Assignment 7

- Problem Statement:
  Write a program in C++/C to analyze following packet formats captured through wireshark for wired network
  1. FTP 2. IP 3. TCP 4. UDP

- Objective:
  - To understand packet formats such as FTP, UDP, IP, TCP.
  - To capture packet formats in wireshark.

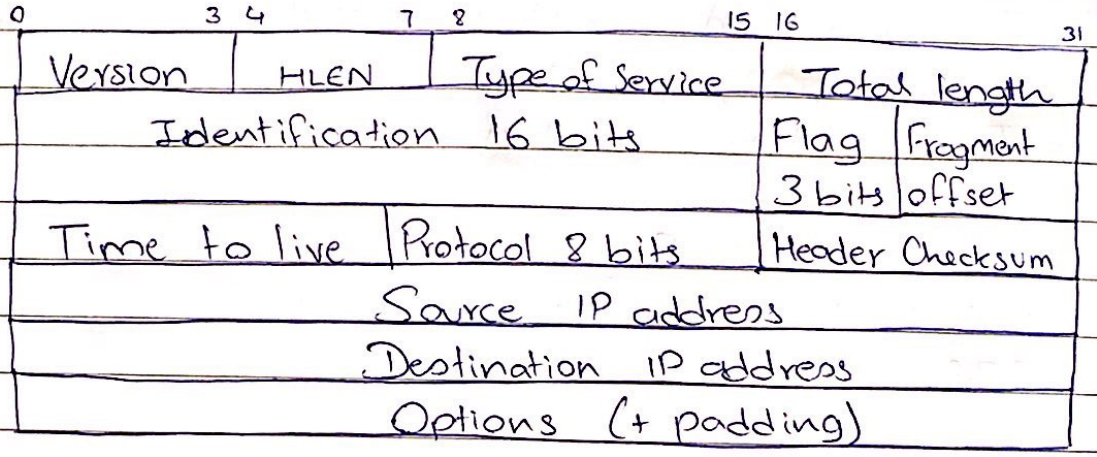
- S/W & H/W Req.
  1GB RAM, 120GB HDD, Monitor, keyboard, IDE.

- Theory:

-Packet analyzer:
It is a computer application used to track, intercept and log network traffic that passes over a digital network. It analyzes network traffic & generates a customized report to assist organization in managing their networks.

- Wireshark:
Wireshark is network packet analyzer A network packet, analyzer presents captured packet data in as much detail as possible. It is an open source software project.

Headers format:
IP Header format:



1) Version: (4 bits)
Version of IP.

2) HLEN (Header length : 4 bits)
no. of 32-bit words in IP header.

3) Type of service: (8 bits)
Importance level of datagram.

4) Total length (16 bits)
length of entire IP packet in bytes.

5) Identification field (16 bits)
Identifies current datagram

6) Flag bit (4 bits)
Indicates the parts of packet to the receiver

7) Time to live (8 bits):
At the end of this counter datagram is discarded

8) Protocol (8 bits):
Indicates which upper level protocol receives packets.

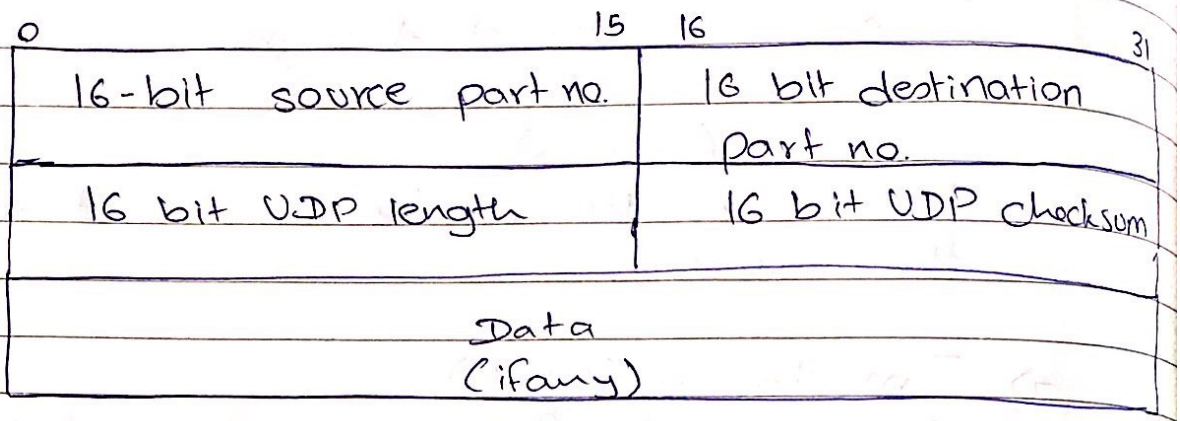
9) Header checksum (16 bits):
help ensure IP header integrity.

10) Source Address field (32 bits):
specifies sending node

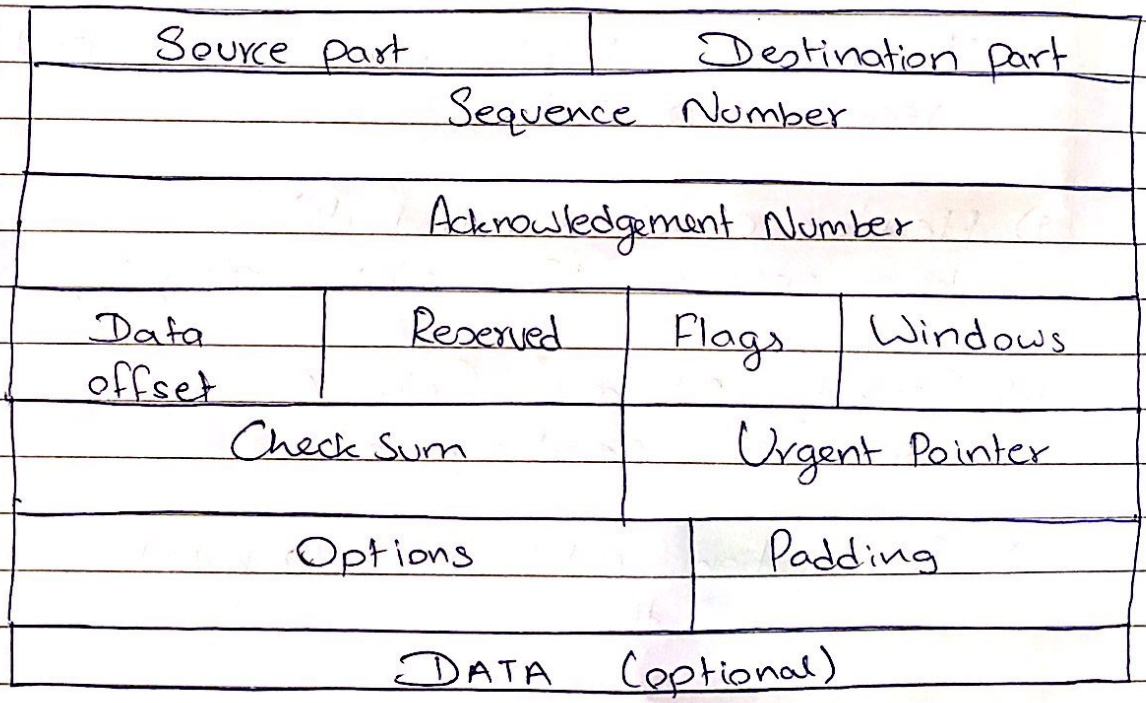
11) Destination Address (32 bits):
specifies the receiving code.

12) Options (32 bits)
allow IP to support various IP options.
Eg:- Security.

UDP header



TCP header:



1) Source port & Destination Port (16 bits each)
2) Sequence number (32 bits)
3) Acknowlegement No. field (32 bits)
4) Data offset (variable length)
5) Reserved field (6 bits)

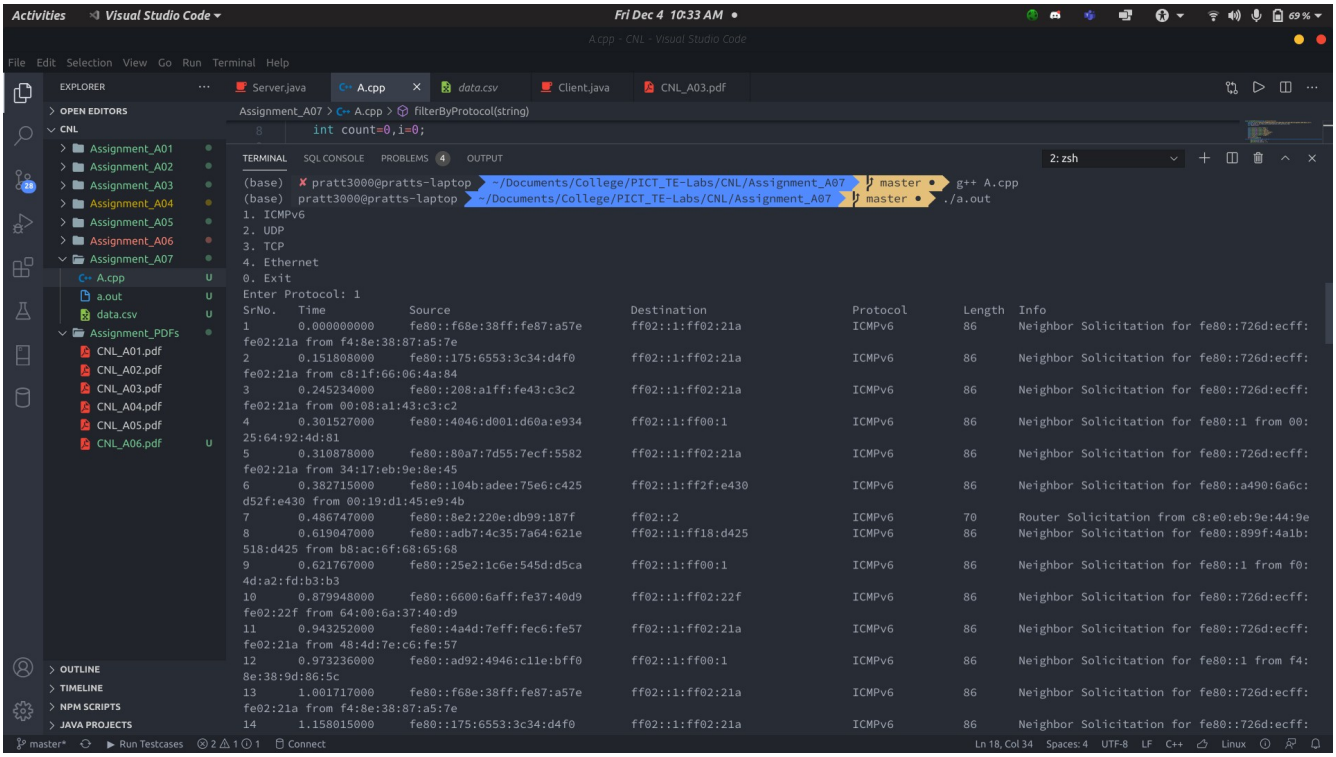
6) Flag field (6 bits)
7) Windows field (16 bits)
8) Checksum (16 bits)
9) Urgent pointer (16 bits)
10) Option field (variable lenghts)
11) Data fields (variable length)

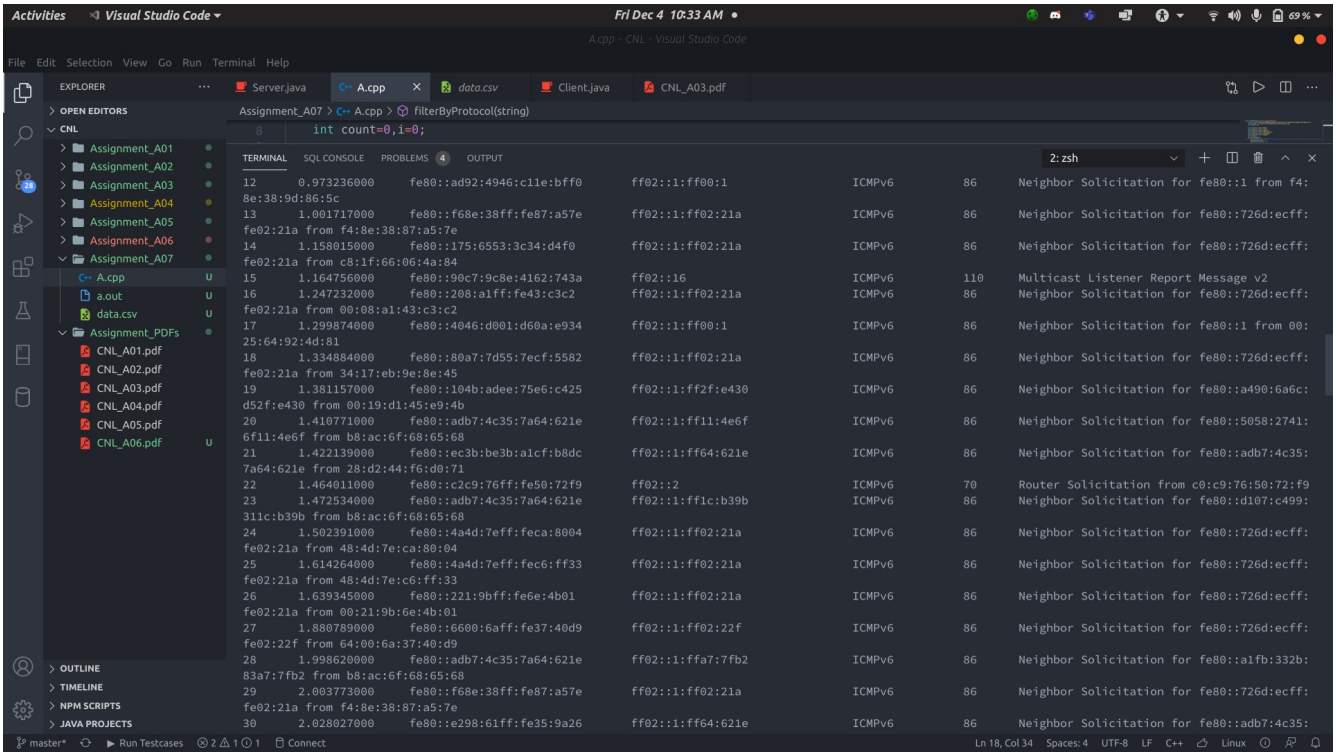
- Conclusion:-

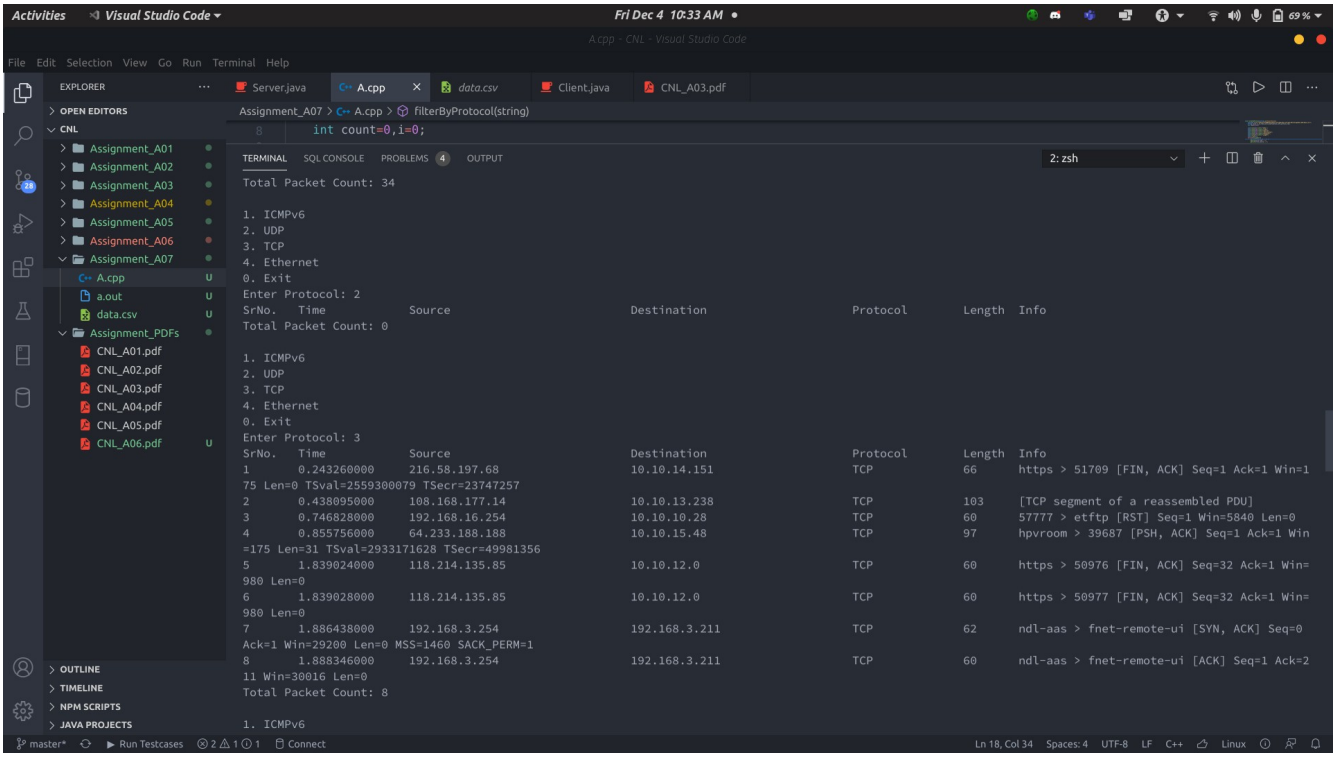
Thus, I successfully implemented to capture TCP, UDP, IP packets using wireshark.

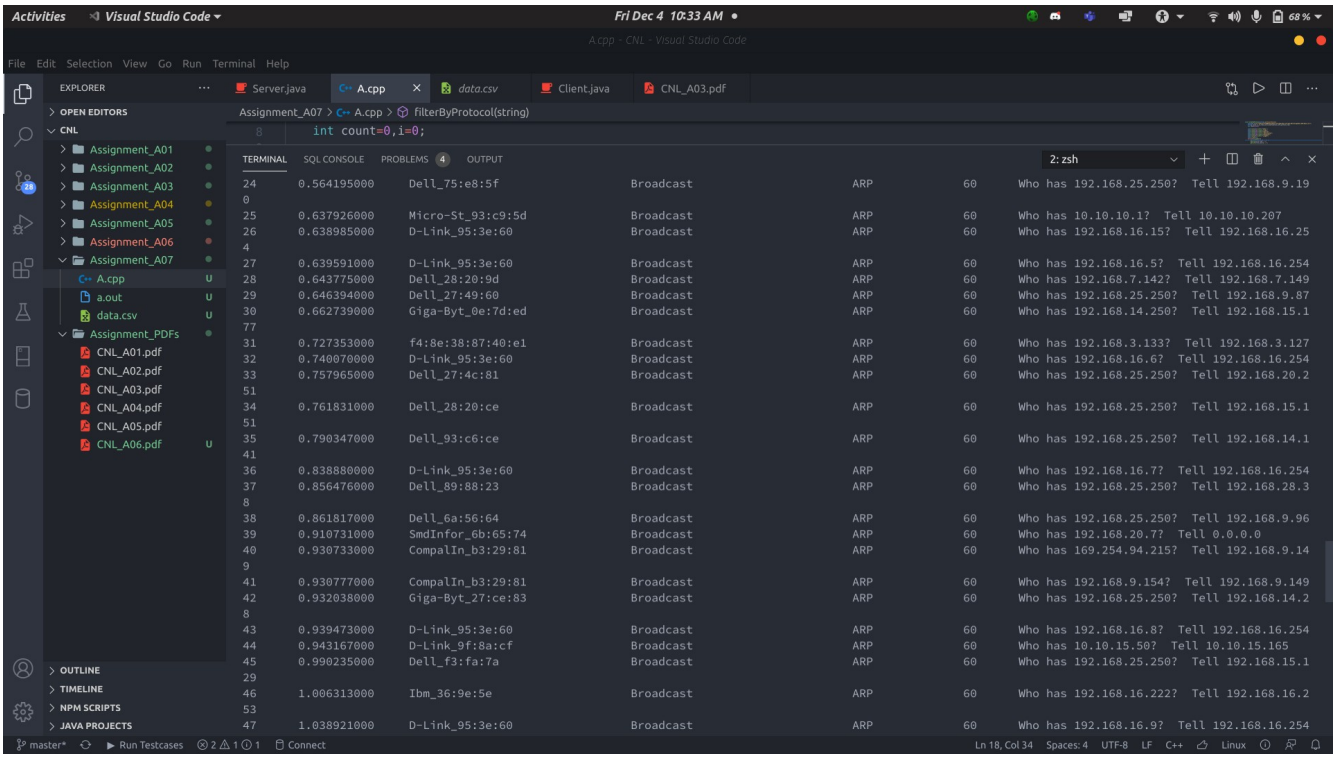
```
#include "bits/stdc++.h"

using namespace std;

void filterByProtocol(string protocolChoice){
   ifstream file("/home/pratt3000/Documents/College/PICT_TE-Labs/CNL/Assignment_A07/
data.csv");
   string value,sr_no,time,source,destination,info,protocol,len;
   int count=0,i=0;

   cout<<setw(8)<<left<<"SrNo.";
   cout<<setw(16)<<left<<"Time";
   cout<<setw(32)<<left<<"Source";
   cout<<setw(32)<<left<<"Destination";
   cout<<setw(16)<<left<<"Protocol";
   cout<<setw(8)<<left<<"Length";
   cout<<"Info\n";
   while(file.good()){
      getline(file,sr_no,',');
      getline(file,time,',');
      getline(file,source,',');
      getline(file,destination,',');
      getline(file,protocol,',');
      getline(file,len,',');
      getline(file,info,'\n');

      protocol=string(protocol,1,protocol.length()-2);

      if(protocol == protocolChoice){
         cout<<setw(8)<<left<<++i;
         cout<<setw(16)<<left<< string( time, 1, time.length()-2 );
         cout<<setw(32)<<left<<string( source, 1, source.length()-2 );
         cout<<setw(32)<<left<<string( destination, 1, destination.length()-2 );
         cout<<setw(16)<<left<<protocol;
         cout<<setw(8)<<left<< string( len, 1, len.length()-2 );
         cout<<string( info, 1, info.length()-2 )<<"\n";
         count++;
      }
   }
   file.close();
   cout<<"Total Packet Count: "<<count;
}

int main()
{
   int choice;

   cout<<"1. ICMPv6\n2. UDP\n3. TCP\n4. Ethernet\n0. Exit";
   cout<<"\nEnter Protocol: ";
```

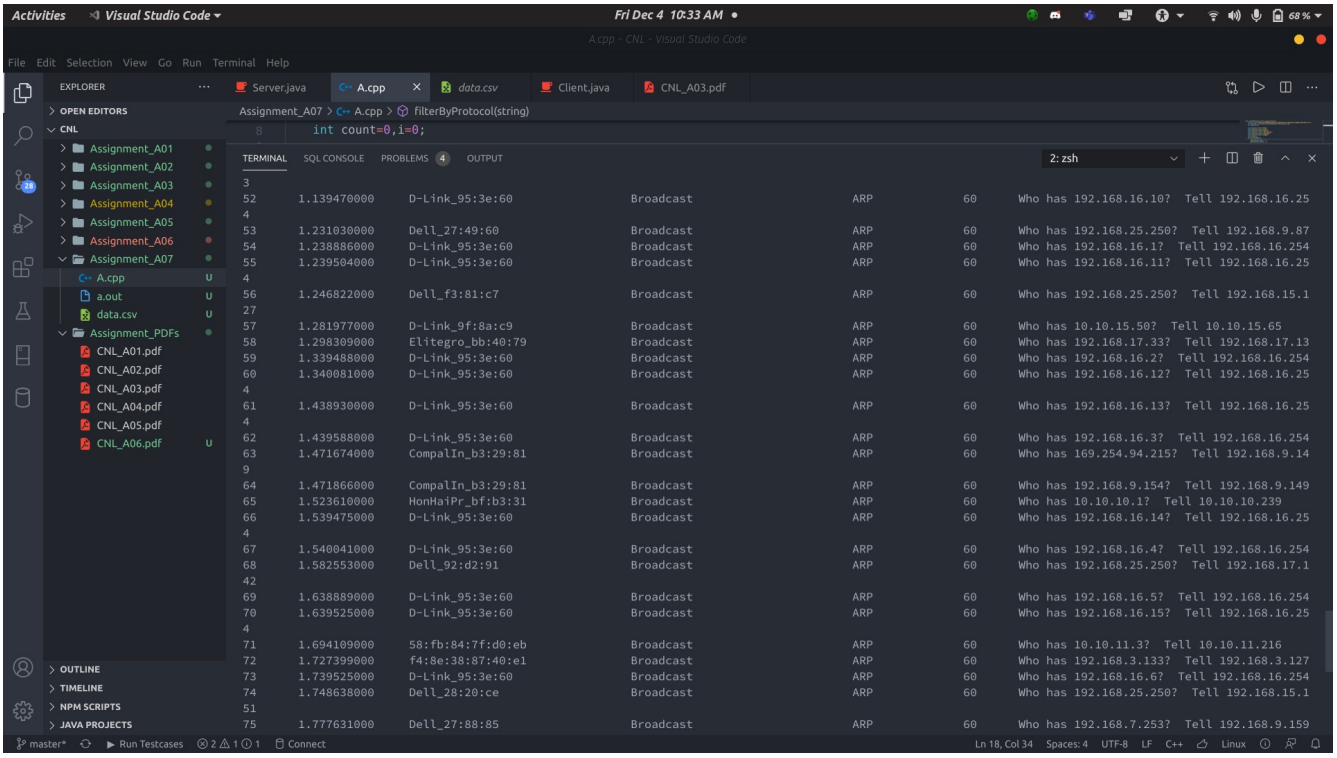
```
    cin>>choice;

    while (choice!=0){
        switch(choice){
            case 1: filterByProtocol("ICMPv6");
                break;
            case 2: filterByProtocol("UDP");
                break;
            case 3: filterByProtocol("TCP");
                break;
            case 4: filterByProtocol("ARP");
                break;
            default:
                cout<<"\nInvalid Protocol Choice";
                break;
        }
        cout<<"\n\n1. ICMPv6\n2. UDP\n3. TCP\n4. Ethernet\n0. Exit";
        cout<<"\nEnter Protocol: ";
        cin>>choice;
    }
    return 0;
}
```

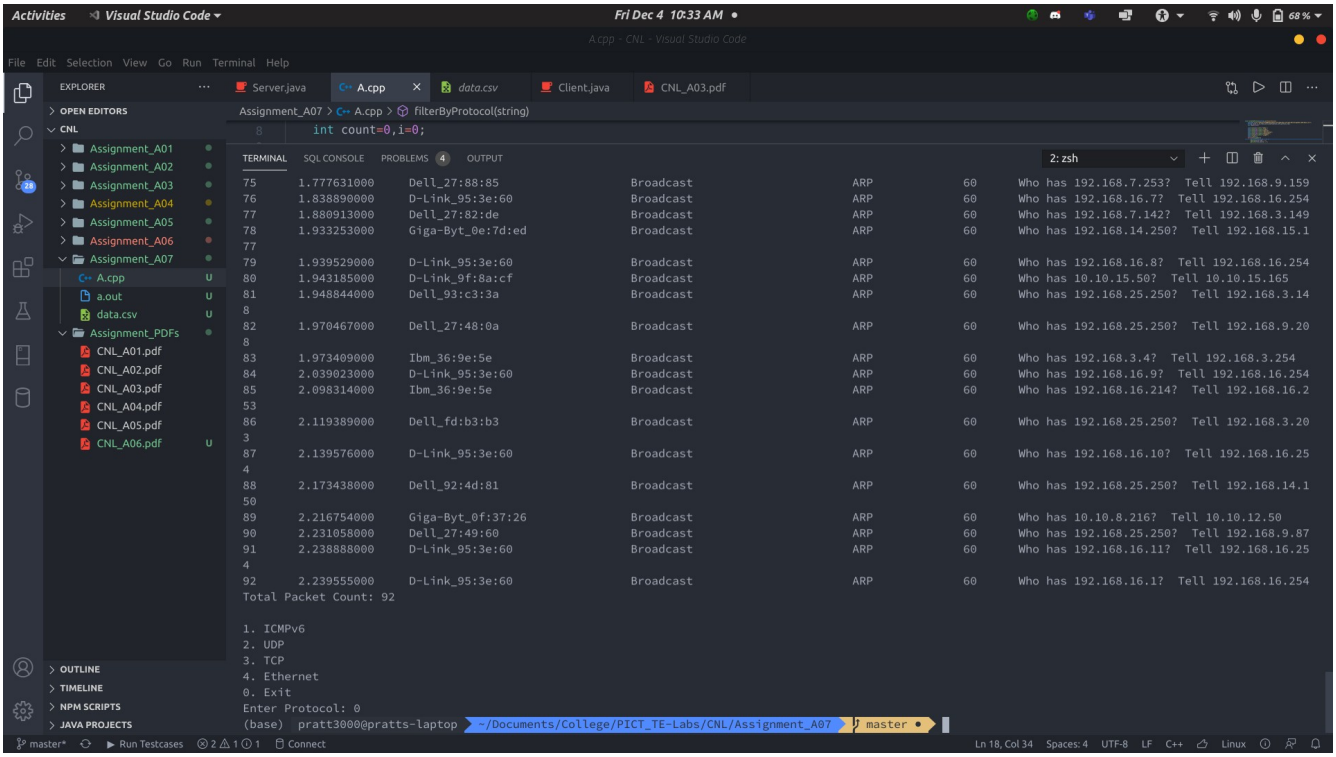



Activities
Visual Studio Code
Fri Dec 4 10:33 AM
A.cpp - CNL - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
OPEN EDITORS
CNL
Assignment_A01
Assignment_A02
Assignment_A03
Assignment_A04
Assignment_A05
Assignment_A06
Assignment_A07
A.cpp
a.out
data.csv
Assignment_PDFs
CNL_A01.pdf
CNL_A02.pdf
CNL_A03.pdf
CNL_A04.pdf
CNL_A05.pdf
CNL_A06.pdf
OUTLINE
TIMELINE
NPM SCRIPTS
JAVA PROJECTS
Server.java
A.cpp
data.csv
Client.java
CNL_A03.pdf
Assignment_A07 > A.cpp > filterByProtocol(string)
int count=0,i=0;
TERMINAL
SQL CONSOLE
PROBLEMS 4
OUTPUT
2: zsh
12 0.973236000 fe80::ad92:4946:c11e:bff0 ff02::1:ff00:1 ICMPv6 86 Neighbor Solicitation for fe80::1 from f4:8e:38:9d:86:5c
13 1.001717000 fe80::f68e:38ff:fe87:a57e ff02::1:ff02:21a ICMPv6 86 Neighbor Solicitation for fe80::726d:ecff:fe02:21a from f4:8e:38:87:a5:7e
14 1.158015000 fe80::175:6553:3c34:d4f0 ff02::1:ff02:21a ICMPv6 86 Neighbor Solicitation for fe80::726d:ecff:fe02:21a from c8:1f:66:06:4a:84
15 1.164756000 fe80::90c7:9c8e:4162:743a ff02::16 ICMPv6 110 Multicast Listener Report Message v2
16 1.247232000 fe80::208:a1ff:fe43:c3c2 ff02::1:ff02:21a ICMPv6 86 Neighbor Solicitation for fe80::726d:ecff:fe02:21a from 00:08:a1:43:c3:c2
17 1.299874000 fe80::4046:d001:d60a:e934 ff02::1:ff00:1 ICMPv6 86 Neighbor Solicitation for fe80::1 from 00:25:64:92:4d:81
18 1.334884000 fe80::80a7:7d55:7ecf:5582 ff02::1:ff02:21a ICMPv6 86 Neighbor Solicitation for fe80::726d:ecff:fe02:21a from 34:17:eb:9e:8e:45
19 1.381157000 fe80::104b:adee:75e6:c425 ff02::1:ff2f:e430 ICMPv6 86 Neighbor Solicitation for fe80::a490:6a6c:d52f:e430 from 00:19:d1:45:e9:4b
20 1.410771000 fe80::adb7:4c35:7a64:621e ff02::1:ff11:4e6f ICMPv6 86 Neighbor Solicitation for fe80::5058:2741:6f11:4e6f from b8:ac:6f:68:65:68
21 1.422139000 fe80::ec3b:be3b:a1cf:b8dc ff02::1:ff64:621e ICMPv6 86 Neighbor Solicitation for fe80::adb7:4c35:7a64:621e from 28:d2:44:f6:d0:71
22 1.464011000 fe80::c2c9:76ff:fe50:72f9 ff02::2 ICMPv6 70 Router Solicitation from c0:c9:76:50:72:f9
23 1.472534000 fe80::adb7:4c35:7a64:621e ff02::1:ff1c:b39b ICMPv6 86 Neighbor Solicitation for fe80::d107:c499:311c:b39b from b8:ac:6f:68:65:68
24 1.502391000 fe80::4a4d:7eff:feca:8004 ff02::1:ff02:21a ICMPv6 86 Neighbor Solicitation for fe80::726d:ecff:fe02:21a from 48:4d:7e:ca:80:04
25 1.614264000 fe80::4a4d:7eff:fec6:ff33 ff02::1:ff02:21a ICMPv6 86 Neighbor Solicitation for fe80::726d:ecff:fe02:21a from 48:4d:7e:c6:ff:33
26 1.639345000 fe80::221:9bff:fe6e:4b01 ff02::1:ff02:21a ICMPv6 86 Neighbor Solicitation for fe80::726d:ecff:fe02:21a from 00:21:9b:6e:4b:01
27 1.880789000 fe80::6600:6aff:fe37:40d9 ff02::1:ff02:22f ICMPv6 86 Neighbor Solicitation for fe80::726d:ecff:fe02:22f from 64:00:6a:37:40:d9
28 1.998620000 fe80::adb7:4c35:7a64:621e ff02::1:ffa7:7fb2 ICMPv6 86 Neighbor Solicitation for fe80::a1fb:332b:83a7:7fb2 from b8:ac:6f:68:65:68
29 2.003773000 fe80::f68e:38ff:fe87:a57e ff02::1:ff02:21a ICMPv6 86 Neighbor Solicitation for fe80::726d:ecff:fe02:21a from f4:8e:38:87:a5:7e
30 2.028027000 fe80::e298:61ff:fe35:9a26 ff02::1:ff64:621e ICMPv6 86 Neighbor Solicitation for fe80::adb7:4c35:
master*
Run Testcases
Connect
Ln 18, Col 34
Spaces: 4
UTF-8
LF
C++
Linux


Activities
Visual Studio Code
Fri Dec 4 10:33 AM
A.cpp - CNL - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
OPEN EDITORS
CNL
Assignment_A01
Assignment_A02
Assignment_A03
Assignment_A04
Assignment_A05
Assignment_A06
Assignment_A07
A.cpp
a.out
data.csv
Assignment_PDFs
CNL_A01.pdf
CNL_A02.pdf
CNL_A03.pdf
CNL_A04.pdf
CNL_A05.pdf
CNL_A06.pdf
OUTLINE
TIMELINE
NPM SCRIPTS
JAVA PROJECTS
Server.java
A.cpp
data.csv
Client.java
CNL_A03.pdf
Assignment_A07 > A.cpp > filterByProtocol(string)
int count=0,i=0;
TERMINAL
SQL CONSOLE
PROBLEMS 4
OUTPUT
2: zsh
Total Packet Count: 34
1. ICMPv6
2. UDP
3. TCP
4. Ethernet
0. Exit
Enter Protocol: 2
SrNo. Time Source Destination Protocol Length Info
Total Packet Count: 0
1. ICMPv6
2. UDP
3. TCP
4. Ethernet
0. Exit
Enter Protocol: 3
SrNo. Time Source Destination Protocol Length Info
1 0.243260000 216.58.197.68 10.10.14.151 TCP 66 https > 51709 [FIN, ACK] Seq=1 Ack=1 Win=175 Len=0 TSval=2559300079 TSecr=23747257
2 0.438095000 108.168.177.14 10.10.13.238 TCP 103 [TCP segment of a reassembled PDU]
3 0.746828000 192.168.16.254 10.10.10.28 TCP 60 57777 > etftp [RST] Seq=1 Win=5840 Len=0
4 0.855756000 64.233.188.188 10.10.15.48 TCP 97 hpvroom > 39687 [PSH, ACK] Seq=1 Ack=1 Win=175 Len=31 TSval=2933171628 TSecr=49981356
5 1.839024000 118.214.135.85 10.10.12.0 TCP 60 https > 50976 [FIN, ACK] Seq=32 Ack=1 Win=980 Len=0
6 1.839028000 118.214.135.85 10.10.12.0 TCP 60 https > 50977 [FIN, ACK] Seq=32 Ack=1 Win=980 Len=0
7 1.886438000 192.168.3.254 192.168.3.211 TCP 62 ndl-aas > fnet-remote-ui [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1
8 1.888346000 192.168.3.254 192.168.3.211 TCP 60 ndl-aas > fnet-remote-ui [ACK] Seq=1 Ack=211 Win=30016 Len=0
Total Packet Count: 8
1. ICMPv6
master*
Run Testcases
Connect
Ln 18, Col 34
Spaces: 4
UTF-8
LF
C++
Linux


Activities
Visual Studio Code
Fri Dec 4 10:33 AM
A.cpp - CNL - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
OPEN EDITORS
CNL
Assignment_A01
Assignment_A02
Assignment_A03
Assignment_A04
Assignment_A05
Assignment_A06
Assignment_A07
A.cpp
a.out
data.csv
Assignment_PDFs
CNL_A01.pdf
CNL_A02.pdf
CNL_A03.pdf
CNL_A04.pdf
CNL_A05.pdf
CNL_A06.pdf
OUTLINE
TIMELINE
NPM SCRIPTS
JAVA PROJECTS
Server.java
A.cpp
data.csv
Client.java
CNL_A03.pdf
Assignment_A07 > A.cpp > filterByProtocol(string)
int count=0,i=0;
TERMINAL
SQL CONSOLE
PROBLEMS 4
OUTPUT
2: zsh
24 0.564195000 Dell_75:e8:5f Broadcast ARP 60 Who has 192.168.25.250? Tell 192.168.9.19
0
25 0.637926000 Micro-St_93:c9:5d Broadcast ARP 60 Who has 10.10.10.1? Tell 10.10.10.207
26 0.638985000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.15? Tell 192.168.16.25
4
27 0.639591000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.5? Tell 192.168.16.254
28 0.643775000 Dell_28:20:9d Broadcast ARP 60 Who has 192.168.7.142? Tell 192.168.7.149
29 0.646394000 Dell_27:49:60 Broadcast ARP 60 Who has 192.168.25.250? Tell 192.168.9.87
30 0.662739000 Giga-Byt_0e:7d:ed Broadcast ARP 60 Who has 192.168.14.250? Tell 192.168.15.1
77
31 0.727353000 f4:8e:38:87:40:e1 Broadcast ARP 60 Who has 192.168.3.133? Tell 192.168.3.127
32 0.740070000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.6? Tell 192.168.16.254
33 0.757965000 Dell_27:4c:81 Broadcast ARP 60 Who has 192.168.25.250? Tell 192.168.20.2
51
34 0.761831000 Dell_28:20:ce Broadcast ARP 60 Who has 192.168.25.250? Tell 192.168.15.1
51
35 0.790347000 Dell_93:c6:ce Broadcast ARP 60 Who has 192.168.25.250? Tell 192.168.14.1
41
36 0.838880000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.7? Tell 192.168.16.254
37 0.856476000 Dell_89:88:23 Broadcast ARP 60 Who has 192.168.25.250? Tell 192.168.28.3
8
38 0.861817000 Dell_6a:56:64 Broadcast ARP 60 Who has 192.168.25.250? Tell 192.168.9.96
39 0.910731000 SmdInfor_6b:65:74 Broadcast ARP 60 Who has 192.168.20.7? Tell 0.0.0.0
40 0.930733000 CompalIn_b3:29:81 Broadcast ARP 60 Who has 169.254.94.215? Tell 192.168.9.14
9
41 0.930777000 CompalIn_b3:29:81 Broadcast ARP 60 Who has 192.168.9.154? Tell 192.168.9.149
42 0.932038000 Giga-Byt_27:ce:83 Broadcast ARP 60 Who has 192.168.25.250? Tell 192.168.14.2
8
43 0.939473000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.8? Tell 192.168.16.254
44 0.943167000 D-Link_9f:8a:cf Broadcast ARP 60 Who has 10.10.15.50? Tell 10.10.15.165
45 0.990235000 Dell_f3:fa:7a Broadcast ARP 60 Who has 192.168.25.250? Tell 192.168.15.1
29
46 1.006313000 Ibm_36:9e:5e Broadcast ARP 60 Who has 192.168.16.222? Tell 192.168.16.2
53
47 1.038921000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.9? Tell 192.168.16.254
master*
Run Testcases
Connect
Ln 18, Col 34
Spaces: 4
UTF-8
LF
C++
Linux


Activities
Visual Studio Code
Fri Dec 4 10:33 AM
A.cpp - CNL - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
OPEN EDITORS
CNL
Assignment_A01
Assignment_A02
Assignment_A03
Assignment_A04
Assignment_A05
Assignment_A06
Assignment_A07
A.cpp
a.out
data.csv
Assignment_PDFs
CNL_A01.pdf
CNL_A02.pdf
CNL_A03.pdf
CNL_A04.pdf
CNL_A05.pdf
CNL_A06.pdf
OUTLINE
TIMELINE
NPM SCRIPTS
JAVA PROJECTS
Server.java
A.cpp
data.csv
Client.java
CNL_A03.pdf
Assignment_A07 > A.cpp > filterByProtocol(string)
int count=0,i=0;
TERMINAL
SQL CONSOLE
PROBLEMS 4
OUTPUT
2: zsh
3
52 1.139470000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.10? Tell 192.168.16.25
4
53 1.231030000 Dell_27:49:60 Broadcast ARP 60 Who has 192.168.25.250? Tell 192.168.9.87
54 1.238886000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.1? Tell 192.168.16.254
55 1.239504000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.11? Tell 192.168.16.25
4
56 1.246822000 Dell_f3:81:c7 Broadcast ARP 60 Who has 192.168.25.250? Tell 192.168.15.1
27
57 1.281977000 D-Link_9f:8a:c9 Broadcast ARP 60 Who has 10.10.15.50? Tell 10.10.15.65
58 1.298309000 Elitegro_bb:40:79 Broadcast ARP 60 Who has 192.168.17.33? Tell 192.168.17.13
59 1.339488000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.2? Tell 192.168.16.254
60 1.340081000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.12? Tell 192.168.16.25
4
61 1.438930000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.13? Tell 192.168.16.25
4
62 1.439588000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.3? Tell 192.168.16.254
63 1.471674000 CompalIn_b3:29:81 Broadcast ARP 60 Who has 169.254.94.215? Tell 192.168.9.14
9
64 1.471866000 CompalIn_b3:29:81 Broadcast ARP 60 Who has 192.168.9.154? Tell 192.168.9.149
65 1.523610000 HonHaiPr_bf:b3:31 Broadcast ARP 60 Who has 10.10.10.1? Tell 10.10.10.239
66 1.539475000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.14? Tell 192.168.16.25
4
67 1.540041000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.4? Tell 192.168.16.254
68 1.582553000 Dell_92:d2:91 Broadcast ARP 60 Who has 192.168.25.250? Tell 192.168.17.1
42
69 1.638889000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.5? Tell 192.168.16.254
70 1.639525000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.15? Tell 192.168.16.25
4
71 1.694109000 58:fb:84:7f:d0:eb Broadcast ARP 60 Who has 10.10.11.3? Tell 10.10.11.216
72 1.727399000 f4:8e:38:87:40:e1 Broadcast ARP 60 Who has 192.168.3.133? Tell 192.168.3.127
73 1.739525000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.6? Tell 192.168.16.254
74 1.748638000 Dell_28:20:ce Broadcast ARP 60 Who has 192.168.25.250? Tell 192.168.15.1
51
75 1.777631000 Dell_27:88:85 Broadcast ARP 60 Who has 192.168.7.253? Tell 192.168.9.159
master*
Run Testcases
Connect
Ln 18, Col 34
Spaces: 4
UTF-8
LF
C++
Linux


Activities
Visual Studio Code
Fri Dec 4 10:33 AM
A.cpp - CNL - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
OPEN EDITORS
CNL
Assignment_A01
Assignment_A02
Assignment_A03
Assignment_A04
Assignment_A05
Assignment_A06
Assignment_A07
A.cpp
a.out
data.csv
Assignment_PDFs
CNL_A01.pdf
CNL_A02.pdf
CNL_A03.pdf
CNL_A04.pdf
CNL_A05.pdf
CNL_A06.pdf
OUTLINE
TIMELINE
NPM SCRIPTS
JAVA PROJECTS
Server.java
A.cpp
data.csv
Client.java
CNL_A03.pdf
Assignment_A07 > A.cpp > filterByProtocol(string)
8 int count=0,i=0;
TERMINAL
SQL CONSOLE
PROBLEMS 4
OUTPUT
2: zsh
75 1.777631000 Dell_27:88:85 Broadcast ARP 60 Who has 192.168.7.253? Tell 192.168.9.159
76 1.838890000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.7? Tell 192.168.16.254
77 1.880913000 Dell_27:82:de Broadcast ARP 60 Who has 192.168.7.142? Tell 192.168.3.149
78 1.933253000 Giga-Byt_0e:7d:ed Broadcast ARP 60 Who has 192.168.14.250? Tell 192.168.15.1
77
79 1.939529000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.8? Tell 192.168.16.254
80 1.943185000 D-Link_9f:8a:cf Broadcast ARP 60 Who has 10.10.15.50? Tell 10.10.15.165
81 1.948844000 Dell_93:c3:3a Broadcast ARP 60 Who has 192.168.25.250? Tell 192.168.3.14
8
82 1.970467000 Dell_27:48:0a Broadcast ARP 60 Who has 192.168.25.250? Tell 192.168.9.20
8
83 1.973409000 Ibm_36:9e:5e Broadcast ARP 60 Who has 192.168.3.4? Tell 192.168.3.254
84 2.039023000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.9? Tell 192.168.16.254
85 2.098314000 Ibm_36:9e:5e Broadcast ARP 60 Who has 192.168.16.214? Tell 192.168.16.2
53
86 2.119389000 Dell_fd:b3:b3 Broadcast ARP 60 Who has 192.168.25.250? Tell 192.168.3.20
3
87 2.139576000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.10? Tell 192.168.16.25
4
88 2.173438000 Dell_92:4d:81 Broadcast ARP 60 Who has 192.168.25.250? Tell 192.168.14.1
50
89 2.216754000 Giga-Byt_0f:37:26 Broadcast ARP 60 Who has 10.10.8.216? Tell 10.10.12.50
90 2.231058000 Dell_27:49:60 Broadcast ARP 60 Who has 192.168.25.250? Tell 192.168.9.87
91 2.238888000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.11? Tell 192.168.16.25
4
92 2.239555000 D-Link_95:3e:60 Broadcast ARP 60 Who has 192.168.16.1? Tell 192.168.16.254
Total Packet Count: 92
1. ICMPv6
2. UDP
3. TCP
4. Ethernet
0. Exit
Enter Protocol: 0
(base) pratt3000@pratts-laptop ~/Documents/College/PICT_TE-Labs/CNL/Assignment_A07 master
master* Run Testcases Connect
Ln 18, Col 34 Spaces: 4 UTF-8 LF C++ Linux