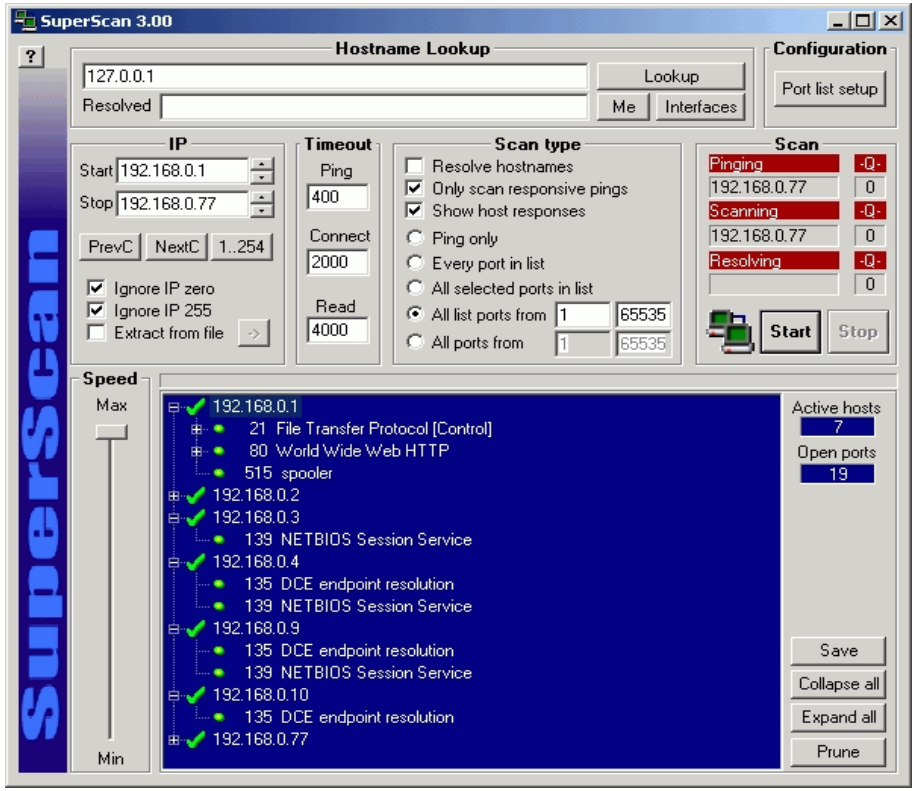
3.1 SCANNING CONCEPTS

- Scanning Objectives
- Scan Types
- Scanning Tools
- Packet Crafting
- IPv6 Scanning

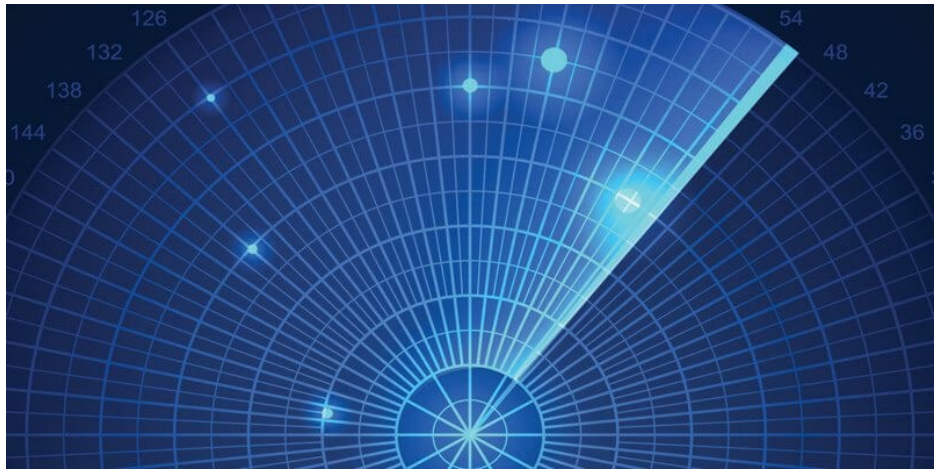
WHAT IS SCANNING?

- First step in active reconnaissance
- Search the network for potential targets



SCANNING OBJECTIVES

- Discover live hosts
- Discover services and listening ports
- Fingerprint OSes and services
- Identify targets for a vulnerability scan

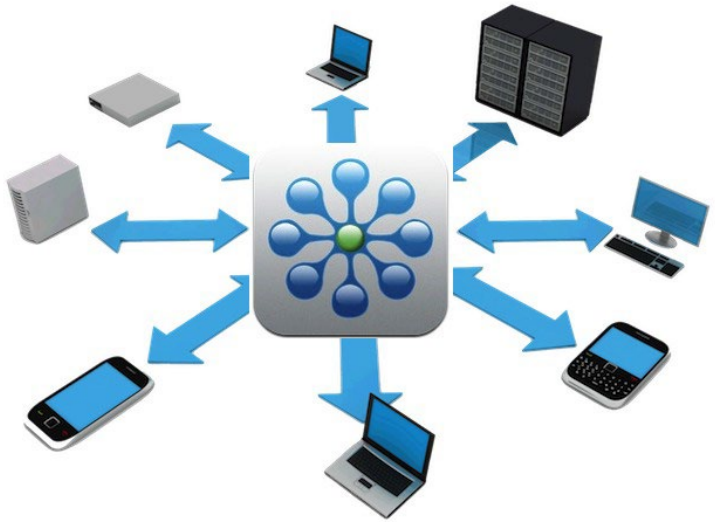


Fingerprinting: identifying an OS or service version through actively engaging the target

The goal of scanning is to ultimately find vulnerable targets that you can exploit!

SCAN TYPES

- Can be:
  - Active (engage the target for information)
  - Passive (sniff traffic for information)
- Discovery Scan
  - Find potential targets
- Port Scan
  - See what services hosts are running
- Vulnerability Scan
  - See if those services are vulnerable to hacking
- Other Scans
  - Map hostnames - IP addresses - MAC addresses
  - Identify additional supported protocols
  - Stealthy alternatives to port scans



Note: Vulnerability scans will be covered later in this course

COMMON SCANNING TASKS

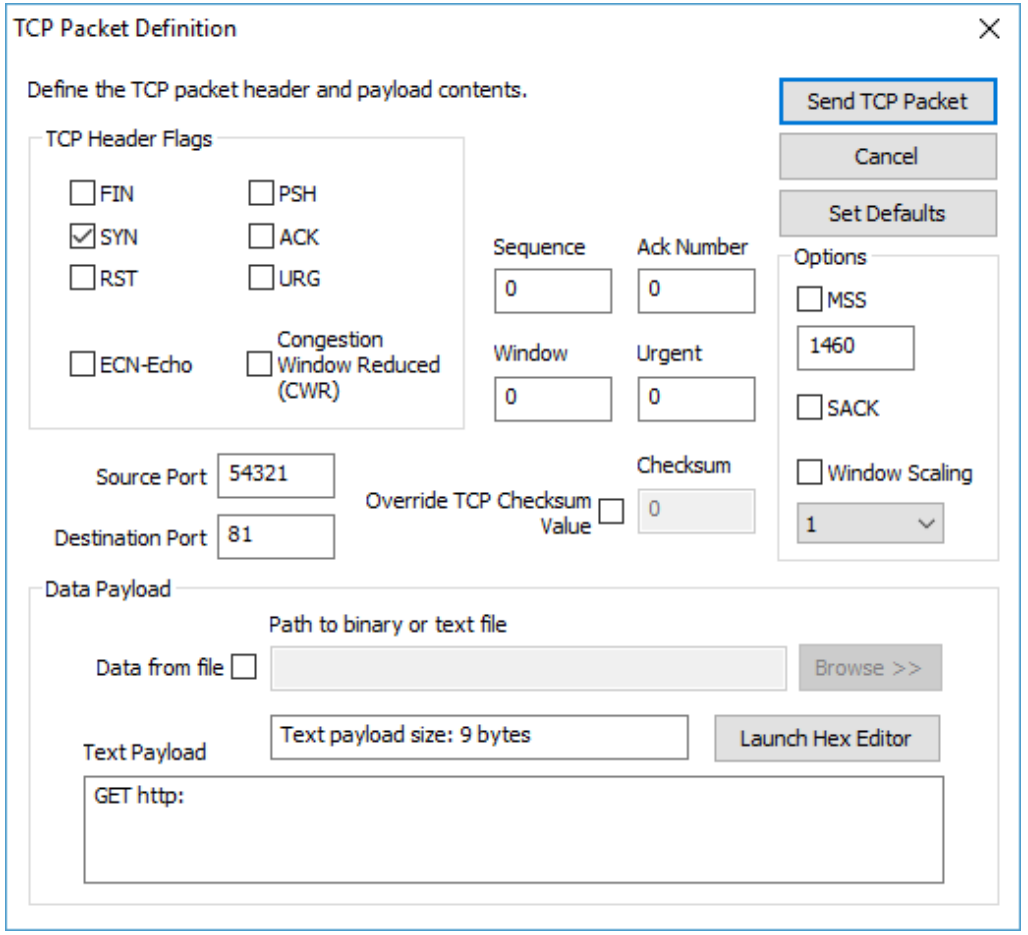
Task	Description
Check for live systems	Ping or ARP to discover live hosts
Check for open ports	Scan live IPs for listening ports
Evade IDS and Firewalls	If necessary, evade detection using proxies, spoofing, fragmented packets, etc.
Perform banner grabbing	Grab from servers Perform OS and service fingerprinting
Scan for vulnerabilities	Test services and OSes for vulnerabilities
Draw network diagrams	Show logical and physical pathways into networks
Pentest Report	Document everything that you find Identify next steps for exploiting vulnerabilities

PACKET CRAFTING

- Used in more advanced scanning
- Doesn't create packets from scratch
- You take a typical IP/ICMP/TCP/UDP packet and:
  1. Specify what settings or values should be in the header fields or payload
  2. Send the packet to the target
  3. See how the target responds to "illegal" or unexpected packet settings
- Different OSes respond in different ways
  - You can often identify the OS based on the response:
    - IP - TTL, Don't Fragment (DF) flag / Don't Fragment ICMP (DFI)
    - TCP - Starting window size, Explicit congestion notification (ECN) flag
    - Sequence number generation
    - ICMP - echo request / echo reply padding

PACKET CRAFTING TOOLS

- Nmap
- Hping3
- Colasoft
- NetScan Tools Pro
- Cat Karat
- Ostinato
- WAN Killer
- Packeth
- LANforge FIRE
- Bit-Twist
- WireEdit



SCANNING IN IPV6 NETWORKS

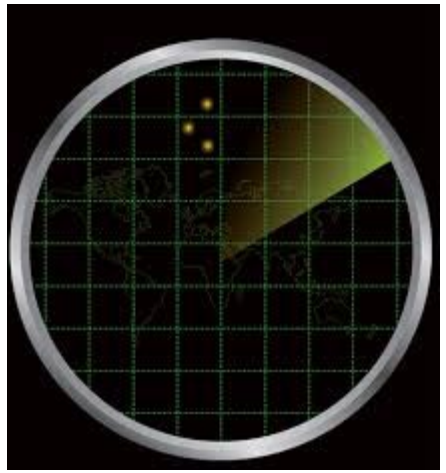
- IPv6 addresses are 128 bits
- Traditional scanning techniques are not feasible because of the larger search space (64 bits)
- Some scanning tools do not support scanning IPv6 networks
- Attackers may gather IPv6 addresses from:
  - network traffic
  - recorded logs
  - header lines in archived emails
  - Usenet news messages
- If an attacker does discover and compromise one host:
  - They can probe the "all hosts" link local multicast address FF01::1
  - Discover additional targets on the link

3.2 ICMP DISCOVERY SCANS

- ICMP
- ARP Discovery
- Other Techniques

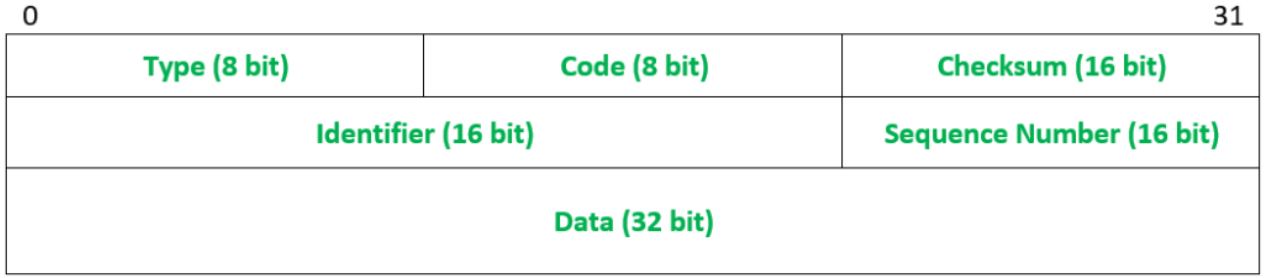
WHAT IS A DISCOVERY SCAN?

- A type of scan that discovers live IP addresses on a network
- A Ping Sweep is the simplest network scanning method
  - It uses ICMP ECHO REQUEST packets to search for live hosts
- Many discovery scans use some form of ARP instead of ICMP to bypass host-based firewalls
- Can also use specially crafted TCP or UDP packets



ICMP

- Internet Control Messaging Protocol
- Layer 3 protocol
- Direct payload of IP
- Protocol ID 1
- Has message types
- Each message type in turn may have codes for further information



Note: You can view ICMP types and codes here:
http://networksorcery.com/enp/protocol/icmp/msg3.htm

IMPORTANT ICMP TYPES

ICMP Message Type	Description and Codes
0: Echo Reply	Answer to a Type 8 Echo Request
3: Destination Unreachable	Error message followed by these codes: 0 - Destination network unreachable 1 - Destination host unreachable 6 - Network unknown 7 - Host unknown 9 - Network administratively prohibited 10 - Host administratively prohibited 13 - Communication administratively prohibited
4: Source Quench	A congestion control message

IMPORTANT ICMP TYPES (CONT'D)

ICMP Message Type	Description and Codes
5: Redirect	Sent when there are two or more gateways available for the sender to use. Followed by these codes: 0 - Redirect datagram for the network 1 - Redirect datagram for the host
8: Echo (request)	A ping message, requesting an echo reply
11: Time Exceeded	Packet took too long to be routed (code 0 is TTL expired)

ICMP SCANNING

- The easiest protocol to use to scan for live systems
  - Scanner sends ICMP ECHO requests to one or more IP addresses
  - If live, hosts will return an ICMP ECHO REPLY
- Useful for locating local devices
- Often blocked by:
  - Software firewall on the host
  - Packet filtering router/firewall between the scanner and target network
- Useful for determining if a firewall is permitting ICMP
  - Example:
    - ICMP Echo returns Type 3 Code of 13 "Destination unreachable administratively prohibited"
    - This type of message is typically returned from a device blocking a port
    - Indicates a firewall that was poorly configured - the firewall should send no response at all

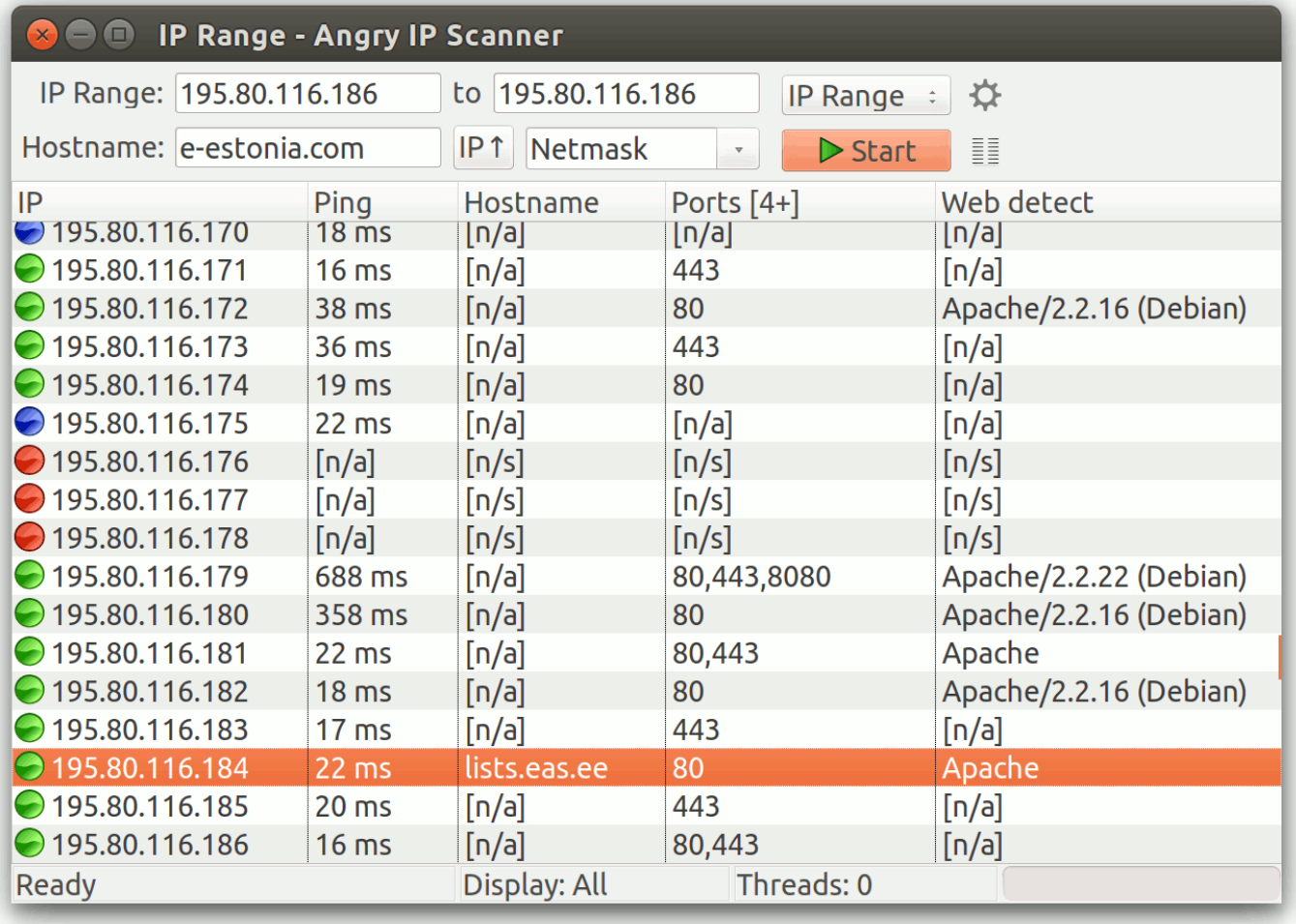
PING SWEEP

- Send ICMP ECHO requests to multiple hosts
  - Traditionally used ICMP ECHO
  - Now uses ARP, TCP, or other protocols
  - Usually swiftly, in numerical order
- Only a live host will reply
- You can use the subnet mask to determine the range of addresses to scan
- You can record the live hosts in a list for further scanning

PING SWEEP TOOL EXAMPLES

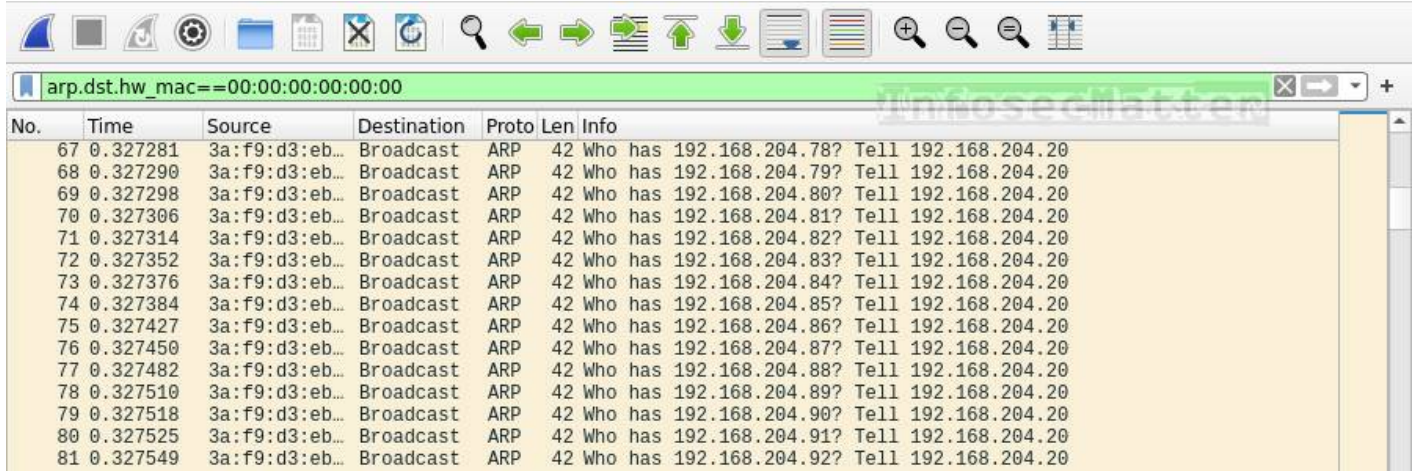
- Nmap
- hping3
- Angry IP Scanner
- SolarWinds Engineer Toolkit
- Colasoft Ping Tool
- SuperScan
- Visual Ping Tester
- Ping Scanner Pro
- OpUtils
- PingInfoView
- Advanced IP Scanner
- Ping Sweep
- Network Ping
- Ping Monitor
- Pinkie

PING SWEEP EXAMPLE



ARP DISCOVERY

- Use ARP requests/replies to discover live hosts
- Cannot be blocked by a personal firewall
  - ARP is required to discover MAC addresses and map them to IP addresses
  - Used on an Ethernet or Wi-Fi LAN
- Tools include:
  - Nmap
  - Ettercap
  - Metasploit
  - Cain & Abel



OTHER DISCOVERY TECHNIQUES

- TCP SYN, ACK, FIN, etc. packets to common ports such as 80 or 443
- ICMP timestamp
  - Used by network routers to synchronize their system clocks for time and date
- SCTP Init
  - A newer Layer 4 protocol that can manage sessions
  - Uses a heartbeat to immediately notify if a connection is down
  - Available in some versions of Linux and Solaris
- You could also:
  - Start port scanning a host without first checking if it is up or down
  - Perform an IP protocol scan to see if the host responds to other Layer 3/4 protocols

3.3 PORT SCANS

- Ports Overview
- Common Ports
- TCP Port Scanning
- UDP Port Scanning

WHAT IS A PORT?

- A number (0 - 65535) that represents a process on a network
  - Well-known services use specific port numbers by convention
  - There is no technical reason for a particular service to use a particular port number
- Both TCP and UDP use port numbers
  - Source and destination each have a port
  - Embedded in the header
  - Indicates the payload
- A client and server will each have its own port in a conversation
  - Usually not the same port
- Some services are only "loosely bound" to a port
  - It is possible for another process to "get in front of" that service
  - Take over the port
  - This happens in hacking
  - Example: netcat getting in front of IIS
    - Intercepts and redirects web traffic

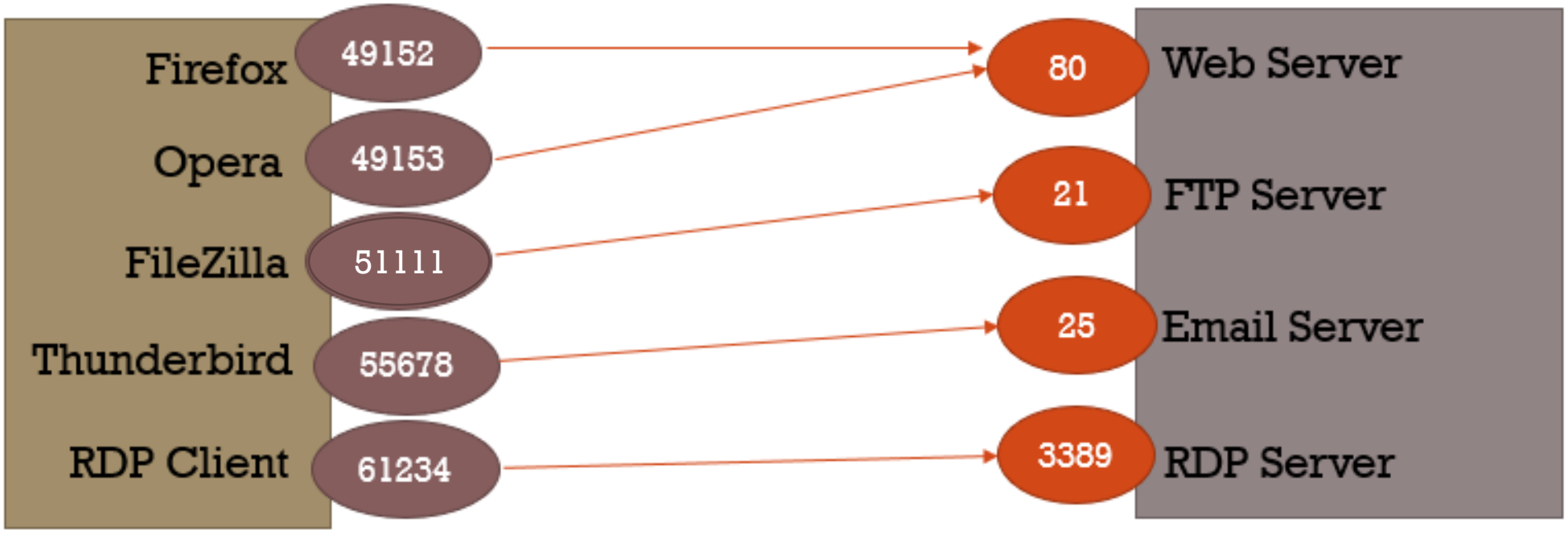
PORT TYPES

- Well-known
  - 0 - 1023
  - 0 is not used
  - Reserved by convention for well-known services
- Registered
  - 1024 - 49151
  - Services can additionally request the use of these ports from the operating system
- Dynamic
  - 49152 - 65535
  - Operating system temporarily assigns a dynamic port to a client process
  - The port is "returned" to the OS when the client process ends

COMMUNICATION USING PORTS

- Client and server ports are usually not the same
- Server listens on well-known port for incoming connection attempts
- Client process, identified by its own port, attempts to make a connection
- The server can accept or reject the connection attempt
  - Usually based on if there is a listening service on that port
  - Can also have firewall filtering or other policies that block connections from specific clients

PORTS EXAMPLE



Client
192.168.1.100

Server
192.168.1.200

COMMON PORT NUMBERS AND SERVICES

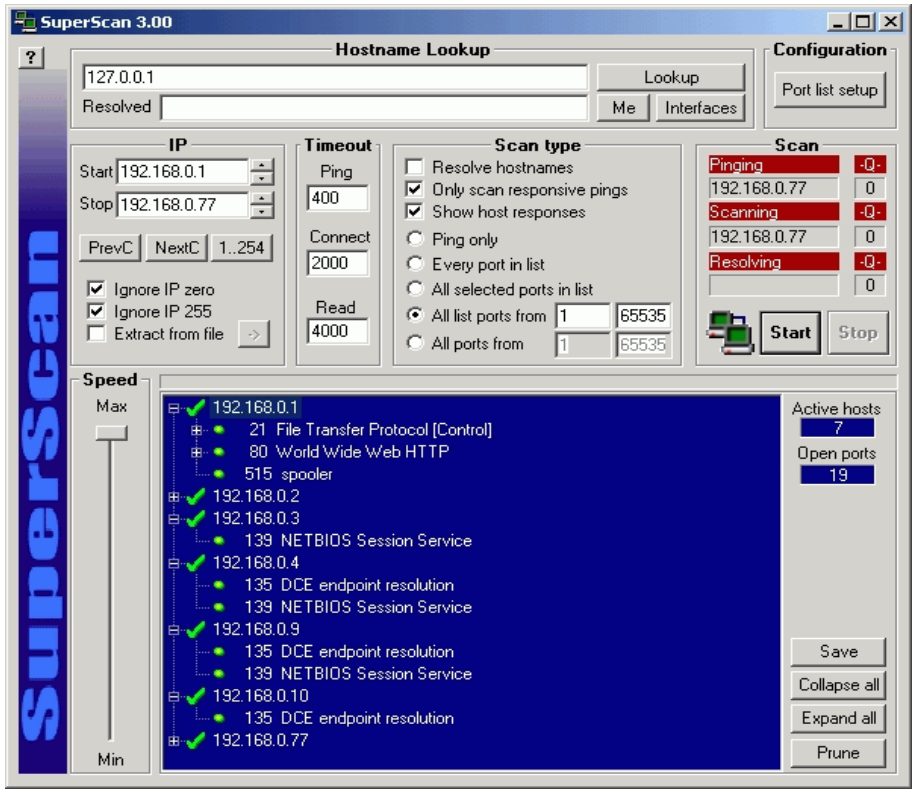
Port Numbers (TCP, unless noted)	Service
21	FTP commands
22	SSH
23	Telnet
25	SMTP
53 (TCP or UDP)	DNS
80	HTTP
88	Kerberos
110	POP3
111 (TCP or UDP)	*nix portmapper

COMMON PORT NUMBERS AND SERVICES (CONT'D)

Port Numbers (TCP, unless noted)	Service
135	Microsoft Remote Procedure Call (RPC)
139	SMB (legacy)
143	IMAP4
161 (TCP or UDP; only UDP is used at this time)	SNMP
162 (TCP or UDP; only UDP is used at this time)	SNMP traps
389	LDAP
443	HTTPS
445	Microsoft-ds (authentication used by SMB)
3389	RDP

SCANNING FOR OPEN PORTS

- Look for open TCP or UDP ports
- An open port indicates a listening service
  - Might have exploitable vulnerabilities
- TCP and UDP respond differently to scans



WHAT IS A TCP PORT SCAN?

- The most common type of port scan
- Attacker sends TCP packets to the target
  - Various TCP header flags are raised (bit set to 1)
- Response can indicate:
  - Listening service
  - OS version
  - Firewall settings

TCP HEADER

TCP Header

Offsets Octet		0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	Source port																Destination port														
4	32	Sequence number																														
8	64	Acknowledgment number (if ACK set)																														
12	96	Data offset				Reserved 0 0 0				NS	CWR	ECE	URG	ACK	PSH	RST	SYN	FIN	Window Size													
16	128	Checksum																Urgent pointer (if URG set)														
20 ...	160 ...	Options (if *data offset* > 5. Padded at the end with "0" bytes if necessary.) ...																														

TCP FLAGS

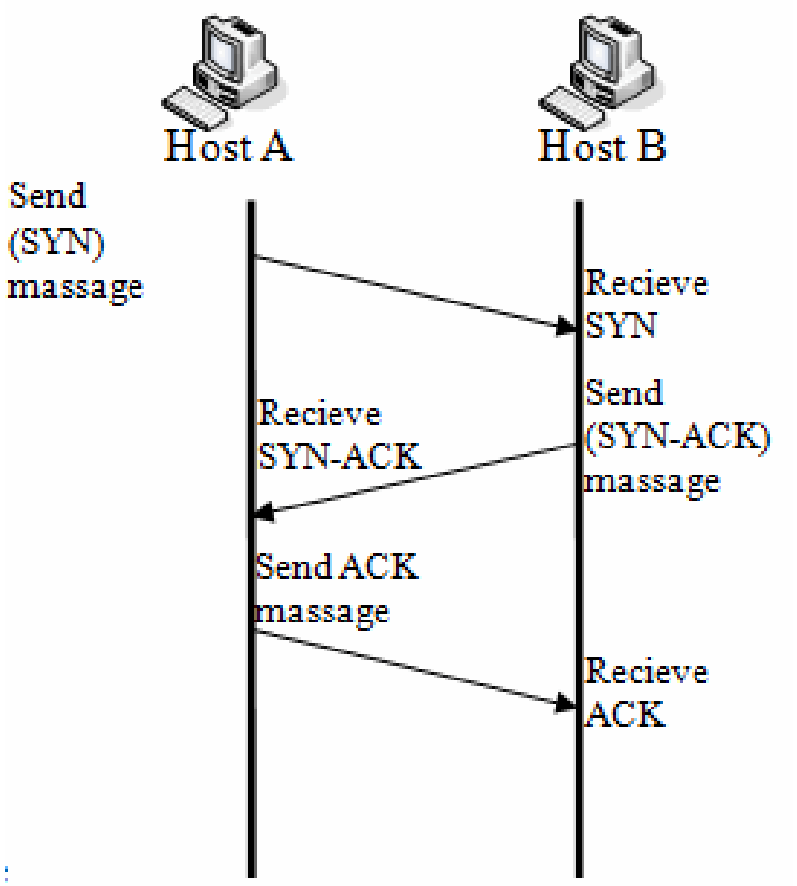
Flag	Name	Function
SYN	Synchronize	Set during initial communication Negotiate parameters and sequence numbers
ACK	Acknowledgment	Set as an acknowledgement to the SYN flag. Always set after initial SYN
RST	Reset	Forces the termination of a connection (in both directions)
FIN	Finish	Part of the close session handshake
PSH	Push	Forces the delivery of data without concern for buffering
URG	Urgent	Data inside is being sent out of band. Example is cancelling a message

TCP FLAGS EXAMPLE

```
⊟ Transmission Control Protocol, Src Port: cisco-wafs (4050),
    Source port: cisco-wafs (4050)
    Destination port: http (80)
    [Stream index: 0]
    Sequence number: 0    (relative sequence number)
    Header length: 32 bytes
  ⊟ Flags: 0x02 (SYN)
      0... .... = Congestion Window Reduced (CWR): Not set
      .0.. .... = ECN-Echo: Not set
      ..0. .... = Urgent: Not set
      ...0 .... = Acknowledgement: Not set
      .... 0... = Push: Not set
      .... .0.. = Reset: Not set
    ⊞ .... ..1. = Syn: Set
      .... ...0 = Fin: Not set
    Window size: 65535
```

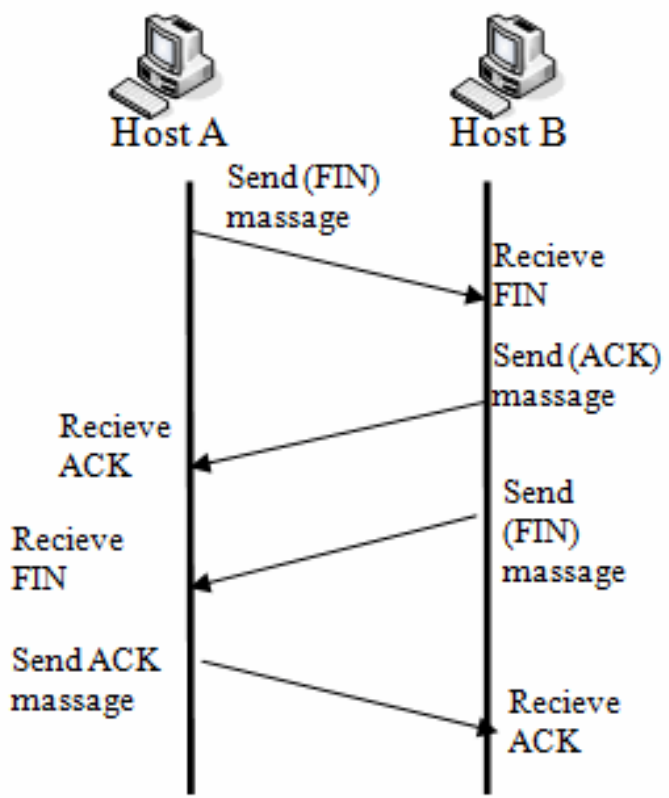
TCP 3-WAY HANDSHAKE

- SYN - SYN-ACK - ACK
- Establish session
- Set starting sequence numbers



TCP 4-WAY GOODBYE HANDSHAKE

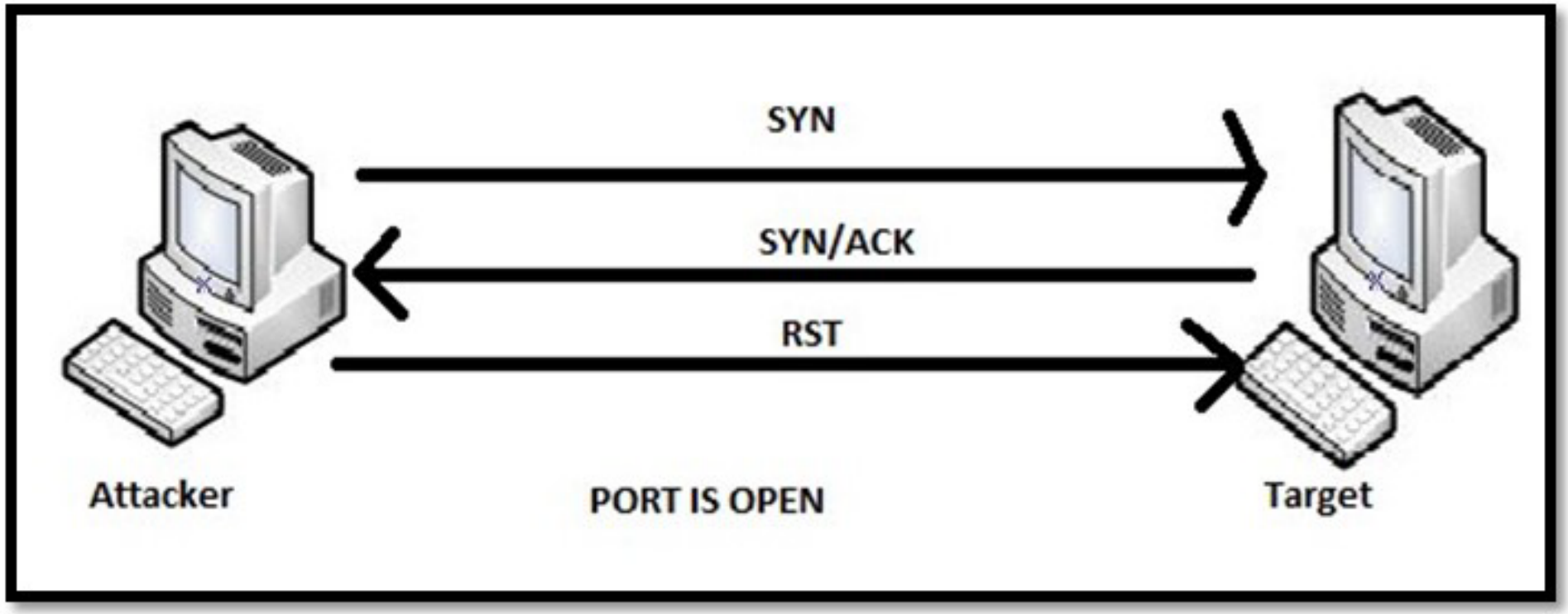
- FIN-ACK - FIN-ACK
- Properly end a session
- Both sides FIN and ACK the other



TCP SYN SCAN

- Most common type of port scan
- AKA Stealth Scan or Half-Open Scan
  - Client sends SYN packet to server
  - Server responds with SYN/ACK packet
  - Server responds with RST packet and remote port is closed
  - Client sends RST packet to close the initiation before connection is established
- Resets TCP connection between client and server in midstream
  - Connection is only “half open”

TCP SYN SCAN EXAMPLE



TCP CONNECT SCAN

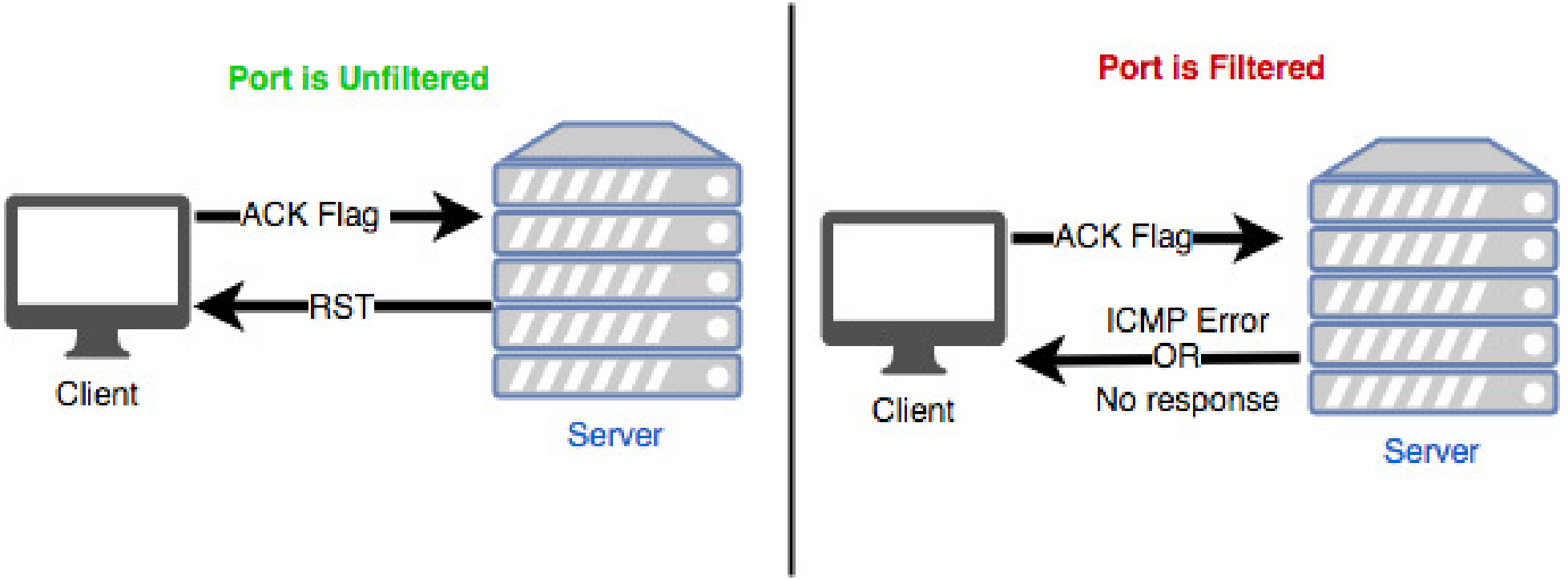
- AKA TCP Full Scan, or TCP Open Scan
- Completes the TCP three-way handshake
- Establishes a full connection
  - Then tears it down by sending a RST packet
- Does not require super user privileges on Linux
- Appears "normal" to intrusion detection
  - Least likely to rouse suspicion

720	35.081879	10.0.0.234	13.107.21.200	TCP	66 21776 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=146
721	35.096204	13.107.21.200	10.0.0.234	TCP	66 443 → 21776 [SYN, ACK] Seq=0 Ack=1 Win=65535 Le
722	35.096234	10.0.0.234	13.107.21.200	TCP	54 21776 → 443 [ACK] Seq=1 Ack=1 Win=263424 Len=0

ACK SCAN

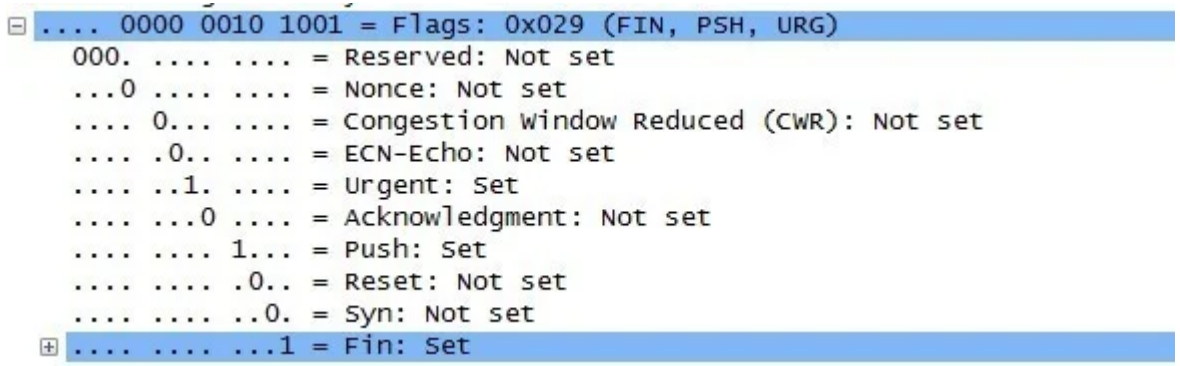
- Used to determine if the host is protected by filtering/firewall
- Since (nearly) every TCP segment contains a raised ACK flag, an ACK scan appears normal
  - Can evade IDS in most cases
  - Can be used against packet filtering routers to see what's behind it
- Attacker sends ACK probe packet with a random sequence number to target
  - No response = protected (filtered) by firewall
  - RST = port is closed
- TTL-based
  - Send 1000s of ACKs to different TCP ports
  - Analyze TTL field in RST packets received
  - If less than the boundary value of 64, then port is open
  - If greater than 64, then port is closed
- Window-based
  - Send 1000s of ACKS to different TCP ports
  - If WINDOW value of RST received has non-zero value, then port is open

ACK SCAN EXAMPLE

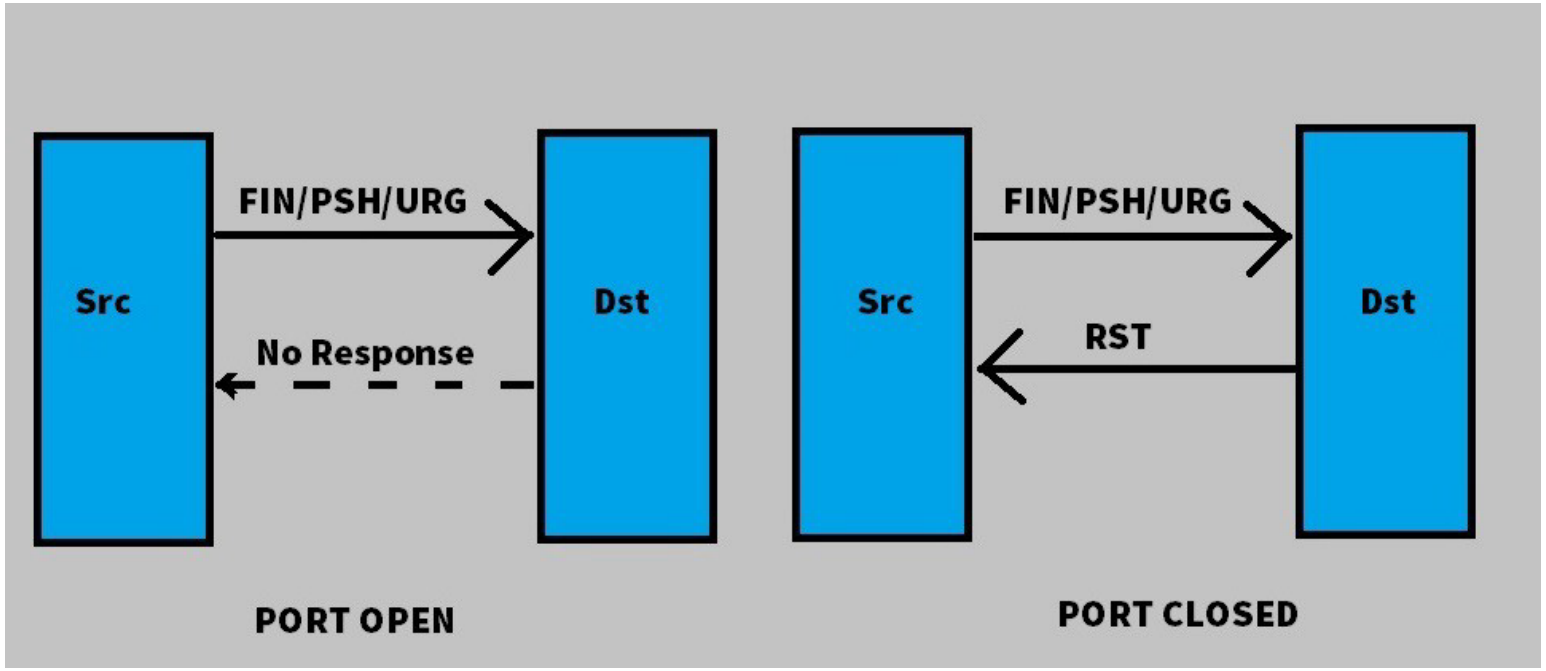


INVERSE TCP FLAG SCANNING

- Stealthier than a SYN scan
- Does not attempt to start a TCP connection
- Used to discover firewall rules / evade detection by IDS
- TCP flags are raised in an unusual / illegal pattern
- Types include:
  - XMAS Scan (PSH, URG, FIN)
  - Null Scan
  - FIN Scan



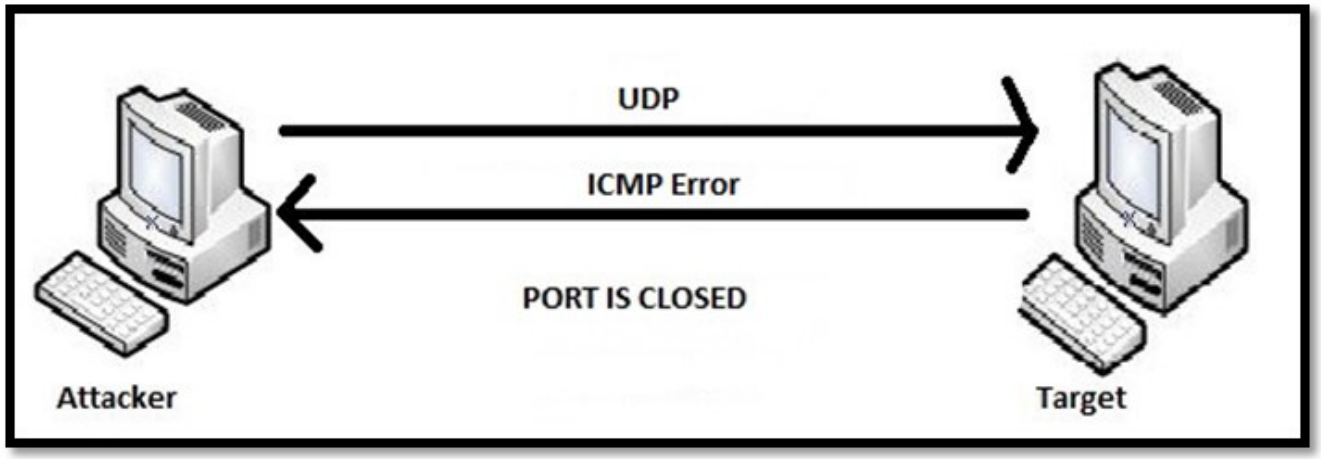
XMAS SCAN EXAMPLE



Note: We will examine TCP scans in greater detail when we study NMAP and Firewalls

UDP PORT SCAN

- No handshake involved
  - UDP is a stateless protocol
- You can send a UDP datagram
  - You often won't get a response
  - UDP itself cannot determine if host is alive, dead or filtered
- Sometimes a UDP closed port will return an ICMP port unreachable message



3.4 OTHER SCAN TYPES

- Headers
- Banner Grabbing
- List Scan
- Zombie Scan
- FTP Bounce
- SSDP Scan

FINGERPRINTING VIA HEADER INFORMATION

TCP:

- Window Size
  - 0x7D78 (32120) = Linux
  - Cisco & Microsoft constantly change

IP:

- TTL
  - 64 = Linux / FreeBSD
- Don't Fragment (DF) bit
  - See if the DF (Don't Fragment) bit is set
  - SCO & OpenBSD do not use the DF flag
- Type of Service (ToS)
  - Indicates the protocol (priority) more than the OS

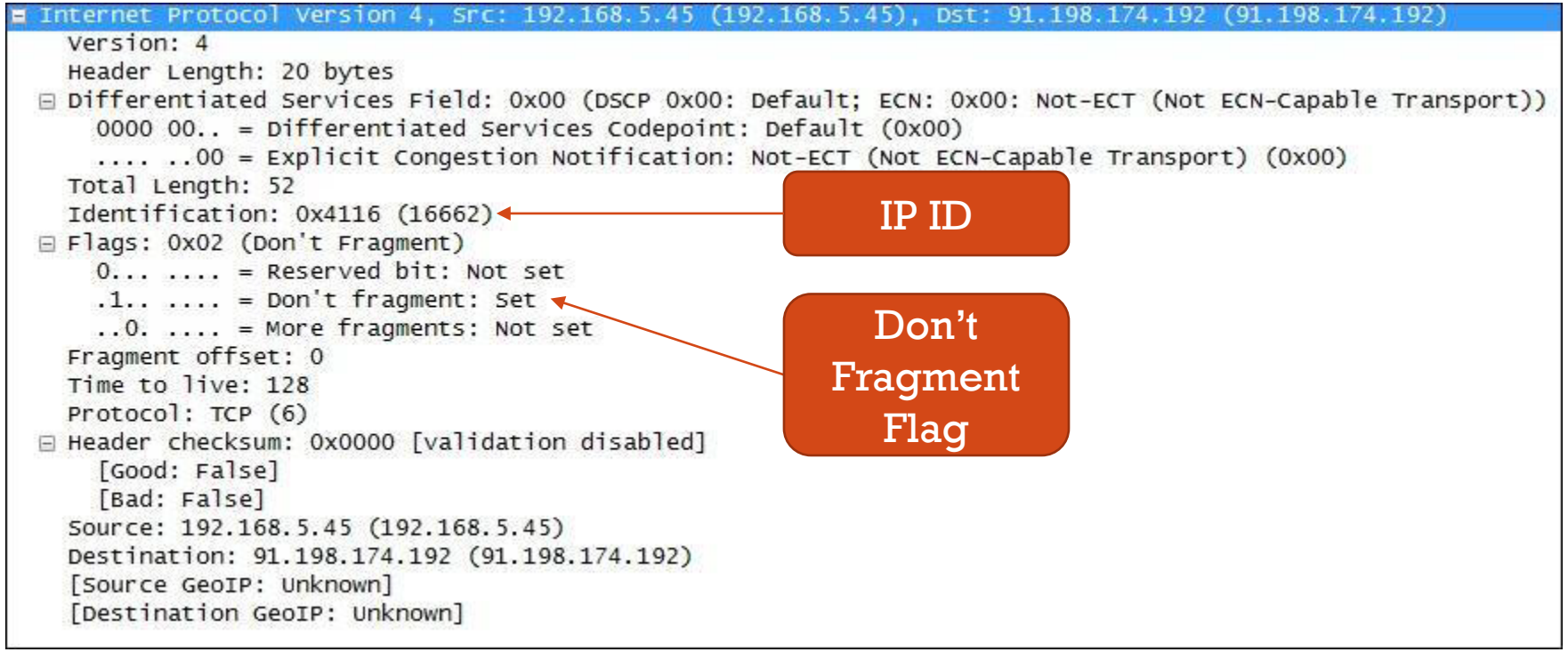
IP HEADER

Version (4 bits)	Header length (4 bits)	Priority and Type of Service (8 bits)	Total length (16 bits)
Identification (16 bits)		Flags (3 bits)	Fragmented offset (13 bits)
Time to live (8 bits)	Protocol (8 bits)	Header checksum (16 bits)	
Source IP address (32 bits)			
Destination IP address (32 bits)			
Options (up to 32 bits)			

```
Internet Protocol Version 4, Src: 192.168.5.45 (192.168.5.45), Dst: 91.198.174.192 (91.198.174.192)
    Version: 4
    Header Length: 20 bytes
  ⊟ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
      0000 00.. = Differentiated Services Codepoint: Default (0x00)
      .... ..00 = Explicit Congestion Notification: Not-ECT (Not ECN-Capable Transport) (0x00)
    Total Length: 52
    Identification: 0x4116 (16662)
  ⊟ Flags: 0x02 (Don't Fragment)
      0... .... = Reserved bit: Not set
      .1.. .... = Don't fragment: Set
      ..0. .... = More fragments: Not set
    Fragment offset: 0
    Time to live: 128
    Protocol: TCP (6)
  ⊟ Header checksum: 0x0000 [validation disabled]
      [Good: False]
      [Bad: False]
    Source: 192.168.5.45 (192.168.5.45)
    Destination: 91.198.174.192 (91.198.174.192)
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
```

IP HEADER

Version (4 bits)	Header length (4 bits)	Priority and Type of Service (8 bits)	Total length (16 bits)
Identification (16 bits)		Flags (3 bits)	Fragmented offset (13 bits)
Time to live (8 bits)	Protocol (8 bits)	Header checksum (16 bits)	
Source IP address (32 bits)			
Destination IP address (32 bits)			
Options (up to 32 bits)			



WHAT IS BANNER GRABBING?

- AKA OS fingerprinting
- A way to determine the operating system running on the remote target system
- Some services identify themselves when queried
- Error messages can reveal information about the service or OS
  - You can use banner grabbing to identify the service and/or OS version
- Can also examine TCP and ICMP messages to identify OS

BANNER GRABBING TYPES

Active Banner Grabbing	Passive Banner Grabbing
Specially constructed packets are sent to the remote OS and responses are captured	Error message provide information including type of server, type of OS, and SSL tools used by target
Responses are compared with a database to determine the OS	Sniffing network traffic on the target allows attacker to determine OS
Response from different Oses vary because of differences in the TCP/IP stack	Page extensions in a URL may assist the attacker in determining versions

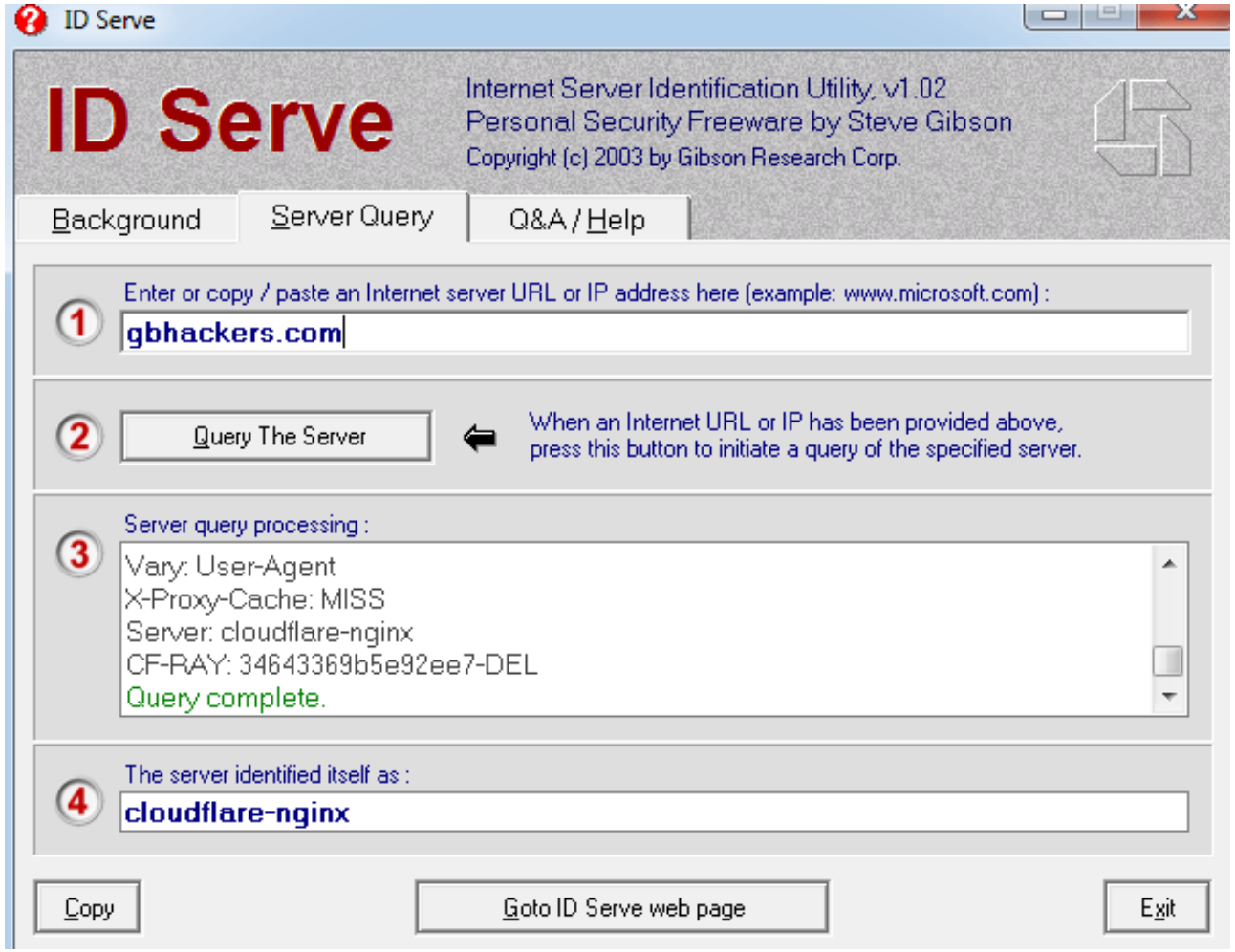
BANNER GRABBING TOOLS

- Many tools can grab banners from various services
  - FTP, SSH, HTTP, SMTP, POP3, IMAP4, DNS, Telnet, Microsoft-DS, Microsoft netbios-ssn, etc.
  - Can help you focus your attacks on specific services
- ID Serve
  - Identifies the make, model, and version of any web site's server software
  - Can also used to identify non-HTTP (non-web) Internet servers: FTP, SMTP, POP, NEWS, etc.
- Netcraft
  - Reports a site's operating system, web server, and netblock owner together with a graphical view at the time of the last reboot for each computer in the site
- Netcat
  - A command-line utility
  - Reads and writes data across network connections using TCP/IP
- Telnet
  - A command-line remote connection utility
  - Will attempt to open a session to whatever port you specify
  - Will display any response received from the server

BANNER GRABBING COMMAND EXAMPLES

```
telnet <target IP> <port number>

nc -vv <target IP> <port number>

echo -en "GET / HTTP/1.0\n\n\n"|nc www.comptia.org 80|grep Server

nmap -sV <target IP> -p <port number>
nmap -sV --script=banner <target>
```

ID SERVE


ID Serve
ID Serve
Internet Server Identification Utility, v1.02
Personal Security Freeware by Steve Gibson
Copyright (c) 2003 by Gibson Research Corp.
Background
Server Query
Q&A / Help
Enter or copy / paste an Internet server URL or IP address here (example: www.microsoft.com) :
gbhackers.com
Query The Server
When an Internet URL or IP has been provided above,
press this button to initiate a query of the specified server.
Server query processing :
Vary: User-Agent
X-Proxy-Cache: MISS
Server: cloudflare-nginx
CF-RAY: 34643369b5e92ee7-DEL
Query complete.
The server identified itself as :
cloudflare-nginx
Copy
Goto ID Serve web page
Exit

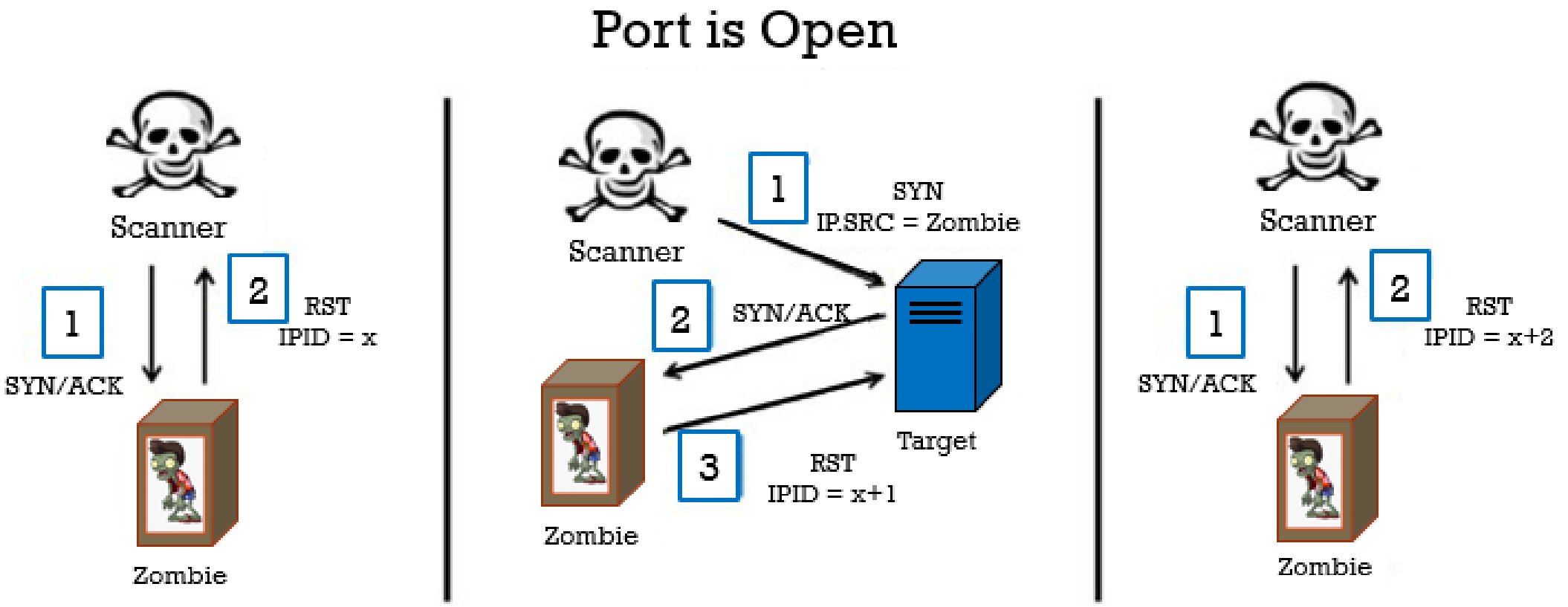
LIST SCAN

- You provide a list of IPs/Names to the scanner
- Does not actually ping
- Performs reverse DNS lookup

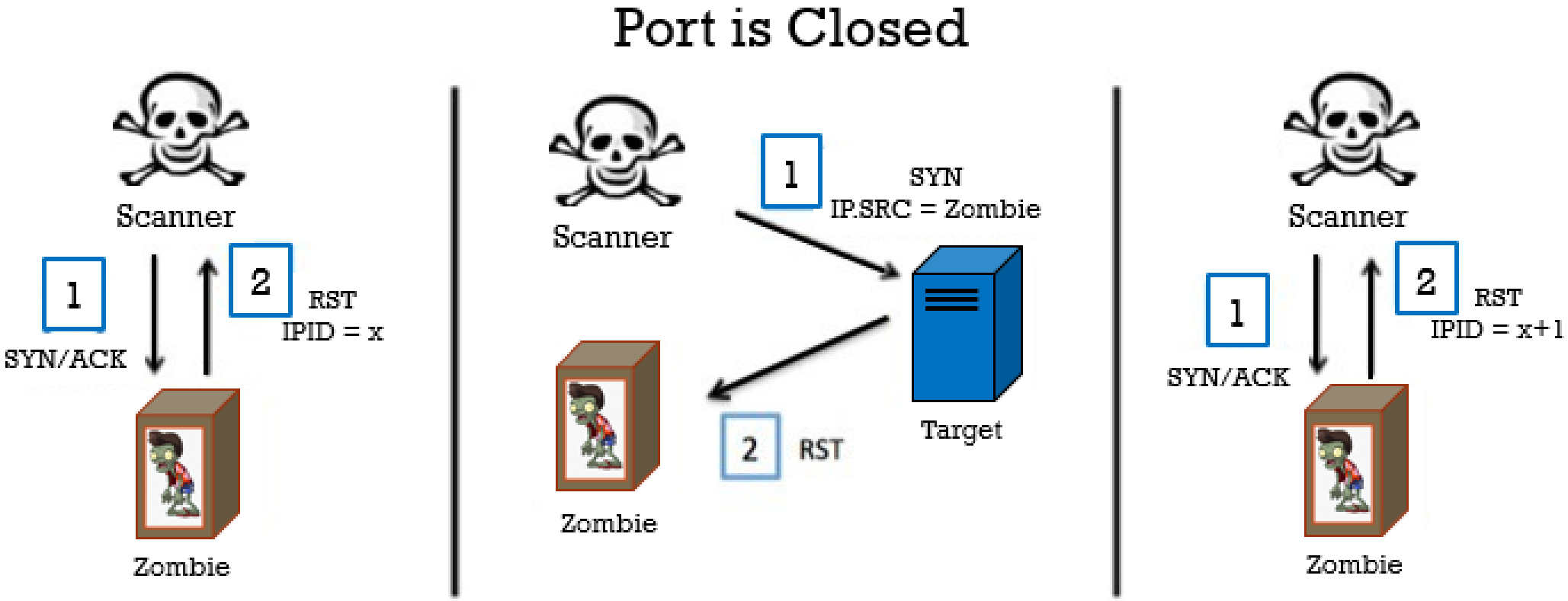
ZOMBIE SCAN

- AKA “blind” scan or “idle” scan
- Map open ports on a remote system without producing any evidence that you have interacted with that system
- Force target to interact with a third machine (zombie)
- Check Zombie’s IPID to see if it incremented
  - IP identification (IPID) identifies a packet in a communication session
  - Its primary purpose is to recover from IP fragmentation

ZOMBIE SCAN EXAMPLE



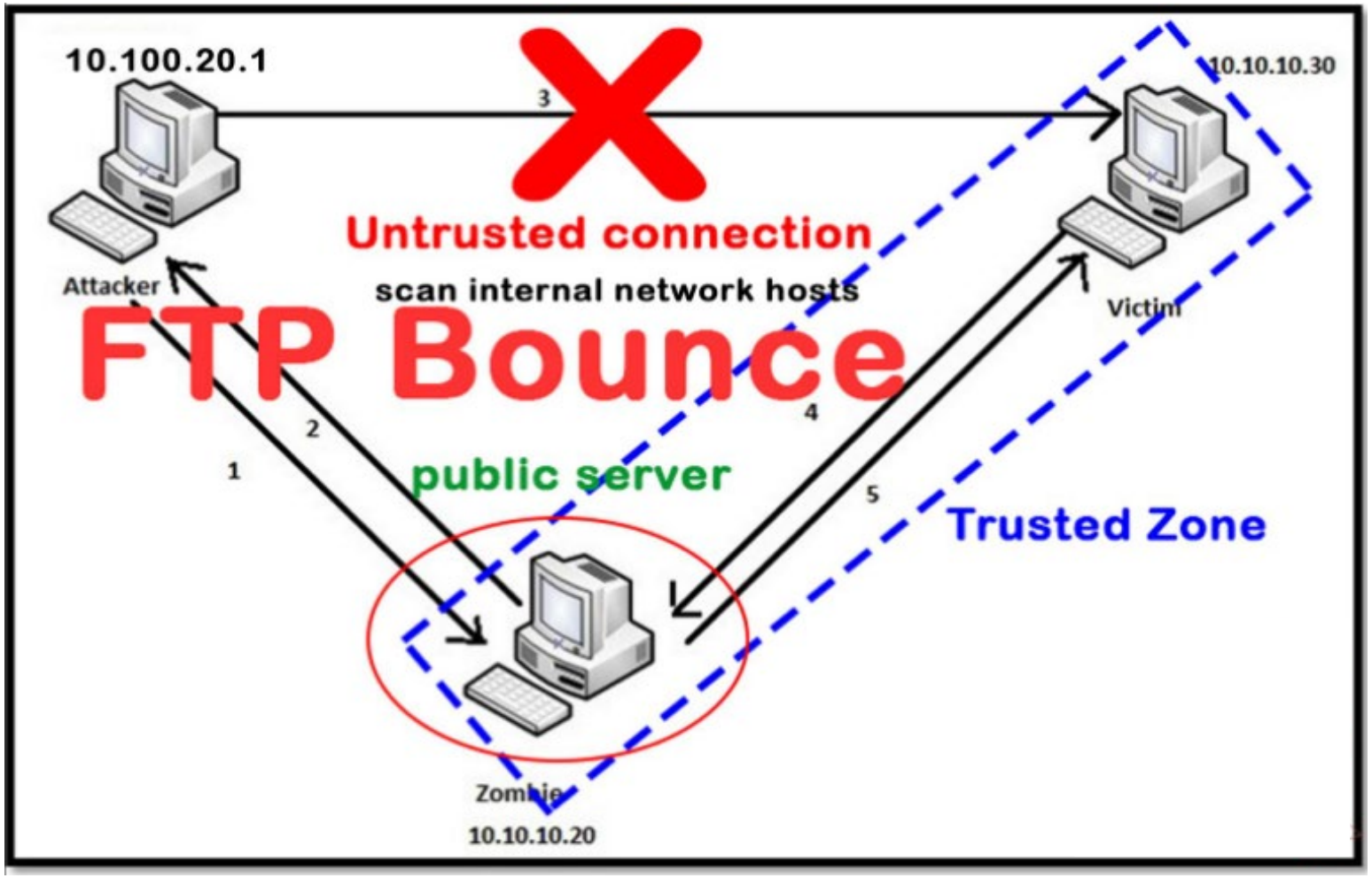
ZOMBIE SCAN EXAMPLE (CONT'D)



FTP BOUNCE SCAN

- Abuses the FTP PORT command and File Exchange Protocol (FXP)
  - An attacker sends the PORT command to an FTP server to redirect the data connection to a third (target) device
  - Target device can be anything the FTP server is capable of reaching
- Used to anonymously scan ports of a target system
- User asks an FTP server to send files to another server
- The returned error message indicates whether the target port is open or not
- Used to bypass firewalls
  - Organizational FTP servers are often:
    - Accessible to the Internet
    - Able to access otherwise protected internal hosts
- Most modern FTP servers now have the PORT command disabled

FTP BOUNCE SCAN EXAMPLE



UNIVERSAL PLUG AND PLAY (UPNP)

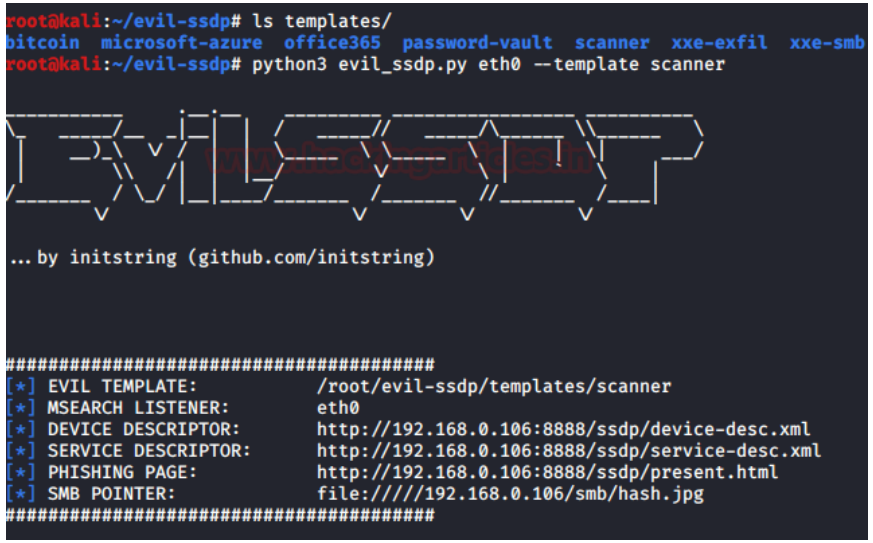
- TCP 1900
- Enables devices like personal computers, Wi-Fi, Mobile devices, printers etc. to discover each other
  - Establish connections for sharing services and data
  - Also for entertainment purposes
  - Intended to be used on residential networks
- Enabled by default on millions of systems
- UPnP-exposed systems connected to the Internet with exploitable vulnerabilities result in a severe security impact
  - These issues potentially expose millions of users to remote attacks
  - Could result in theft of sensitive information or further assaults on connected machines

SSDP (SIMPLE SERVICE DISCOVERY PROTOCOL)

- Used to advertise and discover network services and presence information
- The basis for UPnP device discovery
- Accomplishes this without assistance of server-based configuration mechanisms
  - Such as DHCP or DNS
  - Without special static configuration of a network host
- Intended for use in residential or small office environments

SSDP SCANNING

- Used to discover plug and play devices on the network
- Can discover vulnerabilities you can use to launch Buffer overflow or DoS attacks
- Check if a machine can be exploited
- Usually works when machine is not firewalled
- Can be sent over IPv4 or IPv6



3.5 SCANNING TOOLS

- Tools

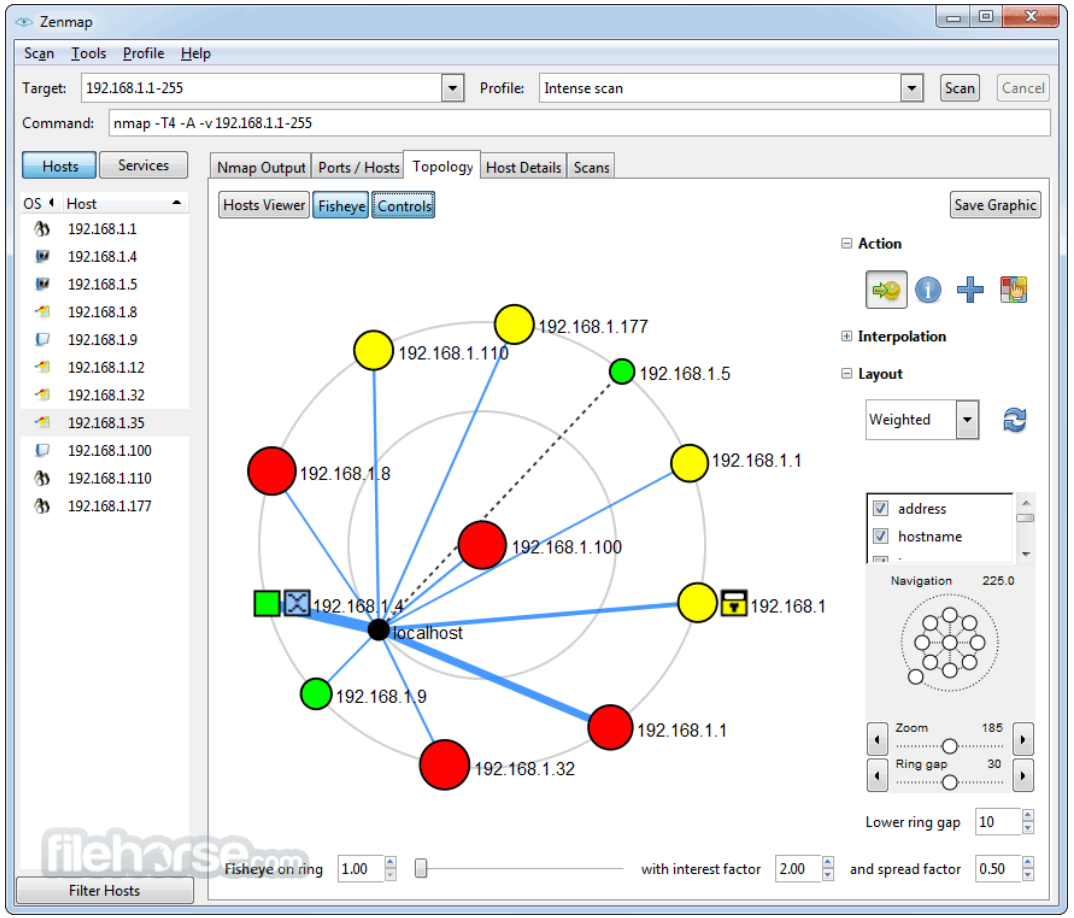
NETWORK MAPPER (NMAP)

- A highly flexible open source tool for scanning networks
- Command-line based for Linux and Windows

```
root@kali:~# nmap 192.168.74.50
Starting Nmap 7.70 ( https://nmap.org ) at 2018-06-13 03:23 EDT
Nmap scan report for 192.168.74.50
Host is up (0.00019s latency).
Not shown: 991 closed ports
PORT    STATE SERVICE
25/tcp  open  smtp
53/tcp  open  domain
80/tcp  open  http
110/tcp open  pop3
135/tcp open  msrpc
139/tcp open  netbios-ssn
143/tcp open  imap
445/tcp open  microsoft-ds
587/tcp open  submission
MAC Address: 00:0C:29:2D:0C:A3 (VMware)
```

ZENMAP

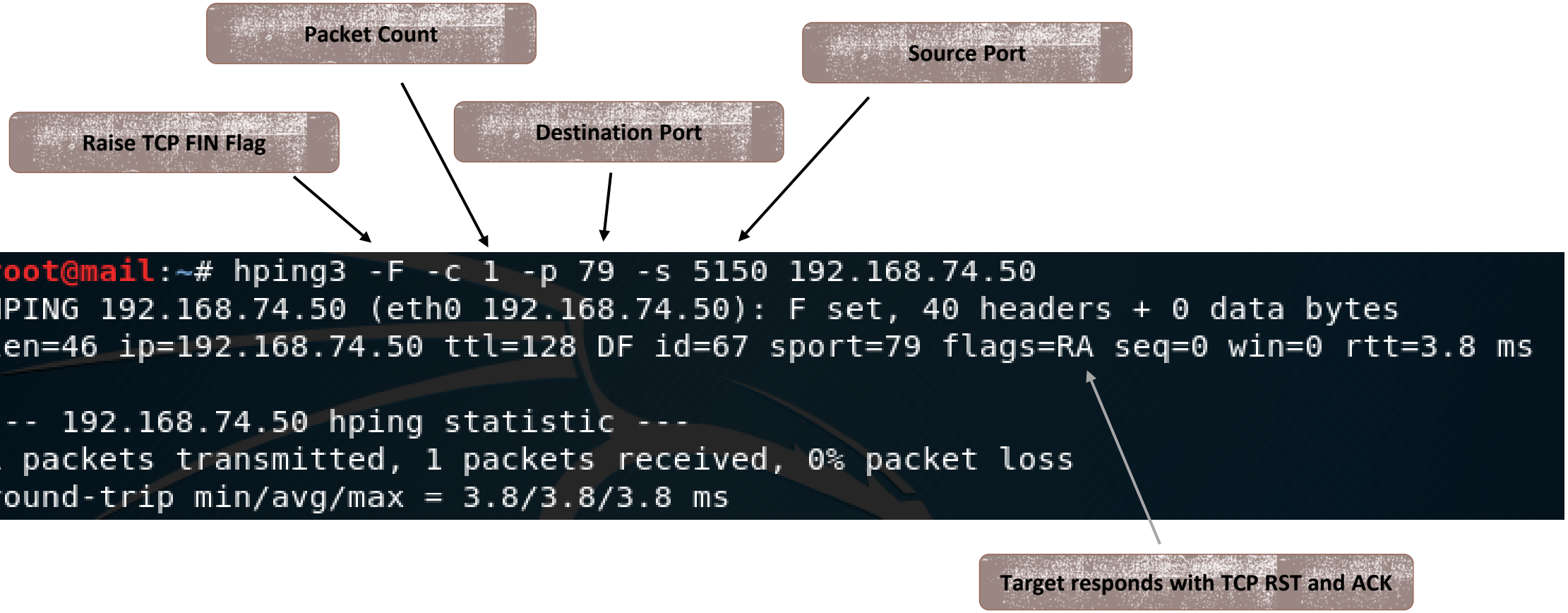
- GUI version of NMAP
- Uses NMAP syntax
- Created for Windows users



HPING2/HPING3

- Command line network scanning
- Packet crafting
- Can perform various scan types
- Used for:
  - Host discovery
  - Network security auditing
  - Firewall testing
  - Manual path MTU discovery
  - Advanced traceroute
  - Remote OS fingerprinting
  - Remote uptime estimating
  - TCP/IP stack auditing

HPING3 PACKET CRAFTING EXAMPLE



HPING AVAILABLE COMMANDS

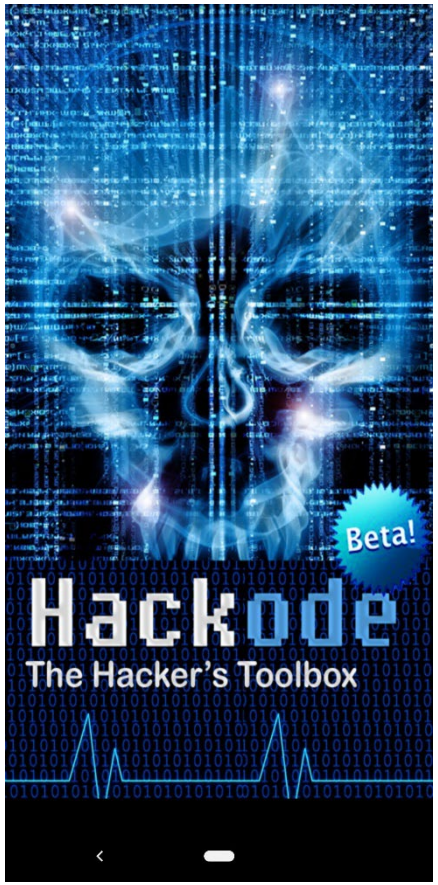
- ICMP Ping
- ACK scan on port 80
- UDP scan on port 80
- Collecting Initial Sequence Number
- Firewalls and Time Stamps
- SYN scan on port 80
- FIN, PUSH, and URG scan on port 80
- Scan entire subnet for live host
- Intercept all traffic containing HTTP signature
- SYN flooding a target

OTHER SCANNING TOOLS

- Angry IP Scanner
- SuperScan
- PRTG
- OmniPeek
- MiTeC Network Scanner
- NEWT Professional
- MegaPing
- Slitheris Network Discovery
- TamoSoft's CommView
- IP-Tools
- Network Scanner
- Global Network Inventory
- Advanced Port Scanner
- CurrPorts
- Masscan
- DRACNMAP
- NEET

SCANNING TOOLS FOR MOBILE DEVICES

- IP Scanner
- Fing
- Hackode
- zANTI
- cSploit
- FaceNiff
- PortDroid Network Analysis
- Pamn IP Scanner



3.6 NMAP

- Features
- Syntax

NETWORK MAPPER (NMAP)

- A highly flexible open source tool for scanning networks
- Command-line based for Linux and Windows
- Also a GUI version (Zenmap) for Windows

NMAP FEATURES

- Host discovery
- Port and service discovery
- Operating system and service fingerprinting
- Enumeration
- MAC address detection
- Vulnerability and exploit detection

Usage: `nmap [Scan Type(s)] [Options] {target specification}`

NMAP SYNTAX

nmap <scan options> <target>

You can combine certain switches: `nmap -sUV 192.168.1.100`

You can refer to a target by name, IP, range, subnet

Examples:

```
nmap 192.168.1.100
nmap 192.168.1.0/24
nmap 192.168.1.*
nmap scanme.nmap.org
nmap 192.168.0.50-100,1.50
```

NMAP EXAMPLE

```
root@kali:~# nmap 192.168.74.50
Starting Nmap 7.70 ( https://nmap.org ) at 2018-06-13 03:23 EDT
Nmap scan report for 192.168.74.50
Host is up (0.00019s latency).
Not shown: 991 closed ports
PORT    STATE SERVICE
25/tcp  open  smtp
53/tcp  open  domain
80/tcp  open  http
110/tcp open  pop3
135/tcp open  msrpc
139/tcp open  netbios-ssn
143/tcp open  imap
445/tcp open  microsoft-ds
587/tcp open  submission
MAC Address: 00:0C:29:2D:0C:A3 (VMware)
```

NMAP HELP

Scan Option	Example	Description
-h	nmap -h	Help on Nmap
-V	nmap -V	Nmap version
-d	nmap -d 192.168.1.50	Enable debugging to view all steps of output

NMAP DISCOVERY SCAN

Nmap Discovery Options	Description
-PR	• Send an ARP (address resolution protocol) request to a target for a response • ARPs are not usually blocked by firewalls • Default discovery method for any nmap scan on an ethernet network
-sn	No port scan Discovery only Use combination of: • ICMP ECHO • TCP SYN to port 443 • TCP ACK to port 80 • ICMP timestamp request
-PS <portlist>	• Discover hosts by sending a TCP SYN to specified port/s • Default is port 80 • Any response (SYN, ACK, RST) demonstrates the target is up • Syntax indicates no space between –PS and the port list • Will be followed by a port scan unless the –sn option is used

NMAP ARP SCAN EXAMPLE

```
┌──(root💀kali)-[~]
└─# nmap -PR 192.168.41.0/24
Starting Nmap 7.91 ( https://nmap.org ) at 2022-01-17 18:31 EST
Nmap scan report for 192.168.41.1
Host is up (0.00014s latency).
Not shown: 998 filtered ports
PORT     STATE SERVICE
80/tcp   open  http
3389/tcp open  ms-wbt-server
MAC Address: 00:50:56:C0:00:08 (VMware)
```

NMAP BASIC TCP AND UDP PORT SCANS

Scan Option	Example	Description
-sS	nmap -sS 192.168.1.50	TCP SYN Scan Send TCP SYN to target for response to check Check for TCP 3-way handshake • If port is open, will respond with SYN ACK • RST if port is closed Requires root privilege
-sT	nmap -sT 192.168.1.50	TCP Connect Scan • Complete a TCP 3-way handshake for non-root users
-sU	nmap -sU 192.168.1.50	UDP scan • Can be very slow • Ports that respond are open • Ports that do not respond are displayed as open|filtered (unknown) • A port might be open but not respond to an empty UDP probe packet • Ports that send ICMP unreachable (type 3 code 3) are closed

NMAP SYN SCAN EXAMPLE

```
┌──(root💀kali)-[~]
└─# nmap -sS 192.168.41.132
Starting Nmap 7.91 ( https://nmap.org ) at 2022-01-17 18:37 EST
Nmap scan report for 192.168.41.132
Host is up (0.0010s latency).
Not shown: 996 closed ports
PORT    STATE SERVICE
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
5357/tcp open  wsdapi
MAC Address: 00:0C:29:A2:C9:50 (VMware)
```

NMAP TCP CONNECT SCAN EXAMPLE

```
┌──(root💀kali)-[~]
└─# nmap -sT 192.168.41.132
Starting Nmap 7.91 ( https://nmap.org ) at 2022-01-17 18:34 EST
Nmap scan report for 192.168.41.132
Host is up (0.00056s latency).
Not shown: 996 closed ports
PORT    STATE SERVICE
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
5357/tcp open  wsdapi
MAC Address: 00:0C:29:A2:C9:50 (VMware)
```

NMAP UDP SCAN EXAMPLE

```
┌──(root💀kali)-[~]
└─# nmap -sUV 192.168.41.131
Starting Nmap 7.91 ( https://nmap.org ) at 2022-01-16 19:45 EST
Stats: 0:00:02 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 3.50% done; ETC: 19:46 (0:00:55 remaining)
Stats: 0:03:22 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 2.10% done; ETC: 22:08 (2:19:51 remaining)
Stats: 0:16:28 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 18.10% done; ETC: 21:14 (1:12:51 remaining)
Stats: 0:40:02 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 48.10% done; ETC: 21:08 (0:42:48 remaining)
Stats: 1:07:37 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 82.10% done; ETC: 21:07 (0:14:40 remaining)
Stats: 1:07:42 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 82.10% done; ETC: 21:07 (0:14:41 remaining)
Stats: 1:08:27 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 82.10% done; ETC: 21:08 (0:14:51 remaining)
Stats: 1:08:32 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 82.10% done; ETC: 21:08 (0:14:52 remaining)
Nmap scan report for 192.168.41.131
Host is up (0.00031s latency).
Not shown: 999 open|filtered ports
PORT    STATE SERVICE    VERSION
137/udp open  netbios-ns Microsoft Windows netbios-ssn (workgroup: WORKGROUP)
MAC Address: 00:0C:29:A7:AC:CE (VMware)
Service Info: Host: DESKTOP-M8UA2CO; OS: Windows; CPE: cpe:/o:microsoft:windows
```

NMAP LIST AND SERVICE VERSION SCANS

Scan Type	Example	Description
-sL	nmap -sL 4.2.2.* nmap -sL eccouncil.org	List scan • List the target(s) that will be scanned • Attempts to return IP addresses and names for targets • Good for passive reconnaissance
-sV	nmap -sV 192.168.1.50	Probe open ports for service version • Can help disambiguate UDP scans

NMAP LIST SCAN EXAMPLE

- So Verizon has six DNS servers after all…

```
┌──(root💀kali)-[~]
└─# nmap -sL 4.2.2.1-6
Starting Nmap 7.91 ( https://nmap.org ) at 2022-01-17 20:40 EST
Nmap scan report for a.resolvers.level3.net (4.2.2.1)
Nmap scan report for b.resolvers.Level3.net (4.2.2.2)
Nmap scan report for c.resolvers.level3.net (4.2.2.3)
Nmap scan report for d.resolvers.level3.net (4.2.2.4)
Nmap scan report for e.resolvers.level3.net (4.2.2.5)
Nmap scan report for f.resolvers.level3.net (4.2.2.6)
Nmap done: 6 IP addresses (0 hosts up) scanned in 0.02 seconds
```

NMAP SERVICE VERSION SCAN EXAMPLE

```
┌──(root💀kali)-[~]
└─# nmap -sV 192.168.41.132
Starting Nmap 7.91 ( https://nmap.org ) at 2022-01-17 20:51 EST
Nmap scan report for 192.168.41.132
Host is up (0.00043s latency).
Not shown: 996 closed ports
PORT     STATE SERVICE      VERSION
135/tcp  open  msrpc        Microsoft Windows RPC
139/tcp  open  netbios-ssn  Microsoft Windows netbios-ssn
445/tcp  open  microsoft-ds?
5357/tcp open  http         Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
MAC Address: 00:0C:29:A2:C9:50 (VMware)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows
```

NMAP OPTIONS

Option	Example	Description
-p <port range>	• nmap -p 80 192.168.1.50 • nmap -p 80,443 www.company.com • nmap -p1024-3000 192.168.1.0/24 • nmap -p U:53,111,137,T:21-25,80,139,443 192.168.1.0/24 • nmap -p- 192.168.1.50	Scan only specified port/s • Port status can be OPEN, CLOSED (no service on port), or FILTERED (perhaps a firewall) • UDP ports: U • TCP ports: T • ALL TCP ports: -p-
-r	nmap -r 192.168.1.0/24	Scan ports consecutively; not randomly
--top-ports <number>	nmap --top-ports 200	Scan top <indicated number> ports
-6	• nmap -6 2001:f0d0:1003:51::4 • nmap -6 scanme.company.com • nmap -6 fe80::8d50:86ce:55ad:bc5c	Scan IPv6 addresses

NMAP OPTIONS (CONT'D)

Option	Example	Description
-iL <input file name>	nmap -iL /tmp/test.txt	Scan hosts listed in file
--exclude	map 192.168.1.0/24 --exclude 192.168.1.5	Exclude certain hosts from scan
-n	nmap -n 192.168.1.0/24	Do not resolve names (time saver)
-R	nmap -R 192.168.1.0/24	Try to resolve all names with reserved DNS
-F (fast mode)	nmap -F 192.168.1.50	Scan fewer ports than default
-O	nmap -O 192.168.1.50	Enable OS detection, not always accurate

NMAP OPTIONS (CONT'D)

Option	Example	Description
-A	nmap -A 192.168.1.50	Enable OS detection, service version detection, script scanning, and traceroute
--version-intensity <level>	nmap -sV --version-intensity 9 192.168.1.50	Use with –sV • Specified level of interrogation from 0 (light) to 9 (attempt all probes)
-- script=<scriptname>	nmap --script=banner.nse 192.168.1.50	Use NSE script
-sC	nmap -sC 192.168.1.50	Scan using all default scripts
-v	nmap -A -v 192.168.1.50	Increase verbosity of output
-vv	nmap -vv 192.168.1.50	Very verbose output
-oN/-oX/-oS/-oG/-oA <filename>	nmap 192.168.1.50 -oA results.txt	Save output in normal, XML, script kiddie, Grepable, or all

NMAP STEALTH SCAN

Stealth Option	Example	Description
-sS	nmap -sS 192.168.1.50	The original "stealth" scan Half-open scan • Do not complete TCP handshake • If target responds with a SYN ACK, send RST • This is less likely to be logged by the target • Might, however, be noticed by IDS
-Pn	nmap -Pn -p- 192.168.1.0/24	Skip discovery • Assume all hosts are online for port scan • Useful if targets have their firewall up and only offer services on unusual ports

NMAP ACK SCAN

Scan Type	Example	Description
-sA	nmap -sA www.company.com	ACK Scan Find out if a host/network is protected by a firewall. • "Filtered" results indicate firewall is on • "Unfiltered" results indicate port is accessible, but might be open or closed • Run with -A option to determine if accessible ports are actually open or closed (nmap -sA -A www.comptia.org)

NMAP ACK SCAN EXAMPLE

```
┌──(root💀kali)-[~]
└─# nmap -sA 192.168.41.129
Starting Nmap 7.91 ( https://nmap.org ) at 2022-01-17 20:54 EST
Nmap scan report for 192.168.41.129
Host is up (0.00044s latency).
Not shown: 999 filtered ports
PORT   STATE      SERVICE
22/tcp unfiltered ssh
MAC Address: 00:0C:29:8A:BC:EC (VMware)
```

NMAP FIN, NULL, XMAS SCANS

Scan Type	Example	Description
-sF	nmap -sF 192.168.1.50	FIN scan • Raises only a FIN flag • Can be used to disambiguate results of other scans
-sN	nmap -sN 192.168.1.50	NULL Scan • No flags raised • Can sometimes penetrate firewalls and edge routers • An open port will discard with no response • A closed port will send a RST
-sX	nmap -sX 192.168.1.50	XMAS Scan • Raises FIN, URG, PSH flags

These scans can be used to sneak through some stateless firewalls
Works against most UNIX-based systems, but not Microsoft and only some Cisco

NMAP ZOMBIE AND BOUNCE SCANS

Stealth Option	Example	Description
-sI <zombie candidate>	nmap –sI server.example.com	Find a zombie • The target is the zombie candidate
-sI <zombie> <target>	nmap -sI -Pn -p- zombie.example.com www.company.com	Conduct a blind TCP port scan (idle scan) • Assume the target is “up” • Scan all TCP ports • Use the "zombie" (middle man) host to obtain information about open ports on the target
-b <FTP relay> <FTP target>	nmap -Pn -b ftp.microsoft.com google.com	Conduct an FTP bounce scan • Exploit FTP proxy connections (using the PORT command) • A user asks a "middle man" FTP server to send files to another FTP server • Because of widespread abuse, the FTP relay feature has been disabled by most vendors

NMAP DECOYS AND SPOOFING

Stealth Option	Example	Description
-f	nmap -f 192.168.1.50	Split packets (include pings) into 8-byte fragments • Make it more difficult for packet filtering firewalls and intrusion detection to detect the purpose of packets • MTU is the maximum fragment size
-D [decoy1, decoy2, decoy3, etc.] <target>	nmap -D 192.168.1.10 192.168.1.15 192.168.1.30 192.138.1.50	Used to mask a port scan by using decoys • Creates bogus packets from the decoys so the actual attacker blends in with the crowd • Appears that both the decoys and the actual attackers are performing attacks
-e <interface>	nmap -e eth0 192.168.1.50	Specify the interface Nmap should use
-S <spoofed source address>	nmap -e eth0 -S www.google.com 192.168.1.50	Spoof the source address • Will not return useful reports to you • Can be used to confuse an IDS or the target administrator

NMAP DECOYS AND SPOOFING (CONT'D)

Stealth Option	Example	Description
--spoof-mac [vendor type | MAC address]	• nmap -sT -Pn --spoof-mac apple 192.168.1.50 • nmap -sT -PN --spoof-mac B7:B1:F9:BC:D4:56 192.168.1.50	Use a bogus source hardware address • You can specify a random MAC based on vendor, or explicitly specify the MAC address • Hides actual source of scan • Good with ARP ping scan (since ARP will broadcast its response)
--source-port <port number>	nmap --source-port 53 192.168.1.36	Use a specific source port number (spoof source port) • Dupes packet filters configured to trust that port • Same as -g <port number> option
--randomize-hosts	nmap --randomize-hosts 192.168.1.1-100	Randomize the order of the hosts being scanned
--proxies <proxy:port, proxy:port…>	nmap --proxies http://192.168.1.30:8080, http://192.168.1.90:8008	Relay TCP connections through a chain of HTTP or SOCKS4 proxies • Especially useful on the Internet.

NMAP TIMING

Stealth Option	Example	Description
-T <0-5>	nmap 192.168.1.0/24 -T 2	Use different timing templates to throttle the speed of your queries • Slower = make scan less noticeable • T0 is the slowest • T5 is the fastest • Nmap denotes these speeds as: • paranoid, sneaky, polite, normal, aggressive, and insane, respectively • T4 is the recommended choice for a fast scan that is still stable • T3 is the default

3.7 FIREWALL AND IDS EVASION

- NMAP Port States
- Packet Fragmentation
- Source Manipulation
- Decoys
- Timing
- Packet Customization
- Firewalking

FIREWALLS AND SCANNING

- Each firewall configuration presents its own challenges to scanning
- If you are scanning a network "black box" style you do not know which, if any, firewall type you will encounter
- If you can infer which type you're encountering, this will give you an advantage in a pentest
- You'll have a better idea of:
  - Which techniques to not spend too much time on
  - Other approaches you should consider to break into the network

NMAP REPORTED PORT STATES

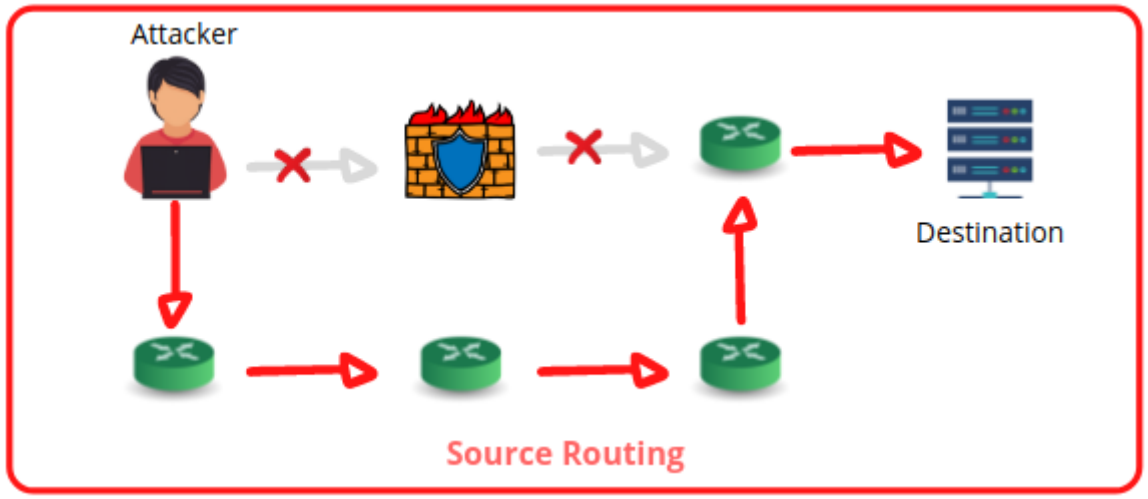
Reported State	Description
Open	This port is actively accepting TCP, UDP or SCTP connections Open ports are the ones that are directly vulnerable to attacks They show available services on a network.
Closed	Target responds (usually with RST) but there is no application listening on that port Useful for identifying that the host exists and for OS detection
Filtered	Nmap can't determine if the port is open because the probe is being blocked by a firewall or router rules Usually no response or "Destination unreachable"

NMAP REPORTED PORT STATES (CONT'D)

Reported State	Description
Unfiltered	Port is accessible but Nmap doesn't know if its open or closed. Only used in ACK scan which is used to map firewall rulesets. Other scan types can be used to identify whether the port is open.
Open/filtered	Nmap is unable to determine between open and filtered. The port is open but gives no response. No response could mean that the probe was dropped by a packet filter or any response is blocked.
Closed/filtered	Nmap is unable to determine whether port is closed or filtered Only used in the IP ID idle scan

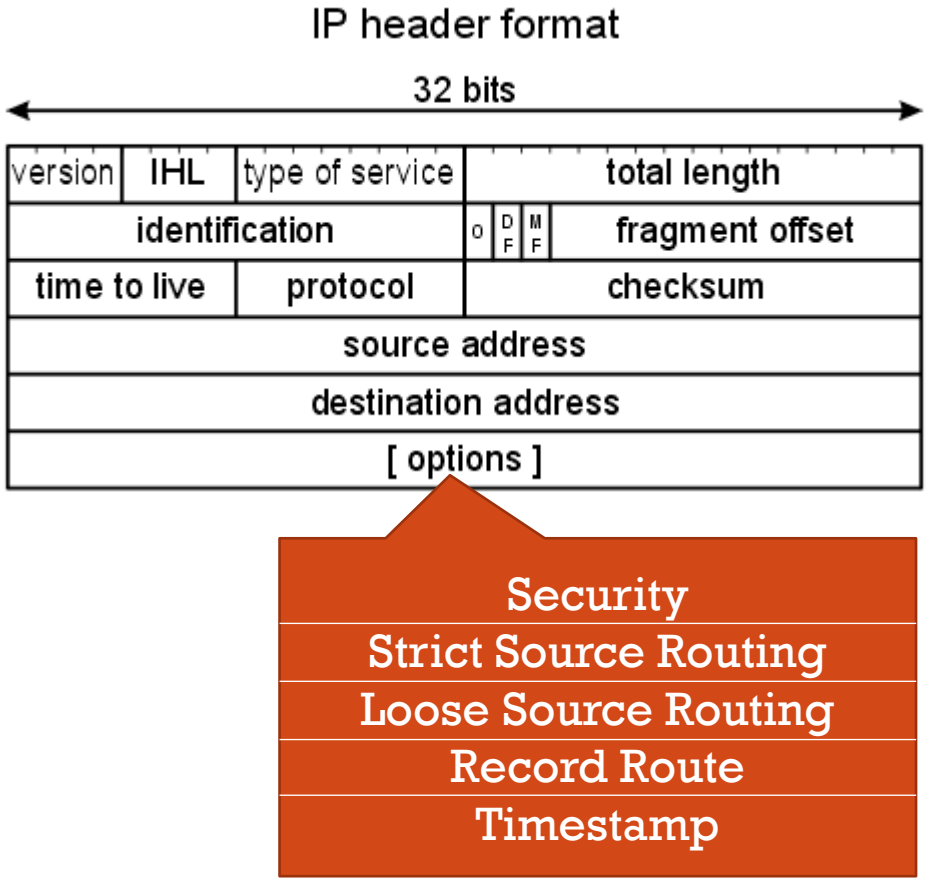
SOURCE ROUTING

- Usually routers dynamically choose the best route to send the packet to its destination
- The IP header OPTIONS field allows the sender to specify the route
- Admins often disable source routing support on routers for security reasons



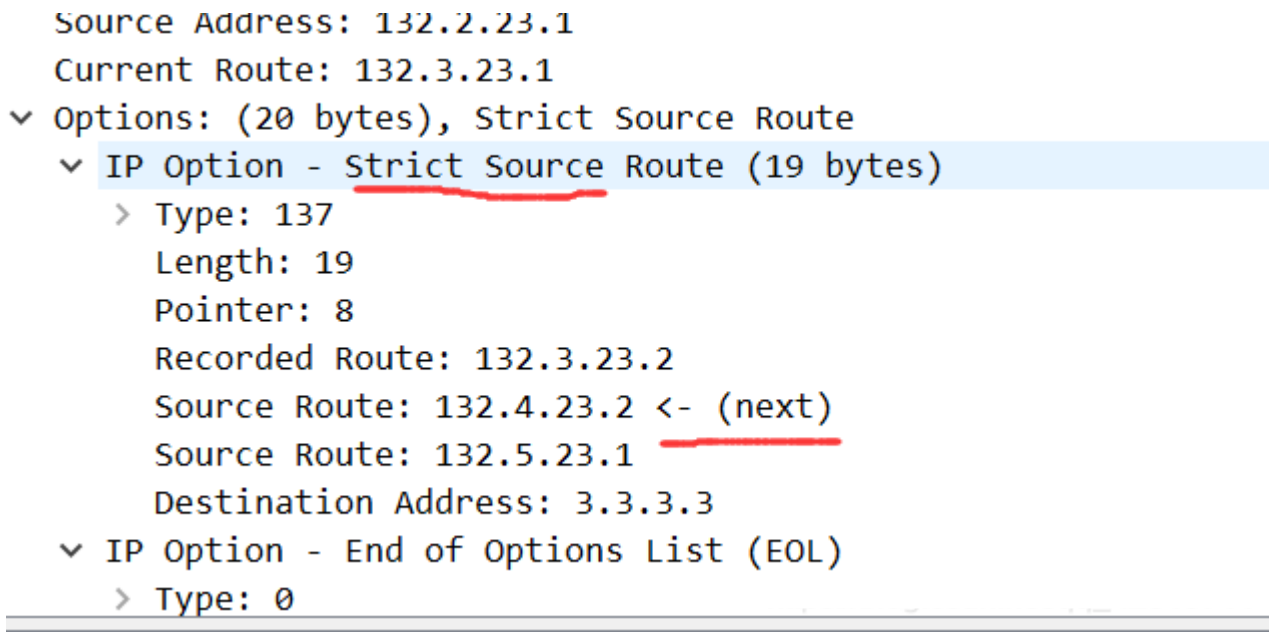
SPECIFYING SOURCE ROUTING

- The sender can specify:
  - EXACTLY which hops a packet must pass through (Strict Source Routing)
  - SOME of the hops a packet must pass through (Loose Source Routing)
- Specified in the [options] field
- You can specify up to 9 hops
- Useful if you know there is an alternate route you can use to go around a firewall
  - Perhaps a dialup connection that would ordinarily not be used



SOURCE ROUTING EXAMPLE

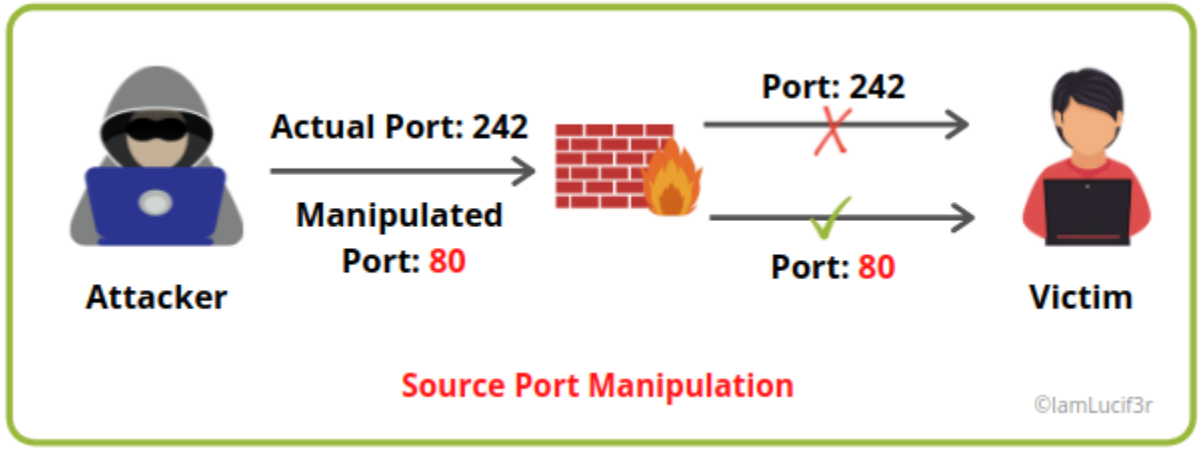
- This Wireshark capture shows that Strict Source Routing was set in the IP header of the captured packet
- Two source routes were inserted into the header
  - One was the sender's outbound address



SOURCE PORT MANIPULATION

- A stateless firewall might be configured to allow packets through that appear to be from a server
- E.g. set the TCP source port to 80 makes the packet appear to come from a webserver

```
nmap -A -T4 -Pn -g 80 -sS 192.168.1.200
```

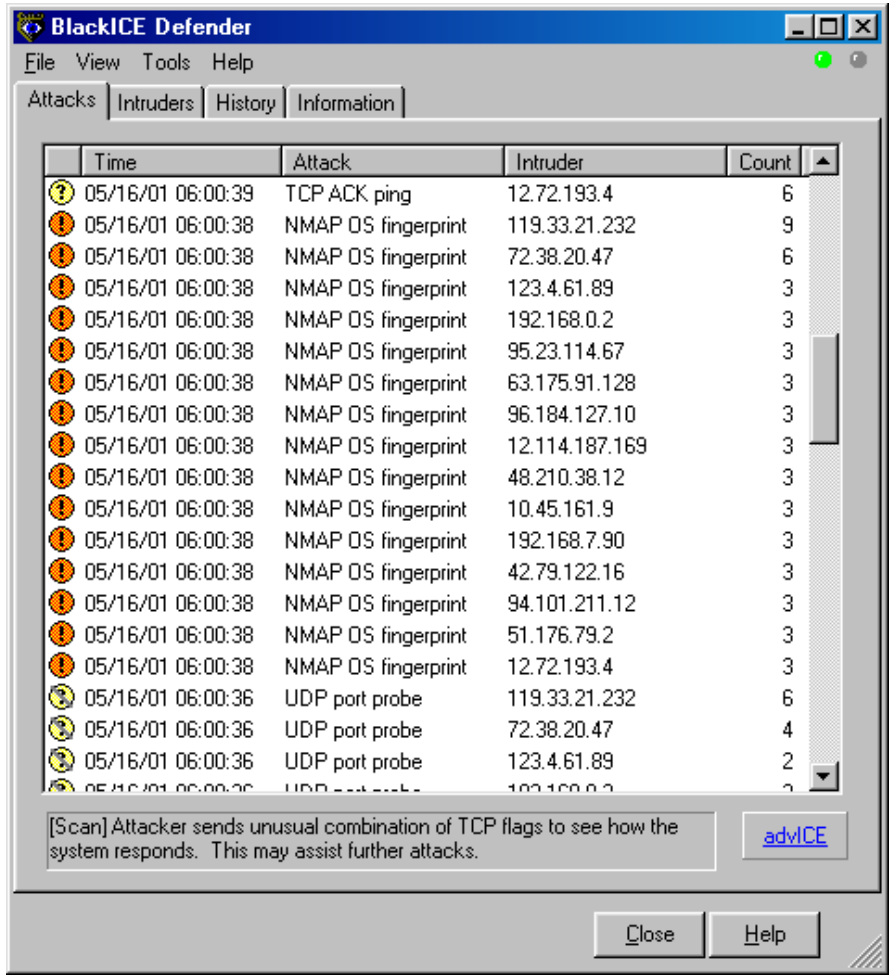


IP ADDRESS DECOYS

- Generates “noise” you can hide in
- Multiple IP addresses appear to be scanning a target simultaneously
- This makes it very difficult for the IDS or sysadmin to determine who the real attacker is
- You can explicitly specify source addresses or allow the scanner to randomly generate addresses

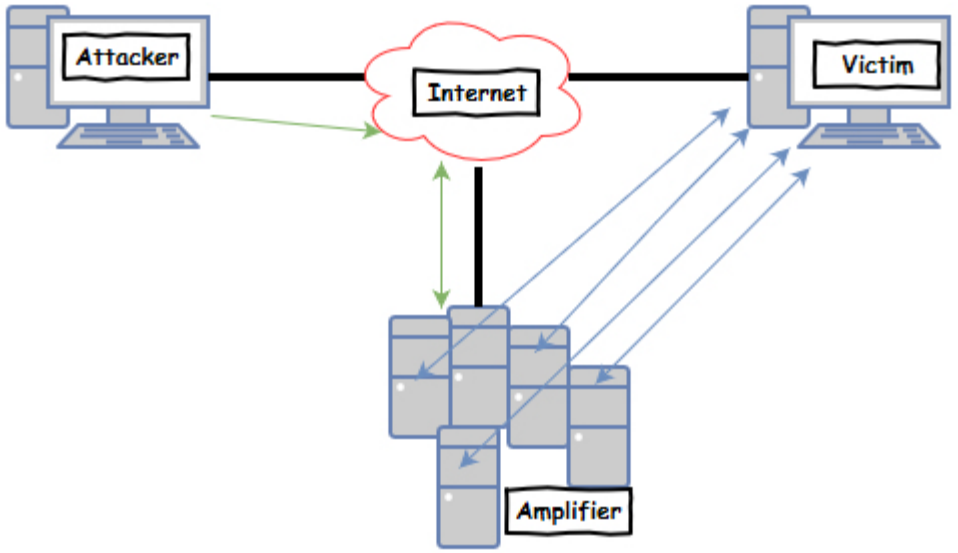
DECOY EXAMPLE

- Which one is the real attacker?



IP ADDRESS SPOOFING

- Used when you want an intermediate machine to "respond" to a victim
- You craft the packet so its source address is actually the victim's address
- Common in Denial-of-Service attacks



TIMING

- A very slow scan will just appear as random noise to the IDS
- It will fall below the threshold necessary to fire an alert
- Make sure addresses and ports are targeted in random order
- A SIEM might detect a very slow scan whereas an IDS might not

WHY PROBE A FIREWALL?

Try to determine:

- The firewall rule set (allowed and blocked ports)
- Firewall type (stateful or stateless)
- Weaknesses in the firewall's configuration
- Devices behind the firewall

PACKET FRAGMENTATION

- The attacker splits the probe packets into several smaller fragments
  - Then sends them to the target network
  - The packet is then reassembled at the final destination
- The IDS/Firewall processes each packet separately
  - Doesn't recognize that the packet is malicious
  - The payload fragments are each too short to match a known signature
- IDSes are often configured to skip fragmented packets during scanning

CUSTOMIZED TCP PACKETS

Technique	Purpose
ACK Scan	• Map out firewall rulesets • Determine if firewall is stateful or stateless
SYN/FIN Scan	• Sets both the SYN and FIN bits • A good way to bypass a rule that drops packets with ONLY SYN raised

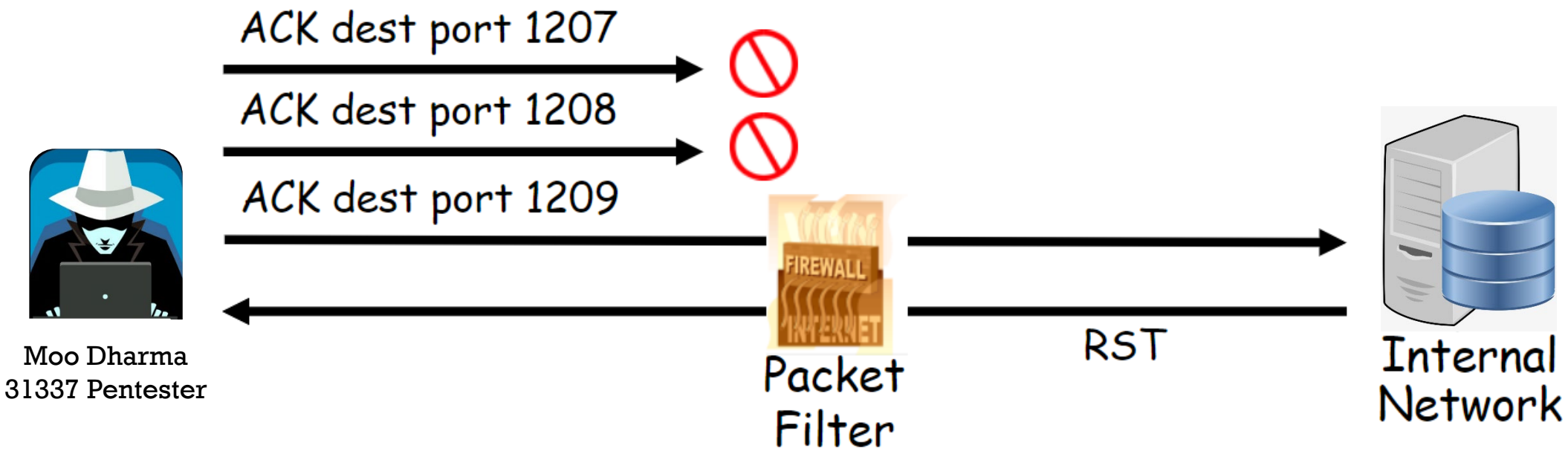
```
krad# nmap -sS --scanflags SYNFIN -T4 www.google.com

Starting Nmap ( http://nmap.org )
Warning: Hostname www.google.com resolves to 4 IPs. Using 74.125.19.99.
Nmap scan report for cf-in-f99.google.com (74.125.19.99)
Not shown: 996 filtered ports
PORT    STATE  SERVICE
80/tcp  open   http
113/tcp closed auth
179/tcp closed bgp
443/tcp open   https

Nmap done: 1 IP address (1 host up) scanned in 7.58 seconds
```

ACK SCAN TO DETERMINE IF A FIREWALL IS STATEFUL OR STATELESS

- A stateless firewall will be easier to get past than a stateful one
- A stateless firewall will block SYN packets based on port number
  - However, it is far less likely to block ACK packets because those could be a response to an outgoing connection
  - Perform separate SYN and ACK scans against the same ports
  - IF the SYN shows some ports open and some closed AND the ACK shows all ports unfiltered, the firewall is likely stateless or disabled
- A stateful firewall will know from its state table if the ACK is legitimate or not
  - If an ACK scan shows at least some ports as "filtered" then it is likely a stateful firewall

ACK SCAN EXAMPLE

- Simple packet filter might have higher level ports open



SYN AND ACK SCAN EXAMPLE

- Scan against Windows 10 with Windows Defender firewall dropped


```
└─# nmap -sS 192.168.41.131
Starting Nmap 7.91 ( https://nmap.org ) at 2022-01-16 18:16 EST
Nmap scan report for 192.168.41.131
Host is up (0.0035s latency).
Not shown: 996 closed ports
PORT     STATE SERVICE
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
5357/tcp open  wsdapi
MAC Address: 00:0C:29:A7:AC:C
```

SYN

ACK

```
┌──(root💀kali)-[~]
└─# nmap -sA 192.168.41.131
Starting Nmap 7.91 ( https://nmap.org ) at 2022-01-1  18:16 EST
Nmap scan report for 192.168.41.131
Host is up (0.0010s latency).
All 1000 scanned ports on 192.168.41.131 are unfiltered
MAC Address: 00:0C:29:A7:AC:CE (VMware)
```

No firewall or stateless; SYN scan returns 4 open ports - no firewall

SYN AND ACK SCAN EXAMPLE (CONT'D)

- Scan against Windows 10 with Windows Defender firewall turned on

Windows Defender = stateful firewall

```
┌──(root💀kali)-[~]
└─# nmap -sS 192.168.41.131
Starting Nmap 7.91 ( https://nmap.org ) at 2022-01-16 18:42 EST
Nmap scan report for 192.168.41.131
Host is up (0.00038s latency).
Not shown: 999 filtered ports
PORT     STATE SERVICE
5357/tcp open  wsdapi
MAC Address: 00:0C:29:A
```

SYN

Stateful firewall

```
┌──(root💀kali)-[~]
└─# nmap -sA 192.168.41.131
Starting Nmap 7.91 ( https:/ g ) at 2022-01 6 18:46 EST
Nmap scan report for 192.168.41.131
Host is up (0.00025s latency).
All 1000 scanned ports on 192.168.41.131 are filtered
MAC Address: 00:0C:29:A7:AC:CE (VMware)
```

ACK

SYN AND ACK SCAN EXAMPLE (CONT'D)

- Scan against CentOS 7 with firewall turned off

```
[root@localhost user]# systemctl status firewalld
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.servic
   Active: inactive (dead) since Sun 2022-01-16 15:57:53 PS
```

```
┌──(root💀kali)-[~]
└─# nmap -sS 192.168.41.129
Starting Nmap 7.91 ( https://nmap.org ) at
Nmap scan report for 192.168.41.129
Host is up (0.0016s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
80/tcp  open  http
111/tcp open  rpcbind
MAC Address: 00:0C:29:8A:BC:
```

SYN

No firewall or stateless; SYN scan returns 3 open ports - no firewall

```
┌──(root💀kali)-[~]
└─# nmap -sA 192.168.41.129
Starting Nmap 7.91 ( https://nmap.org
Nmap scan report for 192.168.41.129
Host is up (0.00078s latency).
All 1000 scanned ports on 192.168.41.129 are unfiltered
MAC Address: 00:0C:29:8A:BC:EC (VMware)
```

ACK

SYN AND ACK SCAN EXAMPLE (CONT'D)

- Scan against CentOS 7 with firewall turned on

iptables = stateful firewall

```
[user@localhost ~]$ firewall-cmd --state
running
[user@localhost ~]$
```

SYN

```
┌──(root💀kali)-[~]
└─# nmap -sS 192.168.41.129
Starting Nmap 7.91 ( https://nmap.org )
Nmap scan report for 192.168.41.129
Host is up (0.00033s latency).
Not shown: 999 filtered ports
PORT   STATE SERVICE
22/tcp open  ssh
MAC Address: 00:0C:29:8A:BC:EC (VMwa
```

Stateful firewall

ACK

```
┌──(root💀kali)-[
└─# nmap -sA 192.   .41.129
Starting Nmap 7.9   ( https://nmap.org ) at
Nmap scan report    or 192.168.41.129
Host is up (0.000   43s latency).
Not shown: 999 filtered ports
PORT   STATE      SERVICE
22/tcp unfiltered ssh
MAC Address: 00:0C:29:8A:BC:EC (VMware)
```

SCANNING THE FIREWALL ITSELF VS SCANNING PAST THE FIREWALL

- Port scanning the firewall's front-facing IP might show ports the firewall itself uses
- A firewall that NATs and port forwards will present the target ports as if they are its own ports
- To distinguish between a permitted port and the firewall's management port:
  - Open a browser to that port
  - Banner grab that port
  - Use nmap –sV to interrogate that port

NULL, FIN, XMAS SCANS

- These are little more stealthy than a SYN scan
- They can sneak past some stateless firewalls and packet filtering routers
- With SYN bit off, they can go past rules that look for SYN raised and ACK set to 0
- You'll need to add -sV to disambiguate open | filtered ports

Technique	Purpose
FIN Scan	Sets only the FIN bit - breaks the rules of TCP; should be accompanied by ACK
NULL Scan	Does not set any TCP bits - breaks the rules; every packet should have some bit set
XMAS Scan	FIN, URG, PSH raised - illogical combination

Probe Response	Assigned State
No response received (even after retransmissions)	open|filtered
TCP RST packet	closed
ICMP unreachable error (type 3, code 1, 2, 3, 9, 10, or 13)	filtered

UDP SCAN

- UDP scan works by sending a UDP packet to every targeted port
- For most ports, this packet will be empty (no payload
- For a few of the more common ports a protocol-specific payload will be sent
- Based on the response, or lack thereof, the port is assigned to one of four states

Probe Response	Assigned State
Any UDP response from target port (unusual)	open
No response received (even after retransmissions)	open|filtered
ICMP port unreachable error (type 3, code 3)	closed
Other ICMP unreachable errors (type 3, code 1, 2, 9, 10, or 13)	filtered

UDP scanning is very slow because nmap must wait for timeout on each port
A Linux kernel will also rate-limit ICMP destination unreachable messages to 1 / second

ABOUT UDP OPEN | FILTERED STATUS

- The the biggest challenges with UDP scanning is that open ports rarely respond to empty probes
- Those ports for which Nmap has a protocol-specific payload are more likely to get a response and be marked open
- For the rest, the target TCP/IP stack simply passes the empty packet up to a listening application
  - which usually discards it immediately as invalid
- If ports in all other states would respond, then open ports could all be deduced by elimination
- Unfortunately, firewalls and filtering devices also drop packets without responding
- If Nmap receives no response after several attempts, it cannot determine whether the port is open or filtered or filtered by a firewall

NMAP UDP SCAN + SERVICE VERSIONING

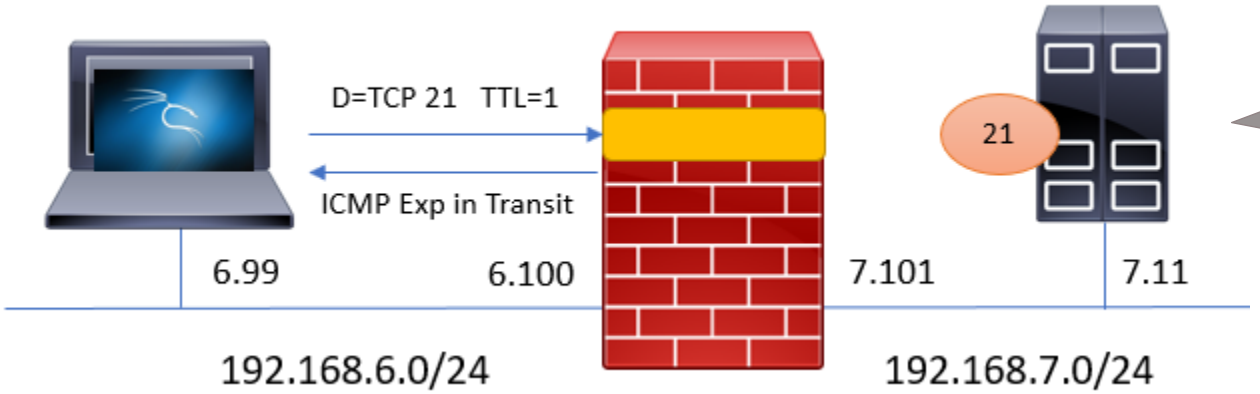
- Adding service versioning to a UDP scan helps disambiguate the responses

```
krad# nmap -sUV -T4 scanme.nmap.org

Starting Nmap ( http://nmap.org )
Nmap scan report for scanme.nmap.org (64.13.134.52)
Not shown: 999 open|filtered ports
PORT   STATE SERVICE VERSION
53/udp open  domain  ISC BIND 9.3.4

Nmap done: 1 IP address (1 host up) scanned in 3691.89 seconds
```

FIREWALKING

- Used to determine exactly which device permits/blocks a port
- Good for probing past a firewall into an internal network
- Can be used to probe past multiple daisy-chained firewalls
- You can manipulate the IP TTL in a scan to distinguish between:
  - A protected server that does not have that port open
  - An intermediate firewall that blocks the port from being reached by the scanner
- A firewall will return ICMP Type 11, Code 0 (Time Exceeded) if:
  - The port is allowed
  - The probe TTL expires at the firewall
    - Probe must be sent to a live final target
    - Does not matter if the final target actually listens on that port
  - Nmap itself will report the port status as filtered
    - It's looking for a TCP response, not an ICMP response
  - A firewalker will notice the ICMP response and report the port as permitted

FIREWALKING EXAMPLE



Packet expires at firewall

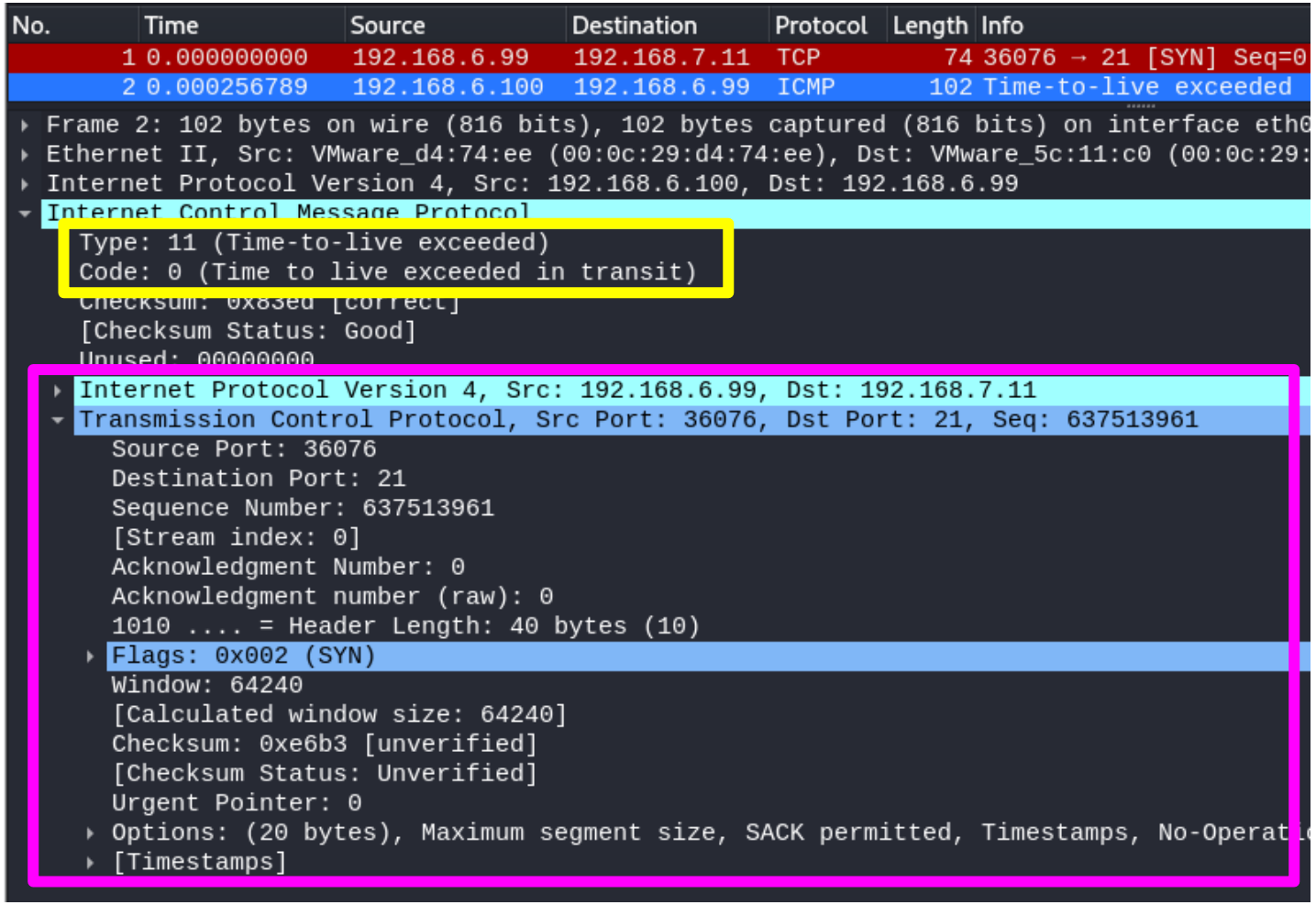
```
┌──(root💀kali)-[~]
└─# nmap -Pn -p21 -n --ttl 1 -sT 192.168.7.11
Host discovery disabled (-Pn). All addresses will be marked 'up' a
PORT   STATE    SERVICE
21/tcp filtered ftp
```

Nmap expects TCP response - reports false negative

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.6.99	192.168.7.11	TCP	74	36076 → 21 [SYN] Seq=0
2	0.000256789	192.168.6.100	192.168.6.99	ICMP	102	Time-to-live exceeded

Firewall actually returns ICMP Type 11, Code 0 TTL Exceeded message

FIREWALKING EXAMPLE (CONT'D)



ICMP Error includes original packet as payload

AUTOMATED FIREWALKING

- Linux tools such as firewalk and Nmap firewalk script attempt to automate the firewalking process
- Steps:
  1. Use a TCP-based traceroute to first establish the number of hops to the target firewall you are probing ("ramp up" period)
  2. Send a probe with the TTL that expires at the firewall to see if the firewall will:
     - Return an ICMP TTL Exceeded error (port allowed)
     - Send no response (port disallowed)
  3. Send a probe with the TTL + 1 to see if there is another filtering router/firewall behind it
  4. Continue incrementing the TTL by 1 until all firewalls in the path are tested and:
     - Either the max hop count is reached
     - Or a server actually responds to the probe
- Because the tool is searching for ICMP errors, it is not necessary to actually reach the protected host server
- If there are multiple packet filtering routers/firewalls in the path, they are all tested

Note: These tools assume there are multiple hops between the attacker and the target firewall.
They may not work as expected in all scenarios.

LINUX FIREWALK TOOL EXAMPLE

```
root@kali:~# firewalk -S8079-8081  -i eth0 -n -pTCP 192.168.1.1 192.168.0.1
Firewalk 5.0 [gateway ACL scanner]
Firewalk state initialization completed successfully.
TCP-based scan.
Ramping phase source port: 53, destination port: 33434
Hotfoot through 192.168.1.1 using 192.168.0.1 as a metric.
Ramping Phase:
 1 (TTL  1): expired [192.168.1.1]
Binding host reached.
Scan bound at 2 hops.
Scanning Phase:
port 8079: *no response*
port 8080: A! open (port not listen) [192.168.0.1]
port 8081: *no response*

Scan completed successfully.

Total packets sent:                4
Total packet errors:               0
Total packets caught               2
Total packets caught of interest   2
Total ports scanned                3
Total ports open:                  1
Total ports unknown:               0
```

- Scan TCP ports 8079 – 8080
- Send probes out eth0
- No name resolution
- Firewall (target) is 192.168.1.1
- Server (metric) is 192.168.0.1

- Port 8079 disallowed
- Port 8080 allowed
  - Server does not use 8080
- Port 8081 disallowed

NMAP FIREWALK SCRIPT EXAMPLE

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap --script=firewalk --traceroute scanme.nmap.org
Starting Nmap 7.91 ( https://nmap.org ) at 2022-01-27 10:51 EST
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.087s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:
bb2f
Not shown: 992 closed ports
PORT      STATE    SERVICE
22/tcp    open     ssh
25/tcp    filtered smtp
80/tcp    open     http
135/tcp   filtered msrpc
139/tcp   filtered netbios-ssn
445/tcp   filtered microsoft-ds
9929/tcp  open     nping-echo
31337/tcp open     Elite

Host script results:
| firewalk:
| HOP  HOST       PROTOCOL   BLOCKED PORTS
|_1    10.0.0.1   tcp        25,135,139,445
```

Do not specify a firewall. Just specify the server that hosts the services

Simple nmap scan of the server returns these results

This firewall is doing the filtering

```
TRACEROUTE (using port 995/tcp)
HOP RTT      ADDRESS
1   5.04 ms  10.0.0.1
2   15.26 ms 96.120.104.25
3   25.57 ms 24.124.181.165
4   16.36 ms 68.86.204.101
5   20.96 ms 96.216.131.253
6   18.78 ms be-31411-cs01.ashburn.va.ibone.comcast.net (96.110.40.17)
7   16.52 ms be-2311-pe11.ashburn.va.ibone.comcast.net (96.110.32.130)
8   19.08 ms ix-ae-21-0.tcore3.aeq-ashburn.as6453.net (216.6.87.125)
9   ...
10  77.75 ms if-ae-13-2.tcore1.sqn-sanjose.as6453.net (63.243.205.65)
11  85.39 ms 216.6.33.114
12  76.75 ms if-2-4.csw6-fnc1.linode.com (173.230.159.87)
13  82.63 ms scanme.nmap.org (45.33.32.156)

Nmap done: 1 IP address (1 host up) scanned in 54.66 seconds
```

The filtering firewall in this case is the first hop to the destination

NMAP FIREWALK SCRIPT EXAMPLE #2

- Hops 2, 6, and 7 are all filtered

```
| firewalk:
| HOP HOST           PROTOCOL   BLOCKED PORTS
| 2   192.168.1.1    tcp        21-23,80
|                    udp        21-23,80
| 6   10.0.1.1       tcp        67-68
| 7   10.0.1.254     tcp        25
|_                   udp        25
```

3.8 PROXIES

- Proxies
- Anonymizers
- VPNs
- TOR

PROXY SERVERS

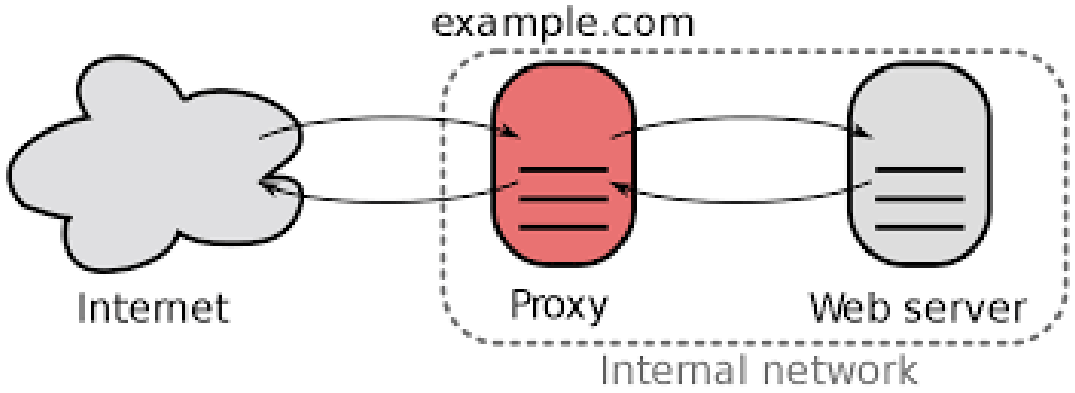
- A proxy server is an intermediary between:
  - Internal user and Internet resource
  - Internet user and internal resource
- Use an online proxy to:
  - Hide source IP address to avoid discovery
    - Increase privacy
    - Conduct anonymous hacking attacks
    - Mask the source of an attack by impersonating a false source
  - Remotely access intranets and website resources that are normally protected
  - Interrupt all requests sent by a user and re-route them to a different destination, making it see only the proxy server address
  - Chain multiple proxy servers to avoid detection

PROXY APPROACHES

- Edge Proxy
- Anonymizer
- Proxy Chaining
- VPN

EDGE PROXY

- Typically used by private organizations to protect their internal network
- Forward proxy:
  - An edge firewall or separate proxy server fetches content from the Internet on behalf of internal clients
- Reverse proxy:
  - An edge firewall fetches content from the private network/DMZ on behalf of Internet clients



ANONYMIZER

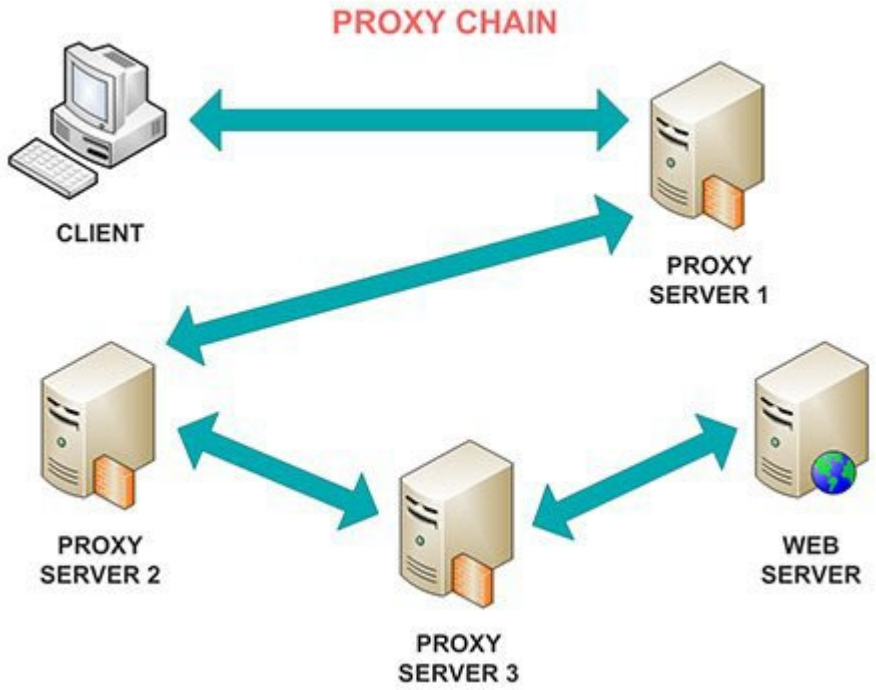
- A proxy server on the Internet
- Created specifically so people can hide their connection's true origin
- May be free or a paid commercial service
  1. The user connects across the Internet to the proxy server
  2. The proxy puts the user "on hold"
  3. The proxy starts a separate connection to fetch the desired content for the user
  4. The proxy hands the content to the user
- Since the proxy is the one actually fetching the content:
  - The requesting IP address is different
  - No one knows that the request is actually coming from the user
- Because IP addresses are country/region specific, they can be blocked or tracked
- An anonymizer located in a different country, using its own IP, will not be blocked
  - It can be tracked, but only to the proxy, not to the end user

ANONYMIZER USE CASES

- People use anonymizers to:
  - Surf anonymously
  - Hack anonymously
  - Bypass censorship or regional firewalls
  - Evade surveillance or restrictions based on their IP
- Anonymizers make it difficult for others to track you:
  - Most anonymizers do not keep logs of their activity or client connections
  - Anonymizers are typically in different legal jurisdictions
    - They're not compelled to cooperate with your country's law enforcement

PROXY CHAINING

- The use of multiple proxies, in series, to fulfill a request
  1. The client connects to a proxy and makes a request
  2. That proxy makes a connection to another proxy making the same request
  3. That proxy might connect to a third proxy, etc.
  4. This is repeated through as many proxies as desired
  5. At the end, the last proxy fetches the requested content
  6. The requested content is passed back through the entire chain, ultimately given to the client
- You can use as many proxy servers as you want
- The more proxies you use, the harder it is to trace the activity back to you



PROXY TOOLS

- Proxy Switcher
  - Hides your IP address from the website you visit
- Proxy Workbench
  - A proxy server that displays data passing through it in real time
  - You can examine TCP/IP connections, view history, save to a file, view a socket connection diagram
- Tor
  - Routing through the deep web for privacy protection, defense against network surveillance/traffic analysis
- CyberGhost
  - Anonymous browsing and access to blocked/censored content
  - Replaces the user's original IP with an address of their choice

ADDITIONAL PROXY TOOLS

Test web apps by capturing and manipulating your browser's interaction with the server:

- Burp Suite
- Charles
- Fiddler

These tools run on your own computer

Proxy clients:

- Proxifier
- SocksChain

For Mobile Devices

Online VPN/Proxy service:

- Shadowsocks
- CyberGhost VPN
- Hotspot Shield
- NetShade

Client app to manage your various proxy/VPN connections:

- Proxy Manager

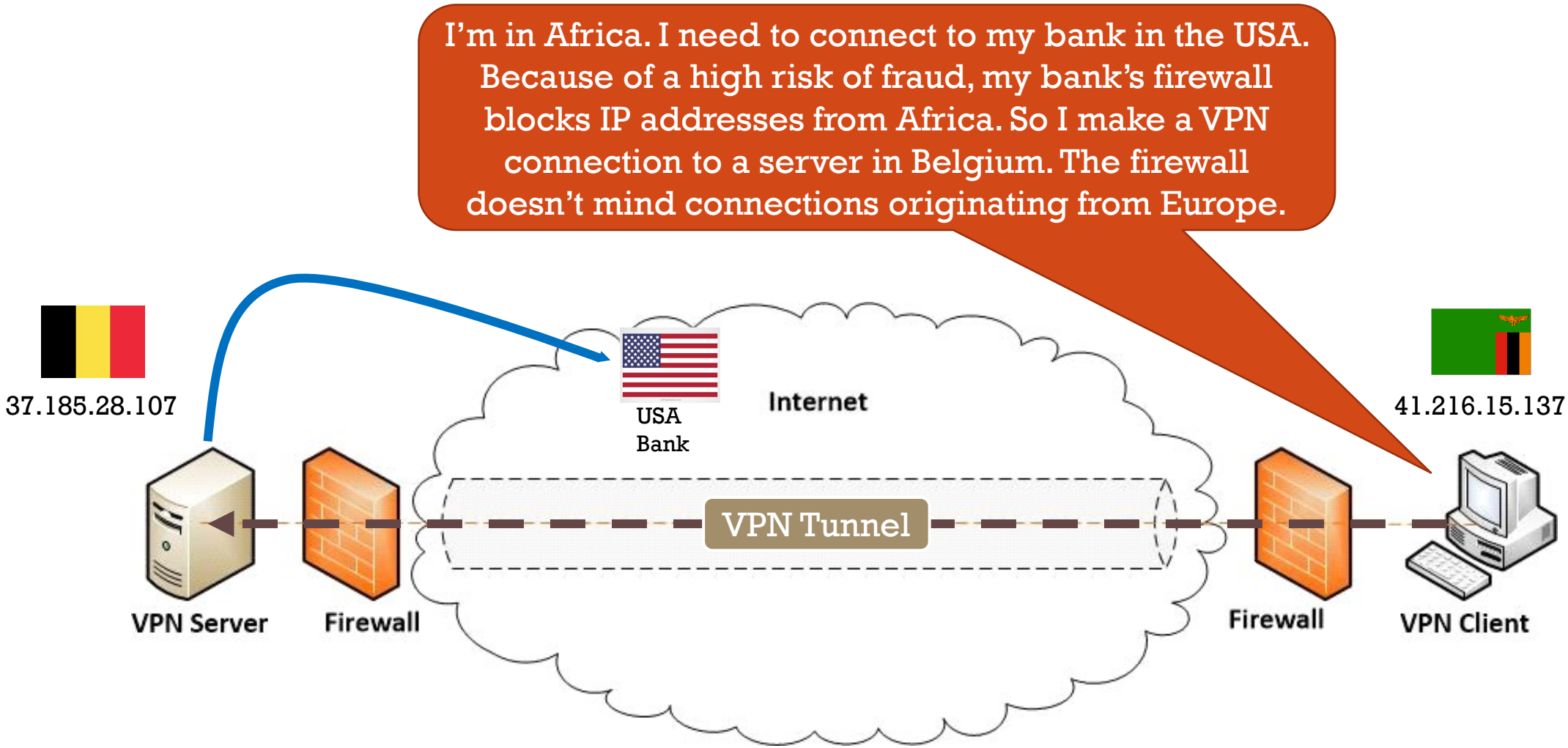
Create a proxy (and other services) on your mobile device:

- Servers Ultimate

VIRTUAL PRIVATE NETWORK (VPN)

- Your original packets are encrypted and encapsulated (hidden) inside other packets
- A client app on your computer/phone starts the VPN
  - Your traffic is encapsulated and encrypted from the very start
- You send your traffic to a VPN server on the Internet, in some other part of the world
- That server then:
  - discards the outer packaging
  - decrypts your original packets
  - gives your packets a source IP address from its own network
  - sends your unencrypted packets out on the Internet to their final destination
- Your traffic looks like it originated from where the VPN server is
  - Not where you actually are

VPN EXAMPLE


I'm in Africa. I need to connect to my bank in the USA. Because of a high risk of fraud, my bank's firewall blocks IP addresses from Africa. So I make a VPN connection to a server in Belgium. The firewall doesn't mind connections originating from Europe.
37.185.28.107
USA Bank
Internet
41.216.15.137
VPN Tunnel
VPN Server
Firewall
Firewall
VPN Client

VPN COMPONENTS

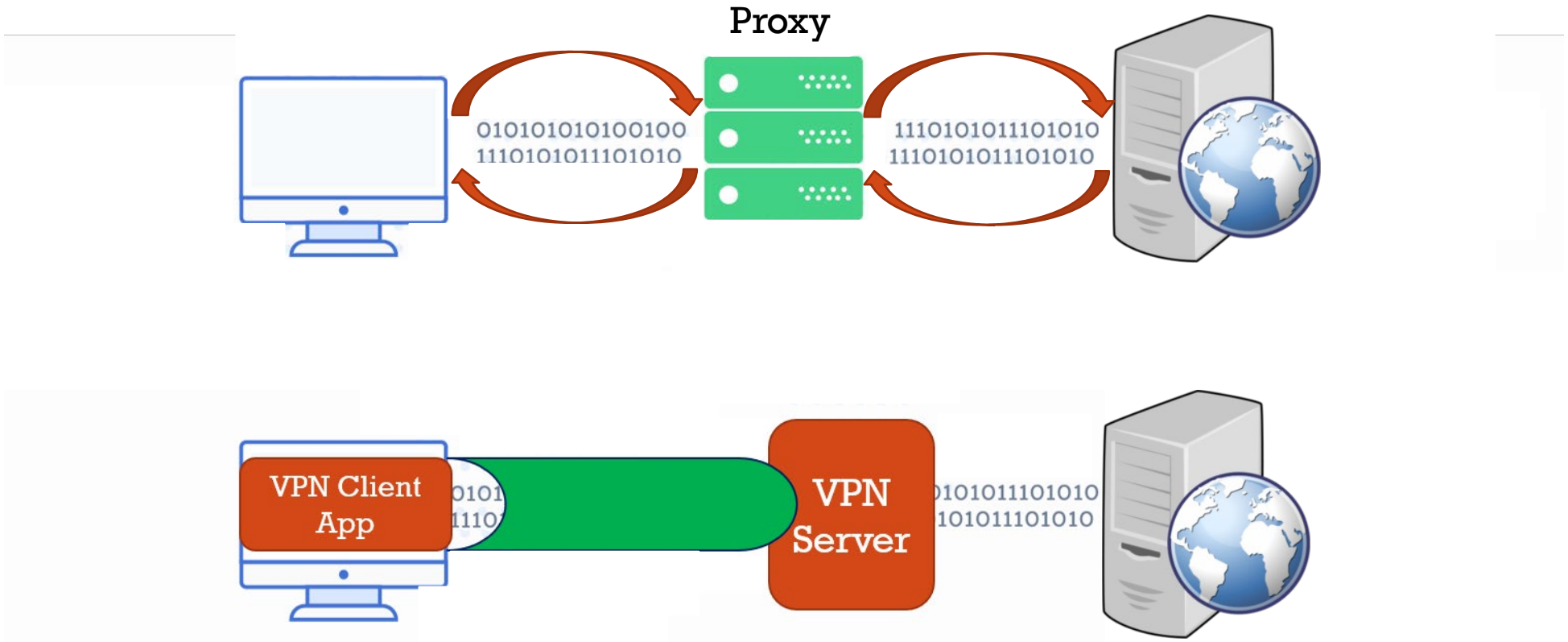
- VPN client app running on your computer/phone
- VPN server of your choosing (somewhere on the Internet)
- VPN protocols to encapsulate and encrypt your data
- Common protocols today:
  - IPSEC
  - SSTP
    - Secure Socket Tunneling Protocol
    - HTTP/TLS
  - OpenVPN
    - TLS-encrypted payload over TCP or UDP

- Legacy protocols:
  - L2TP
    - Encapsulation + Encapsulating Security Payload (ESP)
  - PPTP
    - Point-to-Point Tunneling Protocol
    - Generic Routing Encapsulation (GRE) + Point-to-Point Protocol (PPP)

WHAT'S THE DIFFERENCE BETWEEN A PROXY AND A VPN?

- They functionally accomplish the same thing
  - Both are used to hide your true origin
- The mechanisms however are quite different
- In both cases you secretly connect to a server on the Internet
  - A proxy fetches content on your behalf while you "wait at home"
  - The VPN server decrypts your original traffic and sends it unencrypted to its final destination
- Traditionally, proxies did not use encryption
  - Your connection to the proxy, and the proxy's connection on your behalf, were unencrypted
- Today, however, most anonymizers use VPNs
  - VPN from you to the first proxy
  - VPN between proxies
  - Clear unencrypted connection from the last proxy to the web (resource) server

PROXY VS VPN


Proxy
0101010101001001111010101111010101111010101111010101
VPN Client App
VPN Server

ANONYMIZERS AND ONLINE VPNS

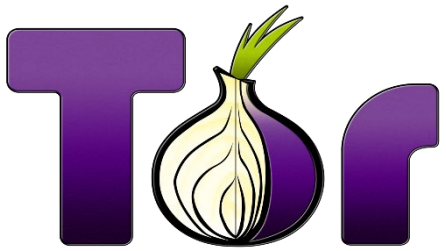
- UltraVPN
- TunnelBear
- TotalVPN
- Hotspot Shield
- NordVPN
- ExpressVPN
- CyberGhost
- IPVanish
- SaferVPN
- PrivateVPN
- Surfshark
- Norton
- ZenMate
- ProtonVPN

Additional privacy recommendations:

- Increase privacy settings on your browser including private/incognito browsing
- Clear cookies and history on your browser
- Use a search engine such as DuckDuckGo that does not track your history

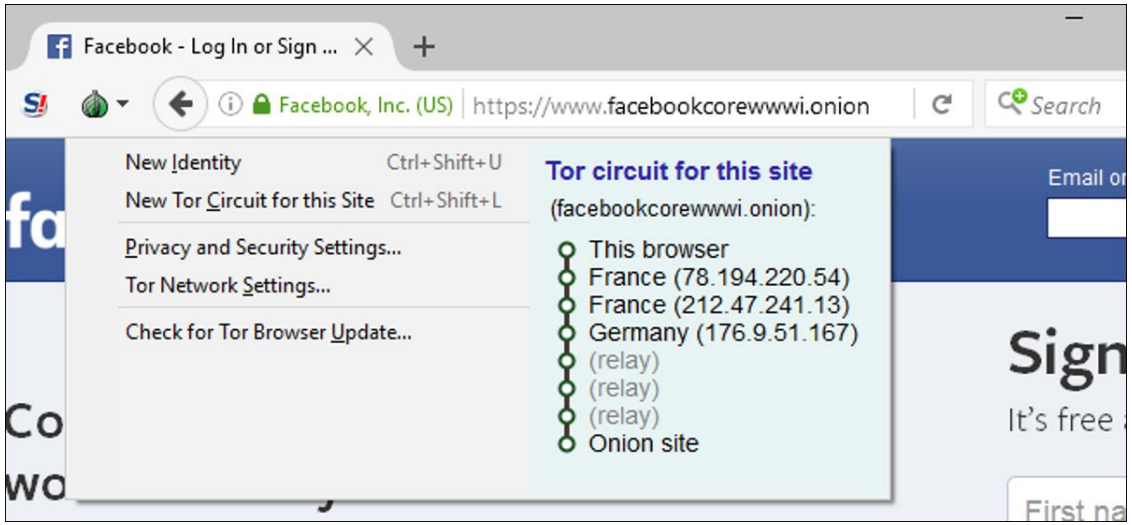
TOR

- AKA The Onion Router
- A free and open-source software for enabling anonymous communication
- Directs Internet traffic through a worldwide overlay network
  - Over 6000 relays
  - Conceals a user's location and usage from network surveillance and traffic analysis
  - Your route changes every 10 minutes
- Makes it more difficult to trace Internet activity to the user
- Intended use is to protect personal privacy
  - Unfortunately has also become home to "dark web" criminal activity
- A TOR browser aims to make all users look the same
  - Making it difficult to fingerprint you based on your browser or device
  - Easy to set up and use
  - Download a TOR browser from https://www.torproject.org/download/

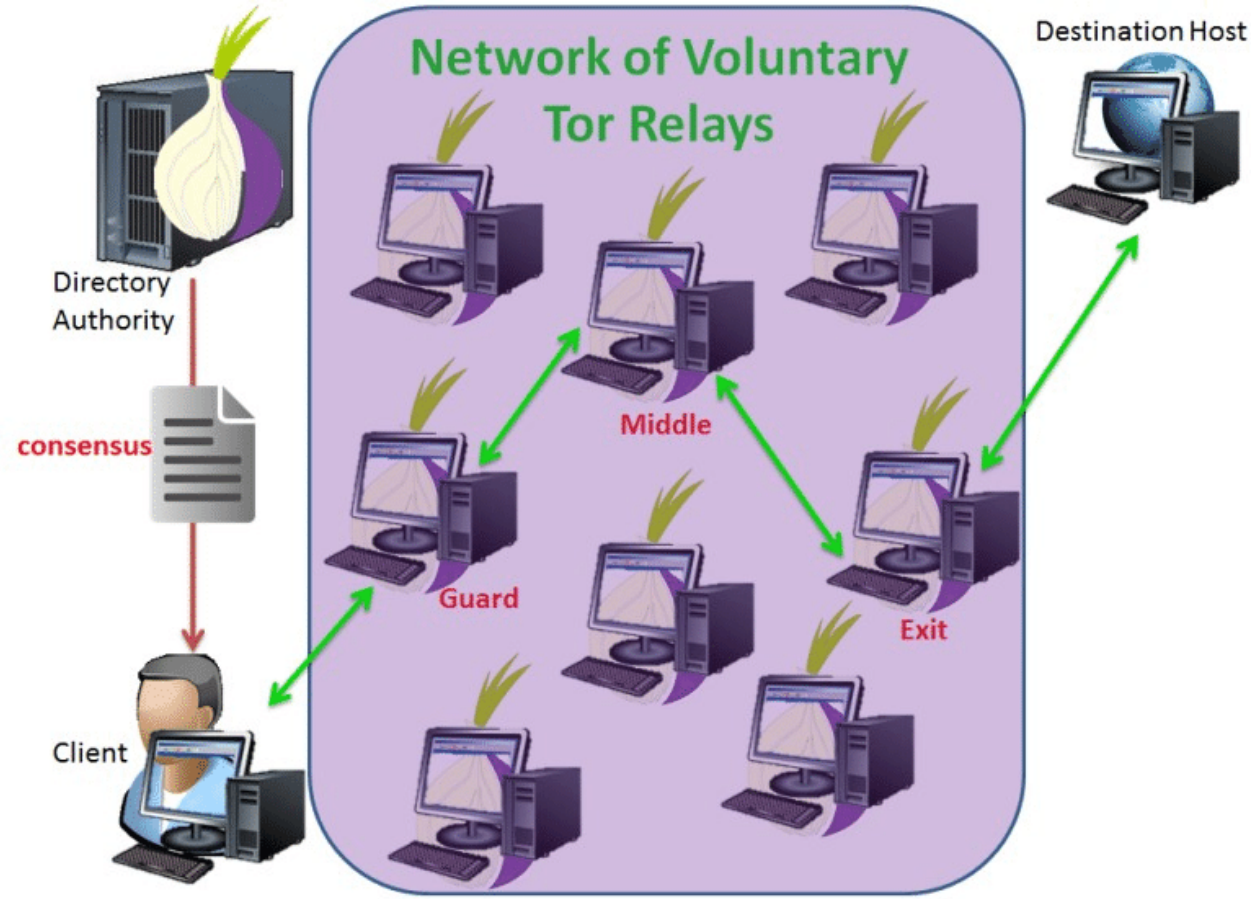


ONION SITES

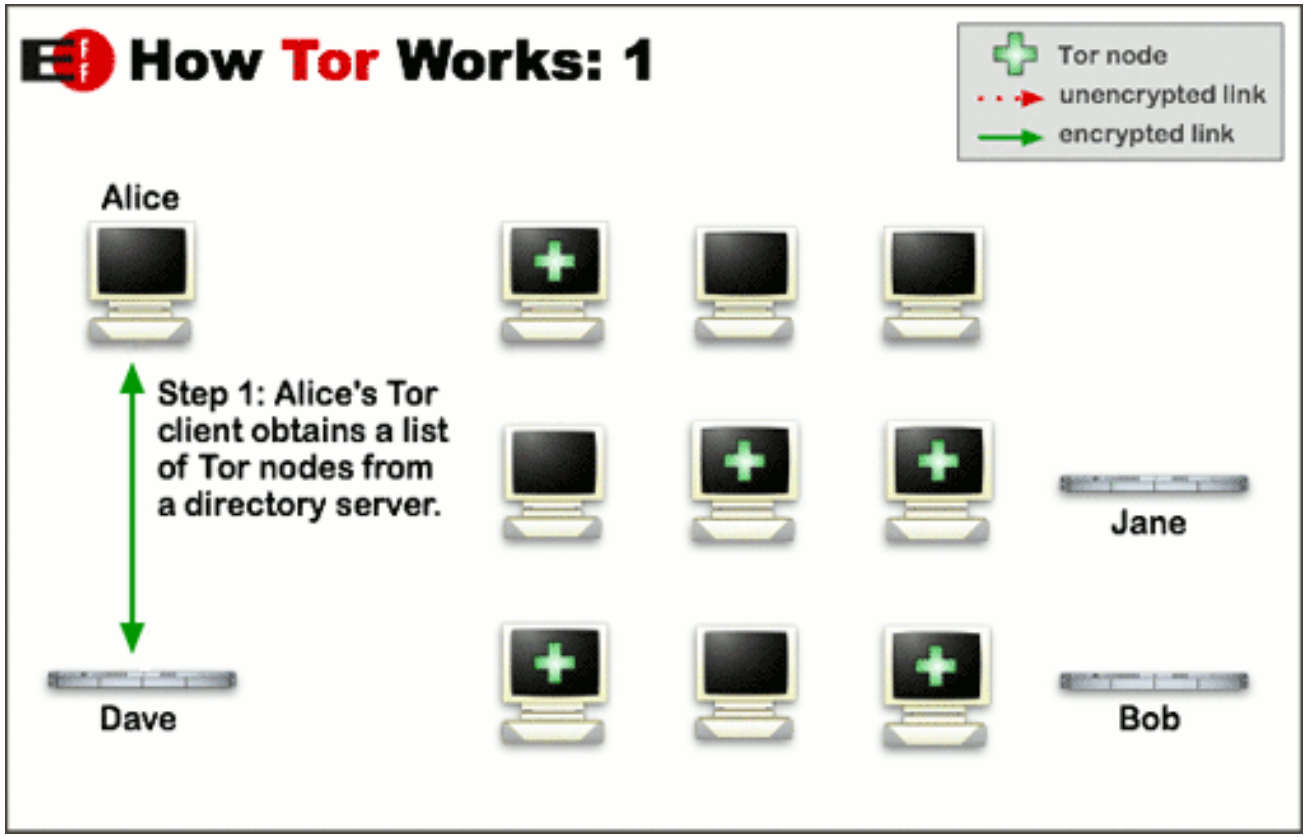
- Website addresses that end in ".onion"
- Not like normal domain names
- You can't access them with a normal web browser
- Addresses that end with ".onion" point to Tor hidden services on the "deep web"



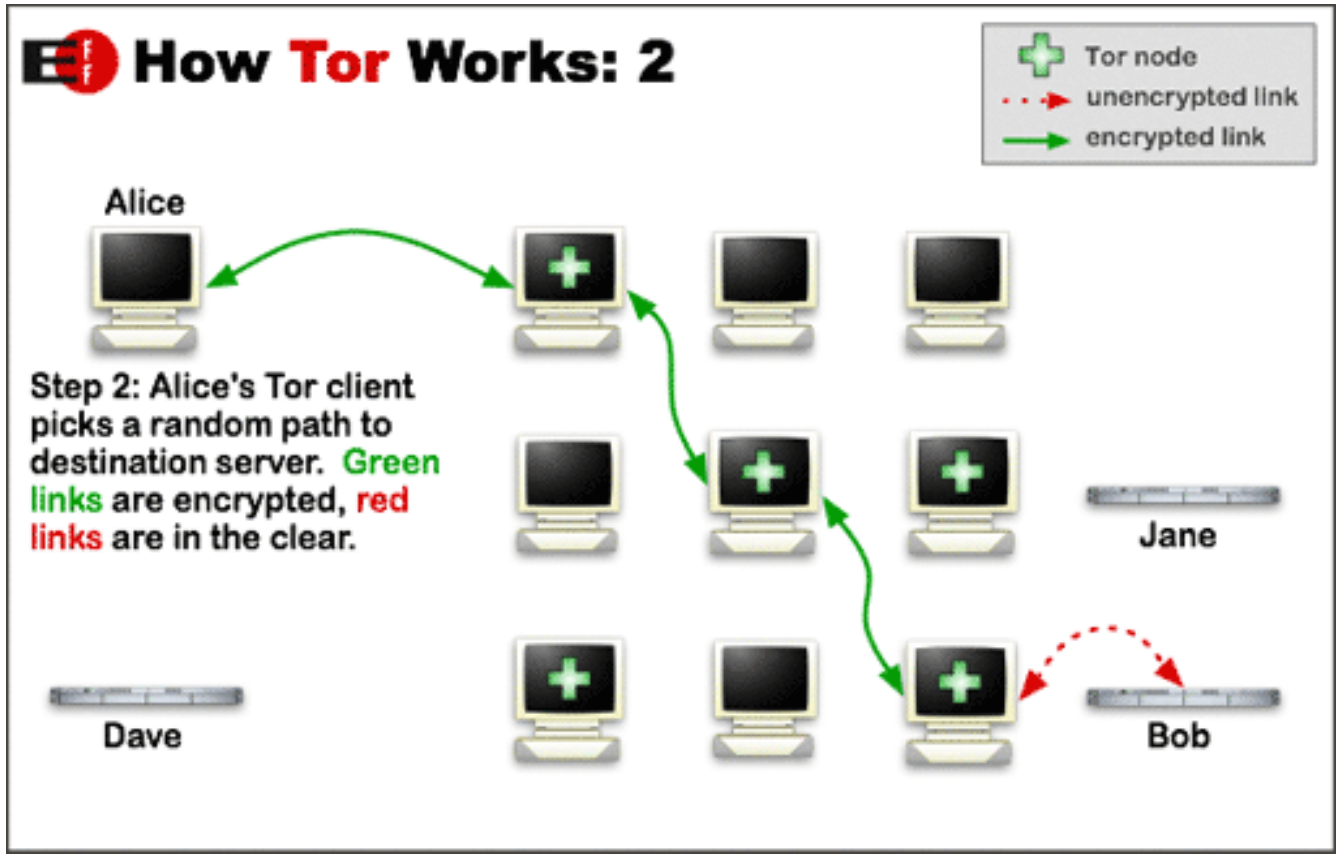
TOR ARCHITECTURE



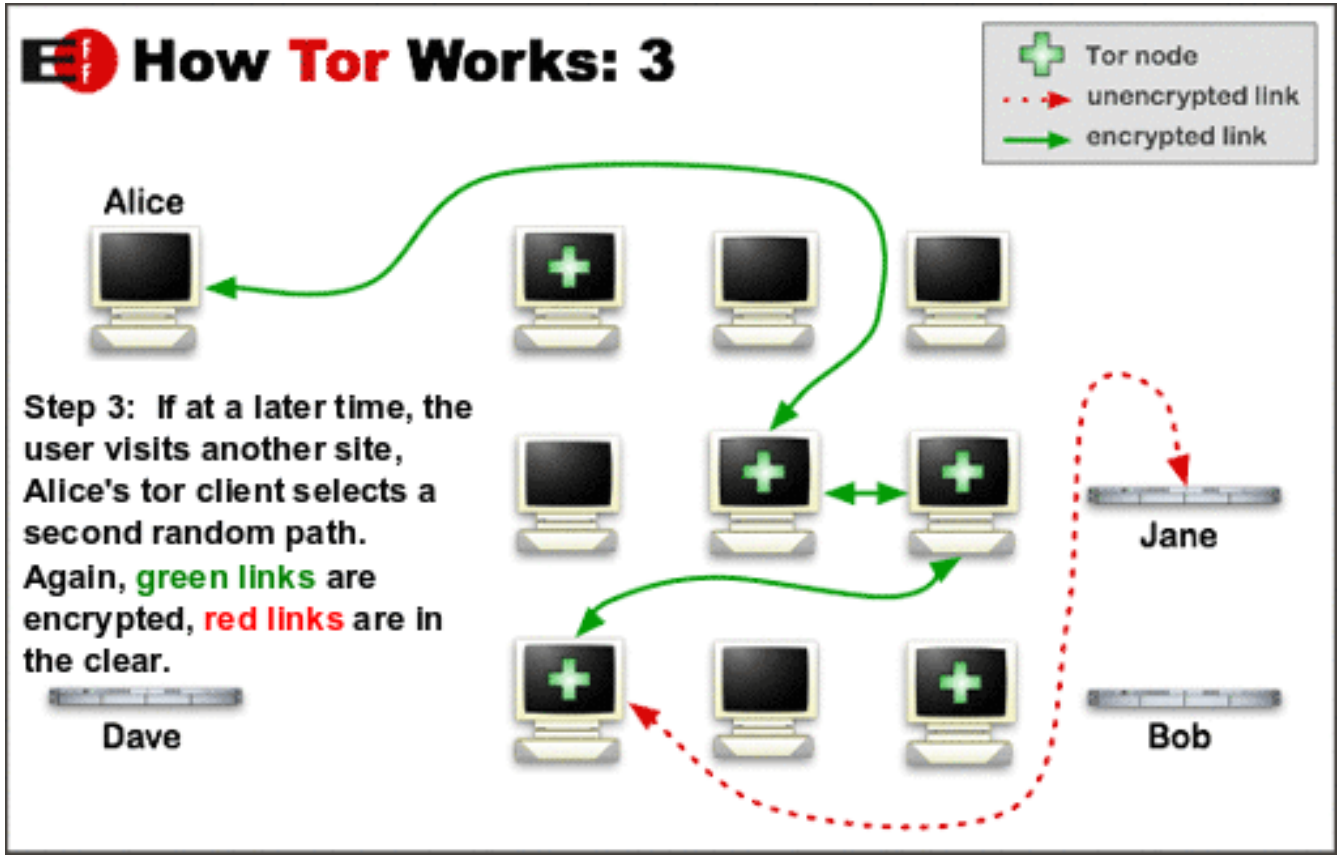
TOR EXAMPLE


How Tor Works: 1
Tor node
unencrypted link
encrypted link
Alice
Step 1: Alice's Tor client obtains a list of Tor nodes from a directory server.
Jane
Dave
Bob

TOR EXAMPLE (CONT'D)



TOR EXAMPLE (CONT'D)



3.9 SCANNING COUNTER-MEASURES

- Scanning Countermeasures
- Spoofing Countermeasures
- Banner Grabbing Countermeasures
- Firewall Bypass Countermeasures

PORT SCANNING COUNTERMEASURES

- Implement a software firewall on all devices
- Limit the number of open ports
- Block ICMP
- Configure routers to disallow vulnerable features such as source routing and IP fragments
- Use an IDS/IPS to monitor network traffic

PORT SCANNING COUNTERMEASURES (CONT'D)

- Patch hosts
- Conduct your own scans pre-emptively
- Ensure that the IDS, routers, and firewall firmware are updated to their latest releases
- Consider using a cloud-based SIEM to leverage more sophisticated/longer term traffic analysis
- In a high-security environment, consider hard-coding MAC-to-IP address mappings for each host

SPOOFING COUNTERMEASURES

- Do not rely on IP-based authentication
- Digitally sign all transmissions
- Use stateful firewalls with deep packet inspection
- Disallow source routing
- Disallow incoming packets that appear to come from your own network
  - Spoofed source IP

SPOOFING COUNTERMEASURES (CONT'D)

- Be cautious when allowing traffic based on source port
- Hard-code ARP entries where practical
- Hard-code IP addresses where practical
- Use switchport security
- Secure DNS server cache against pollution

BANNER GRABBING COUNTERMEASURES

- Disable or change the banner
  - Display false/misleading banners
  - Make sure banner does not advertise the service version
  - Add an "authorized users only" warning to a banner to protect yourself legally
    - Especially for services that require a user to log on
- Turn off unnecessary services
- Hide file extensions from web pages such as .asp or .htm
  - IIS can use tools like PageXchanger to manage file extensions
  - Apache can edit httpd.conf with mod_negotiation directives

FIREWALL BYPASS COUNTERMEASURES

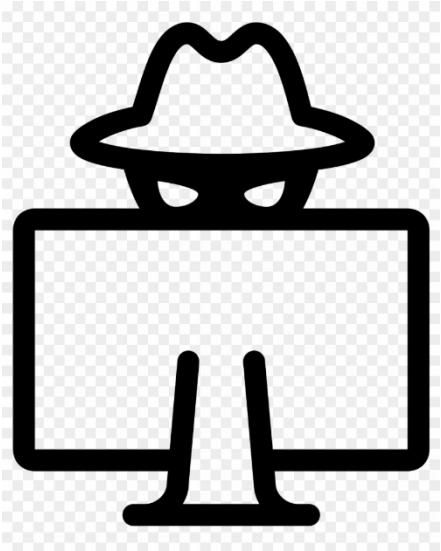
- Use a multilayer defense strategy
- Implement multiple firewall solutions at different levels
- Implement strong change management
- Stay on top of security patches/updates
- Set strong password policies and multifactor authentication
- Look for "side doors" and "back doors" that can bypass the firewall
  - Wi-Fi access points
  - VPN / Remote Access servers
  - Private WAN links / VPNs to other company sites
  - "Sneakernet" (physically moving data in and out of the network on removeable media)
- Perform your own firewall tests to ensure rules behave as desired
- Regularly perform penetration tests

3.10 SCANNING NETWORKS REVIEW

- Review

SCANNING NETWORKS REVIEW

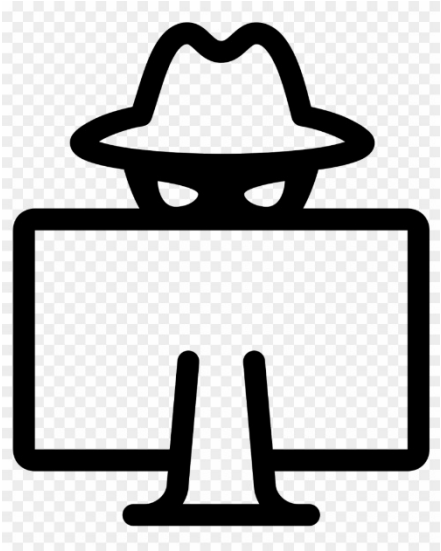
- Scanning is part of active reconnaissance
- Scanning discovers possible targets on a network:
  - Live hosts
  - Open ports
  - Protocols
  - Service and operating system versions
  - Can include banner grabbing



- Ping sweeps previously used ICMP echo requests to discover hosts
- Modern ping sweeps use ARP, TCP, or some other protocol for host discovery
- ICMP has numerous message types, which in turn may have codes

SCANNING NETWORKS REVIEW

- A port represents a process on the network
- Both TCP and UDP use ports
- Client and server processes each use their own port (typically not the same)
- Server services listen on well-known ports 1-1023
- Services may request additional registered ports (1024-49151) from their operating system



- Clients borrow dynamic ports (41952-65535) from their operating system
- A client port is returned to the OS when that client process is terminated

SCANNING NETWORKS REVIEW

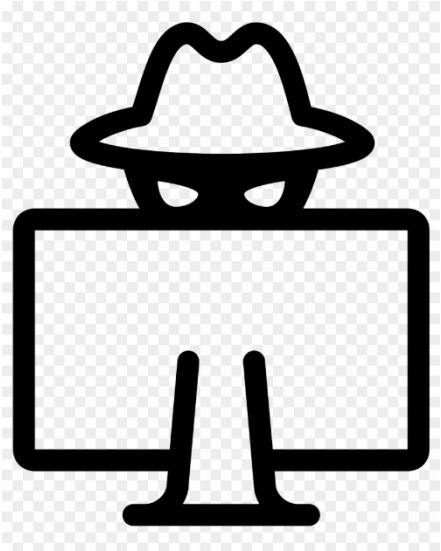
- Common server ports include:

21	FTP commands
22	SSH
23	Telnet
25	SMTP
53 (TCP or UDP)	DNS
80	HTTP
88	Kerberos
110	POP3
111 (TCP or UDP)	*nix portmapper

135	Microsoft Remote Procedure Call (RPC)
139	SMB (legacy)
143	IMAP4
161 (UDP)	SNMP
162 (UDP)	SNMP traps
389	LDAP
443	HTTPS
445	Microsoft-ds
3389	RDP

SCANNING NETWORKS REVIEW (CONT'D)

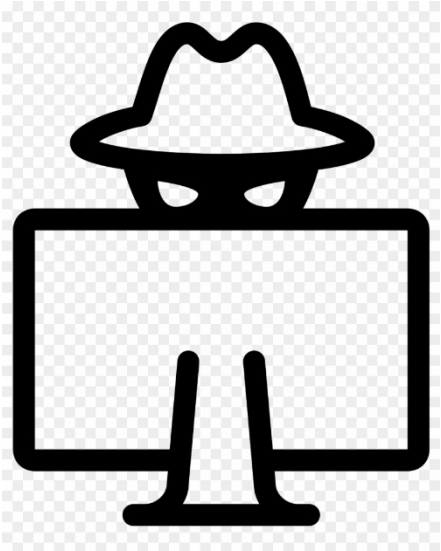
- TCP uses a three-way handshake to establish sequence numbers and start a session
  - SYN, SYN-ACK, ACK
- TCP uses a four-way handshake to end a session
  - FIN, ACK, FIN, ACK
- A TCP SYN scan (aka stealth or half-open scan) does not complete the handshake
- A TCP Connect scan (aka full or open scan) does complete the handshake



- You can also scan raising various TCP flags to test a firewall
- You can perform a UDP port scan, but no handshake is involved, and you might not receive any response from the target

SCANNING NETWORKS REVIEW (CONT'D)

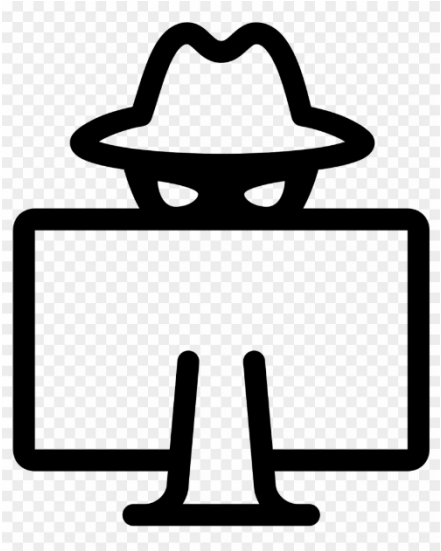
- Port scanning is the immediate prelude to vulnerability testing
  - Some scanning tools perform discovery, port scanning, and vulnerability testing all in one comprehensive scan
- Packet crafting manipulates TCP/UDP/IP headers to:
  - Probe open ports
  - Test firewalls / IDS



- Anonymizers / proxies hide the source of a packet
- You can use a VPN to encrypt your connection to a proxy
- Creating a network diagram gives you an overview of the entire target network
  - Can be useful in planning your attack

SCANNING NETWORKS REVIEW (CONT'D)

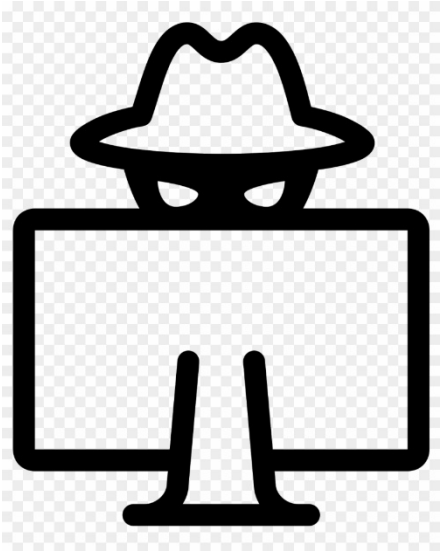
- You can fingerprint an OS by examining its TCP or IP headers
- You can banner grab to capture information about a network service and the OS it resides on
- A list scan only performs DNS lookups, and does not actually scan the target
- A zombie (idle) scan uses an intermediary machine to interact with the target
- An FTP bounce scan uses a vulnerable FTP server to perform a scan against the real target



- You can perform SSDP scans to identify vulnerable home and small office networks
- Nmap is the primary scanning tool used by hackers
- Zenmap is a GUI version of nmap for Windows
- Hping can also perform scans and packet crafting

SCANNING NETWORKS REVIEW (CONT'D)

- You can use a number of tactics to evade firewall and IDS detection when scanning:
  - Source routing
  - Fragmentation
  - Source port manipulation
  - Decoys
  - Address spoofing
  - Slow timing



- Raising various TCP flags such as ACK, NULL, FIN, and PSH/URG/FIN
- Firewalking is the process of identifying which ports network firewalls will allow traffic through