# Introduction to Security Tools (Nmap & Wireshark)

**Name : Syed Azharuddin Ali Khan**
**Role : Cybersecurity Analyst Intern**
**Date : 02/07/2026**

### Objective

To introduce foundational cybersecurity tools used for **network reconnaissance** and **traffic analysis**, enabling the intern to gain their **first hands-on exposure** to real-world security tooling.

---

### Tools Covered

- **Nmap**
- **Wireshark**

---

## 1: Introduction to Security Tools

### 1.1 Why Security Tools Matter

Cybersecurity is not only theoretical—it is **tool-driven**. Security professionals rely on specialized tools to:

- Discover network assets
- Identify vulnerabilities
- Monitor suspicious activity
- Investigate security incidents

Without tools like Nmap and Wireshark, **visibility into a network is impossible**.

---

### 1.2 Categories of Security Tools

| Category | Example | Purpose |
|---|---|---|
| Network Scanning | Nmap | Discover systems & services |
| Packet Analysis | Wireshark | Inspect network traffic |
| Vulnerability Scanning | Nessus | Find known weaknesses |
| Exploitation | Metasploit | Test exploitability |
| Monitoring | SIEM tools | Detect threats |

This task focuses on **Network Scanning** and **Packet Analysis**, the **foundation** of all security operations.

---

## 2: Nmap – Network Mapper

### 2.1 What is Nmap?

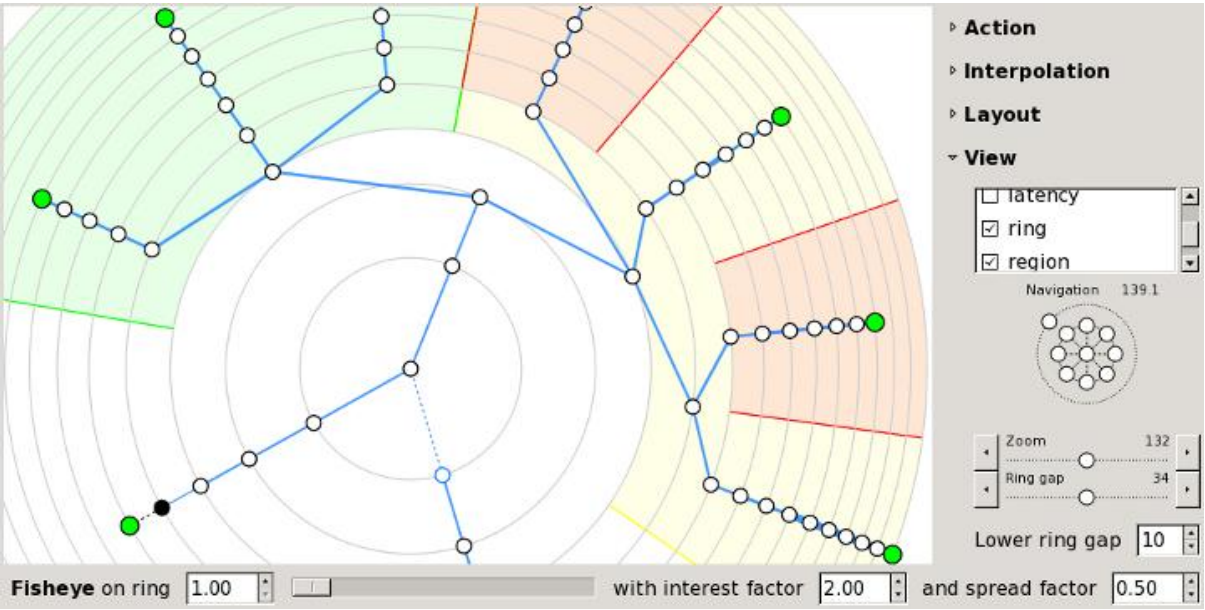
**Nmap (Network Mapper)** is an open-source tool used to:

- Discover live hosts on a network
- Identify open ports
- Detect running services and versions
- Map network topology

It is widely used by:

- **Penetration testers**
- **SOC analysts**
- **Network administrators**
- **Red team & blue team professionals**

---



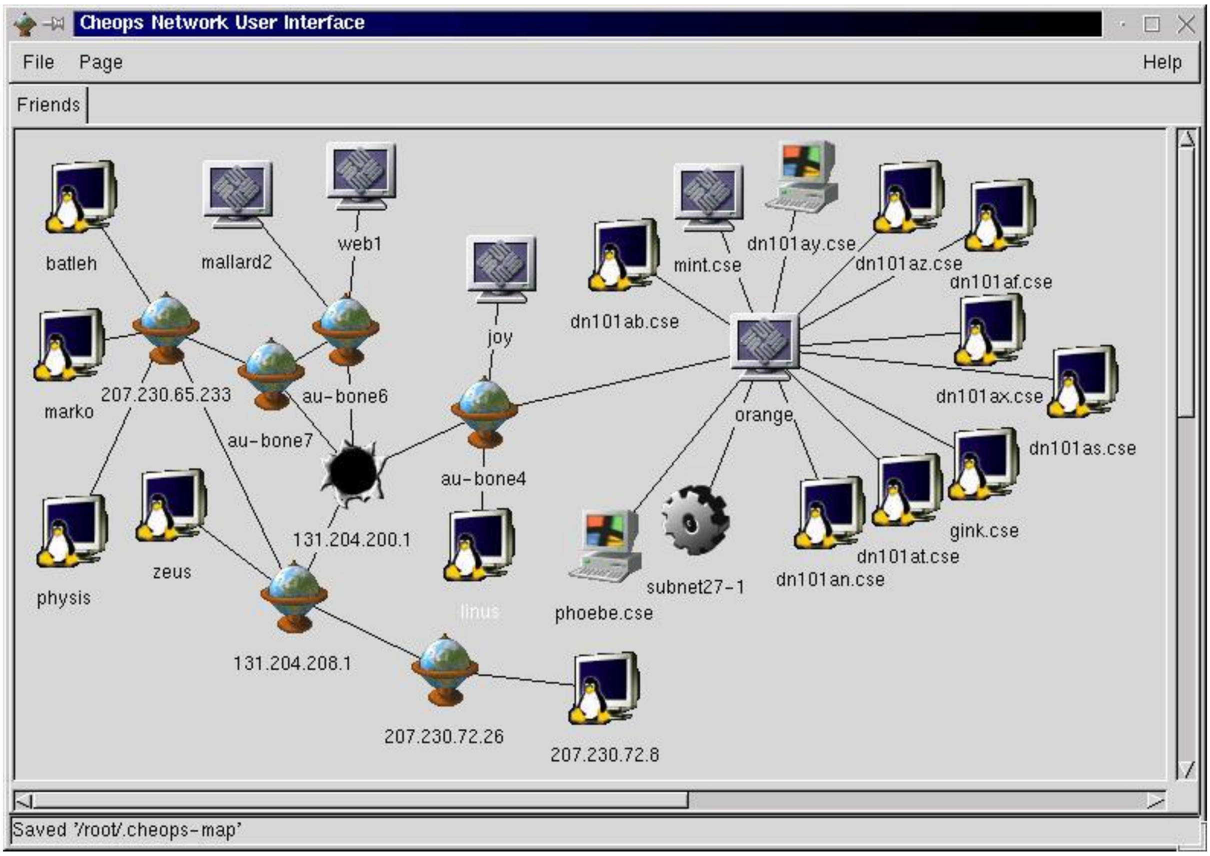
□ latency
☑ ring
☑ region

Navigation 139.1

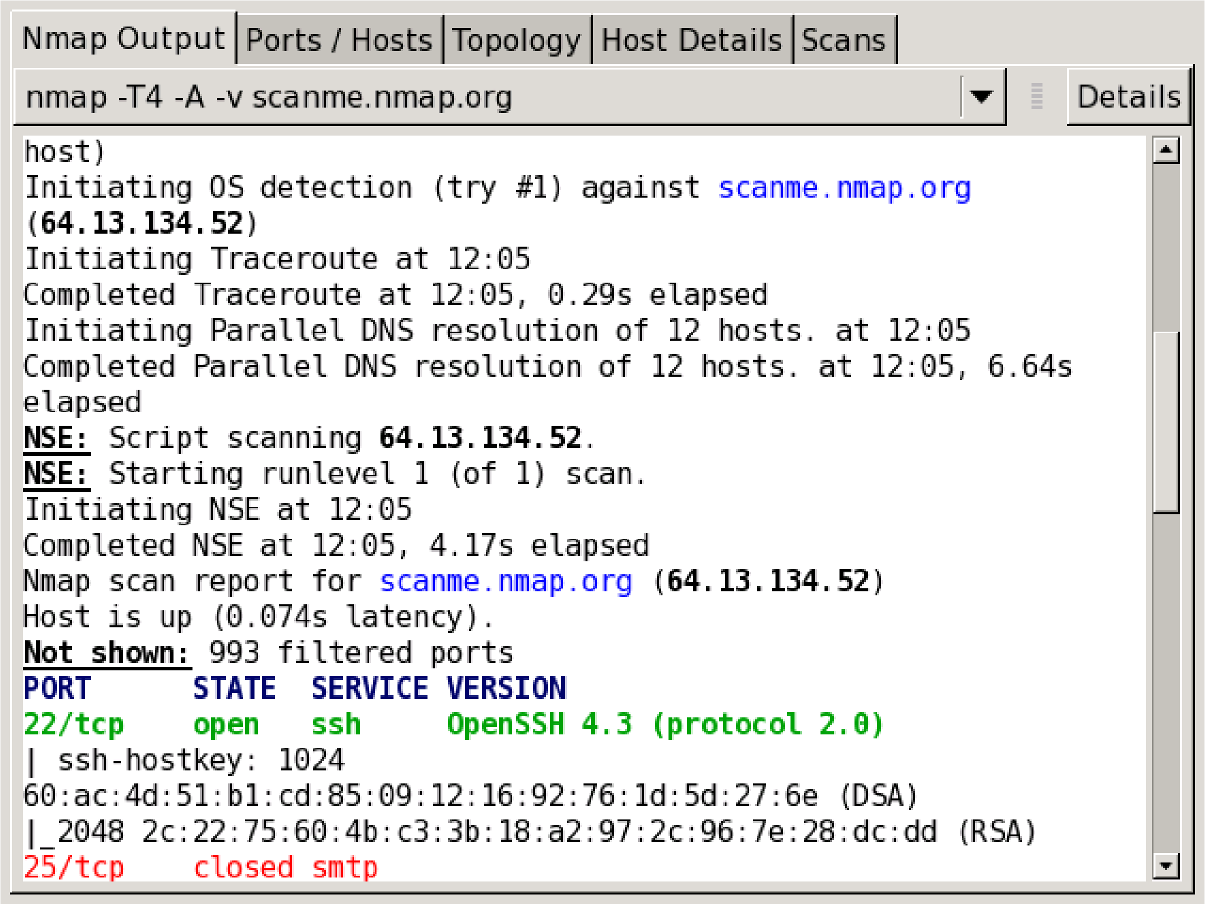
Zoom 132
Ring gap 34

Lower ring gap 10

**Fisheye** on ring 1.00 with interest factor 2.00 and spread factor 0.50



Saved '/root/.cheops-map'



```
nmap -T4 -A -v scanme.nmap.org
```

```
host)
Initiating OS detection (try #1) against scanme.nmap.org
(64.13.134.52)
Initiating Traceroute at 12:05
Completed Traceroute at 12:05, 0.29s elapsed
Initiating Parallel DNS resolution of 12 hosts. at 12:05
Completed Parallel DNS resolution of 12 hosts. at 12:05, 6.64s
elapsed
NSE: Script scanning 64.13.134.52.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 12:05
Completed NSE at 12:05, 4.17s elapsed
Nmap scan report for scanme.nmap.org (64.13.134.52)
Host is up (0.074s latency).
Not shown: 993 filtered ports
PORT      STATE   SERVICE VERSION
22/tcp    open    ssh     OpenSSH 4.3 (protocol 2.0)
| ssh-hostkey: 1024
60:ac:4d:51:b1:cd:85:09:12:16:92:76:1d:5d:27:6e (DSA)
|_2048 2c:22:75:60:4b:c3:3b:18:a2:97:2c:96:7e:28:dc:dd (RSA)
25/tcp    closed smtp
```

### 2.2 Why Nmap is Critical in Cybersecurity

Attackers always begin with **reconnaissance**.
Nmap helps defenders **think like attackers**.

Security Use Cases:

- Identifying exposed services
- Detecting misconfigured ports
- Validating firewall rules
- Incident investigation

---

### 2.3 Installing Nmap

**On Windows**

- Download from official website
- Install with default options

**On Linux (Kali/Ubuntu)**

```
sudo apt update

sudo apt install nmap
```

**On macOS**

```
brew install nmap
```

---

### 2.4 Basic Nmap Scan Commands

**Ping Scan (Host Discovery)**

```
nmap -sn 192.168.1.0/24
```

**Purpose:**
Identifies which hosts are **alive** on the network.

**Security Insight:**
Helps attackers find targets; defenders use it to inventory assets.

---

**Basic TCP Scan**

```
nmap 192.168.1.10
```

**Purpose:**
Scans the most common ports.

---

**Specific Port Scan**

```
nmap -p 22,80,443 192.168.1.10
```

**Purpose:**
Checks whether SSH, HTTP, and HTTPS are open.

---

**Service Detection**

```
nmap -sV 192.168.1.10
```

**Purpose:**
Detects service names and versions.

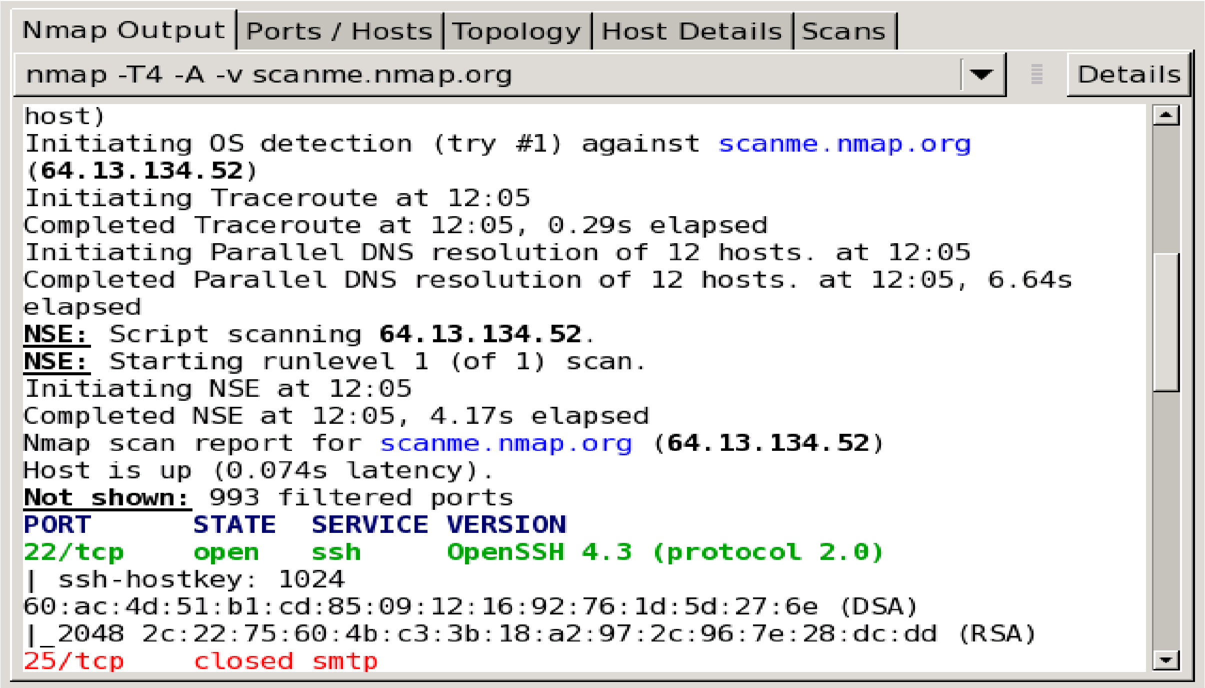
**Security Insight:**
Outdated services are a **major attack vector**.

---

### 2.5 Interpreting Nmap Results

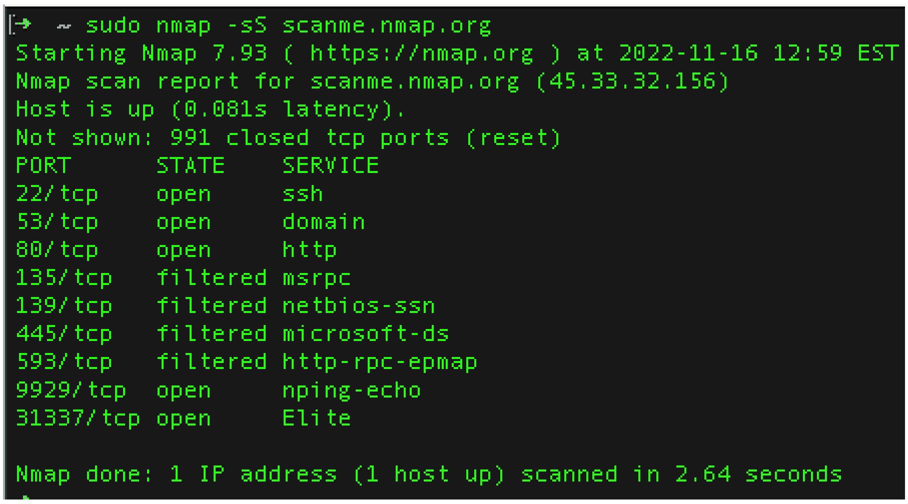
| State | Meaning |
|---|---|
| Open | Service actively accepting connections |
| Closed | No service running |
| Filtered | Firewall blocking probe |

---



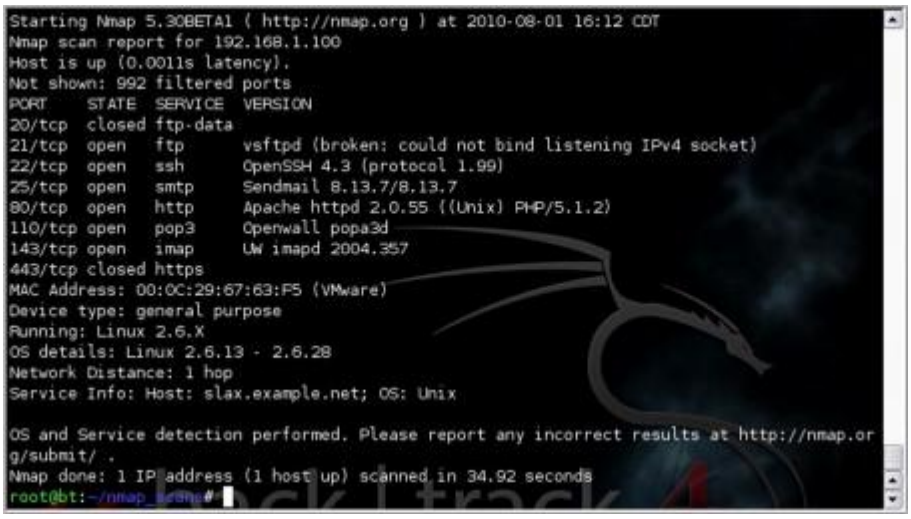
```
nmap -T4 -A -v scanme.nmap.org
```

```
host)
Initiating OS detection (try #1) against scanme.nmap.org
(64.13.134.52)
Initiating Traceroute at 12:05
Completed Traceroute at 12:05, 0.29s elapsed
Initiating Parallel DNS resolution of 12 hosts. at 12:05
Completed Parallel DNS resolution of 12 hosts. at 12:05, 6.64s
elapsed
NSE: Script scanning 64.13.134.52.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 12:05
Completed NSE at 12:05, 4.17s elapsed
Nmap scan report for scanme.nmap.org (64.13.134.52)
Host is up (0.074s latency).
Not shown: 993 filtered ports
PORT      STATE  SERVICE VERSION
22/tcp    open   ssh     OpenSSH 4.3 (protocol 2.0)
| ssh-hostkey: 1024
60:ac:4d:51:b1:cd:85:09:12:16:92:76:1d:5d:27:6e (DSA)
|_2048 2c:22:75:60:4b:c3:3b:18:a2:97:2c:96:7e:28:dc:dd (RSA)
25/tcp    closed smtp
```



```
➜  ~ sudo nmap -sS scanme.nmap.org
Starting Nmap 7.93 ( https://nmap.org ) at 2022-11-16 12:59 EST
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.081s latency).
Not shown: 991 closed tcp ports (reset)
PORT      STATE    SERVICE
22/tcp    open     ssh
53/tcp    open     domain
80/tcp    open     http
135/tcp   filtered msrpc
139/tcp   filtered netbios-ssn
445/tcp   filtered microsoft-ds
593/tcp   filtered http-rpc-epmap
9929/tcp  open     nping-echo
31337/tcp open     Elite

Nmap done: 1 IP address (1 host up) scanned in 2.64 seconds
```



```
Starting Nmap 5.30BETA1 ( http://nmap.org ) at 2010-08-01 16:12 CDT
Nmap scan report for 192.168.1.100
Host is up (0.0011s latency).
Not shown: 992 filtered ports
PORT    STATE  SERVICE  VERSION
20/tcp  closed ftp-data
21/tcp  open   ftp      vsftpd (broken: could not bind listening IPv4 socket)
22/tcp  open   ssh      OpenSSH 4.3 (protocol 1.99)
25/tcp  open   smtp     Sendmail 8.13.7/8.13.7
80/tcp  open   http     Apache httpd 2.0.55 ((Unix) PHP/5.1.2)
110/tcp open   pop3     Openwall popa3d
143/tcp open   imap     UW imapd 2004.357
443/tcp closed https
MAC Address: 00:0C:29:67:63:F5 (VMware)
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.13 - 2.6.28
Network Distance: 1 hop
Service Info: Host: slax.example.net; OS: Unix

OS and Service detection performed. Please report any incorrect results at http://nmap.or
g/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 34.92 seconds
root@bt:~/nmap_scans#
```

## 2.6 Security Risks Revealed by Nmap

- Open admin ports (22, 3389)
- Unnecessary services running
- Legacy software versions
- Exposed databases

**Real-World Example:**
An open MySQL port (3306) exposed to the internet can lead to **data breaches**.

## Section 3: Wireshark – Packet Analyzer

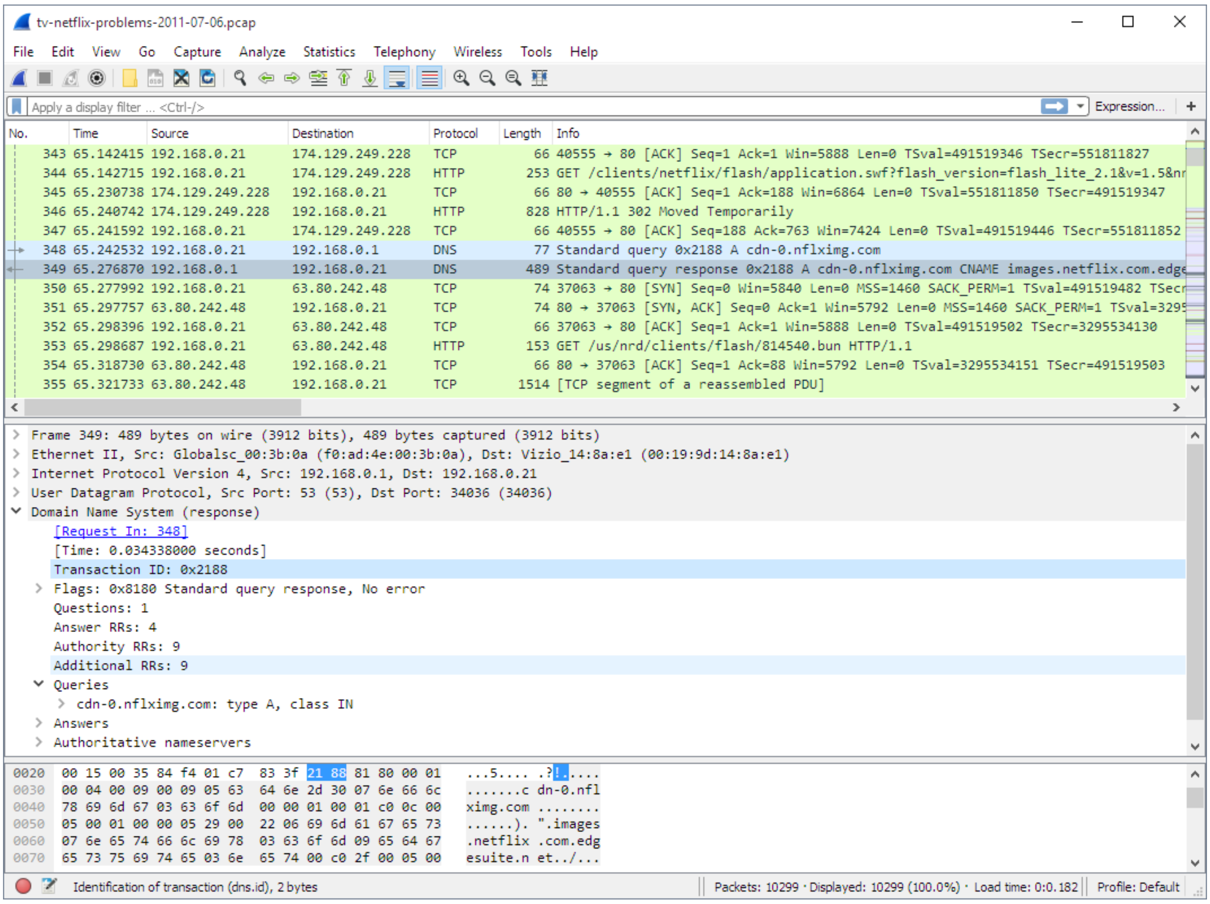
### 3.1 What is Wireshark?

Wireshark is a **network protocol analyzer** that allows you to:

- Capture live network traffic

- Inspect packet contents
- Analyze protocols
- Detect suspicious communication

It works at a **very low level** of networking.


tv-netflix-problems-2011-07-06.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
+

```
No.   Time      Source          Destination     Protocol  Length Info
  343 65.142415 192.168.0.21    174.129.249.228 TCP         66 40555 → 80 [ACK] Seq=1 Ack=1 Win=5888 Len=0 TSval=491519346 TSecr=551811827
  344 65.142715 192.168.0.21    174.129.249.228 HTTP       253 GET /clients/netflix/flash/application.swf?flash_version=flash_lite_2.1&v=1.5&nr
  345 65.230738 174.129.249.228 192.168.0.21    TCP         66 80 → 40555 [ACK] Seq=1 Ack=188 Win=6864 Len=0 TSval=551811850 TSecr=491519347
  346 65.240742 174.129.249.228 192.168.0.21    HTTP       828 HTTP/1.1 302 Moved Temporarily
  347 65.241592 192.168.0.21    174.129.249.228 TCP         66 40555 → 80 [ACK] Seq=188 Ack=763 Win=7424 Len=0 TSval=491519446 TSecr=551811852
  348 65.242532 192.168.0.21    192.168.0.1     DNS         77 Standard query 0x2188 A cdn-0.nflximg.com
  349 65.276870 192.168.0.1     192.168.0.21    DNS        489 Standard query response 0x2188 A cdn-0.nflximg.com CNAME images.netflix.com.edge
  350 65.277992 192.168.0.21    63.80.242.48    TCP         74 37063 → 80 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 SACK_PERM=1 TSval=491519482 TSecr
  351 65.297757 63.80.242.48    192.168.0.21    TCP         74 80 → 37063 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=1460 SACK_PERM=1 TSval=3295
  352 65.298396 192.168.0.21    63.80.242.48    TCP         66 37063 → 80 [ACK] Seq=1 Ack=1 Win=5888 Len=0 TSval=491519502 TSecr=3295534130
  353 65.298687 192.168.0.21    63.80.242.48    HTTP       153 GET /us/nrd/clients/flash/814540.bun HTTP/1.1
  354 65.318730 63.80.242.48    192.168.0.21    TCP         66 80 → 37063 [ACK] Seq=1 Ack=88 Win=5792 Len=0 TSval=3295534151 TSecr=491519503
  355 65.321733 63.80.242.48    192.168.0.21    TCP       1514 [TCP segment of a reassembled PDU]
```

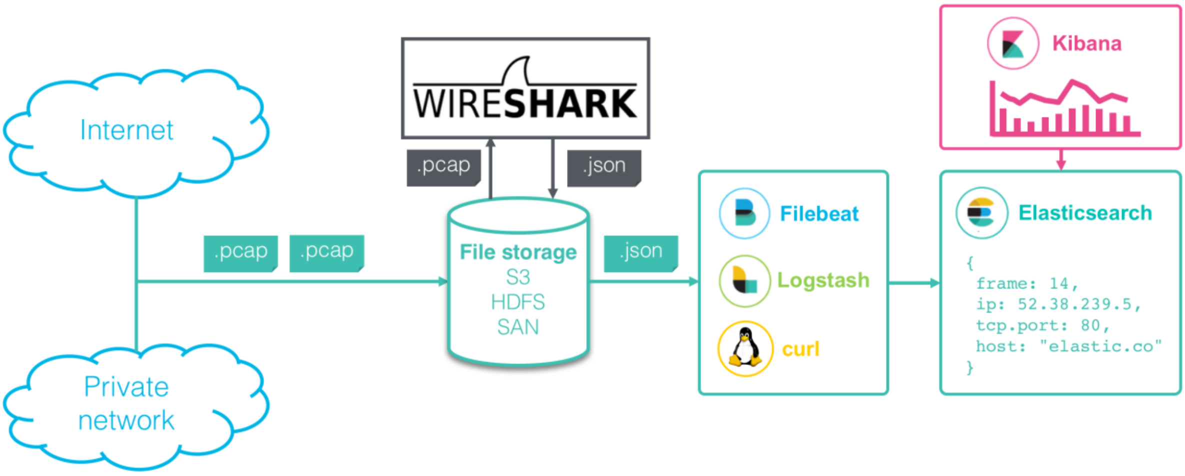
```
> Frame 349: 489 bytes on wire (3912 bits), 489 bytes captured (3912 bits)
> Ethernet II, Src: Globalsc_00:3b:0a (f0:ad:4e:00:3b:0a), Dst: Vizio_14:8a:e1 (00:19:9d:14:8a:e1)
> Internet Protocol Version 4, Src: 192.168.0.1, Dst: 192.168.0.21
> User Datagram Protocol, Src Port: 53 (53), Dst Port: 34036 (34036)
v Domain Name System (response)
    [Request In: 348]
    [Time: 0.034338000 seconds]
    Transaction ID: 0x2188
  > Flags: 0x8180 Standard query response, No error
    Questions: 1
    Answer RRs: 4
    Authority RRs: 9
    Additional RRs: 9
  v Queries
    > cdn-0.nflximg.com: type A, class IN
  > Answers
  > Authoritative nameservers
```

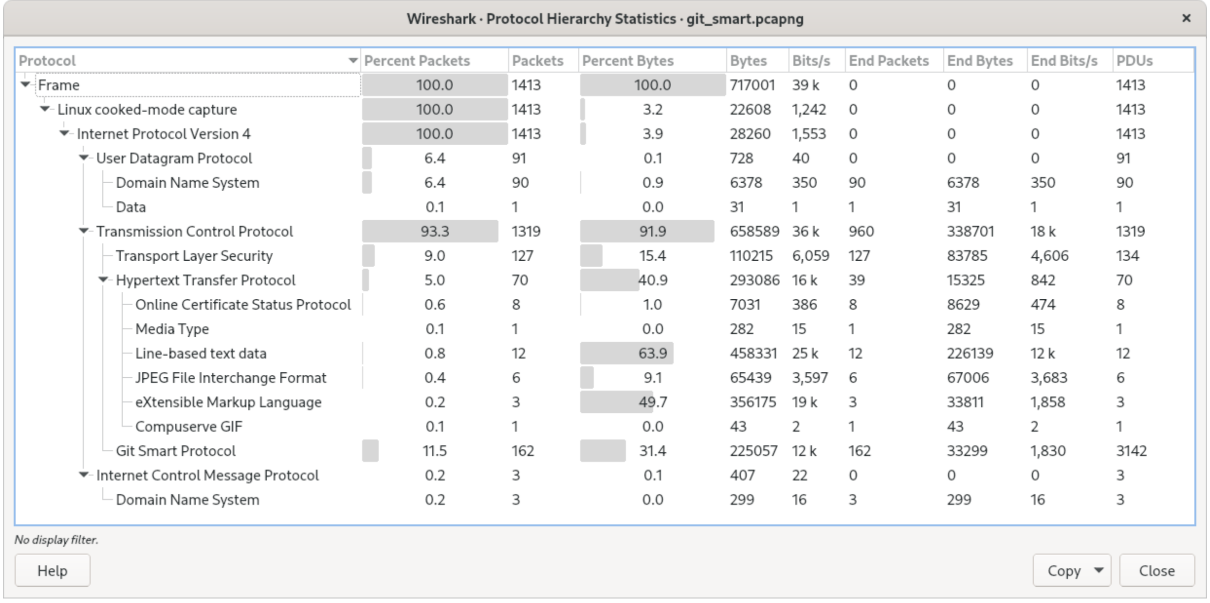
```
0020  00 15 00 35 84 f4 01 c7  83 3f 21 88 81 80 00 01   ...5.... .?!.....
0030  00 04 00 09 00 09 05 63  64 6e 2d 30 07 6e 66 6c   .......c dn-0.nfl
0040  78 69 6d 67 03 63 6f 6d  00 00 01 00 01 c0 0c 00   ximg.com ........
0050  05 00 01 00 00 05 29 00  22 06 69 6d 61 67 65 73   ......). ".images
0060  07 6e 65 74 66 6c 69 78  03 63 6f 6d 09 65 64 67   .netflix .com.edg
0070  65 73 75 69 74 65 03 6e  65 74 00 c0 2f 00 05 00   esuite.n et../...
```

Identification of transaction (dns.id), 2 bytes

Packets: 10299 · Displayed: 10299 (100.0%) · Load time: 0:0.182

Profile: Default





Wireshark · Protocol Hierarchy Statistics · git_smart.pcapng
Help
Copy
Close

| Protocol | Percent Packets | Packets | Percent Bytes | Bytes | Bits/s | End Packets | End Bytes | End Bits/s | PDUs |
|---|---|---|---|---|---|---|---|---|---|
| Frame | 100.0 | 1413 | 100.0 | 717001 | 39 k | 0 | 0 | 0 | 1413 |
| Linux cooked-mode capture | 100.0 | 1413 | 3.2 | 22608 | 1,242 | 0 | 0 | 0 | 1413 |
| Internet Protocol Version 4 | 100.0 | 1413 | 3.9 | 28260 | 1,553 | 0 | 0 | 0 | 1413 |
| User Datagram Protocol | 6.4 | 91 | 0.1 | 728 | 40 | 0 | 0 | 0 | 91 |
| Domain Name System | 6.4 | 90 | 0.9 | 6378 | 350 | 90 | 6378 | 350 | 90 |
| Data | 0.1 | 1 | 0.0 | 31 | 1 | 1 | 31 | 1 | 1 |
| Transmission Control Protocol | 93.3 | 1319 | 91.9 | 658589 | 36 k | 960 | 338701 | 18 k | 1319 |
| Transport Layer Security | 9.0 | 127 | 15.4 | 110215 | 6,059 | 127 | 83785 | 4,606 | 134 |
| Hypertext Transfer Protocol | 5.0 | 70 | 40.9 | 293086 | 16 k | 39 | 15325 | 842 | 70 |
| Online Certificate Status Protocol | 0.6 | 8 | 1.0 | 7031 | 386 | 8 | 8629 | 474 | 8 |
| Media Type | 0.1 | 1 | 0.0 | 282 | 15 | 1 | 282 | 15 | 1 |
| Line-based text data | 0.8 | 12 | 63.9 | 458331 | 25 k | 12 | 226139 | 12 k | 12 |
| JPEG File Interchange Format | 0.4 | 6 | 9.1 | 65439 | 3,597 | 6 | 67006 | 3,683 | 6 |
| eXtensible Markup Language | 0.2 | 3 | 49.7 | 356175 | 19 k | 3 | 33811 | 1,858 | 3 |
| Compuserve GIF | 0.1 | 1 | 0.0 | 43 | 2 | 1 | 43 | 2 | 1 |
| Git Smart Protocol | 11.5 | 162 | 31.4 | 225057 | 12 k | 162 | 33299 | 1,830 | 3142 |
| Internet Control Message Protocol | 0.2 | 3 | 0.1 | 407 | 22 | 0 | 0 | 0 | 3 |
| Domain Name System | 0.2 | 3 | 0.0 | 299 | 16 | 3 | 299 | 16 | 3 |

*No display filter.*

## 3.2 Why Wireshark is Important

Wireshark answers questions like:

- What data is moving on the network?
- Is sensitive data transmitted unencrypted?
- Is malware communicating externally?
- Are there suspicious DNS requests?

**Wireshark shows what firewalls cannot**.

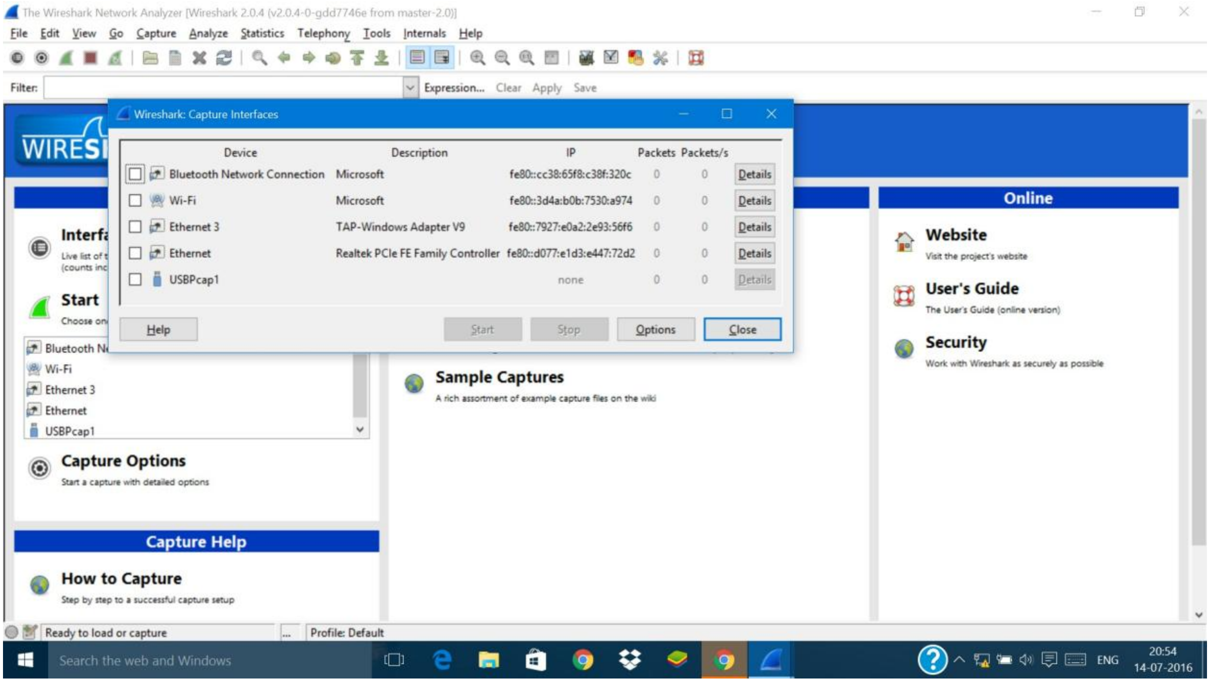
## 3.3 Installing Wireshark

- Download and install from official site
- Install **Npcap** (Windows) for packet capture
- Run as administrator/root

## 3.4 Capturing Network Traffic

Steps:

1. Select network interface (Wi-Fi/Ethernet)
2. Click **Start Capture**
3. Generate traffic (open website, ping)
4. Stop capture



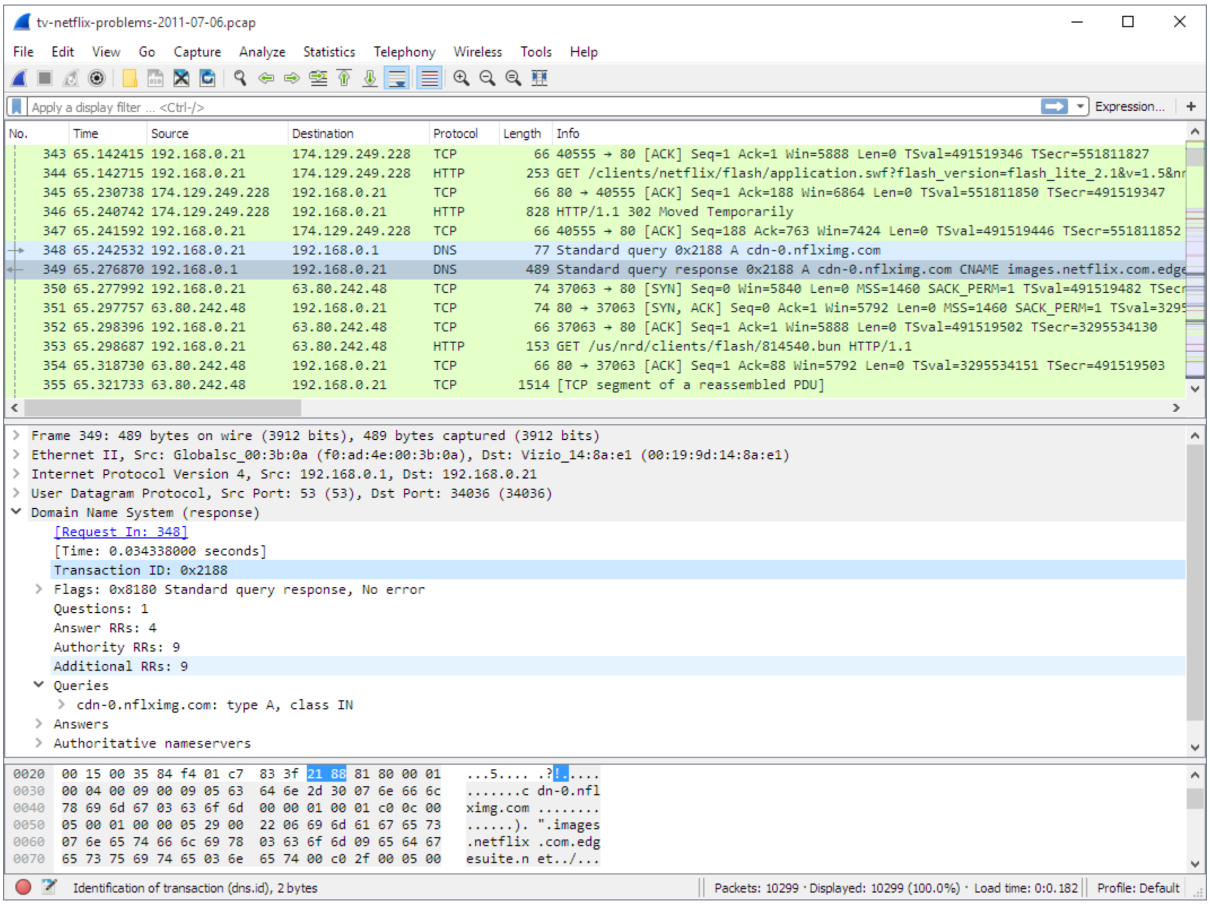
The Wireshark Network Analyzer [Wireshark 2.0.4 (v2.0.4-0-gdd7746e from master-2.0)]

File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter:
Expression... Clear Apply Save

Wireshark: Capture Interfaces

| | Device | Description | IP | Packets | Packets/s | |
|---|---|---|---|---|---|---|
| □ | Bluetooth Network Connection | Microsoft | fe80::cc38:65f8:c38f:320c | 0 | 0 | Details |
| □ | Wi-Fi | Microsoft | fe80::3d4a:b0b:7530:a974 | 0 | 0 | Details |
| □ | Ethernet 3 | TAP-Windows Adapter V9 | fe80::7927:e0a2:2e93:56f6 | 0 | 0 | Details |
| □ | Ethernet | Realtek PCIe FE Family Controller | fe80::d077:e1d3:e447:72d2 | 0 | 0 | Details |
| □ | USBPcap1 | | none | 0 | 0 | Details |

Help
Start
Stop
Options
Close
WIRESH
Interfa
Live list of t
(counts inc
Start
Choose on
Bluetooth N
Wi-Fi
Ethernet 3
Ethernet
USBPcap1
Capture Options
Start a capture with detailed options
Capture Help
How to Capture
Step by step to a successful capture setup
Sample Captures
A rich assortment of example capture files on the wiki
Online
Website
Visit the project's website
User's Guide
The User's Guide (online version)
Security
Work with Wireshark as securely as possible
Ready to load or capture
Profile: Default
Search the web and Windows
ENG
20:54
14-07-2016



tv-netflix-problems-2011-07-06.pcap

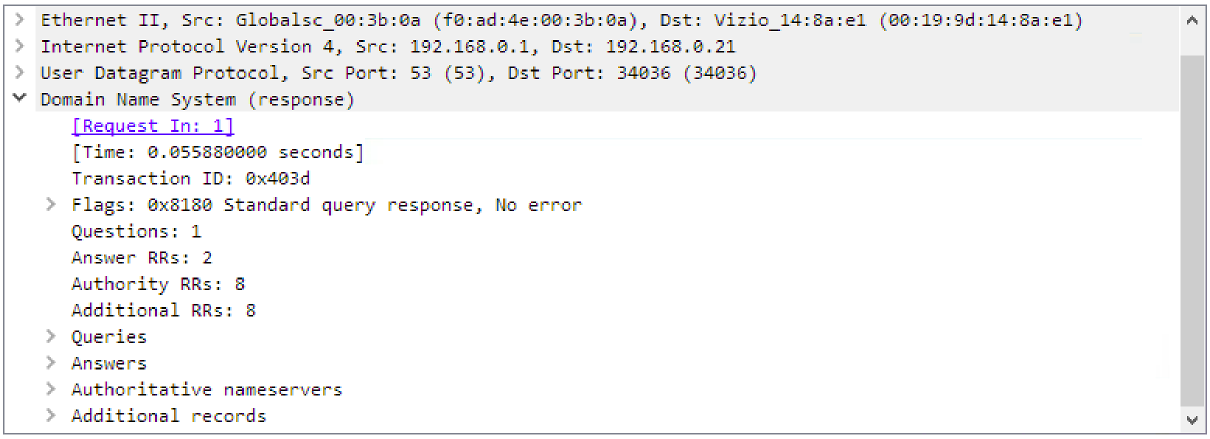
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
+

```
No.   Time      Source          Destination     Protocol Length Info
  343 65.142415 192.168.0.21    174.129.249.228 TCP        66 40555 → 80 [ACK] Seq=1 Ack=1 Win=5888 Len=0 TSval=491519346 TSecr=551811827
  344 65.142715 192.168.0.21    174.129.249.228 HTTP      253 GET /clients/netflix/flash/application.swf?flash_version=flash_lite_2.1&v=1.5&nr
  345 65.230738 174.129.249.228 192.168.0.21    TCP        66 80 → 40555 [ACK] Seq=1 Ack=188 Win=6864 Len=0 TSval=551811850 TSecr=491519347
  346 65.240742 174.129.249.228 192.168.0.21    HTTP      828 HTTP/1.1 302 Moved Temporarily
  347 65.241592 192.168.0.21    174.129.249.228 TCP        66 40555 → 80 [ACK] Seq=188 Ack=763 Win=7424 Len=0 TSval=491519446 TSecr=551811852
  348 65.242532 192.168.0.21    192.168.0.1     DNS        77 Standard query 0x2188 A cdn-0.nflximg.com
  349 65.276870 192.168.0.1     192.168.0.21    DNS       489 Standard query response 0x2188 A cdn-0.nflximg.com CNAME images.netflix.com.edge
  350 65.277992 192.168.0.21    63.80.242.48    TCP        74 37063 → 80 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 SACK_PERM=1 TSval=491519482 TSecr
  351 65.297757 63.80.242.48    192.168.0.21    TCP        74 80 → 37063 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=1460 SACK_PERM=1 TSval=3295
  352 65.298396 192.168.0.21    63.80.242.48    TCP        66 37063 → 80 [ACK] Seq=1 Ack=1 Win=5888 Len=0 TSval=491519502 TSecr=3295534130
  353 65.298687 192.168.0.21    63.80.242.48    HTTP      153 GET /us/nrd/clients/flash/814540.bun HTTP/1.1
  354 65.318730 63.80.242.48    192.168.0.21    TCP        66 80 → 37063 [ACK] Seq=1 Ack=88 Win=5792 Len=0 TSval=3295534151 TSecr=491519503
  355 65.321733 63.80.242.48    192.168.0.21    TCP      1514 [TCP segment of a reassembled PDU]
```

```
> Frame 349: 489 bytes on wire (3912 bits), 489 bytes captured (3912 bits)
> Ethernet II, Src: Globalsc_00:3b:0a (f0:ad:4e:00:3b:0a), Dst: Vizio_14:8a:e1 (00:19:9d:14:8a:e1)
> Internet Protocol Version 4, Src: 192.168.0.1, Dst: 192.168.0.21
> User Datagram Protocol, Src Port: 53 (53), Dst Port: 34036 (34036)
v Domain Name System (response)
    [Request In: 348]
    [Time: 0.034338000 seconds]
    Transaction ID: 0x2188
  > Flags: 0x8180 Standard query response, No error
    Questions: 1
    Answer RRs: 4
    Authority RRs: 9
    Additional RRs: 9
  v Queries
    > cdn-0.nflximg.com: type A, class IN
  > Answers
  > Authoritative nameservers
```

```
0020  00 15 00 35 84 f4 01 c7  83 3f 21 88 81 80 00 01   ...5.... .?!.....
0030  00 04 00 09 00 09 05 63  64 6e 2d 30 07 6e 66 6c   .......c dn-0.nfl
0040  78 69 6d 67 03 63 6f 6d  00 00 01 00 01 c0 0c 00   ximg.com ........
0050  05 00 01 00 00 05 29 00  22 06 69 6d 61 67 65 73   ......). ".images
0060  07 6e 65 74 66 6c 69 78  03 63 6f 6d 09 65 64 67   .netflix .com.edg
0070  65 73 75 69 74 65 03 6e  65 74 00 c0 2f 00 05 00   esuite.n et../...
```

Identification of transaction (dns.id), 2 bytes | Packets: 10299 · Displayed: 10299 (100.0%) · Load time: 0:0.182 | Profile: Default



```
> Ethernet II, Src: Globalsc_00:3b:0a (f0:ad:4e:00:3b:0a), Dst: Vizio_14:8a:e1 (00:19:9d:14:8a:e1)
> Internet Protocol Version 4, Src: 192.168.0.1, Dst: 192.168.0.21
> User Datagram Protocol, Src Port: 53 (53), Dst Port: 34036 (34036)
v Domain Name System (response)
    [Request In: 1]
    [Time: 0.055880000 seconds]
    Transaction ID: 0x403d
  > Flags: 0x8180 Standard query response, No error
    Questions: 1
    Answer RRs: 2
    Authority RRs: 8
    Additional RRs: 8
  > Queries
  > Answers
  > Authoritative nameservers
  > Additional records
```

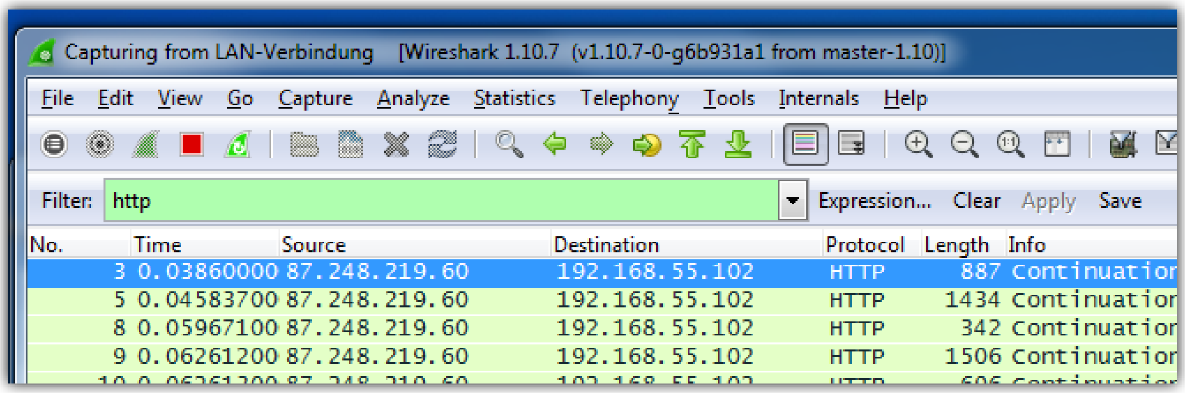
### 3.5 Understanding Packets

Each packet contains:

- **Frame** – Physical layer
- **Ethernet** – MAC addresses
- **IP** – Source & destination IP
- **Transport** – TCP/UDP
- **Application** – HTTP, DNS, FTP

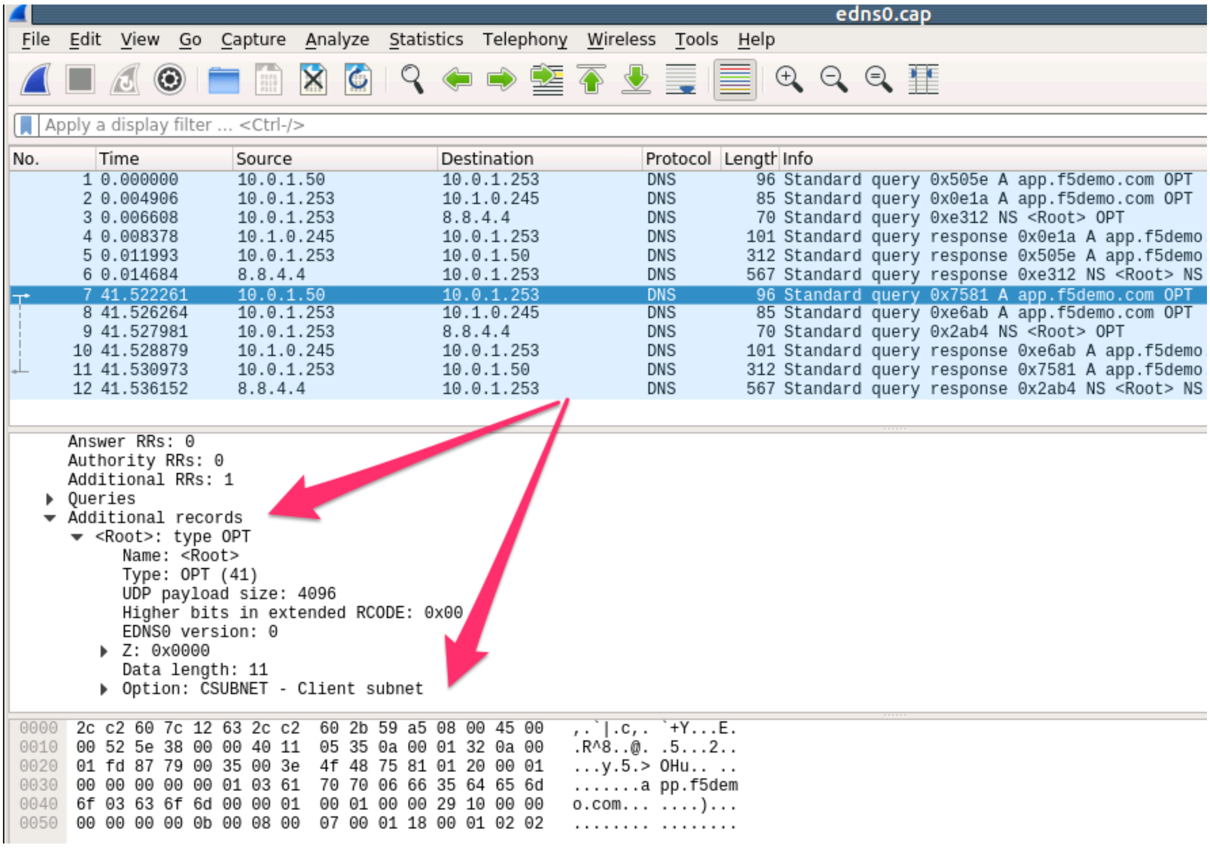
### 3.6 Common Protocol Filters

| Filter | Purpose |
|---|---|
| http | View web traffic |
| dns | DNS requests |
| tcp | TCP packets |
| udp | UDP packets |
| icmp | Ping traffic |


Capturing from LAN-Verbindung [Wireshark 1.10.7 (v1.10.7-0-g6b931a1 from master-1.10)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Expression... Clear Apply Save

Filter: http

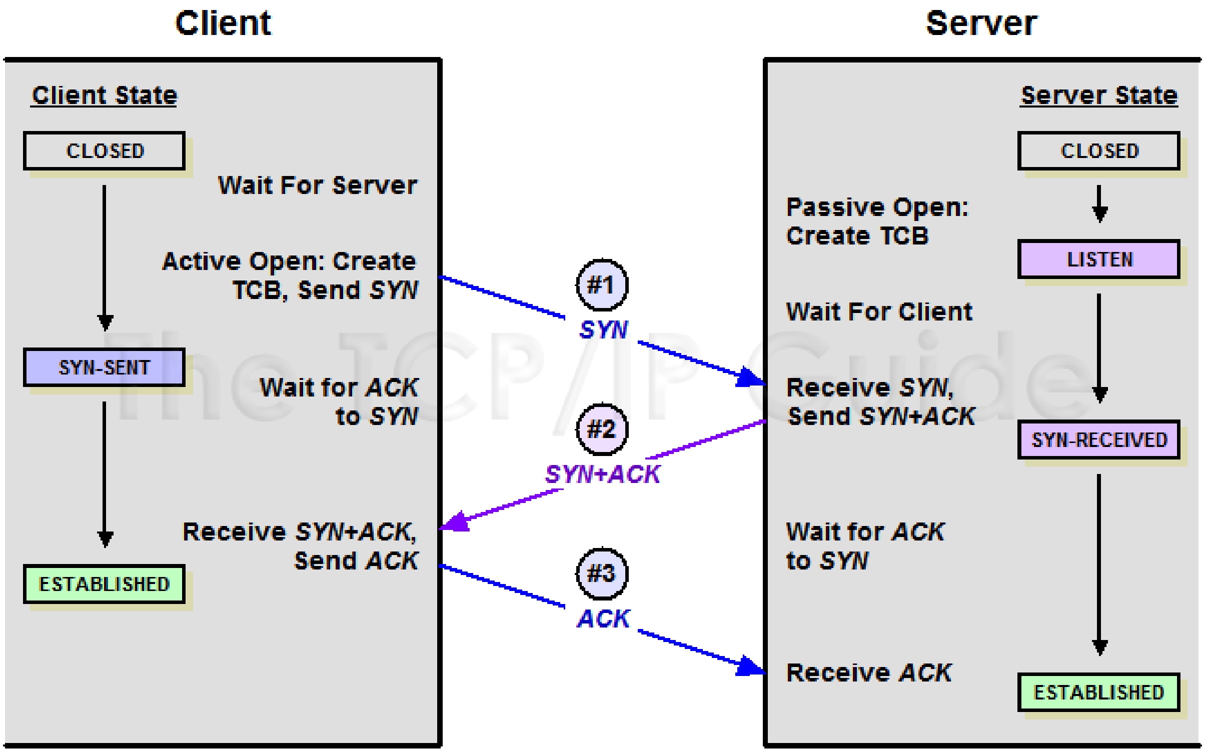
```
No.   Time        Source         Destination      Protocol Length Info
    3 0.03860000 87.248.219.60   192.168.55.102   HTTP       887 Continuation
    5 0.04583700 87.248.219.60   192.168.55.102   HTTP      1434 Continuation
    8 0.05967100 87.248.219.60   192.168.55.102   HTTP       342 Continuation
    9 0.06261200 87.248.219.60   192.168.55.102   HTTP      1506 Continuation
   10 0.06261200 87.248.219.60   192.168.55.102   HTTP       606 Continuation
```


edns0.cap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.  Time       Source       Destination  Protocol Length Info
   1 0.000000   10.0.1.50    10.0.1.253   DNS        96 Standard query 0x505e A app.f5demo.com OPT
   2 0.004906   10.0.1.253   10.1.0.245   DNS        85 Standard query 0x0e1a A app.f5demo.com OPT
   3 0.006608   10.0.1.253   8.8.4.4      DNS        70 Standard query 0xe312 NS <Root> OPT
   4 0.008378   10.1.0.245   10.0.1.253   DNS       101 Standard query response 0x0e1a A app.f5demo
   5 0.011993   10.0.1.253   10.0.1.50    DNS       312 Standard query response 0x505e A app.f5demo
   6 0.014684   8.8.4.4      10.0.1.253   DNS       567 Standard query response 0xe312 NS <Root> NS
   7 41.522261  10.0.1.50    10.0.1.253   DNS        96 Standard query 0x7581 A app.f5demo.com OPT
   8 41.526264  10.0.1.253   10.1.0.245   DNS        85 Standard query 0xe6ab A app.f5demo.com OPT
   9 41.527981  10.0.1.253   8.8.4.4      DNS        70 Standard query 0x2ab4 NS <Root> OPT
  10 41.528879  10.1.0.245   10.0.1.253   DNS       101 Standard query response 0xe6ab A app.f5demo
  11 41.530973  10.0.1.253   10.0.1.50    DNS       312 Standard query response 0x7581 A app.f5demo
  12 41.536152  8.8.4.4      10.0.1.253   DNS       567 Standard query response 0x2ab4 NS <Root> NS
```

```
      Answer RRs: 0
      Authority RRs: 0
      Additional RRs: 1
    ▸ Queries
    ▾ Additional records
      ▾ <Root>: type OPT
            Name: <Root>
            Type: OPT (41)
            UDP payload size: 4096
            Higher bits in extended RCODE: 0x00
            EDNS0 version: 0
          ▸ Z: 0x0000
            Data length: 11
          ▸ Option: CSUBNET - Client subnet
```

```
0000  2c c2 60 7c 12 63 2c c2  60 2b 59 a5 08 00 45 00   ,.`|.c,. `+Y...E.
0010  00 52 5e 38 00 00 40 11  05 35 0a 00 01 32 0a 00   .R^8..@. .5...2..
0020  01 fd 87 79 00 35 00 3e  4f 48 75 81 01 20 00 01   ...y.5.> OHu.. ..
0030  00 00 00 00 00 01 03 61  70 70 06 66 35 64 65 6d   .......a pp.f5dem
0040  6f 03 63 6f 6d 00 00 01  00 01 00 00 29 10 00 00   o.com... ....)...
0050  00 00 00 00 0b 00 08 00  07 00 01 18 00 01 02 02   ........ ........
```



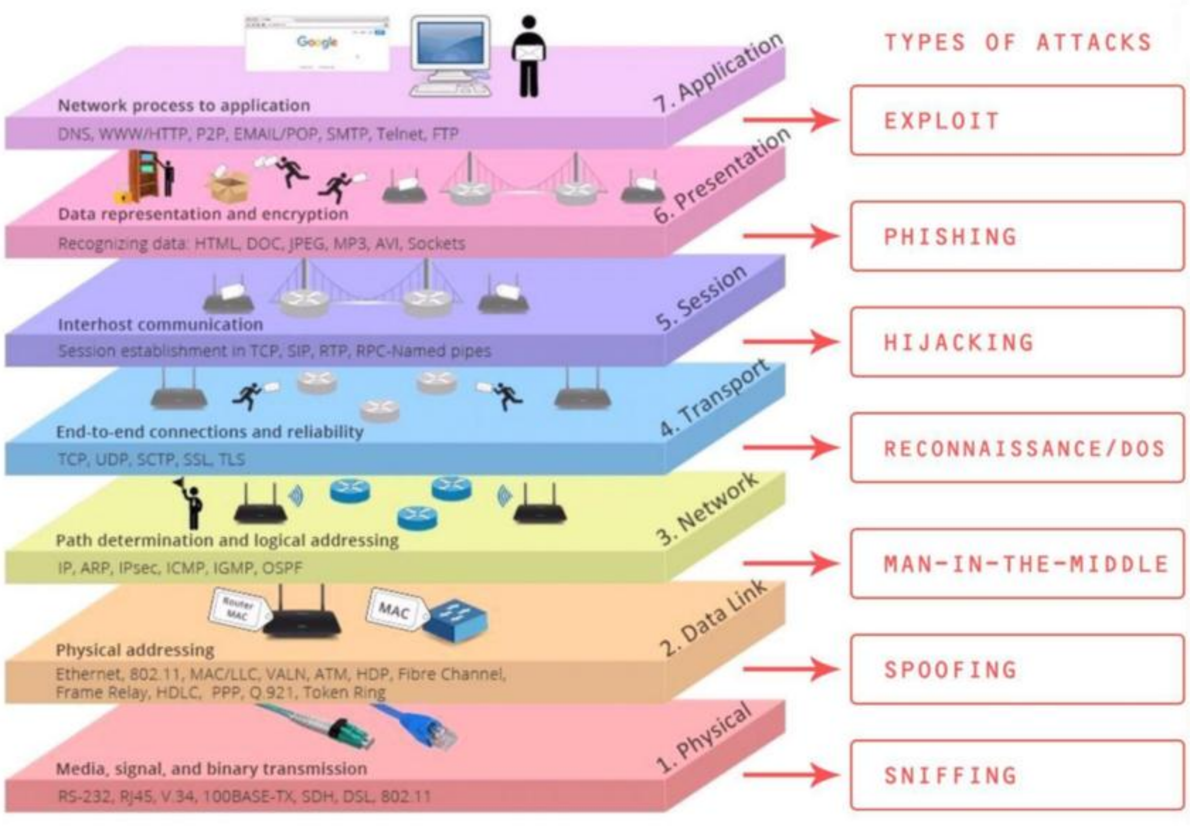
### 3.7 Security Insights from Wireshark

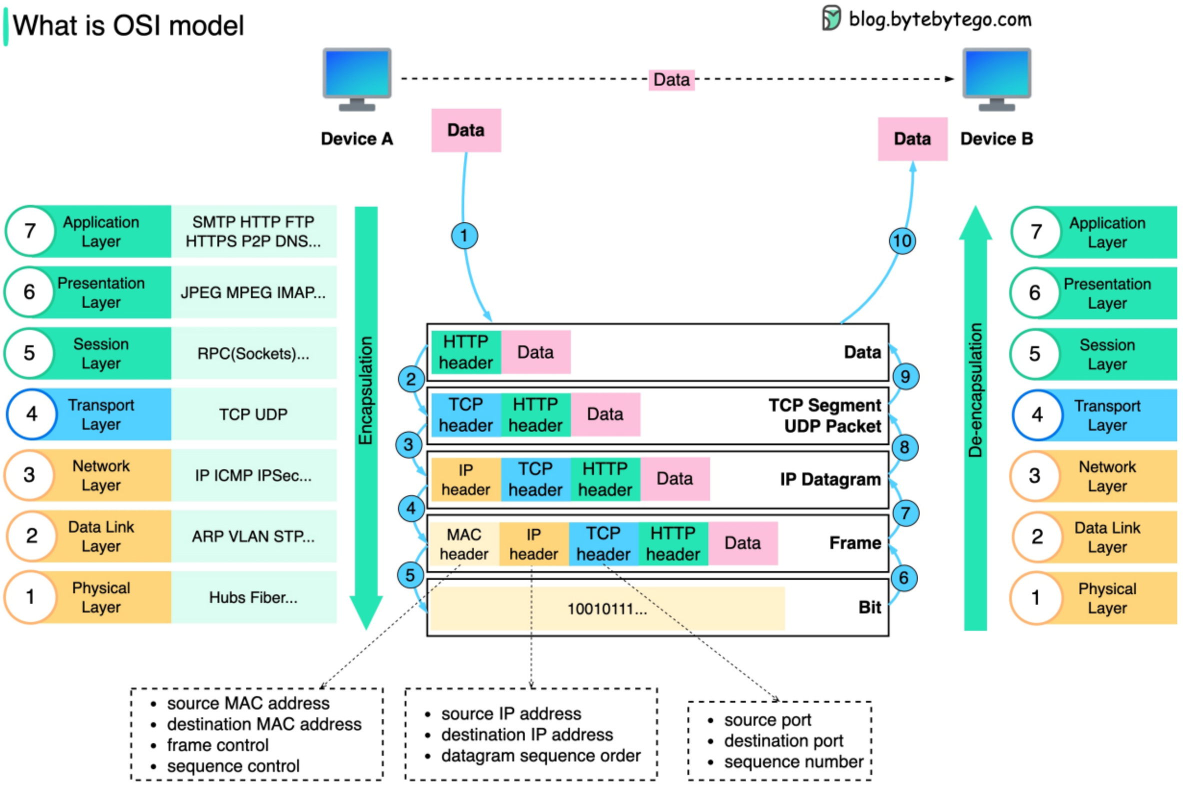
- Plaintext credentials in HTTP
- DNS tunneling attempts
- Suspicious IP connections
- Abnormal packet frequency (DDoS signs)

**SOC teams rely heavily on packet analysis during incidents.**

## 4: Mapping Tools to OSI Model

| **OSI Layer** | **Tool Usage** |
|---|---|
| Layer 2 | Wireshark (Ethernet) |
| Layer 3 | Nmap & Wireshark (IP) |
| Layer 4 | Nmap (TCP/UDP ports) |
| Layer 7 | Wireshark (HTTP/DNS) |


TYPES OF ATTACKS
7. Application
Network process to application
DNS, WWW/HTTP, P2P, EMAIL/POP, SMTP, Telnet, FTP
EXPLOIT
6. Presentation
Data representation and encryption
Recognizing data: HTML, DOC, JPEG, MP3, AVI, Sockets
PHISHING
5. Session
Interhost communication
Session establishment in TCP, SIP, RTP, RPC-Named pipes
HIJACKING
4. Transport
End-to-end connections and reliability
TCP, UDP, SCTP, SSL, TLS
RECONNAISSANCE/DOS
3. Network
Path determination and logical addressing
IP, ARP, IPsec, ICMP, IGMP, OSPF
MAN-IN-THE-MIDDLE
2. Data Link
Physical addressing
Ethernet, 802.11, MAC/LLC, VALN, ATM, HDP, Fibre Channel, Frame Relay, HDLC, PPP, Q.921, Token Ring
Router MAC
MAC
SPOOFING
1. Physical
Media, signal, and binary transmission
RS-232, RJ45, V.34, 100BASE-TX, SDH, DSL, 802.11
SNIFFING



## 5: Real-World Security Scenarios

### Scenario 1: Data Breach Investigation

- Nmap → Find exposed ports
- Wireshark → Inspect leaked traffic

**Scenario 2: Malware Detection**

- Nmap → Identify suspicious services
- Wireshark → Detect command-and-control traffic