Packet Filtering and Analysis using Wireshark/Burp Suite

Aim

To capture, filter, and analyze network packets using Wireshark on Kali Linux in VirtualBox.

Tools Required

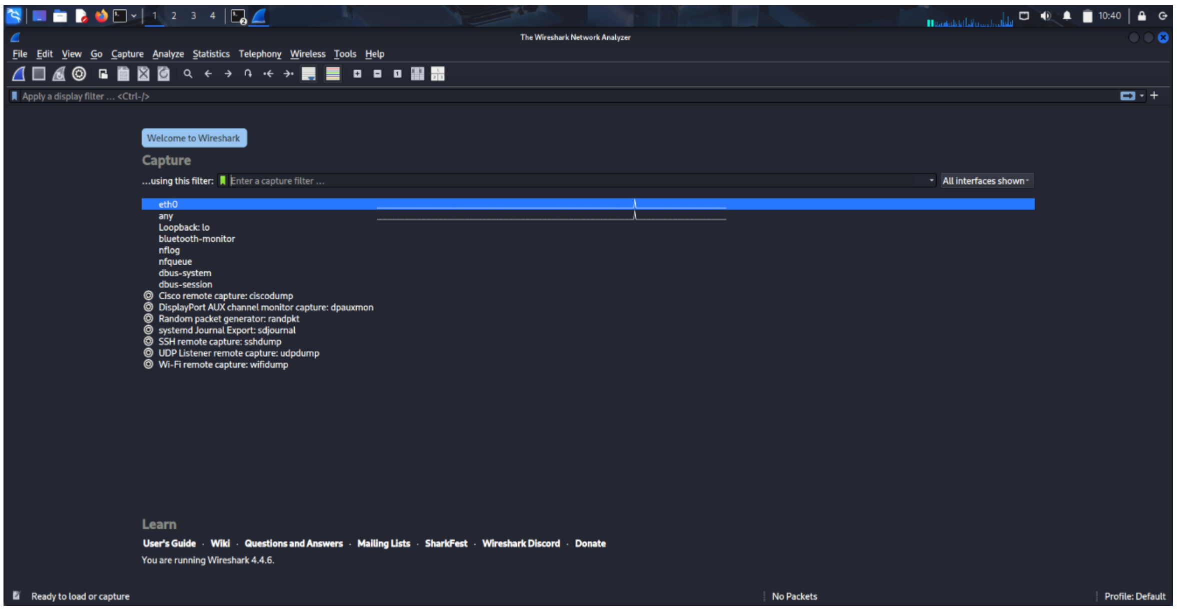
- **Hardware:** Laptop/PC with VirtualBox installed
- **Software:**
  - Kali Linux (running inside VirtualBox)
  - Wireshark Network Analyzer
  - Apache2 web server
  - PHP interpreter
  - Browser

Procedure

1. **Configure VirtualBox Network for Packet Capture**
   - Open VirtualBox → Kali VM → *Settings* → *Network*.
   - Set **Adapter 1**:
     - Attached to: **Bridged Adapter**
     - Promiscuous Mode: **Allow All**
     - Cable Connected: ✓
   - Save settings.
2. **Start Kali Linux VM**
   - Boot the Kali Linux VM in VirtualBox.
3. **Run Wireshark as Root**

   ```
   sudo wireshark
   ```

   - Select network interface `eth0`
4. **Start Packet Capture**
   - Click on `eth0` to begin capturing packets.
   - Generate some traffic (open websites, ping servers, or run curl commands).

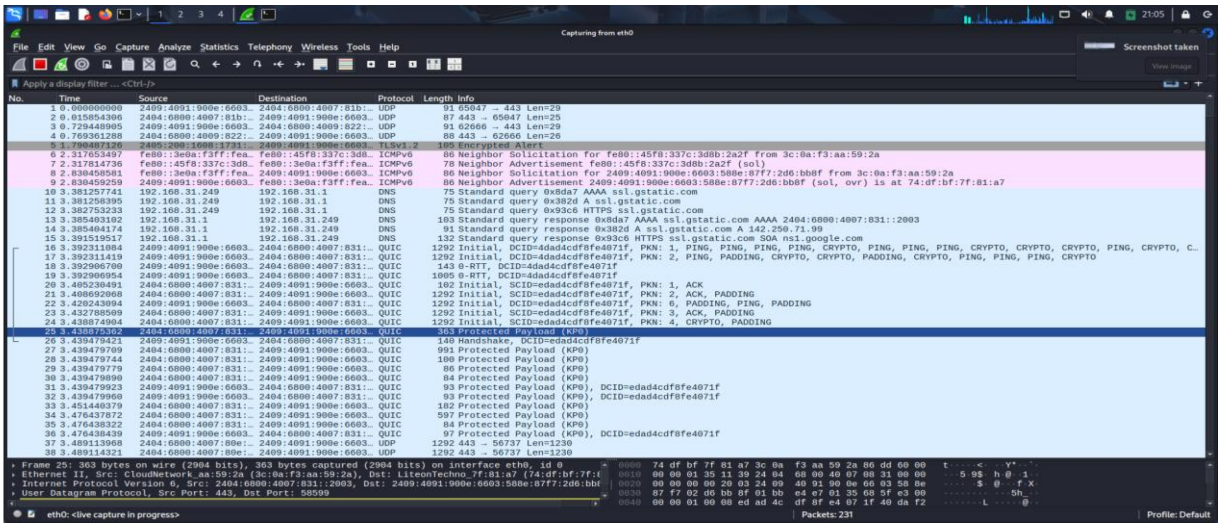

The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
Welcome to Wireshark
Capture
Learn
User's Guide · Wiki · Questions and Answers · Mailing Lists · SharkFest · Wireshark Discord · Donate

…using this filter: Enter a capture filter … All interfaces shown

eth0
any
Loopback: lo
bluetooth-monitor
nflog
nfqueue
dbus-system
dbus-session
Cisco remote capture: ciscodump
DisplayPort AUX channel monitor capture: dpauxmon
Random packet generator: randpkt
systemd Journal Export: sdjournal
SSH remote capture: sshdump
UDP Listener remote capture: udpdump
Wi-Fi remote capture: wifidump

You are running Wireshark 4.4.6.

Ready to load or capture | No Packets | Profile: Default


Capturing from eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Screenshot taken
View image
Apply a display filter … <Ctrl-/>

```
No.   Time         Source                  Destination             Protocol Length Info
    1 0.000000000  2409:4091:900e:6603…    2404:6800:4007:81b:…    UDP        91 65047 → 443 Len=29
    2 0.015854306  2404:6800:4007:81b:…    2409:4091:900e:6603…    UDP        87 443 → 65047 Len=25
    3 0.729448905  2409:4091:900e:6603…    2404:6800:4009:822:…    UDP        91 62666 → 443 Len=29
    4 0.769361288  2404:6800:4009:822:…    2409:4091:900e:6603…    UDP        88 443 → 62666 Len=26
    5 1.790487126  2405:200:1608:1731:…    2409:4091:900e:6603…    TLSv1.2   105 Encrypted Alert
    6 2.317653497  fe80::3e0a:f3ff:fea…    fe80::45f8:337c:3d8…    ICMPv6     86 Neighbor Solicitation for fe80::45f8:337c:3d8b:2a2f from 3c:0a:f3:aa:59:2a
    7 2.317814736  fe80::45f8:337c:3d8…    fe80::3e0a:f3ff:fea…    ICMPv6     78 Neighbor Advertisement fe80::45f8:337c:3d8b:2a2f (sol)
    8 2.830458581  fe80::3e0a:f3ff:fea…    2409:4091:900e:6603…    ICMPv6     86 Neighbor Solicitation for 2409:4091:900e:6603:588e:87f7:2d6:bb8f from 3c:0a:f3:aa:59:2a
    9 2.830459259  2409:4091:900e:6603…    fe80::3e0a:f3ff:fea…    ICMPv6     86 Neighbor Advertisement 2409:4091:900e:6603:588e:87f7:2d6:bb8f (sol, ovr) is at 74:df:bf:7f:81:a7
   10 3.381257741  192.168.31.249          192.168.31.1            DNS        75 Standard query 0x8da7 AAAA ssl.gstatic.com
   11 3.381258395  192.168.31.249          192.168.31.1            DNS        75 Standard query 0x382d A ssl.gstatic.com
   12 3.382753233  192.168.31.249          192.168.31.1            DNS        75 Standard query 0x93c6 HTTPS ssl.gstatic.com
   13 3.385403102  192.168.31.1            192.168.31.249          DNS       103 Standard query response 0x8da7 AAAA ssl.gstatic.com AAAA 2404:6800:4007:831::2003
   14 3.385404174  192.168.31.1            192.168.31.249          DNS        91 Standard query response 0x382d A ssl.gstatic.com A 142.250.71.99
   15 3.391519517  192.168.31.1            192.168.31.249          DNS       132 Standard query response 0x93c6 HTTPS ssl.gstatic.com SOA ns1.google.com
   16 3.392311084  2409:4091:900e:6603…    2404:6800:4007:831:…    QUIC     1292 Initial, DCID=4dad4cdf8fe4071f, PKN: 1, PING, PING, PING, PING, CRYPTO, PING, PING, PING, CRYPTO, CRYPTO, CRYPTO, PING, CRYPTO, C…
   17 3.392311419  2409:4091:900e:6603…    2404:6800:4007:831:…    QUIC     1292 Initial, DCID=4dad4cdf8fe4071f, PKN: 2, PING, PADDING, CRYPTO, CRYPTO, PADDING, CRYPTO, PING, PING, PING, CRYPTO
   18 3.392906700  2409:4091:900e:6603…    2404:6800:4007:831:…    QUIC      143 0-RTT, DCID=4dad4cdf8fe4071f
   19 3.392906954  2409:4091:900e:6603…    2404:6800:4007:831:…    QUIC     1005 0-RTT, DCID=4dad4cdf8fe4071f
   20 3.405230491  2404:6800:4007:831:…    2409:4091:900e:6603…    QUIC      102 Initial, SCID=edad4cdf8fe4071f, PKN: 1, ACK
   21 3.408692068  2404:6800:4007:831:…    2409:4091:900e:6603…    QUIC     1292 Initial, SCID=edad4cdf8fe4071f, PKN: 2, ACK, PADDING
   22 3.420243094  2409:4091:900e:6603…    2404:6800:4007:831:…    QUIC     1292 Initial, DCID=edad4cdf8fe4071f, PKN: 6, PADDING, PING, PADDING
   23 3.432788509  2404:6800:4007:831:…    2409:4091:900e:6603…    QUIC     1292 Initial, SCID=edad4cdf8fe4071f, PKN: 3, ACK, PADDING
   24 3.438874904  2404:6800:4007:831:…    2409:4091:900e:6603…    QUIC     1292 Initial, SCID=edad4cdf8fe4071f, PKN: 4, CRYPTO, PADDING
   25 3.438875362  2404:6800:4007:831:…    2409:4091:900e:6603…    QUIC      363 Protected Payload (KP0)
   26 3.439479421  2409:4091:900e:6603…    2404:6800:4007:831:…    QUIC      140 Handshake, DCID=edad4cdf8fe4071f
   27 3.439479709  2404:6800:4007:831:…    2409:4091:900e:6603…    QUIC      991 Protected Payload (KP0)
   28 3.439479744  2404:6800:4007:831:…    2409:4091:900e:6603…    QUIC      100 Protected Payload (KP0)
   29 3.439479779  2404:6800:4007:831:…    2409:4091:900e:6603…    QUIC       86 Protected Payload (KP0)
   30 3.439479890  2404:6800:4007:831:…    2409:4091:900e:6603…    QUIC       84 Protected Payload (KP0)
   31 3.439479923  2409:4091:900e:6603…    2404:6800:4007:831:…    QUIC       93 Protected Payload (KP0), DCID=edad4cdf8fe4071f
   32 3.439479960  2409:4091:900e:6603…    2404:6800:4007:831:…    QUIC       93 Protected Payload (KP0), DCID=edad4cdf8fe4071f
   33 3.451440379  2404:6800:4007:831:…    2409:4091:900e:6603…    QUIC      182 Protected Payload (KP0)
   34 3.476437872  2404:6800:4007:831:…    2409:4091:900e:6603…    QUIC      597 Protected Payload (KP0)
   35 3.476438322  2404:6800:4007:831:…    2409:4091:900e:6603…    QUIC       84 Protected Payload (KP0)
   36 3.476438439  2409:4091:900e:6603…    2404:6800:4007:831:…    QUIC       97 Protected Payload (KP0), DCID=edad4cdf8fe4071f
   37 3.489113968  2404:6800:4007:80e:…    2409:4091:900e:6603…    UDP      1292 443 → 56737 Len=1230
   38 3.489114321  2404:6800:4007:80e:…    2409:4091:900e:6603…    UDP      1292 443 → 56737 Len=1230
```

```
› Frame 25: 363 bytes on wire (2904 bits), 363 bytes captured (2904 bits) on interface eth0, id 0
› Ethernet II, Src: CloudNetwork_aa:59:2a (3c:0a:f3:aa:59:2a), Dst: LiteonTechno_7f:81:a7 (74:df:bf:7f:8
› Internet Protocol Version 6, Src: 2404:6800:4007:831::2003, Dst: 2409:4091:900e:6603:588e:87f7:2d6:bb8
› User Datagram Protocol, Src Port: 443, Dst Port: 58599
```

```
0000  74 df bf 7f 81 a7 3c 0a  f3 aa 59 2a 86 dd 60 00   t·····<···Y*··`·
0010  00 00 01 35 11 39 24 04  68 00 40 07 08 31 00 00   ···5·9$·h·@··1··
0020  00 00 00 00 20 03 24 09  40 91 90 0e 66 03 58 8e   ···· ·$·@···f·X·
0030  87 f7 02 d6 bb 8f 01 bb  e4 e7 01 35 68 5f e3 00   ···········5h_··
0040  00 00 01 00 08 ed ad 4c  df 8f e4 07 1f 40 da f2   ·······L·····@··
```

eth0: <live capture in progress> | Packets: 231 | Profile: Default

5. **Apply Display Filters for Analysis**
   Examples:
   - **DNS Packets:**

     ```
     dns
     ```

   - **HTTPS Packets:**

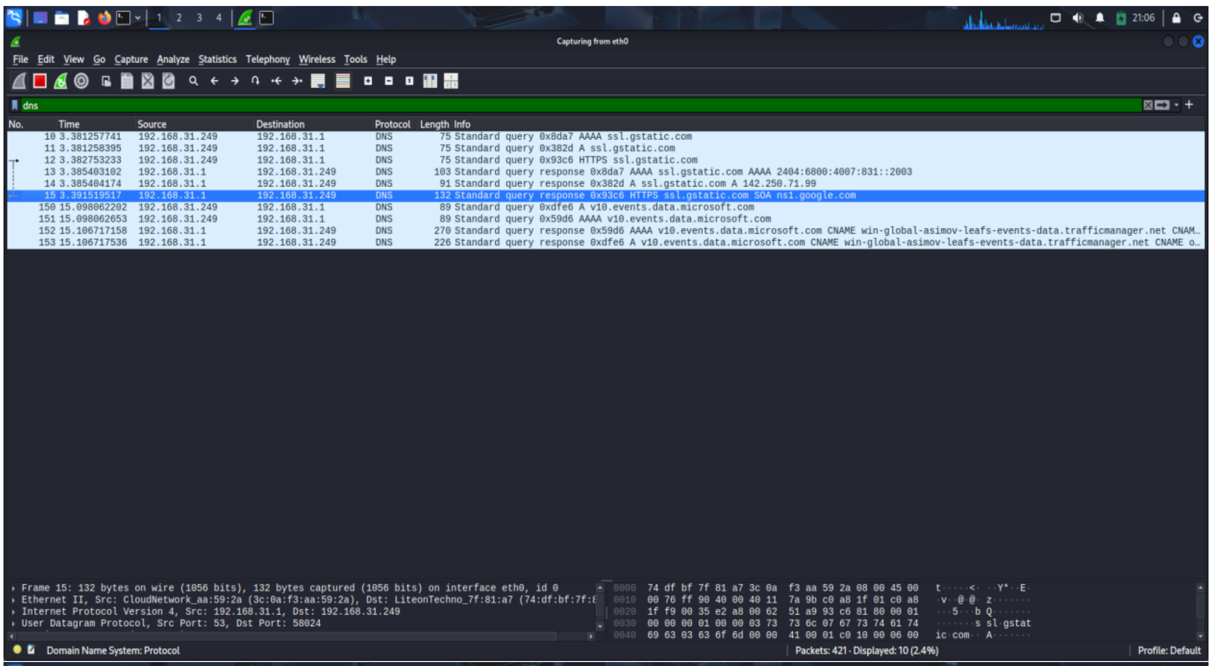
     ```
     tls
     ```

   - **Packets from specific IP:**

     ```
     ip.addr == 18.161.216.37
     ```

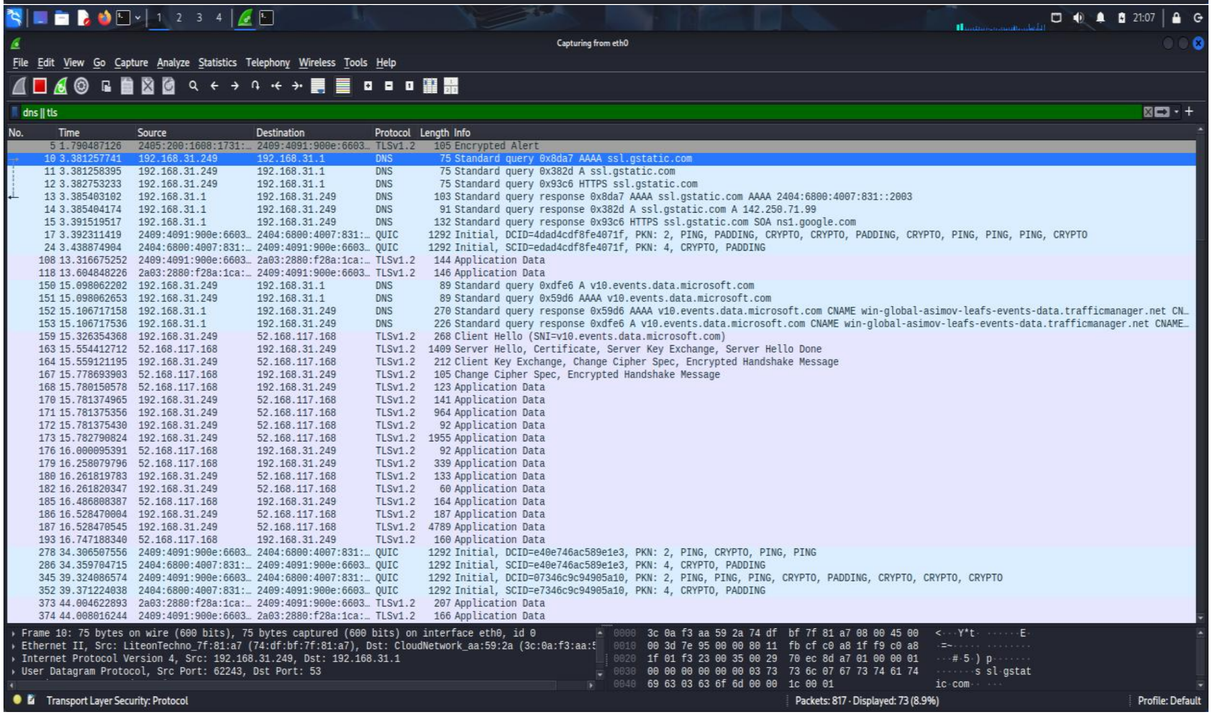
   - **TCP Port 443:**

     ```
     tcp.port == 443
     ```



```
dns
No.  Time           Source           Destination      Protocol Length Info
  10 3.381257741    192.168.31.249   192.168.31.1     DNS      75 Standard query 0x8da7 AAAA ssl.gstatic.com
  11 3.381258395    192.168.31.249   192.168.31.1     DNS      75 Standard query 0x382d A ssl.gstatic.com
  12 3.382753233    192.168.31.249   192.168.31.1     DNS      75 Standard query 0x93c6 HTTPS ssl.gstatic.com
  13 3.385403102    192.168.31.1     192.168.31.249   DNS     103 Standard query response 0x8da7 AAAA ssl.gstatic.com AAAA 2404:6800:4007:831::2003
  14 3.385404174    192.168.31.1     192.168.31.249   DNS      91 Standard query response 0x382d A ssl.gstatic.com A 142.250.71.99
  15 3.391519517    192.168.31.1     192.168.31.249   DNS     132 Standard query response 0x93c6 HTTPS ssl.gstatic.com SOA ns1.google.com
 150 15.098062202   192.168.31.249   192.168.31.1     DNS      89 Standard query 0xdfe6 A v10.events.data.microsoft.com
 151 15.098062653   192.168.31.249   192.168.31.1     DNS      89 Standard query 0x59d6 AAAA v10.events.data.microsoft.com
 152 15.106717158   192.168.31.1     192.168.31.249   DNS     270 Standard query response 0x59d6 AAAA v10.events.data.microsoft.com CNAME win-global-asimov-leafs-events-data.trafficmanager.net CNAM…
 153 15.106717536   192.168.31.1     192.168.31.249   DNS     226 Standard query response 0xdfe6 A v10.events.data.microsoft.com CNAME win-global-asimov-leafs-events-data.trafficmanager.net CNAME o…

› Frame 15: 132 bytes on wire (1056 bits), 132 bytes captured (1056 bits) on interface eth0, id 0
› Ethernet II, Src: CloudNetwork_aa:59:2a (3c:0a:f3:aa:59:2a), Dst: LiteonTechno_7f:81:a7 (74:df:bf:7f:8
› Internet Protocol Version 4, Src: 192.168.31.1, Dst: 192.168.31.249
› User Datagram Protocol, Src Port: 53, Dst Port: 58024

0000  74 df bf 7f 81 a7 3c 0a  f3 aa 59 2a 08 00 45 00   t·····<· ··Y*··E·
0010  00 76 ff 90 40 00 40 11  7a 9b c0 a8 1f 01 c0 a8   ·v··@·@· z·······
0020  1f f9 00 35 e2 a8 00 62  51 a9 93 c6 81 80 00 01   ···5···b Q·······
0030  00 00 00 01 00 00 03 73  73 6c 07 67 73 74 61 74   ·······s sl·gstat
0040  69 63 03 63 6f 6d 00 00  41 00 01 c0 10 00 06 00   ic·com·· A·······

Domain Name System: Protocol          Packets: 421 · Displayed: 10 (2.4%)          Profile: Default
```

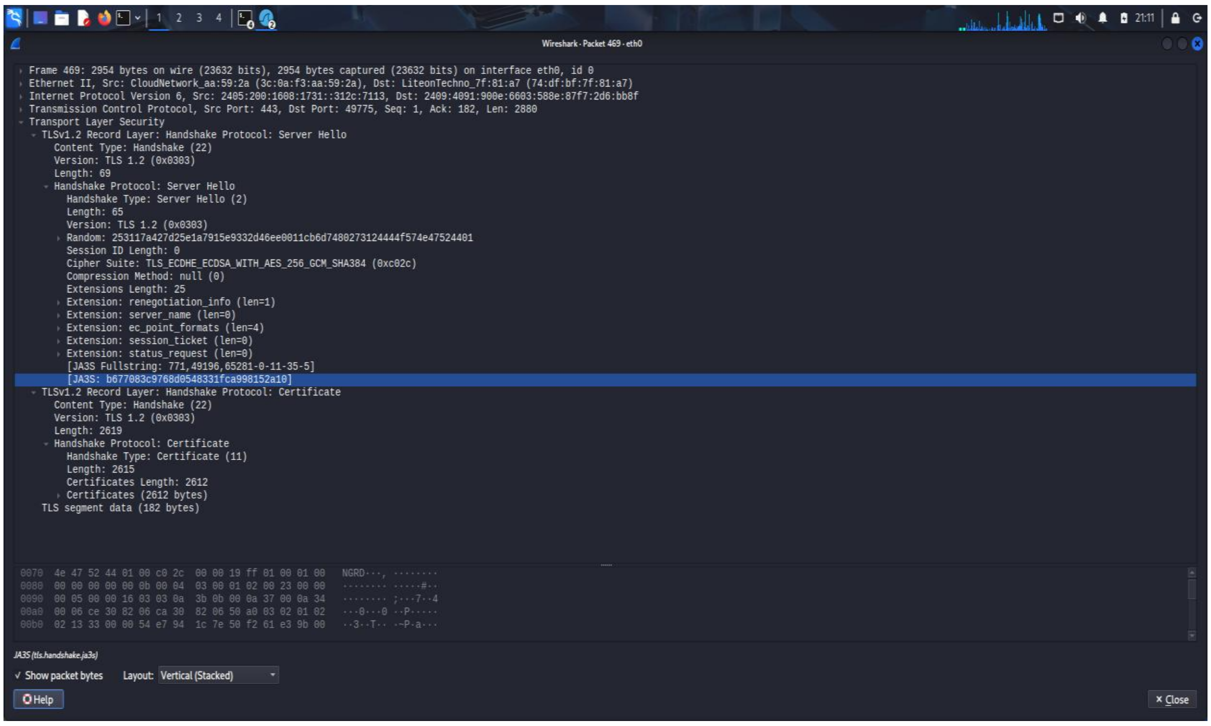
Capturing from eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
21:06



```
dns || tls
No.  Time           Source                Destination           Protocol Length Info
   5 1.790487126    2405:200:1608:1731:…  2409:4091:900e:6603…  TLSv1.2  105 Encrypted Alert
  10 3.381257741    192.168.31.249        192.168.31.1          DNS       75 Standard query 0x8da7 AAAA ssl.gstatic.com
  11 3.381258395    192.168.31.249        192.168.31.1          DNS       75 Standard query 0x382d A ssl.gstatic.com
  12 3.382753233    192.168.31.249        192.168.31.1          DNS       75 Standard query 0x93c6 HTTPS ssl.gstatic.com
  13 3.385403102    192.168.31.1          192.168.31.249        DNS      103 Standard query response 0x8da7 AAAA ssl.gstatic.com AAAA 2404:6800:4007:831::2003
  14 3.385404174    192.168.31.1          192.168.31.249        DNS       91 Standard query response 0x382d A ssl.gstatic.com A 142.250.71.99
  15 3.391519517    192.168.31.1          192.168.31.249        DNS      132 Standard query response 0x93c6 HTTPS ssl.gstatic.com SOA ns1.google.com
  17 3.392311419    2409:4091:900e:6603…  2404:6800:4007:831:…  QUIC    1292 Initial, DCID=4dad4cdf8fe4071f, PKN: 2, PING, PADDING, CRYPTO, CRYPTO, PADDING, CRYPTO, PING, PING, PING, CRYPTO
  24 3.438874904    2404:6800:4007:831:…  2409:4091:900e:6603…  QUIC    1292 Initial, SCID=edad4cdf8fe4071f, PKN: 4, CRYPTO, PADDING
 108 13.316675252   2409:4091:900e:6603…  2a03:2880:f28a:1ca:…  TLSv1.2  144 Application Data
 118 13.604848226   2a03:2880:f28a:1ca:…  2409:4091:900e:6603…  TLSv1.2  146 Application Data
 150 15.098062202   192.168.31.249        192.168.31.1          DNS       89 Standard query 0xdfe6 A v10.events.data.microsoft.com
 151 15.098062653   192.168.31.249        192.168.31.1          DNS       89 Standard query 0x59d6 AAAA v10.events.data.microsoft.com
 152 15.106717158   192.168.31.1          192.168.31.249        DNS      270 Standard query response 0x59d6 AAAA v10.events.data.microsoft.com CNAME win-global-asimov-leafs-events-data.trafficmanager.net CN…
 153 15.106717536   192.168.31.1          192.168.31.249        DNS      226 Standard query response 0xdfe6 A v10.events.data.microsoft.com CNAME win-global-asimov-leafs-events-data.trafficmanager.net CNAME…
 159 15.326354368   192.168.31.249        52.168.117.168        TLSv1.2  268 Client Hello (SNI=v10.events.data.microsoft.com)
 163 15.554412712   52.168.117.168        192.168.31.249        TLSv1.2 1409 Server Hello, Certificate, Server Key Exchange, Server Hello Done
 164 15.559121195   192.168.31.249        52.168.117.168        TLSv1.2  212 Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
 167 15.778693903   52.168.117.168        192.168.31.249        TLSv1.2  105 Change Cipher Spec, Encrypted Handshake Message
 168 15.780150578   52.168.117.168        192.168.31.249        TLSv1.2  123 Application Data
 170 15.781374965   192.168.31.249        52.168.117.168        TLSv1.2  141 Application Data
 171 15.781375356   192.168.31.249        52.168.117.168        TLSv1.2  964 Application Data
 172 15.781375430   192.168.31.249        52.168.117.168        TLSv1.2   92 Application Data
 173 15.782790824   192.168.31.249        52.168.117.168        TLSv1.2 1955 Application Data
 176 16.000095391   52.168.117.168        192.168.31.249        TLSv1.2   92 Application Data
 179 16.258079796   52.168.117.168        192.168.31.249        TLSv1.2  339 Application Data
 180 16.261819783   192.168.31.249        52.168.117.168        TLSv1.2  133 Application Data
 182 16.261820347   192.168.31.249        52.168.117.168        TLSv1.2   60 Application Data
 185 16.486808387   52.168.117.168        192.168.31.249        TLSv1.2  164 Application Data
 186 16.528470004   192.168.31.249        52.168.117.168        TLSv1.2  187 Application Data
 187 16.528470545   192.168.31.249        52.168.117.168        TLSv1.2 4789 Application Data
 193 16.747188340   52.168.117.168        192.168.31.249        TLSv1.2  160 Application Data
 278 34.306507556   2409:4091:900e:6603…  2404:6800:4007:831:…  QUIC    1292 Initial, DCID=e40e746ac589e1e3, PKN: 2, PING, CRYPTO, PING, PING
 286 34.359704715   2404:6800:4007:831:…  2409:4091:900e:6603…  QUIC    1292 Initial, SCID=e40e746ac589e1e3, PKN: 4, CRYPTO, PADDING
 345 39.324086574   2409:4091:900e:6603…  2404:6800:4007:831:…  QUIC    1292 Initial, DCID=07346c9c94905a10, PKN: 2, PING, PING, PING, CRYPTO, PADDING, CRYPTO, CRYPTO, CRYPTO
 352 39.371224038   2404:6800:4007:831:…  2409:4091:900e:6603…  QUIC    1292 Initial, SCID=e7346c9c94905a10, PKN: 4, CRYPTO, PADDING
 373 44.004622893   2a03:2880:f28a:1ca:…  2409:4091:900e:6603…  TLSv1.2  207 Application Data
 374 44.008016244   2409:4091:900e:6603…  2a03:2880:f28a:1ca:…  TLSv1.2  166 Application Data

› Frame 10: 75 bytes on wire (600 bits), 75 bytes captured (600 bits) on interface eth0, id 0
› Ethernet II, Src: LiteonTechno_7f:81:a7 (74:df:bf:7f:81:a7), Dst: CloudNetwork_aa:59:2a (3c:0a:f3:aa:5
› Internet Protocol Version 4, Src: 192.168.31.249, Dst: 192.168.31.1
› User Datagram Protocol, Src Port: 62243, Dst Port: 53

0000  3c 0a f3 aa 59 2a 74 df  bf 7f 81 a7 08 00 45 00   <···Y*t· ······E·
0010  00 3d 7e 95 00 00 80 11  fb cf c0 a8 1f f9 c0 a8   ·=~····· ········
0020  1f 01 f3 23 00 35 00 29  70 ec 8d a7 01 00 00 01   ···#·5·) p·······
0030  00 00 00 00 00 00 03 73  73 6c 07 67 73 74 61 74   ·······s sl·gstat
0040  69 63 03 63 6f 6d 00 00  1c 00 01                  ic·com·· ···

Transport Layer Security: Protocol          Packets: 817 · Displayed: 73 (8.9%)          Profile: Default
```

Capturing from eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
21:07

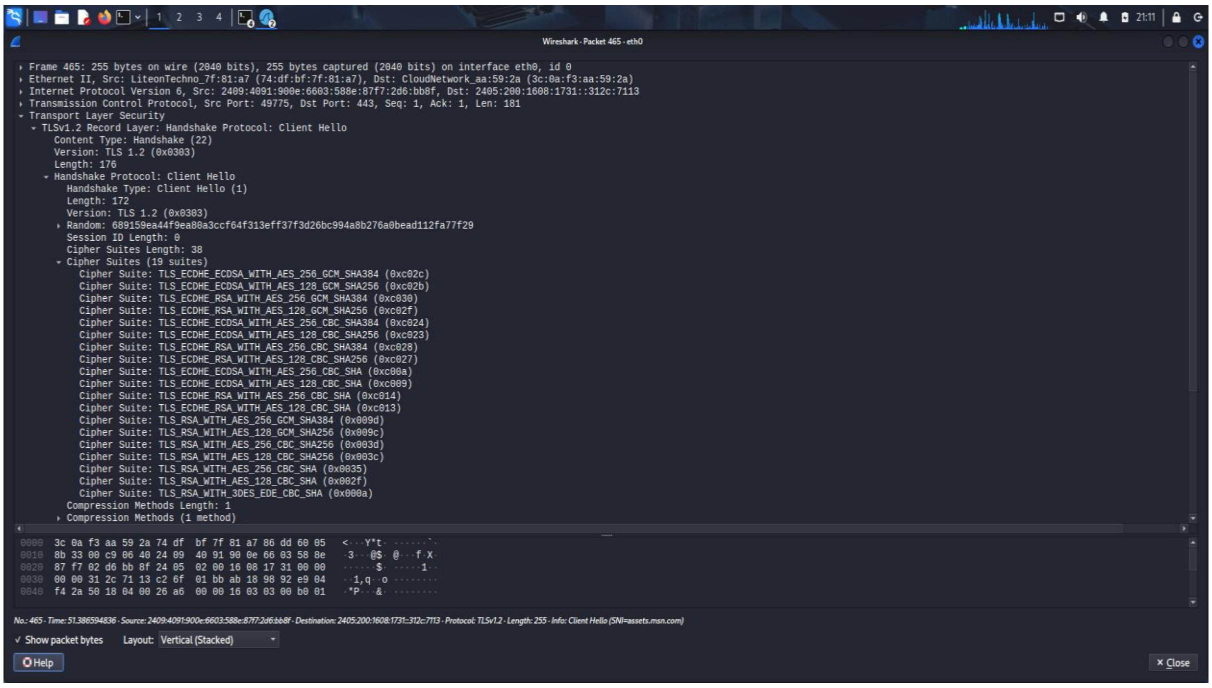
6. **Analyze Captured Packets**
   - Select any packet and expand:
     - **Ethernet Layer:** Source & Destination MAC addresses
     - **IP Layer:** Source & Destination IP addresses
     - **TCP/UDP Layer:** Port numbers, flags (SYN, ACK)
     - **Application Layer:** Protocol details (DNS query, HTTP request, TLS handshake)


Wireshark · Packet 469 · eth0

```
› Frame 469: 2954 bytes on wire (23632 bits), 2954 bytes captured (23632 bits) on interface eth0, id 0
› Ethernet II, Src: CloudNetwork_aa:59:2a (3c:0a:f3:aa:59:2a), Dst: LiteonTechno_7f:81:a7 (74:df:bf:7f:81:a7)
› Internet Protocol Version 6, Src: 2405:200:1608:1731::312c:7113, Dst: 2409:4091:900e:6603:588e:87f7:2d6:bb8f
› Transmission Control Protocol, Src Port: 443, Dst Port: 49775, Seq: 1, Ack: 182, Len: 2880
⌄ Transport Layer Security
  ⌄ TLSv1.2 Record Layer: Handshake Protocol: Server Hello
      Content Type: Handshake (22)
      Version: TLS 1.2 (0x0303)
      Length: 69
    ⌄ Handshake Protocol: Server Hello
        Handshake Type: Server Hello (2)
        Length: 65
        Version: TLS 1.2 (0x0303)
      › Random: 253117a427d25e1a7915e9332d46ee0011cb6d7480273124444f574e47524401
        Session ID Length: 0
        Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0xc02c)
        Compression Method: null (0)
        Extensions Length: 25
      › Extension: renegotiation_info (len=1)
      › Extension: server_name (len=0)
      › Extension: ec_point_formats (len=4)
      › Extension: session_ticket (len=0)
      › Extension: status_request (len=0)
        [JA3S Fullstring: 771,49196,65281-0-11-35-5]
        [JA3S: b677083c9768d0548331fca998152a10]
  ⌄ TLSv1.2 Record Layer: Handshake Protocol: Certificate
      Content Type: Handshake (22)
      Version: TLS 1.2 (0x0303)
      Length: 2619
    ⌄ Handshake Protocol: Certificate
        Handshake Type: Certificate (11)
        Length: 2615
        Certificates Length: 2612
      › Certificates (2612 bytes)
    TLS segment data (182 bytes)

0070  4e 47 52 44 01 00 c0 2c  00 00 19 ff 01 00 01 00   NGRD···, ········
0080  00 00 00 00 00 0b 00 04  03 00 01 02 00 23 00 00   ········ ·····#··
0090  00 05 00 00 16 03 03 0a  3b 0b 00 0a 37 00 0a 34   ········ ;···7··4
00a0  00 06 ce 30 82 06 ca 30  82 06 50 a0 03 02 01 02   ···0···0 ··P·····
00b0  02 13 33 00 00 54 e7 94  1c 7e 50 f2 61 e3 9b 00   ··3··T·· ·~P·a···
```

JA3S (tls.handshake.ja3s)

✓ Show packet bytes Layout: Vertical (Stacked)


Wireshark · Packet 465 · eth0

```
› Frame 465: 255 bytes on wire (2040 bits), 255 bytes captured (2040 bits) on interface eth0, id 0
› Ethernet II, Src: LiteonTechno_7f:81:a7 (74:df:bf:7f:81:a7), Dst: CloudNetwork_aa:59:2a (3c:0a:f3:aa:59:2a)
› Internet Protocol Version 6, Src: 2409:4091:900e:6603:588e:87f7:2d6:bb8f, Dst: 2405:200:1608:1731::312c:7113
› Transmission Control Protocol, Src Port: 49775, Dst Port: 443, Seq: 1, Ack: 1, Len: 181
⌄ Transport Layer Security
  ⌄ TLSv1.2 Record Layer: Handshake Protocol: Client Hello
      Content Type: Handshake (22)
      Version: TLS 1.2 (0x0303)
      Length: 176
    ⌄ Handshake Protocol: Client Hello
        Handshake Type: Client Hello (1)
        Length: 172
        Version: TLS 1.2 (0x0303)
      › Random: 689159ea44f9ea80a3ccf64f313eff37f3d26bc994a8b276a0bead112fa77f29
        Session ID Length: 0
        Cipher Suites Length: 38
      ⌄ Cipher Suites (19 suites)
          Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0xc02c)
          Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (0xc02b)
          Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030)
          Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f)
          Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 (0xc024)
          Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 (0xc023)
          Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (0xc028)
          Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (0xc027)
          Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (0xc00a)
          Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (0xc009)
          Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)
          Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (0xc013)
          Cipher Suite: TLS_RSA_WITH_AES_256_GCM_SHA384 (0x009d)
          Cipher Suite: TLS_RSA_WITH_AES_128_GCM_SHA256 (0x009c)
          Cipher Suite: TLS_RSA_WITH_AES_256_CBC_SHA256 (0x003d)
          Cipher Suite: TLS_RSA_WITH_AES_128_CBC_SHA256 (0x003c)
          Cipher Suite: TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
          Cipher Suite: TLS_RSA_WITH_AES_128_CBC_SHA (0x002f)
          Cipher Suite: TLS_RSA_WITH_3DES_EDE_CBC_SHA (0x000a)
        Compression Methods Length: 1
      › Compression Methods (1 method)

0000  3c 0a f3 aa 59 2a 74 df  bf 7f 81 a7 86 dd 60 05   <···Y*t· ······`·
0010  8b 33 00 c9 06 40 24 09  40 91 90 0e 66 03 58 8e   ·3···@$· @···f·X·
0020  87 f7 02 d6 bb 8f 24 05  02 00 16 08 17 31 00 00   ······$· ·····1··
0030  00 00 31 2c 71 13 c2 6f  01 bb ab 18 98 92 e9 04   ··1,q··o ········
0040  f4 2a 50 18 04 00 26 a6  00 00 16 03 03 00 b0 01   ·*P···&· ········
```

No.: 465 · Time: 51.386594836 · Source: 2409:4091:900e:6603:588e:87f7:2d6:bb8f · Destination: 2405:200:1608:1731::312c:7113 · Protocol: TLSv1.2 · Length: 255 · Info: Client Hello (SNI=assets.msn.com)

✓ Show packet bytes Layout: Vertical (Stacked)

7. **Stop Capture and Save**

   - Stop capture (red square button).
   - Save capture file in `.pcap` format for submission.

Part B: Capturing Login Credentials over HTTP

Step 1 – Setup Apache Web Server

```
sudo service apache2 start
sudo mkdir -p /var/www/html/testlogin
cd /var/www/html/testlogin
```

Step 2 – Create Login HTML Page

```
sudo nano login.html
```

```
html
CopyEdit
<!DOCTYPE html>
<html>
<head>
    <title>Test Login Page</title>
</head>
<body>
    <h2>Login Form</h2>
    <form action="login.php" method="post">
        Username: <input type="text" name="username"><br><br>
        Password: <input type="password" name="password"><br><br>
        <input type="submit" value="Login">
    </form>
</body>
</html>
```

Save (`Ctrl + O` → Enter → `Ctrl + X`).

Step 3 – Create PHP Script

```
sudo nano login.php
```

```
php
CopyEdit
<?php
$username = $_POST['username'];
$password = $_POST['password'];

echo "<h2>Login Attempt</h2>";
echo "Username: " . $username . "<br>";
echo "Password: " . $password . "<br>";
?>
```

Step 4 – Start/Enable Apache

```
sudo service apache2 status
sudo service apache2 start
```

Step 5 – Test Page

Browser:

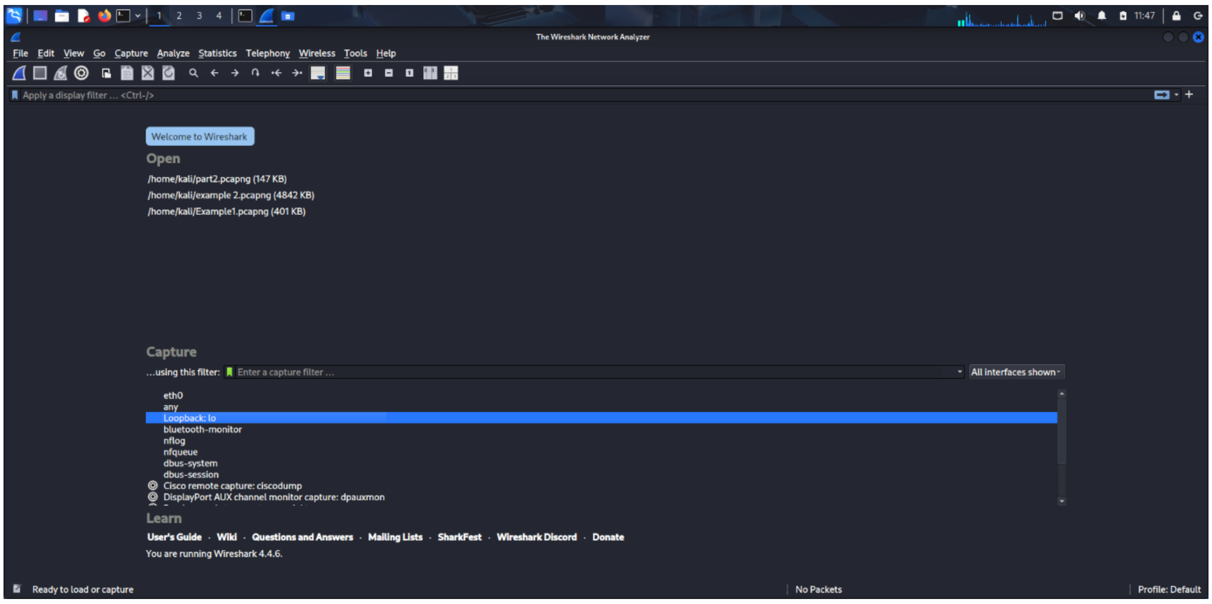
```
http://127.0.0.1/testlogin/login.html
```

Or: Type in bash

```
curl http://127.0.0.1/testlogin/login.html
```

Step 6 – Capture with Wireshark

- Open Wireshark → select `lo` (loopback) interface.



The Wireshark Network Analyzer

Welcome to Wireshark

Open

/home/kali/part2.pcapng (147 KB)
/home/kali/example 2.pcapng (4842 KB)
/home/kali/Example1.pcapng (401 KB)

Capture

...using this filter: Enter a capture filter ... All interfaces shown

eth0
any
Loopback: lo
bluetooth-monitor
nflog
nfqueue
dbus-system
dbus-session
Cisco remote capture: ciscodump
DisplayPort AUX channel monitor capture: dpauxmon

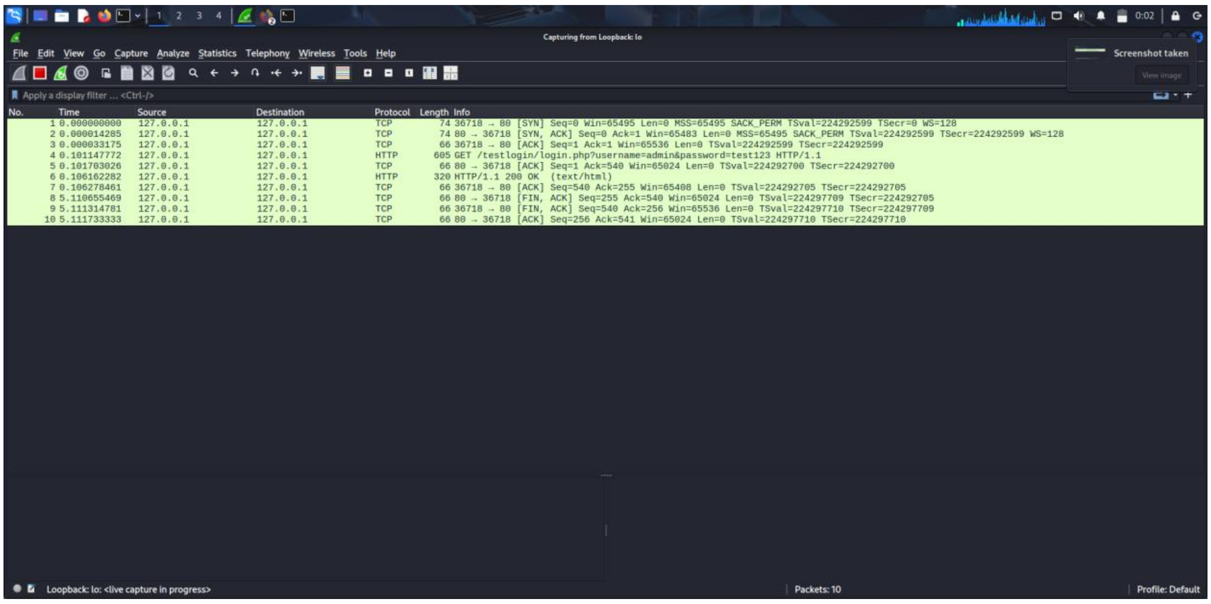
Learn

User's Guide · Wiki · Questions and Answers · Mailing Lists · SharkFest · Wireshark Discord · Donate

You are running Wireshark 4.4.6.

Ready to load or capture | No Packets | Profile: Default

- Start capture.



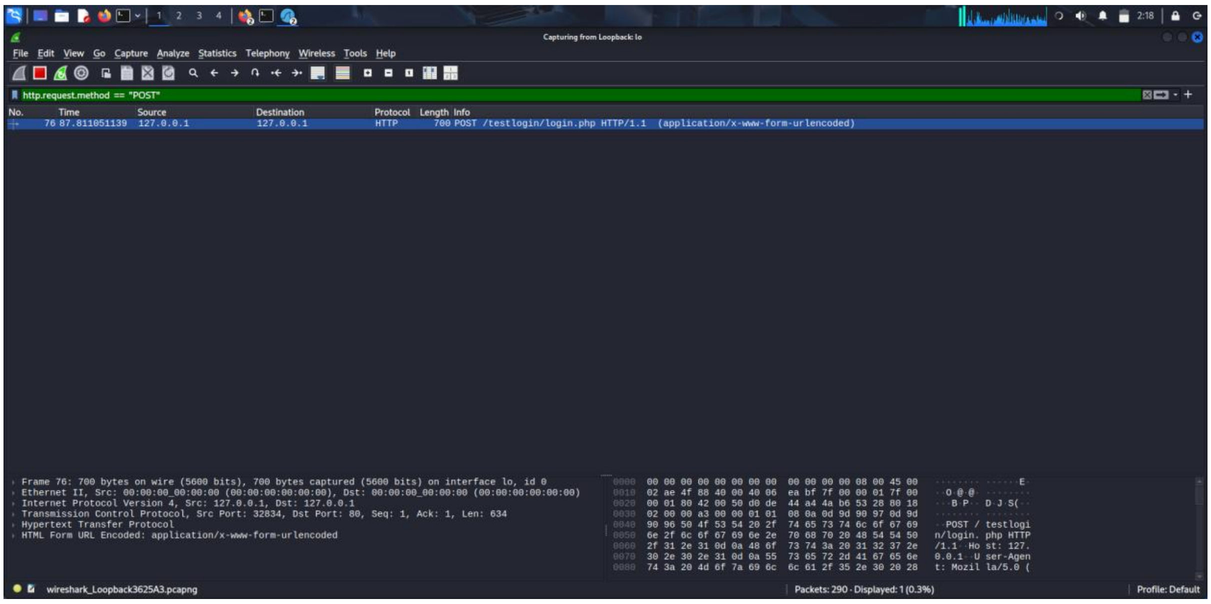
Capturing from Loopback: lo

```
No.  Time         Source     Destination  Protocol Length Info
   1 0.000000000  127.0.0.1  127.0.0.1    TCP        74 36718 → 80 [SYN] Seq=0 Win=65495 Len=0 MSS=65495 SACK_PERM TSval=224292599 TSecr=0 WS=128
   2 0.000014285  127.0.0.1  127.0.0.1    TCP        74 80 → 36718 [SYN, ACK] Seq=0 Ack=1 Win=65483 Len=0 MSS=65495 SACK_PERM TSval=224292599 TSecr=224292599 WS=128
   3 0.000033175  127.0.0.1  127.0.0.1    TCP        66 36718 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0 TSval=224292599 TSecr=224292599
   4 0.101147772  127.0.0.1  127.0.0.1    HTTP      605 GET /testlogin/login.php?username=admin&password=test123 HTTP/1.1
   5 0.101703026  127.0.0.1  127.0.0.1    TCP        66 80 → 36718 [ACK] Seq=1 Ack=540 Win=65024 Len=0 TSval=224292700 TSecr=224292700
   6 0.106162282  127.0.0.1  127.0.0.1    HTTP      320 HTTP/1.1 200 OK  (text/html)
   7 0.106278461  127.0.0.1  127.0.0.1    TCP        66 36718 → 80 [ACK] Seq=540 Ack=255 Win=65408 Len=0 TSval=224292705 TSecr=224292705
   8 5.110655469  127.0.0.1  127.0.0.1    TCP        66 80 → 36718 [FIN, ACK] Seq=255 Ack=540 Win=65024 Len=0 TSval=224297709 TSecr=224292705
   9 5.111314781  127.0.0.1  127.0.0.1    TCP        66 36718 → 80 [FIN, ACK] Seq=540 Ack=256 Win=65536 Len=0 TSval=224297710 TSecr=224297709
  10 5.111733333  127.0.0.1  127.0.0.1    TCP        66 80 → 36718 [ACK] Seq=256 Ack=541 Win=65024 Len=0 TSval=224297710 TSecr=224297710
```

Loopback: lo: <live capture in progress> | Packets: 10 | Profile: Default

- Apply filter:

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


Capturing from Loopback: lo

http.request.method == "POST"

```
No.   Time          Source      Destination  Protocol  Length Info
   76 87.811051139  127.0.0.1   127.0.0.1    HTTP         700 POST /testlogin/login.php HTTP/1.1  (application/x-www-form-urlencoded)
```

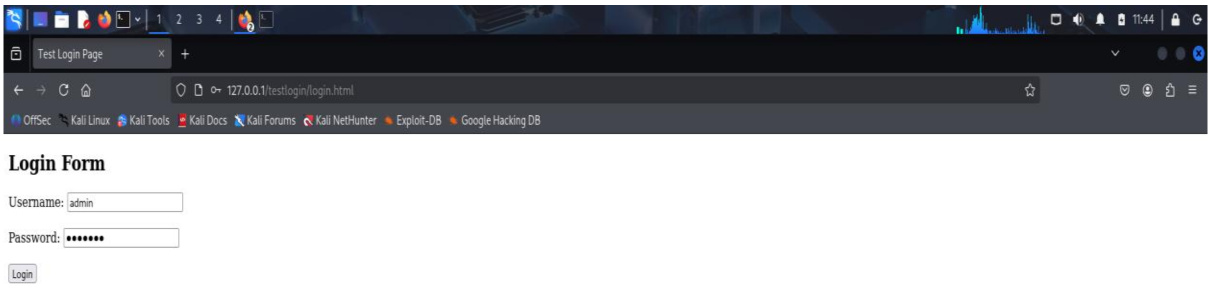
```
› Frame 76: 700 bytes on wire (5600 bits), 700 bytes captured (5600 bits) on interface lo, id 0
› Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
› Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
› Transmission Control Protocol, Src Port: 32834, Dst Port: 80, Seq: 1, Ack: 1, Len: 634
› Hypertext Transfer Protocol
› HTML Form URL Encoded: application/x-www-form-urlencoded
```

```
0000  00 00 00 00 00 00 00 00  00 00 00 00 08 00 45 00   ········ ······E·
0010  02 ae 4f 88 40 00 40 06  ea bf 7f 00 00 01 7f 00   ··O·@·@· ········
0020  00 01 80 42 00 50 d0 de  44 a4 4a b6 53 28 80 18   ···B·P·· D·J·S(··
0030  02 00 00 a3 00 00 01 01  08 0a 0d 9d 90 97 0d 9d   ········ ········
0040  90 96 50 4f 53 54 20 2f  74 65 73 74 6c 6f 67 69   ··POST / testlogi
0050  6e 2f 6c 6f 67 69 6e 2e  70 68 70 20 48 54 54 50   n/login. php HTTP
0060  2f 31 2e 31 0d 0a 48 6f  73 74 3a 20 31 32 37 2e   /1.1··Ho st: 127.
0070  30 2e 30 2e 31 0d 0a 55  73 65 72 2d 41 67 65 6e   0.0.1··U ser-Agen
0080  74 3a 20 4d 6f 7a 69 6c  6c 61 2f 35 2e 30 20 28   t: Mozil la/5.0 (
```

wireshark_Loopback3625A3.pcapng   Packets: 290 · Displayed: 1 (0.3%)   Profile: Default

Step 7 – Perform Login Attempt

- Username: `admin`
- Password: `test123`
- Submit form.


Test Login Page

127.0.0.1/testlogin/login.html

Login Form

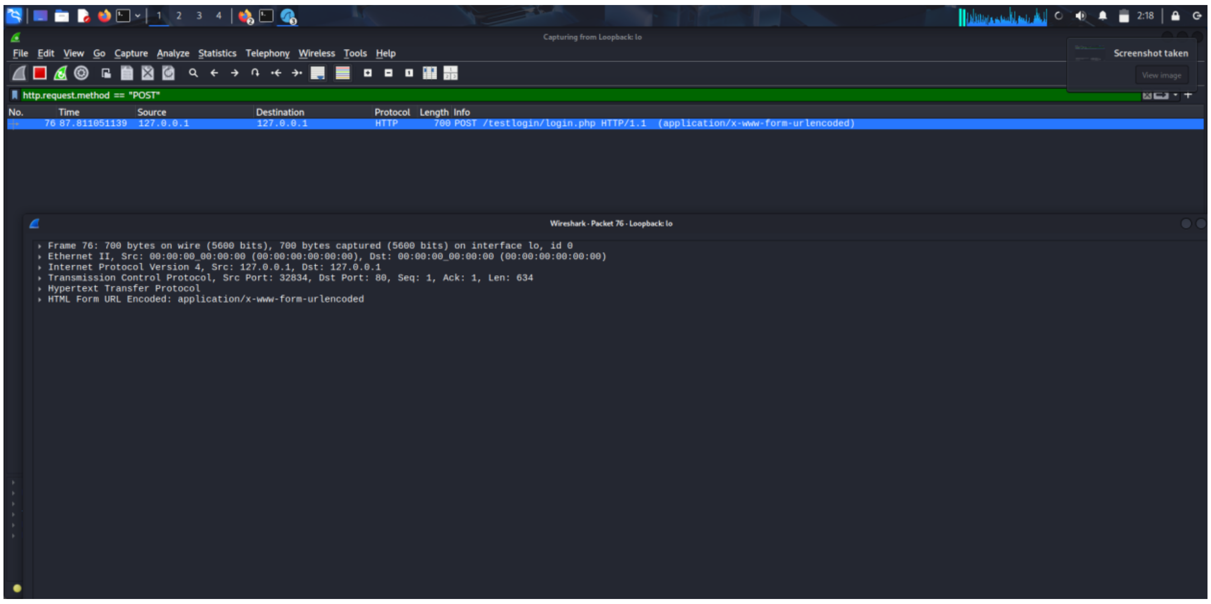
Username: admin

Password: •••••••

Login

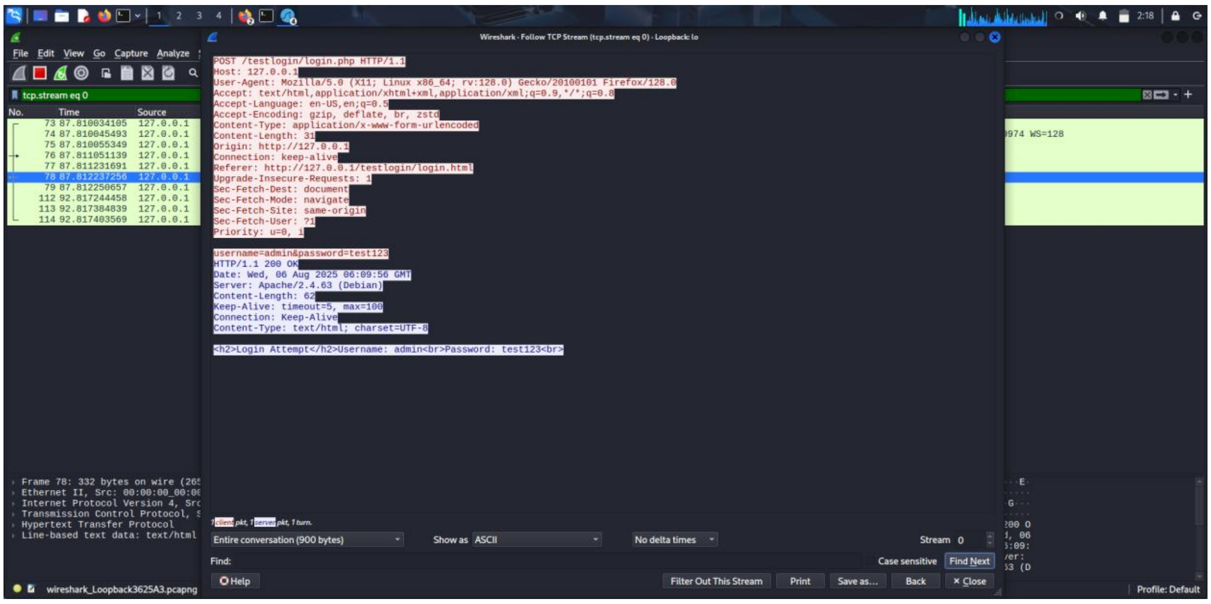
Step 8 – Analyze Packets

- Locate POST packet.
- Right-click → **Follow** → **TCP Stream**.
- Credentials will be visible in plain text.


Capturing from Loopback: lo
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Wireshark · Packet 76 · Loopback: lo

```
http.request.method == "POST"
No.    Time          Source      Destination  Protocol  Length  Info
   76  87.811051139  127.0.0.1   127.0.0.1    HTTP         700  POST /testlogin/login.php HTTP/1.1  (application/x-www-form-urlencoded)
```

```
› Frame 76: 700 bytes on wire (5600 bits), 700 bytes captured (5600 bits) on interface lo, id 0
› Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
› Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
› Transmission Control Protocol, Src Port: 32834, Dst Port: 80, Seq: 1, Ack: 1, Len: 634
› Hypertext Transfer Protocol
› HTML Form URL Encoded: application/x-www-form-urlencoded
```



```
tcp.stream eq 0
No.   Time          Source
   73 87.810034105  127.0.0.1
   74 87.810045493  127.0.0.1
   75 87.810055349  127.0.0.1
   76 87.811051139  127.0.0.1
   77 87.811231691  127.0.0.1
   78 87.812237256  127.0.0.1
   79 87.812250657  127.0.0.1
  112 92.817244458  127.0.0.1
  113 92.817384839  127.0.0.1
  114 92.817403569  127.0.0.1

› Frame 78: 332 bytes on wire (265
› Ethernet II, Src: 00:00:00_00:0
› Internet Protocol Version 4, Src
› Transmission Control Protocol, S
› Hypertext Transfer Protocol
› Line-based text data: text/html

wireshark_Loopback3625A3.pcapng
```

```
POST /testlogin/login.php HTTP/1.1
Host: 127.0.0.1
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br, zstd
Content-Type: application/x-www-form-urlencoded
Content-Length: 31
Origin: http://127.0.0.1
Connection: keep-alive
Referer: http://127.0.0.1/testlogin/login.html
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
Priority: u=0, i

username=admin&password=test123
HTTP/1.1 200 OK
Date: Wed, 06 Aug 2025 06:09:56 GMT
Server: Apache/2.4.63 (Debian)
Content-Length: 62
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8

<h2>Login Attempt</h2>Username: admin<br>Password: test123<br>
```

Observations

- Network traffic from various applications can be captured in real-time.
- Protocol filters help narrow down the traffic of interest.
- Details such as source/destination IP, MAC addresses, ports, and application data can be examined.
- HTTP POST requests transmit data in plaintext.
- Credentials can be directly read in the captured packets when using HTTP.
- Using HTTPS would encrypt this data and prevent such interception.

Post-Lab Discussion

- **Bridged Adapter mode** allows the VM to be part of the physical network, enabling packet sniffing.
- **Promiscuous mode** is essential to capture packets not directly addressed to the capturing machine.

- **HTTPS traffic** is encrypted, so packet payloads are unreadable without decryption keys.
- HTTP lacks encryption, making it vulnerable to credential theft.
- Tools like Wireshark can be used by attackers to sniff sensitive data if encryption is not in place.
- Transitioning to HTTPS mitigates this risk by encrypting data in transit.
- The **loopback interface** captures traffic within the local system (127.0.0.1).

Viva Questions

1. What is the difference between a **display filter** and a **capture filter** in Wireshark?
2. Why do we use **promiscuous mode** in packet capturing?
3. How can you identify a **TCP three-way handshake** in Wireshark?
4. What is the difference between **HTTP** and **HTTPS** in terms of packet analysis?
5. Can Wireshark **decrypt HTTPS traffic**? Under what conditions?
6. Why are credentials visible in **HTTP packet captures** but not in HTTPS?
7. What is the significance of the **loopback interface (lo)** in Wireshark?
8. How can a **POST request** be identified in Wireshark?
9. What is the difference between **GET** and **POST** methods in HTTP?
10. How does HTTPS protect against **packet sniffing attacks**?