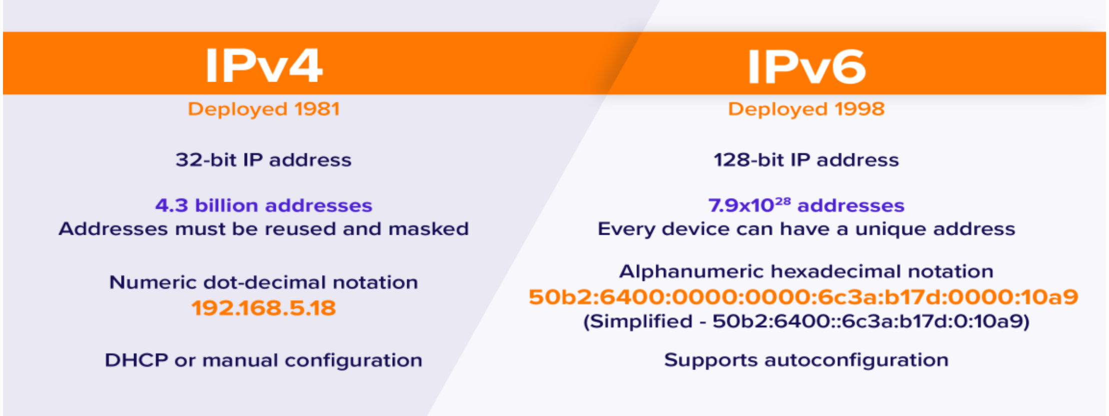
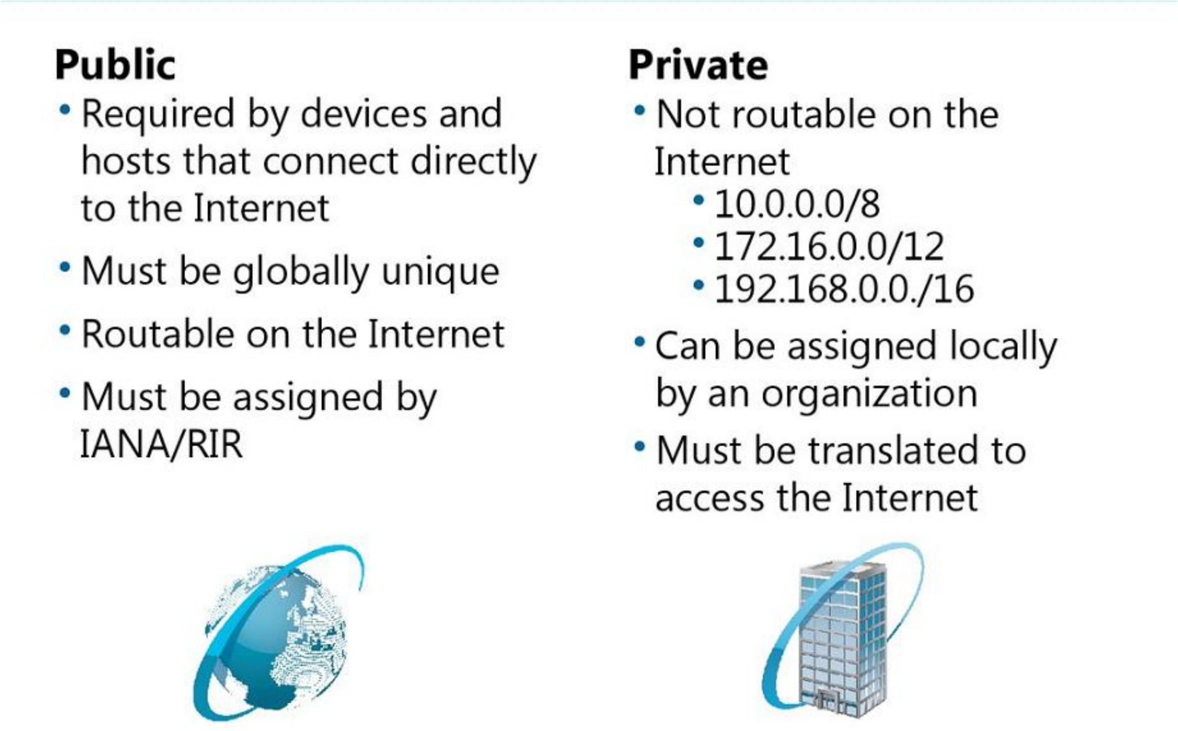
CYBER SECURITY INTERNSHIP

TASK -3

1. IP Address (Internet Protocol)



IPv4	IPv6
Deployed 1981	Deployed 1998
32-bit IP address	128-bit IP address
4.3 billion addresses Addresses must be reused and masked	$7.9x10^{28}$ addresses Every device can have a unique address
Numeric dot-decimal notation 192.168.5.18	Alphanumeric hexadecimal notation 50b2:6400:0000:0000:6c3a:b17d:0000:10a9 (Simplified - 50b2:6400::6c3a:b17d:0:10a9)
DHCP or manual configuration	Supports autoconfiguration



Public

- Required by devices and hosts that connect directly to the Internet
- Must be globally unique
- Routable on the Internet
- Must be assigned by IANA/RIR

Private

- Not routable on the Internet
  - 10.0.0.0/8
  - 172.16.0.0/12
  - 192.168.0.0./16
- Can be assigned locally by an organization
- Must be translated to access the Internet

What it is

An **IP address** is a **unique number** given to a device on a network so it can **send and receive data**.

Simple way to understand

Think of an IP address as your **home address**.
Without it, the internet wouldn’t know **where to send data**.

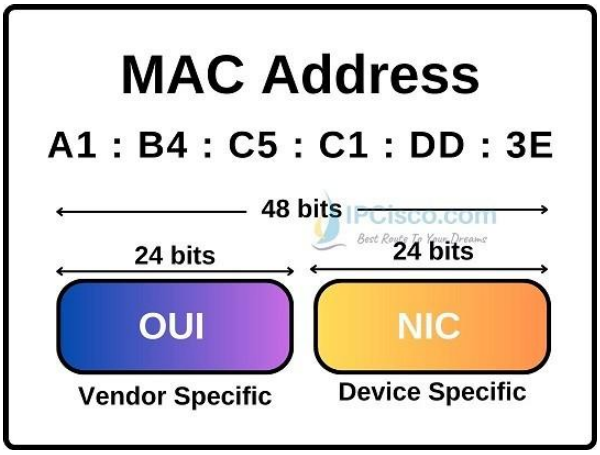
Example

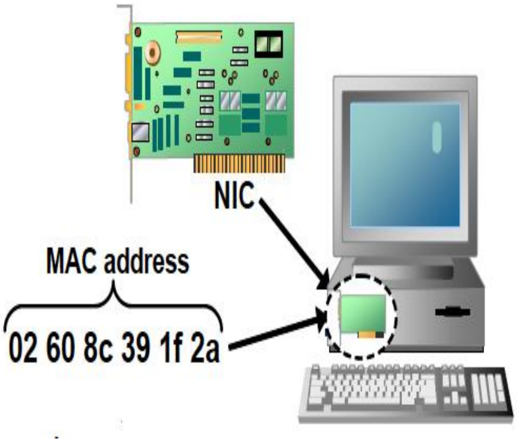
```
192.168.1.10
```

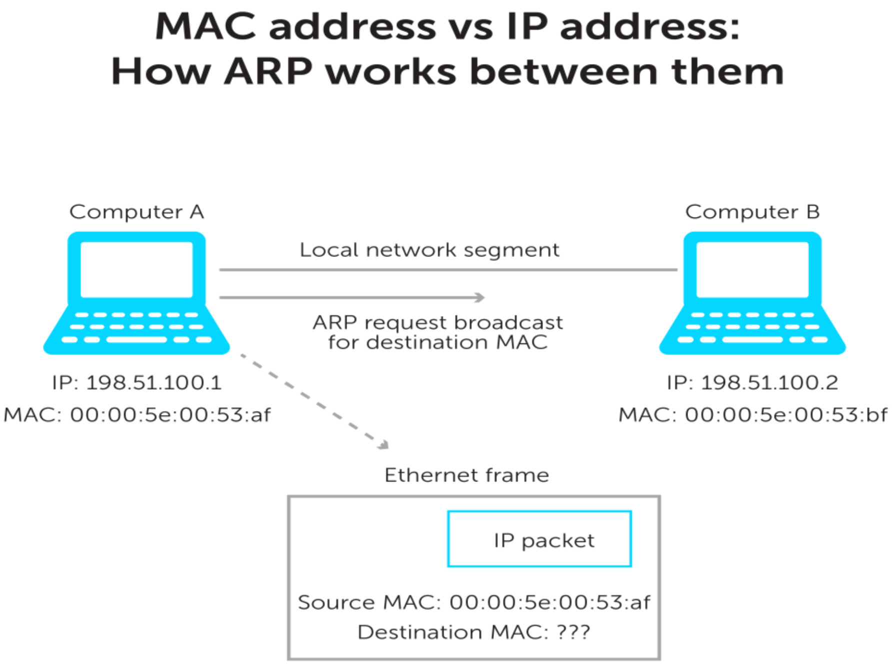
Key points

- Identifies devices on a network
- Can **change** (dynamic IP)
- Works on the **internet and local networks**

□ MAC Address (Media Access Control)







What it is

A **MAC address** is a **permanent hardware ID** assigned to your network card.

Simple way to understand

If IP is your **home address**,
MAC is your **fingerprint** — unique and fixed.

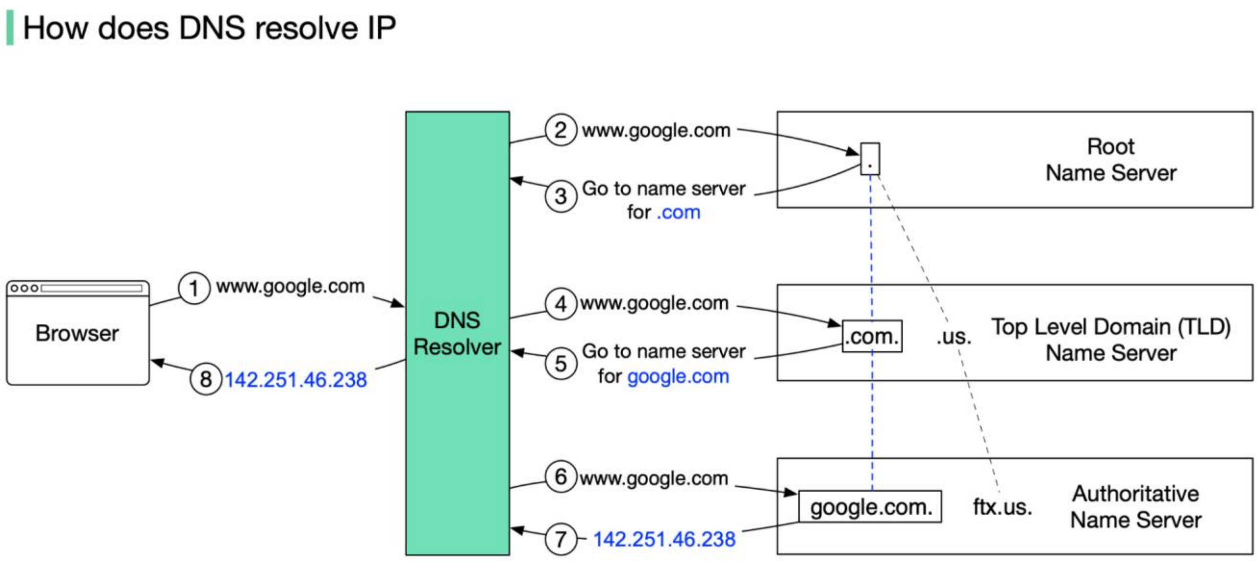
Example

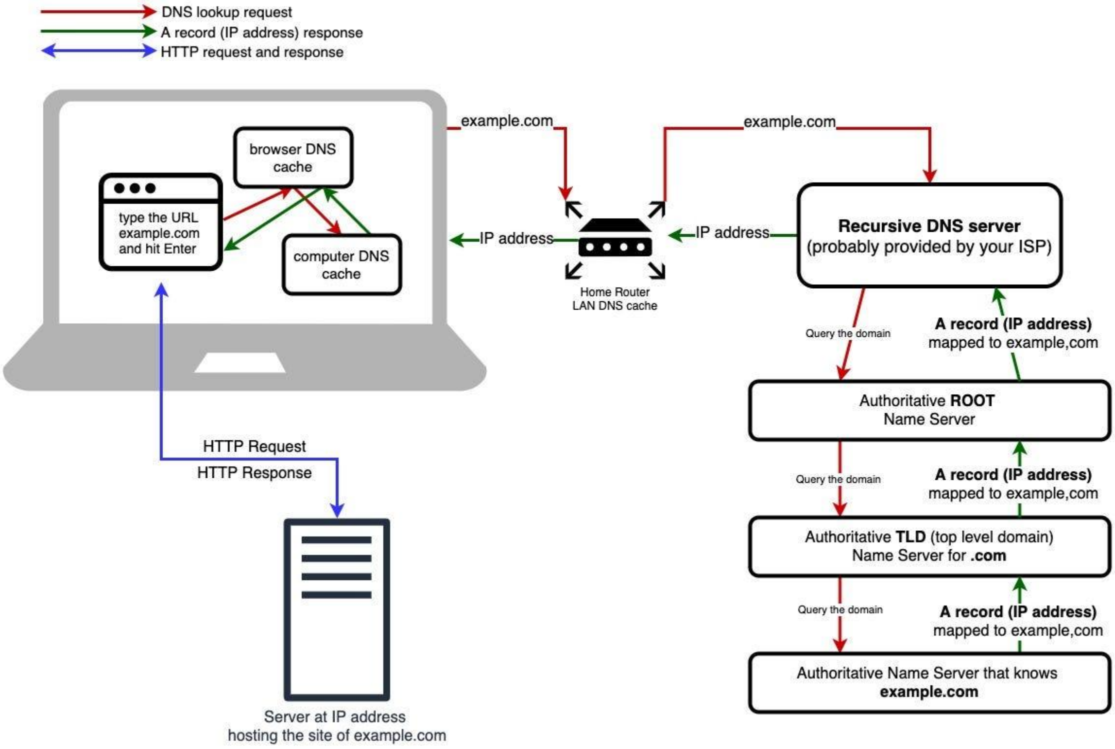
```
00:1A:2B:3C:4D:5E
```

Key points

- **Does not change**
- Used inside **local networks**
- Helps switches identify devices

□ DNS (Domain Name System)




DNS lookup request
A record (IP address) response
HTTP request and response
type the URL example.com and hit Enter
browser DNS cache
computer DNS cache
example.com
IP address
Home Router LAN DNS cache
example.com
IP address
Recursive DNS server (probably provided by your ISP)
Query the domain
A record (IP address) mapped to example,com
Authoritative ROOT Name Server
Query the domain
A record (IP address) mapped to example,com
Authoritative TLD (top level domain) Name Server for .com
Query the domain
A record (IP address) mapped to example,com
Authoritative Name Server that knows example.com
HTTP Request
HTTP Response
Server at IP address hosting the site of example.com

What it is

DNS converts **website names into IP addresses**.

Simple way to understand

DNS is like a **phone contact list**:

- You save **names**
- Phone uses **numbers**

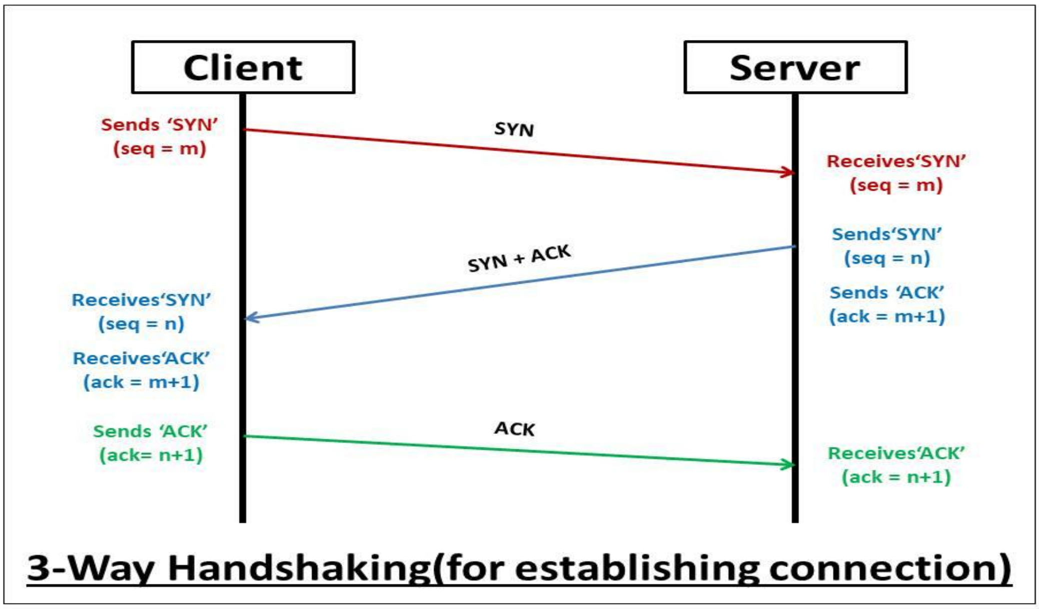
Example

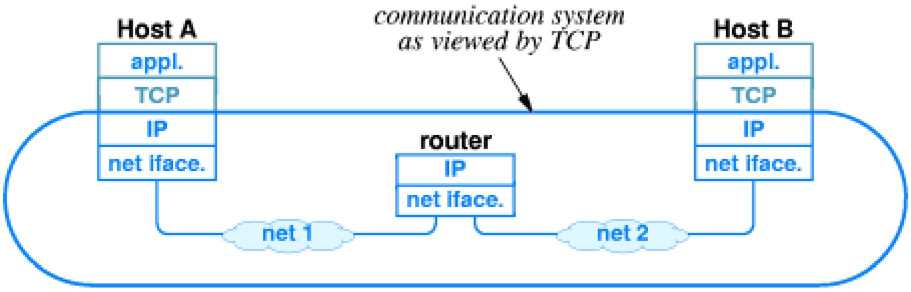
```
www.google.com → 142.250.190.14
```

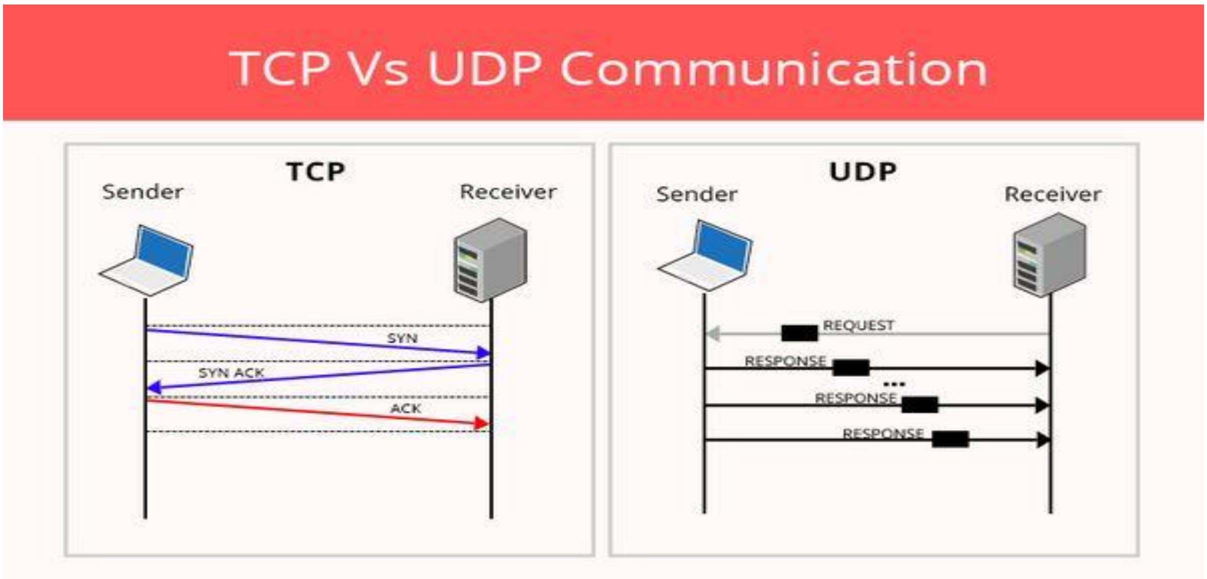
Key points

- Makes the internet **user-friendly**
- Without DNS, we must remember IPs
- Used every time you open a website

□ TCP (Transmission Control Protocol)







What it is

TCP is a **reliable communication method** that ensures data reaches correctly.

Simple way to understand

TCP is like **sending a registered parcel**:

- Confirmation required
- Resent if lost
- Order maintained

Used in

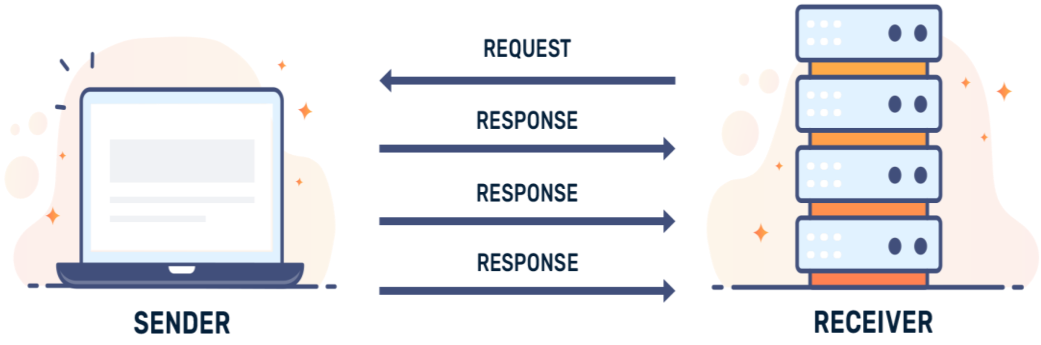
- Websites (HTTP/HTTPS)
- Emails
- File downloads

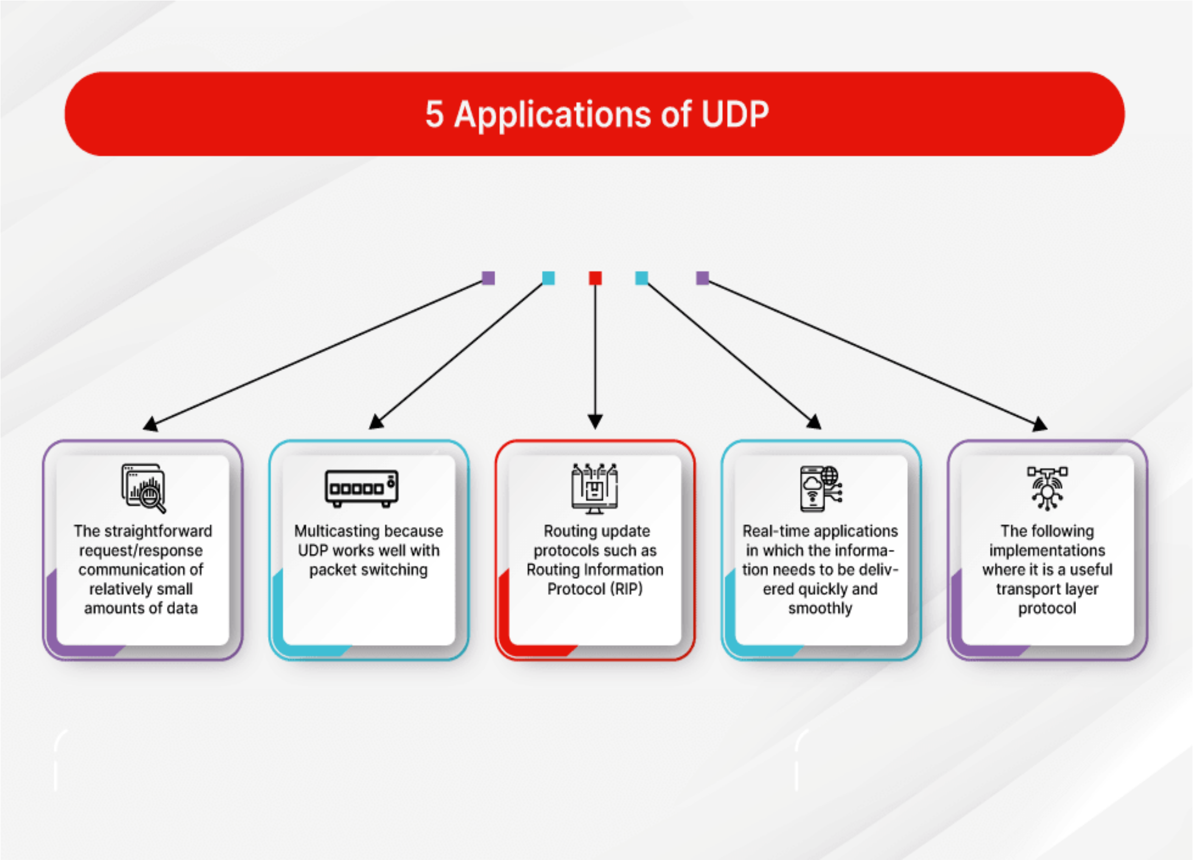
Key features

- Reliable
- Slower but accurate
- Connection-based

□ UDP (User Datagram Protocol)

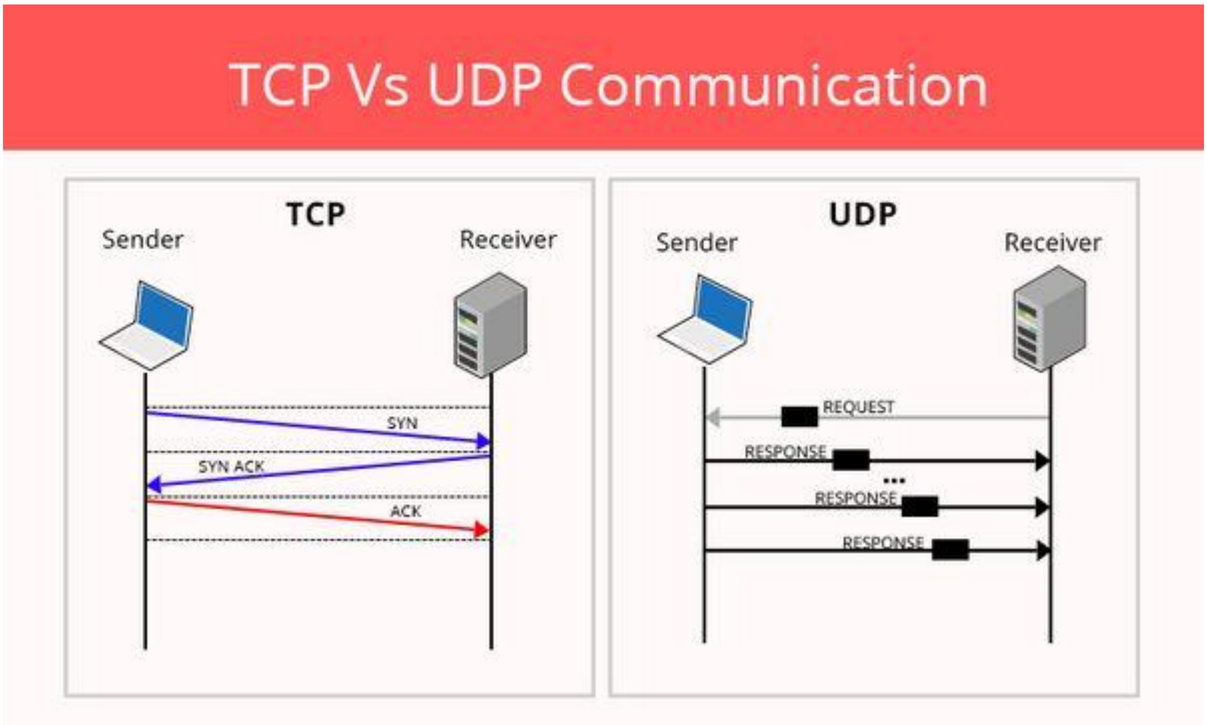
USER DATAGRAM PROTOCOL (UDP)





5 Applications of UDP

- The straightforward request/response communication of relatively small amounts of data
- Multicasting because UDP works well with packet switching
- Routing update protocols such as Routing Information Protocol (RIP)
- Real-time applications in which the information needs to be delivered quickly and smoothly
- The following implementations where it is a useful transport layer protocol



What it is

UDP is a **fast but unreliable communication method**.

Simple way to understand

UDP is like **shouting information**:

- No confirmation
- Some data may be missed
- Very fast

Used in

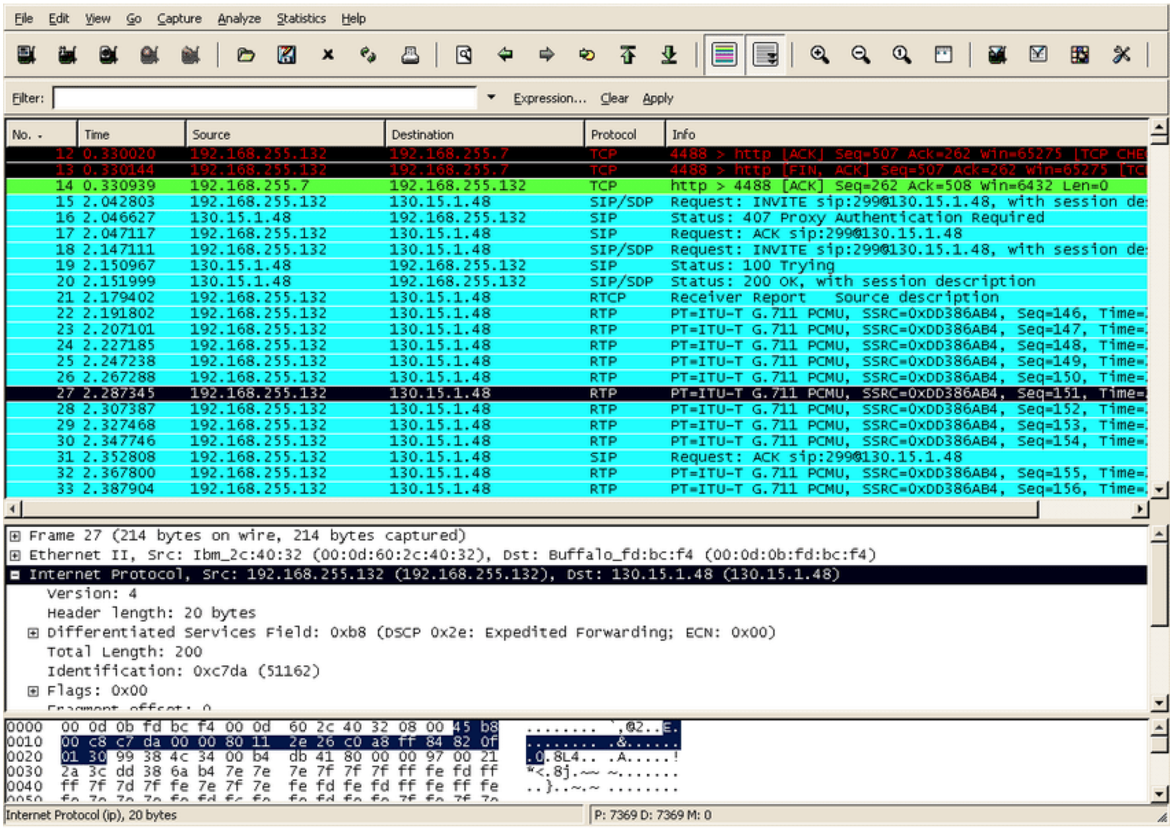
- Online games
- Video calls
- Live streaming

Key features

- Very fast
- No error checking
- Connection-less

.

2.Install Wireshark and Capture Live Network Traffic

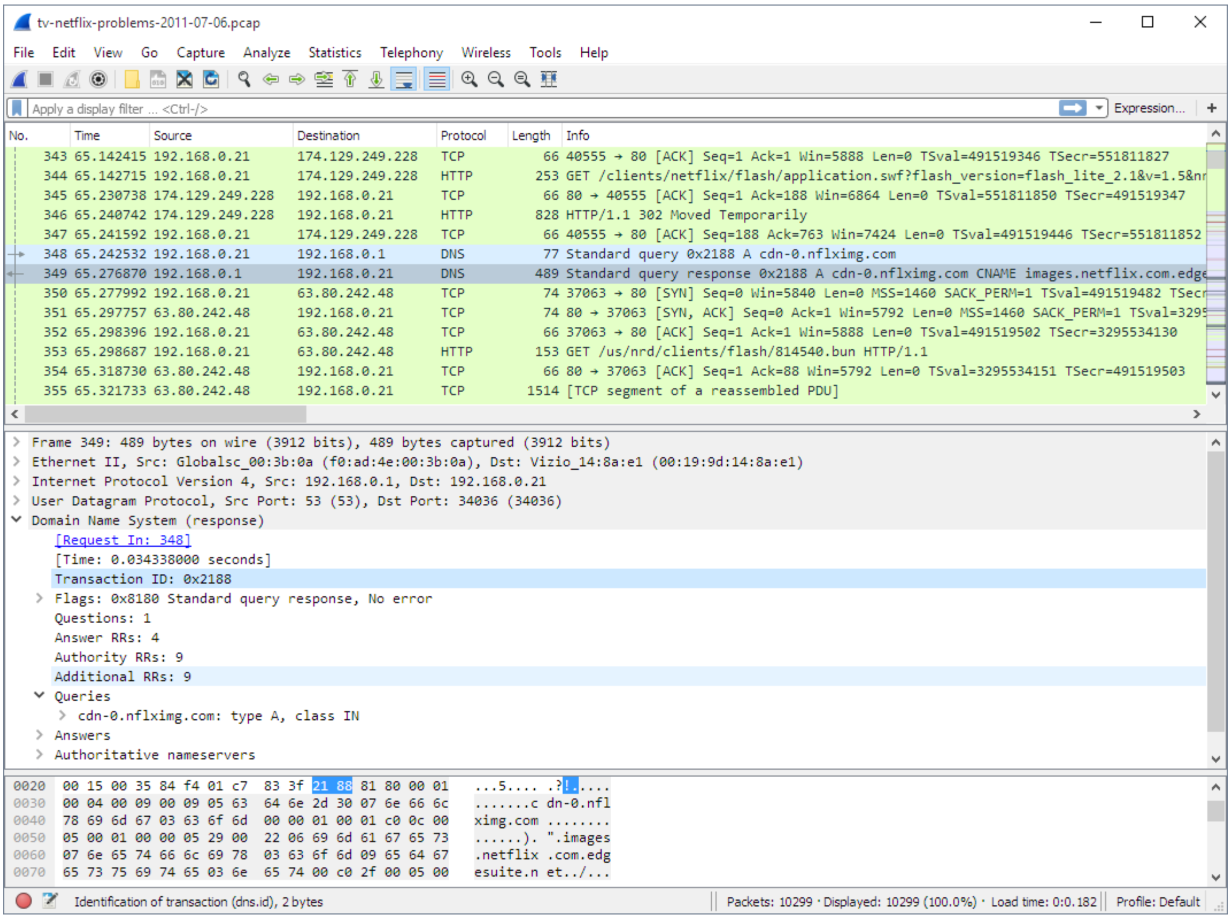


```
No. -  Time      Source            Destination       Protocol  Info
    12 0.330020  192.168.255.132   192.168.255.7     TCP       4488 > http [ACK] Seq=507 Ack=262 Win=65275 [TCP CHE
    13 0.330144  192.168.255.132   192.168.255.7     TCP       4488 > http [FIN, ACK] Seq=507 Ack=262 Win=65275 [TC
    14 0.330939  192.168.255.7     192.168.255.132   TCP       http > 4488 [ACK] Seq=262 Ack=508 Win=6432 Len=0
    15 2.042803  192.168.255.132   130.15.1.48       SIP/SDP   Request: INVITE sip:299@130.15.1.48, with session de
    16 2.046627  130.15.1.48       192.168.255.132   SIP       Status: 407 Proxy Authentication Required
    17 2.047117  192.168.255.132   130.15.1.48       SIP       Request: ACK sip:299@130.15.1.48
    18 2.147111  192.168.255.132   130.15.1.48       SIP/SDP   Request: INVITE sip:299@130.15.1.48, with session de
    19 2.150967  130.15.1.48       192.168.255.132   SIP       Status: 100 Trying
    20 2.151999  130.15.1.48       192.168.255.132   SIP/SDP   Status: 200 OK, with session description
    21 2.179402  192.168.255.132   130.15.1.48       RTCP      Receiver Report   Source description
    22 2.191802  192.168.255.132   130.15.1.48       RTP       PT=ITU-T G.711 PCMU, SSRC=0xDD386AB4, Seq=146, Time=
    23 2.207101  192.168.255.132   130.15.1.48       RTP       PT=ITU-T G.711 PCMU, SSRC=0xDD386AB4, Seq=147, Time=
    24 2.227185  192.168.255.132   130.15.1.48       RTP       PT=ITU-T G.711 PCMU, SSRC=0xDD386AB4, Seq=148, Time=
    25 2.247238  192.168.255.132   130.15.1.48       RTP       PT=ITU-T G.711 PCMU, SSRC=0xDD386AB4, Seq=149, Time=
    26 2.267288  192.168.255.132   130.15.1.48       RTP       PT=ITU-T G.711 PCMU, SSRC=0xDD386AB4, Seq=150, Time=
    27 2.287345  192.168.255.132   130.15.1.48       RTP       PT=ITU-T G.711 PCMU, SSRC=0xDD386AB4, Seq=151, Time=
    28 2.307387  192.168.255.132   130.15.1.48       RTP       PT=ITU-T G.711 PCMU, SSRC=0xDD386AB4, Seq=152, Time=
    29 2.327468  192.168.255.132   130.15.1.48       RTP       PT=ITU-T G.711 PCMU, SSRC=0xDD386AB4, Seq=153, Time=
    30 2.347746  192.168.255.132   130.15.1.48       RTP       PT=ITU-T G.711 PCMU, SSRC=0xDD386AB4, Seq=154, Time=
    31 2.352808  192.168.255.132   130.15.1.48       SIP       Request: ACK sip:299@130.15.1.48
    32 2.367800  192.168.255.132   130.15.1.48       RTP       PT=ITU-T G.711 PCMU, SSRC=0xDD386AB4, Seq=155, Time=
    33 2.387904  192.168.255.132   130.15.1.48       RTP       PT=ITU-T G.711 PCMU, SSRC=0xDD386AB4, Seq=156, Time=
```

```
⊞ Frame 27 (214 bytes on wire, 214 bytes captured)
⊞ Ethernet II, Src: Ibm_2c:40:32 (00:0d:60:2c:40:32), Dst: Buffalo_fd:bc:f4 (00:0d:0b:fd:bc:f4)
⊟ Internet Protocol, Src: 192.168.255.132 (192.168.255.132), Dst: 130.15.1.48 (130.15.1.48)
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0xb8 (DSCP 0x2e: Expedited Forwarding; ECN: 0x00)
    Total Length: 200
    Identification: 0xc7da (51162)
  ⊞ Flags: 0x00
    Fragment offset: 0
```

```
0000  00 0d 0b fd bc f4 00 0d  60 2c 40 32 08 00 45 b8   ........ `,@2..E.
0010  00 c8 c7 da 00 00 80 11  2e 26 c0 a8 ff 84 82 0f   ........ .&......
0020  01 30 99 38 4c 34 00 b4  db 41 80 00 00 97 00 21   .0.8L4.. .A.....!
0030  2a 3c dd 38 6a b4 7e 7e  7e 7f 7f 7f ff fe fd ff   *<.8j.~~ ~.......
0040  ff 7f 7d 7f fe 7e 7f 7e  fe fd fe fd ff fe ff fe   ..}..~.~ ........
```

```
Internet Protocol (ip), 20 bytes          P: 7369 D: 7369 M: 0
```


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

What this means

Wireshark is a **network packet analyzer** that allows you to **see live data packets** flowing through your network.

What you do

1. Install Wireshark
2. Open it
3. Select a **network interface** (Wi-Fi or Ethernet)
4. Click **Start Capture**

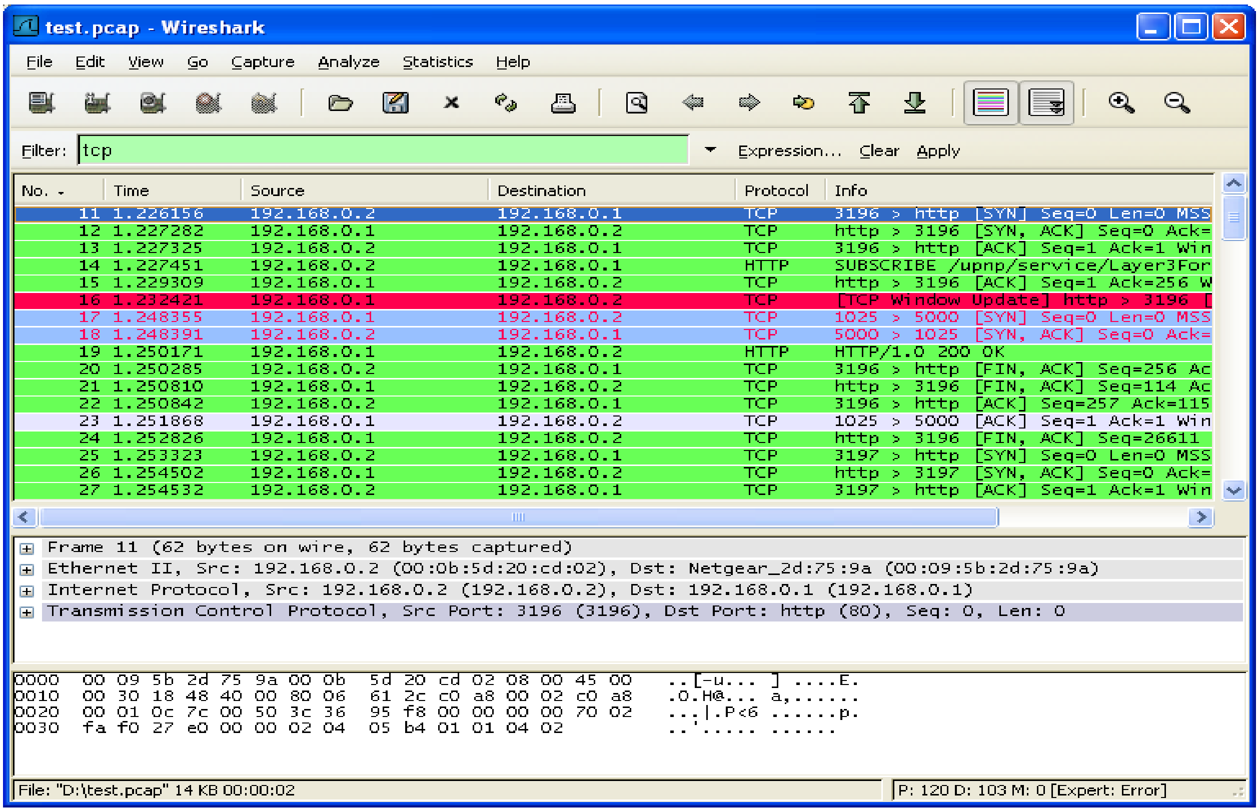
What happens

Wireshark begins showing **real-time packets** like:

- Website traffic
- DNS requests
- TCP connections

□ *This helps you understand how data moves in a network.*

3.Filter Packets by Protocol (HTTP, DNS, TCP)


test.pcap - Wireshark
File Edit View Go Capture Analyze Statistics Help
Expression... Clear Apply

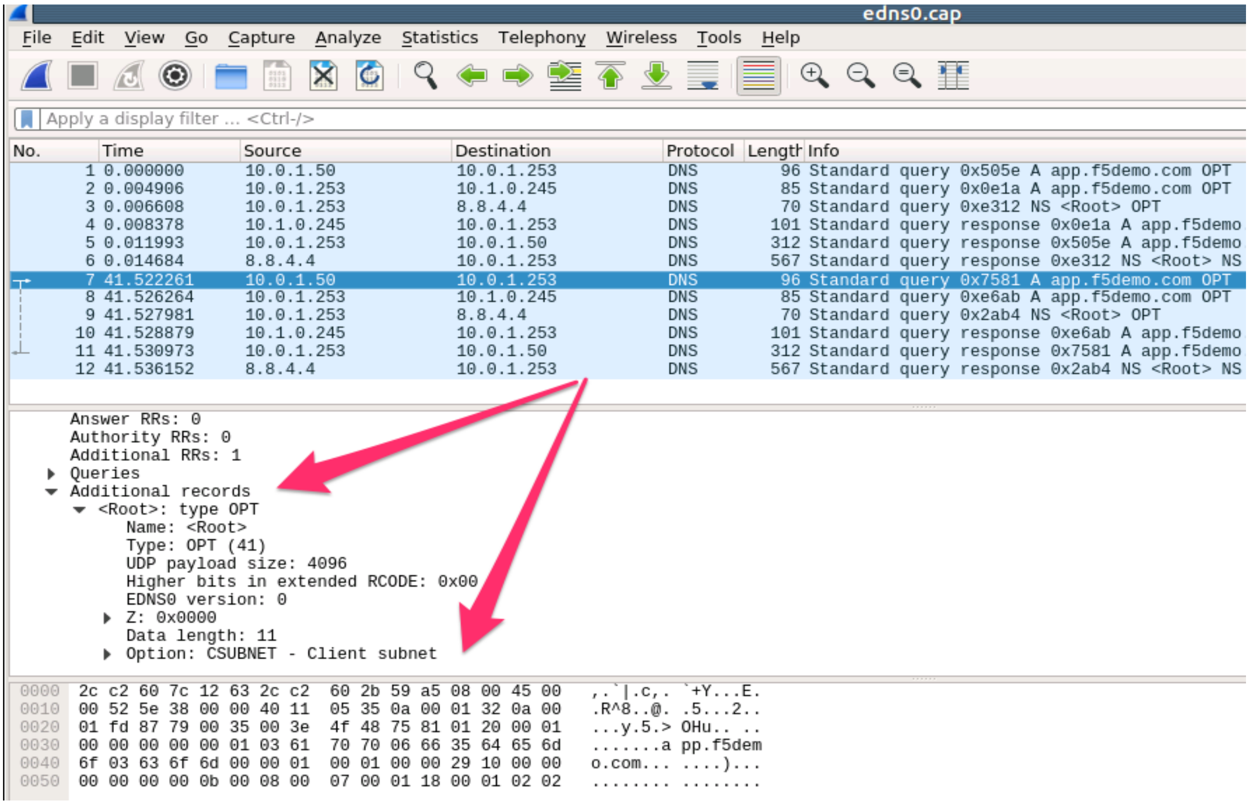
Filter: tcp

```
No. -  Time      Source       Destination  Protocol Info
    11 1.226156  192.168.0.2  192.168.0.1  TCP      3196 > http [SYN] Seq=0 Len=0 MSS
    12 1.227282  192.168.0.1  192.168.0.2  TCP      http > 3196 [SYN, ACK] Seq=0 Ack=
    13 1.227325  192.168.0.2  192.168.0.1  TCP      3196 > http [ACK] Seq=1 Ack=1 Win
    14 1.227451  192.168.0.2  192.168.0.1  HTTP     SUBSCRIBE /upnp/service/Layer3For
    15 1.229309  192.168.0.1  192.168.0.2  TCP      http > 3196 [ACK] Seq=1 Ack=256 W
    16 1.232421  192.168.0.1  192.168.0.2  TCP      [TCP Window Update] http > 3196 [
    17 1.248355  192.168.0.1  192.168.0.2  TCP      1025 > 5000 [SYN] Seq=0 Len=0 MSS
    18 1.248391  192.168.0.2  192.168.0.1  TCP      5000 > 1025 [SYN, ACK] Seq=0 Ack=
    19 1.250171  192.168.0.1  192.168.0.2  HTTP     HTTP/1.0 200 OK
    20 1.250285  192.168.0.2  192.168.0.1  TCP      3196 > http [FIN, ACK] Seq=256 Ac
    21 1.250810  192.168.0.1  192.168.0.2  TCP      http > 3196 [FIN, ACK] Seq=114 Ac
    22 1.250842  192.168.0.2  192.168.0.1  TCP      3196 > http [ACK] Seq=257 Ack=115
    23 1.251868  192.168.0.1  192.168.0.2  TCP      1025 > 5000 [ACK] Seq=1 Ack=1 Win
    24 1.252826  192.168.0.1  192.168.0.2  TCP      http > 3196 [FIN, ACK] Seq=26611
    25 1.253323  192.168.0.2  192.168.0.1  TCP      3197 > http [SYN] Seq=0 Len=0 MSS
    26 1.254502  192.168.0.1  192.168.0.2  TCP      http > 3197 [SYN, ACK] Seq=0 Ack=
    27 1.254532  192.168.0.2  192.168.0.1  TCP      3197 > http [ACK] Seq=1 Ack=1 Win
```

```
⊞ Frame 11 (62 bytes on wire, 62 bytes captured)
⊞ Ethernet II, Src: 192.168.0.2 (00:0b:5d:20:cd:02), Dst: Netgear_2d:75:9a (00:09:5b:2d:75:9a)
⊞ Internet Protocol, Src: 192.168.0.2 (192.168.0.2), Dst: 192.168.0.1 (192.168.0.1)
⊞ Transmission Control Protocol, Src Port: 3196 (3196), Dst Port: http (80), Seq: 0, Len: 0
```

```
0000  00 09 5b 2d 75 9a 00 0b  5d 20 cd 02 08 00 45 00   ..[-u... ] ....E.
0010  00 30 18 48 40 00 80 06  61 2c c0 a8 00 02 c0 a8   .0.H@... a,......
0020  00 01 0c 7c 00 50 3c 36  95 f8 00 00 00 00 70 02   ...|.P<6 ......p.
0030  fa f0 27 e0 00 00 02 04  05 b4 01 01 04 02         ..'..... ......
```

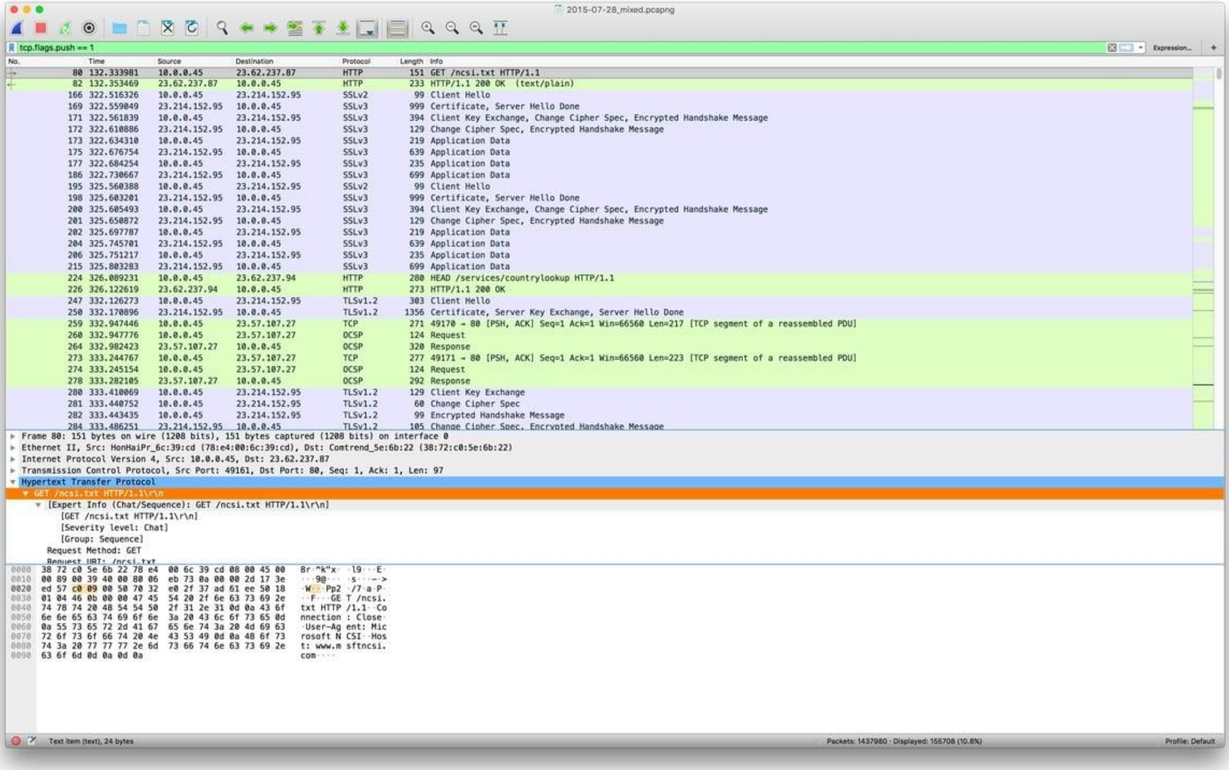
File: "D:\test.pcap" 14 KB 00:00:02 | P: 120 D: 103 M: 0 [Expert: Error]


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   ........ ........
```



```
tcp.flags.push == 1

No.     Time        Source          Destination     Protocol  Length Info
     80 132.333981  10.0.0.45       23.62.237.87    HTTP       151 GET /ncsi.txt HTTP/1.1
     82 132.353469  23.62.237.87    10.0.0.45       HTTP       233 HTTP/1.1 200 OK  (text/plain)
    166 322.516326  10.0.0.45       23.214.152.95   SSLv2       99 Client Hello
    169 322.559049  23.214.152.95   10.0.0.45       SSLv3      999 Certificate, Server Hello Done
    171 322.561839  10.0.0.45       23.214.152.95   SSLv3      394 Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
    172 322.610886  23.214.152.95   10.0.0.45       SSLv3      129 Change Cipher Spec, Encrypted Handshake Message
    173 322.634310  10.0.0.45       23.214.152.95   SSLv3      219 Application Data
    175 322.676754  23.214.152.95   10.0.0.45       SSLv3      639 Application Data
    177 322.684254  10.0.0.45       23.214.152.95   SSLv3      235 Application Data
    186 322.730667  23.214.152.95   10.0.0.45       SSLv3      699 Application Data
    195 325.560388  10.0.0.45       23.214.152.95   SSLv2       99 Client Hello
    198 325.603201  23.214.152.95   10.0.0.45       SSLv3      999 Certificate, Server Hello Done
    200 325.605493  10.0.0.45       23.214.152.95   SSLv3      394 Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
    201 325.650872  23.214.152.95   10.0.0.45       SSLv3      129 Change Cipher Spec, Encrypted Handshake Message
    202 325.697787  10.0.0.45       23.214.152.95   SSLv3      219 Application Data
    204 325.745701  23.214.152.95   10.0.0.45       SSLv3      639 Application Data
    206 325.751217  10.0.0.45       23.214.152.95   SSLv3      235 Application Data
    215 325.803283  23.214.152.95   10.0.0.45       SSLv3      699 Application Data
    224 326.089231  10.0.0.45       23.62.237.94    HTTP       280 HEAD /services/countrylookup HTTP/1.1
    226 326.122619  23.62.237.94    10.0.0.45       HTTP       273 HTTP/1.1 200 OK
    247 332.126273  10.0.0.45       23.214.152.95   TLSv1.2    303 Client Hello
    250 332.170896  23.214.152.95   10.0.0.45       TLSv1.2   1356 Certificate, Server Key Exchange, Server Hello Done
    259 332.947446  10.0.0.45       23.57.107.27    TCP        271 49170 → 80 [PSH, ACK] Seq=1 Ack=1 Win=66560 Len=217 [TCP segment of a reassembled PDU]
    260 332.947776  10.0.0.45       23.57.107.27    OCSP       124 Request
    264 332.982423  23.57.107.27    10.0.0.45       OCSP       320 Response
    273 333.244767  10.0.0.45       23.57.107.27    TCP        277 49171 → 80 [PSH, ACK] Seq=1 Ack=1 Win=66560 Len=223 [TCP segment of a reassembled PDU]
    274 333.245154  10.0.0.45       23.57.107.27    OCSP       124 Request
    278 333.282105  23.57.107.27    10.0.0.45       OCSP       292 Response
    280 333.410069  10.0.0.45       23.214.152.95   TLSv1.2    129 Client Key Exchange
    281 333.440752  10.0.0.45       23.214.152.95   TLSv1.2     60 Change Cipher Spec
    282 333.443435  10.0.0.45       23.214.152.95   TLSv1.2     99 Encrypted Handshake Message
    284 333.486251  23.214.152.95   10.0.0.45       TLSv1.2    105 Change Cipher Spec, Encrypted Handshake Message

▶ Frame 80: 151 bytes on wire (1208 bits), 151 bytes captured (1208 bits) on interface 0
▶ Ethernet II, Src: HonHaiPr_6c:39:cd (78:e4:00:6c:39:cd), Dst: Comtrend_5e:6b:22 (38:72:c0:5e:6b:22)
▶ Internet Protocol Version 4, Src: 10.0.0.45, Dst: 23.62.237.87
▶ Transmission Control Protocol, Src Port: 49161, Dst Port: 80, Seq: 1, Ack: 1, Len: 97
▼ Hypertext Transfer Protocol
  ▼ GET /ncsi.txt HTTP/1.1\r\n
    ▼ [Expert Info (Chat/Sequence): GET /ncsi.txt HTTP/1.1\r\n]
        [GET /ncsi.txt HTTP/1.1\r\n]
        [Severity level: Chat]
        [Group: Sequence]
      Request Method: GET
      Request URI: /ncsi.txt

0000  38 72 c0 5e 6b 22 78 e4  00 6c 39 cd 08 00 45 00   8r·^k"x· ·l9···E·
0010  00 89 00 39 40 00 80 06  eb 73 0a 00 00 2d 17 3e   ···9@··· ·s···-·>
0020  ed 57 c0 09 00 50 70 32  e0 2f 37 ad 61 ee 50 18   ·W···Pp2 ·/7·a·P·
0030  01 04 46 0b 00 00 47 45  54 20 2f 6e 63 73 69 2e   ··F···GE T /ncsi.
0040  74 78 74 20 48 54 54 50  2f 31 2e 31 0d 0a 43 6f   txt HTTP /1.1··Co
0050  6e 6e 65 63 74 69 6f 6e  3a 20 43 6c 6f 73 65 0d   nnection : Close·
0060  0a 55 73 65 72 2d 41 67  65 6e 74 3a 20 4d 69 63   ·User-Ag ent: Mic
0070  72 6f 73 6f 66 74 20 4e  43 53 49 0d 0a 48 6f 73   rosoft N CSI··Hos
0080  74 3a 20 77 77 77 2e 6d  73 66 74 6e 63 73 69 2e   t: www.m sftncsi.
0090  63 6f 6d 0d 0a 0d 0a                               com····

Text item (text), 24 bytes          Packets: 1437980 · Displayed: 155708 (10.8%)          Profile: Default
```

What filtering means

Filtering helps you **view only the packets you want**, instead of thousands of packets.

Common filters

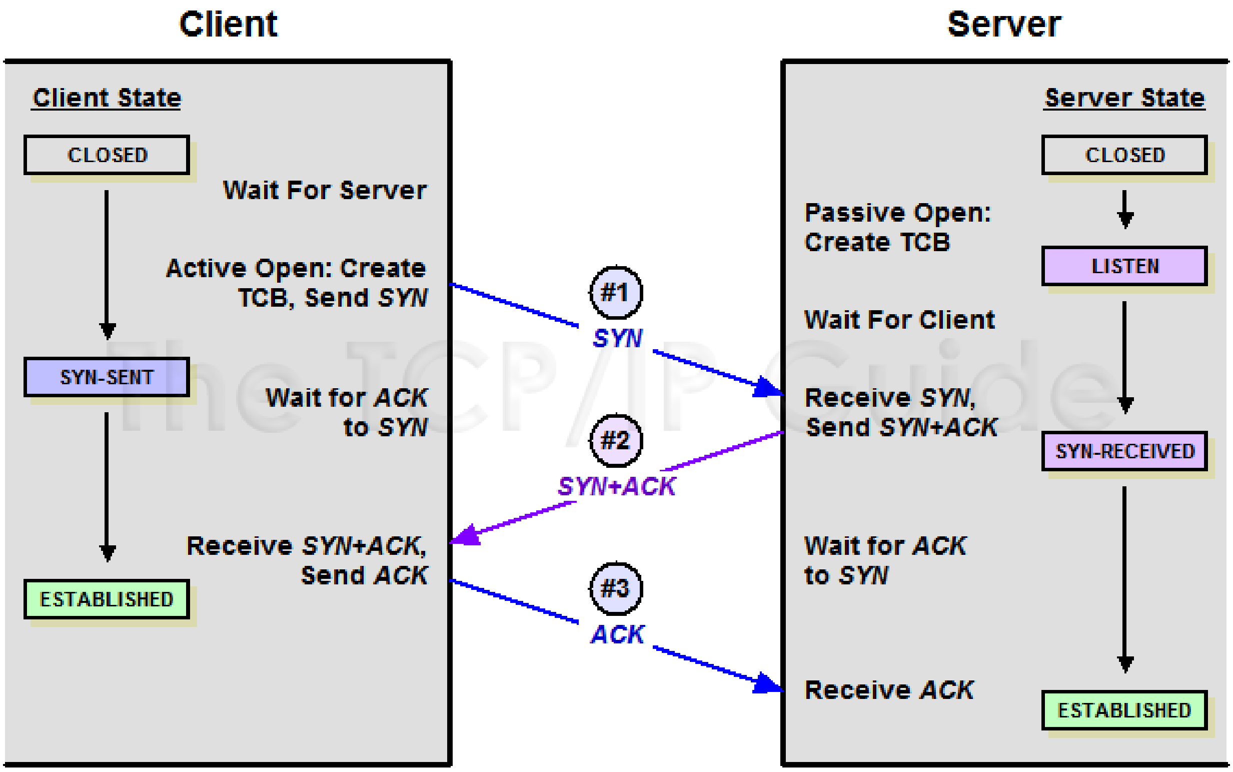
- `http` → Shows web traffic
- `dns` → Shows DNS queries
- `tcp` → Shows TCP packets

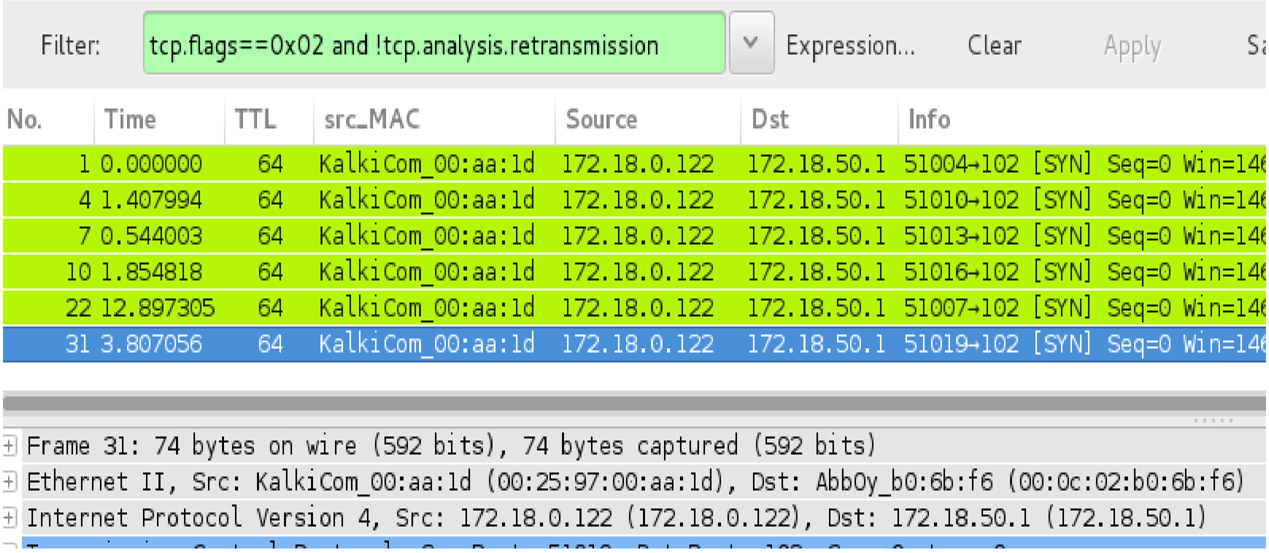
Why it’s important

- Saves time
- Helps focus on **specific protocols**
- Makes analysis easy

□ *Filters do not delete packets, they only hide unwanted ones.*

4.Observe the Three-Way TCP Handshake

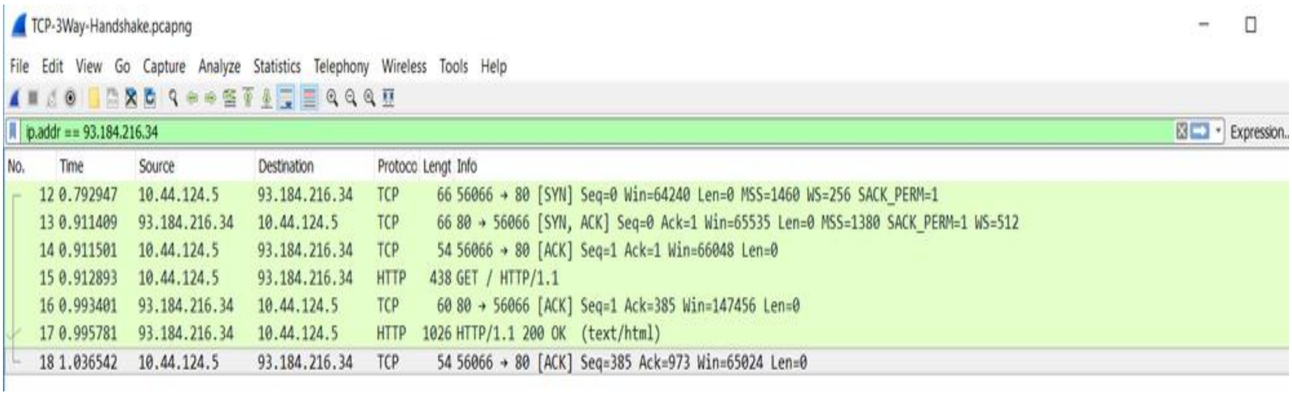




tcp.flags==0x02 and !tcp.analysis.retransmission

No.	Time	TTL	src_MAC	Source	Dst	Info
1	0.000000	64	KalkiCom_00:aa:1d	172.18.0.122	172.18.50.1	51004→102 [SYN] Seq=0 Win=146
4	1.407994	64	KalkiCom_00:aa:1d	172.18.0.122	172.18.50.1	51010→102 [SYN] Seq=0 Win=146
7	0.544003	64	KalkiCom_00:aa:1d	172.18.0.122	172.18.50.1	51013→102 [SYN] Seq=0 Win=146
10	1.854818	64	KalkiCom_00:aa:1d	172.18.0.122	172.18.50.1	51016→102 [SYN] Seq=0 Win=146
22	12.897305	64	KalkiCom_00:aa:1d	172.18.0.122	172.18.50.1	51007→102 [SYN] Seq=0 Win=146
31	3.807056	64	KalkiCom_00:aa:1d	172.18.0.122	172.18.50.1	51019→102 [SYN] Seq=0 Win=146

```
⊞ Frame 31: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)
⊞ Ethernet II, Src: KalkiCom_00:aa:1d (00:25:97:00:aa:1d), Dst: AbbOy_b0:6b:f6 (00:0c:02:b0:6b:f6)
⊞ Internet Protocol Version 4, Src: 172.18.0.122 (172.18.0.122), Dst: 172.18.50.1 (172.18.50.1)
```


TCP-3Way-Handshake.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression...

ip.addr == 93.184.216.34

No.	Time	Source	Destination	Protoco	Lengt	Info
12	0.792947	10.44.124.5	93.184.216.34	TCP	66	56066 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
13	0.911409	93.184.216.34	10.44.124.5	TCP	66	80 → 56066 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1380 SACK_PERM=1 WS=512
14	0.911501	10.44.124.5	93.184.216.34	TCP	54	56066 → 80 [ACK] Seq=1 Ack=1 Win=66048 Len=0
15	0.912893	10.44.124.5	93.184.216.34	HTTP	438	GET / HTTP/1.1
16	0.993401	93.184.216.34	10.44.124.5	TCP	60	80 → 56066 [ACK] Seq=1 Ack=385 Win=147456 Len=0
17	0.995781	93.184.216.34	10.44.124.5	HTTP	1026	HTTP/1.1 200 OK (text/html)
18	1.036542	10.44.124.5	93.184.216.34	TCP	54	56066 → 80 [ACK] Seq=385 Ack=973 Win=65024 Len=0

What it is

The **TCP three-way handshake** is how two devices **establish a reliable connection**.

The three steps

1. **SYN** → Client asks to connect
2. **SYN-ACK** → Server agrees
3. **ACK** → Connection confirmed

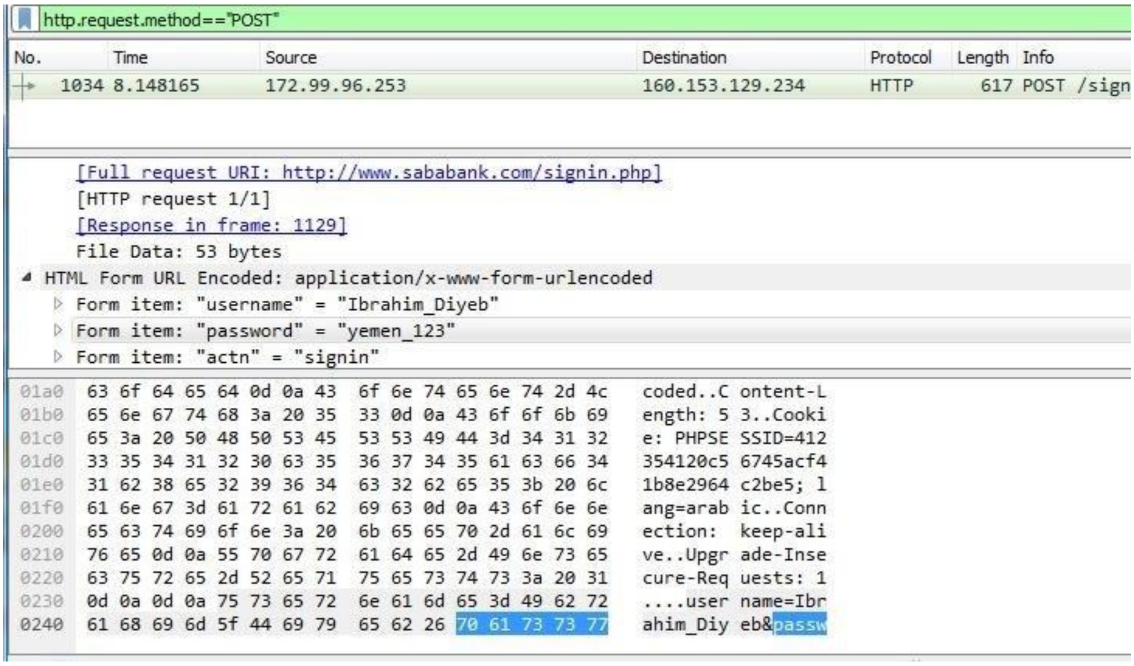
What you see in Wireshark

Packets with flags:

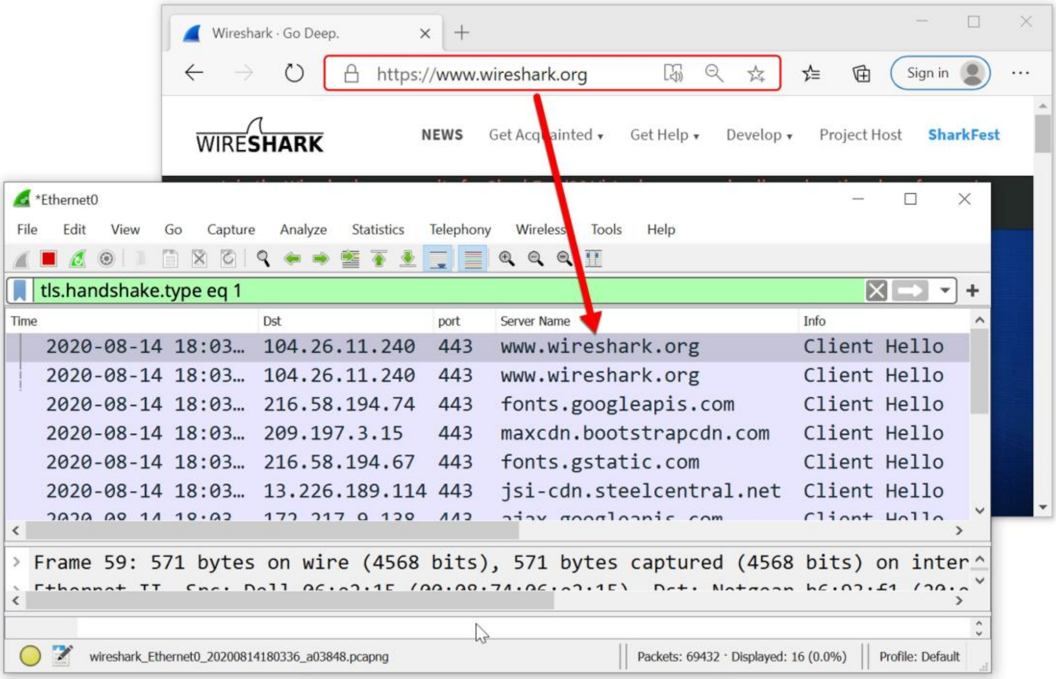
- SYN
- SYN, ACK
- ACK

□ *This proves a TCP connection is successfully established.*

5.Identify Plain-Text Traffic vs Encrypted Traffic



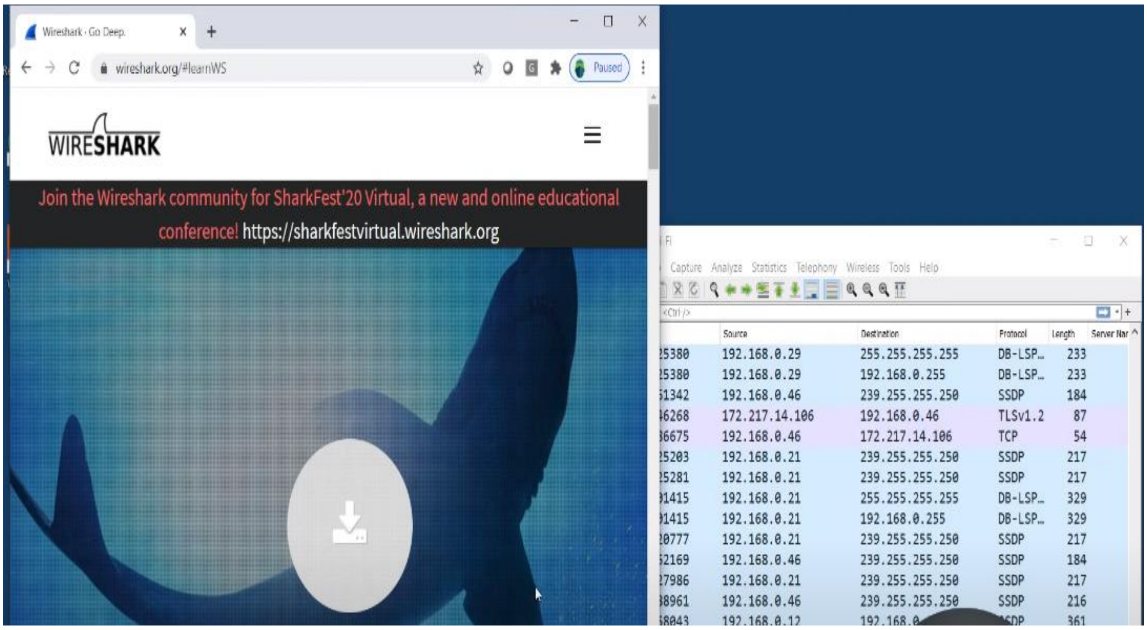
```
http.request.method=="POST"

No.     Time        Source            Destination        Protocol  Length  Info
  1034  8.148165    172.99.96.253     160.153.129.234    HTTP         617  POST /sign

      [Full request URI: http://www.sababank.com/signin.php]
      [HTTP request 1/1]
      [Response in frame: 1129]
      File Data: 53 bytes
  ◢ HTML Form URL Encoded: application/x-www-form-urlencoded
    ▷ Form item: "username" = "Ibrahim_Diyeb"
    ▷ Form item: "password" = "yemen_123"
    ▷ Form item: "actn" = "signin"

01a0  63 6f 64 65 64 0d 0a 43  6f 6e 74 65 6e 74 2d 4c   coded..C ontent-L
01b0  65 6e 67 74 68 3a 20 35  33 0d 0a 43 6f 6f 6b 69   ength: 5 3..Cooki
01c0  65 3a 20 50 48 50 53 45  53 53 49 44 3d 34 31 32   e: PHPSE SSID=412
01d0  33 35 34 31 32 30 63 35  36 37 34 35 61 63 66 34   354120c5 6745acf4
01e0  31 62 38 65 32 39 36 34  63 32 62 65 35 3b 20 6c   1b8e2964 c2be5; l
01f0  61 6e 67 3d 61 72 61 62  69 63 0d 0a 43 6f 6e 6e   ang=arab ic..Conn
0200  65 63 74 69 6f 6e 3a 20  6b 65 65 70 2d 61 6c 69   ection:  keep-ali
0210  76 65 0d 0a 55 70 67 72  61 64 65 2d 49 6e 73 65   ve..Upgr ade-Inse
0220  63 75 72 65 2d 52 65 71  75 65 73 74 73 3a 20 31   cure-Req uests: 1
0230  0d 0a 0d 0a 75 73 65 72  6e 61 6d 65 3d 49 62 72   ....user name=Ibr
0240  61 68 69 6d 5f 44 69 79  65 62 26 70 61 73 73 77   ahim_Diy eb&passw
```



tls.handshake.type eq 1

```
Time                 Dst             port  Server Name               Info
2020-08-14 18:03…   104.26.11.240   443   www.wireshark.org         Client Hello
2020-08-14 18:03…   104.26.11.240   443   www.wireshark.org         Client Hello
2020-08-14 18:03…   216.58.194.74   443   fonts.googleapis.com      Client Hello
2020-08-14 18:03…   209.197.3.15    443   maxcdn.bootstrapcdn.com   Client Hello
2020-08-14 18:03…   216.58.194.67   443   fonts.gstatic.com         Client Hello
2020-08-14 18:03…   13.226.189.114  443   jsi-cdn.steelcentral.net  Client Hello
2020-08-14 18:03…   172.217.9.138   443   ajax.googleapis.com       Client Hello

> Frame 59: 571 bytes on wire (4568 bits), 571 bytes captured (4568 bits) on inter
```

wireshark_Ethernet0_20200814180336_a03848.pcapng | Packets: 69432 · Displayed: 16 (0.0%) | Profile: Default



Join the Wireshark community for SharkFest'20 Virtual, a new and online educational conference! https://sharkfestvirtual.wireshark.org

```
        Source          Destination      Protocol  Length  Server Nar
25380   192.168.0.29    255.255.255.255  DB-LSP…   233
25380   192.168.0.29    192.168.0.255    DB-LSP…   233
51342   192.168.0.46    239.255.255.250  SSDP      184
46268   172.217.14.106  192.168.0.46     TLSv1.2    87
36675   192.168.0.46    172.217.14.106   TCP        54
25203   192.168.0.21    239.255.255.250  SSDP      217
25281   192.168.0.21    239.255.255.250  SSDP      217
91415   192.168.0.21    255.255.255.255  DB-LSP…   329
91415   192.168.0.21    192.168.0.255    DB-LSP…   329
20777   192.168.0.21    239.255.255.250  SSDP      217
52169   192.168.0.46    239.255.255.250  SSDP      184
27986   192.168.0.21    239.255.255.250  SSDP      217
38961   192.168.0.46    239.255.255.250  SSDP      216
58043   192.168.0.12    192.168.0        SSDP      361
```

Plain-text traffic

- Data is **readable**
- Example: HTTP
- You can see usernames, URLs, data

Encrypted traffic

- Data is **scrambled**
- Example: HTTPS
- Content is unreadable

Why this matters

- Plain-text traffic is **unsafe**
- Encrypted traffic protects **confidential data**

□ *Cyber attackers target plain-text traffic.*

---

6. Capture DNS Queries and Analyze Them

What DNS capture shows

DNS packets show:

- Which website is requested
- DNS server response
- IP address returned

Example

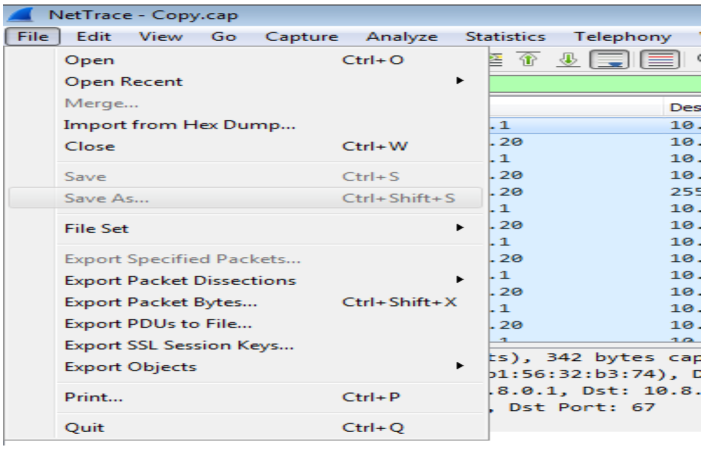
```
Query: www.google.com
Response: 142.250.190.14
```

Why DNS analysis is useful

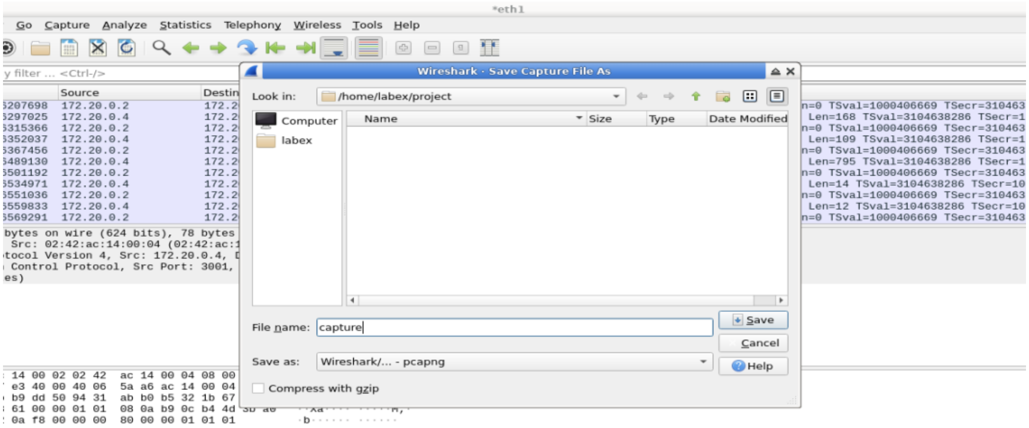
- Detect suspicious domains
- Identify DNS spoofing
- Track browsing behavior

□ *DNS traffic reveals where a system is trying to connect.*

7.Save Packet Captures for Analysis



```
ts), 342 bytes cap
o1:56:32:b3:74), D
.8.0.1, Dst: 10.8.
, Dst Port: 67
```


*eth1
Go Capture Analyze Statistics Telephony Wireless Tools Help
y filter ... <Ctrl-/>
Wireshark · Save Capture File As
Computer
labex
Name
Size
Type
Date Modified
Save
Cancel
Help

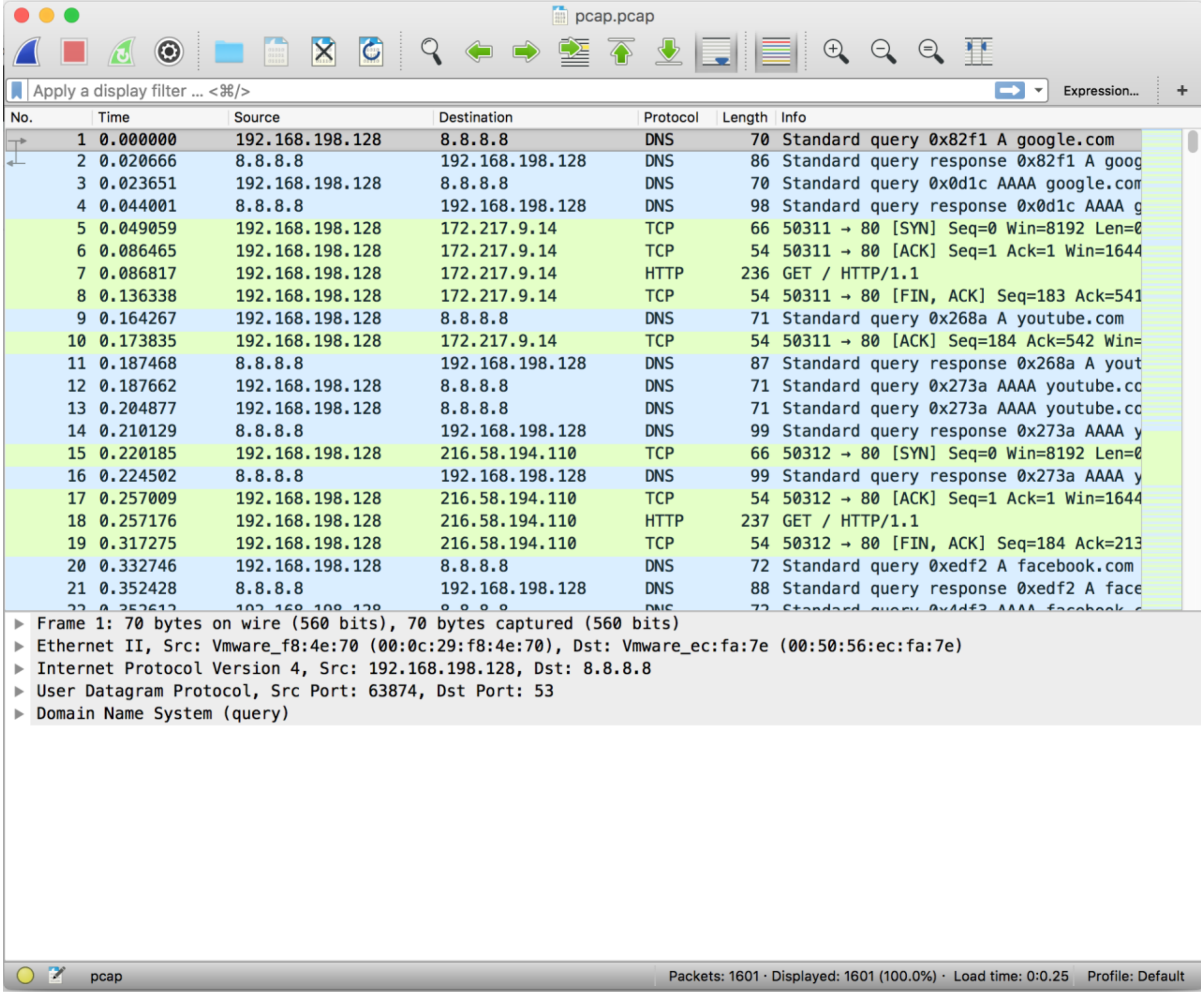
```
      Source       Destin
5207698   172.20.0.2   172.2   n=0 TSval=1000406669 TSecr=310463
5297025   172.20.0.4   172.2    Len=168 TSval=3104638286 TSecr=1
5315366   172.20.0.2   172.2   n=0 TSval=1000406669 TSecr=310463
5352037   172.20.0.4   172.2    Len=109 TSval=3104638286 TSecr=1
5367456   172.20.0.2   172.2   n=0 TSval=1000406669 TSecr=310463
5489130   172.20.0.4   172.2    Len=795 TSval=3104638286 TSecr=1
5501192   172.20.0.2   172.2   n=0 TSval=1000406669 TSecr=310463
5534971   172.20.0.4   172.2    Len=14 TSval=3104638286 TSecr=10
5551036   172.20.0.2   172.2   n=0 TSval=1000406669 TSecr=310463
5559833   172.20.0.4   172.2    Len=12 TSval=3104638286 TSecr=10
5569291   172.20.0.2   172.2   n=0 TSval=1000406669 TSecr=310463
bytes on wire (624 bits), 78 bytes
 Src: 02:42:ac:14:00:04 (02:42:ac:1
tocol Version 4, Src: 172.20.0.4, D
 Control Protocol, Src Port: 3001,
es)

 14 00 02 02 42  ac 14 00 04 08 00
 e3 40 00 40 06  5a a6 ac 14 00 04
 b9 dd 50 94 31  ab b0 b5 32 1b 67
 61 00 00 01 01  08 0a b9 0c b4 4d
 0a f8 00 00 00  80 00 00 01 01 01
```

Look in: /home/labex/project

File name: capture

Save as: Wireshark/... - pcapng

☐ Compress with gzip



```
No.  Time      Source           Destination      Protocol Length Info
   1 0.000000  192.168.198.128  8.8.8.8          DNS        70 Standard query 0x82f1 A google.com
   2 0.020666  8.8.8.8          192.168.198.128  DNS        86 Standard query response 0x82f1 A goog
   3 0.023651  192.168.198.128  8.8.8.8          DNS        70 Standard query 0x0d1c AAAA google.com
   4 0.044001  8.8.8.8          192.168.198.128  DNS        98 Standard query response 0x0d1c AAAA g
   5 0.049059  192.168.198.128  172.217.9.14     TCP        66 50311 → 80 [SYN] Seq=0 Win=8192 Len=0
   6 0.086465  192.168.198.128  172.217.9.14     TCP        54 50311 → 80 [ACK] Seq=1 Ack=1 Win=1644
   7 0.086817  192.168.198.128  172.217.9.14     HTTP      236 GET / HTTP/1.1
   8 0.136338  192.168.198.128  172.217.9.14     TCP        54 50311 → 80 [FIN, ACK] Seq=183 Ack=541
   9 0.164267  192.168.198.128  8.8.8.8          DNS        71 Standard query 0x268a A youtube.com
  10 0.173835  192.168.198.128  172.217.9.14     TCP        54 50311 → 80 [ACK] Seq=184 Ack=542 Win=
  11 0.187468  8.8.8.8          192.168.198.128  DNS        87 Standard query response 0x268a A yout
  12 0.187662  192.168.198.128  8.8.8.8          DNS        71 Standard query 0x273a AAAA youtube.cc
  13 0.204877  192.168.198.128  8.8.8.8          DNS        71 Standard query 0x273a AAAA youtube.cc
  14 0.210129  8.8.8.8          192.168.198.128  DNS        99 Standard query response 0x273a AAAA y
  15 0.220185  192.168.198.128  216.58.194.110   TCP        66 50312 → 80 [SYN] Seq=0 Win=8192 Len=0
  16 0.224502  8.8.8.8          192.168.198.128  DNS        99 Standard query response 0x273a AAAA y
  17 0.257009  192.168.198.128  216.58.194.110   TCP        54 50312 → 80 [ACK] Seq=1 Ack=1 Win=1644
  18 0.257176  192.168.198.128  216.58.194.110   HTTP      237 GET / HTTP/1.1
  19 0.317275  192.168.198.128  216.58.194.110   TCP        54 50312 → 80 [FIN, ACK] Seq=184 Ack=213
  20 0.332746  192.168.198.128  8.8.8.8          DNS        72 Standard query 0xedf2 A facebook.com
  21 0.352428  8.8.8.8          192.168.198.128  DNS        88 Standard query response 0xedf2 A face

▶ Frame 1: 70 bytes on wire (560 bits), 70 bytes captured (560 bits)
▶ Ethernet II, Src: Vmware_f8:4e:70 (00:0c:29:f8:4e:70), Dst: Vmware_ec:fa:7e (00:50:56:ec:fa:7e)
▶ Internet Protocol Version 4, Src: 192.168.198.128, Dst: 8.8.8.8
▶ User Datagram Protocol, Src Port: 63874, Dst Port: 53
▶ Domain Name System (query)
```

pcap

Packets: 1601 · Displayed: 1601 (100.0%) · Load time: 0:0.25 | Profile: Default

What saving means

Saving captures allows you to **analyze traffic later**.

File format

- `.pcap` or `.pcapng`

Why save captures

- For reports
- For incident investigation
- For learning and practice

□ *Saved files can be reopened anytime in Wireshark.*

---

8. Write Observations in Simple Language

What observations are

Observations explain **what you saw** during the capture.

Example observations

- “DNS queries were observed for google.com”
- “TCP handshake completed successfully”
- “HTTP traffic was visible in plain text”
- “HTTPS traffic was encrypted”

Why this is important

- Helps in documentation
- Useful for lab records
- Important for SOC and VAPT roles

□ *Always write observations clearly and simply.*