**Nmap Advanced Port Scans**

**Giới thiệu**

Bài này giải thích những loại quét nâng cao hơn. Có thể hữu ích với những hệ thống riêng biệt, trong các thiết lập mạng cụ thể. Chúng ta sẽ đi qua những loại quét cổng sau:

* Null Scan
* FIN Scan
* Xmas Scan
* Maimon Scan
* ACK Scan
* Window Scan
* Custom Scan

Hơn nữa, chúng ta sẽ bao gồm những điều sau đây:

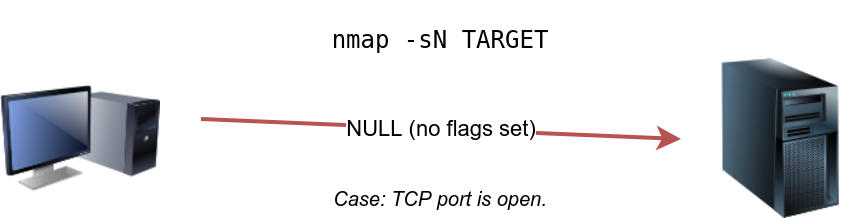
* Spoofing IP
* Spoofing MAC
* Decoy Scan
* Fragmented Packets
* Idle/Zombie Scan

Chúng ta sẽ thảo luận những lựa chọn và kĩ thuật để tránh tường lửa và những hệ thống IDS. Chúng ta cũng xem qua những lựa chọn để có thêm thông tin chi tiết từ Nmap.

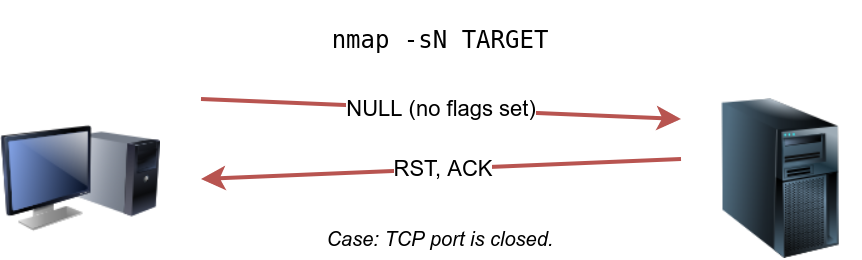
**TCP Null Scan, FIN Scan và Xmas Scan**

**Null Scan**

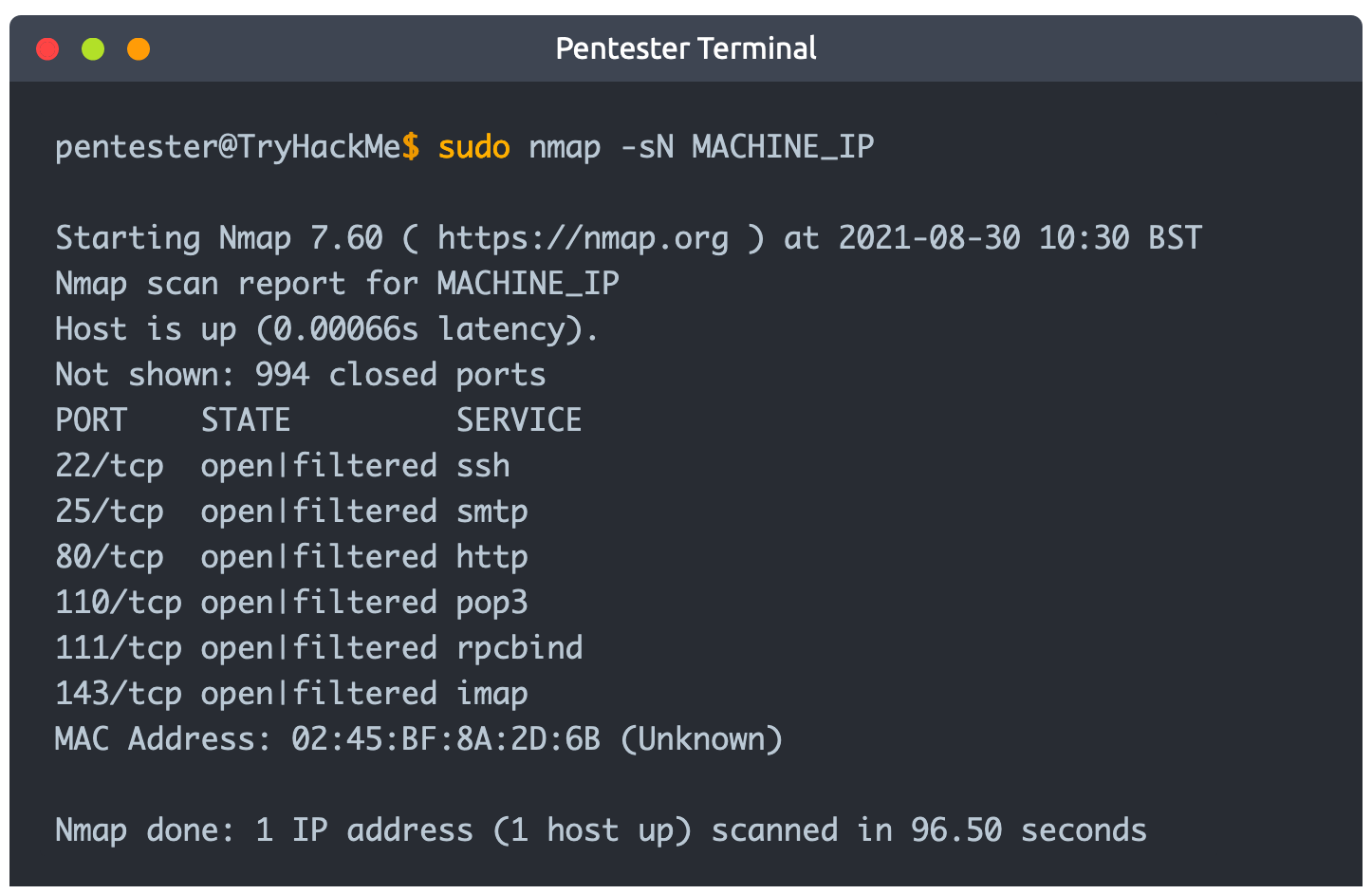
Null scan không đặt bất cứ flag nào. tất cả 6 bits flag đều đặt là 0. Bạn có thể chọn loại quét này bằng cách dùng -sN. 1 gói tin TCP không có flag sẽ không thực thi bất cứ phản hồi nào khi nó đi đến 1 cổng đang mở. Do đó, từ quan điểm của Nmap, thiếu phản hồi trong quá trình quét null chỉ ra rằng cổng đang mở hoặc tường lửa đang chặn gói tin đó.



Tuy nhiên, chúng ta mong đợi máy chủ đích phản hồi với gói tin RST nếu cổng đã đóng. Cuối cùng, chúng ta có thể tận dụng việc thiếu phản hồi RST để tìm ra các cổng chưa đóng: mở hoặc bị lọc.



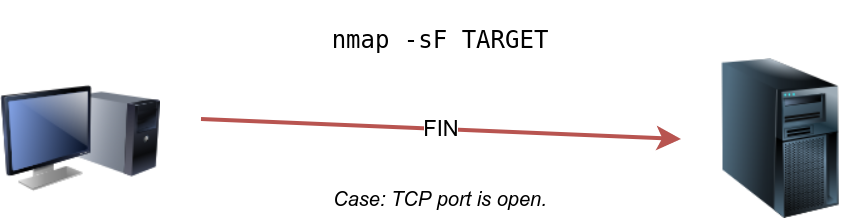
Vì null scan dựa vào việc thiếu phản hồi để chỉ ra cổng chưa đóng, nó không thể xác định chắc chắn cổng đó có mở không; có khả năng cổng đó không phản hồi vì tường lửa đã chặn gói tin.



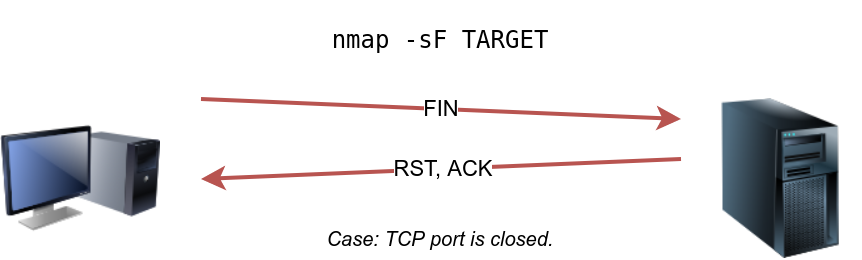
Lưu ý rất nhiều lựa chọn Nmap yêu cầu đặc quyền root. Nếu bạn không chạy Nmap như root, bạn cần dùng sudo.

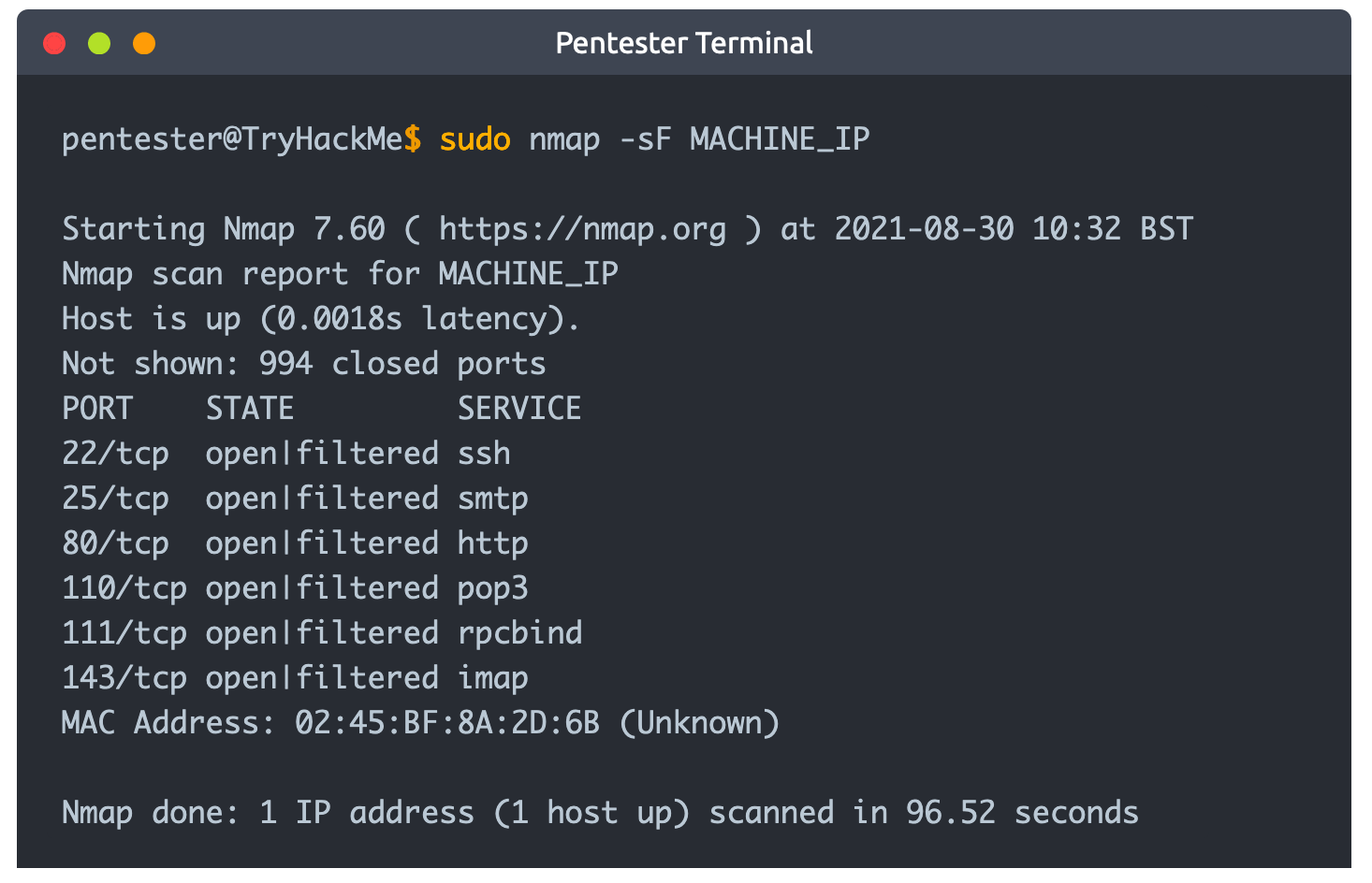
**FIN Scan**

FIN scan gửi 1 gói tin TCP với trạng thái FIN. Bạn có thể chọn loại quét này bằng cách dùng -sF. Tương tự, phản hồi sẽ không gửi đi nếu cổng TCP đang mở. Nmap không thể chắc chắn cổng đó có mở không hoặc bị tường lửa chặn.



Tuy nhiên hệ thống sẽ phản hồi với RST nếu cổng đã đóng. Điều đáng lưu ý là một vài tường lửa sẽ âm thầm drop lưu lượng mà không gửi RST.

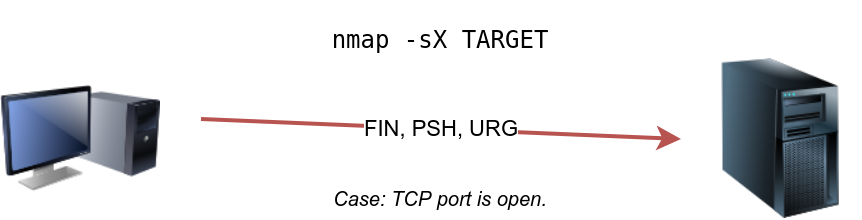


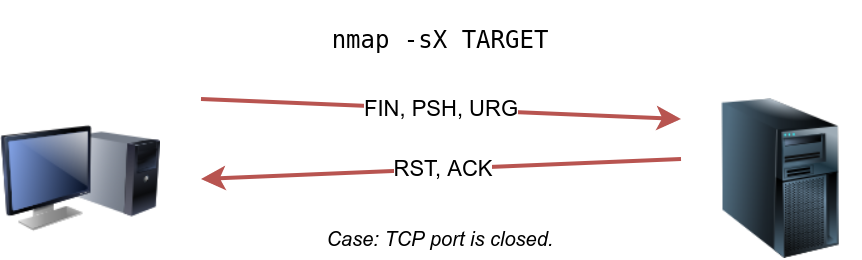


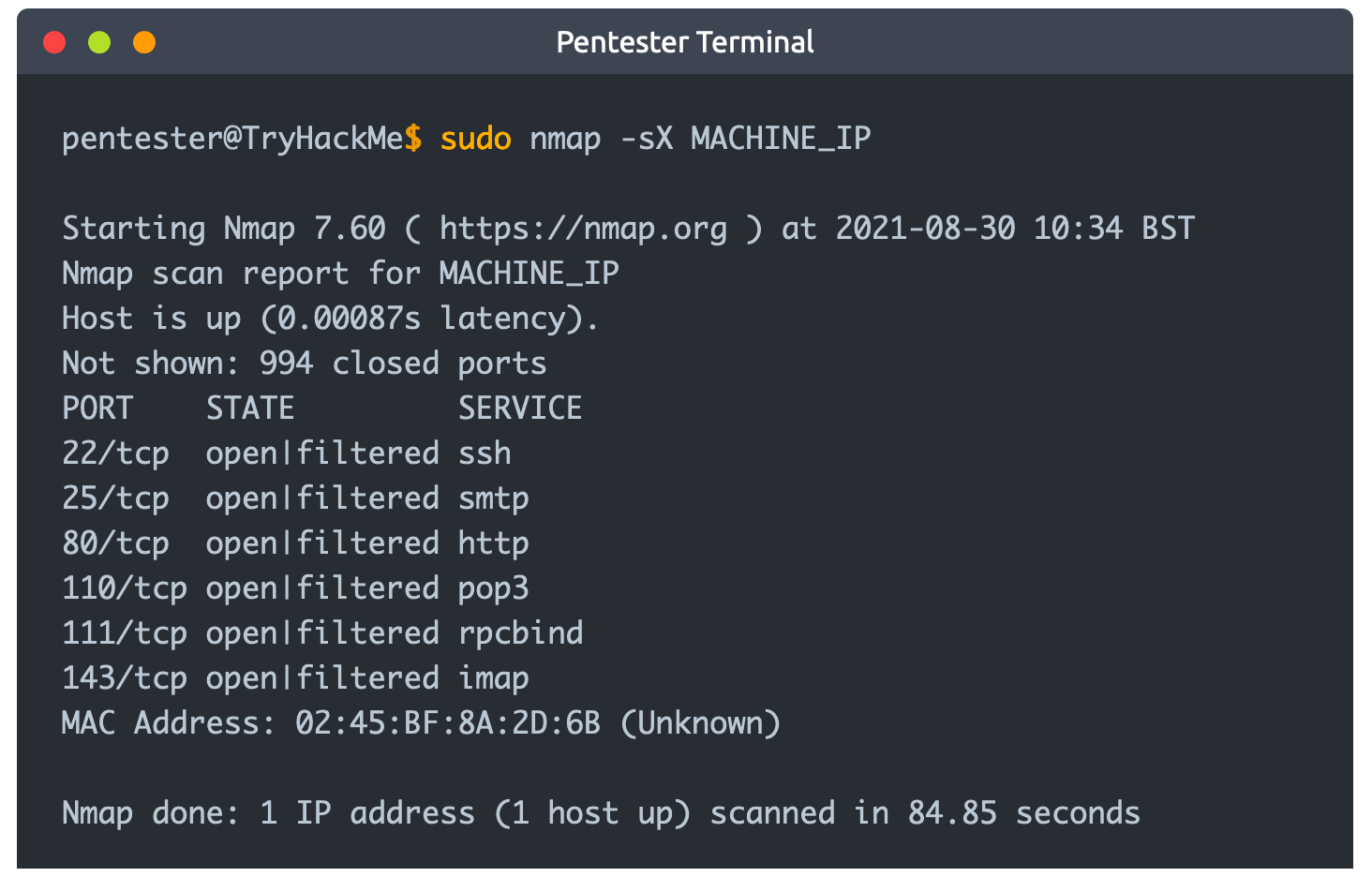
**Xmas Scan**

Xmas scan đặt đồng thời các trạng thái FIN, PSH, và URG. Bạn có thể chon Xmas scan với -sX

Như Null scan và FIN scan, nếu 1 gói tin RST được nhận cũng có nghĩa là cổng đã bị đóng. Ngược lại, nó sẽ thông báo open|filtered.





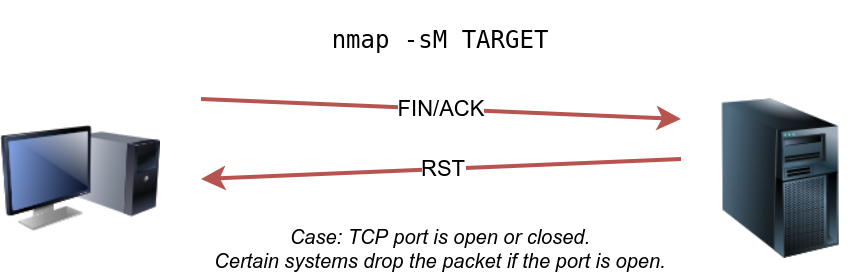


Ba loại quét này có thể rất hiệu quả khi quét 1 đối tượng đằng sau tường lửa phi trạng thái. Tường lửa phi trạng thái sẽ kiểm tra xem gói tin đến có syn flag để thiết lập kết nối không. Sử dụng những flag không phải là SYN có thể đánh lừa được tường lửa và chạm đến hệ thống đằng sau nó. Tuy vậy, 1 tường lửa có trạng thái trên thực tế sẽ chặn tất cả những gói tin như vậy và khiến kiểu quét này trở lên vô dụng.

**TCP Maimon Scan**

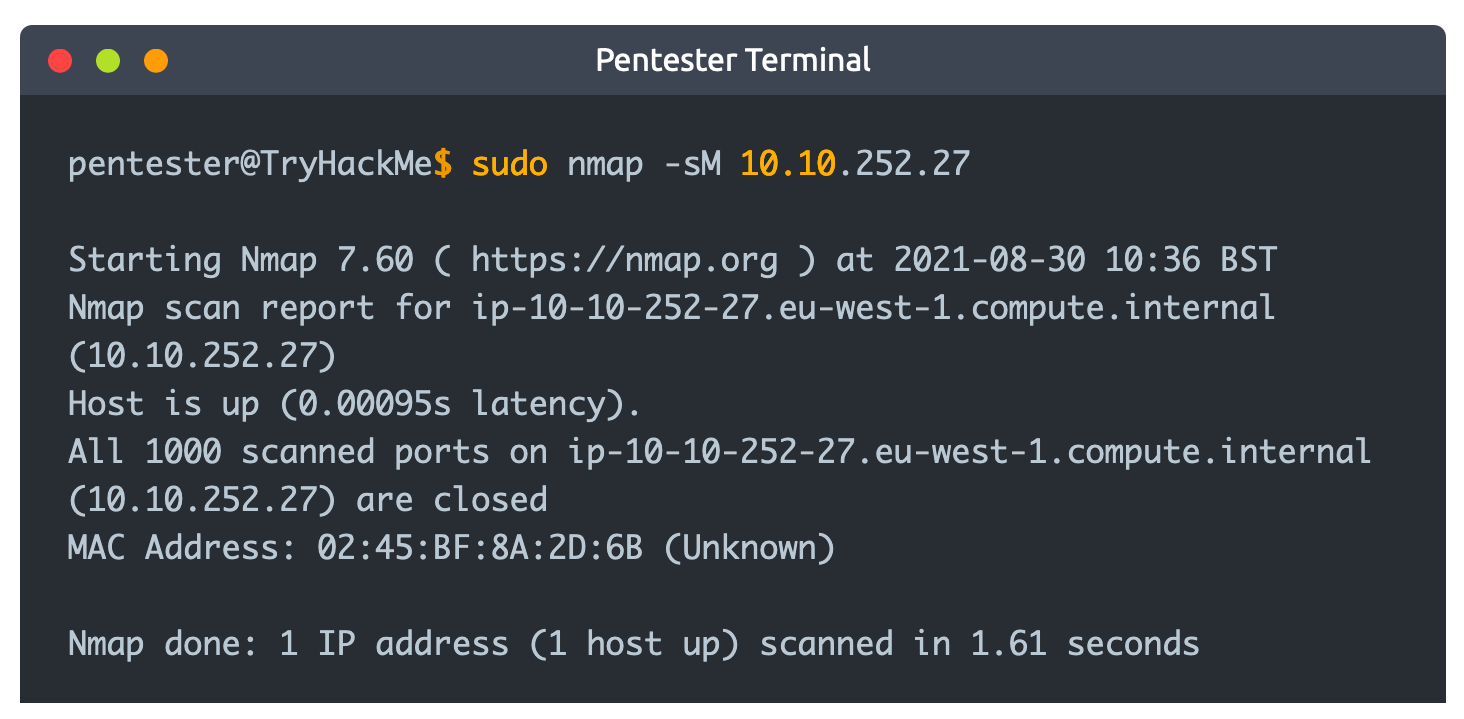
Uriel Maimon lần đầu tiên đã mô tả cách quét này năm 1996. FIN và ACK được đặt. Mục tiêu sẽ phản hồi RST. Tuy nhiên, những hệ thống bắt nguồn từ BSD nhất định bỏ qua gói tin này nếu nó là 1 cổng đang mở. Cách quét này sẽ không hoạt động trên đa số các đối tượng trong mạng hiện đại; Tuy nhiên, chúng ta vẫn sẽ học để hiểu rõ hơn về cơ chế quét cổng và tư duy hack. Để chọn loại quét này, dùng -sM.

Đa số hệ thống đích phản hồi với gói tin RST bất kể cổng TCP có mở hay không. Trong trường hợp như vậy, chúng ta sẽ không thể khám phá những cổng đang mở.



Vì những cổng đang mở hay đóng đều hoạt động giống nhau, Maimon scan không thể khám phá bất cứ cổng nào đang mở trên hệ thống đích.

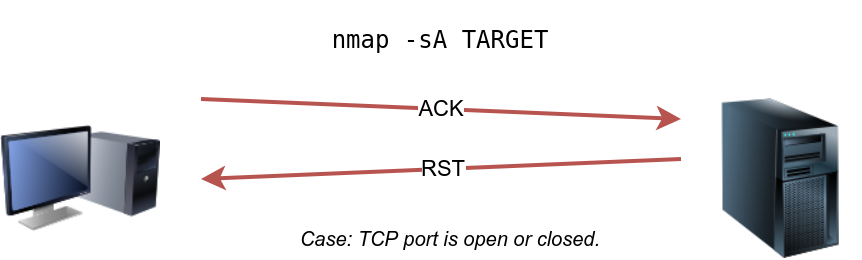
Loại quét này không phải là lần quét đầu tiên mà người ta chọn để khám phá hệ thống; Tuy nhiên, điều quan trọng là phải biết về nó vì bạn không biết khi nào nó có ích.



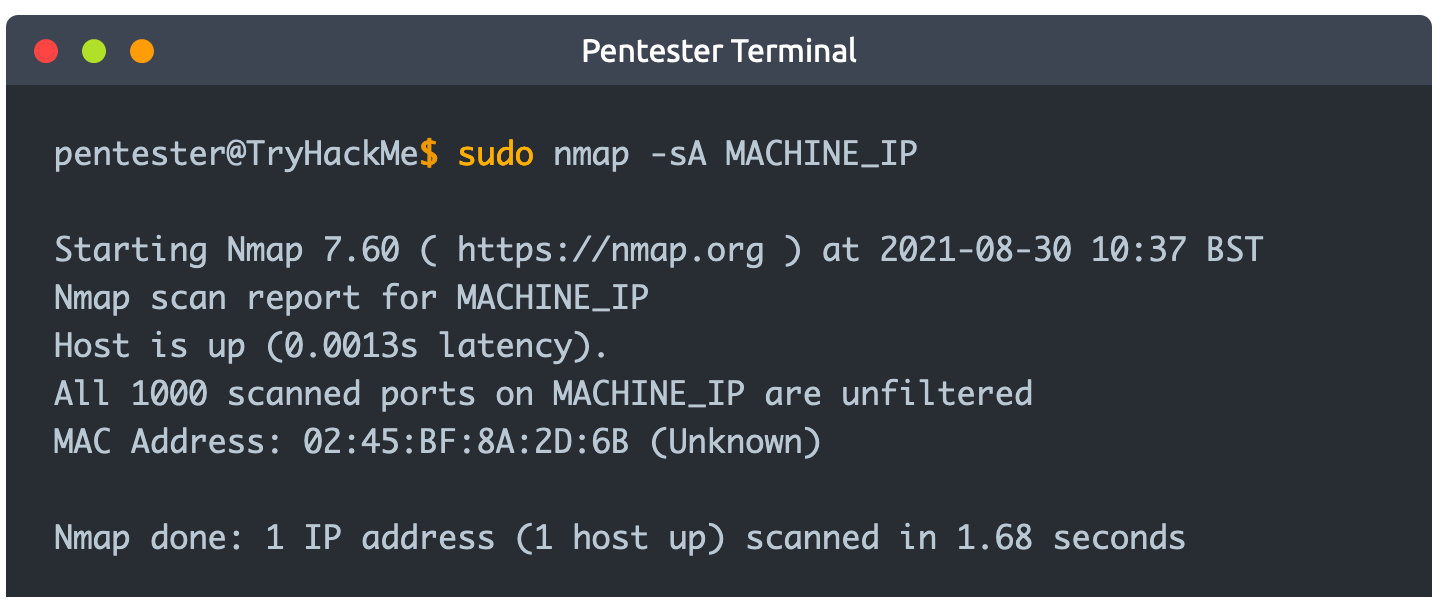
**TCP ACK, Window and Custom Scan**

**TCP ACK Scan:**

ACK scan sẽ gửi 1 gói tin TCP với flag ACK. Sử dụng -sA để scan. Đối tượng sẽ phản hồi RST bất kể trạng thái của cổng. Hành vi này xảy ra vì gói tin TCP với flag ACK chỉ được gửi để phản hồi gói tin TCP đã nhận để xác nhận. Do vậy, cách quét này không nói cho chúng ta cổng của đối tượng có mở không.

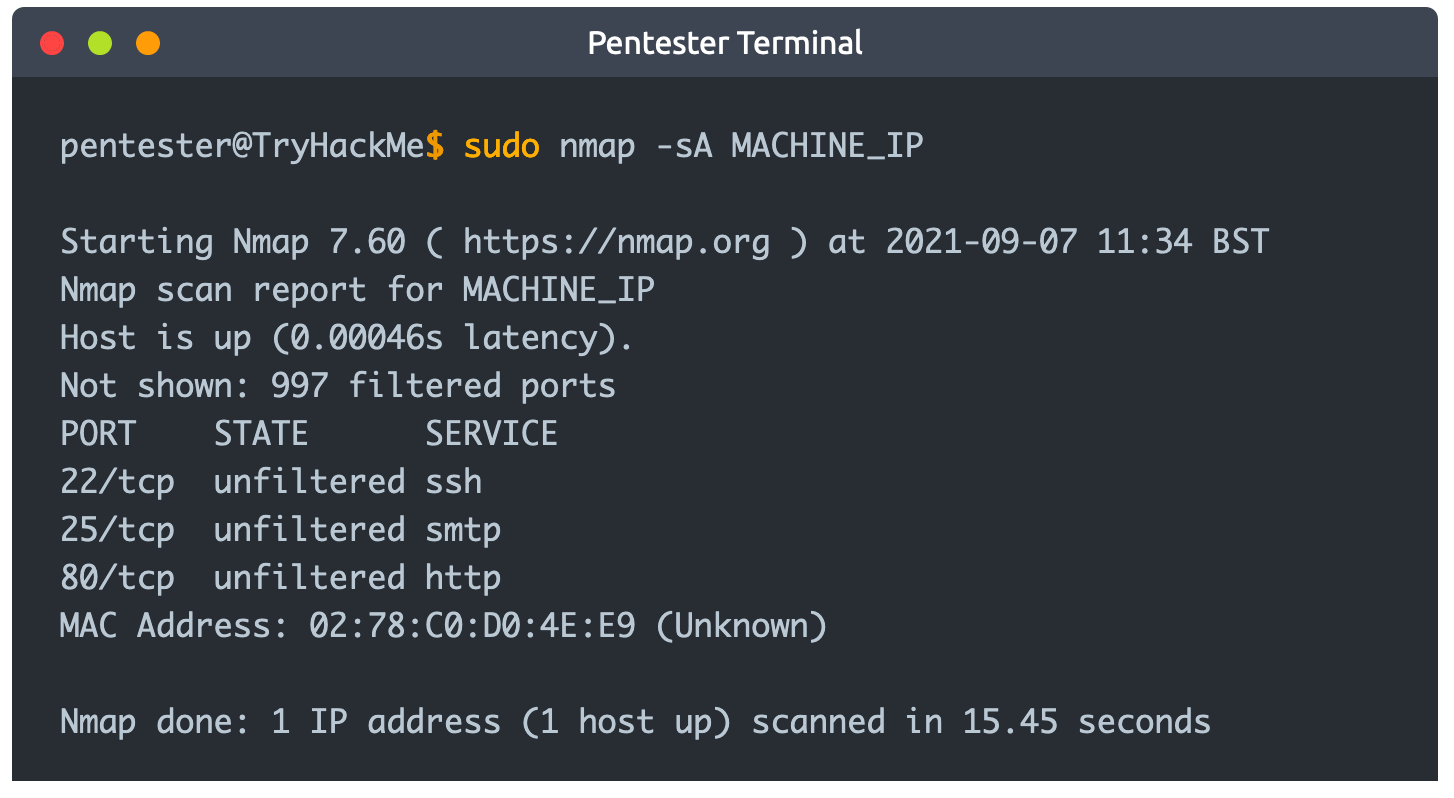


Trong ví dụ sau, chúng ta quét 1 đối tượng chưa cài tường lửa. Chúng ta không biết cổng nào đang mở.



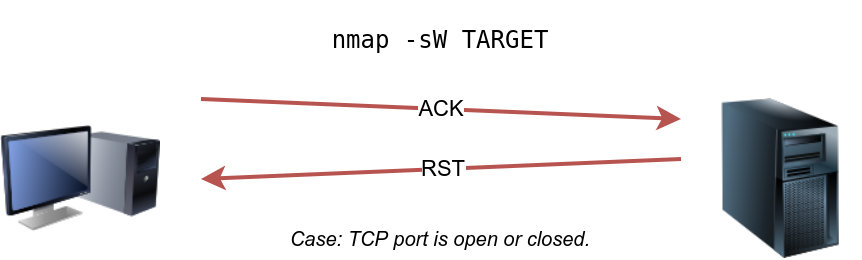
Loại quét này sẽ rất có ích nếu có tường lửa đặt trước đối tượng. Do đó, dựa trên các gói ACK được gửi đi, bạn sẽ biết cổng nào bị tường lửa chặn. Nói cách khác, cách quét này rất phù hợp để khám phá những qui tắc và cấu hình của tường lửa.

Sau khi thiết lập đối tượng với tường lửa, chúng ta lặp lại ACK scan. Chúng ta có 3 cổng không bị chặn.

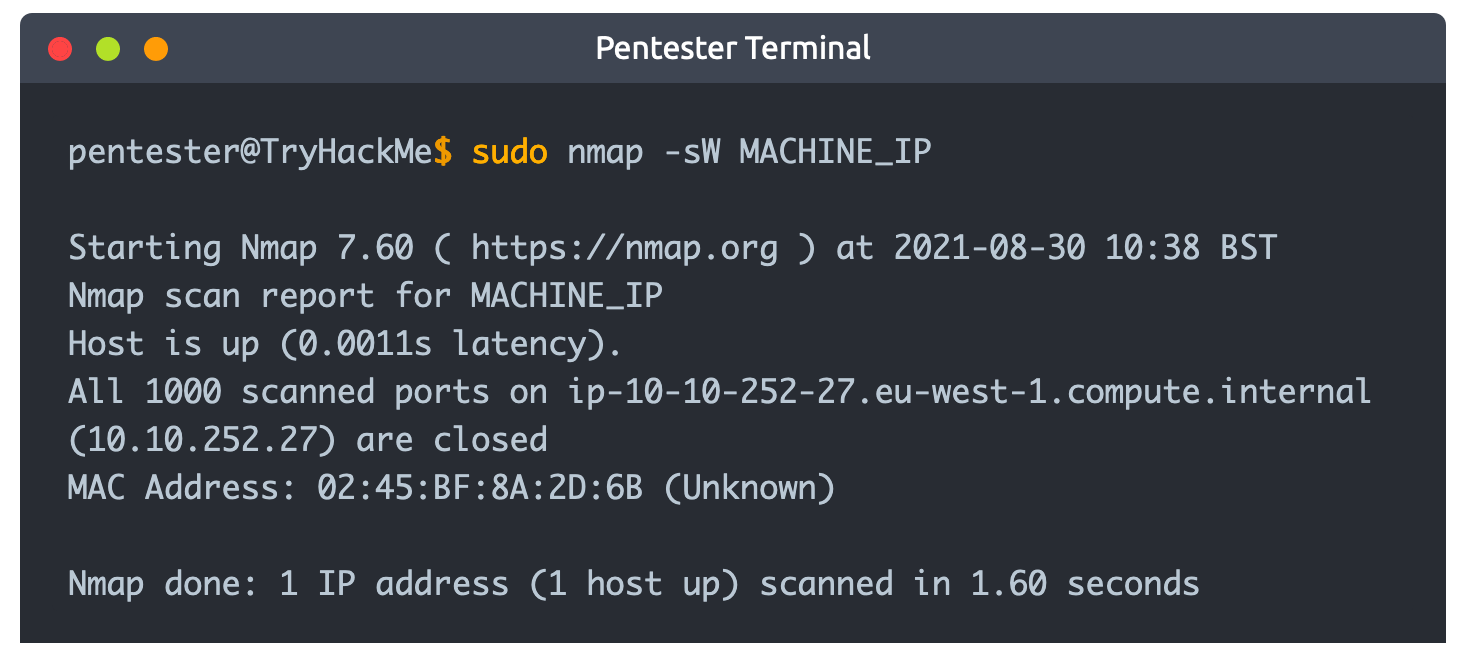


**Window Scan:**

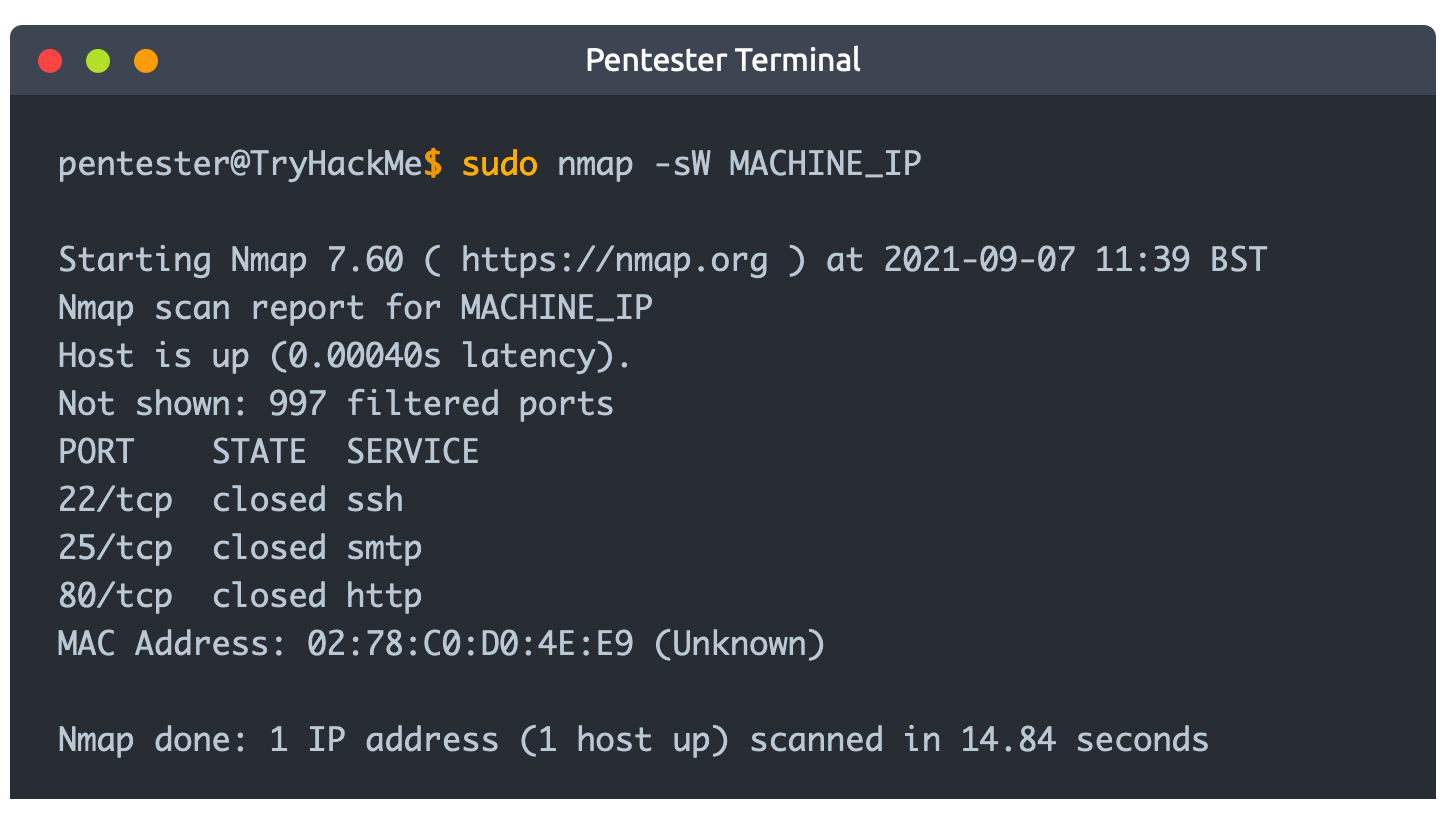
Một cách quét tương tự là TCP Window Scan. Tuy nhiên, nó kiểm tra miền TCP Window mà những gói tin RST trả về. Trên những hệ thống đặc biệt, nó có thể tiết lộ những cổng đang mở. Bạn có thể chọn loại quét này với -sW. Trong hình dưới, chúng ta mong đợi nhận được gói tin RST để phản hồi những gói tin ACK không mời, bất kể cổng có mở hay không.



Tương tự thực hiện TCP window scan với hệ thống Linux không có tường lửa sẽ không cung cấp nhiều thông tin.

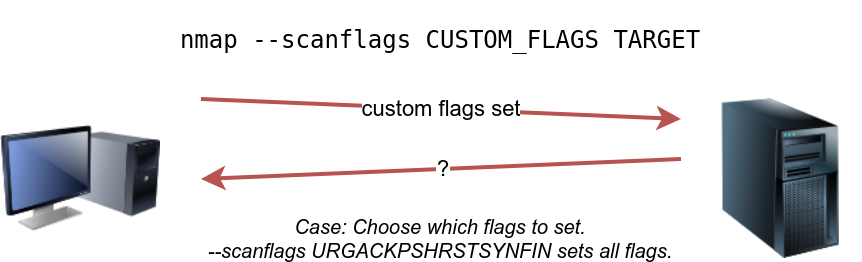


Tuy nhiên nếu chúng ta lặp lại quá trình quét cửa sổ TCP với máy chủ phía sau tường lửa, chúng ta sẽ nhận được nhiều kết quả thú vị hơn. TCP window scan chỉ ra 3 cổng đã đóng. Mặc dù chúng ta biết 3 cổng này không đóng, chúng ta biết chúng phản hồi rất khác nhau, chỉ ra rằng tường lửa không chặn chúng.



**Custom Scan:**

Nếu bạn muốn thử những flag TCP mới, bạn có thể dùng –scanflags. Ví dụ, nếu bạn muốn đặt SYN, RST và FIN cùng 1 lúc, bạn có thể làm như sau –scanflags RSTSYNFIN. Nếu bạn điều chỉnh quá trình quét, bạn cần biết những cổng khác nhau sẽ xử lí như thế nào để giải thích kết quả trong các tình huống khác nhau một cách chính xác.



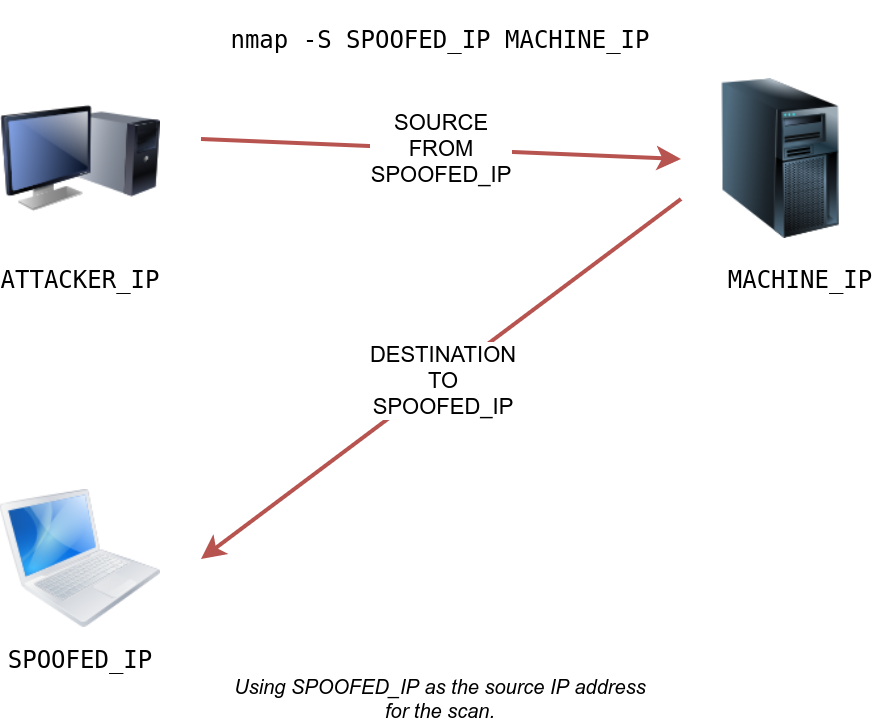
Cuối cùng, chú ý là ACK scan và window scan rất hiệu quả để giúp chúng ta khám phá được những qui tắc của tường lửa. Tuy nhiên, điều quan trọng cần nhớ là chỉ vì tường lửa không chặn 1 cổng cụ thể, điều đó không có nghĩa là 1 dịch vụ đang lắng nghe trên cổng đó. Ví dụ, những qui tắc của tường lửa cần được chỉnh sửa để phản ánh những thay đổi gần đây của dịch vụ. Do đó, ACK và Window scan đang phơi bày các qui tắc của tường lửa, không phải những dịch vụ.

**Spoofing and Decoys**

Trong một vài thiết lập mạng, bạn sẽ có thể quét 1 hệ thống đích bằng việc dùng 1 địa chỉ IP giả mạo và thậm chí là địa chỉ MAC giả mạo. Việc quét như vậy chỉ có lợi ích khi bạn có thể đảm bảo lấy được phản hồi. Nếu bạn cố gắng quét 1 đối tượng từ 1 vài mạng ngẫu nhiên bằng việc dùng 1 địa chỉ Ip giả mạo, rất có thể bạn không nhận được phản hồi nào và kết quả quét không đáng tin cậy.

nmap -S SPOOFED\_IP MACHINE\_IP

Nmap sẽ đóng gói tất cả gói tin bằng cách dùng địa chỉ IP nguồn SPOOFED\_IP. Đối tượng sẽ phản hồi đến địa chỉ IP đích SPOOFED\_IP. Để loại quét này hoạt động và đưa ra kết quả chính xác, kẻ tấn công cần theo dõi lưu lượng mạng để phân tích những phản hồi.



Tóm lại, việc quét với địa chỉ IP giả mạo có 3 bước:

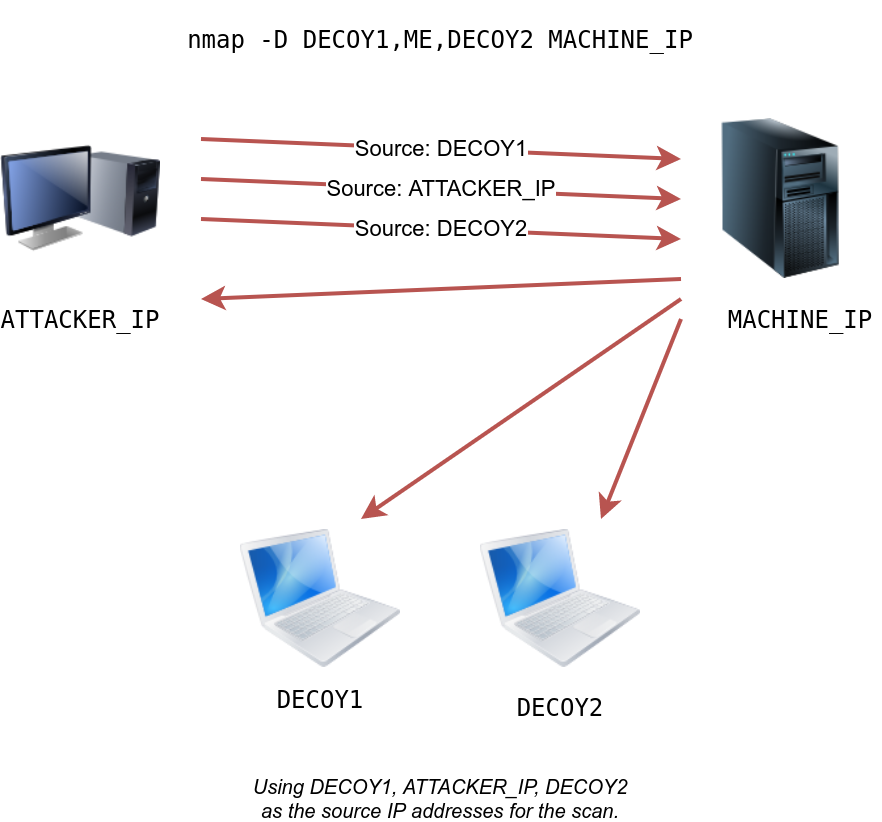
1. Kẻ tấn công gửi 1 gói tin với địa chỉ IP nguồn giả mạo đến đối tượng
2. Đối tượng phản hồi đến địa chỉ giả mạo
3. Kẻ tấn công thu thập những phản hồi để tìm ra những cổng đang mở

Nhìn chung, bạn mong muốn xác định giao diện mạng bằng -e và vô hiệu hóa ping scan -Pn. Do đó, thay vì nmap -S SPOOFED\_IP MACHINE\_IP, bạn sẽ cần phát hành nmap -e NET\_INTERFACE -Pn -S SPOOFED\_IP MACHINE\_IP để nói Nmap giao diện mạng sử dụng và không mong nhận được phản hồi ping. Điều đáng lưu ý là loại quét này sẽ vô ích nếu hệ thống kẻ tấn công không thể theo dõi mạng để lấy những phản hồi.

Khi bạn ở chung mạng con với đối tượng, bạn có thể giả mạo địa chỉ MAC luôn. Bạn có thể chỉ định địa chỉ MAC nguồn bằng cách dùng --spoof-mac SPOOFED\_MAC. Địa chỉ giả mạo này chỉ khả thi nếu kẻ tấn công và đối tượng cùng chung 1 mạng con (Ethernet hoặc Wifi)

Việc giả mạo chỉ hoạt động trong 1 số trường hợp tối thiểu khi điều kiện phù hợp. Do đó, kẻ tấn công có thể dùng mồi nhử để khiến việc xác định chính xác trở nên khó khăn hơn. Khái niệm rất đơn giản, làm cho quá trình quét dường như đến từ rất nhiều địa chỉ IP để mà địa chỉ IP của kẻ tấn công sẽ bị mất trong số họ.

Hình bên dưới cho thấy quá trình quét máy ảo sẽ đến từ 3 nguồn khác nhau và kết quả là những phản hồi cũng sẽ đến các mồi nhử luôn.



Bạn có thể dùng mồi nhử quét bằng cách chỉ định 1 địa chỉ IP xác định hay ngẫu nhiên sau -D. Ví dụ, nmap -D 10.10.0.1,10.10.0.2, ME MACHINE\_IP

nmap -D 10.10.0.1,10.10.0.2,RND,RND,ME MACHINE\_IP

Địa chỉ IP nguồn thứ 3, 4 được gán ngẫu nhiên, trong khi nguồn thứ 5 sẽ là địa chỉ IP của kẻ tấn công.

**Fragmented Packets**

**Firewall:**

Tường lửa là 1 phần mềm hay phần cứng cho phép những gói tin được đi qua hay chặn nó. Ví dụ, bạn có thể chặn tất cả lưu lượng đến máy chủ của mình ngoại trừ những lưu lượng đến máy chủ web. Một tường lửa truyền thống sẽ kiểm tra IP header và header của lớp vận chuyển. Tường lửa phức tạp hơn thì kiểm tra dữ liệu được thực hiện bởi lớp vận chuyển.

**IDS:**

Hệ thống phát hiện xâm nhập (IDS) kiểm tra các gói mạng để tìm kiếm những hành vi và nội dung phù hợp. Nó sẽ cảnh bảo khi có hành vi độc hại. Ngoài header của IP và lớp vận chuyển, IDS cũng sẽ điều tra dữ liệu trong lớp vận chuyển để phát hiện những hành vi độc hại.

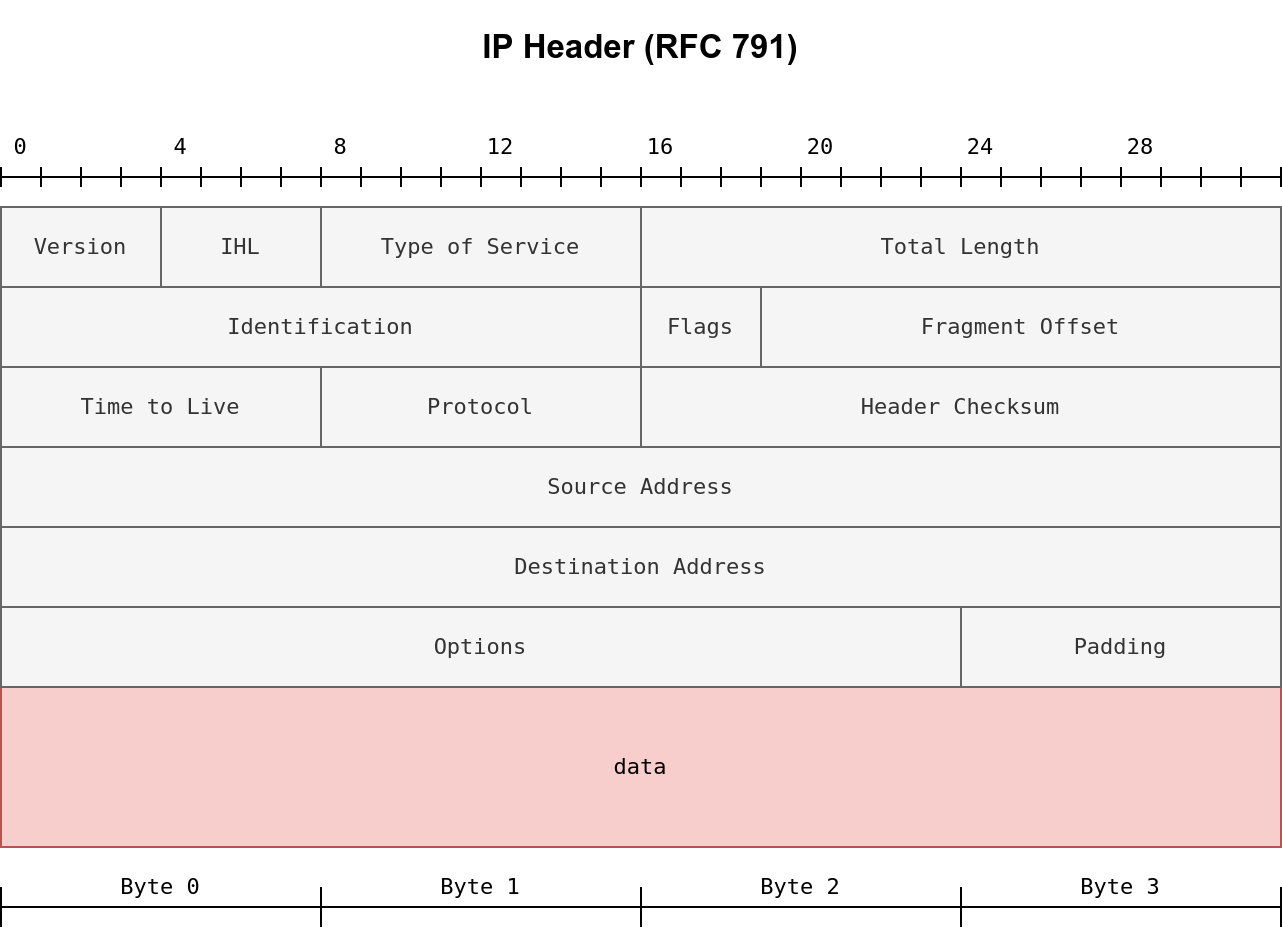
Bạn có thể làm cho tường lửa truyền thống khó phát hiện hoạt động của Nmap như thế nào? Rất khó để trả lời; Tuy nhiên, phụ thuộc vào loại tường lửa hay IDS, bạn có thể lợi dụng nó bằng cách chia nhỏ gói tin.

**Fragmented Packets:**

Nmap cung cấp lựa chọn -f để chia nhỏ những gói tin. Dữ liệu của IP sẽ được chia ra thành 8 bytes hoặc ít hơn. Thêm -f nữa (-f -f hoặc -ff) sẽ chia dữ liệu thành 16 bytes. Bạn có thể thay đổi giá trị mặc định bằng việc dùng --mtu; tuy nhiên, bạn nên chọn 8.

Để hiều hơn về sự phân chia, chúng ta cần xem IP header. Chú ý địa chỉ nguồn tốn 32 bits (4 bytes) ở dòng thứ 4, trong khi địa chỉ đích tốn thêm 4 bytes nữa. Dữ liệu chúng ta sẽ phân chia thành nhiều gói tin được tô đậm màu đỏ.

Để lắp ráp lại dữ liệu ở phía người nhận, IP sử dụng mã định danh (ID) và fragment offset ở dòng hai.



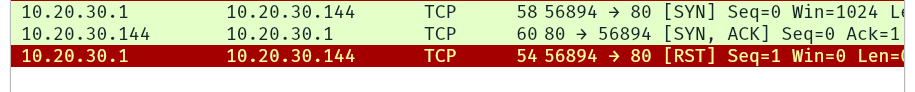
Cùng so sánh việc chạy

sudo nmap -sS -p80 10.20.30.144. (TCP SYN scan)

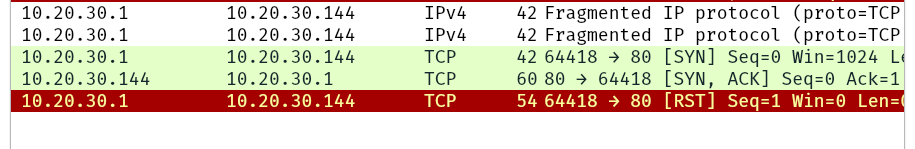
sudo nmap -sS -p80 -f 10.20.30.144

Nó sẽ quét TCP SYN trên cổng 80; Tuy vậy, ở câu lệnh thứ hai, chúng ta yêu cầu Nmap phân chia gói tin IP.

Ở hai dòng đầu tiên, chúng ta có thể thấy câu truy vấn ARP và phản hồi. Nmap phát hành câu truy vấn ARP vì đối tượng ở chung mạng Ethernet. Hai dòng tiếp theo hiển thị TCP SYN ping và phản hồi. Dòng thứ 5 bắt đầu quét cổng; Nmap gửi 1 gói tin TCP SYN đến cổng 80. Trong trường hợp này, IP header là 20 bytes và TCP header là 24 bytes. Lưu ý kích thước nhỏ nhất của TCP header là 20 bytes.



Với phân đoạn -f, 24 bytes của TCP header sẽ được chia ra thành nhiều đoạn 8 bytes. Chúng ta có 3 phân đoạn IP; mỗi cái có 20 bytes IP header và 8 bytes TCP header.



Nếu bạn thêm -ff, phân đoạn của dữ liệu sẽ là nhiều đoạn 16 bytes. Trong trường hợp này, sẽ chia thành 2 đoạn IP, cái thứ nhất chứa 16 bytes và cái thứ hai chứa 8 bytes TCP header.

Nói cách khác, nếu bạn thích tăng kích thước của những gói tin để làm cho chúng trở nên vô hại, bạn có thể dùng --data-length NUM, num chỉ định số lượng bytes bạn muốn .

**Idle/Zombie Scan:**

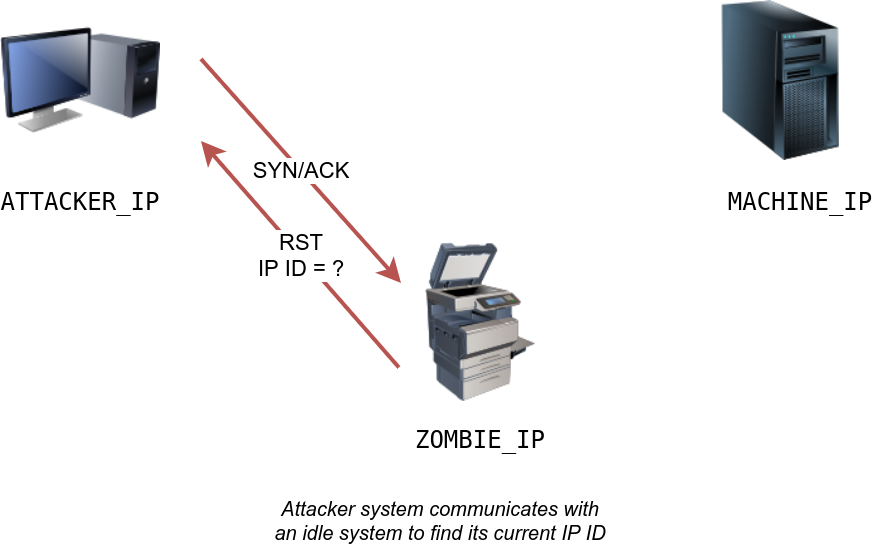
Làm giả địa chỉ IP nguồn có thể là 1 cách tiếp cận tốt để quét lén. Tuy nhiên, việc giả mạo chỉ có thể hoạt động ở một số mạng nhất định. Nó yêu cầu bạn phải ở vị trí mà bạn có theo dõi lưu lượng. Xem xét những giới hạn đó, việc làm giả địa chỉ IP có thể rất ít được sử dụng; Tuy nhiên, chúng ta có thể nâng cấp nó với quét nhàn rỗi (idle scan).

Idle scan hay zombie scan yêu cầu 1 hệ thống nhàn rỗi kết nối mạng mà bạn có thể giao tiếp. Thực tế, Nmap sẽ làm cho mỗi đầu dò xuất hiện như thể đến từ máy chủ nhàn rỗi, sau đó nó sẽ kiểm tra những chỉ số mà máy chủ đó nhận được. Nó được hoàn thành bằng việc kiểm tra mã định danh IP trong IP header. Bạn có thể chạy idle scan bằng việc dùng nmap -sI ZOMBIE\_IP MACHINE\_IP

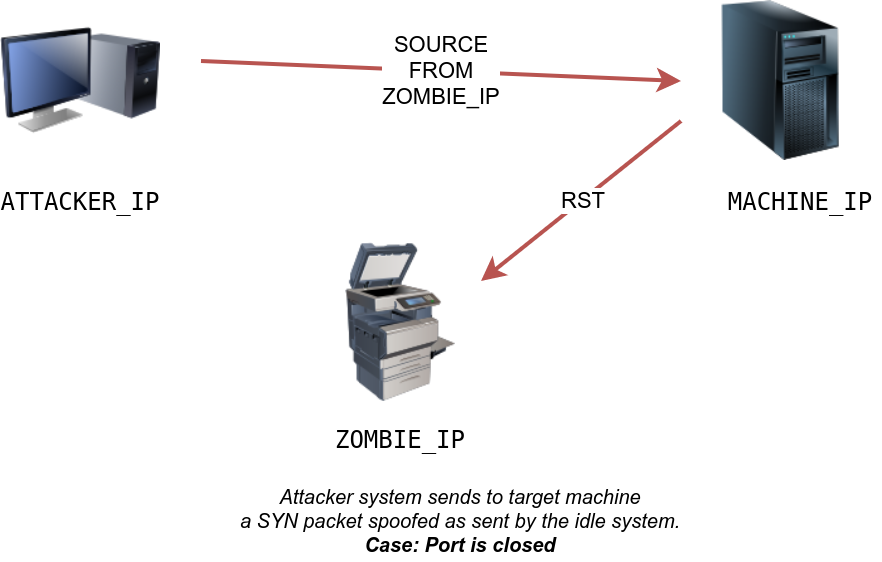
Idle scan yêu cầu làm theo 3 bước sau để khám phá những cổng đang mở:

1. Kích hoạt máy chủ nhàn rỗi để bạn có thể ghi lại IP ID hiện tại của nó.
2. Gửi 1 gói tin SYN đến 1 cổng TCP của đối tượng. Gói tin nên được làm giả để xuất hiện như thể đến từ địa chỉ IP của máy chủ zombie.
3. Kích hoạt máy ảo idle 1 lần nữa để bạn có thê so sánh IP ID mới với cái bạn đã nhận trước đó.

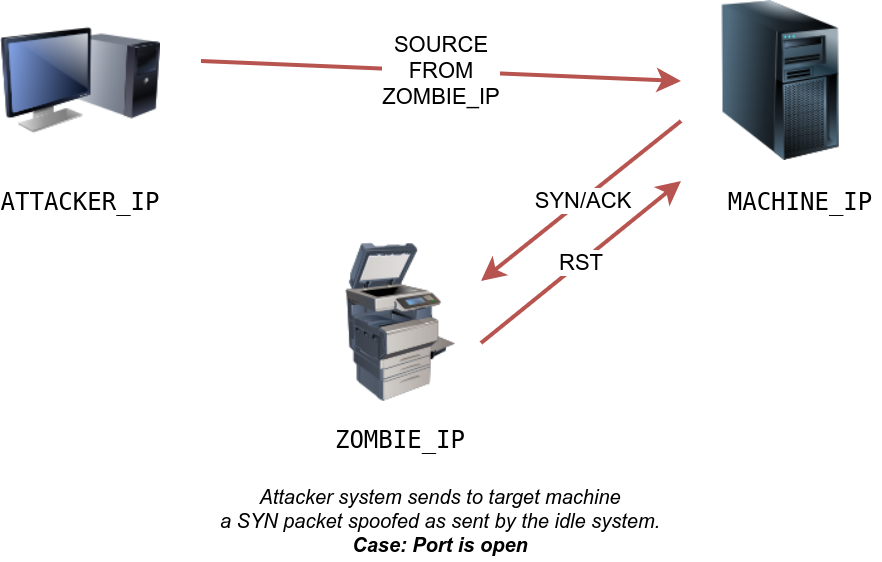
Trong hình dưới, chúng ta có hệ thống của kẻ tấn công đang thăm dò máy ảo idle, 1 máy in nhiều chức năng. Bằng việc gửi SYN/ACK, nó phản hồi với gói tin RST chứa IP ID mới.



Kẻ tấn công sẽ gửi 1 gói tin SYN đến cổng TCP của đối tượng. Tuy nhiên, gói tin này sẽ sử dụng địa chỉ IP của máy chủ idle. Ba tình huống sẽ xảy ra. Tình huống thứ 1, cổng TCP đã đóng; do đó, máy ảo phản hồi máy chủ idle với gói tin RST. Máy chủ idle không phản hồi nên IP ID của nó không tăng lên.



Trong tình huống thứ hai, cổng TCP đang mở vì vậy đối tượng phản hồi với SYN/ACK đến máy chủ idle. Máy chủ idle phản hồi với gói tin RST, do đó tăng IP ID.



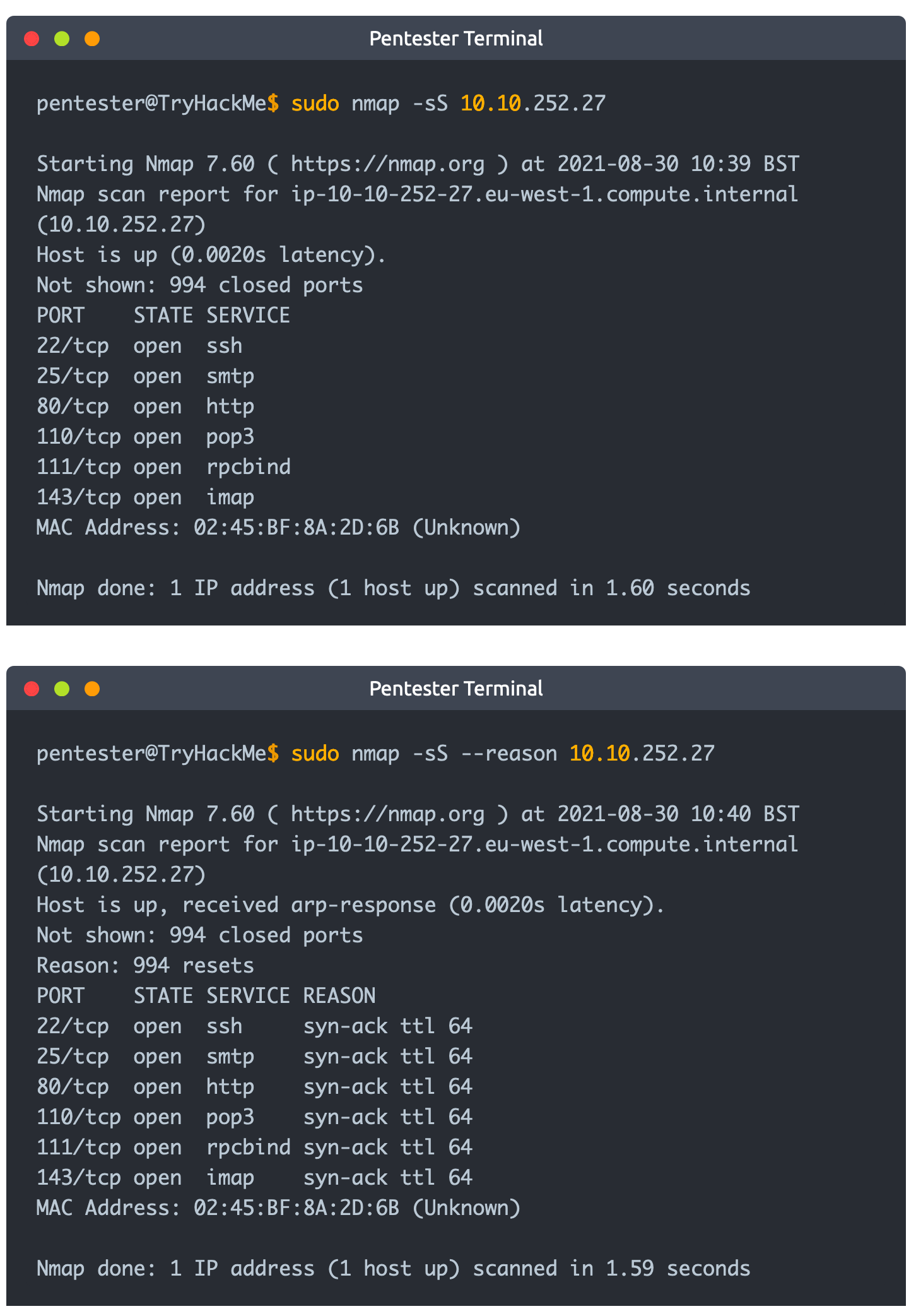
Trong tình huống thứ 3, đối tượng không phản hồi vì gói tin bị chặn bởi tường lửa. IP ID của máy chủ idle không tăng lên.

Bước cuối cùng, kẻ tấn công gửi SYN/ACK khác đến máy chủ idle. Máy chủ idle phản hồi với gói tin RST, tăng IP ID 1 lần nữa. Kẻ tấn công cần so sánh IP ID mới với cái đã nhận được trước đó. Nếu sự khác nhau là 1, điều này có nghĩa là cổng trên đối tượng bị đóng hoặc chặn. Tuy nhiên nếu sự khác nhau là 2 cũng có nghĩa là cổng trên đối tượng đang mở.

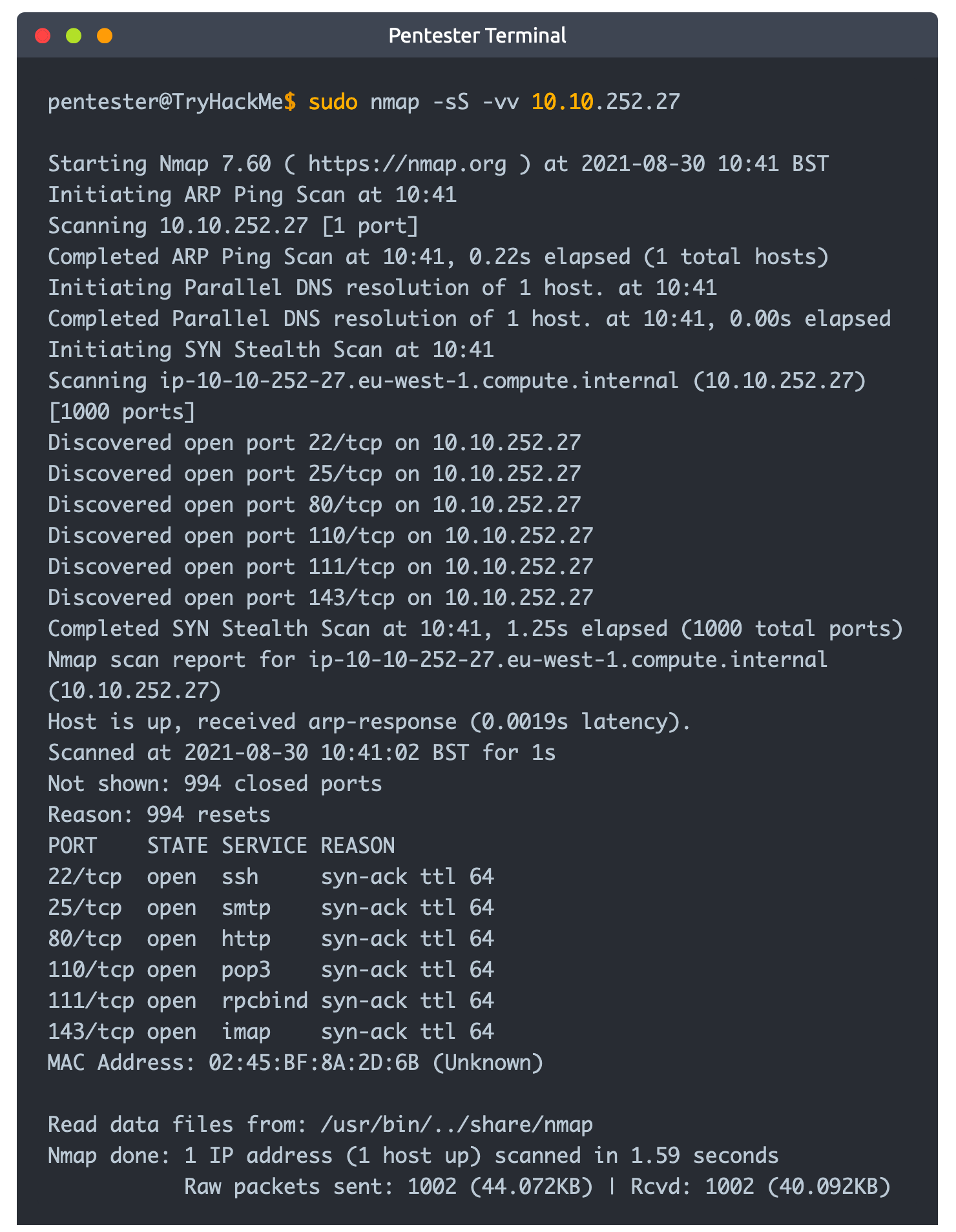
Điều đáng lưu ý là loại quét này được gọi là quét nhàn rỗi vì chọn 1 máy chủ nhàn rỗi là không thể thiếu với độ chính xác của loại quét này. Nếu máy chủ nhàn rỗi bị bận, tất cả IP IDs trả về sẽ vô ích.

**Getting More Details**

Bạn có thể thêm --reason nếu bạn muốn Nmap cung cấp thêm thông tin về lý do và kết luận của nó. Xem xét hai quá trình quét:



Để biết thêm thông tin chi tiết, bạn có thể dung -v hoặc -vv để rõ ràng hơn.



**Tổng hợp:**

| **Port Scan Type** | **Example Command** |
| --- | --- |
| TCP Null Scan | sudo nmap -sN MACHINE\_IP |
| TCP FIN Scan | sudo nmap -sF MACHINE\_IP |
| TCP Xmas Scan | sudo nmap -sX MACHINE\_IP |
| TCP Maimon Scan | sudo nmap -sM MACHINE\_IP |
| TCP ACK Scan | sudo nmap -sA MACHINE\_IP |
| TCP Window Scan | sudo nmap -sW MACHINE\_IP |
| Custom TCP Scan | sudo nmap --scanflags URGACKPSHRSTSYNFIN MACHINE\_IP |
| Spoofed Source IP | sudo nmap -S SPOOFED\_IP MACHINE\_IP |
| Spoofed MAC Address | --spoof-mac SPOOFED\_MAC |
| Decoy Scan | nmap -D DECOY\_IP,ME MACHINE\_IP |
| Idle (Zombie) Scan | sudo nmap -sI ZOMBIE\_IP MACHINE\_IP |
| Fragment IP data into 8 bytes | -f |
| Fragment IP data into 16 bytes | -ff |

| **Option** | **Purpose** |
| --- | --- |
| --source-port PORT\_NUM | specify source port number |
| --data-length NUM | append random data to reach given length |

| **Option** | **Purpose** |
| --- | --- |
| --reason | explains how Nmap made its conclusion |
| -v | verbose |
| -vv | very verbose |
| -d | debugging |
| -dd | more details for debugging |