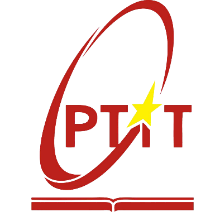
**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**KHOA AN TOÀN THÔNG TIN**

-------🙞🙜🕮🙞🙜-------



**Báo cáo xây dựng lab trên nền tảng labtainer**

**Chủ đề:** Nghiên cứu và xây dựng các bài thực hành dựa trên khung Mitre Att&ck trong kiểm thử xâm nhập

**Lab**: T1483 - Domain Generation Algorithms

Sinh viên thực hiện: Ngô Quang Dũng - B20DCAT029

Giảng viên hướng dẫn: TS. Nguyễn Ngọc Điệp

**Hà Nội - 2025**

**MỤC LỤC**

[MỤC LỤC](#_gjdgxs) 2

[DANH MỤC CÁC HÌNH VẼ](#_30j0zll) 3

[DANH MỤC CÁC BẢNG BIỂU](#_1fob9te) 3

[DANH MỤC CÁC TỪ VIẾT TẮT](#_3znysh7) 4

[1.1 Giới thiệu chung về bài thực hành](#_2et92p0) 5

[1.2 Nội dung và hướng dẫn bài thực hành](#_tyjcwt) 5

[**1.2.1** Mục đích](#_3dy6vkm) 5

[**1.2.2** Yêu cầu đối với sinh viên](#_1t3h5sf) 5

[**1.2.3** Nội dung thực hành](#_4d34og8) 5

[1.3 Phân tích yêu cầu bài thực hành](#_2s8eyo1) 6

[1.4 Thiết kế bài thực hành](#_17dp8vu) 6

[1.5 Cài đặt và cấu hình các máy ảo 8](https://docs.google.com/document/d/1hfn2yF06gobzwfBq4KR7ymxMCejHgIZDlf25SwvjSpI/edit#heading=h.1ksv4uv)

[1.6 Tích hợp và triển khai 10](#_1ci93xb)

[***1.6.1*** Docker Hub 10](#_3whwml4)

[***1.6.2*** Github 10](#_1pxezwc)

[1.7 Thử nghiệm và đánh giá 11](#_3o7alnk)

[TÀI LIỆU THAM KHẢO 1](#_vx1227)5

**DANH MỤC CÁC HÌNH VẼ**

[Hình 1 Giao diện Labedit của bài lab 8](#_44sinio)

[Hình 2 Cài đặt phần Result 8](#_2jxsxqh)

[Hình 3 Dockerfiles của máy victim 9](#_4i7ojhp)

[Hình 4 Dockerfiles của attacker 9](#_2xcytpi)

[Hình 5 Add và commit bài lab 10](#_2bn6wsx)

[Hình 6 Đẩy các vùng chứa lên dockerhub 10](#_qsh70q)

[Hình 7 Tạo imodule.tar chứa bài thực hành 10](#_3as4poj)

[Hình 8 Các vùng chứa được đẩy lên dockerhub 11](#_49x2ik5)

[Hình 9 Tạo file Imodule.tar 11](#_2p2csry)

[Hình 10 File imodule.tar chứa bài thực hành 11](#_147n2zr)

[Hình 11 Đẩy file imodule.tar lên github 12](#_23ckvvd)

[Hình 12 Chạy imodule của bài 12](#_ihv636)

[Hình 13 IP máy server 12](#_ihv636)

[Hình 14 Máy server chạy thuật toán sinh miền ngẫu nhiên 12](#_32hioqz)

[Hình 15 Trả lời Task1 1](#_32hioqz)3

[Hình 16 Server cấu hình dnsmasq 1](#_32hioqz)3

[Hình 17 Client cấu hình 1](#_1hmsyys)3

[Hình 18 Client gửi truy vấn 1 tên miền không hợp lệ 1](#_41mghml)4

[Hình 19 Client gửi truy vấn 1 tên miền hợp lệ 1](#_2grqrue)4

[Hình 20 Trả lời ip miền truy vấn hợp lệ Task2 1](#_2grqrue)5

[Hình 21 Theo dõi các truy vấn DNS 1](#_2grqrue)5

[Hình 22 Checkwork bài lab 1](#_2grqrue)5

**DANH MỤC CÁC BẢNG BIỂU**

[Bảng 1. Bảng Result 7](https://docs.google.com/document/d/1hfn2yF06gobzwfBq4KR7ymxMCejHgIZDlf25SwvjSpI/edit#heading=h.3rdcrjn)

**DANH MỤC CÁC TỪ VIẾT TẮT**

| **Từ  viết tắt** | **Thuật ngữ tiếng Anh/Giải thích** | **Thuật ngữ tiếng Việt/Giải thích** |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

* 1. **Giới thiệu chung về bài thực hành**

Bài lab này được thiết kế để giúp sinh viên hiểu rõ hơn về cách thức hoạt động của Domain Generation Algorithms (DGA), một kỹ thuật thường được mã độc sử dụng để duy trì liên lạc với máy chủ điều khiển (Command & Control - C2). Trong bài lab, server đóng vai trò mô phỏng mã độc bằng cách sử dụng script generate\_dga.py để tạo ra các tên miền ngẫu nhiên, tương tự như cách các mã độc như Conficker hoặc Dridex hoạt động. Đồng thời, server cũng cấu hình dịch vụ DNS giả lập thông qua dnsmasq để xử lý các truy vấn DNS từ client và ghi nhận chúng vào file log để phân tích. Client được cấu hình để gửi truy vấn DNS đến các tên miền được sinh ra, giúp mô phỏng hành vi của mã độc khi cố gắng kết nối với máy chủ C2.

* 1. **Nội dung và hướng dẫn bài thực hành**
     1. ***Mục đích***

Mục tiêu chính của bài lab là giúp sinh viên thực hành các kỹ thuật phát hiện và phân tích các mẫu truy vấn DNS bất thường, từ đó nâng cao khả năng phát hiện và phòng thủ trước các cuộc tấn công sử dụng DGA.

* + 1. ***Yêu cầu đối với sinh viên***

Hiểu cơ bản về giao thức DNS và cách hoạt động của truy vấn DNS.

* + 1. ***Nội dung thực hành***

Khởi động bài lab:

Vào terminal, gõ:

*startlab pen\_attk\_mitre-t1483*

*(chú ý: sinh viên sử dụng mã sinh viên của mình để nhập thông tin email người thực hiện bài lab khi có yêu cầu, để sử dụng khi chấm điểm)*

Sau khi khởi động xong hai terminal ảo sẽ xuất hiện, một cái là đại diện cho máy tấn công: ***client***, một cái là đại diện cho máy victim: ***server***.

Chạy file python để sinh các tên miền ngẫu nhiên trên **server**:

*python3 generate\_dga.py*

cấu hình ghi log truy vấn DNS và thiết lập địa chỉ IP cho tên miền ngẫu nhiên:

*address=/ljpewe9mjx.com/192.168.1.100*

*log-queries*

*log-facility=/var/log/dnsmasq.log*

Khởi động lại dịch vụ dnsmasq để áp dụng thay đổi:

*sudo systemctl restart dnsmasq*

Cấu hình client để sử dụng dnsmasq

*sudo nano /etc/resolv.conf*

*nameserver 172.21.0.20*

Sử dụng công cụ dig để gửi truy vấn DNS đến máy chủ DNS giả lập (dnsmasq):

*dig @172.21.0.20 ljpewe9mjx.com*

*dig @172.21.0.20 test.dfsdf*

Theo dõi log của dnsmasq để xem các truy vấn DNS và phát hiện tên miền ngẫu nhiên trên server

*sudo tail -f /var/log/dnsmasq.log*

Kết thúc bài lab:

Trên terminal đầu tiên sử dụng câu lệnh sau để kết thúc bài lab:

*stoplab domain\_generation*

Khi bài lab kết thúc, một tệp zip lưu kết quả được tạo và lưu vào một vị trí được hiển thị bên dưới stoplab.

Khởi động lại bài lab:

Trong quá trình làm bài sinh viên cần thực hiện lại bài lab, dùng câu lệnh:

*startlab –r domain\_generation*

* 1. **Phân tích yêu cầu bài thực hành**

Bài thực hành gồm 2 máy tính nằm trong cùng mạng LAN. Trong đó 1 máy được gọi là attacker, máy còn lại là victim. Để hoàn thành bài thực hành sinh viên cần sử dụng script để tạo tên miền ngẫu nhiên, truy vấn các DNS từ client và server theo dõi được các truy vấn từ client .

* 1. **Thiết kế bài thực hành**

Trên môi trường máy ảo Ubuntu được cung cấp, sử dụng docker tạo ra 2 container: 1 container mang tên “client” đóng vai trò máy tấn công và 1 container mang tên “server” đóng vai trò máy nạn nhân.

Tạo mạng LAN có cấu hình: 172.21.0.0/24 và gateway: 172.21.0.1

Cấu hình docker gồm có:

Client: lưu cấu hình cho máy tấn công, trong đó gồm có:

Tên máy: client

Địa chỉ trong mạng LAN: 172.21.0.10

Gateway: 172.21.0.1

Server: lưu cấu hình cho máy chủ, trong đó gồm có:

Tên máy: Server

Địa chỉ trong mạng LAN: 172.21.0.20

Gateway: 172.21.0.1

config: lưu cấu hình hoạt động của hệ thống

dockerfiles: mô tả cấu hình của 2 container: client và server, trong đó:

Client: sử dụng các thư viện mặc định hệ thống và cài sẵn curl giúp thực hiện các lệnh và chức năng của tool này.

Server: sử dụng các thư viện mặc định hệ thống cùng với cấu hình tương tự máy attacker trong đó đã cài sẵn các dịch vụ apache.

Các nhiệm vụ cần phải thực hiện để thực hành thành công:

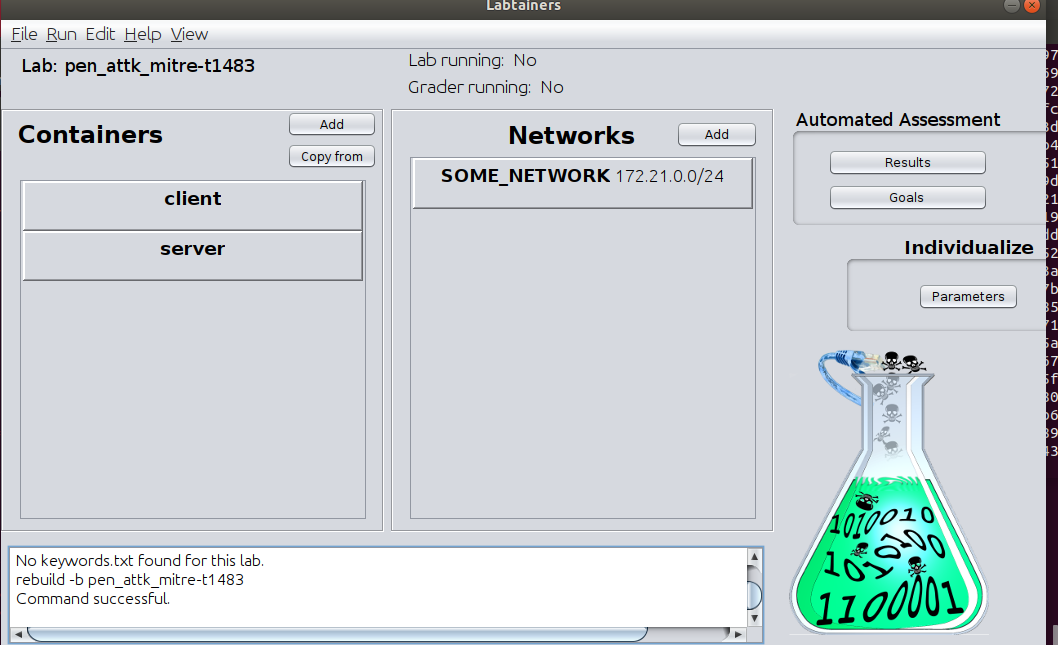
Task 1: Sinh viên cần điền kết quả phần đuôi tên miền sau khi chạy file generate\_dga.py trả lời vào file domain.txt trong máy server.

Task 2: Sinh viên cần truy vấn được vào tên miền đã config ở server và ghi vào file ip.txt trên client.

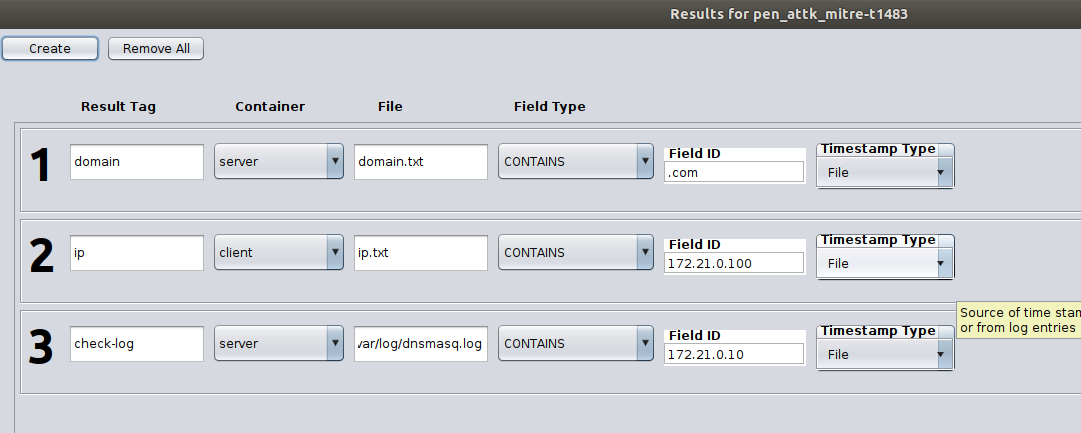
Task 3: Sinh viên theo dõi được log của server những tên miền nào gửi truy vấn DNS đến.

Kết thúc bài lab và đóng gói kết quả.

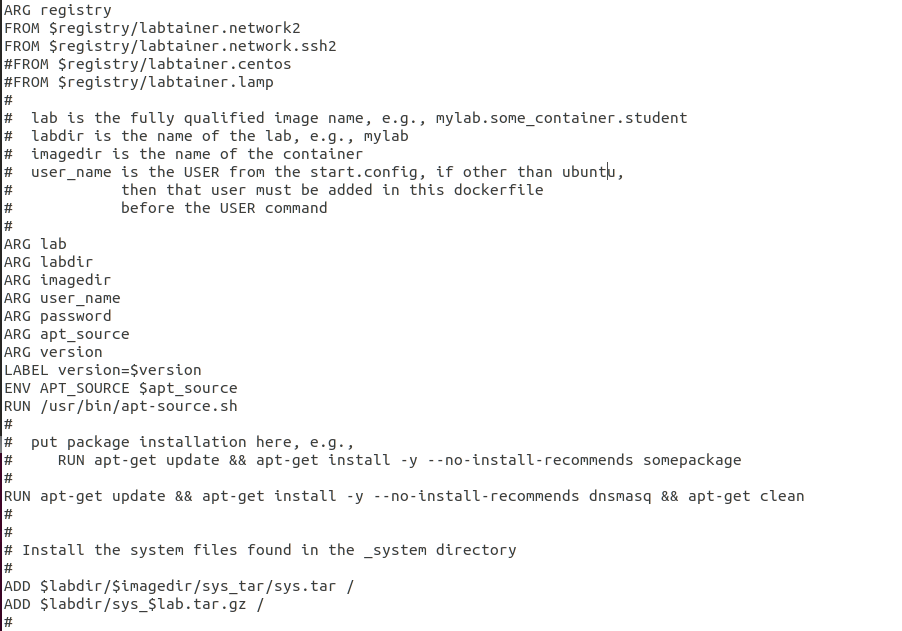
Thiết lập hệ thống mạng sao cho máy attacker và máy victim cùng một mạng LAN.



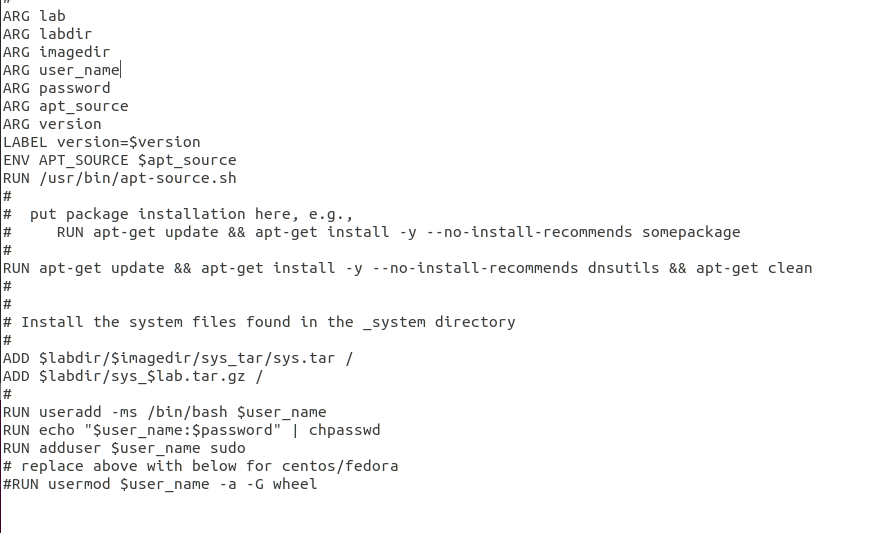
*Hình 1: Giao diện Labedit của bài lab*



*Hình 2: Cài đặt phần Result*



*Hình 3: Dockerfiles của máy victim*



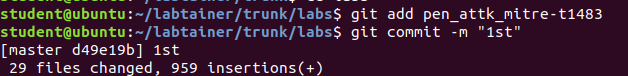
*Hình 4: Dockerfiles của client*

* 1. **Tích hợp và triển khai**

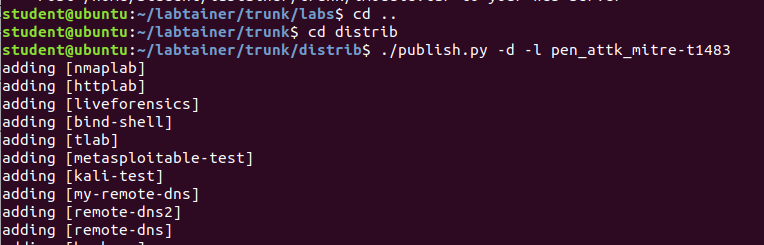
Bài thực hành đã được triển khai như sau:

* + 1. Docker Hub

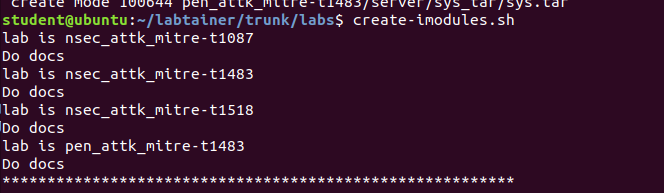
https://hub.docker.com/repositories/dung1604



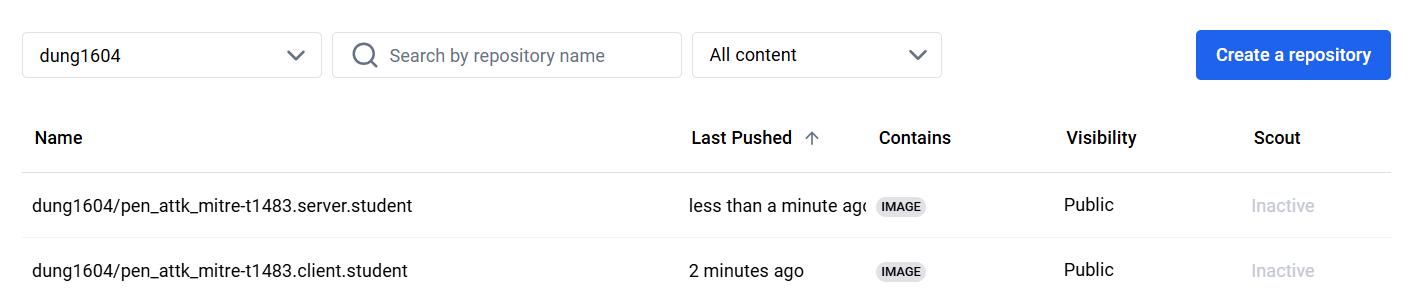
*Hình 5: Add và commit bài lab*



*Hình 6: Đẩy các vùng chứa lên dockerhub*



*Hình 7: Tạo imodule.tar chứa bài thực hành*

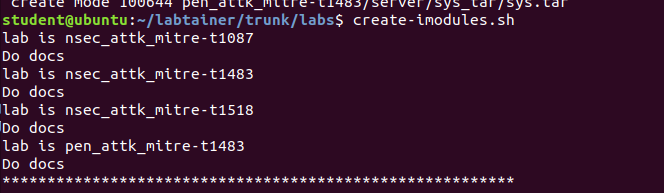


*Hình 8: Các vùng chứa được đẩy lên dockerhub*

* + 1. Github

https://github.com/DUNG1604/nsec\_attk\_mitre-t1483

Nhập lệnh create-imodules.sh

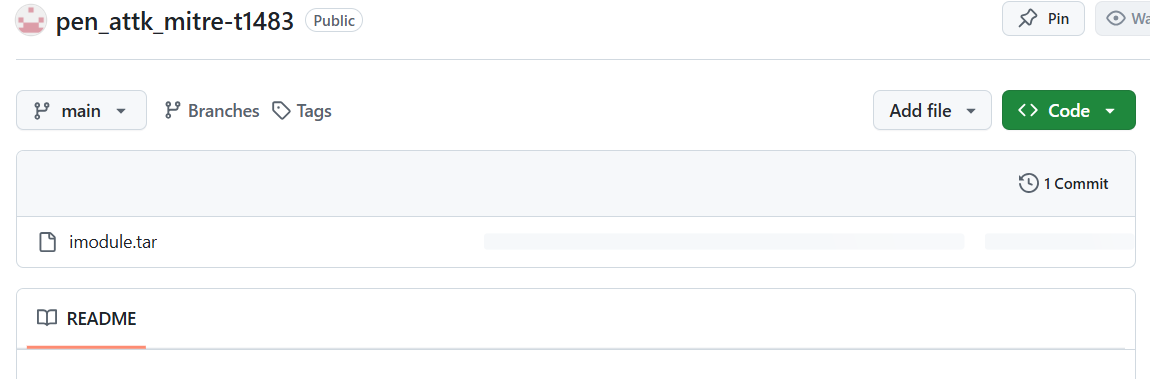


*Hình 9: Tạo file Imodule.tar*



*Hình 10: File imodule.tar chứa bài thực hành*

Tạo repo mới để đẩy imodule.tar lên và tạo phần release mới



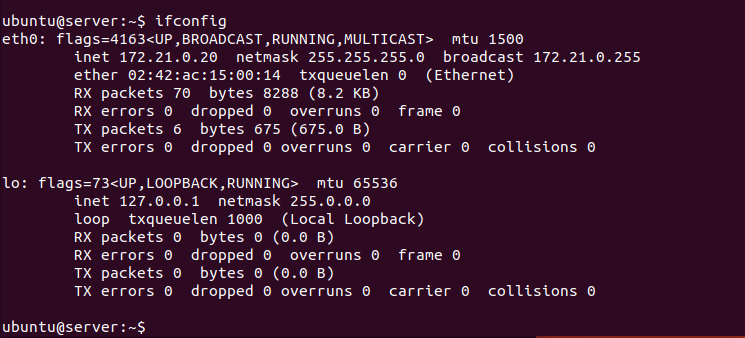
*Hình 11: Đẩy file imodule.tar lên github*

* 1. **Thử nghiệm và đánh giá**

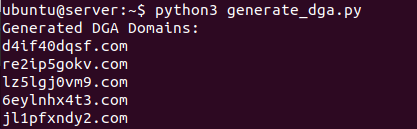
Bài thực hành đã được xây dựng thành công, dưới đây là hình ảnh minh họa về bài thực hành:



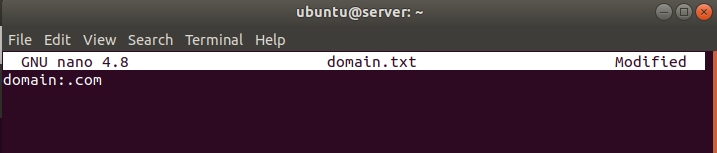
*Hình 12: Chạy imodule của bài*



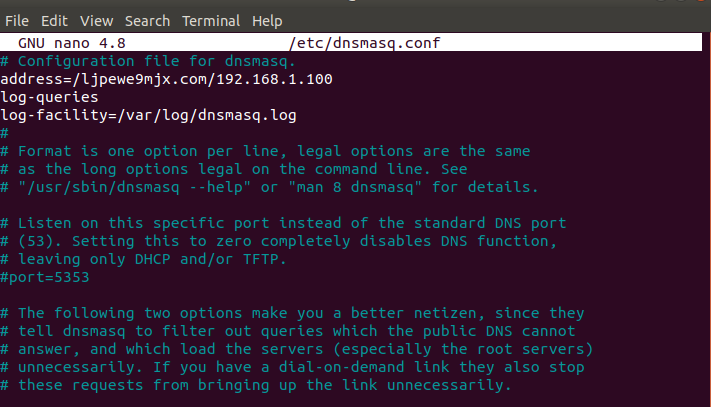
*Hình 13: IP của máy server*



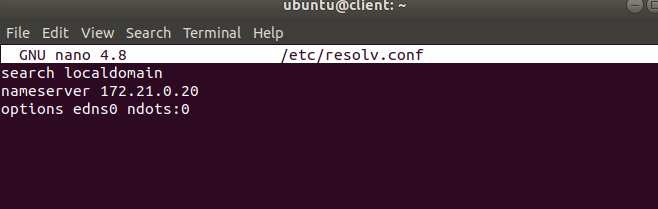
*Hình 14 Máy server chạy thuật toán sinh miền ngẫu nhiên*



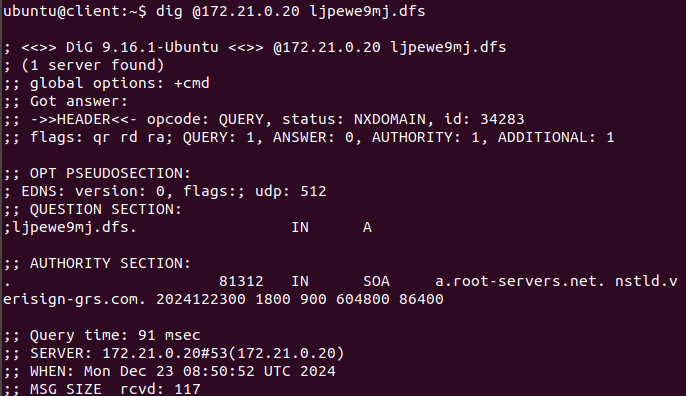
*Hình 15: Trả lời task1*



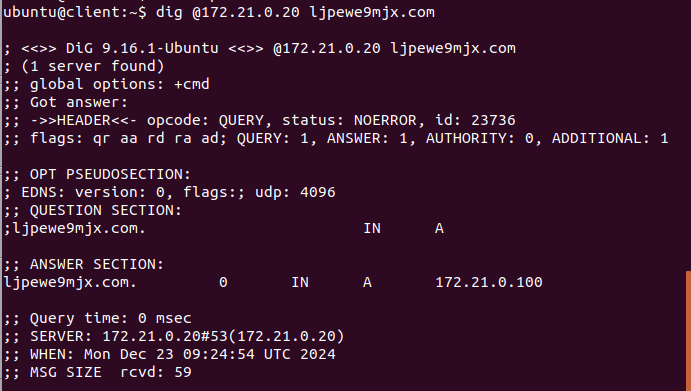
*Hình 16: Server cấu hình dnsmasq*



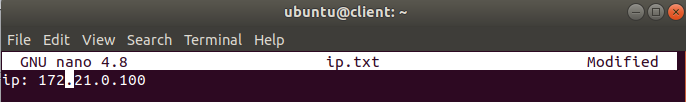
*Hình 17: Client cấu hình*



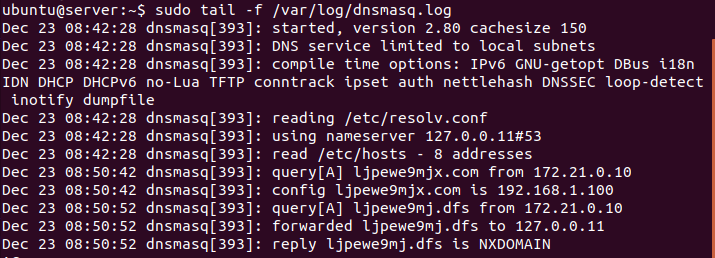
*Hình 18: Client gửi truy vấn 1 tên miền không hợp lệ*



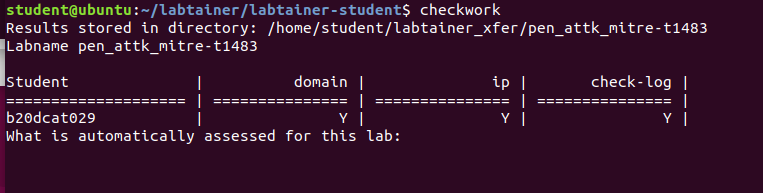
*Hình 19: Client gửi truy vấn 1 tên miền hợp lệ*

**

*Hình 20: Trả lời ip miền truy vấn hợp lệ Task2*

**

*Hình 21: Theo dõi các truy vấn DNS*



*Hình 22: Checkwork bài lab*

**TÀI LIỆU THAM KHẢO**

1. https://attack.mitre.org/techniques/T1568/002/
2. https://nps.edu/documents/107523844/117289221/labdesigner.pdf/9ab93fc3-7cfe-48f4-987b-2efda8d99346?t=1606868757623