BÁO CÁO BÀI TẬP

**Môn học: AN TOÀN MẠNG**

**PACKET SNIFFING AND SPOOFING LAB**

**Nhóm: 05**

1. **THÔNG TIN CHUNG:**

*(Liệt kê tất cả các thành viên trong nhóm)*

Lớp: NT140.P11.ANTT

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Họ và tên | MSSV | Email |
| 1 | Hồ Vỉ Khánh | 22520633 | 22520633@gm.uit.edu.vn |
| 2 | Trần Anh Khôi | 22520701 | 22520701@gm.uit.edu.vn |
| 3 | Nguyễn Hồ Nhật Khoa | 22520677 | 22520677@gm.uit.edu.vn |
| 4 | Diệp Tấn Phát | 22521066 | 22521066@gm.uit.edu.vn |

1. **NỘI DUNG THỰC HIỆN:[[1]](#footnote-2)**

|  |  |  |
| --- | --- | --- |
| STT | Nội dung | Tình trạng |
| 1 | Task 1.1 | 100% |
| 2 | Task 1.2 | 100% |
| 3 | Task 1.3 | 100% |
| 4 | Task 1.4 | 100% |
| 5 | Task 2.1 | 100% |
| 6 | Task 2.2 | 100% |
| 7 | Task 2.3 | 100% |
| Điểm tự đánh giá 10/10 | | |

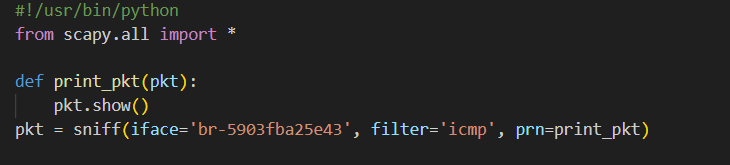
**Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.**

BÁO CÁO CHI TIẾT

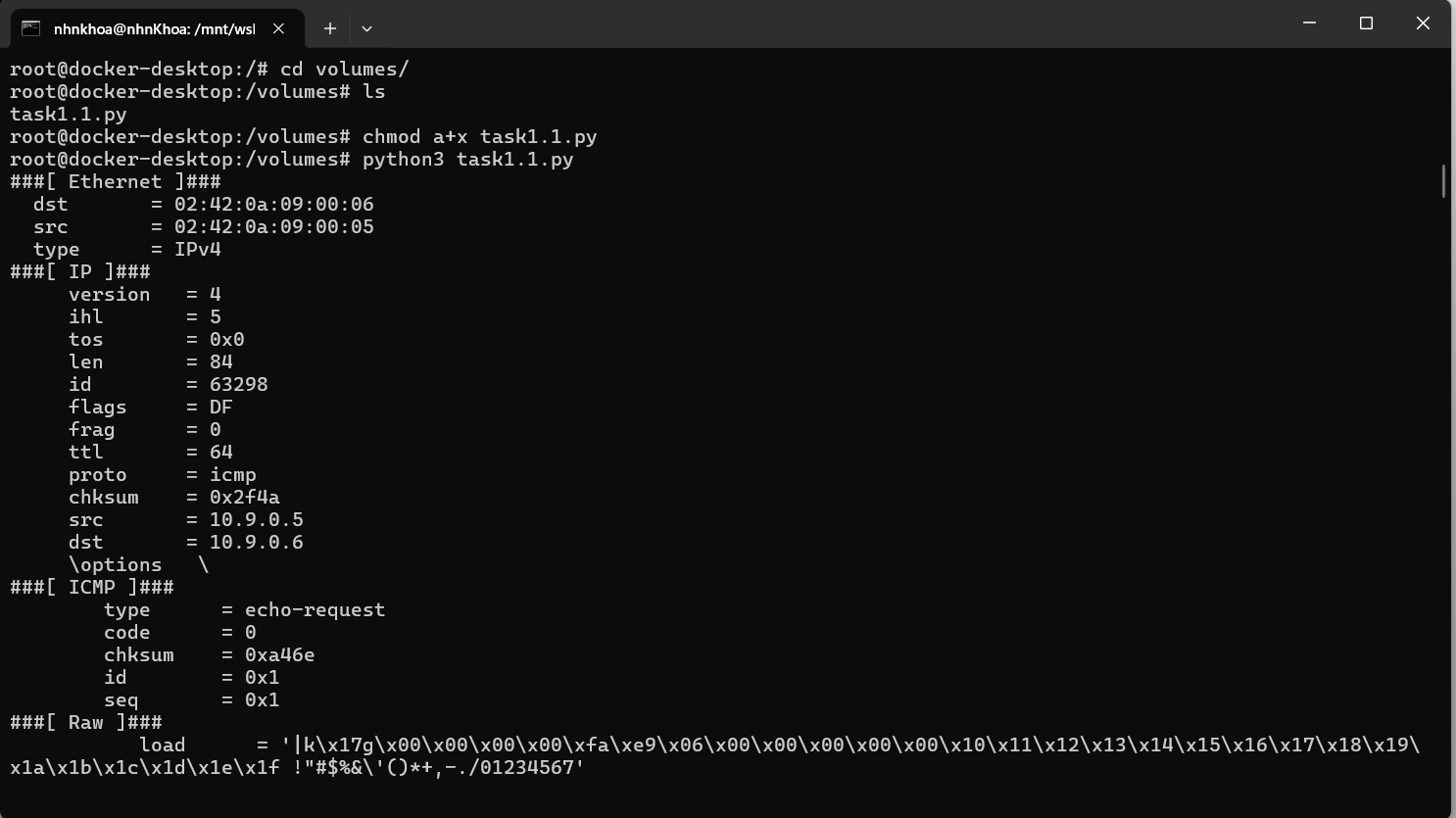
# Lab Task Set 1: Using Scapy to Sniff and Spoof Packets

## Task 1.1.A: In the above program, for each captured packet, the callback function print pkt() will be invoked; this function will print out some of the information about the packet. Run the program with the root privilege and demonstrate that you can indeed capture packets. After that, run the program again, but without using the root privilege; describe and explain your observations.

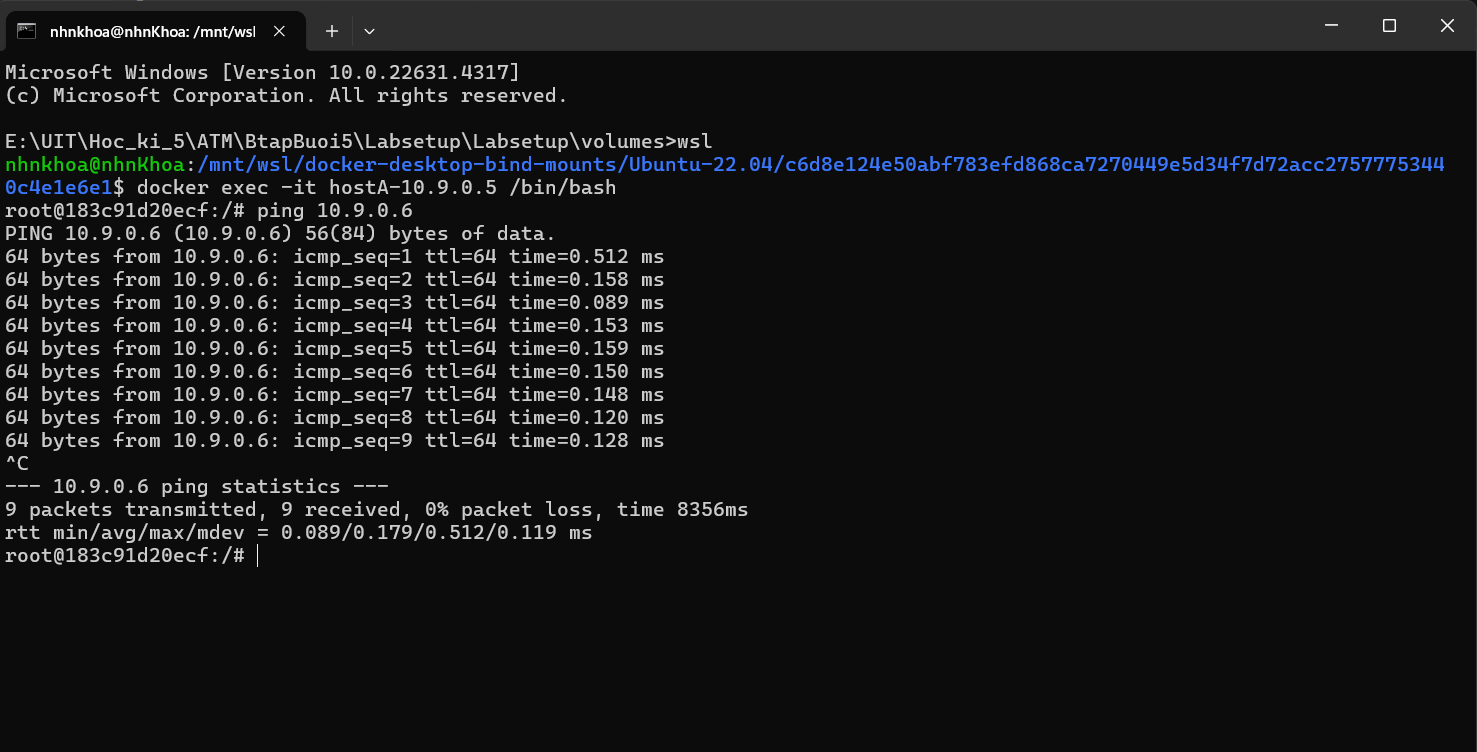
- Code python dùng để bắt các gói tin ICMP:



- Chạy code với quyền root của trên máy seed-attacker



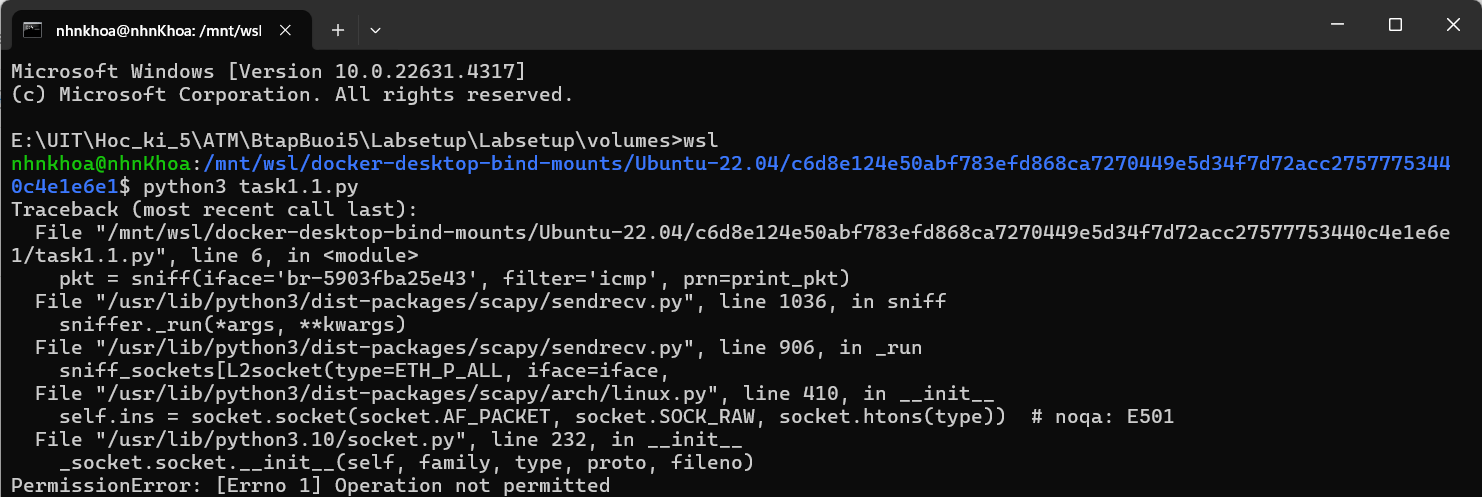
- Thực hiện ping từ host A đến host B



- Ta có thể nhìn thấy một số thông tin của các gói tin này trên máy seed-attacker khi chạy code bắt các gói tin ICMP



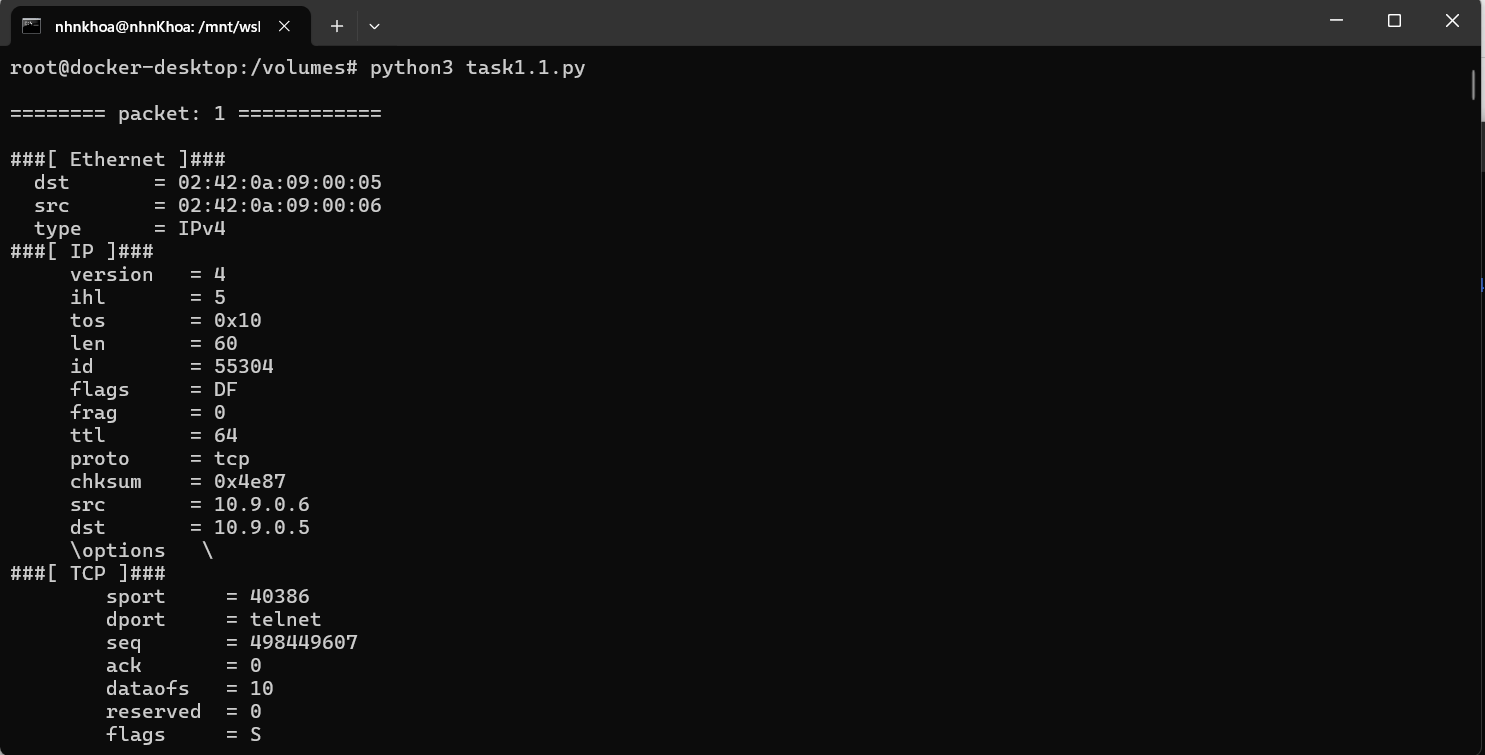
- Chạy đoạn code python trên nhưng không sử dụng quyền root:

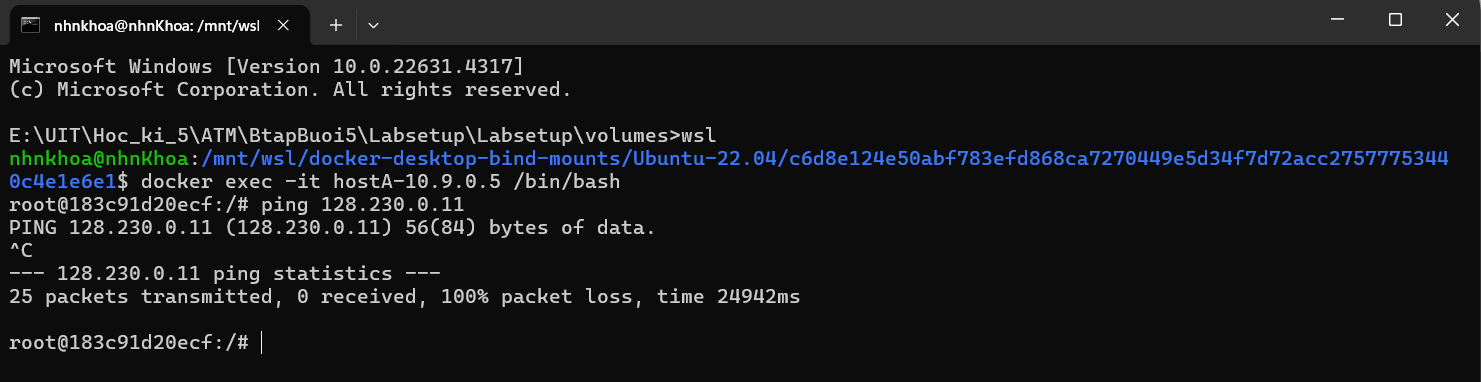
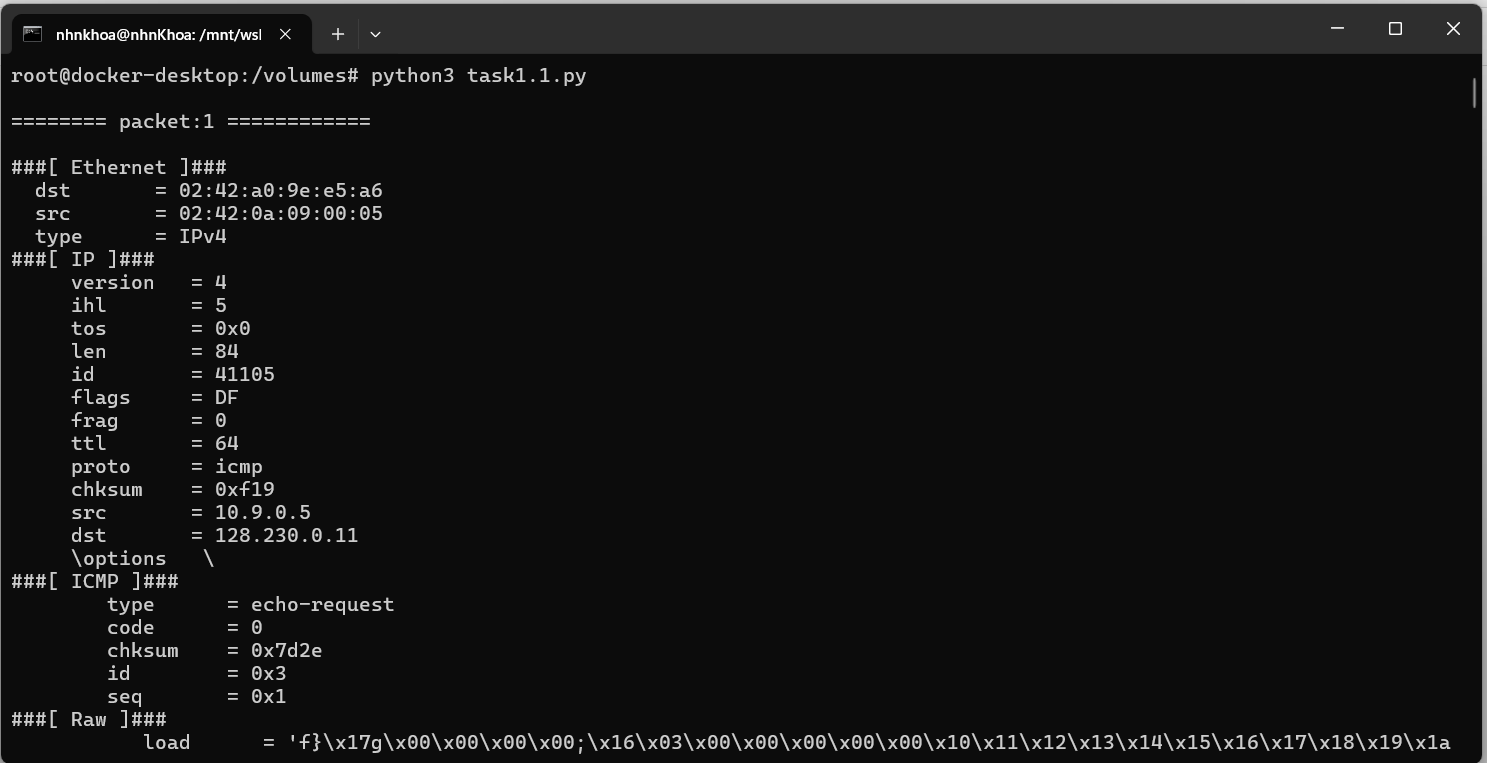


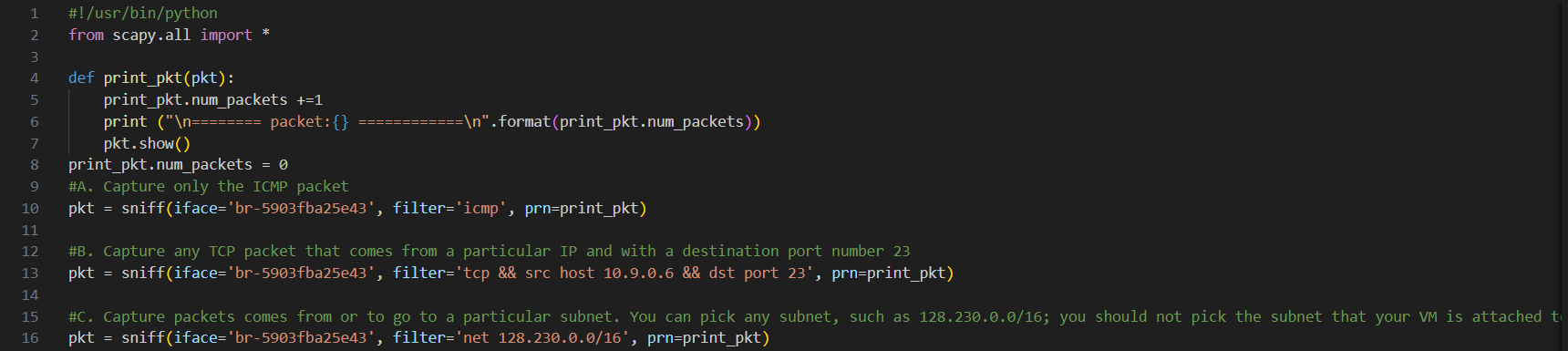
- Ta gặp lỗi “Operation not permitted”, vì vậy khi muốn chặn bắt các gói tin chúng ta cần có quyền root để có thể xem lưu lượng và chặn bắt các gói tin liên quan.

## Task 1.1.B. Usually, when we sniff packets, we are only interested certain types of packets. We can do that by setting filters in sniffing. Scapy’s filter use the BPF (Berkeley Packet Filter) syntax; you can find the BPF manual from the Internet. Please set the following filters and demonstrate your sniffer program again (each filter should be set separately):

- Thực hiện chạy code python để bắt gói tin TCP ở Port 23 của host 10.9.0.6

  
 - Thực hiện chạy code python để bắt các gói tin khi host 10.9.0.5 ping tới một địa chỉ ip bất kì thuộc mạng 128.230.0.0/16

- Đoạn code python được sử dụng trong bài tập này:

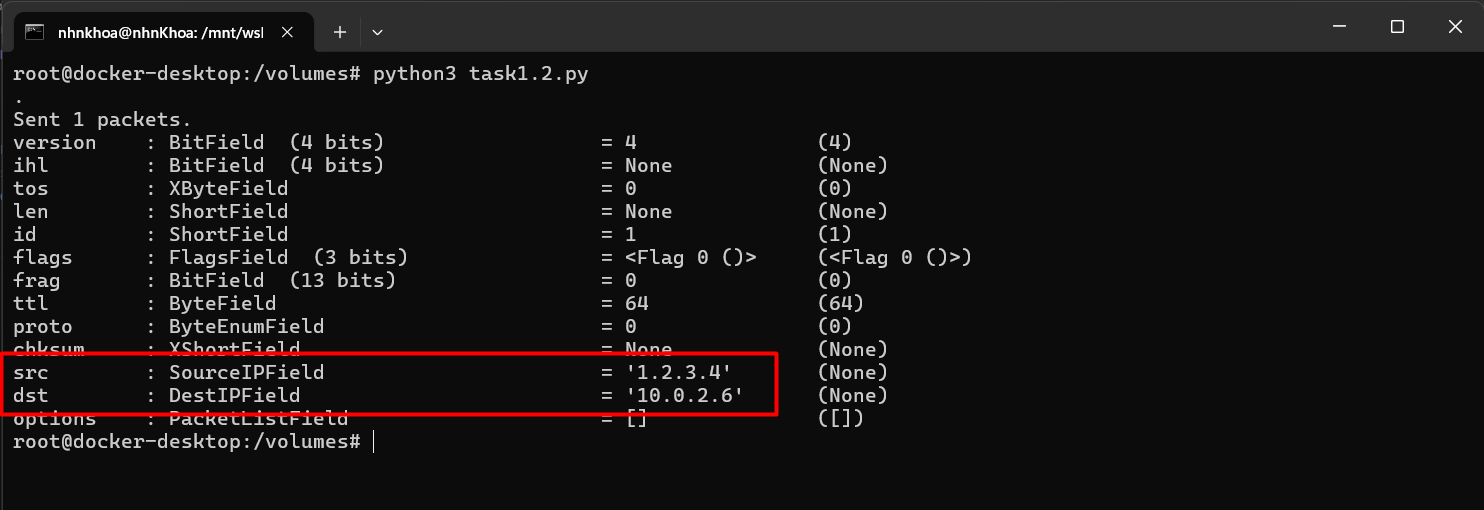


## Task 1.2: Spoofing ICMP Packets

- Đoạn code bên dưới được dùng để gửi một gói tin ICMP từ một địa chỉ IP giả mạo đến một địa chỉ IP bất kì



- Thực thi đoạn code:

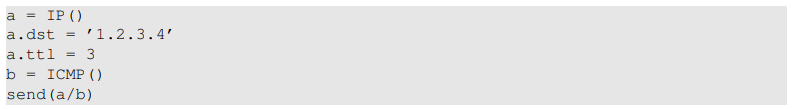


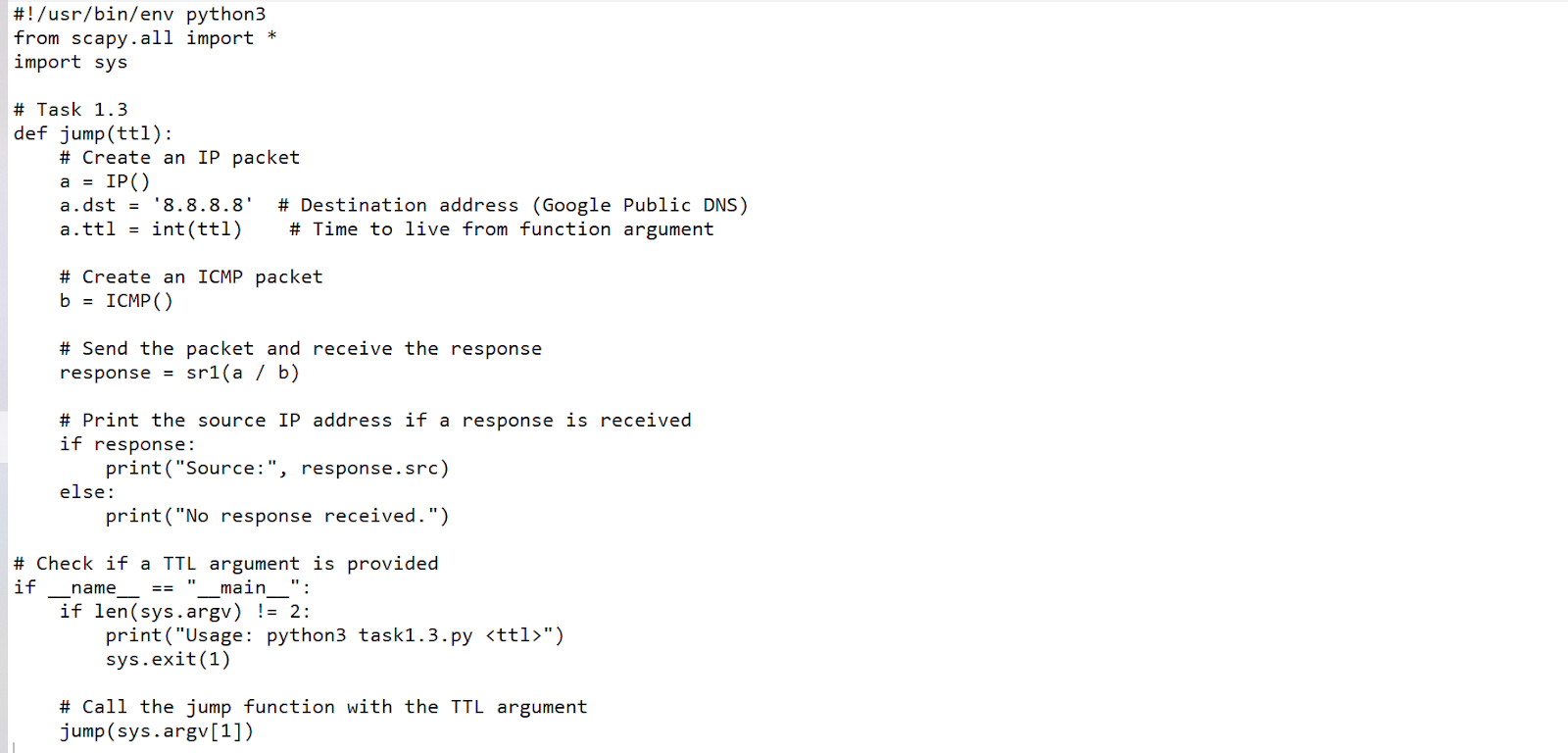
## Task 1.3: Traceroute

- Nhiệm vụ của task này là sử dụng Snapy để ước tính khoảng cách, theo số lượng định tuyến, giữa VM của ta và máy đích đã chọn

- Trong bài này sẽ chọn VM và host A

- Đoạn code demo ban đầu:

- Code công cụ được tạo:



- Đoạn mã Python trên sử dụng thư viện Scapy để tạo và gửi một gói tin ICMP tới địa chỉ IP 8.8.8.8 (DNS server của Google). Dưới đây là chức năng của từng phần trong mã:

* Mô-đun sys được sử dụng để truy cập các tham số dòng lệnh. Trong trường hợp này, nó được sử dụng để lấy giá trị TTL (Time to Live) từ dòng lệnh.
* Hàm jump nhận một tham số ttl, đại diện cho giá trị Time to Live của gói tin IP. Hàm này sẽ thực hiện tất cả các bước để tạo và gửi gói tin ICMP.
* Tạo một gói tin IP mới (a) với địa chỉ đích là 8.8.4.4 và TTL được lấy từ tham số dòng lệnh (biến ttl). TTL xác định số lượng bước mà gói tin có thể đi qua trước khi bị loại bỏ.
* Tạo một gói ICMP (b). ICMP được sử dụng để gửi thông báo lỗi và kiểm tra kết nối mạng (chẳng hạn như trong lệnh ping).
* Kết hợp gói IP và gói ICMP, sau đó gửi gói tin đi bằng cách sử dụng hàm sr1. Hàm này gửi gói tin và chờ nhận phản hồi đầu tiên từ gói đã gửi.
* In ra địa chỉ IP nguồn của gói tin phản hồi mà đã được nhận. Nếu gói tin đã gửi không nhận được phản hồi, a sẽ là None, và sẽ gây ra lỗi khi cố gắng truy cập thuộc tính src.

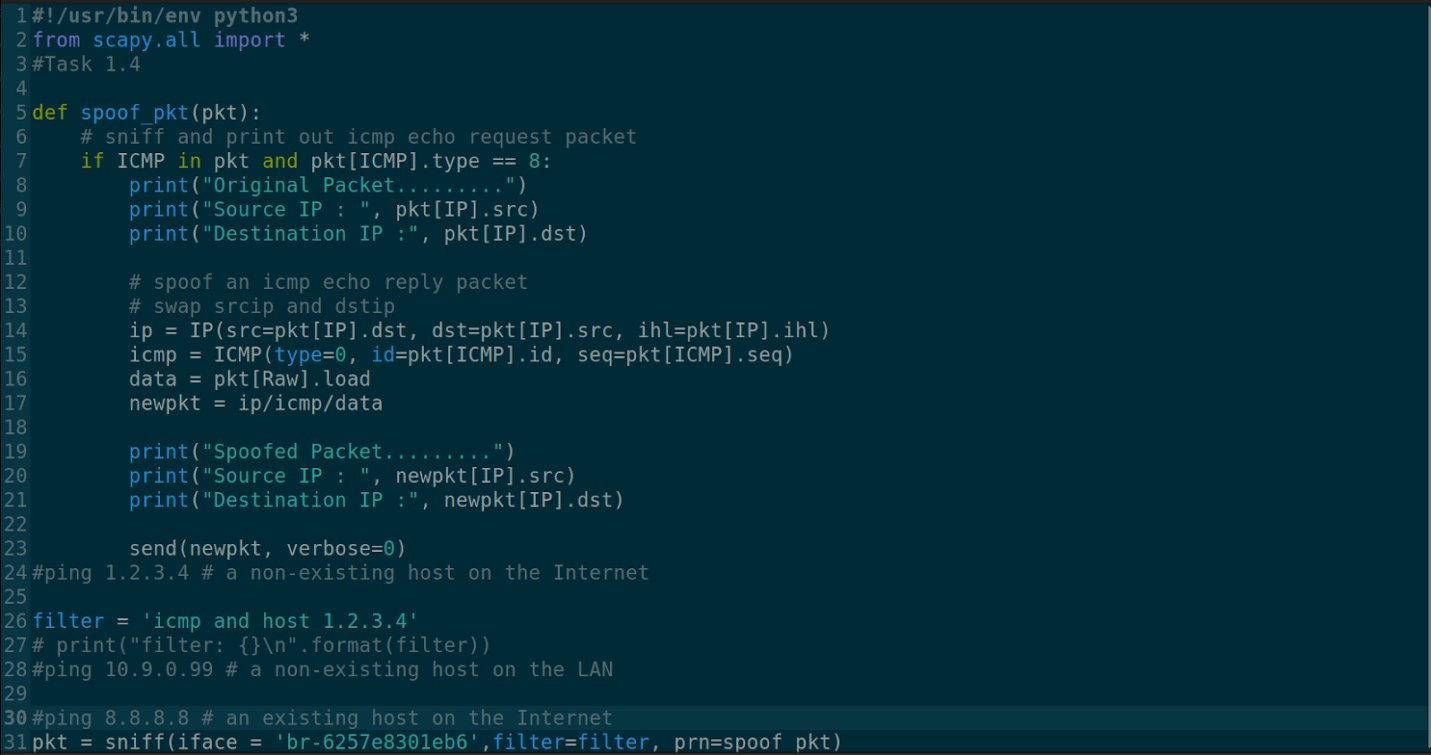
**Chức năng tổng quát:** Chức năng tổng quát của mã này là gửi một gói tin ICMP tới một địa chỉ IP cụ thể (trong trường hợp này là Google DNS) với TTL được chỉ định và sau đó in ra địa chỉ IP nguồn của gói tin phản hồi. Đây có thể được xem như là một cách để kiểm tra kết nối mạng đến một địa chỉ cụ thể, tương tự như lệnh ping trong hệ điều hành.

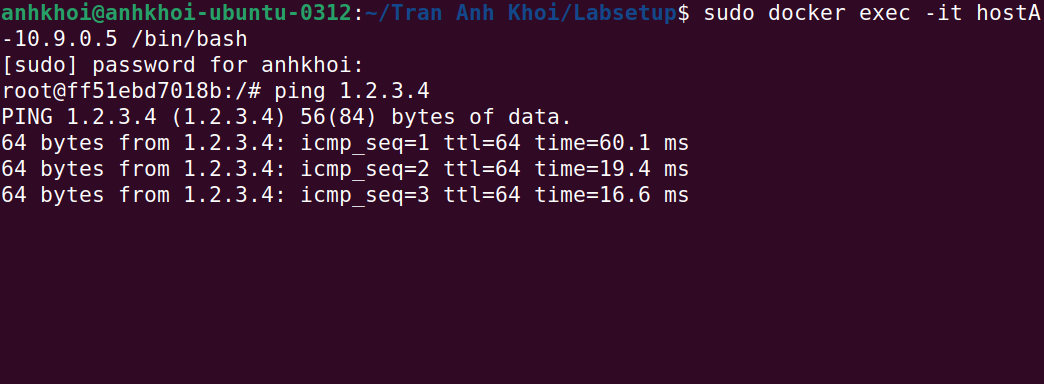
## Task 1.4: Sniffing and-then Spoofing

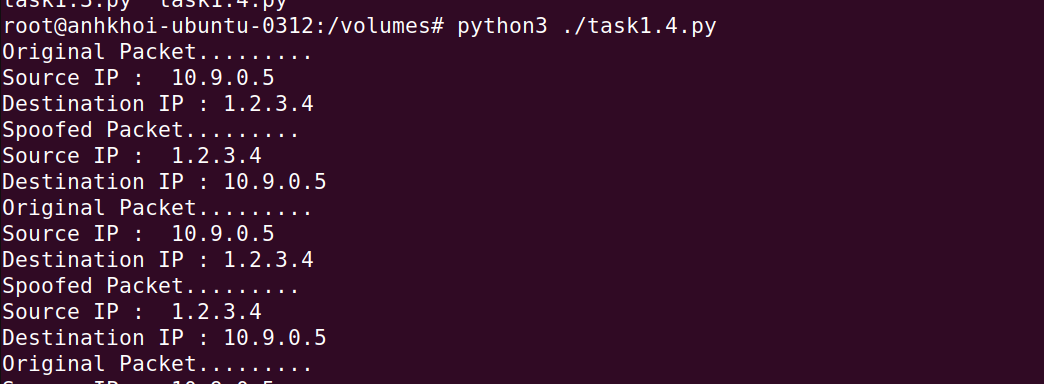
- Bài này yêu cầu bạn kết hợp các kỹ thuật sniffing (nghe lén gói tin) và spoofing (giả mạo gói tin) để triển khai chương trình.

- Từ container người dùng, ta thực hiện lệnh ping đến một địa chỉ IP X. Điều này sẽ tạo ra gói tin yêu cầu ICMP echo. Nếu IP X còn hoạt động, chương trình ping sẽ nhận được phản hồi và hiển thị kết quả. Chương trình sniff-and-then-spoof chạy trên VM, giám sát mạng LAN bằng cách nghe lén gói tin.

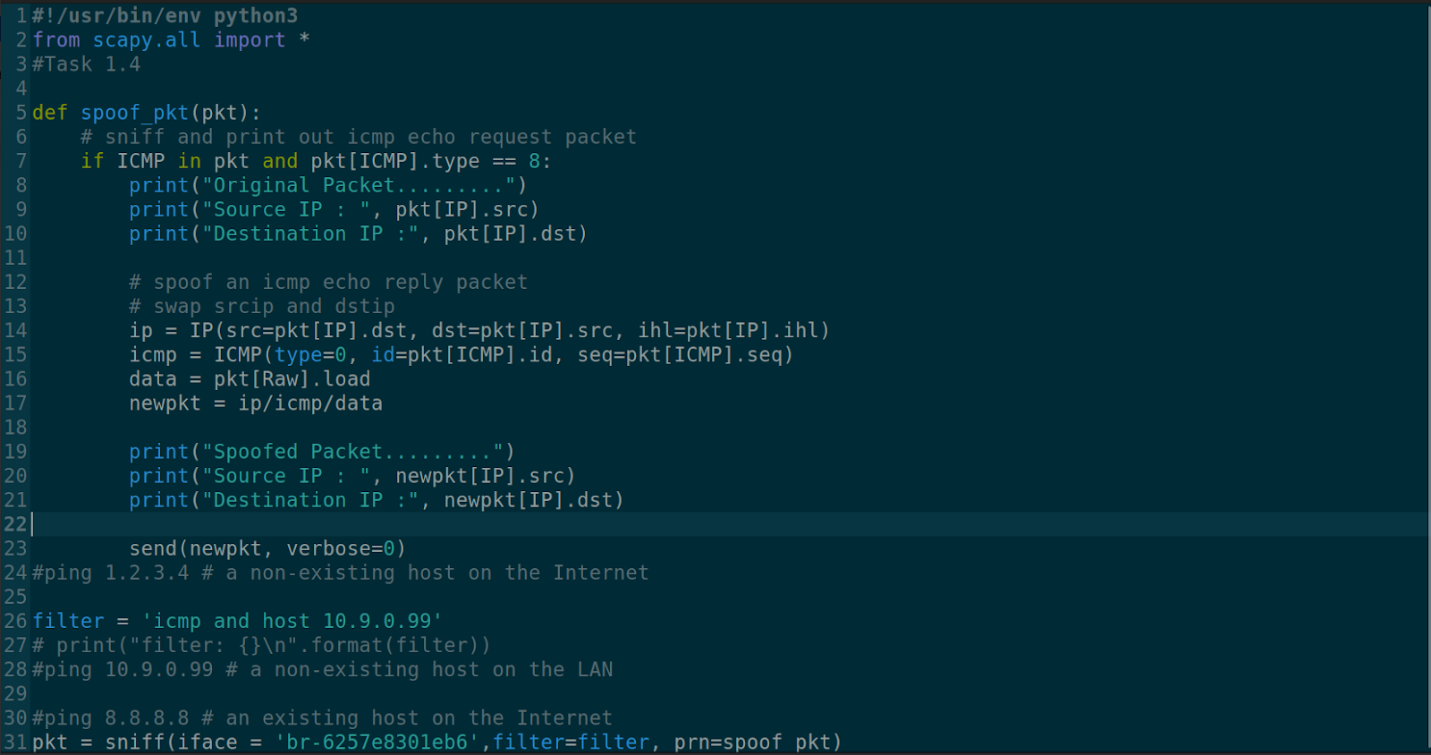
- Bắt đầu với địa chỉ ip 1.2.3.4

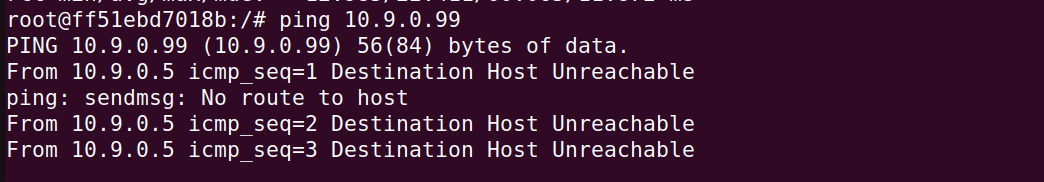


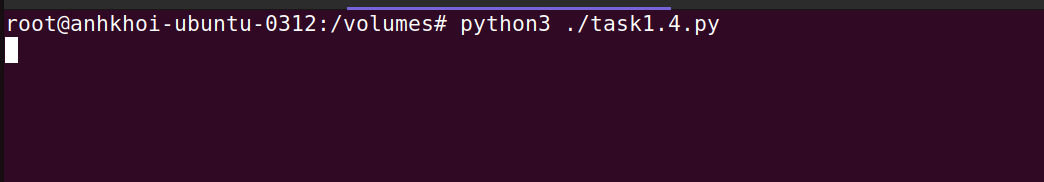




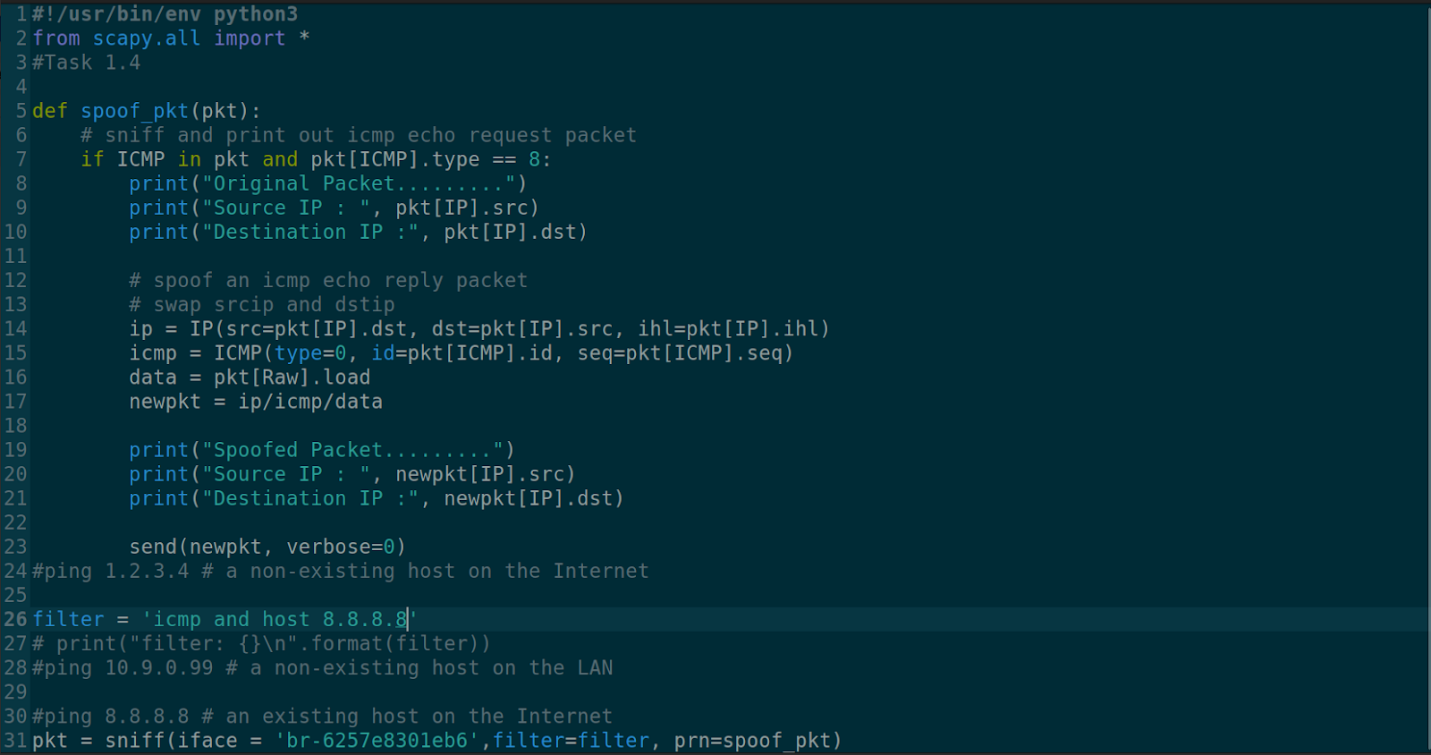
- Tiếp theo là địa chỉ IP 10.9.0.99, do địa chỉ này không hoạt động nên không thấy gói tin nào trả về

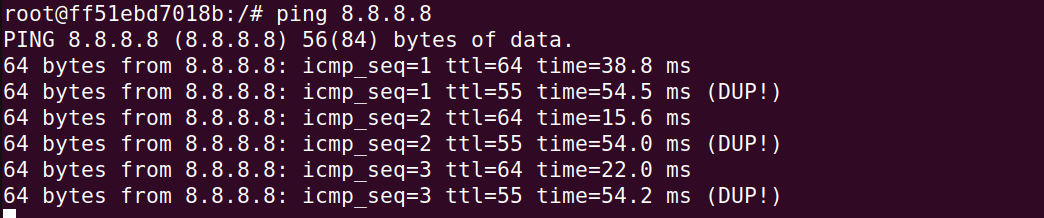


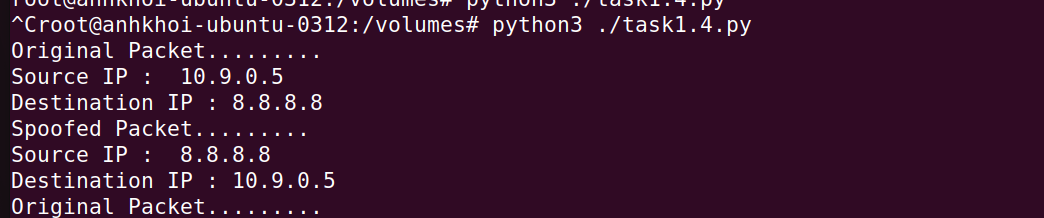




- Cuối cùng là địa chỉ IP 8.8.8.8





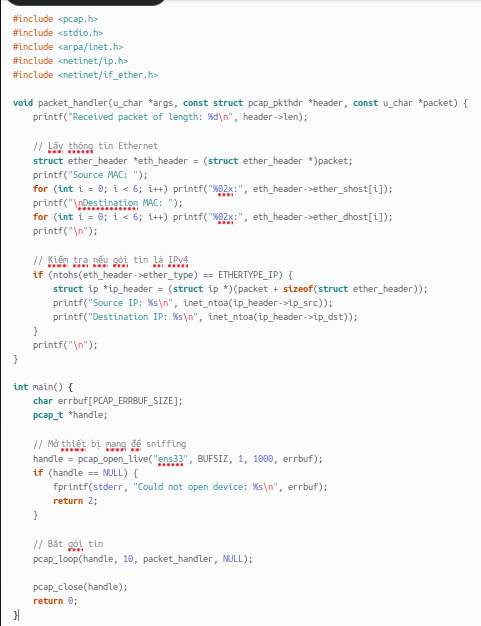


# Lab Task Set 2: Writing Programs to Sniff and Spoof Packets

## Task 2.1: Writing Packet Sniffing Program

- Sniffer programs can be easily written using the pcap library. With pcap, the task of sniffers becomes invoking a simple sequence of procedures in the pcap library. At the end of the sequence, packets will be put in buffer for further processing as soon as they are captured. All the details of packet capturing are handled by the pcap library.

## Task 2.1A: Understanding How a Sniffer Works



#### Question 1. Please use your own words to describe the sequence of the library calls that are essential for sniffer programs. This is meant to be a summary, not detailed explanation like the one in the tutorial or book.

Các lệnh thư viện chính:

* pcap\_open\_live: Mở một phiên để bắt gói tin.
* pcap\_compile: Chuyển bộ lọc thành mã để áp dụng.
* pcap\_setfilter: Áp dụng bộ lọc lên phiên bắt gói tin.
* pcap\_loop: Liên tục bắt gói tin và xử lý qua hàm got\_packet.
* pcap\_close: Đóng phiên bắt gói tin.

#### Question 2. Why do you need the root privilege to run a sniffer program? Where does the programfail if it is executed without the root privilege?

* Chương trình cần quyền root vì thao tác bắt gói tin yêu cầu truy cập trực tiếp vào giao diện mạng. Nếu không có quyền root, pcap\_open\_live sẽ gặp lỗi.

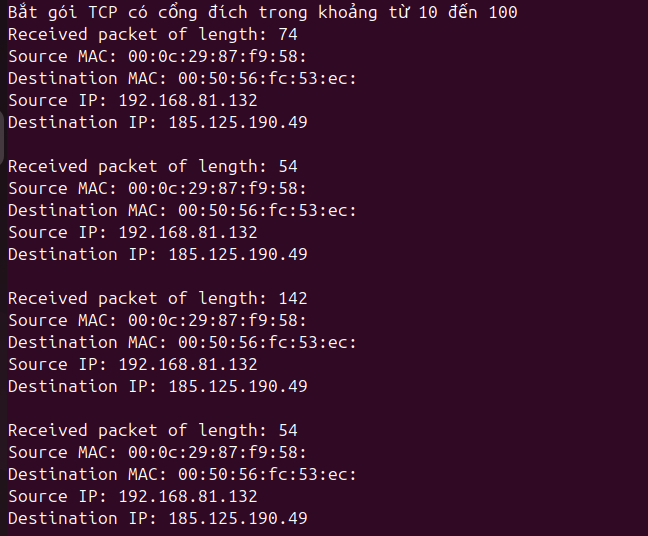
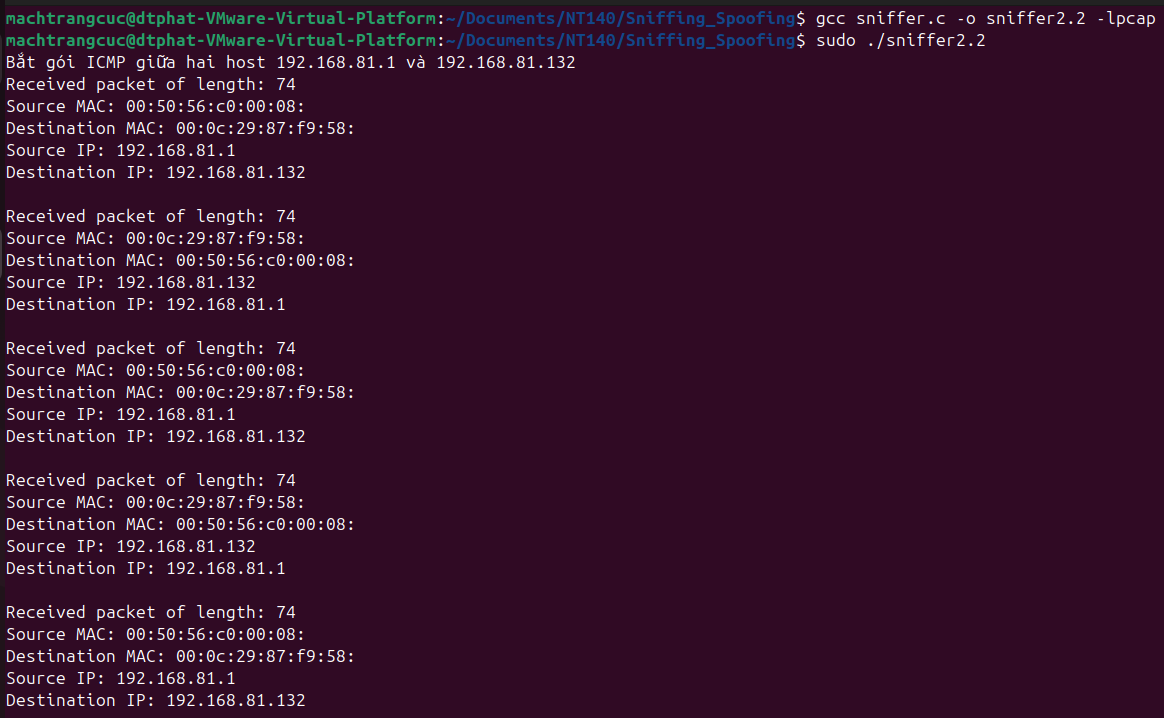
#### Question 3. Please turn on and turn off the promiscuous mode in your sniffer program. The value 1 of the third parameter in pcap open live() turns on the promiscuous mode (use 0 to turn it off). Can you demonstrate the difference when this mode is on and off? Please describe how you can demonstrate this.

* Để kiểm tra chế độ promiscuous, thử thay đổi tham số thứ ba trong pcap\_open\_live giữa 1 (bật) và 0 (tắt) và dùng lệnh sau để kiểm tra:

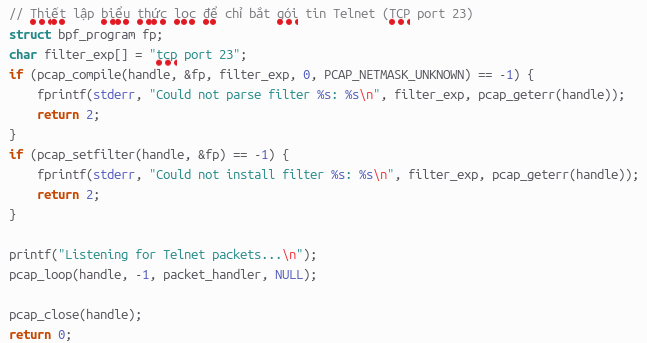
**ip -d link show dev eth0**

## Task 2.1B: Writing Filters.

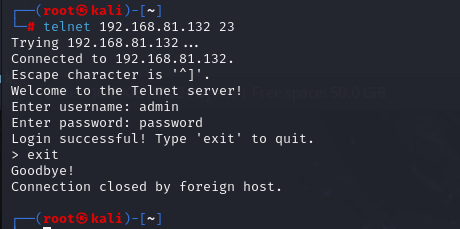
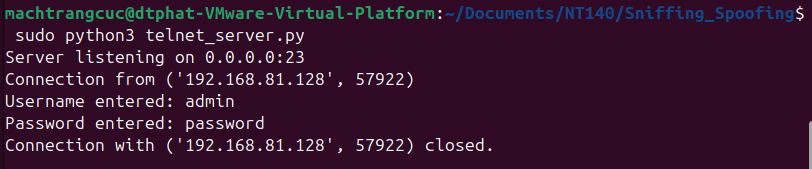


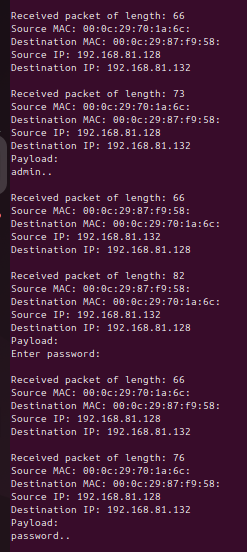
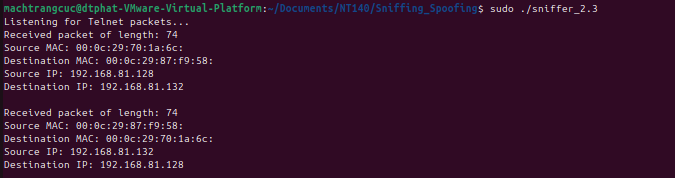


## Task 2.1C: Sniffing Passwords.

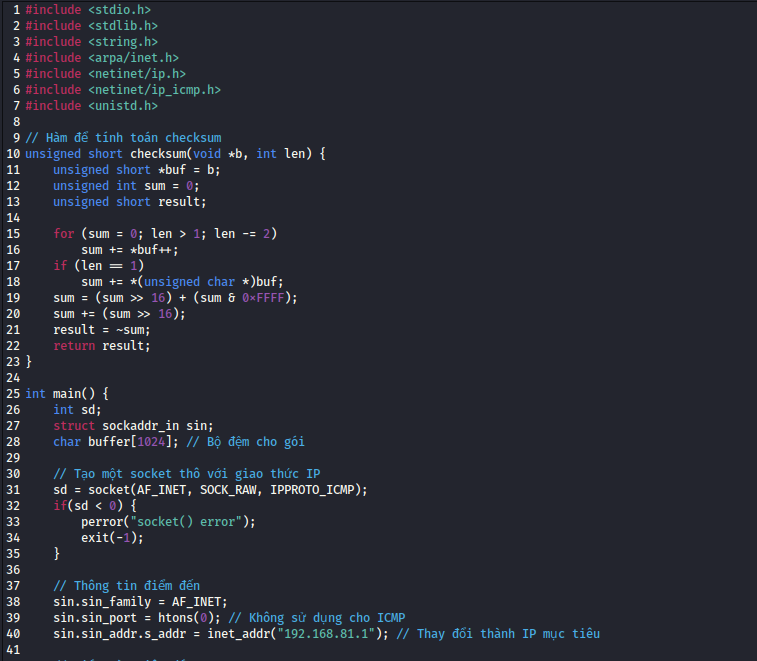


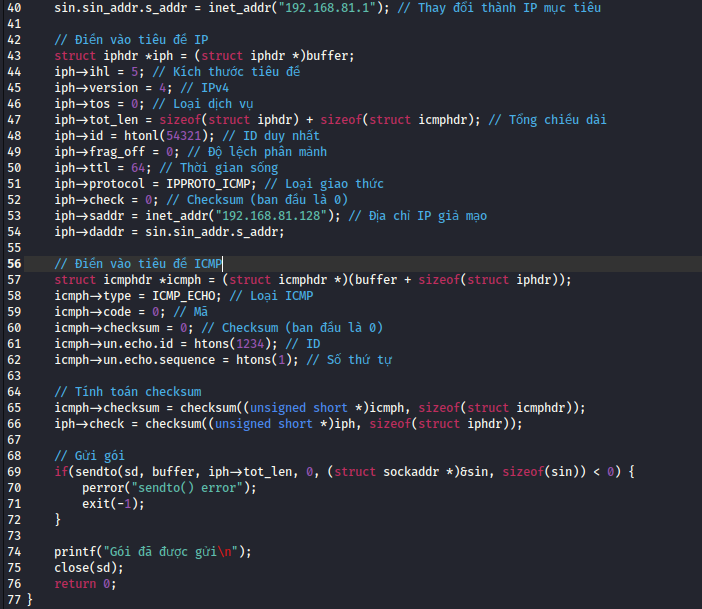
**- Tạo server và kết nối:**

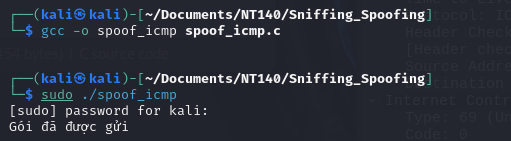
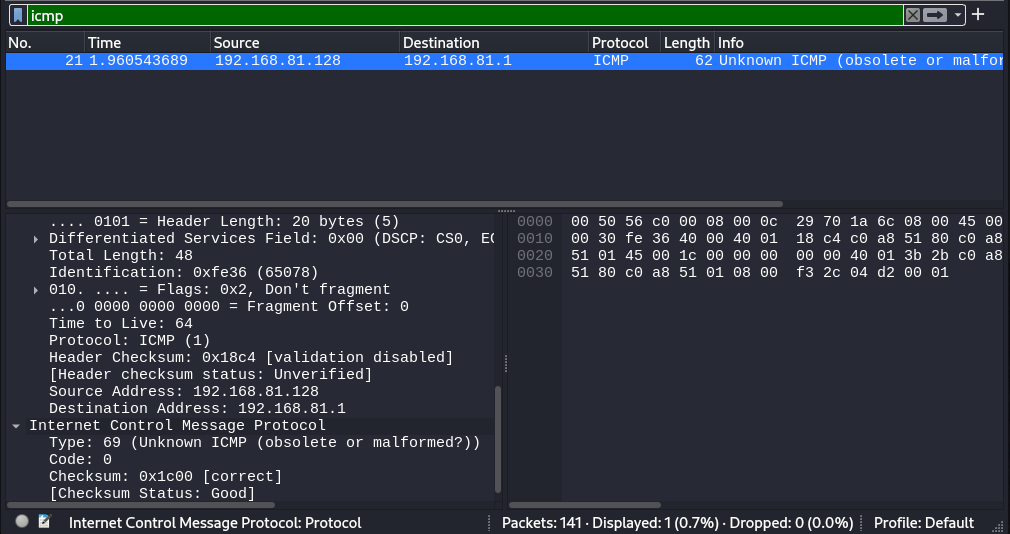

- Chạy code để bắt gói tin và hiển thị nếu bắt được mật khẩu:  
  


## Task 2.2A: Write a spooﬁng program.





## Task 2.2B: Spoof an ICMP Echo Request.

#### Question 4. Can you set the IP packet length field to an arbitrary value, regardless of how big the actual packet is?

- Về lý thuyết, có thể đặt nó thành bất kỳ giá trị nào, nhưng nó phải phù hợp với kích thước gói thực tế. Nếu không, gói có thể bị loại bỏ bởi máy nhận hoặc các thiết bị trung gian.

#### Question 5. Using the raw socket programming, do you have to calculate the checksum for the IP header?

- Có, phải tính toán checksum cho tiêu đề IP để đảm bảo rằng gói được hiểu đúng bởi hệ thống nhận.

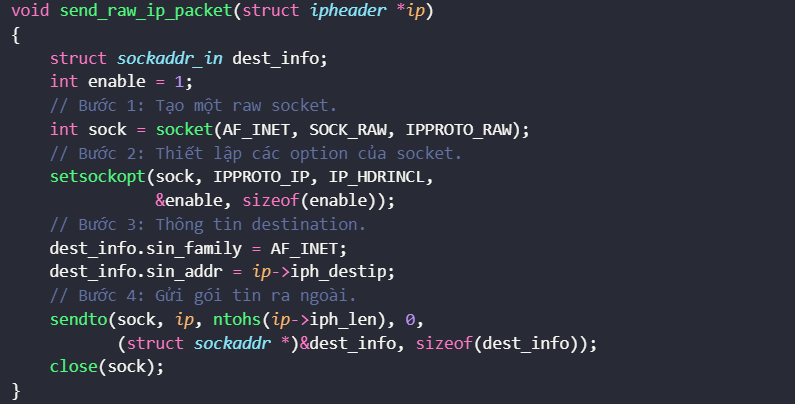
#### Question 6. Why do you need the root privilege to run the programs that use raw sockets? Where does the program fail if executed without the root privilege?

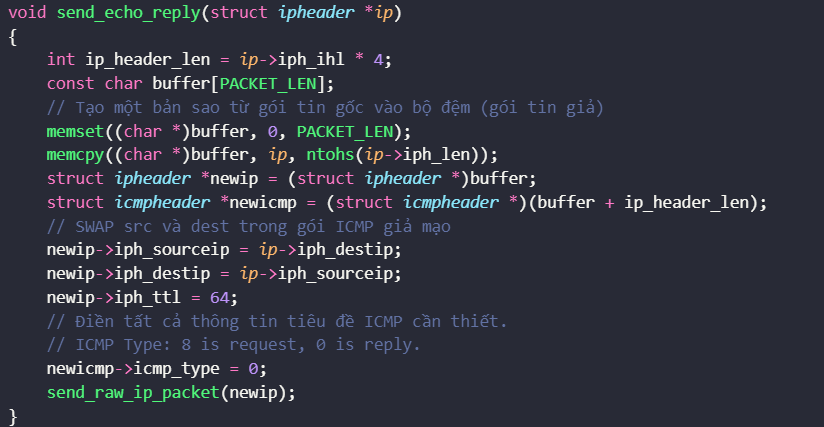
- Socket thô cho phép xây dựng các gói thủ công, điều này có thể được sử dụng cho các hoạt động độc hại như giả mạo gói. Do đó, quyền root là cần thiết để ngăn người dùng không có quyền gửi các gói được chế tạo. Nếu không có quyền root, việc tạo socket sẽ thất bại và sẽ nhận được thông báo lỗi "Permission denied".

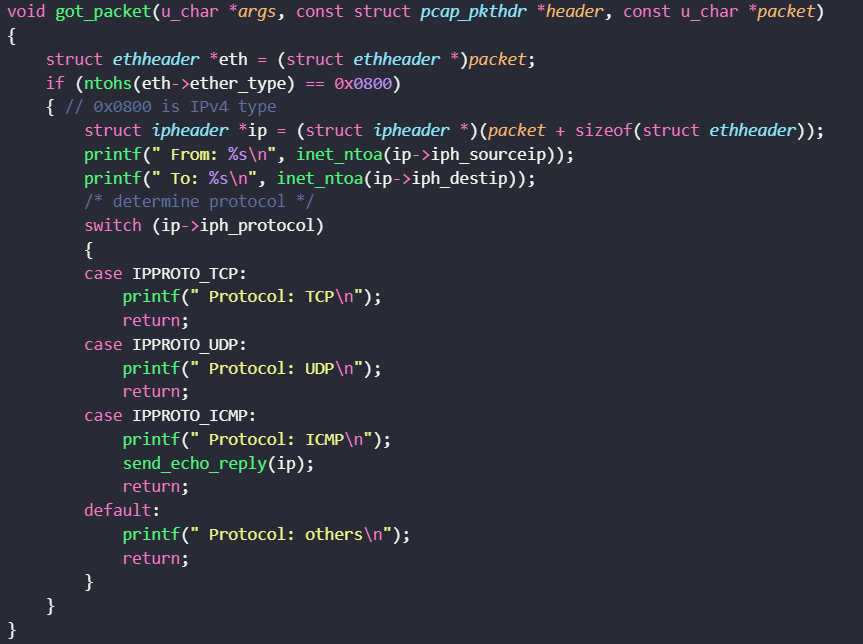
## Task 2.3: Sniff and then Spoof

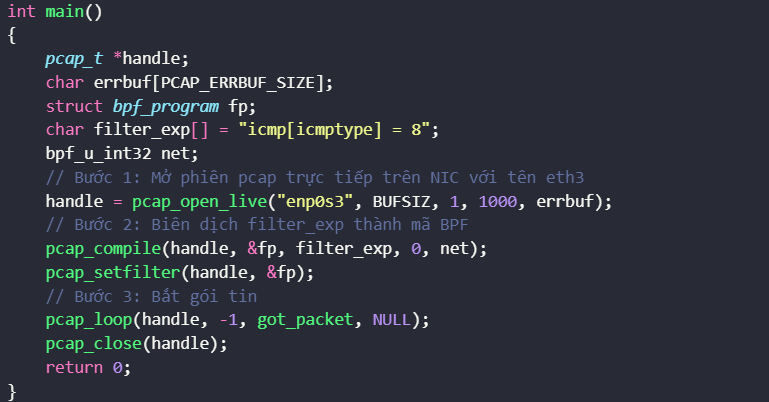
- Chương trình "sniff-and-then-spoof" chạy trên máy tấn công, giám sát mạng thông qua việc bắt gói tin. Khi nó phát hiện một gói tin yêu cầu ICMP echo, chương trình sẽ ngay lập tức gửi một phản hồi echo giả mạo. Như vậy, dù máy X có hoạt động hay không, chương trình ping sẽ luôn nhận được phản hồi, cho thấy máy X đang hoạt động

**Code:**





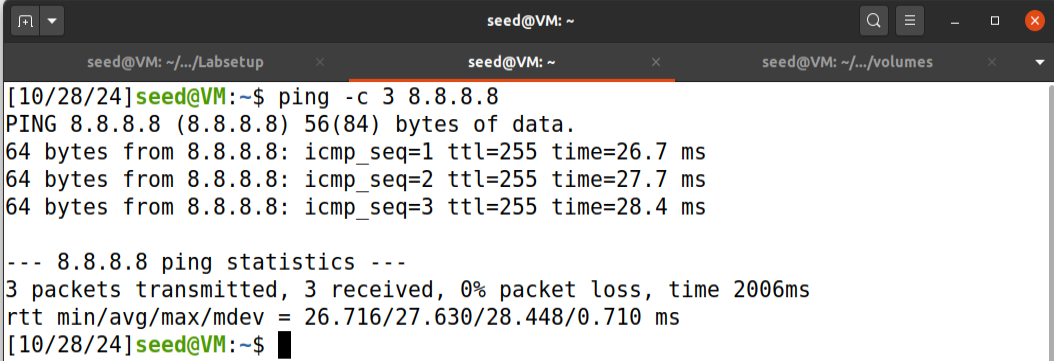


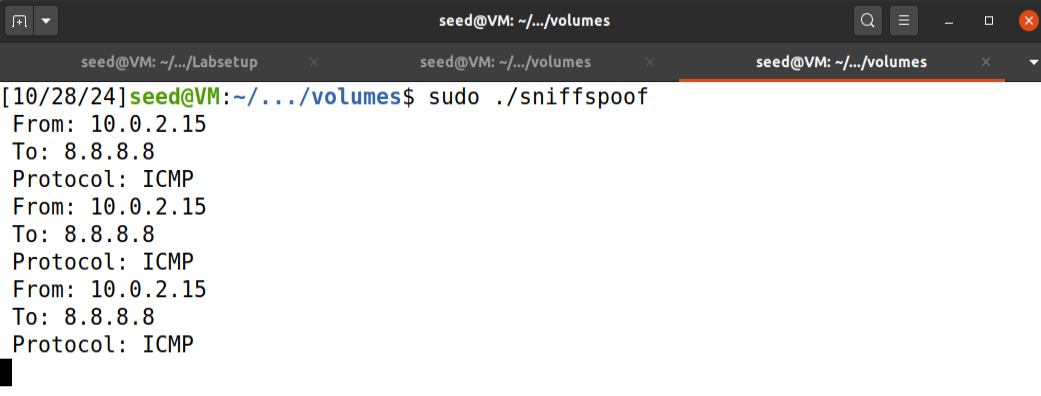


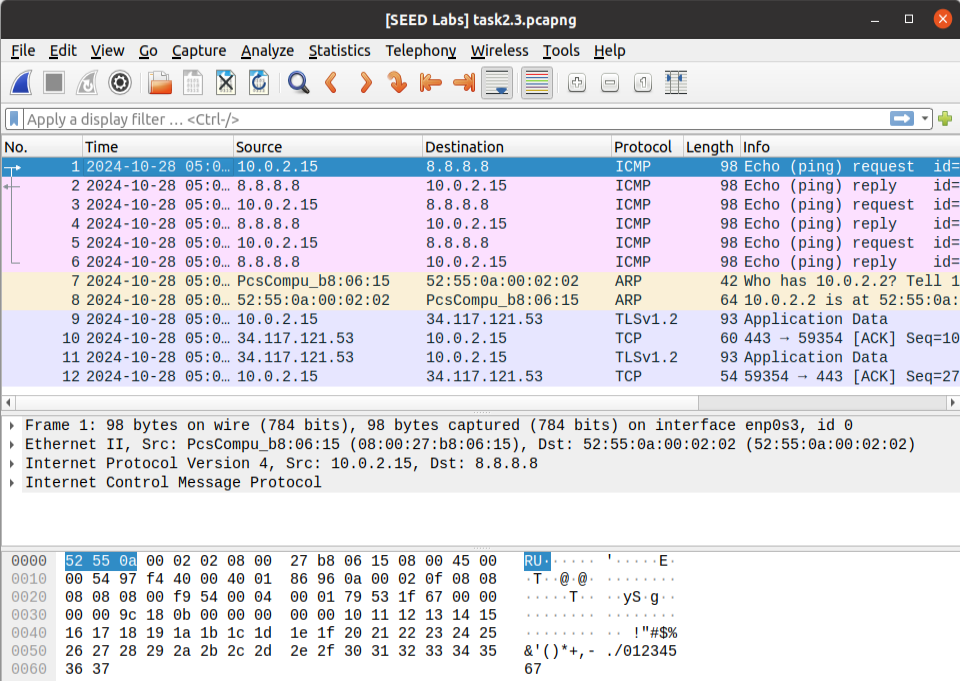
- Máy tấn công được đặt ở chế độ "promiscuous", nghĩa là nó có thể nhận tất cả các gói tin đi qua mạng mà không chỉ gói tin gửi đến nó. Khi chạy chương trình giả mạo (spoofing), thẻ mạng (NIC) của máy attacker sẽ bắt tất cả các gói tin đến.

- Chương trình sẽ xử lý các gói tin này theo cách mà nó thay đổi địa chỉ đích thành địa chỉ nguồn và địa chỉ nguồn thành địa chỉ đích. Sau khi tạo gói tin mới, nó sẽ gửi gói tin ra ngoài. Kết quả là, nạn nhân sẽ nhận được gói tin đó. Như vậy, chúng ta đã giả mạo thành công yêu cầu echo ICMP.

**Screenshots:**

The Victim

The Attacker



-- Hết --

1. Ghi nội dung công việc, các kịch bản trong bài Thực hành [↑](#footnote-ref-2)