NMAP TOOL

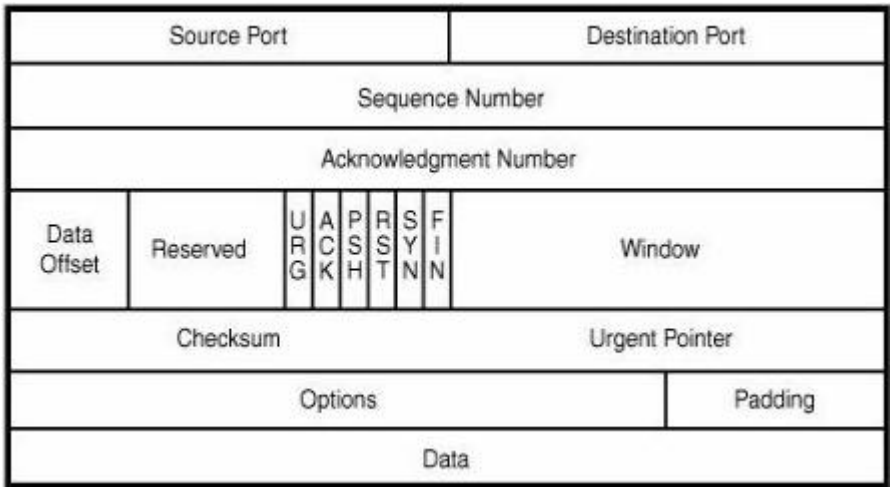
I. Giới thiệu

Nmap là công cụ bảo mật được phát triển bởi Floydor, ban đầu nó chỉ là một tool*nix nhưng về sau đã phát triển mạnh mẽ phù hợp với nhiều platform và phát triển cả giao diện.

Nmap là phần mềm mã nguồn mở miễn phí khai thác mạng và kiểm tra nhiều hệ thống và cũng như tài khoản người dùng . Nmap rất có ích trong việc giám sát các host hoặc các dịch vụ cập nhật thời gian. Nmaps sử dụng gói IP để xác định các host trên một mạng như hệ điều hành đang sử dụng, các gói filters/firewall đang sử dụng .

II. Nguyên tắc truyền thông TCP

1. Cấu tạo gói TCP



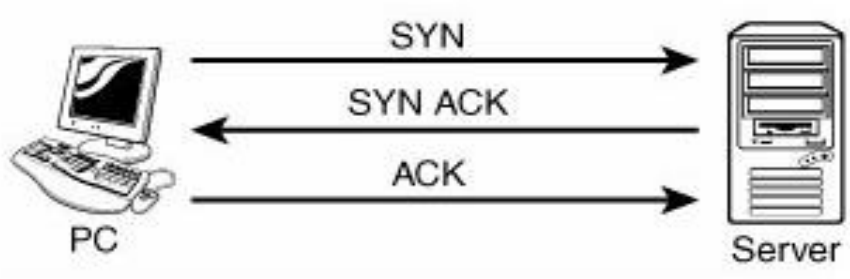
Ta chỉ quan tâm tới cờ Flag trong gói tin TCP nhằm mục đích sử dụng Scan Port:

- Thông số SYN để yêu cầu kết nối giữa hai máy tính
- Thông số ACK để trả lời kết nối giữa hai máy có thể bắt đầu được thực hiện

- Thông số FIN để kết thúc quá trình kết nối giữa hai máy
- Thông số RST từ Server để nói cho Client biết rằng giao tiếp này bị cấm(không thể sử dụng)
- Thông số PSH sử dụng kết hợp với thông số URG
- Thông số URG sử dụng để thiết lập độ ưu tiên cho gói tin này

  Thực tế các thông số này trong gói tin nó chỉ thể hiện là 1 hoặc 0 nếu 0 thì gói tin TCP không thiết lập thông số. Nếu là 1 thì thông số náo đó được thực hiện nó sẽ lần lượt trong 8 bits trong phần Flag.

2. **Khi Client muốn thực hiện kết nối TCP tới Server**



- **Bước 1:** Client gửi đến Server một gói tin SYN để yêu cầu kết nối
- **Bước 2:** Server trả lời Client một gói tin SYN/ACK
- **Bước 3:** Khi Client nhận được gói tin SYN/ACK sẽ gửi lại Server một gói ACK và quá trình trao đổi thông tin giữa hai máy bắt đầu

3. **Khi Client muốn kết thúc một phiên làm việc với Server**

- **Bước 1:** Client gửi đến Server một gói tin FIN ACK
- **Bước 2:**Server gửi lại cho Client một gói tin ACK
- **Bước 3:**Server lại gửi cho Client một gói FIN ACK
- **Bước 4:** Client gửi lại cho Server gói ACK và quá trình ngắt kết nối giữa Server và Client được thực hiện

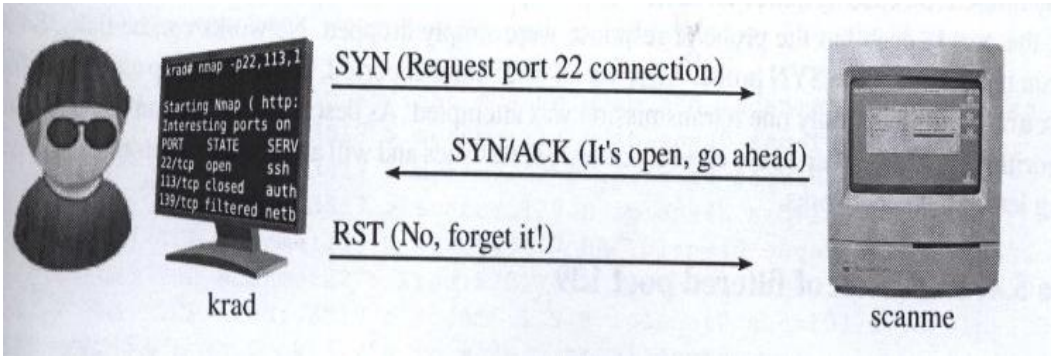
4. Nguyên tắc Scan Port trên một hệ thống

1. TCP Scan

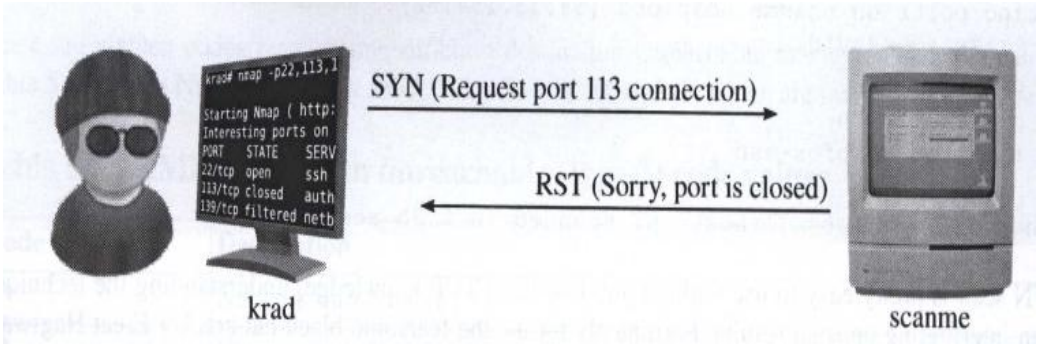
Trên gói TCP/UDP có 16 bít dành cho Port Number điều đó có nghĩa nó có tử 1-65535 port. Không thể một hacker nào lại scan toàn bộ các port trên hệ thống, chúng chỉ scan những port hay sử dụng nhất thường chỉ sử dụng từ port 1 tới port 1024 ma . Dựa vào nguyên tắc truyền thông tin TCP ta có thể biết được trạng thái các port trên hệ thống máy mục tiêu.

- **SYN Scan**: Khi Client gửi gói SYN với một thông số Port nhất định tới Server nếu Server gửi về gói SYN/ACK thì Client biết Port đó trên Server được mở. Nếu Server gửi về cho Client gói RST/SYN thì biết port đó trên Server đóng.

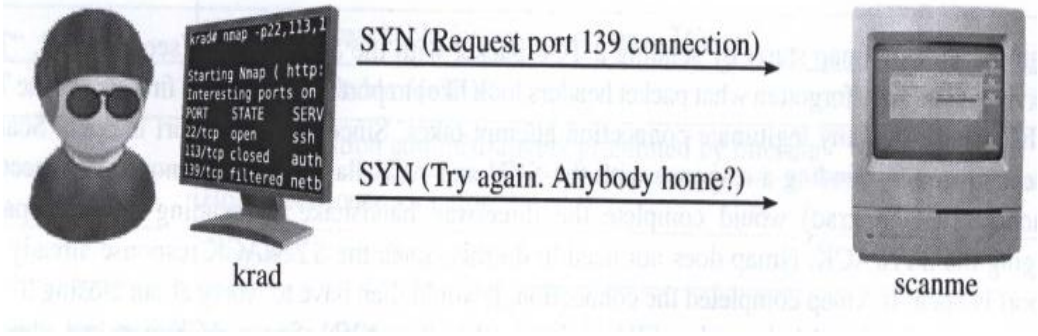
SYN scan với cổng mở port 22



SYN scan với cổng đóng 113



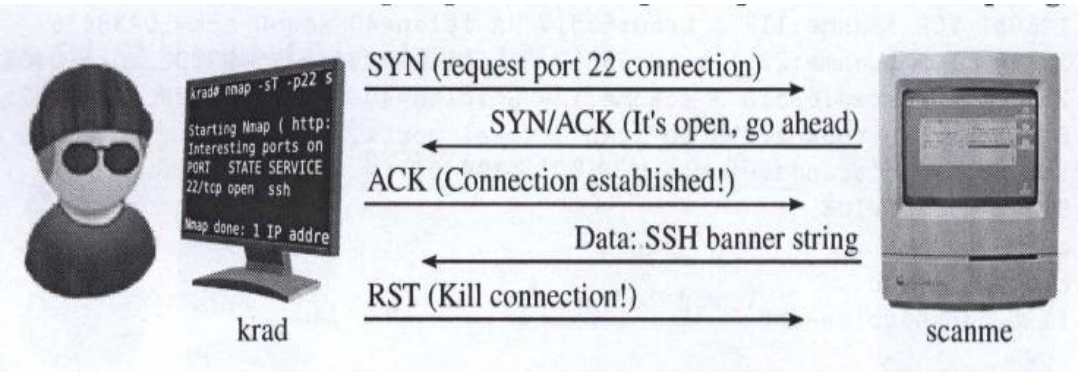
SYN scan với cổng filtered 139



- **FIN Scan**: Khi Client chưa có kết nối tới Server nhưng vẫn tạo ra gói FIN với số port nhất định gửi tới Server cần scan. Nếu Server gửi về gói ACK thì Client biết Server đó mở port. Nếu Server gửi về gói RST thì Client biết Server đó đóng port.
- **NULL Scan**: Client sẽ gửi tới Server những gói TCP với số port nhất định cần scan mà không chứa các thống số Flag như : FIN, URG, PSH, nếu Server gửi lại gói RST thì biết port đó trên Server bị đóng.
- **XMAS Scan**: Client sẽ gửi những gói tin TCP với số port nhất định cần scan chứa nhiều thông số Flag như: FIN, URG, PSH. Nếu Server trả về gói RST tôi biết port đó trên Server bị đóng.
- **TCP Connect**: Phương thức này rất thực tế nó gửi đến Server những gói tin yêu cầu kết nối thực tế tới các port cụ thể trên Server. Nếu Server trả về gói SYN/ACK thì Client biết port đó mở,

nếu Server gửi về gói RST/ACK thì Client biết port đó trên Server bị đóng.

Scan kết nối cổng mở 22



- **ACK Scan**: dạng Scan này nhằm mục đích tìm những Access Controll List trên Server cố gắng kết nối tới Server bằng gói ICMP nếu nhận được gói là Host Unreachable thì Client sẽ biết port đó trên server đã bị lọc.

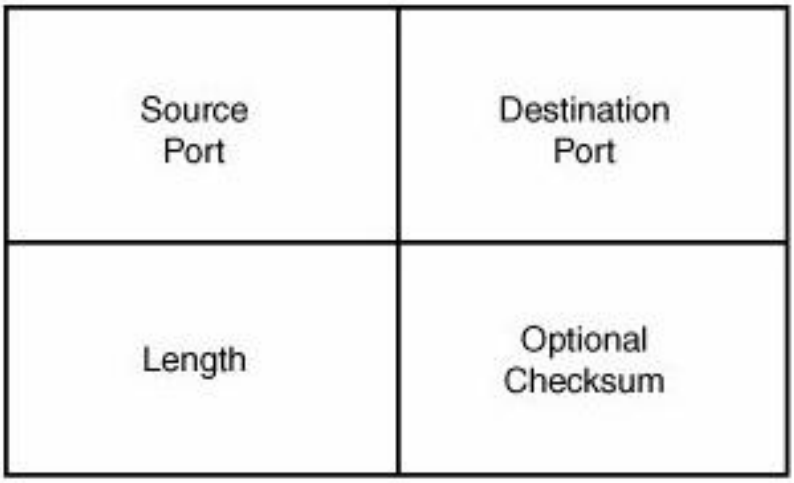
Có vài dạng Scan cho các dịch vụ điển hình dễ bị tấn công như:

- **RPC Scan**: Cố gắng kiểm tra xem hệ thống có mở port cho dịch vụ RPC không.
- **Windows Scan**: Tương tự như ACK Scan, nhưng nó có thể chỉ thực hiện trên một số port nhất định
- **FTP Scan**: Có thể sử dụng để xem dịch vụ FTP có được sử dụng trên Server hay không.

2. UDP Scan

Nếu như gói tin truyền bằng TCP để đảm bảo sự toàn vẹn của gói tin sẽ luôn được truyền tới đích. Gói tin truyền bằng UDP sẽ đáp ứng nhu cầu truyền tải dữ liệu nhanh với các gói tin nhỏ. Với quá trình thực hiện truyền tin bằng TCP kẻ tấn công dễ dàng Scan được hệ thống đang mở những port nào dựa trên các thông số Flag trên gói TCP

Cấu tạo gói UDP



- Như ta thấy gói UDP không chứa các thông số Flag, cho nên không thể sử dụng các phương thức Scan port của TCP sử dụng cho UDP được. Thật không may hầu hết hệ thống đều cho phép gói ICMP.
- Nếu một port bị đóng, khi Server nhận được gói ICMP từ client nó sẽ cố gắng gửi một gói ICMP type 3 code 3 port với nội dung là "unreachable" về Client. Khi thực hiện UDP Scan bạn hãy chuẩn bị tinh thần nhận được các kết quả không có độ tin cây cao

III. Các giai đoạn của Nmap Scan

Target enumeration: Nmap tìm kiếm các máy chủ được cung cấp bởi người sử dụng, có thể là một sự kết hợp của các tên máy chủ DNS, địa chỉ IP, các kí hiệu mạng CIDR và nhiều hơn nữa. Sử dụng (-iR) để yêu cầu nmap chọn máy mục tiêu cho bạn. Nmap sư dụng –sL –n là lựa chọn thực hiện quét một danh sách các địa chỉ IP được lưu trong một file.

Host discovery(ping scanning): Quét mạng bắt đầu bằng việc khai thắc các máy mục tiêu trên mạng có đang hoạt động hay không. Tiến trình này gọi là **host discovery** hoặc **ping scanning.** Nmap cung cấp nhiều kỹ thuật phát hiện máy chủ, có thể sử dụng yêu cầu ARP kết hợp với TCP, ICMP và các kiểu khác.

Reverse-DNS resolution: Nmap xác định host để scan, tìm kiếm reverse-DNS name của toàn bộ host đang online bằng việc ping scan.

Port scanning: Thăm dò là gửi và trả lời (hoặc không trả lời) đối vowid các thăm dò là sử dụng truy nhập cổng tử xa để xác định trạng thái của công hiện thời là open, closed hoặc filtered.

Version detection: Nếu một vài cổng xác định là mở, Nmap có thể xác định phần mền máy chủ đang chạy trên hệ thống từ xa (-sV).

OS detection: Nếu yêu cầu với lựa chọn –O, Nmap sẽ phát hiện hệ điều hành đang sử dụng.

Traceroute: Nmap chứa một thành phần traceroute, --traceroute. Có thể tìm kiếm các route mạng tới nhiều host . Traceroute liên quan tới phân giải tên miền cho việc xác định host.

Script scanning: Nmap Script Engine(NSE) sử dụng các kịch bản để có được nhiều thông tin hơn về hệ thống từ xa. Như việc khai thắc các điểm yếu, backdoor và nhiều malware. Lựa chọn –script hoặc –sC.

Output: Nmap thu thập toàn bộ thông tin và đưa ra một file. Nmap có thể đưa ra một vài định dạng của file.

IV. Các Option trong Nmap

1. Host discovery

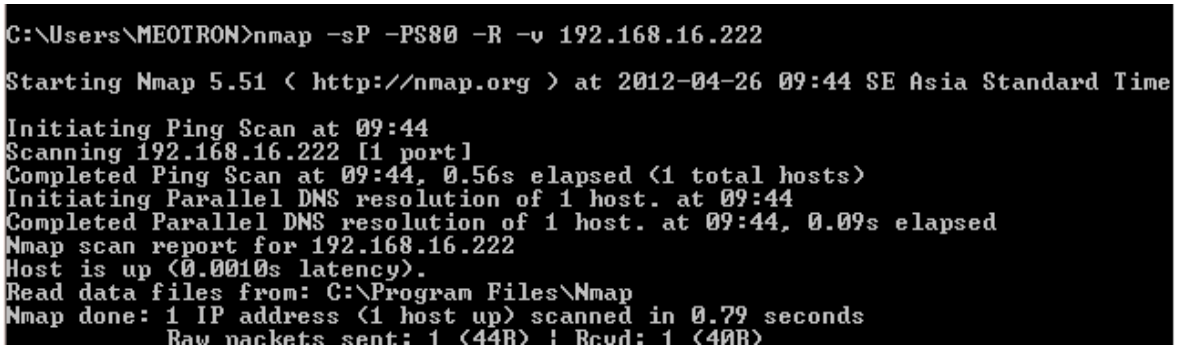
List Scan (-sL): Đưa ra một danh sách các host mục tiêu cần quét.

Ping Scan (-sP): Nmap chỉ thực hiện ping scan, đưa ra trả lời các host. Nó có thể sử dụng để đếm các máy trên mạng hoặc giám sát các máy. Thường gọi là ping sweep, và nhiều tin cậy hơn ping địa chỉ broadcast bởi vì nhiều host không reply tới broadcast. Lựa chọn –sP sẽ gửi ICMP echo và TCP ACK tới cổng mặc định 80. Một gói SYN được gửi và được sử dụng TCP connect system call tới cổng 80 của máy mục tiêu cần quét. Khi một user quét một target trên một mạng local, ARP yêu cầu (-PR) được sử dụng trừ khi lựa chọng – send-ip là đặc biệt.

Disable Ping (-PN): Nmap sử dụng để xác định máy đang được hoạt động phục vụ cho việc quét. Nmap chỉ thực hiện hành vi thăm dò như quét cổng, xác định phiên bản, hệ điều hành đối với các host đang được up. Vô hiệu hóa máy chủ với lựa chọn –PN, nmap thử các chức năng quét các địa chỉ IP mục tiêu yêu cầu đặc biệt.

2. Kỹ thuật phát hiện máy chủ

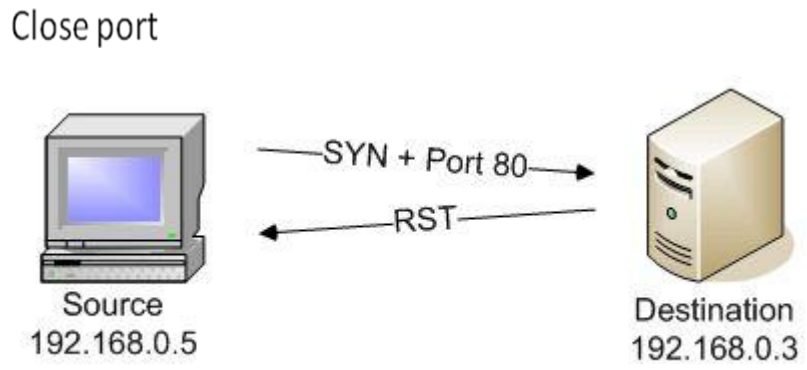
TCP SYN Ping (-PS <port list>) : Tùy chọn –PS sẽ gửi gói TCP trống với thiết lập cờ SYN. Cổng mặc định là 80. Cũng có thể khai báo một danh sách các cổng vi du: -PS22-25,80,113,1050,35000.

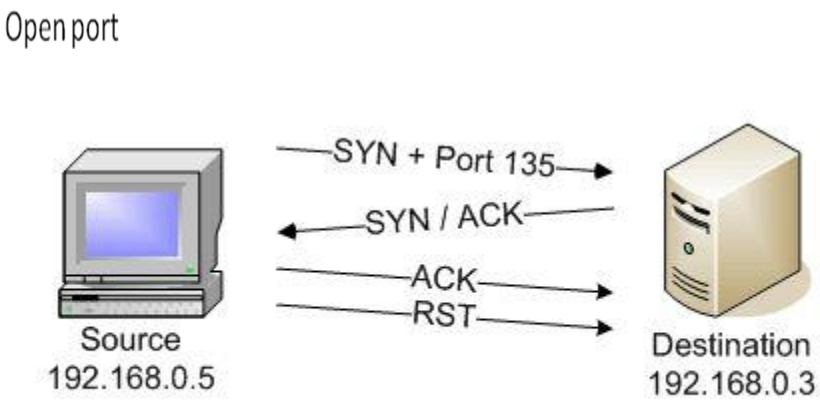


TCP ACK Ping (-PA <port list>): TCP ACK cũng tương tự như SYN Ping. Điểm khác là TCP ACK được thiết lập thay vì SYN flag. Như gói ACK mục đích thừa nhận thiết lập kết nối TCP. Nếu không tồn tại một kết nối như vậy thì máy chủ từ xa sẽ gửi một gói RST.

Lựa chon –PA sử dụng cổng mặc định như thăm dò gói tin SYN 80 và cũng có thể đưa ra một danh sách các cổng trong một định dạng tương tự nhau.

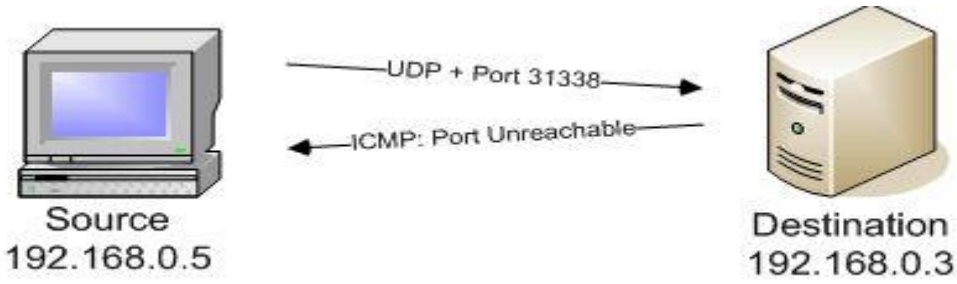
Cố gắng ping chống lại Microsoft. Gói tin bị firewall drops, dẫn đến nmap sai kết luận nên host down.





UDP Ping (-PU <port list>): Một phát hiện khác với lựa chọn UDP ping là sẽ gửi gói trin UDP rỗng tới các port. Một danh sách định dạng tương tự với –PS và –PA. Nếu không có cổng nào đặc biệt thì cổng mặc định là 31 và 338.

Một cổng đóng trên máy mục tiêu, thăm dò kiểu UDP có thể gợi ra một gói tin ICMP port unreachable . Nhiều kiểu khác của ICMP lỗi như host/network unreachable hoặc vượt quá TTL hoặc unreachable host. Nếu một cổng mở là đạt, hầu hết các dịch vụ đơn giản bỏ qua gói tin trống và fail để trả về bất kỳ trả lời nào.



ICMP Ping (-PE, -PP, -PM): Nmap gửi gói tin ICMP (-8 echo request) tới các địa chỉ đích, kiểu mong đợi là 0 (echo reply) trả về giá trị cho host. Nhiều host và firewall bây giờ blog gói tin, không trả lời các yêu cầu khi nmap gửi gói tin tới. ICMP chỉ quét hiếm khi đủ tin cậy để khai thác thông tin trên mạng. Nhưng đối với các quản trị viên hệ thống giám sát một mạng nội bộ, điều này có thể là một cách tiếp cận thực tế và hiệu quả khi sử dụng –PE để cho phép hành vi echo request.

IP Protocol Ping (-PO <protocol list>): Gửi gói tin IP với số giao thức đặc biệt trong IP header. Danh sách giao thức có định dạng tương tự như danh sách cổng bàn luận trước đó TCP và UDP. Nếu không có giao thức đặc biệt, mặc định là gửi nhiều gói tin IP cho ICMP (protocol 1), IGMP (protocol 2) , và IP –in-IP (protocol 4). Giao thức mặc định có thể được cấu hình ở compile-time bằng cách thay đổi DEFAULT_PROTO_PROBE_PORT_SPEC trong nmap.

ARP Scan (-PR): Khi Nmap cố gắng gửi gói tin raw IP như ICMP echo request, hệ điều hành phải xác định địa chỉ phần cứng đích (ARP) tương ứng với địa chỉ IP đích. Yêu cầu này là gửi một loạt các yêu cầu ARP.

V. Các lựa chọn trong Nmap

-v (giống như –verbose**):** Nmap thường chỉ đưa ra hoạt động đáp ứng tới host.

-- source-port <portnum> (giống với -g**):** thiết lập một cổng nguồn không đổi cho việc ping scan (TCP và UDP). Một vài người quản trị đã vô tình mở cổng 53 (DNS) hoặc cổng 20(FTP-DATA) . Tất nhiên mở cổng này đủ để nmap có thể quét thăm dò thông tin của máy đích.

-n, -R: Với –n lựa chọn này sẽ disable phân giải tên miền DNS, trong khi –R thì enable DNS cho toàn bộ host thậm chí cả máy đang down. Hành vi mặc định là để giới hạn tên miền DNS đối với các host đang hoạt động.

--data-length <length>: Lựa chọn này là thêm chiều dài bytes dữ liệu đối với mỗi gói tin, va làm việc với TCP, UDP và cả ICMP ping scan.

--ttl <value>: Thiết lập thời gian sống của gói tin nếu vượt quá sẽ tự động out, là một biện pháp hữu ích phòng ngừa an toàn để đảm bảo một máy quét không truyền vượt ra ngoài mạng nội bộ.

-T (-T3, -T4, -T5, vv..):Tăng tốc độ quét cho Nmap.

-iL <filename>, -ỉR <number>: Người sử dụng thường kết hợp với một danh sách các địa chỉ IP được nhập vào một file với lựa chọn –PN để tránh ping-scanning host đối với các host đang hoạt động. –iR là để chọn host ở chế độ ngẫu nhiên từ một khoảng địa chỉ IP.

Output (-oA, -oN, -oG, -oX,…): Kết quả Nmap sau khi quét sẽ được lưu dưới các dạng như normal, grepable và XML.

--randomize-hosts: xáo trộn thứ tự quét máy chủ lưu trữ với tùy chọn này có thể làm cho quá trình quét ít bị chú ý, mặc dù nó cũng có thể làm cho sản lượng quét một chút khó khăn.

--reason: Nmap bình thường đầu ra cho biết máy chủ đang up hay down nhưng không mô tả các host kiểm tra yêu cầu.

--packet-trace: Khi muốn nhiều thông tin chi tiết hơn –reason. Lựa chọn này sẽ đưa ra thông tin các gói được gửi và nhận bởi Nmap, bao gồm các chi tiết như sequence numbers, giá trị TTL, và TCP flags.

-D <decoy1, decoy2, …>: Decoy được hỗ trợ đầy đủ cho phép đặc quyền quét IPv4, để ngụy trang những kẻ thực sự tấn công.

-6: TCP kết nối dựa trên ping scan –PS hỗ trợ giao thức IPv6, bao gồm chế độ đa cổng như –PS22,80,113.

-S <source IP address>, -e <sending device name> : Như với các chức năng khác của Nmap, địa chỉ nguồn và thiết bị gửi có thể là lựa chọn đặc biệt.

VI. Các trạng thái công nhận với Nmap

Open port: Một ứng dụng tích cực chấp nhân kết nối gói tin TCP hoặc UDP trên cổng này. Các attacker luôn luôn muốn khai thác những cổng đang được mở, trong khi người quản trị lại cố găng muốn đóng hoặc bảo vệ các cổng bằng firewalls. Cổng mở thì không an toàn bởi chúng khi bị quét sẽ thể hiện các dịch vụ đang chạy khi sử dụng trên mạng.

Open closed: Một cổng đóng có thể truy cập (Nó nhận và trả lời các thăm dò của Nmap), nhưng không có ứng dụng nào đang lắng nghe trên nó. Chúng có thể có ích khi thể hiện các host đang online hay đang sử dụng một địa chỉ IP (host discovery, hoặc ping scanning), và một phần phat hiện về hệ điều hành. Bởi cổng đóng là có thể truy cập. Người quản trị có thể muốn khóa các cổng với ứng dụng firewall xuất hiện trong trạng thái lọc.

Port filtered: Nmap không thể xác định bất kỳ cổng nào đang mở bởi vì gói tin được lọc và ngăn cản trước khi tiến tới cổng. Filtering có thể từ một thiết bị tường lửa, router rules, hoặc phần mền host-base firewall. Các cổng này làm thất bại các kể tấn công bởi vì chúng cung cấp quá ít thông tin. Đôi khi chúng trả lời với gói tin ICMP lỗi như kiểu 3 mã 13 (destination unreachable: communication administratively prohibited), nhưng filter đơn giản là drop các thăm dò không trả lời phổ biến.

Port Unfiltered: Trạng thái unfiltered nghĩa là cổng có thể truy nhập, nhưng Nmap không thể xác định nó là cổng mở hay cổng đóng. Chỉ là ACK scan, được sử dụng để map firewall rulesets, classifies port vào trong trạng thái này. Quét cổng unfiltered vowid các kiểu scan khác như

Window scan, SYN scan, hoặc FIN scan có thể giải quyết các cổng đang mở.

Port open|filtered: Nmap đặt các cổng vào trong trạng thái khi nó không thể xác định các cổng là đang mở hay đang filtered. Điều này xảy ra cho các kiểu quét trong khi cổng mở không đưa ra trả lời. Thiếu phản ứng cũng có thể có nghĩa là một bộ lọc gói tin giảm thăm dò hoặc đáp ứng bất kỳ gợi ra. Nmap không biết chắc rằng cổng đang mở hay đang được filtered. Scan kiểu UDP, IP protocol, FIN, NULL và Xmas là một trong những cách để quét cổng dạng này.

Port closed|filtered: Trạng thái này được sử dụng khi Nmap không thể xác đinh các cổng được closed hay filtered. Nó chỉ sử dụng cho việc xác định IPID Idle.

VII. Kỹ thuật lựa chọn quét cổng

1. Phương thức quét cổng hỗ trợ bởi Nmap

TCP SYN Stealth (-sS): Đây là cách quét cổng phổ biển bởi vì nó là cách nhanh nhất để quét các cổng trên hầu hết các giao thức phổ biển (TCP). Nó quét âm thầm hơn là kiểu quét kết nối, và nó làm việc dựa trên toàn bộ chức năng stack TCP. (Không giống với các kiểu quét với mục đích đặc biệt như FIN scan).

TCP Connect (-sT): Scan connect sử dụng hệ thống gọi giống với tên để quét các máy, mặt khác sẽ trả lời bằng gói tin raw như hầu hết các phương thức khác. Thường sử dụng đối với người sử dụng Unix không có đặc quyền và dựa trên mục tiêu IPv6 bởi vì SYN scan không làm việc trong trường hợp này.

UDP (-sU): Đừng quên cổng UDP, chúng thường quá nhiều lỗ hổng bảo mật

TCP FIN, Xmas, và NULL (-sF, -sX. -sN): Kiểu quét mục đích đặc biệt vượt qua firewall để khai thác hệ thống đằng sau chúng. Tiếc là

chúng trả lời trên các hành vi mục tiêu đối với một vài hệ thống (đặc biệt đối với các biến thể window) không thể hiện.

TCP ACK (-sA): ACK scan được sử dụng phổ biến đối với các luật của firewall. Nó giúp hiểu luật firewall là statefull hoặc không. Nhược điểm là nó không phân biệt được là các cổng mở hay đóng.

TCP Window (-sW): Window scan thì giống với ACK scan, Nó có thể xác định cổng so với các cổng đóng dựa trên các máy trung tâm.

TCP Maimon (-sM): Ít người biêt firewall-evading scan thì tương tự đối với kiểu quét FIN, nhưng bao gồm ACK flag. Điều này cho phép nhiều gói tin firewall lọc, với nhược điểm là nó làm việc dựa trên một số ít hệ thông hơn là FIN scan.

TCP Idle (-sI <zombie host>): Idle scan là kiểu quét toàn bộ, và đôi khi nó khai thac các địa chỉ có mối quan hệ đáng tin cậy. Nhược điểm là nó chậm và phức tạp

IP protocol (-sO): Protocol scan xác đinh các giao thức IP như (TCP, ICMP, IGMP, vv..) được hỗ trợ bởi các máy mục tiêu. Đây không phải là một kỹ thuật quét cổng, thông qua các giao thức chứ không phải là số cổng TCP hay UDP. Vẫn sử dụng lựa chọn –p để chọn số giao thức được quet, báo cáo kết quả với một bảng định dạng các cổng, và sử dụng quét cơ bản như phương thức quét cổng thực.

TCP FTP bounce (-b <FTP bounce proxy>): Phản đối quét thủ thuật máy chủ FTP vào thực hiện quét cổng bằng proxy. Hầu hết FTP server bây giờ được vá để ngăn cản điều này. Nhưng nó là cách tốt để vượt qua tường lửa khi nó làm việc.

2. **Lựa chọn cổng để quet**

-p 22: Quét cổng đơn trong trường hợp này là cổng 22

-p ssh: Tên cổng có thể tốt hơn là dạng số. Lưu ý một tên có thể thay thể cho nhiều cổng.

-p 22, 25, 80: Nhiều cổng được cách nhau bởi dấu phẩy. Nếu một TCP scan như SYN scan (-sS) là đặc biệt, cổng TCP 22, 25, và 80 được quét. Tương ứng với các dịch vụ SSH, SMTP, và HTTP. Nếu là UDP scan thì được chọn (-sU).

-p 80-85, 443, 8000-8005, 8080-8085: Đó là một giải cổng được quét được cách nhau bởi dấu phẩy.

-p-100, 6000-: Bạn có thể bỏ qua cổng bắt đầu của một giải cổng, được bắt đầu từ cổng 1, hoặc cổng cuối là 65535 đối với TCP và UDP, 225 đối với giao thức quét.

-p-: Bỏ qua cổng bắt đầu và kết thúc để quét toàn bộ giải cổng (không bao gồm cổng Zero) .

-pT: 21, 23, 110, U: 53, 111, 137, 161: Một danh sách cổng TCP và UDP có thể được đưa ra với T: (cho TCP) hoặc U. Đây giống quét 3 cổng TCP (FTP, Telnet, và POP3), và 4 dịch vụ UDP (DNS, rpcbind, NetBIOS, và SNMP).

-p http* : Có thể sử dụng để phù hợp với tên tương tự. vi dụ như http (80), http-mgmt(280), https (443), và http-proxy (8080).

VIII. Demo quet cổng khi bị firewall chặn

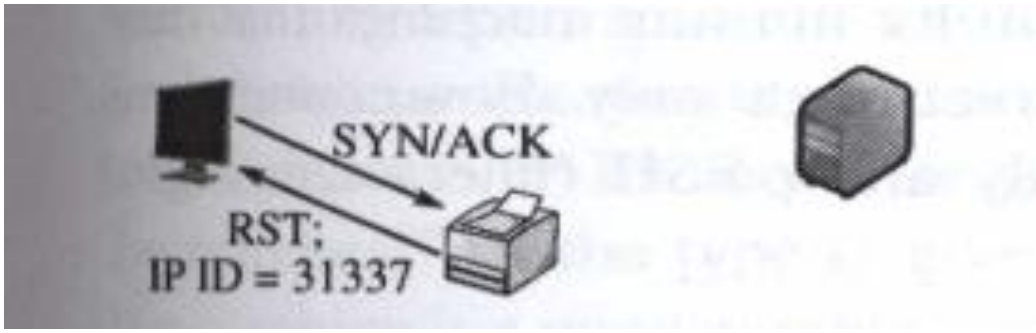
Sử dụng phương thức quét Idle. Với kiểu quét này ta sử dụng một zombie để làm trung gian thực hiện quét qua tường lửa. Zombie là một máy tính có thể thực hiện quét máy mục tiêu mà không bị tường lửa ngăn cấm. Lợi dụng điểm yếu này mà ta sử dụng phương pháp Idle scan. Dựa vào sự biến đổi tăng của IPID để kết luận cổng đang được đóng hay đang mở.

Mẫu lệnh thực hiện:

```
Attacker# nmap -sI Zombie -PN -p20-25,110 -r --packet-trace -v Target
Starting Nmap ( http://nmap.org )
```

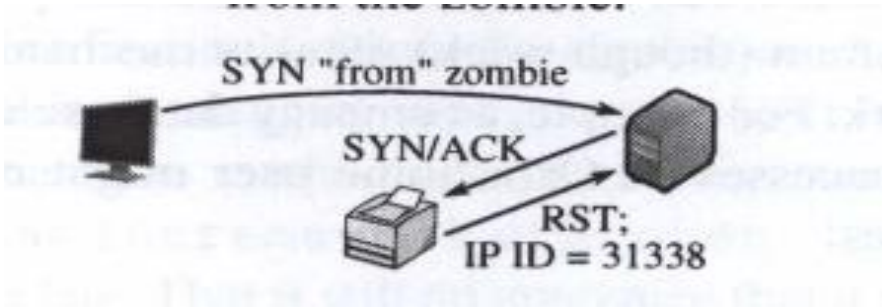
1. Idle scan với trường hợp cổng mở được mô tả như sau:

Bước 1: Thăm dò Zombie



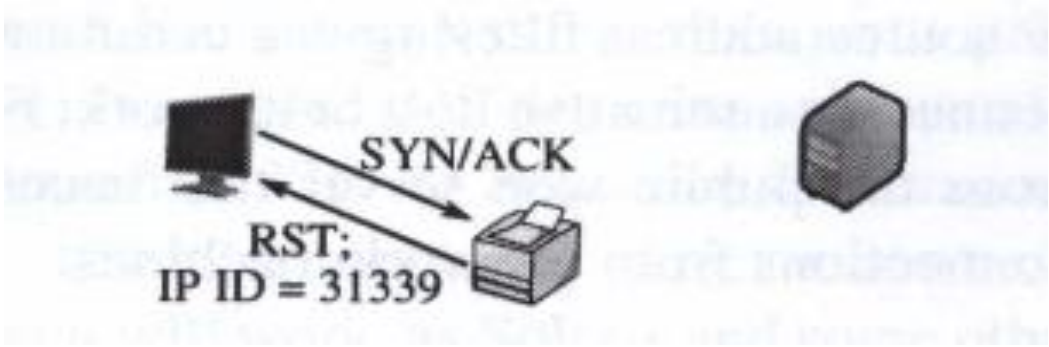
Attacker gửi một gói tin SYN/ACK tới Zombie. Zombie, không mong đợi SYN/ACK, gửi quay lại một gói RST và tiết lộ thông tin IP ID.

Bước 2: Giả mạo một gói tin từ Zombie



Máy mục tiêu(target) gửi một gói SYN/ACK đến từ zombie. Zombie không hy vọng nó, nó sẽ gửi lại một RST, và tăng IP ID lên trong quá quá trình đó.

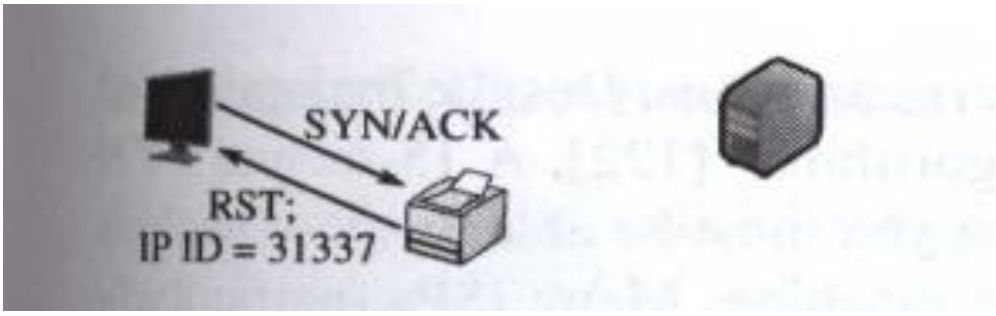
Bước 3: Thăm dò IP ID của Zombie một lần nữa



IP ID của Zombie đã được tăng lên 2 từ bước 1, do vậy cổng là mở.

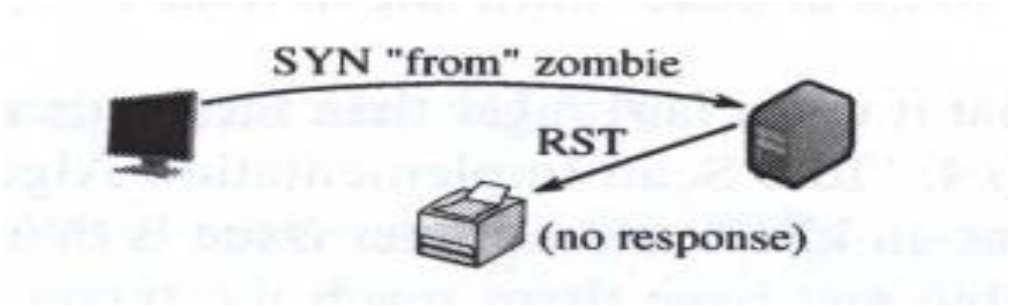
2. **Idle scan với trường hợp cổng đóng**

Bước 1: Thăm dò IP ID của Zombie



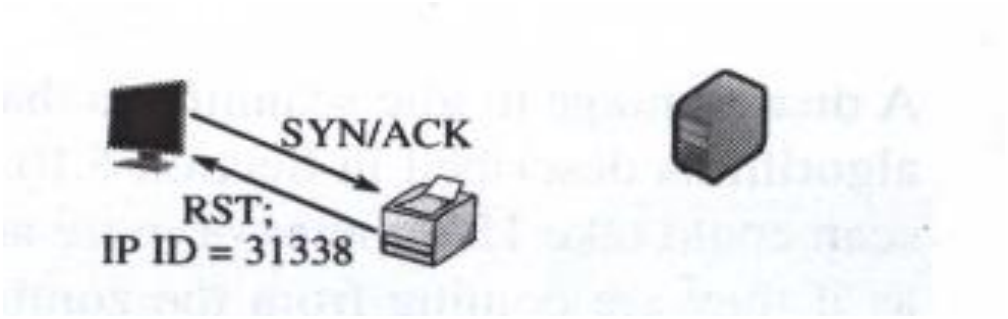
Attacker gửi gói tin SYN/ACK tới zombie. Zombie, không mong đợi gói SYN/ACK, Nó sẽ gửi lại một gói tin RST, tiết lộ thông tin IP ID. Bước này luôn luôn giống nhau.

Bước 2: Giả mạo một gói SYN từ Zombie



Máy mục tiêu (Target) gửi một gói RST (cổng được đóng) bằng trả lời gói SYN xuất hiện đến từ zombie. Zombie bỏ qua gói RST không được yêu cầu, để lại IP ID của nó không thay đổi.

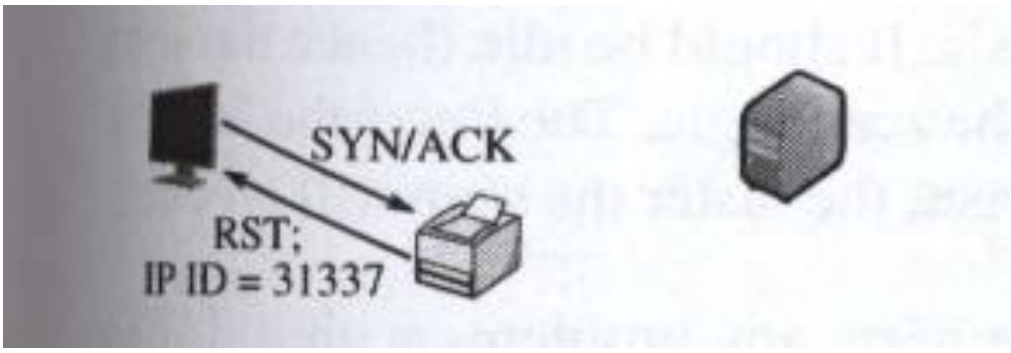
Bước 3: Thăm dò IP ID của Zombie một lần nữa



IP ID của Zombie đã được tăng chỉ 1 từ bước 1, do vậy cổng là không được mở.

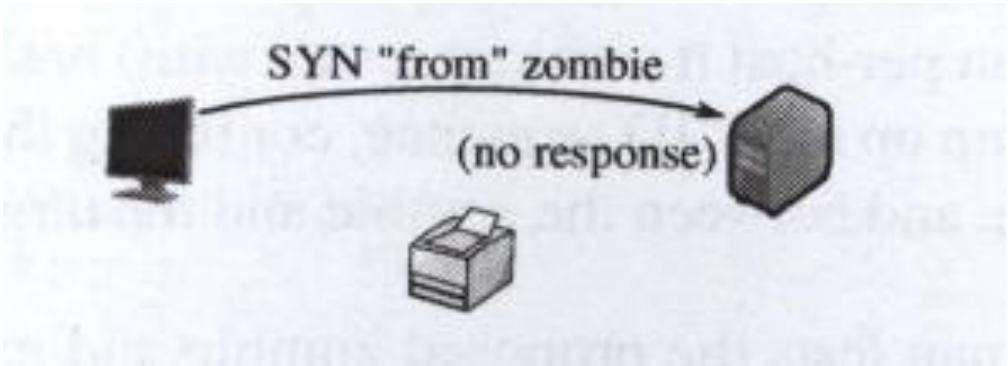
3. **Idle scan với trường hợp cổng được lọc filtered**

Bước 1: Thăm dò IP ID của Zombie



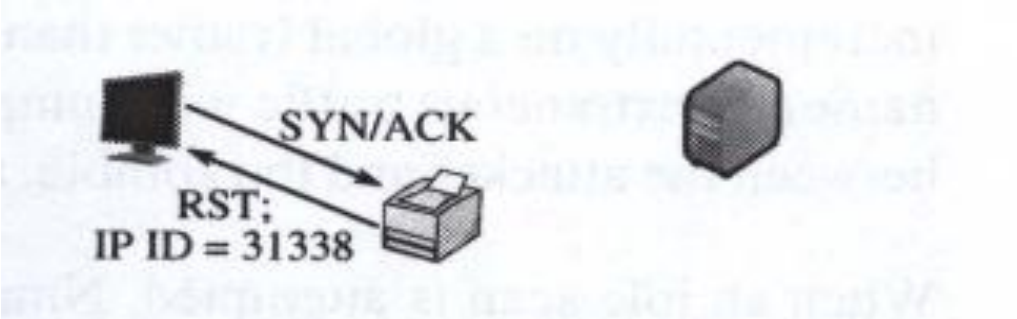
Giống 2 trường hợp trên. Attacker gửi gói SYN/ACK tới Zombie. Zombie tiết lộ thông tin về IP ID.

Bước 2: Giả mạo gói SYN từ Zombie



Máy mục tiêu (Target), ngoan cố lọc cổng của nó, bỏ qua gói SYN xuất hiện đến từ zombie. Zombie, không biết rằng bất kỳ điều gì đã xảy ra, không tăng IP ID của nó.

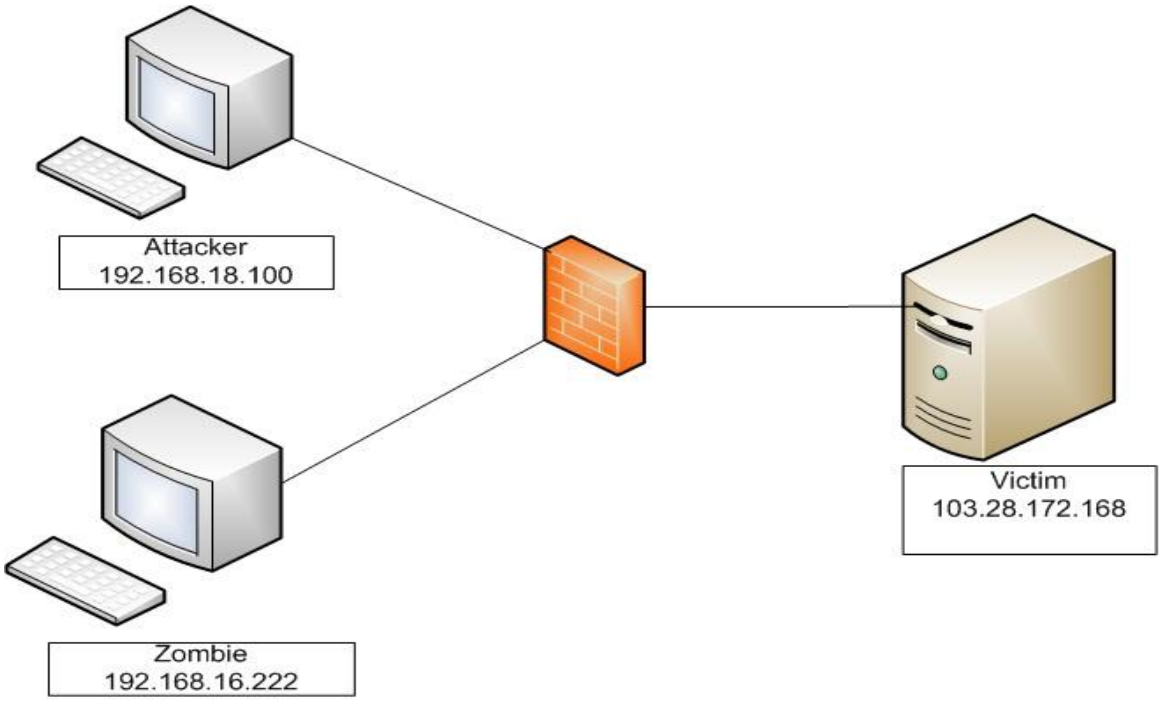
Bước 3: Thăm dò IP ID của Zombie lại một lần nữa.



IP ID của Zombie đã được tăng chỉ 1 từ bước 1, do vậy cổng không mở. Từ quan sát của attacker cổng này lọc là không thể phân biệt được cổng là đóng.

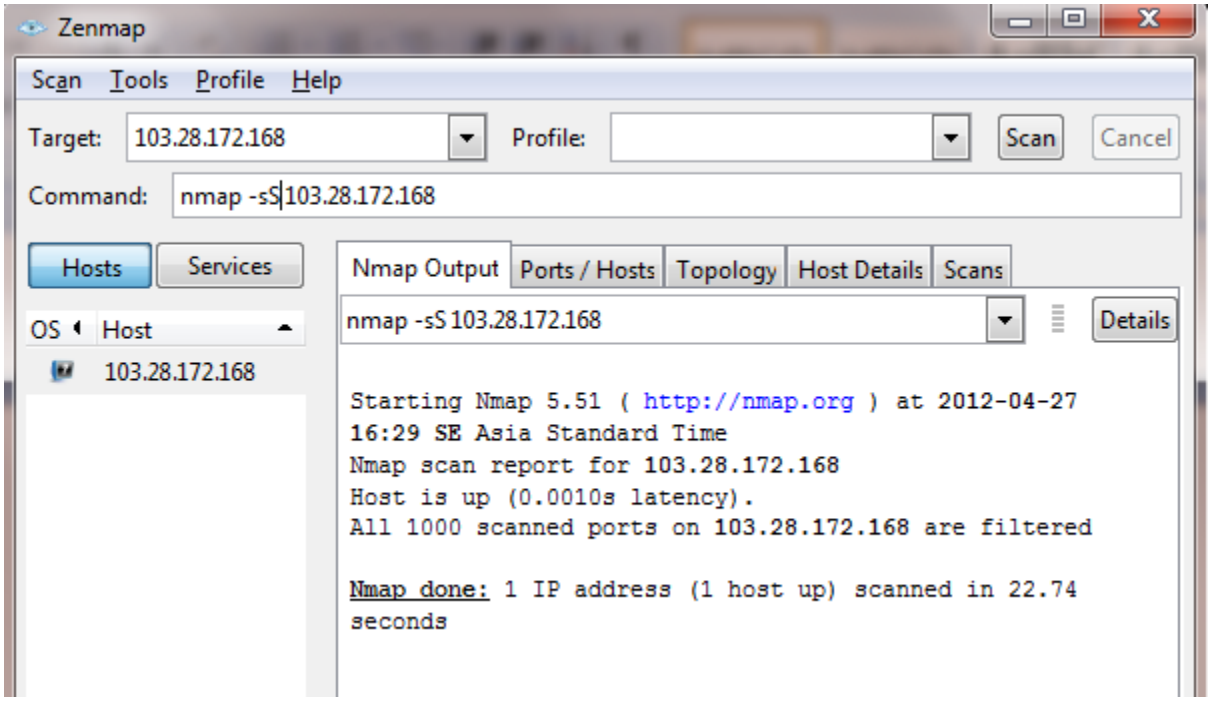
4. Demo tấn công.

Mô hình như sau:

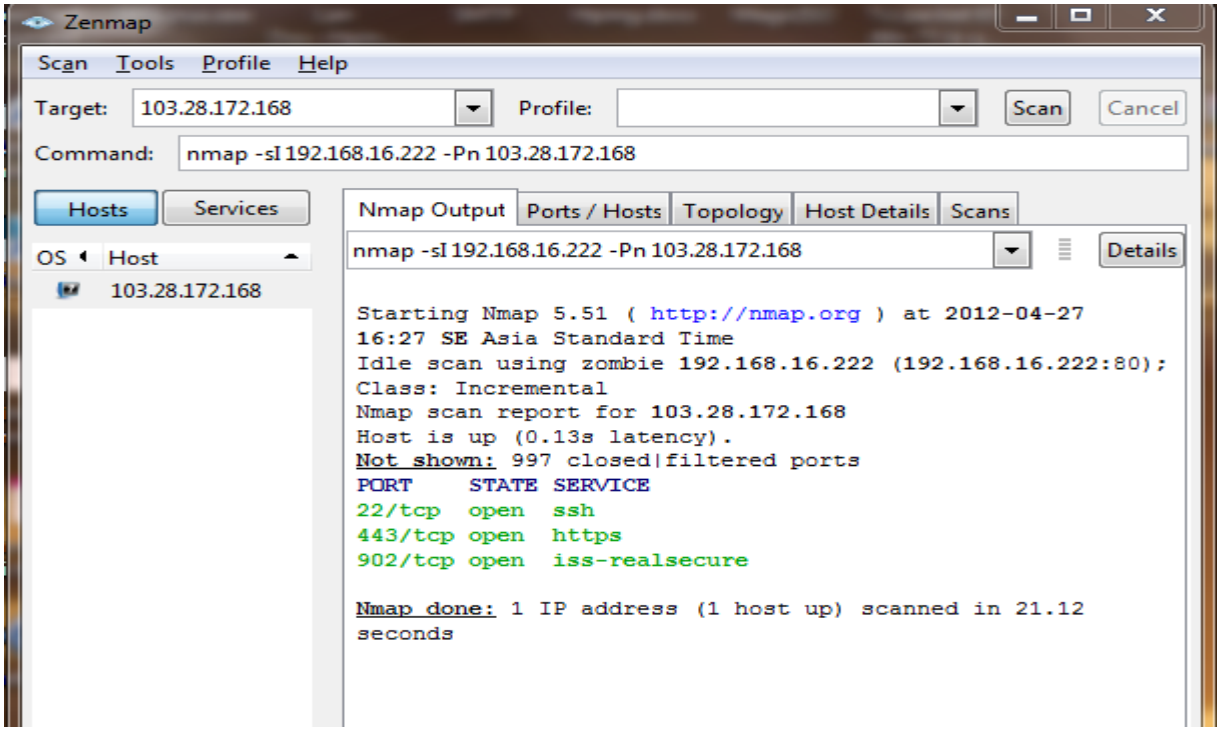


Kết quả:

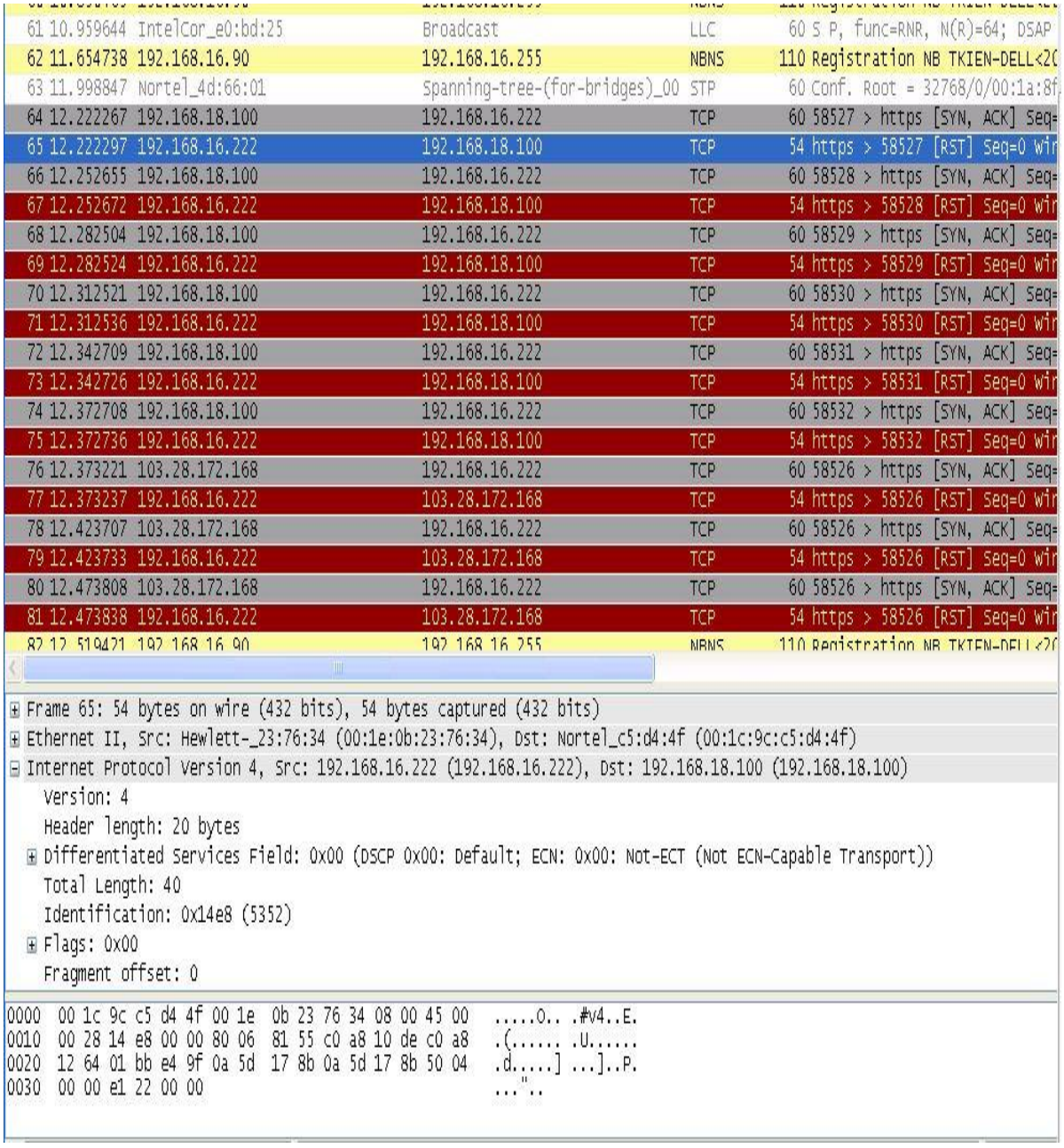
Khi thực hiện quét trực tiếp từ máy Attacker.



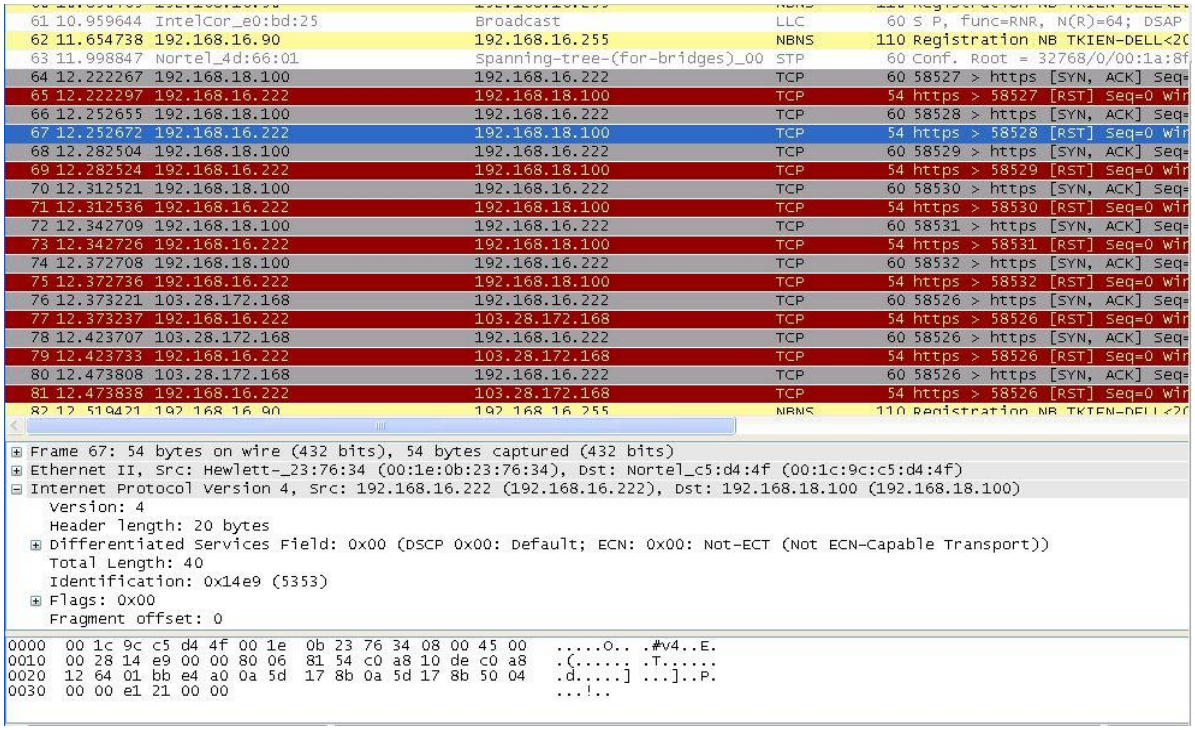
Khi thực hiện quét với Idle:



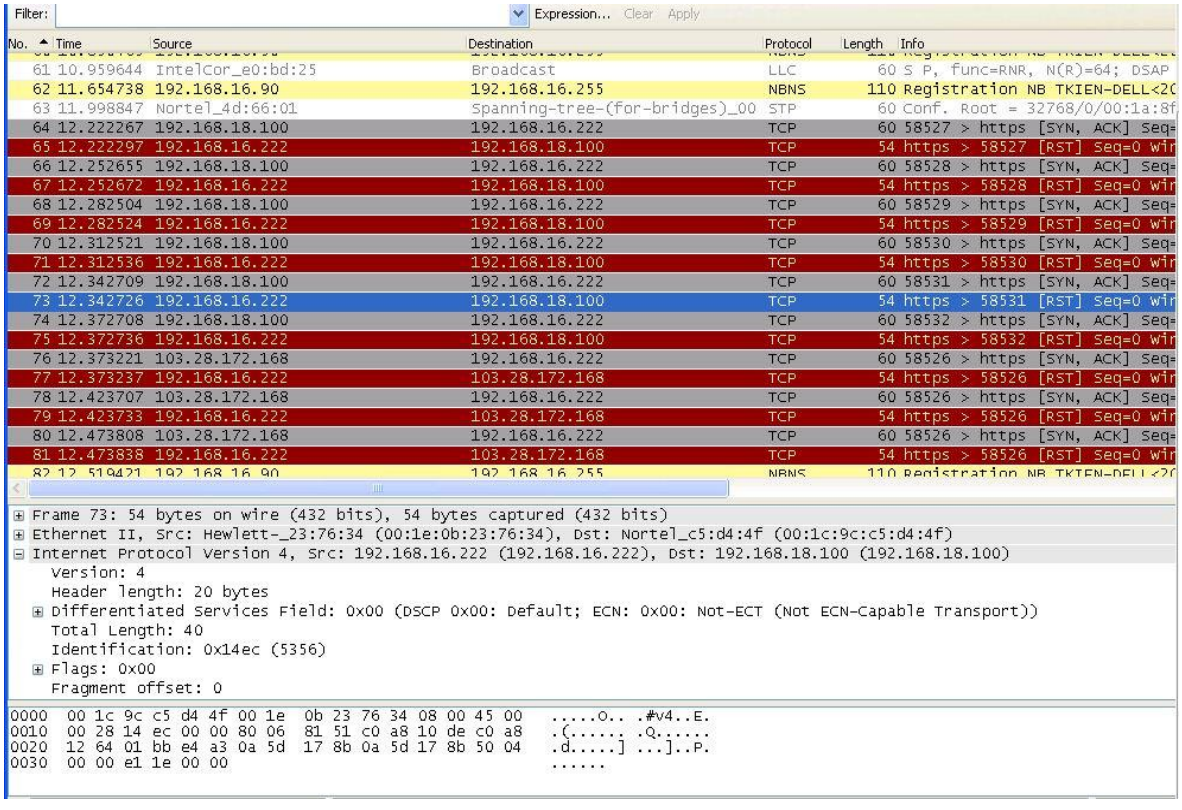
a.Từ packet 65 đến 73 là quá trình nmap khảo sát trước IPID của zombie. Nmaps gửi liên tục SYN/ACK và chờ RST để ghi lại IPID. IPID tăng.



Hinh 2

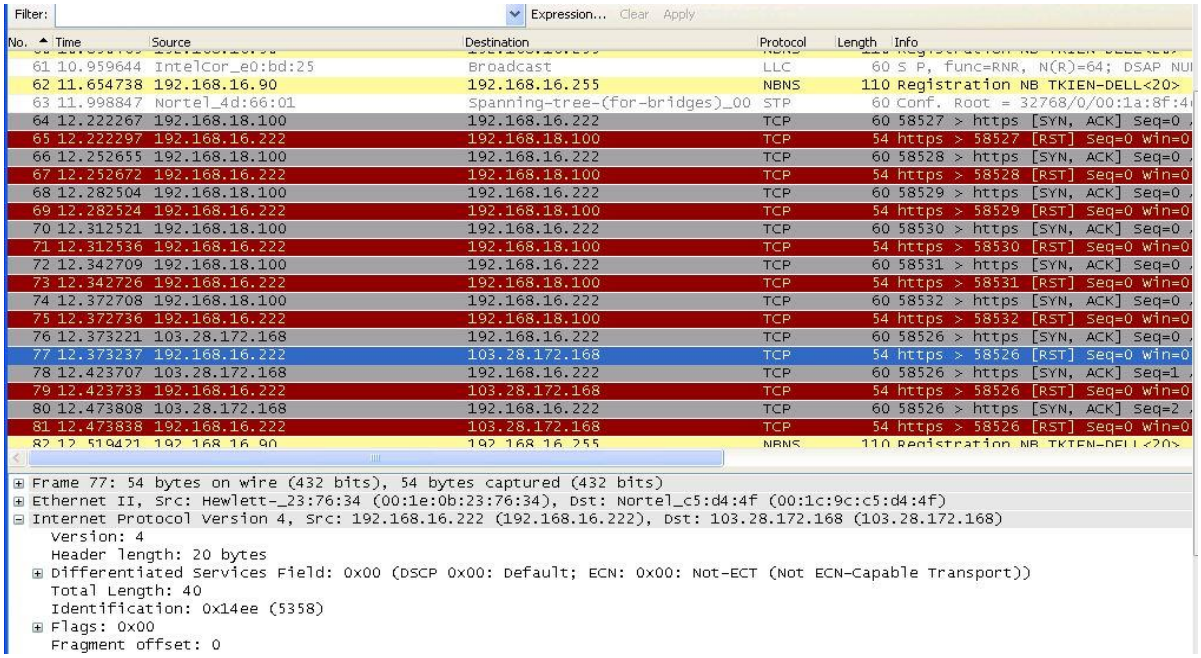

61 10.959644 IntelCor_e0:bd:25 Broadcast LLC 60 S P, func=RNR, N(R)=64; DSAP
62 11.654738 192.168.16.90 192.168.16.255 NBNS 110 Registration NB TKIEN-DELL<2
63 11.998847 Nortel_4d:66:01 Spanning-tree-(for-bridges)_00 STP 60 Conf. Root = 32768/0/00:1a:8f
64 12.222267 192.168.18.100 192.168.16.222 TCP 60 58527 > https [SYN, ACK] Seq=
65 12.222297 192.168.16.222 192.168.18.100 TCP 54 https > 58527 [RST] Seq=0 Wir
66 12.252655 192.168.18.100 192.168.16.222 TCP 60 58528 > https [SYN, ACK] Seq=
67 12.252672 192.168.16.222 192.168.18.100 TCP 54 https > 58528 [RST] Seq=0 Wir
68 12.282504 192.168.18.100 192.168.16.222 TCP 60 58529 > https [SYN, ACK] Seq=
69 12.282524 192.168.16.222 192.168.18.100 TCP 54 https > 58529 [RST] Seq=0 Wir
70 12.312521 192.168.18.100 192.168.16.222 TCP 60 58530 > https [SYN, ACK] Seq=
71 12.312536 192.168.16.222 192.168.18.100 TCP 54 https > 58530 [RST] Seq=0 Wir
72 12.342709 192.168.18.100 192.168.16.222 TCP 60 58531 > https [SYN, ACK] Seq=
73 12.342726 192.168.16.222 192.168.18.100 TCP 54 https > 58531 [RST] Seq=0 Wir
74 12.372708 192.168.18.100 192.168.16.222 TCP 60 58532 > https [SYN, ACK] Seq=
75 12.372736 192.168.16.222 192.168.18.100 TCP 54 https > 58532 [RST] Seq=0 Wir
76 12.373221 103.28.172.168 192.168.16.222 TCP 60 58526 > https [SYN, ACK] Seq=
77 12.373237 192.168.16.222 103.28.172.168 TCP 54 https > 58526 [RST] Seq=0 Wir
78 12.423707 103.28.172.168 192.168.16.222 TCP 60 58526 > https [SYN, ACK] Seq=
79 12.423733 192.168.16.222 103.28.172.168 TCP 54 https > 58526 [RST] Seq=0 Wir
80 12.473808 103.28.172.168 192.168.16.222 TCP 60 58526 > https [SYN, ACK] Seq=
81 12.473838 192.168.16.222 103.28.172.168 TCP 54 https > 58526 [RST] Seq=0 Wir
82 12.519421 192.168.16.90 192.168.16.255 NBNS 110 Registration NB TKIEN-DELL<2
Frame 67: 54 bytes on wire (432 bits), 54 bytes captured (432 bits)
Ethernet II, Src: Hewlett-_23:76:34 (00:1e:0b:23:76:34), Dst: Nortel_c5:d4:4f (00:1c:9c:c5:d4:4f)
Internet Protocol Version 4, Src: 192.168.16.222 (192.168.16.222), Dst: 192.168.18.100 (192.168.18.100)
Version: 4
Header length: 20 bytes
Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
Total Length: 40
Identification: 0x14e9 (5353)
Flags: 0x00
Fragment offset: 0
0000 00 1c 9c c5 d4 4f 00 1e 0b 23 76 34 08 00 45 00 .....O.. .#v4..E.
0010 00 28 14 e9 00 00 80 06 81 54 c0 a8 10 de c0 a8 .(...... .T......
0020 12 64 01 bb e4 a0 0a 5d 17 8b 0a 5d 17 8b 50 04 .d.....] ...]..P.
0030 00 00 e1 21 00 00 ...!..

Hinh 3:


Filter: Expression... Clear Apply
No. Time Source Destination Protocol Length Info
61 10.959644 IntelCor_e0:bd:25 Broadcast LLC 60 S P, func=RNR, N(R)=64; DSAP
62 11.654738 192.168.16.90 192.168.16.255 NBNS 110 Registration NB TKIEN-DELL<2
63 11.998847 Nortel_4d:66:01 Spanning-tree-(for-bridges)_00 STP 60 Conf. Root = 32768/0/00:1a:8f
64 12.222267 192.168.18.100 192.168.16.222 TCP 60 58527 > https [SYN, ACK] Seq=
65 12.222297 192.168.16.222 192.168.18.100 TCP 54 https > 58527 [RST] Seq=0 Wir
66 12.252655 192.168.18.100 192.168.16.222 TCP 60 58528 > https [SYN, ACK] Seq=
67 12.252672 192.168.16.222 192.168.18.100 TCP 54 https > 58528 [RST] Seq=0 Wir
68 12.282504 192.168.18.100 192.168.16.222 TCP 60 58529 > https [SYN, ACK] Seq=
69 12.282524 192.168.16.222 192.168.18.100 TCP 54 https > 58529 [RST] Seq=0 Wir
70 12.312521 192.168.18.100 192.168.16.222 TCP 60 58530 > https [SYN, ACK] Seq=
71 12.312536 192.168.16.222 192.168.18.100 TCP 54 https > 58530 [RST] Seq=0 Wir
72 12.342709 192.168.18.100 192.168.16.222 TCP 60 58531 > https [SYN, ACK] Seq=
73 12.342726 192.168.16.222 192.168.18.100 TCP 54 https > 58531 [RST] Seq=0 Wir
74 12.372708 192.168.18.100 192.168.16.222 TCP 60 58532 > https [SYN, ACK] Seq=
75 12.372736 192.168.16.222 192.168.18.100 TCP 54 https > 58532 [RST] Seq=0 Wir
76 12.373221 103.28.172.168 192.168.16.222 TCP 60 58526 > https [SYN, ACK] Seq=
77 12.373237 192.168.16.222 103.28.172.168 TCP 54 https > 58526 [RST] Seq=0 Wir
78 12.423707 103.28.172.168 192.168.16.222 TCP 60 58526 > https [SYN, ACK] Seq=
79 12.423733 192.168.16.222 103.28.172.168 TCP 54 https > 58526 [RST] Seq=0 Wir
80 12.473808 103.28.172.168 192.168.16.222 TCP 60 58526 > https [SYN, ACK] Seq=
81 12.473838 192.168.16.222 103.28.172.168 TCP 54 https > 58526 [RST] Seq=0 Wir
82 12.519421 192.168.16.90 192.168.16.255 NBNS 110 Registration NB TKIEN-DELL<2
Frame 73: 54 bytes on wire (432 bits), 54 bytes captured (432 bits)
Ethernet II, Src: Hewlett-_23:76:34 (00:1e:0b:23:76:34), Dst: Nortel_c5:d4:4f (00:1c:9c:c5:d4:4f)
Internet Protocol Version 4, Src: 192.168.16.222 (192.168.16.222), Dst: 192.168.18.100 (192.168.18.100)
Version: 4
Header length: 20 bytes
Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
Total Length: 40
Identification: 0x14ec (5356)
Flags: 0x00
Fragment offset: 0
0000 00 1c 9c c5 d4 4f 00 1e 0b 23 76 34 08 00 45 00 .....O.. .#v4..E.
0010 00 28 14 ec 00 00 80 06 81 51 c0 a8 10 de c0 a8 .(...... .Q......
0020 12 64 01 bb e4 a3 0a 5d 17 8b 0a 5d 17 8b 50 04 .d.....] ...]..P.
0030 00 00 e1 1e 00 00 ......

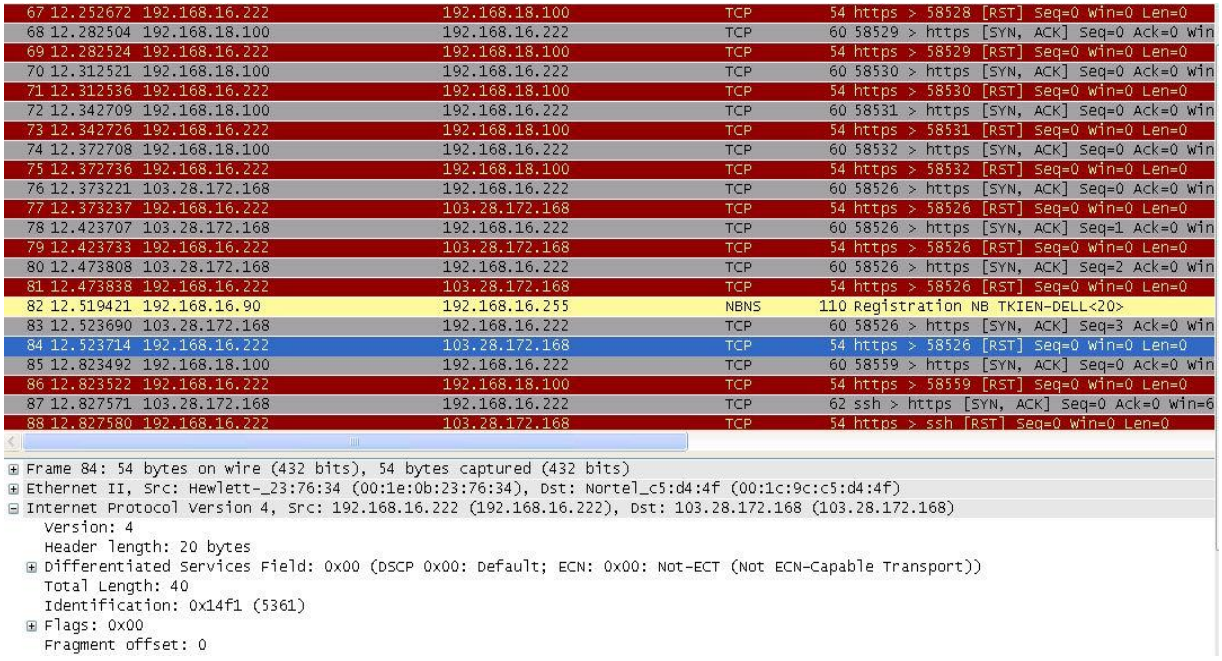
b. Từ packet 77 đến 84 Nmap giả lập Targer mở port, nó dùng IP Targer thử xem IPID có vẫn tiếp tục tăng nếu trả Target gửi SYN/ACK.

Hình 4:



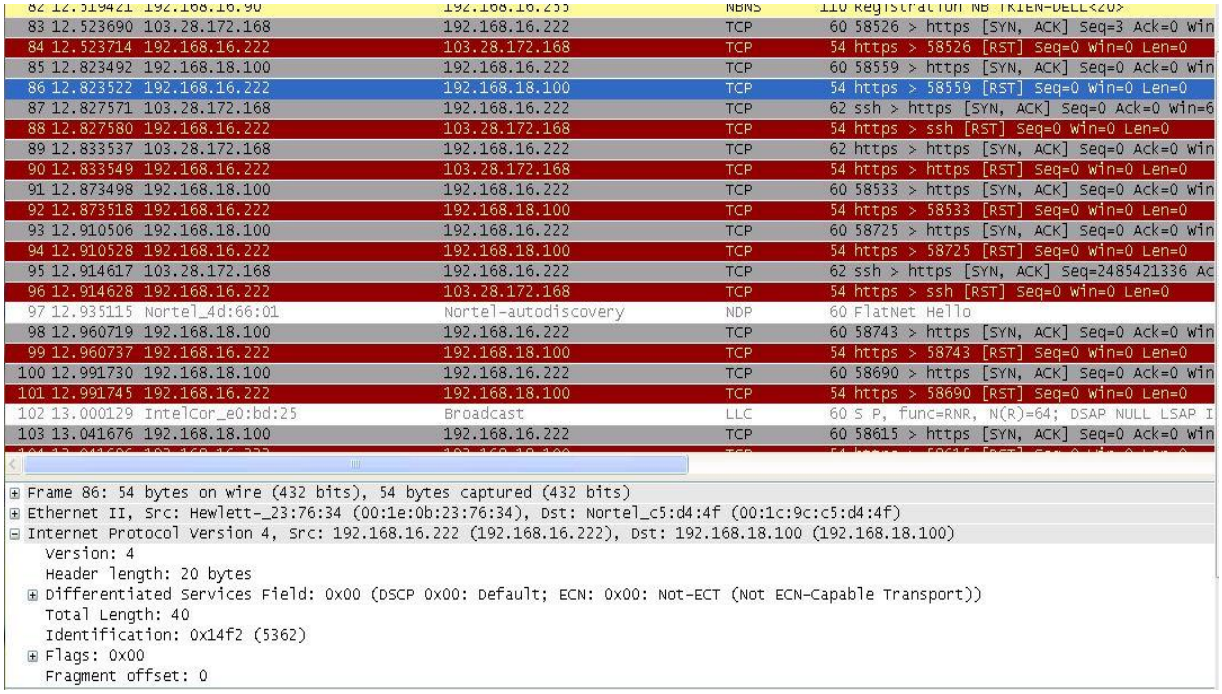
Hình 5:

Hinh 6:



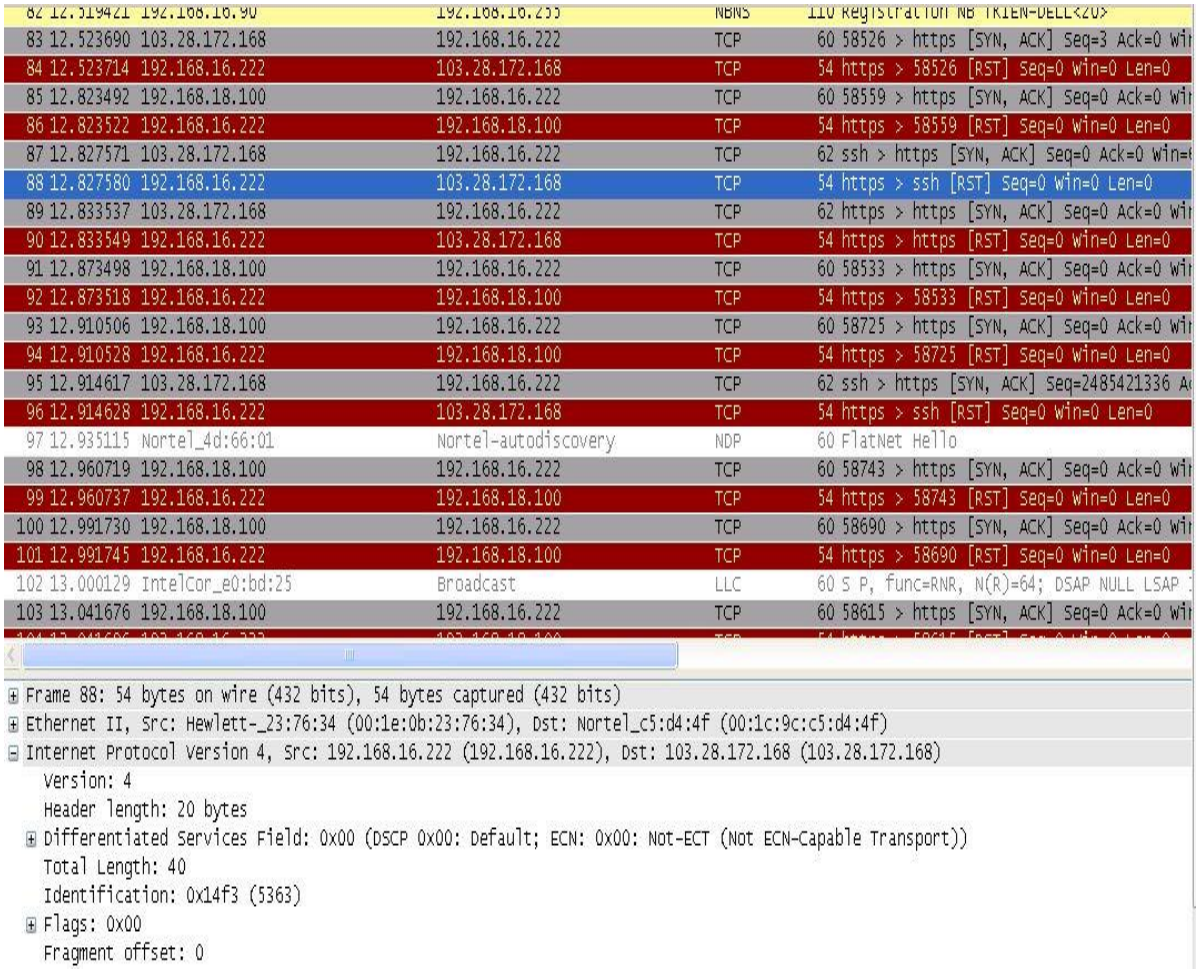
c.Packet 85,86 là để kiểm tra lại sau khi giả lập Targer mở port. Nmap gửi SYN/ACK. IPID lúc này là 5362

Hinh 7:

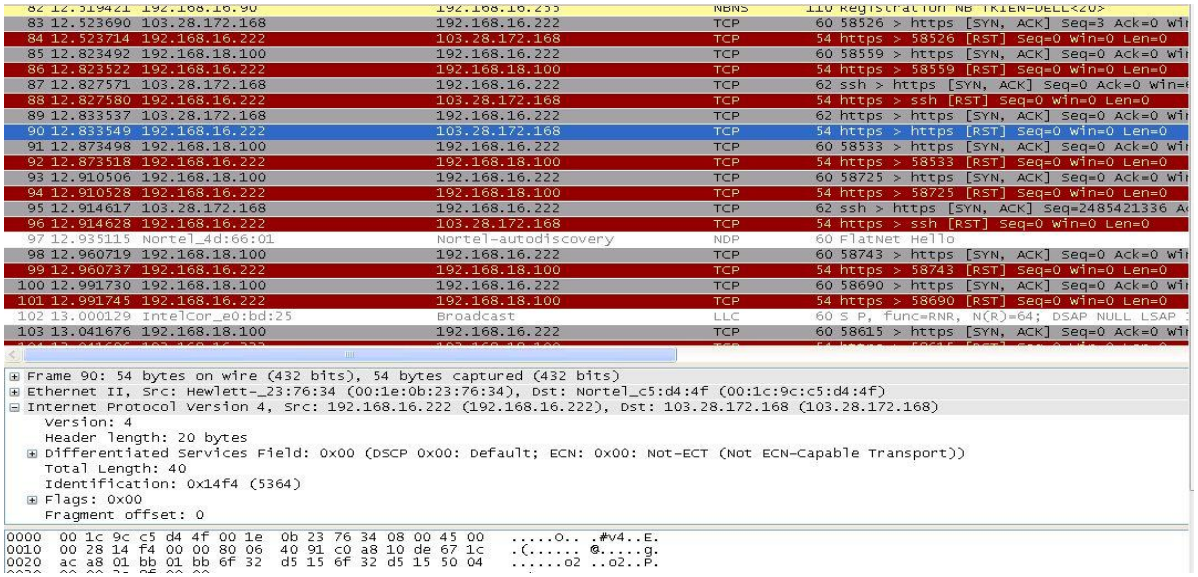


d.Sau khi kiểm tra mọi thứ hoạt động tốt, nmap bắt đầu SYN Targer với IP của zombie: packet 88 đến 90. Do mở port nên Target trả lời SYN/ACK cho zombie (packet 89), zombie trả lời RST và IPID tăng 5369 (packet 90).

Hình 8:

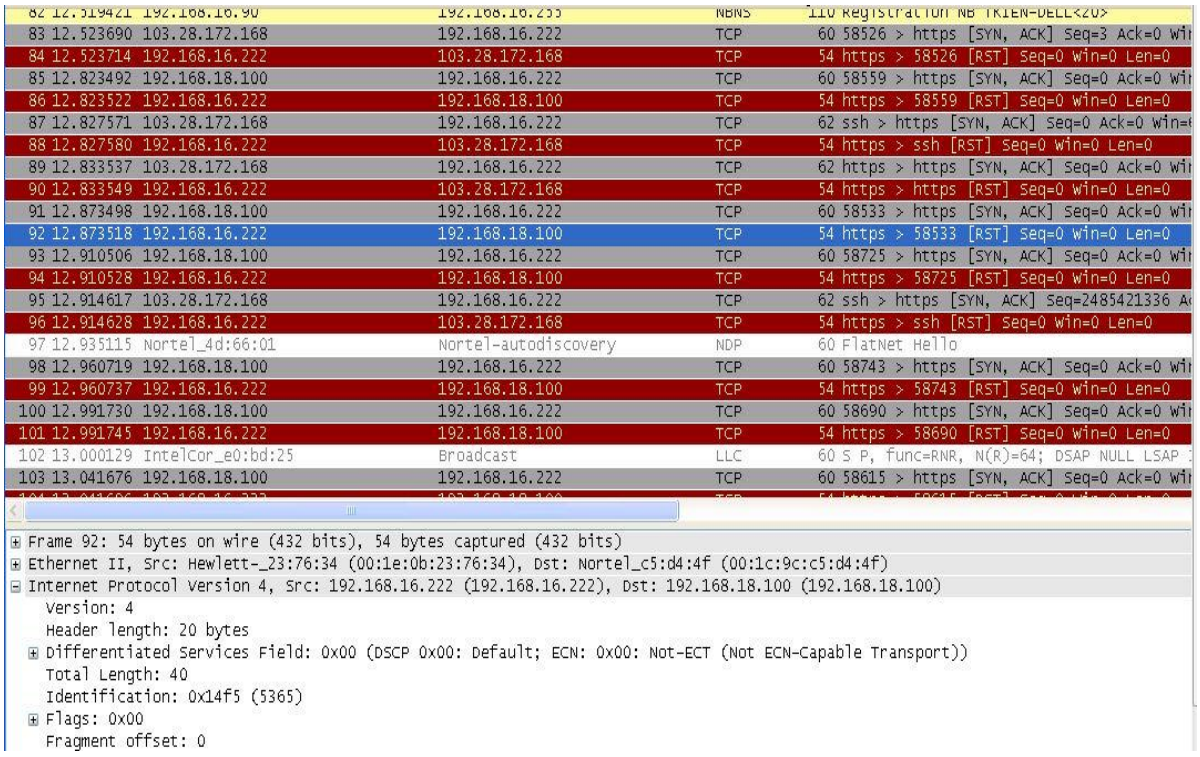


Hinh 9:



e. Packet 91, 92 nmap kiểm tra lại IPID: 5365 =➔ kết luận port mở.

hinh 10:



IX. Nmap script engine

Nmap script engine là một trong những chức năng mạnh và linh hoạt của nmap. Nó cho phép người sử dụng viết những script đơn giản để tự động mở rộng nhiệm vụ của mạng. Script sau đó được thực hiện song song với tốc độ và hiệu quả mà bạn mong đợi từ nmap.

Script NSE linh hoạt với những nhiệm vụ sau:

Network discovery: Tìm kiếm toàn bộ dữ liệu dựa trên domain của máy mục tiêu, truy vấn ARIN, RIPE, hoặc APNIC đối với IP máy mục tiêu để xác định quyền sở hữu, thực hiện các tra cứu các cổng mở, các truy vấn SNMP và NFS/SMB/RPC đối với các dịch vụ có sẵn.

Phát hiện nhiều phiên bản phức tạp hơn: Hệ thống phát hiện phiên bản của Nmap có thể công nhận hàng trăm dịch vụ khác nhau khi thăm dò và thường xuyên phát hiện chữ ký dựa trên hệ thống, Nhưng nó không nhận ra mọi thứ.

Phát hiện điểm yếu(Vulnerability detection): Khi một điểm yếu mới được phát hiện, bạn thường muốn quét mạng của bạn nhanh để xác định điểm yếu của hệ thống trước khi nguy cơ xấu có thể xảy ra. Trong khi Nmap không phải là công cụ quét toàn diện, NSE là đủ mạnh để xử lý kiểm tra những điểm yếu đang đe dọa. Nhiều script phát hiện điểm yếu thực sự có sẵn.

Phát hiện Backdoor: Nhiều attacker và một vài worms tự động chuyển backdoor tới và một vài trong số chúng có thể phát hiện bằng Nmap dựa trên nhiều phiên bản phát hiện.

Khai thác điểm yếu(Vulnerability exploitation): NSE có thể được sử dụng để khai thác điểm yếu hơn là để tìm chúng. Khả năng thêm kịch bản khai thác điểm yếu có thể có giá trị với một số người.

Chúng ta có thể sử dụng --script hoặc -sC

Câu lệnh: *Nmap –sc -p <port list> target*

NSE Scripts:

(server message block) protocol.

Smb-security-mode:

Smb-enum-shares: được sử dụng để kiểm tra thông tin được chia sẻ trên máy target.

Smb-enum-users: liệt thông tin về các user khi đăng nhập tài khoản may target

Smb-enum-sessions: kiểm tra thông tin người dùng

Smb-enum-processes: liệt kê các tiến trình đang chạy trên một máy từ xa, đọc file HKEY_PERFORMAINCE_DATA registry ẩn, và phân tích các dữ liệu tìm thấy trong đó. Chỉ dùng cho win2000.

Smb-system-info: thông tin chi tiêt về hệ điều hành chỉ dùng trên win2000.

Smb-check-vulns: giúp người quản trị tìm các lỗi hệ điều hành MS08-067. Là lỗ hổng nghiêm trọng trong windows vào tháng 10 năm 2008. Phát hiện conficker worm khi khai thác điểm yếu.

Smb-brute: nỗ lực đăng nhập vào một tài khoản SMB bằng việc đoán username và password. Mã và thuật toán được thiết kế để tận dụng lợi thế của giao thức SMB trong nhiều cách khác nhau để khám phá mà người dùng tồn tại và có hay không thể xác định mật khẩu . mục đích của smb-brute là để thực hiện kiểm tra nhanh đối với các password phổ biến, không khởi động một brutefore đầy đủ. Phần lớn sức mạnh của nó đến từ một sự hiểu biết sâu sắc của giao thức SMB.

Smb-pwdump: đưa ra thông tin danh sách hashes từ hệ thống từ xa của người dùng.

Smb-os-discovery: đưa ra thông tin về hệ điều hành, tên máy tính, domain, workgroup, thời gian hiện tại trên giao thức SMB.

Asn-query: đưa ra thông tin số hiệu mạng và quốc gia. Sử dụng script này khi quét sẽ được lưu lại trên máy chủ mục tiêu bao gồm địa chỉ ip máy và số hiệu mạng của một máy chủ DNS (máy chủ DNS mặc định của bạn).

nmap --script asn-query [--script-args dns=<DNS server>] <target>

Auth-owners: Cố gắng tìm chủ sở hữu của một cổng TCP.

nmap -sV -sC <target>

auth-spoof: Kiểm tra đáp ứng máy chủ bằng việc trả lời trước khi gửi truy vấn.

nmap -sV --script=auth-spoof <target>

daytime: lấy ngày và thời gian

nmap -sV --script=daytime <target>

dns-random-srcport: kiểm tra máy chủ dns. Giúp phat hiển lỗ hổng cổng nguồn máy chủ DNS để tấn công đầu độc bộ nhớ cache (CVE-2008-1447). Kịch bản này có khả năng ghi lại bởi một hoặc nhiều máy chủ dns. Ngoài ra địa chỉ ip của bạn sẽ được gửi đi cùng với các truy vấn porttest đến máy chủ DNS đang chạy trên mục tiêu.

nmap -sV --script=dns-random-srcport <target>

dns-recursion: kiểm tra nếu máy chủ DNS cho phép truy vấn cho tên của bên thứ ba.

nmap -sV -sC <target>

dns-zone-transfer: yêu cầu một zone transfer(AXFR) từ máy chủ DNS. Script sẽ gửi một truy vấn AXFR tới một máy chủ DNS.

```
nmap --script dns-zone-transfer.nse \
  --script-args dns-zone-transfer.domain=<domain>
```

Finger: lấy một danh sách tên người dùng sử dụng dịch vụ finger.

```
nmap -sV -sC –script=finger  <target>
```

ftp-bounce:kiểm tra xem một máy chủ FTP cho phép quét cổng bằng cách sử dụng phương thức FTP bounce.

Html-title: hiện title của một trang mặc định của một web server

http-auth: lấy xác thực và các lĩnh vực của một dịch vụ web

http-open-proxy: kiểm tra HTTP proxy mở. script cố gắng kết nối với google.com thông qua proxy và kiểm tra mã phản hồi HTTP hợp lệ. Mã số đáp ứng HTTP hợp lệ là 200, 301 và 302.

http-passwd: kiểm tra nếu một web server bị lỗ hổng đối với một vài thư mục /etc/passwd hoặc \boot.ini.

http-trace: Gửi yêu cầu HTTP TRACE và hiện các trường header được chỉnh sửa .

iax2-version: Xác định dịch vụ UDP IAX2. Script sẽ gửi một Inter-Asterisk eXchange yêu cầu và kiểm tra cho một đáp ứng thích hợp. Giao thức này sử dụng để cho phép các kết nối VoIP giữa các máy chủ cũng như giao tiếp client-server.

Irc-info: Thông tin từ một IRC server.

Ms-sql-info: Xác định chính xác thông tin từ Microsoft SQL

Mysql-info: Kết nối máy chủ MySQL và in các thông tin như về giao thức và số version, thread ID, status, capabilities và password salt.

Nbstat: Xác định tên NetBIOS và địa chỉ MAC. Script hiển thị tên máy tính và loged-in user.

sudo *nmap -sU --script nbstat.nse -p137 <host>*

pop3-brute: Thử đăng nhập vào tài khoản POP3 bằng username guesing và password

sql-injection: Tìm kiếm các URL có chứa các lỗ hổng do một cuộc tấn công SQL injection. Tìm kiếm một máy chủ HTTP cho các URL có chứa các truy vấn, sau đó nó tiến hành kết hợp với các lệnh SQL crafted với URL nhạy cảm để có được các lỗi. Các lỗi được phân tích để xem nếu URL là dễ bị tấn công. Điều này sử dụng hình thức cơ bản nhất của SQL injection.

nmap -sV --script=sql-injection <target>

whois: Truy vấn dịch vụ WHOIS của Regional Internet Registries (RIR) và xác định các thông tin về địa chỉ IP.

Nmap –script=whois target

Xampp-default-auth: Kiểm tra nếu XAMP hoặc XAMPP FTP server sử dụng tên và password mặc định

++

TOOL HPING

I. Giới thiệu

Hping là chương trình Ping sử dụng ICMP echo requests và chờ cho thông tin phản hồi echo reply để kiểm tra kết nối mạng.Hping là command-line hướng kết nối TCP/IP. Một chương trình được gọi là Hping cho phép bạn thực hiện với nhiều loại thử nghiệm kiểm tra sử dụng gói IP, bao gồm ICMP, UDP, TCP.

Hping có thể download từ http://www.hping.org/ và có sẵn nguồn mở.Được chạy trên hệ điều hành Unix.

II. Giới thiệu Hping tool

Hping cũng có các kiểu quét giống với nmap như:

TCP SYN Scan, TCP ACK Scan, FIN Scan, Xmas Scan, …..

ICMP ping (-1): bao gồm hai loại

+ ICMP loại 13 (timestamp): Yêu cầu thời gian trên hệ thống. Xem múi giờ thời gian tại vị trí của hệ thống (-C 13).

+ ICMP loại 17 (address mask request) netmask của thẻ mạng

Có thể xác đin rõ tất cả các mạng cấp dưới dạng sử dụng –C 17.

TCP Ping (-S)

UDP Ping (-2): Một phát hiện khác với lựa chọn UDP ping là sẽ gửi gói trin UDP rỗng tới các port.

III. Các option trong Hping

-h : là lựa chọn gọi sự trợ giúp có sẵn trong hping.

-v : Hiện thông tin về phiên bản và API sử dụng để truy cập tầng liên kết dữ liệu, gói linux sock hoặ libpcap.

-c : Dừng sau khi gửi (và nhận) đếm gói tin trả lời. Sau gói tin cuối cùng được gửi hping chờ COUNTREACHED_TIMEOUT hai giây trả lời host.

-I –interval: Chờ số giây giữa mỗi lần gửi mỗi gói tin. – interval X thiết lập X giây, --interval uX sẽ chờ X micro giây.

--fast: Alias –I u10000. Hping sẽ gửi 10 gói tin cho một giây

-n –numeric: Chỉ số output, tra cứu tên biểu tượng cho địa chỉ máy chủ.

-q –quiet: Đưa ra output lặng lẽ, không tạo ra tra cưu địa chỉ tên máy chủ.

-I –interface interface name: Mặc định trên linux và hệ thống BSD hping sử dụng mặc định định tuyến interface. Trong hệ thống khác hoặc khi không có định tuyến mặc định hping sử dụng firt-non-loopback interface. Tuy nhiên bạn có thể giả mạo hping để sử dụng interface ban cần sử dụng trong lựa chọn này.

-V –verbose: Cho phép verbose output. TCP reply sẽ hiện ra như sau:

en=46 ip=192.168.1.1 flags=RA DF seq=0 ttl=255 id=0 win=0 rtt=0.4 ms tos=0 iplen=40 seq=0 ack=1380893504 sum=2010 urp=0

-D –debug: Cho phép chế độ debug, nó rất hữu ích khi thực hiện với hping. Khi ở chế độ debug bạn có thể có nhiều thông tin về interface detection, data link layer access, interface settings, options parsing, fragmentation, HCMP protocol và nhiều công cụ khác

-z –bind: CTRL + Z đối với TTL bạn có thể tăng, giảm ttl của gói tin.

-Z –unbind: lựa chọn dừng hping.

IV. Lựa chọn giao thức

Mặc định giao thức TCP, hping sẽ gửi tcp header tới host mục tiêu cổng 0 với một winsize 64 mà không có bất kỳ cờ dấu nào được bật. Có ích khi được ẩn sau firewall cái mà drop ICMP.

-0 –rawip: Chế độ RAW IP, trong chế độ này hping sẽ gửi IP header với dữ liệu nối với –signature và /hoặc –file.

-1 –icmp: Chế độ ICMP, hping sẽ gửi ICMP echo-request, bạn có thể thiết lập ICMP sử dụng –icmptype , --icmpcode.

-2 –udp: Chế độ UDP, hping sẽ gửi gói udp tới máy mục tiêu port bằng 0. UDP header cho phép: --baseport, --destport, --keep.

-9 –listion signature: Hping ở chế độ lắng nghe, sử dụng lựa chọn này hping chờ cho gói tin chứa signature và dump .

V. Tùy chọn liên quan

1. Tùy chọn IP liên quan

-a –spoof hostname: Sử dụng lựa chọn này để thiết lập giả địa chỉ IP nguồn, lựa chọn này đảm bảo mục tiêu sẽ không dựa vào địa chỉ thực. Tuy nhiên trả lời sẽ gửi địa chỉ spoof, bạn sẽ không thể nhìn thấy địa chỉ nguồn.

-t –ttl time to live: Sử dụng lựa chọn này có thể thiết lập TTL của gói tin, nó giống với việc sử dụng –traceroute hoặc –bind.

-N –id: thiết lập ip -> trường ID. Mặc định id là ngẫu nhiên nhưng nó phân mảnh được bật và id sẽ là getpid() & 0xFF.

-H –ipproto: Thiết lập giao thức IP trong RAW IP.

-W –winid: Windows *id có byte sắp xếp khác nhau, nếu lựa chọn này cho phép hping sẽ hiển thị trả lời.

-r –rel: Hiển thị id tăng.

-f –frag: Chia nhỏ gói tin, điều này có ích để kiểm tra hiệu suất ngăn xếp phân mảnh IP và kiểm tra nếu một số bộ lọc gói tin là quá yếu có thể được thông qua bằng cách sử dugj các mảnh nhỏ. Mặc định MTU là 16 byte.

-x –morefrag: Thiết lập gói tin trong phân mảnh cờ IP, sử dụng lựa chọn này nếu bạn muốn máy mục tiêu gửi ICMP time-exceeded during reassembly.

-y –dontfrag: không thiết lập phân mảnh cờ IP, đây có thể sử dụng để thực hiện MTU path discovery.

-g –fragoff fragment offset value: Thiết lập offset fragment.

2. Tùy chọn TCP/UDP liên quan

-s –baseport source port: Hping sử dụng cổng nguồn để đoán số chuỗi trả lời. Bắt đầu với cổng nguồn, và tăng lên cho mỗi gói tin được gửi. Khi gói tin được nhận có thể được tính như replies.dest.port –base.source.port. Mặc định dựa trên cổng nguồn là ngẫu nhiên, sử dụng lựa chọn này bạn có thể thiết lập số khác.

--keep: bỏ qua cổng nguồn

-w –win: Thiết lập TCP window size. Mặc định là 64.

-O –tcpoff: Thiết lập TCP data offset.

-M –tcpseq: Thiết lập TCP sequence number.

-L –tcpack: Thiết lập TCP ack

VI. Định dạng TCP đầu ra

len=46 ip=192.168.1.1 flags=RA DF seq=0 ttl=255 id=0 win=0 rtt=0.4 ms

- **Len:**Kích cỡ, bằng bytes của data có được từ data link layer
- **Ip:**Địa chỉ ip nguồn

- **Flags:**The TCP flags: R for RESET, S for SYN, A for ACK, F for FIN, P for PUSH, U for URGENT
- **DF:**Nếu sự trả lời bao hàm cả DF, IP Header có "don't fragment" được đặt
- **seq :**Số lượng gói thu được sử dụng nguồn chuyển cho TCP/UDP packets hoặc sequence field cho ICMP packet.
- **Id:**lĩnh vực IP ID
- **Win:**kích thước cửa sổ TCP
- **Rtt:**Thời gian khứ hồi tính bằng mili-giây
  Nếu bạn chạy hping sử dụng "-V" +"dòng lệnh", nó sẽ trình bày về gói bổ xung. Chẳng hạn:

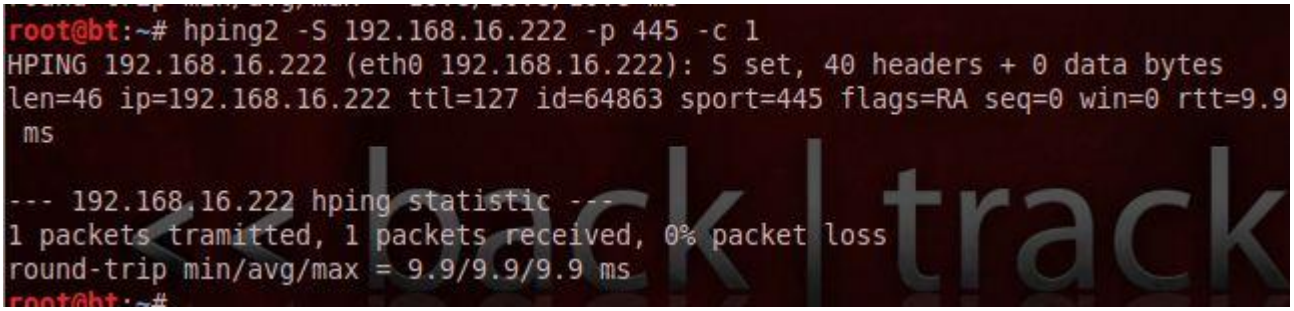
  len=46 ip=192.168.1.1 flags=RA DF seq=0 ttl=255 id=0 win=0 rtt=0.4 ms tos=0 iplen=40 seq=0 ack=1223672061 sum=e61d urp=0
- **Tos:**Kiểu dịch vụ trong IP Header
- **Iplen:**IP total len field
- **seq and ack:**Sự nối tiếp những số 32bit và sự ghi nhận trong IP Header
- **Sum:**Tổng kiểm tra IP Header
- **urp:**Giá trị khẩn cấp trong TCP

VII. Demo

(*Thực hiện thăm dò theo đúng phương pháp thăm dò được mô tả trong phần VIII của phần tìm hiểu về Nmap* **).**

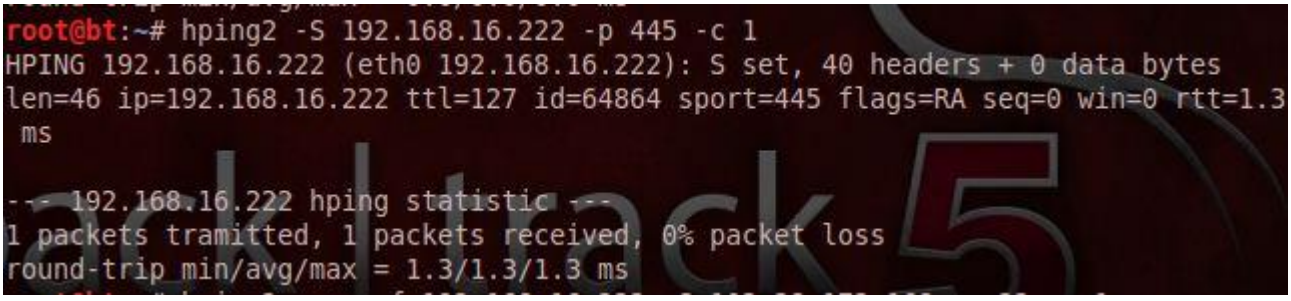
1. Thực hiện quét thăm dò cổng 445.

Bước 1: Thăm dò Zombie cập nhật ID IP: 64863



Bước 2: Thực hiện IDLE

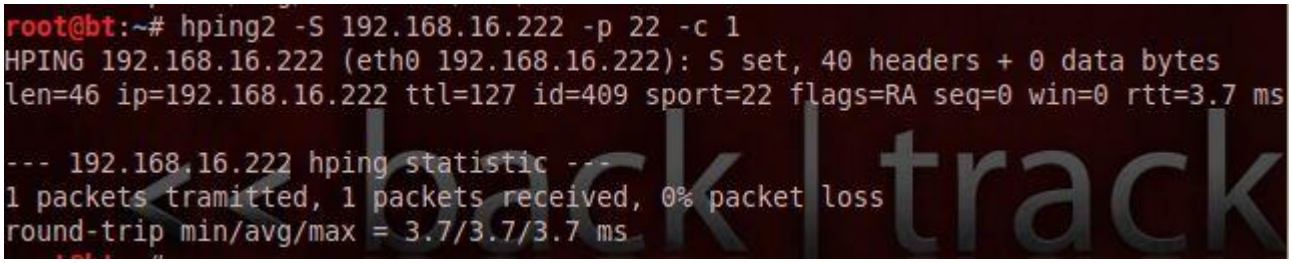
Bước 3: Thăm dò Zombie một lần nữa để lấy lại ID IP : 64864



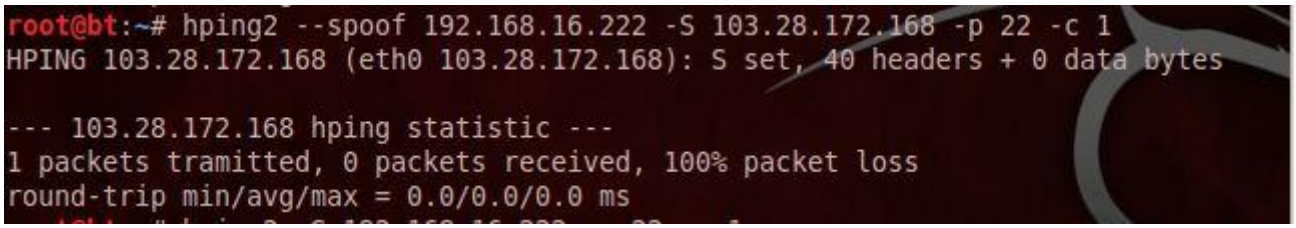
Vậy ta có thể thấy trong trường hợp này ID IP chỉ tăng lên 1 nên ta có thể kết luận rằng cổng 445 đang **close/filter**.

2. Thực hiện thăm dò cổng 22.

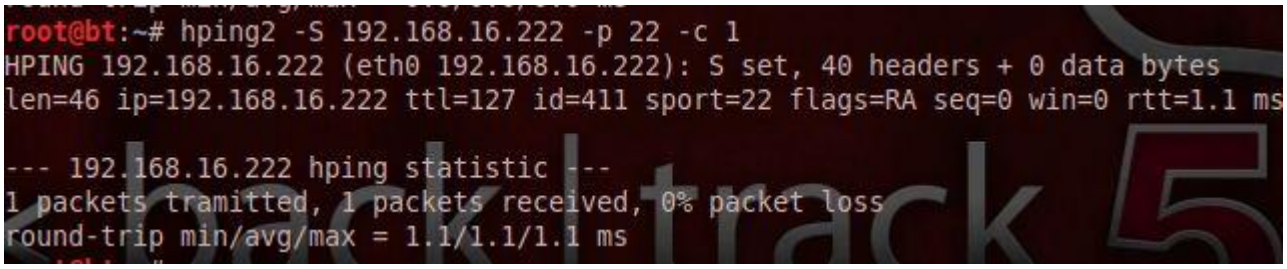
Bước 1: Thăm dò cổng 22 trên Zombie (**ID: 409**)



Bước 2: Thực hiện IDLE với máy Victim



Bước 3: Thăm dò lại cổng 22 trên máy Zombie (**ID: 411**)



Qua thăm dò ta thấy IDIP của Zombie tăng lên 2 (**ID: 409 -- -> ID : 411**)
điều này chứng tỏ cổng 22 đang mở.