

Batch: B-1

Experiment Number: 2 - Application layer protocols.

Roll Number: 16010422234

Name: Chandana Ramesh Galgali

Aim of the Experiment: To explore application layer protocols with packet analysis using Wireshark.

Program/ Steps:

1. Start the machine as an administrator.
 2. Start the internet.
 3. Go to the official website of Wireshark. (www.wireshark.org) and download the old stable version of Wireshark for 32 bit windows operating system.
 4. After successful installation you will get the blue icon of Wireshark on the desktop.
 5. Click on the icon and start the software.
 6. Choose an interface and start capturing the packets.
 7. Study the packet details of any one application layer protocols.
 8. Understand color code in detail.
 9. Perform the statistics for captured application layer protocol packets. (Every student should perform for different protocols.)
 10. Show the output to the teacher and get it approved.
-

Output/Result:

1. Capturing a packet.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	172.17.16.129	172.31.0.6	TCP	54	54000 → 13111 [ACK] Seq=1 Ack=1 Win=8211 Len=0
2	0.011636	172.17.17.113	172.17.17.255	NBNS	92	Name query NB DEVCAC1C0<00>
3	0.020416	172.17.16.134	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
4	0.045605	HP_b8:96:ba	Broadcast	ARP	60	who has 172.17.16.93? Tell 172.17.16.100
5	0.078955	Micro-St_8d:18:c3	Broadcast	ARP	60	who has 169.254.169.254? Tell 172.17.16.180
6	0.098241	172.31.0.6	172.17.16.129	TCP	238	13111 → 54002 [PSH, ACK] Seq=1 Ack=1 Win=510 Len=184
7	0.118859	fe80::d541:c5c9:85fc:e46a	ff02::1:2	DHCPv6	165	Solicit XID: 0x559751 CID: 000100011dc6434bd8c8ba8d1b04
8	0.142299	172.17.16.129	172.31.0.6	TCP	54	54002 → 13111 [ACK] Seq=1 Ack=185 Win=1025 Len=0
9	0.148530	172.17.16.124	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
10	0.278063	Micro-St_8d:18:58	Broadcast	ARP	60	who has 172.17.17.254? Tell 169.254.149.124
11	0.297689	172.17.16.84	239.255.255.250	SSDP	218	M-SEARCH * HTTP/1.1
12	0.301211	Dell_5f:cc:ac	Broadcast	ARP	60	who has 172.17.16.179? Tell 172.17.17.227
13	0.383651	Dell_5f:cc:bf	Broadcast	ARP	60	who has 172.17.16.125? Tell 172.17.17.221
14	0.409366	HewlettP_69:76:93	Broadcast	ARP	60	who has 172.17.17.254? Tell 172.17.16.121
15	0.419727	172.17.16.101	239.255.255.250	SSDP	218	M-SEARCH * HTTP/1.1
16	0.464091	HP_b8:a1:c6	Broadcast	ARP	60	who has 169.254.169.254? Tell 172.17.16.84
17	0.484557	Cisco_70:7c:40	Spanning-tree-(for-bridges)_00	STP	60	RST, Root = 4096/1/b0:aa:77:66:d1:41 Cost = 20002 Port = 0x8049
18	0.510009	172.17.16.70	172.17.17.255	UDP	186	52482 → 51007 Len=144
19	0.536700	Micro-St_8d:14:ba	Broadcast	ARP	60	who has 172.17.16.121? Tell 172.17.16.170
20	0.544633	172.17.17.220	172.17.17.255	NBNS	92	Name query NB ENG-DC-8219-01<00>
21	0.545560	fe80::ed4d:f876:f998:b191	ff02::1:3	LUNMR	94	Standard query 0x8137 A eng-dc-b219-01
22	0.546129	172.17.17.220	224.0.0.252	LUNMR	74	Standard query 0x8137 A eng-dc-b219-01
23	0.547344	fe80::ed4d:f876:f998:b191	ff02::1:3	LUNMR	94	Standard query 0x09fa AAAA eng-dc-b219-01
24	0.547344	172.17.17.220	224.0.0.252	LUNMR	74	Standard query 0x09fa AAAA eng-dc-b219-01
25	0.762075	172.17.17.113	172.17.17.255	NBNS	92	Name query NB DEVCAC1C0<00>
26	0.795035	Dell_77:85:ce	Broadcast	ARP	60	who has 172.17.16.11? Tell 172.17.17.218
27	0.795035	Dell_77:85:ce	Broadcast	ARP	60	who has 172.17.16.18? Tell 172.17.17.218
28	0.841944	HP_b8:a8:a3	Broadcast	ARP	60	who has 172.17.17.56? Tell 172.17.16.98
29	0.891446	Dell_77:8e:d8	Broadcast	ARP	60	who has 172.17.17.56? Tell 172.17.17.92
30	0.919211	HewlettP_e4:75:4e	Broadcast	ARP	60	who has 172.17.17.155? Tell 172.17.17.133
31	0.965471	fe80::ed4d:f876:f998:b191	ff02::1:3	LUNMR	94	Standard query 0x8137 A eng-dc-b219-01
32	0.965471	fe80::ed4d:f876:f998:b191	ff02::1:3	LUNMR	94	Standard query 0x09fa AAAA eng-dc-b219-01
33	0.965471	172.17.17.220	224.0.0.252	LUNMR	74	Standard query 0x8137 A eng-dc-b219-01
34	0.965471	172.17.17.220	224.0.0.252	LUNMR	74	Standard query 0x09fa AAAA eng-dc-b219-01
35	0.971093	HP_b8:96:ba	Broadcast	ARP	60	who has 172.17.16.93? Tell 172.17.16.100
36	0.974198	Dell_5f:cc:bf	Broadcast	ARP	60	who has 172.17.16.125? Tell 172.17.17.221
37	1.023583	172.17.16.134	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
38	1.066857	HP_b8:a1:c6	Broadcast	ARP	60	who has 169.254.169.254? Tell 172.17.16.84
39	1.078731	Micro-St_8d:18:c3	Broadcast	ARP	60	who has 169.254.169.254? Tell 172.17.16.180
40	1.156811	172.17.16.124	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1

2. Color coding of different protocols.

Name	Filter
<input checked="" type="checkbox"/> Bad TCP	tcp.analysis.flags && !tcp.analysis.window_update && !tcp.analysis.keep_alive && !tcp.analysis.keep_alive_ack
<input checked="" type="checkbox"/> HSRP State Change	hsrp.state != 8 && hsrp.state != 16
<input checked="" type="checkbox"/> Spanning Tree Topology Change	stp.type == 0x80
<input checked="" type="checkbox"/> OSPF State Change	ospf.msg != 1
<input checked="" type="checkbox"/> ICMP errors	icmp.type in { 3..5, 11 } icmpv6.type in { 1..4 }
<input checked="" type="checkbox"/> ARP	arp
<input checked="" type="checkbox"/> ICMP	icmp icmpv6
<input checked="" type="checkbox"/> TCP RST	tcp.flags.reset eq 1
<input checked="" type="checkbox"/> SCTP ABORT	sctp.chunk_type eq ABORT
<input checked="" type="checkbox"/> TTL low or unexpected	(ip.dst != 224.0.0.0/4 && ip.ttl < 5 && !ipm && !ospf) (ip.dst == 224.0.0.0/24 && ip.dst != 224.0.0.251 && ip.ttl != 1 && !(vrrp carp))
<input checked="" type="checkbox"/> Checksum Errors	eth.fcs.status == "Bad" ip.checksum.status == "Bad" tcp.checksum.status == "Bad" udp.checksum.status == "Bad" sctp.checksum.status == "Bad"
<input checked="" type="checkbox"/> SMB	smb nbss nbns netbios
<input checked="" type="checkbox"/> HTTP	http tcp.port == 80 http2
<input checked="" type="checkbox"/> DCERPC	dcerpc
<input checked="" type="checkbox"/> Routing	hsrp eigrp ospf bgp cdp vrrp carp gvrp igmp ismp
<input checked="" type="checkbox"/> TCP SYN/FIN	tcp.flags & 0x02 tcp.flags.fin == 1
<input checked="" type="checkbox"/> TCP	tcp
<input checked="" type="checkbox"/> UDP	udp
<input checked="" type="checkbox"/> Broadcast	eth[0] & 1
<input checked="" type="checkbox"/> System Event	systemd_journal sysdig

3. Statistics for the application layer protocol you have chosen.

DNS:

Wireshark · Packet 520 · cg.pcapng

```
> Frame 520: 227 bytes on wire (1816 bits), 227 bytes captured (1816 bits) on interface \Device\NPF_{001318EF-BC80-4FC4-9895-7AA7D50E320}, id 0
> Ethernet II, Src: Cisco_66:d1:41 (b0:aa:77:66:d1:41), Dst: HewlettP_6d:28:04 (3c:52:82:6d:28:04)
> Internet Protocol Version 4, Src: 172.31.0.25, Dst: 172.17.16.129
> User Datagram Protocol, Src Port: 53, Dst Port: 49868
> Domain Name System (response)
  Transaction ID: 0x409e
  Flags: 0x8180 Standard query response, No error
  Questions: 1
  Answer RRs: 5
  Authority RRs: 0
  Additional RRs: 0
  Queries
  Answers
  [Request In: 519]
  [Time: 0.000578000 seconds]
```

```
0000 3c 52 82 6d 28 04 b0 aa 77 66 d1 41 08 00 45 00 <R·m(··· wf·A··E·
0010 00 d5 43 f2 40 00 7e 11 4f 5b ac 1f 00 19 ac 11 ··C·@·~· O[·····
0020 10 81 00 35 c2 cc 00 c1 54 a5 40 9e 81 00 00 01 ···5···· T·@····
0030 00 05 00 00 00 00 03 77 77 77 0f 6d 73 66 74 63 ····w ww·msftc
0040 6f 6e 6e 65 63 74 74 65 73 74 03 63 6f 6d 00 00 onnectte st·com·
0050 01 00 01 c0 0c 00 05 00 01 00 00 00 94 00 1d 08 ····w ww·msftc
0060 6e 63 73 69 2d 67 65 6f 0e 74 72 61 66 66 69 63 ncsi-geo ·traffic
0070 6d 61 6e 61 67 65 72 03 6e 65 74 00 c0 35 00 05 manager·net·5·
0080 00 01 00 00 00 6c 00 1d 03 77 77 77 08 6d 73 66 ····l·· www·msf
0090 74 6e 63 73 69 03 63 6f 6d 09 65 64 67 65 73 75 tncsi·co m·edgesu
00a0 69 74 65 c0 4d c0 5e 00 05 00 01 00 00 00 05 00 ite M·^· ····
00b0 12 05 61 31 39 36 31 02 67 32 06 61 6b 61 6d 61 ··a1961· g2·akama
00c0 69 c0 4d c0 87 00 01 00 01 00 00 00 27 00 04 17 i·M· ····
00d0 31 68 ab c0 87 00 01 00 01 00 00 00 27 00 04 17 1h· ····
00e0 31 68 aa 1h·
```

Wireshark · DNS · cg.pcapng

Topic / Item	Count	Average	Min Val	Max Val	Rate (ms)	Percent	Burst Rate	Burst Start
▼ Total Packets	26				0.0004	100%	0.1400	42.754
▼ rcode	26				0.0004	100.00%	0.1400	42.754
No error	26				0.0004	100.00%	0.1400	42.754
▼ opcodes	26				0.0004	100.00%	0.1400	42.754
Standard query	26				0.0004	100.00%	0.1400	42.754
▼ Query/Response	26				0.0004	100.00%	0.1400	42.754
Response	13				0.0002	50.00%	0.0700	42.784
Query	13				0.0002	50.00%	0.0700	42.754
▼ Query Type	26				0.0004	100.00%	0.1400	42.754
AAAA (IPv6 Address)	4				0.0001	15.38%	0.0400	42.790
A (Host Address)	22				0.0004	84.62%	0.1000	42.754
▼ Class	26				0.0004	100.00%	0.1400	42.754
IN	26				0.0004	100.00%	0.1400	42.754
▼ Service Stats	0				0.0000	100%	-	-
request-response time (msec)	12	1206.52	0.310000	11626.712891	0.0002		0.0700	42.784
no. of unsolicited responses	0				0.0000		-	-
no. of retransmissions	1				0.0000		0.0100	53.968
▼ Response Stats	0				0.0000	100%	-	-
no. of questions	26	1.00	1	1	0.0004		0.1400	42.784
no. of authorities	26	0.15	0	1	0.0004		0.1400	42.784
no. of answers	26	4.54	1	8	0.0004		0.1400	42.784
no. of additionals	26	0.00	0	0	0.0004		0.1400	42.784
▼ Query Stats	0				0.0000	100%	-	-
Qname Len	13	22.08	16	31	0.0002		0.0700	42.754
▼ Label Stats	0				0.0000		-	-
4th Level or more	4				0.0001		0.0200	51.229
3rd Level	9				0.0001		0.0600	42.754
2nd Level	0				0.0000		-	-
1st Level	0				0.0000		-	-
Payload size	26	103.73	34	191	0.0004	100%	0.1400	42.754

HTTP:

Wireshark · Packet 1898 · cg.pcapng

```

> Frame 1898: 165 bytes on wire (1320 bits), 165 bytes captured (1320 bits) on interface \Device\NPF_{001318EF-BC80-4FC4-9895-7AA77D50E320}, id 0
> Ethernet II, Src: HewlettP_6d:28:04 (3c:52:82:6d:28:04), Dst: Cisco_66:d1:41 (b0:aa:77:66:d1:41)
> Internet Protocol Version 4, Src: 172.17.16.129, Dst: 23.49.104.171
> Transmission Control Protocol, Src Port: 54012, Dst Port: 80, Seq: 1, Ack: 1, Len: 111
▼ Hypertext Transfer Protocol
  > GET /connecttest.txt HTTP/1.1\r\n
    Connection: Close\r\n
    User-Agent: Microsoft NCSI\r\n
    Host: www.msftconnecttest.com\r\n
    \r\n
    [Full request URI: http://www.msftconnecttest.com/connecttest.txt]
    [HTTP request 1/1]
    [Response in frame: 1924]

```

```

0000  b0 aa 77 66 d1 41 3c 52 82 6d 28 04 08 00 45 00  ..w.f.A<R.m(....E.
0010  00 97 a9 56 40 00 80 06 14 9c ac 11 10 81 17 31  ...V@.....1
0020  68 ab d2 fc 00 50 40 dd 5b 40 6c f2 cc 36 50 18  h...P@.[@l...6P-
0030  04 02 07 27 00 00 47 45 54 20 2f 63 6f 6e 6e 65  ....GE T /conne
0040  63 74 74 65 73 74 2e 74 78 74 20 48 54 54 50 2f  cttest.t xt HTTP/
0050  31 2e 31 0d 0a 43 6f 6e 6e 65 63 74 69 6f 6e 3a  1.1...Con nection:
0060  20 43 6c 6f 73 65 0d 0a 55 73 65 72 2d 41 67 65  Close.. User-Age
0070  6e 74 3a 20 4d 69 63 72 6f 73 6f 66 74 20 4e 43  nt: Micr osoft NC
0080  53 49 0d 0a 48 6f 73 74 3a 20 77 77 77 2e 6d 73  SI..Host : www.ms
0090  66 74 63 6f 6e 6e 65 63 74 74 65 73 74 2e 63 6f  ftconnec ttest.co
00a0  6d 0d 0a 0d 0a  m....

```

Wireshark · Packet Counter · cg.pcapng

Topic / Item	Count	Average	Min Val	Max Val	Rate (ms)	Percent	Burst Rate	Burst Start
▼ Total HTTP Packets	312				0.0030	100%	0.0400	74.205
Other HTTP Packets	0				0.0000	0.00%	-	-
▼ HTTP Response Packets	3				0.0000	0.96%	0.0100	13.893
??? : broken	0				0.0000	0.00%	-	-
5xx: Server Error	0				0.0000	0.00%	-	-
4xx: Client Error	0				0.0000	0.00%	-	-
3xx: Redirection	0				0.0000	0.00%	-	-
▼ 2xx: Success	3				0.0000	100.00%	0.0100	13.893
200 OK	3				0.0000	100.00%	0.0100	13.893
1xx: Informational	0				0.0000	0.00%	-	-
▼ HTTP Request Packets	309				0.0030	99.04%	0.0400	74.205
SEARCH	306				0.0030	99.03%	0.0400	75.682
GET	3				0.0000	0.97%	0.0100	13.577

Post Lab Question-Answers:**1. What is the difference between Wireshark software and NMAP software?**

Ans: Wireshark and Nmap are both popular network analysis tools, but they serve different purposes and have distinct features. Here's a breakdown of the key differences between the two:

1. Functionality:

- Wireshark: Wireshark is a network protocol analyzer that captures and analyzes network traffic in real-time. It allows you to inspect packets at a granular level, providing detailed information about network protocols, packet headers, and payload contents. Wireshark is primarily used for network troubleshooting, protocol analysis, and security auditing.
- Nmap: Nmap (Network Mapper) is a network scanning tool used for host discovery, port scanning, and service enumeration. It helps identify open ports, detect operating systems, and gather information about network services running on target systems. Nmap is commonly used for network reconnaissance, vulnerability assessment, and penetration testing.

2. Scope:

- Wireshark: Wireshark focuses on capturing and analyzing network traffic on a specific network interface or network segment. It provides a comprehensive view of the network communication between devices, allowing you to inspect individual packets and analyze network behavior.
- Nmap: Nmap is designed for scanning and mapping networks to identify hosts, open ports, and services. It can scan large networks or specific IP ranges to discover active hosts and gather information about their network services.

3. User Interface:

- Wireshark: Wireshark provides a graphical user interface (GUI) that allows users to interactively capture, filter, and analyze network traffic. It offers powerful filtering capabilities, customizable displays, and various analysis tools to dissect captured packets.
- Nmap: Nmap primarily operates from the command-line interface (CLI), although there are graphical frontends available. It provides a wide range of command-line options and scripting capabilities, making it highly flexible and suitable for automation and scripting purposes.

4. Use Cases:

- Wireshark: Wireshark is commonly used by network administrators, security analysts, and developers for troubleshooting network issues, analyzing network protocols, and investigating security incidents. It helps in diagnosing network performance problems, identifying malicious activities, and understanding network behavior.
- Nmap: Nmap is widely used by network administrators, security professionals, and ethical hackers for network exploration, vulnerability scanning, and penetration testing. It helps in identifying open ports, assessing network security, and discovering potential vulnerabilities in systems.

In summary, Wireshark is primarily used for capturing and analyzing network traffic, while Nmap focuses on network scanning and host discovery. Both tools have their own strengths and are often used together to gain a comprehensive understanding of network behavior and security.

2. At which of the OSI layer Wireshark runs?

Ans: Wireshark operates at the lowest three layers of the OSI (Open Systems Interconnection) model, namely the Physical layer, Data Link layer, and Network layer.

1. **Physical Layer:** Wireshark can capture and analyze raw network traffic at the physical layer, including bits and electrical signals transmitted over the network medium. However, it is important to note that Wireshark typically captures data after it has been converted into a more readable format by the network interface card (NIC).
2. **Data Link Layer:** Wireshark can dissect and interpret data link layer protocols such as Ethernet, Wi-Fi (802.11), and others. It can analyze the frames exchanged between devices on a local network, including MAC addresses, frame types, and error detection.
3. **Network Layer:** Wireshark can capture and analyze network layer protocols such as IP (Internet Protocol), ICMP (Internet Control Message Protocol), and routing protocols like OSPF (Open Shortest Path First). It can provide insights into IP addressing, routing information, and network behavior.

While Wireshark primarily focuses on these lower layers, it also has the capability to analyze higher-layer protocols such as TCP (Transmission Control Protocol), UDP (User Datagram Protocol), HTTP (Hypertext Transfer Protocol), and many others. This allows for comprehensive analysis of network traffic across multiple layers of the OSI model.

3. Just write down the names of the softwares which have similar functionality as Wireshark. (open source or proprietary)

Ans:

1. **tcpdump:** tcpdump is a command-line packet analyzer that captures and analyzes network traffic. It is available for various operating systems, including Linux, macOS, and Windows.
2. **Microsoft Network Monitor:** Microsoft Network Monitor is a proprietary packet analyzer developed by Microsoft. It allows for capturing and analyzing network traffic on Windows-based systems.
3. **Tshark:** Tshark is a command-line tool that is part of the Wireshark suite. It provides similar functionality to Wireshark but operates in a terminal environment, making it suitable for scripting and automation.
4. **Colasoft Capsa:** Capsa is a proprietary network analyzer that offers real-time packet capturing and analysis. It provides a graphical user interface and is available for Windows operating systems.
5. **NetworkMiner:** NetworkMiner is an open-source network forensic analysis tool that captures and parses network traffic. It focuses on extracting files, emails, and other artifacts from captured packets.

6. EtherApe: EtherApe is an open-source graphical network monitor that visualizes network activity in real-time. It provides a visual representation of network traffic flows and can be used for basic packet analysis.
7. PRTG Network Monitor: PRTG is a proprietary network monitoring tool that includes packet sniffing capabilities. It allows for capturing and analyzing network traffic, along with monitoring various network parameters.

Outcomes:

Enumerate the layers of the OSI model and TCP/IP model, their functions and Protocols.

Conclusion (based on the Results and outcomes achieved):

The experiment successfully explored application layer protocols using packet analysis with Wireshark. The insights gained from this analysis can contribute to network troubleshooting, performance optimization, and security enhancement.

References:

Books/ Journals/ Websites:

1. Behrouz A Forouzan, Data Communication and Networking, Tata Mc Graw hill, India, 4th Edition
2. A. S. Tanenbaum, "Computer Networks", 4th edition, Prentice Hall
3. Behrouz A Forouzan, Data Communication and Networking, Tata Mc Graw hill, India, 4th Edition
4. A. S. Tanenbaum, "Computer Networks", 4th edition, Prentice Hall