**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**KHOA AN TOÀN THÔNG TIN**

****

**MÔN HỌC THỰC TẬP CƠ SỞ**

**BÀI THỰC HÀNH SỐ 15**

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

**Giảng viên hướng dẫn :** Đỗ Xuân Chợ

**Sinh viên thực hiện      :** Hoàng Trung Kiên

**Lớp                                :**  D20CQAT02-B

**Mã sinh viên                    :** B20DCAT098

  Hà nội – 2/2023

**I. Tìm hiểu lý thuyết**

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.

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

**-Socket là gì?** Đây chính là điểm cuối end-point tại liên kết truyền thông 2 chiều (two-way communication) và biểu diễn kết nối giữa Server – Client. Những lớp Socket hiện đang ràng buộc với 1 cổng port (thể hiện là 1 con số cụ thể) để những tầng TCP (hay TCP Layer) hoàn toàn có thể định danh được ứng dụng mà dữ liệu gửi đến. Vậy cụ thể cơ chế hoạt động của Socket là gì?

-Cơ chế hoạt động của Socket là gì? Hiện tại, chức năng của socket chính là kết nối giữa server và client thông qua UDP, TCP/IP để có thể truyền cũng như nhận nhẫn dữ liệu thông qua internet.

-Hiện tại giao diện của lập trình ứng dụng mạng chỉ có thể hoạt động nếu như đã có những thông tin liên quan tới thông số IP cũng như số hiệu cổng của hai ứng dụng cần phải trao đổi dữ liệu.

-Như vậy hai ứng dụng đang cần truyền thông tin bắt buộc phải đáp ứng được những điều kiện cơ bản sau đây thì socket mới hoạt động, cụ thể:

+Hai ứng dụng hoàn toàn có thể nằm cùng trên một máy hay hai máy khác nhau.

+Đối với trường hợp nếu như hai ứng dụng cùng trên một máy thì hiệu số cổng bắt buộc không được trùng với nhau.

**II. Nội dung thực hành**

1. Chuẩn bị môi trường

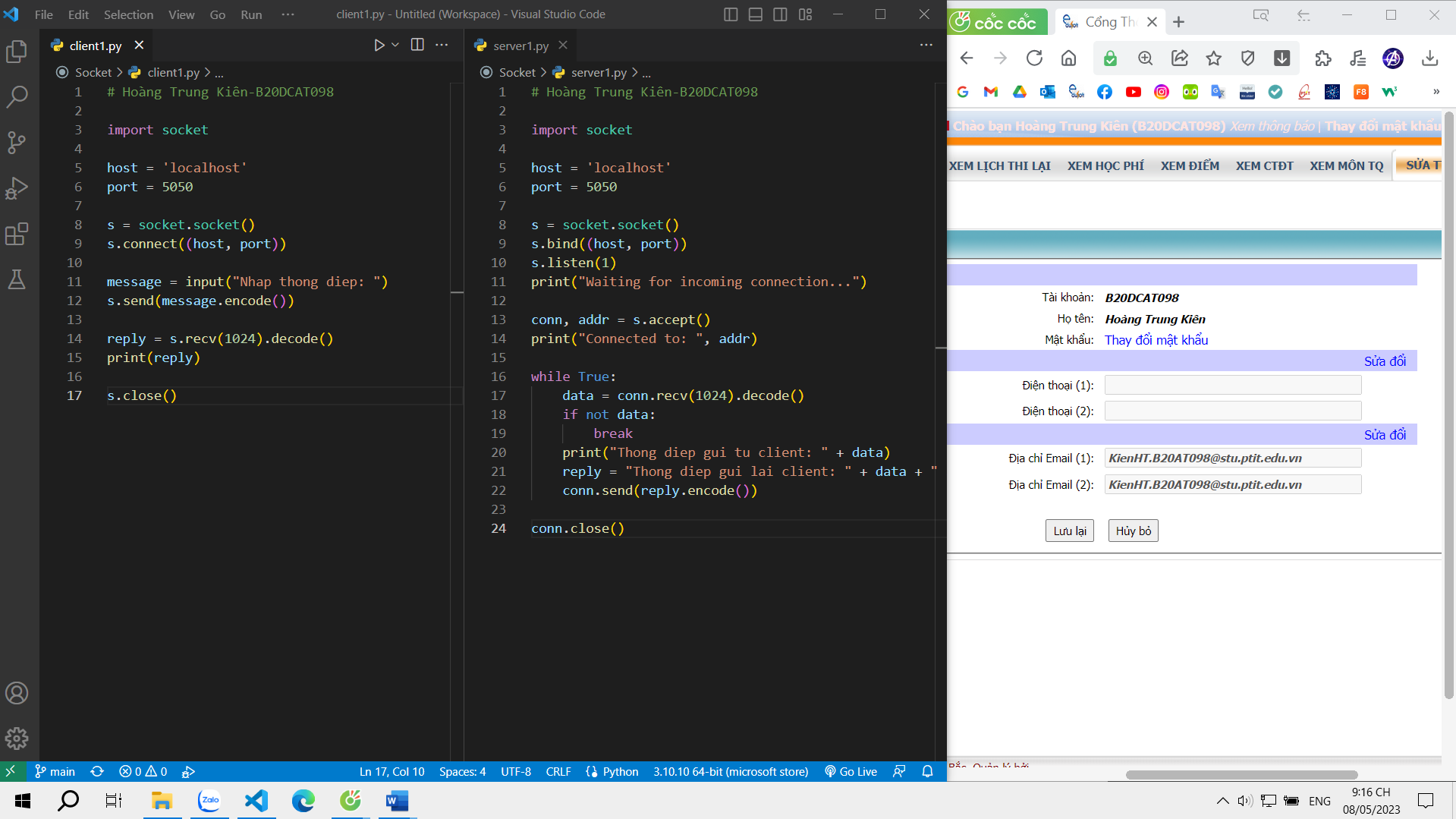
-Môi trường Python hoặc Java để chạy được ứng dụng client/server đã lập

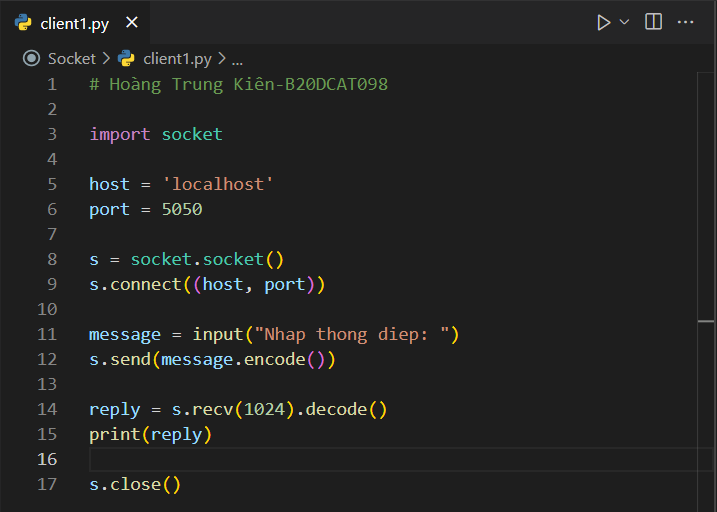
trình.

-Phần mềm Wireshark

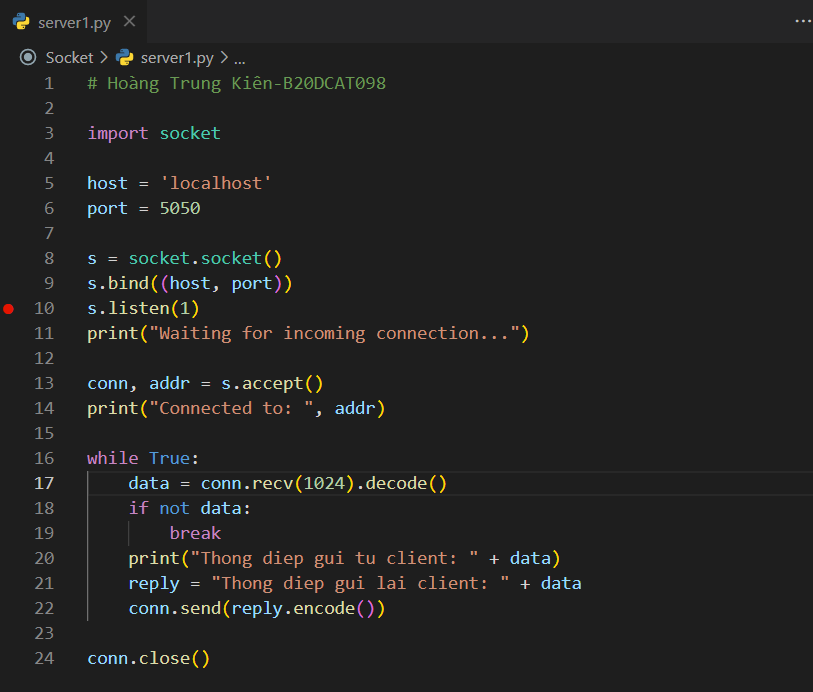
2. Lập trình client và server với TCP socket

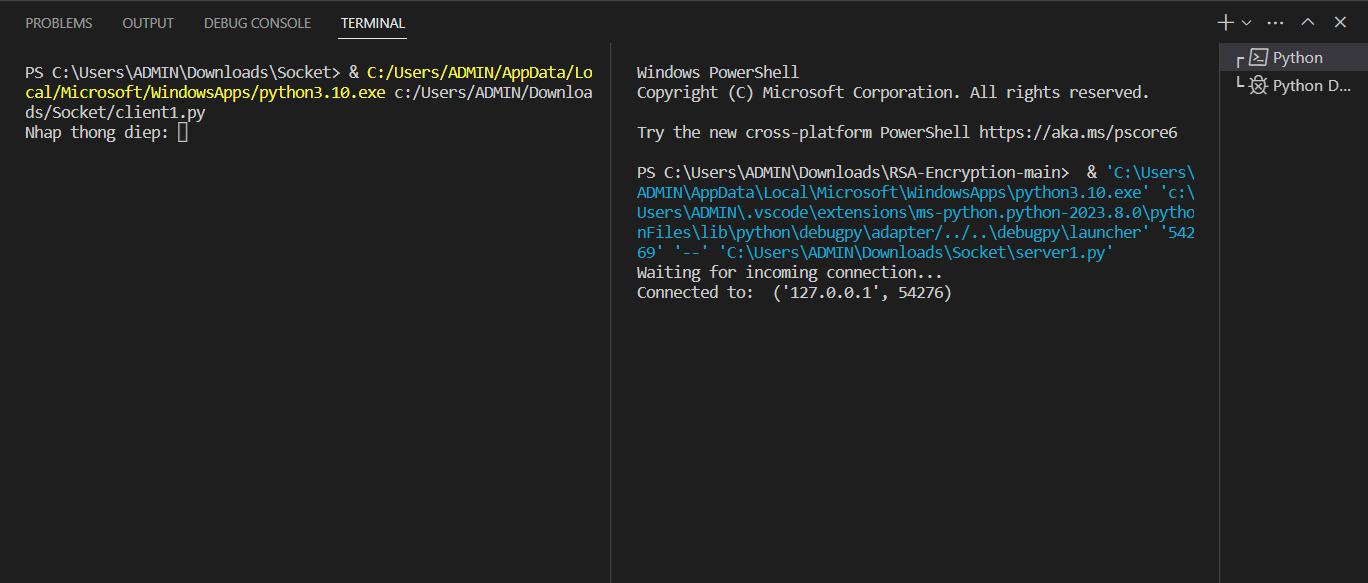
-Các bước thực hiện:

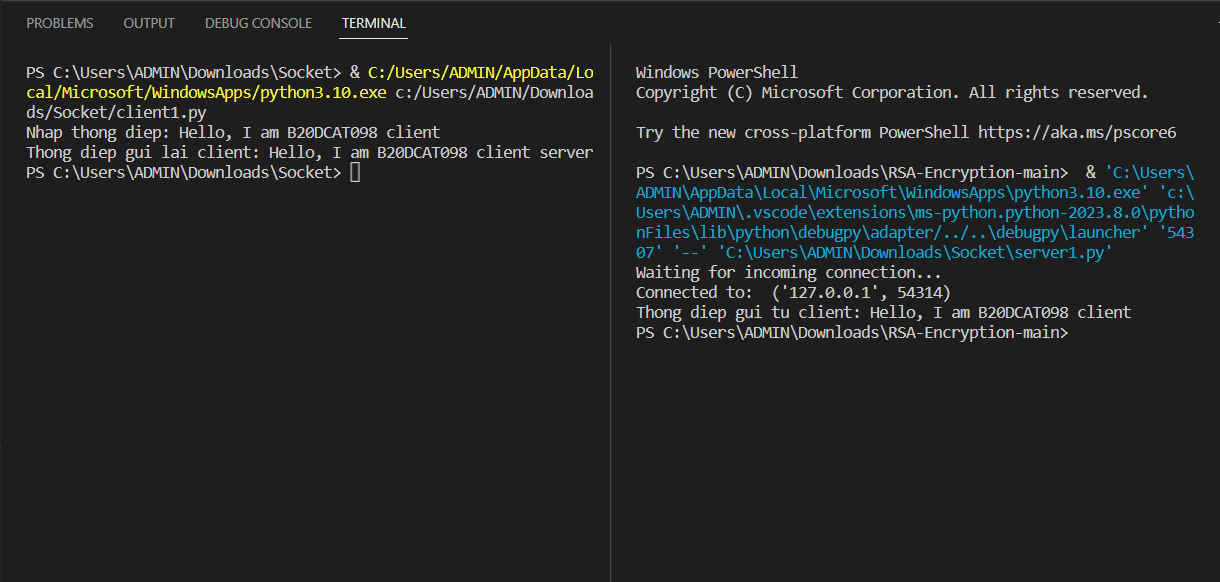


+Lập trình client 

+Lập trình server

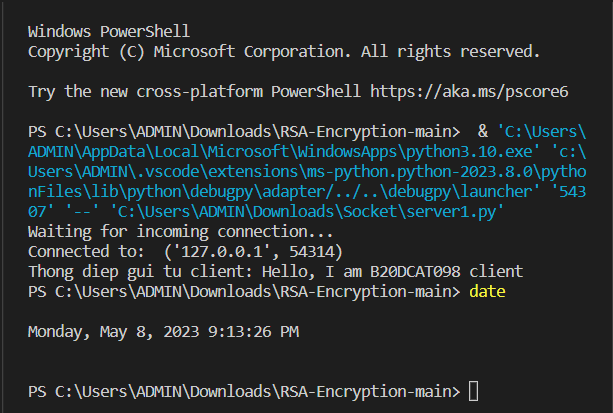






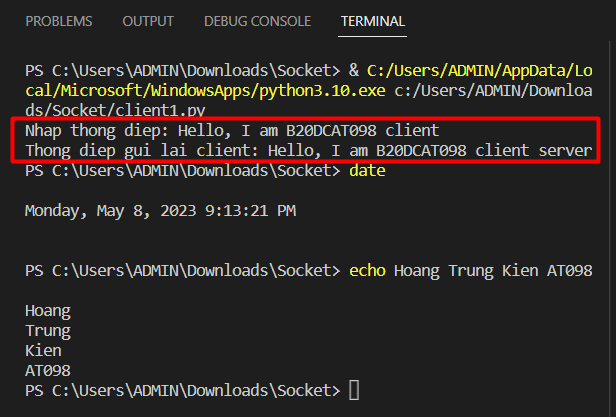
+Client gửi thông điệp cá nhận hóa cho server: “Hello, I am <mã sinh viên>

client.”



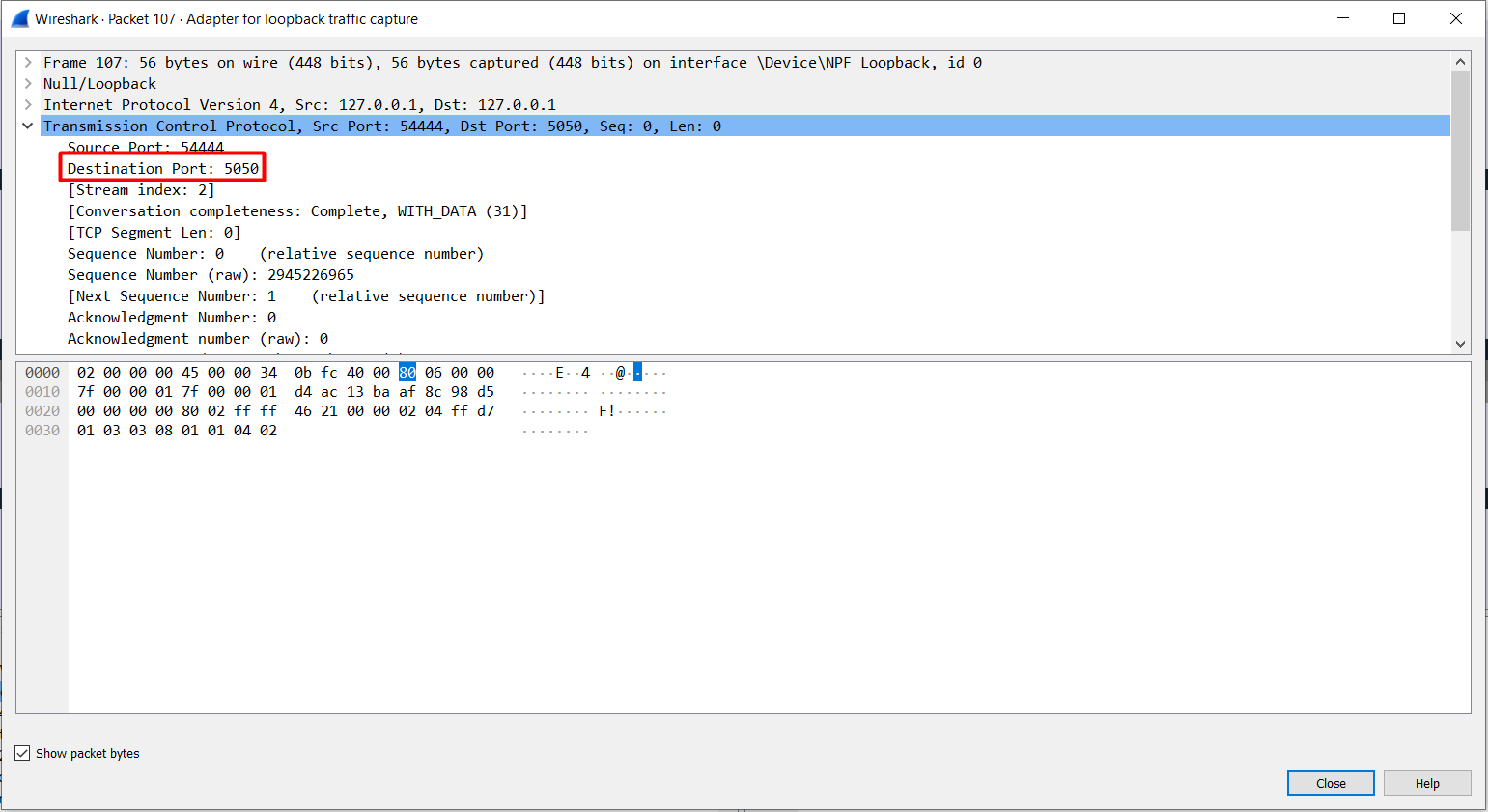
+Server nhận được hiển thị thông điệp nhận được và gửi lại client thông

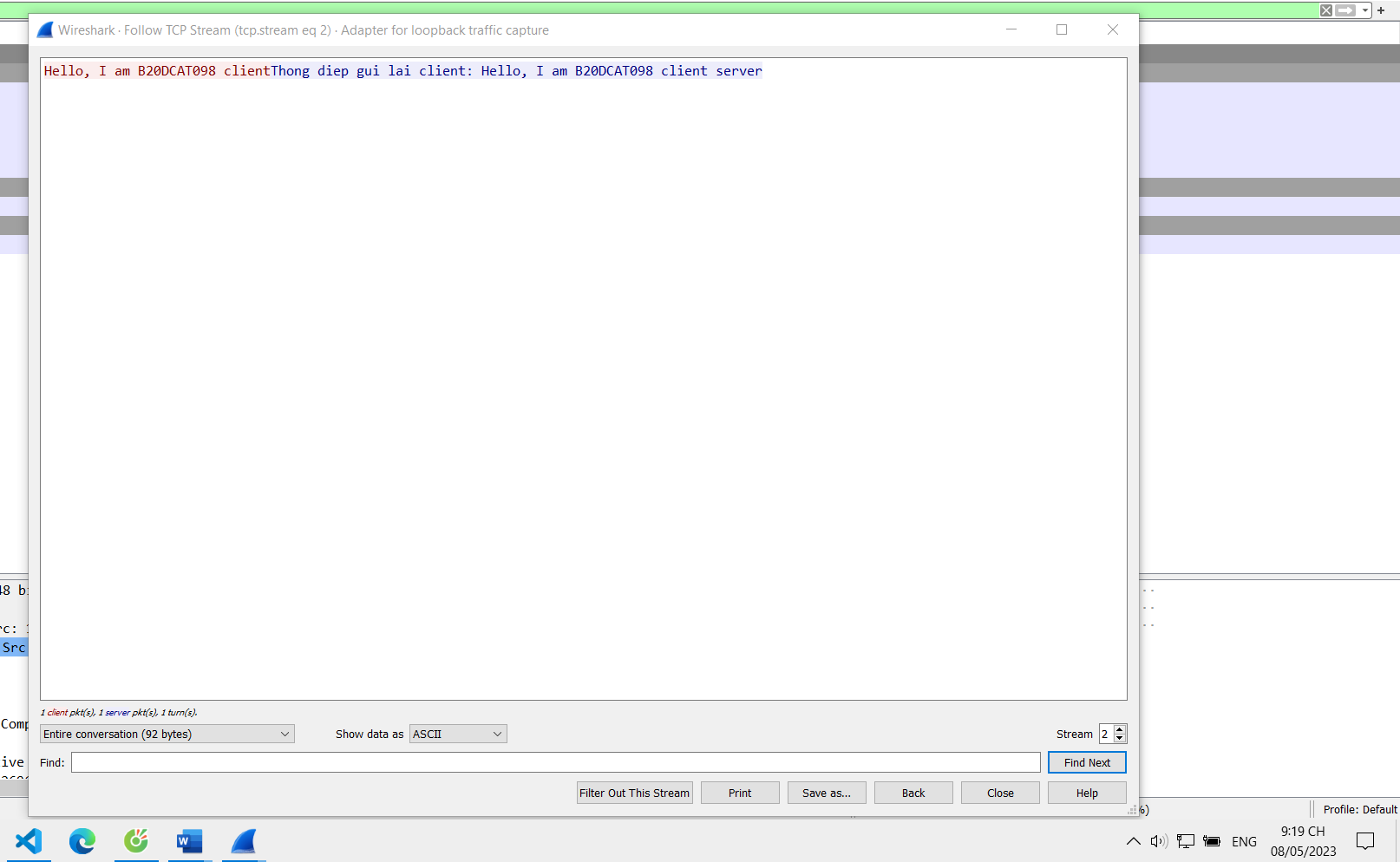
điệp: server gửi lại “Hello, I am <mã sinh viên> server”



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

lại





3. Trao đổi thông điệp giữa client và server và đảm bảo tính toàn vẹn của

thông điệp khi trao đổi.

-Các bước thực hiện

+Từ client và server, sửa đổi để sao cho: khi gửi thông điệp sẽ gửi kèm theo

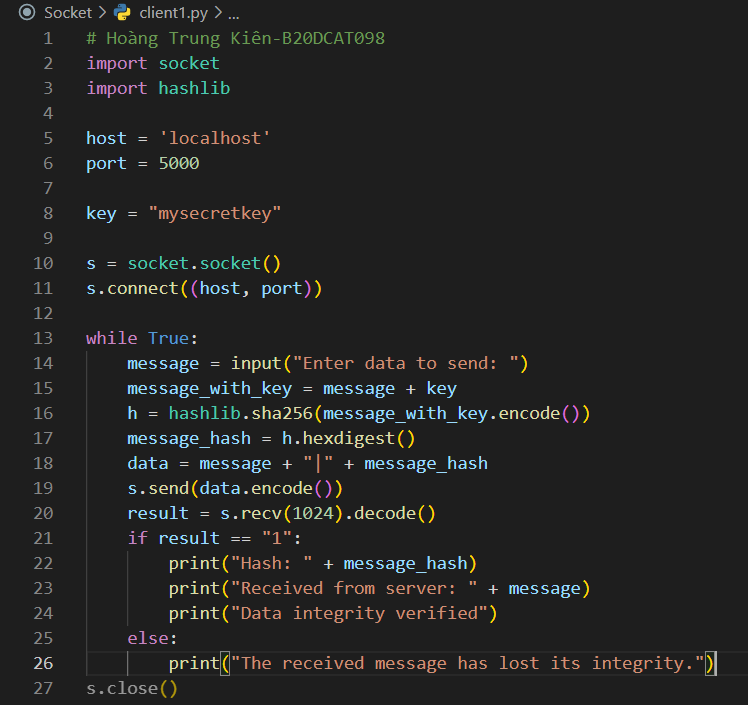
giá trị băm của (thông điệp+key) để phía bên kia kiểm tra xác minh tính

toàn vẹn. Hai bên có thể thống nhất một giá trị key trước đó.

Để sửa đổi client và server để gửi kèm giá trị băm của (thông điệp + key) và kiểm tra tính toàn vẹn, và cho phép client thay đổi giá trị key và phát hiện được tính toàn vẹn của dữ liệu, bạn có thể sử dụng thư viện hashlib để tính toán giá trị băm và sử dụng một giá trị key chung giữa client và server.

Client:

+Key là: mysecretkey

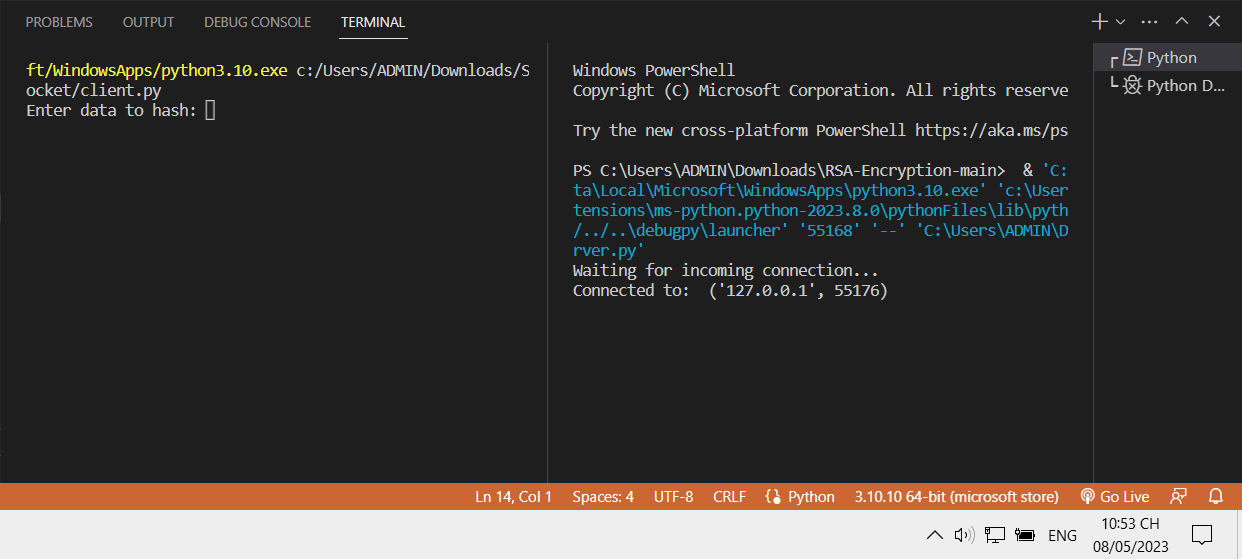


Server:

key: mysecretkey



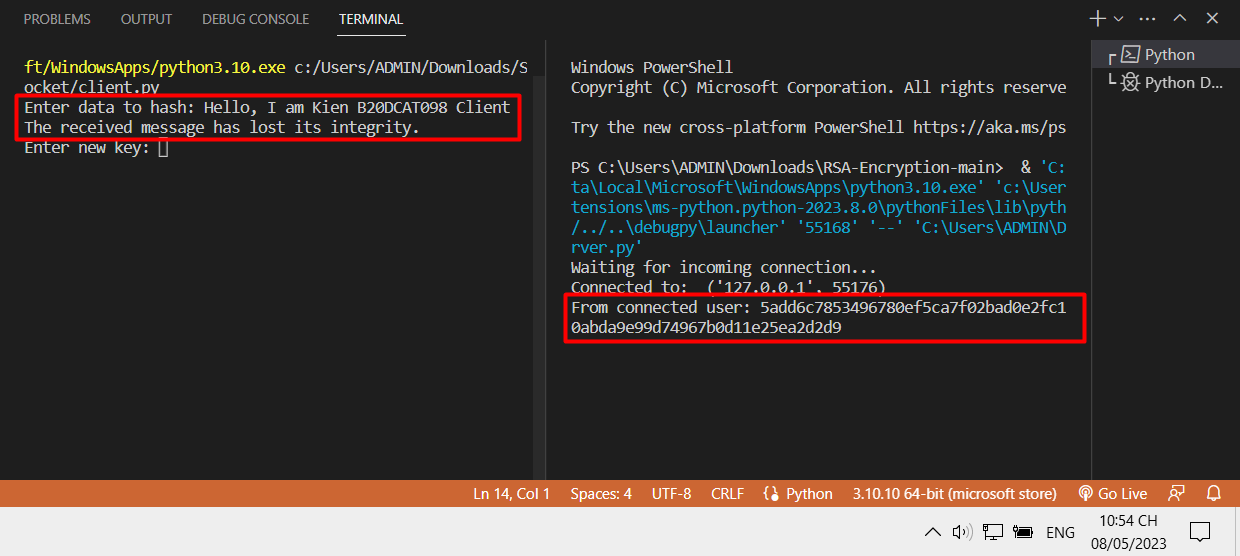
Kết nối thành công



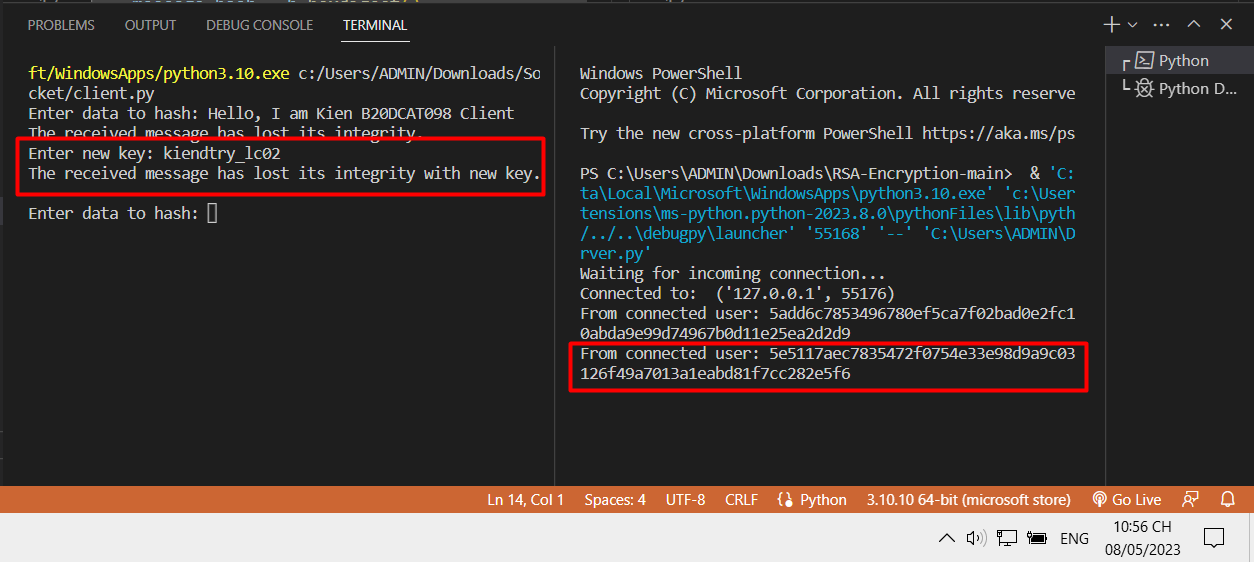
+Trong ví dụ này, server và client đều sử dụng một giá trị key chung để tính toán giá trị băm của (thông điệp + key). Server sẽ gửi lại giá trị băm cho client, và client sẽ kiểm tra tính toàn vẹn của dữ liệu bằng cách so sánh giá trị băm nhận được từ server với giá trị băm của (thông điệp + key) mà nó đã tính toán trước đó.

+Nếu tính toàn vẹn không được đảm bảo, client sẽ in ra thông báo "The received message has lost its integrity."

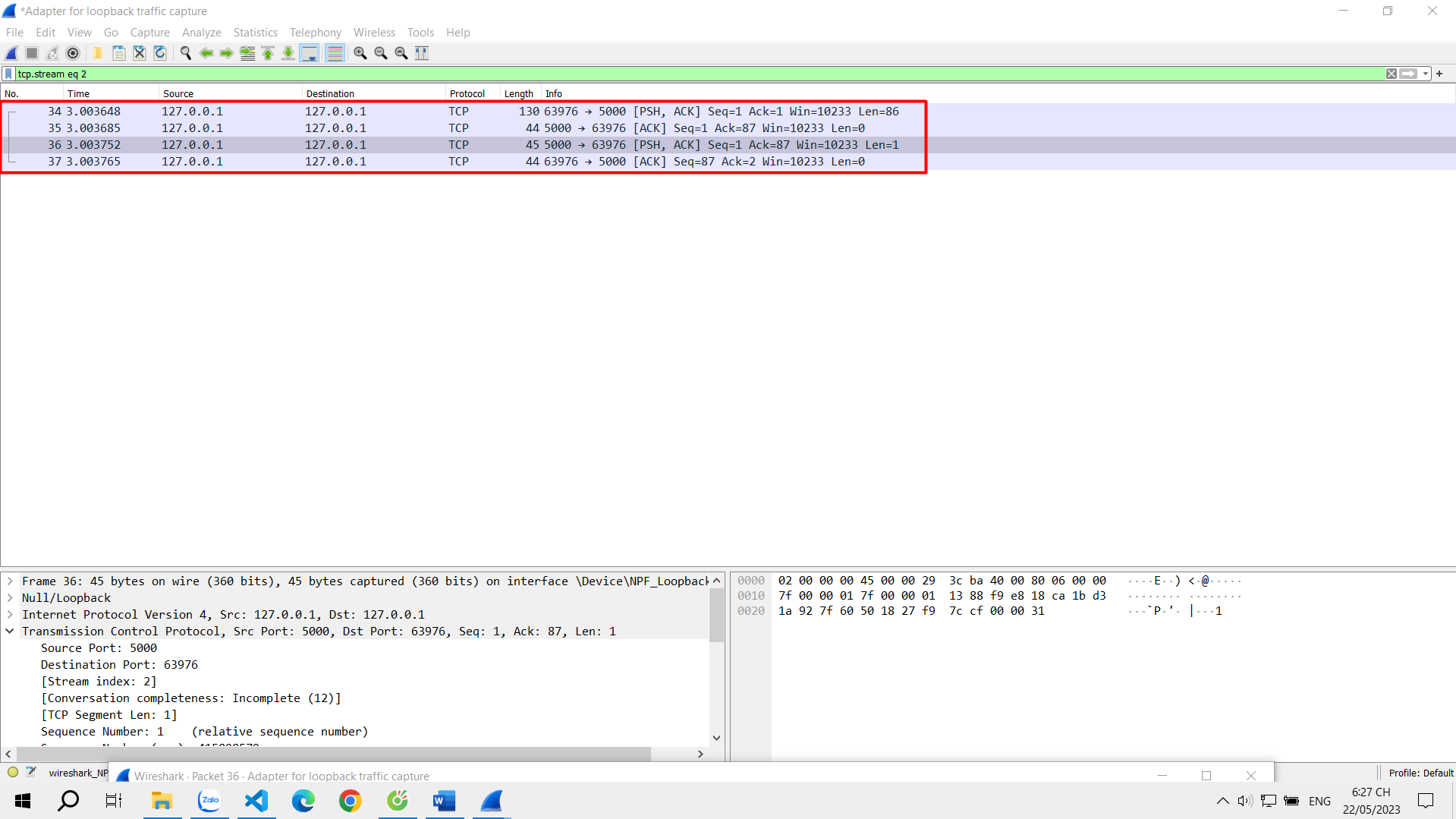
=> Chạy thành công client và server theo mục tiêu ban đầu

  
+Sau đó, client yêu cầu người dùng nhập giá trị key mới và tính toán lại giá trị băm của (thông điệp + key mới). Nó gửi giá trị băm mới đến server và kiểm tra tính toàn vẹn của dữ liệu với key mới.

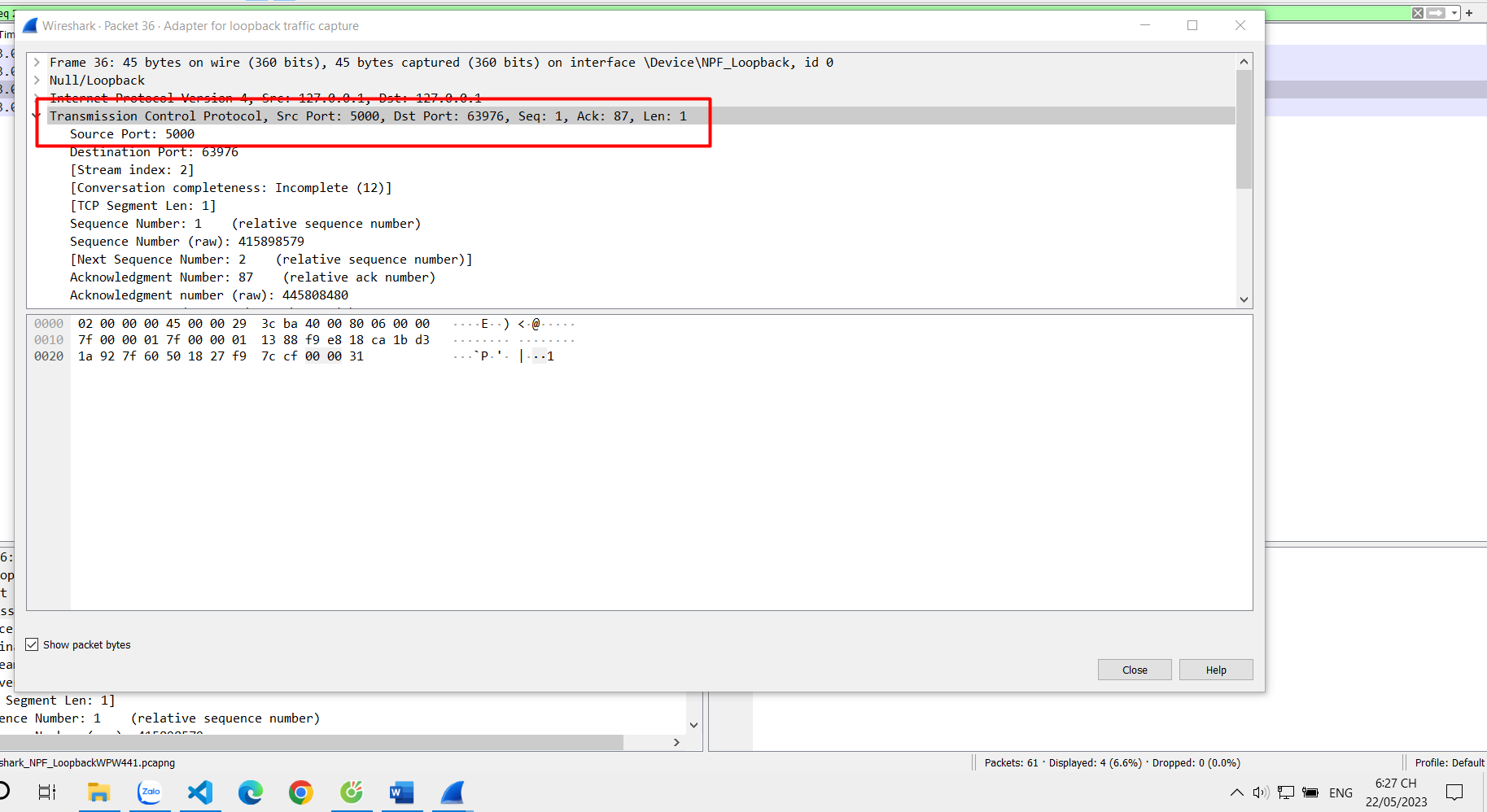
+Nếu tính toàn vẹn được đảm bảo, client sẽ in ra thông báo "Data integrity verified with new key", nếu không, client sẽ in ra thông báo "The received message has lost its integrity with new key."

  
+Với tính năng này, client có thể thay đổi giá trị key và kiểm tra tính toàn vẹn của dữ liệu khi key được thay đổi. Điều này giúp đảm bảo tính toàn vẹn của dữ liệu trong trường hợp giá trị key bị lộ hoặc thay đổi.

+Bắt được các bản tin trao đổi giữa client và server trong Wireshark



Kiểm tra thông tin gói tin



Bắt được các bản tin trao đổi giữa client và server trong Wireshark

