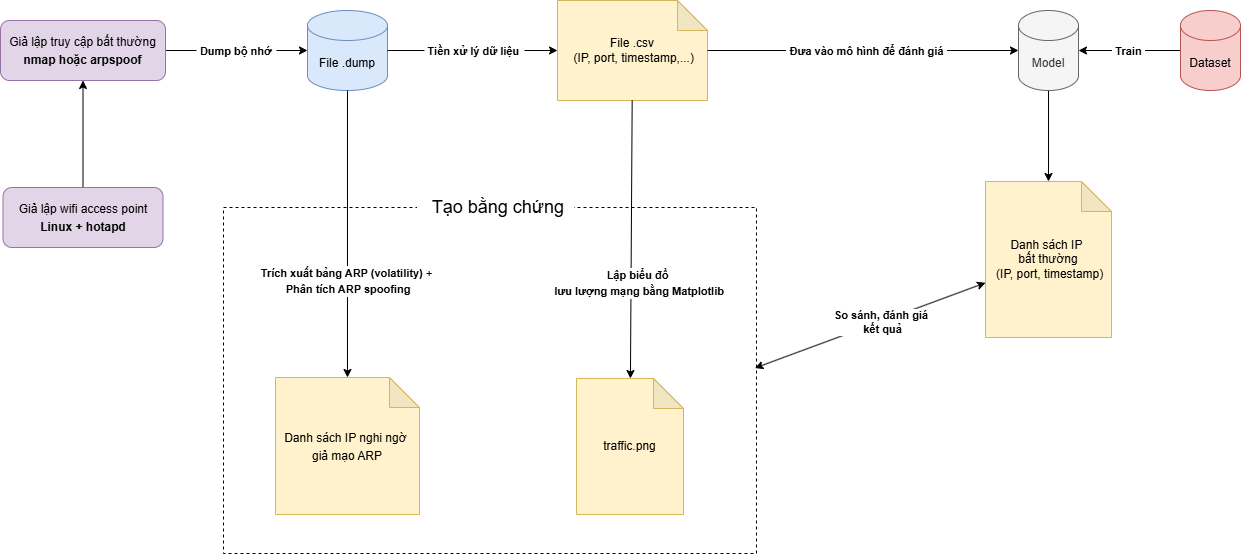
**Kịch bản demo Pháp chứng số**



# Nội dung demo



1. **Setup hệ thống mô phỏng**:

* Dùng máy ảo Linux (như Ubuntu) cài hostapd để tạo wifi access point ảo.
* Kết nối vài thiết bị ảo (hoặc mô phỏng bằng script) để tạo lưu lượng truy cập.

1. **Giả lập truy cập bất thường**:

* Dùng công cụ như nmap (quét cổng - giả lập số lượng truy cập lớn) hoặc arpspoof (giả mạo ARP giả địa chỉ MAC) để tạo hành vi đáng ngờ từ một IP cụ thể (ví dụ: 192.168.1.100).
* Thực hiện ngay trước khi dump để dữ liệu bất thường được ghi vào bộ nhớ.

1. **Dump bộ nhớ**:

Dùng LiME hoặc dd để lấy file .dump từ máy ảo chạy AP. File này sẽ chứa thông tin về các truy cập bất thường vừa tạo.

1. **Tiền xử lý dữ liệu**:

* Dùng Volatility trích xuất thông tin từ file .dump:

vol.py -f memory.dump linux\_netstat (lấy kết nối mạng: IP, port, timestamp).

* Chuyển file .dump thành file CSV (thủ công hoặc dùng script Python). Ví dụ file network\_data.csv chứa cột: IP, Port, Timestamp, Packet\_Count.

1. **Phân tích bằng AI**:

| **Dataset** | **Mô tả** | **Đặc trưng chính** | **Phù hợp với mô hình** |
| --- | --- | --- | --- |
| NSL-KDD | Phiên bản cải tiến của KDD Cup 1999, dùng cho phát hiện xâm nhập. | IP, port, protocol, 41 đặc trưng. | Random Forest |
| CICIDS2017 | Lưu lượng mạng thực tế với các tấn công hiện đại (DDoS, Botnet, v.v.). | IP, port, timestamp, 80+ đặc trưng. | Random Forest, Autoencoder |
| UNSW-NB15 | Lưu lượng mạng với các tấn công phức tạp (Fuzzers, Exploits, v.v.). | IP, port, protocol, 49 đặc trưng. | DNN |
| CSE-CIC-IDS2018 | Mở rộng của CICIDS2017, tập trung vào tấn công hiện đại. | IP, port, timestamp, 80+ đặc trưng. | DNN |
| KDD Cup 1999 | Dataset cổ điển cho phát hiện xâm nhập, có dữ liệu bình thường và tấn công. | IP, port, protocol, 41 đặc trưng. | Autoencoder |
| UGR’16 | Lưu lượng mạng thực tế với tính tuần hoàn (day/night). | Timestamp, IP, port, packet count. | LSTM |
| Bot-IoT | Lưu lượng mạng IoT với các tấn công botnet, DDoS, v.v. | Timestamp, IP, port, packet count. | LSTM |

1. **Chứng minh truy cập bất thường**:

**Thông qua ARP**: ARP (Address Resolution Protocol) là giao thức ánh xạ IP sang địa chỉ MAC. Nếu có giả mạo ARP (ARP spoofing), bảng ARP trong file dump sẽ cho thấy IP bất thường gắn với MAC không hợp lệ hoặc trùng lặp. Dùng lệnh vol.py -f memory.dump linux\_arp để kiểm tra.

**Lập biểu đồ lưu lượng mạng**: Dùng Python (thư viện Matplotlib hoặc Seaborn) để vẽ biểu đồ từ file CSV. Ví dụ: Trục X là thời gian (timestamp), trục Y là số lượng gói tin từ mỗi IP. IP bất thường sẽ có "spike" (đỉnh cao bất thường).

# Chuẩn bị

### **Phần chuẩn bị trước (làm trước buổi báo cáo)**

Những bước này cần hoàn thành trước để có sẵn dữ liệu và kết quả, tránh mất thời gian trong buổi demo:

1. **Setup hệ thống mô phỏng**:
   * Cài máy ảo Linux, dùng hostapd tạo wifi access point ảo.
   * Kết nối vài thiết bị ảo hoặc mô phỏng lưu lượng bằng script.
2. **Giả lập truy cập bất thường**:
   * Dùng nmap hoặc arpspoof để tạo truy cập đáng ngờ (ví dụ: IP 192.168.1.100 quét cổng hoặc giả mạo ARP).
   * Ghi lại hành vi này (có thể dùng Wireshark để lưu log mạng làm bằng chứng bổ sung, lúc làm có thể đánh giá cần bước này không).
3. **Dump bộ nhớ**:
   * Dùng LiME hoặc dd để lấy file .dump từ máy ảo ngay sau khi tạo truy cập bất thường. Lưu file này (ví dụ: memory.dump).
4. **Tiền xử lý dữ liệu**:
   * Dùng Volatility trích xuất thông tin từ file .dump:
     + Lệnh: vol.py -f memory.dump linux\_netstat (lấy kết nối mạng: IP, port, timestamp).
     + Lệnh: vol.py -f memory.dump linux\_arp (lấy bảng ARP: IP và MAC).
   * Chuyển kết quả thành file CSV (thủ công hoặc dùng script Python). Ví dụ file network\_data.csv chứa cột: IP, Port, Timestamp, Packet\_Count.
5. **Phân tích bằng AI**:
   * Train trước một mô hình AI (như Random Forest) với dataset mạng (bình thường + bất thường).
   * Chạy AI trên file CSV để phát hiện truy cập đáng ngờ. Lưu kết quả (ví dụ: "IP 192.168.1.100 là bất thường") vào file text hoặc slide.
6. **Chuẩn bị bằng chứng**:
   * **Bảng ARP**: Lưu output của linux\_arp (ví dụ: IP 192.168.1.100 gắn với MAC bất thường).
   * **Biểu đồ lưu lượng**: Dùng Python (Matplotlib) vẽ biểu đồ từ file CSV (ví dụ: IP 192.168.1.100 có spike gói tin). Lưu thành file ảnh (như traffic.png).

**Kết quả sau chuẩn bị**:

* File memory.dump.
* File network\_data.csv.
* Kết quả AI (text/slide).
* Bằng chứng: Bảng ARP (text) và biểu đồ (ảnh).

### **Phần chạy lúc demo (trong buổi báo cáo)**

Trong buổi báo cáo chỉ nên chạy các bước minh họa nhanh, sử dụng dữ liệu đã chuẩn bị trước:

1. **Giới thiệu ngắn**:
   * "Tôi đã mô phỏng một wifi access point, tạo truy cập bất thường, dump bộ nhớ và dùng AI phân tích. Đây là kết quả."
2. **Chạy Volatility minh họa**:
   * Dùng file memory.dump đã chuẩn bị:
     + Lệnh: vol.py -f memory.dump linux\_netstat (hiển thị kết nối mạng).
     + Lệnh: vol.py -f memory.dump linux\_arp (hiển thị bảng ARP).
   * Không cần trích xuất lại toàn bộ, chỉ chạy 1-2 lệnh để minh họa rằng dữ liệu lấy từ file dump. Kết quả có thể chuẩn bị trước trên slide để tiết kiệm thời gian nếu lệnh chạy chậm.
3. **Hiển thị kết quả AI**:
   * Đưa file CSV (network\_data.csv) vào AI (nếu nhanh) hoặc trình bày kết quả đã chuẩn bị:
     + "AI phát hiện IP 192.168.1.100 là bất thường vì gửi 500 gói tin trong 10 giây."
4. **Chứng minh truy cập bất thường**:
   * **Bảng ARP**: Hiển thị output linux\_arp (slide hoặc terminal) và chỉ ra: "IP 192.168.1.100 gắn với MAC không hợp lệ, nghi ngờ ARP spoofing."
   * **Biểu đồ**: Hiển thị file ảnh traffic.png (slide) và giải thích: "IP này có lưu lượng tăng đột biến, là dấu hiệu bất thường."

# Các loại truy cập bất thường và cách giả lập

#### **1. Quét cổng (Port Scanning)**

* **Mô tả**: Kẻ tấn công sử dụng công cụ để quét các cổng (ports) trên một thiết bị mục tiêu nhằm tìm các cổng mở (open ports). Điều này thường là bước đầu tiên của một cuộc tấn công, giúp kẻ tấn công xác định các dịch vụ đang chạy (như HTTP, SSH) để khai thác.
* **Dấu hiệu**:
  + Lưu lượng mạng tăng đột biến từ một IP (như 192.168.1.100) với nhiều yêu cầu kết nối đến các cổng khác nhau trên mục tiêu (như 192.168.1.1).
  + Ví dụ: IP 192.168.1.100 gửi hàng trăm yêu cầu đến các cổng 22, 80, 443, v.v. trong thời gian ngắn.
* **Tại sao bất thường?**: Hành vi quét cổng không phải là hoạt động bình thường của người dùng hợp lệ, mà thường là dấu hiệu của kẻ tấn công đang thu thập thông tin.
* **Giả lập**:
  + **Công cụ**: Dùng nmap (Network Mapper), một công cụ quét mạng phổ biến.

**Cài đặt**:  
sudo apt update

sudo apt install nmap

* + **Thực hiện**:
    1. Đảm bảo máy ảo tấn công có IP 192.168.1.100 và wifi access point (AP) có IP 192.168.1.1.

Chạy lệnh nmap để quét cổng trên 192.168.1.1:  
nmap -p- 192.168.1.1

* + - * -p-: Quét tất cả các cổng (1-65535).
      * 192.168.1.1: IP mục tiêu (wifi access point).

Để tăng tính bất thường, quét nhanh hơn (aggressive scan):  
nmap -T4 -p- 192.168.1.1

* + - * -T4: Tăng tốc độ quét (từ 1 đến 5, 4 là nhanh).
    1. Kết quả: nmap sẽ gửi hàng trăm/thousands gói tin đến các cổng trên 192.168.1.1, tạo lưu lượng bất thường. Dữ liệu này sẽ được ghi vào bộ nhớ của AP (như bảng kết nối mạng, trích xuất bằng Volatility: linux\_netstat).

#### **2. Giả mạo ARP (ARP Spoofing)**

* **Mô tả**: Kẻ tấn công gửi các gói tin ARP Reply giả mạo, làm cho các thiết bị trong mạng tin rằng IP của một thiết bị hợp lệ (như router 192.168.1.1) ánh xạ với MAC của kẻ tấn công. Điều này cho phép kẻ tấn công nghe lén lưu lượng mạng (man-in-the-middle attack).
* **Dấu hiệu**:
  + Trong bảng ARP, một IP (như 192.168.1.1) ánh xạ với nhiều MAC khác nhau, hoặc ánh xạ với MAC không hợp lệ.
  + Ví dụ: Router 192.168.1.1 đáng lẽ có MAC 00:AA:BB:CC:DD:EE, nhưng bảng ARP lại cho thấy MAC của 192.168.1.100 (00:XX:YY:ZZ:11:22).
* **Tại sao bất thường?**: ARP spoofing là hành vi cố ý giả mạo, không xảy ra trong hoạt động mạng bình thường.
* **Giả lập**:
  + **Công cụ**: Dùng arpspoof, một phần của bộ công cụ dsniff.

**Cài đặt**:  
sudo apt update

sudo apt install dsniff

* + **Thực hiện**:
    1. Đảm bảo máy ảo tấn công có IP 192.168.1.100, router (hoặc AP) có IP 192.168.1.1, và có một thiết bị nạn nhân (ví dụ: 192.168.1.2).

Bật IP forwarding để máy tấn công có thể chuyển tiếp gói tin (tránh làm gián đoạn mạng):  
sudo sysctl -w net.ipv4.ip\_forward=1

* + 1. Chạy arpspoof để giả mạo ARP:

Làm cho nạn nhân (192.168.1.2) tin rằng router (192.168.1.1) là máy tấn công:  
sudo arpspoof -i eth0 -t 192.168.1.2 192.168.1.1

* + - * + -i eth0: Giao diện mạng (thay eth0 bằng giao diện của bạn, xem bằng ifconfig).
        + -t 192.168.1.2: IP của nạn nhân.
        + 192.168.1.1: IP của router/AP.

Đồng thời, làm cho router tin rằng nạn nhân là máy tấn công:  
sudo arpspoof -i eth0 -t 192.168.1.1 192.168.1.2

* + 1. Kết quả: Máy tấn công (192.168.1.100) sẽ gửi ARP Reply giả mạo, làm thay đổi bảng ARP của các thiết bị trong mạng. Khi dump bộ nhớ của AP và trích xuất bảng ARP (Volatility: linux\_arp), bạn sẽ thấy IP 192.168.1.1 ánh xạ với MAC của 192.168.1.100.

#### **3. Tấn công từ chối dịch vụ (DoS - Denial of Service)**

* **Mô tả**: Kẻ tấn công gửi một lượng lớn yêu cầu (requests) đến một mục tiêu (như wifi access point) nhằm làm quá tải hệ thống, khiến nó không thể phục vụ người dùng hợp lệ.
* **Dấu hiệu**:
  + Một IP (như 192.168.1.100) gửi hàng nghìn gói tin trong thời gian ngắn đến một IP mục tiêu (như 192.168.1.1).
  + Ví dụ: Gửi hàng loạt gói SYN (SYN flood) đến cổng 80 của mục tiêu.
* **Tại sao bất thường?**: Lưu lượng mạng tăng đột biến từ một IP duy nhất không phải là hành vi bình thường, mà là dấu hiệu của tấn công.
* **Giả lập**:
  + **Công cụ**: Dùng hping3, một công cụ gửi gói tin tùy chỉnh.

**Cài đặt**:  
sudo apt install hping3

* + **Thực hiện**:

Từ máy tấn công (192.168.1.100), gửi hàng loạt gói SYN (SYN flood) đến cổng 80 của AP (192.168.1.1):  
sudo hping3 -S -p 80 --flood 192.168.1.1

* + - * -S: Gói SYN (dùng cho SYN flood).
      * -p 80: Cổng mục tiêu (cổng 80 - HTTP).
      * --flood: Gửi gói tin nhanh nhất có thể.
      * 192.168.1.1: IP mục tiêu.
    1. Kết quả: hping3 sẽ tạo lưu lượng lớn từ 192.168.1.100 đến 192.168.1.1, làm tăng số lượng gói tin trong bộ nhớ. Khi dump và trích xuất bằng Volatility (linux\_netstat), bạn sẽ thấy hàng nghìn kết nối từ 192.168.1.100.

#### **4. Truy cập không hợp lệ (Unauthorized Access)**

* **Mô tả**: Kẻ tấn công cố gắng truy cập vào một dịch vụ (như SSH, FTP) bằng cách thử nhiều mật khẩu (brute force) hoặc khai thác lỗ hổng.
* **Dấu hiệu**:
  + Nhiều yêu cầu đăng nhập thất bại từ một IP (như 192.168.1.100) đến một cổng (như cổng 22 - SSH).
  + Ví dụ: IP 192.168.1.100 gửi 500 yêu cầu đăng nhập SSH trong 1 phút.
* **Tại sao bất thường?**: Người dùng hợp lệ không thử đăng nhập liên tục với tần suất cao như vậy.
* **Giả lập**:
  + **Công cụ**: Dùng hydra, một công cụ tấn công brute force.

**Cài đặt**:  
sudo apt install hydra

* + **Thực hiện**:
    1. Chuẩn bị một file danh sách mật khẩu (password\_list.txt) để thử.

Từ máy tấn công (192.168.1.100), dùng hydra để tấn công brute force SSH trên AP (192.168.1.1):  
hydra -l admin -P password\_list.txt ssh://192.168.1.1

* + - * -l admin: Tên người dùng (giả sử là admin).
      * -P password\_list.txt: File chứa danh sách mật khẩu.
      * ssh://192.168.1.1: Mục tiêu (SSH trên 192.168.1.1).
    1. Kết quả: hydra sẽ gửi nhiều yêu cầu đăng nhập từ 192.168.1.100 đến cổng 22 của 192.168.1.1. Khi dump bộ nhớ, bạn sẽ thấy nhiều kết nối đến cổng 22 từ 192.168.1.100 (trích xuất bằng Volatility: linux\_netstat).

### **Lưu ý quan trọng**

* **Thời điểm giả lập**: Thực hiện các hành vi trên ngay trước khi dump bộ nhớ để dữ liệu bất thường được ghi vào file .dump. Ví dụ: Chạy nmap hoặc arpspoof trong 1-2 phút, sau đó dump ngay bằng LiME hoặc dd.
* **Môi trường**: Đảm bảo mạng mô phỏng đã được thiết lập đúng:
  + Máy tấn công: 192.168.1.100.
  + Wifi access point: 192.168.1.1.
  + Nạn nhân (nếu cần): 192.168.1.2.
* **Dấu hiệu trong bộ nhớ**:
  + Quét cổng/DoS/Truy cập không hợp lệ: Xem kết nối mạng bằng Volatility (linux\_netstat).
  + Giả mạo ARP: Xem bảng ARP bằng Volatility (linux\_arp).