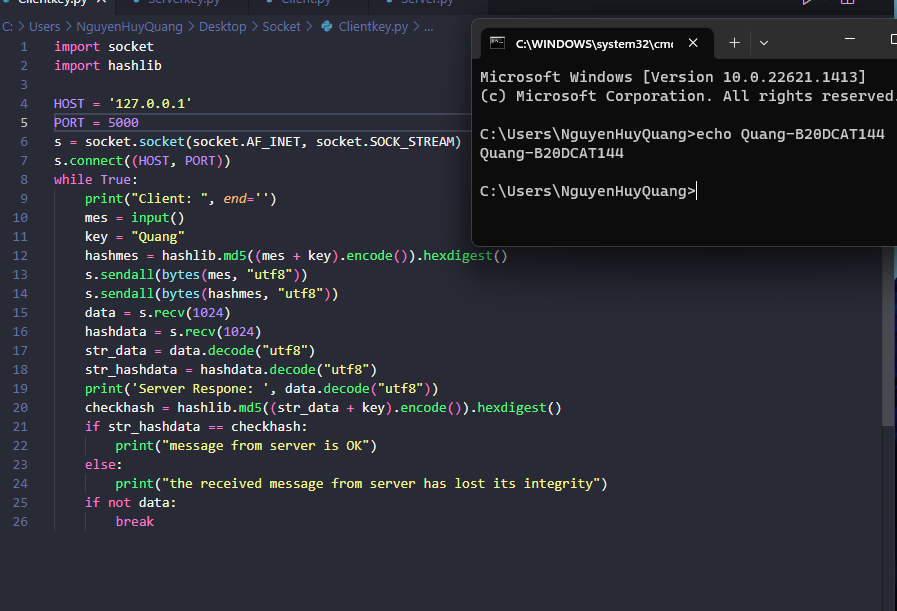
* Sử dụng Wireshark để bắt các thông tin đã gửi từ client đến server và ngược lại



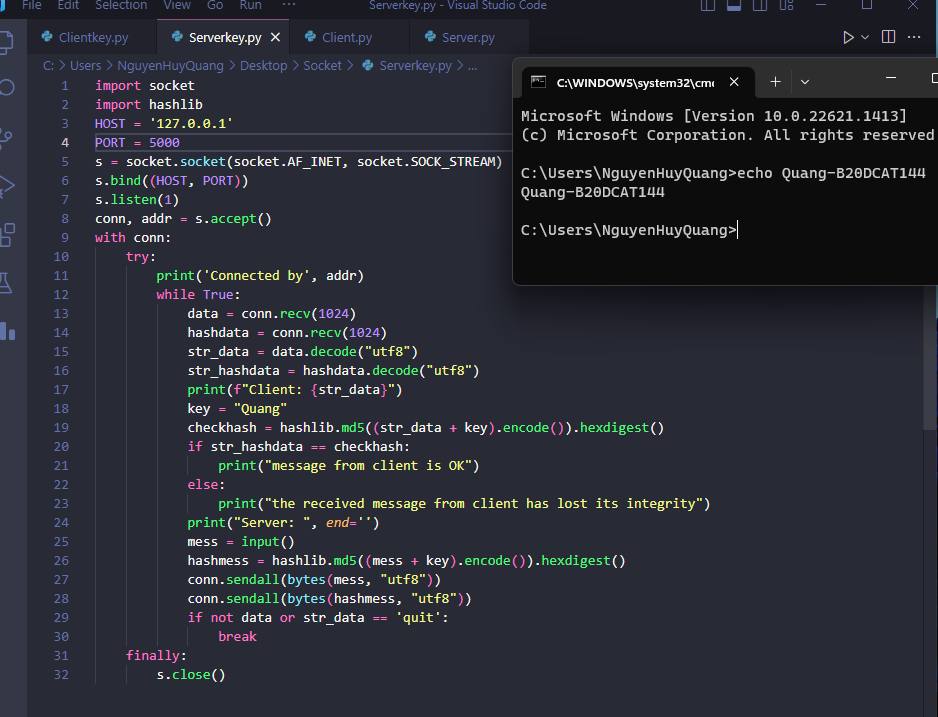


* + 1. *Trao đổi thông điệp giữa Client và Server và đảm bảo tính toàn vẹn của thông điệp khi trao đổi*
* Từ client và server, sửa đổi để sao cho: khi gửi thông điệp sẽ gửi kèm theo giá trị băm của thông điệp + key để phía bên kia kiểm tra xác minh tính toàn vẹn. hai bên có thể thống nhất một giá trị key trước đó

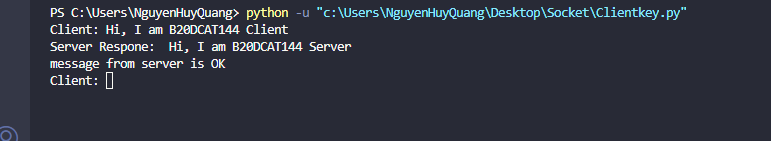
- Sửa code client thống nhất key giữa server và client là “Quang”

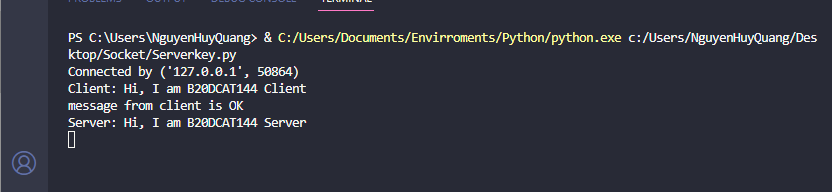


* code server để kiểm tra tính toàn vẹn của thông điệp

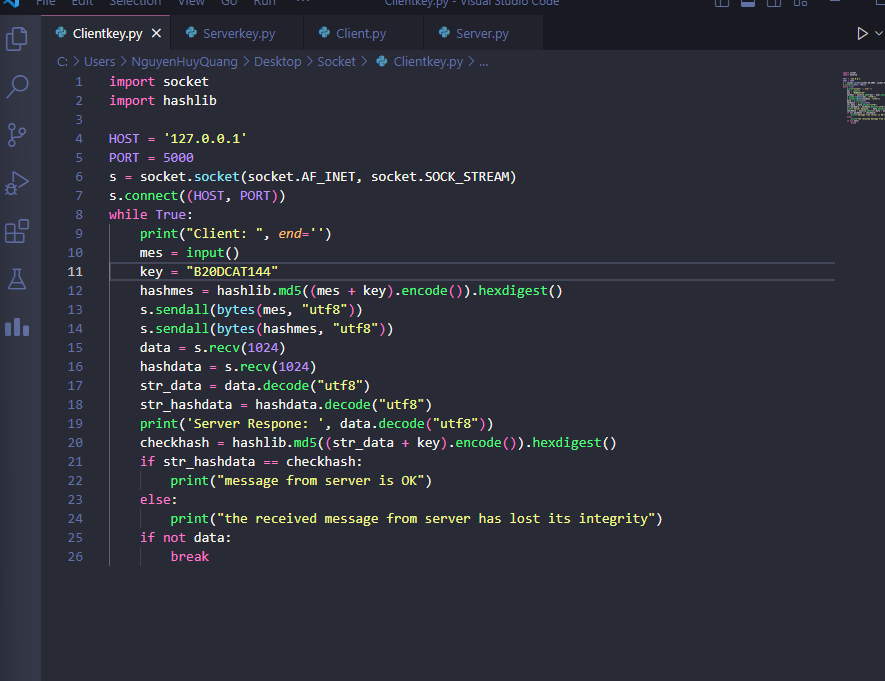


* Trao đổi thông điệp khi chưa đổi key ở client

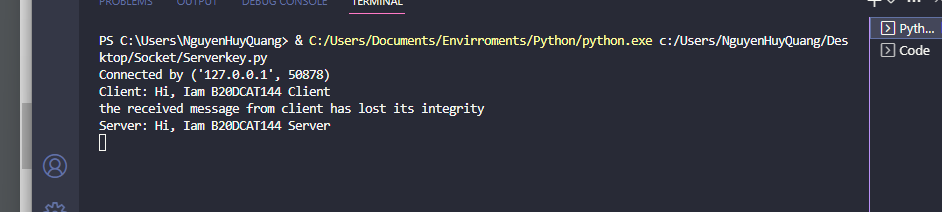




* Thay đổi key ở Client thành “B20DCAT144”



* Trao đổi thông tin



\