Government Engineering College, Thrissur

CS334 – Network Programming Lab

Documentation -

Exp 7 – Raw sockets and Sniffing

Date of Submission

30 May 2021

Submitted By

Kowsik Nandagopan D

Roll No 31

TCR18CS031

GECT CSE S6

Experiment 7

a)Display the header information of UDP and TCP packets during client-server communication using raw sockets.

b) Develop a packet capturing and filtering application using raw sockets.

Executing program

Part a

- Code is provided in the **raw_tcp.c** and **raw_udp.c.** (Tested and verified on Ubuntu 20.04)

```
gcc raw_tcp.c -o raw_tcp && sudo ./raw_tcp
```

```
gcc raw_udp.c -o raw_udp && sudo ./raw_udp
```

- **Note: Raw sockets must be executed in *sudo* mode.**
- **Since we cannot multitask same time, we are looping the send command infinte time. To stop the program, use CTRL + C**

TCP Raw Sockets Algorithm

- TCP raw sockets are created
- Create new datagram and allocate it in memory
- Assign the parameters of headers that include IP header, TCP header, and data
- Checksum can be included to validate the integrity of the information
- Forward the data from the server to the client
- Exit (We can also loop through the action if required)

P. T. O

UDP Raw Sockets Algorithm

- UDP raw sockets are created
- Create new buffer and allocate it in memory
- Assign the parameters of headers that include IP header, UDP header, and data
- Checksum can be included to validate the integrity of the information
- Forward the data from the server to the client
- Exit (We can also loop through the action if required)

Part b

- Code is provided in the **sniffer.c.** (Tested and verified on Ubuntu 20.04)

```
gcc sniffer.c -o sniffer && sudo ./sniffer
```

- **Note: Raw sockets must be executed in *sudo* mode.**

Algorithm

- Create buffers and allocate dynamically
- Create web sockets
- Process the packets based on the packet received.
- Iphrd = size of buffer + ethhdr
- If the TCP is received print the TCP header information, apply the same action to UDP

P. T. O

Output / Screenshots

Part a

Raw TCP

```
└─$ gcc raw_tcp.c -o raw_tcp && sudo ./raw_tcp
Source IP : 192.168.1.2
Packet Send. Length : 46
Source IP : 192.168.1.2
Packet Send. Length : 46
```

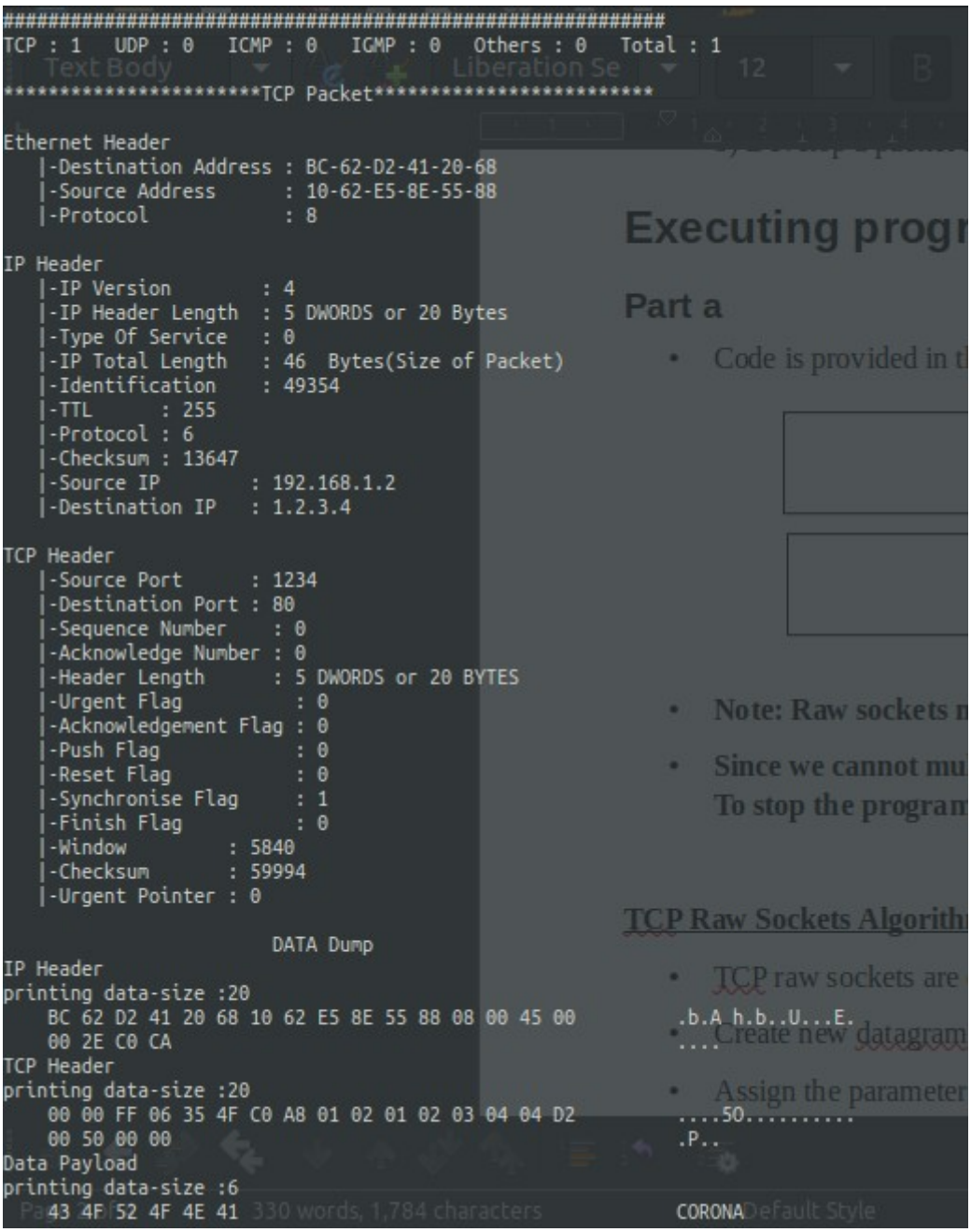
Raw UDP

```
┌─hp@hp ~/Documents/S6/Network Lab/Exp 7 <m
└─$ gcc raw_udp.c -o raw_udp && sudo ./raw_
raw_udp.c: In function ‘main’:
raw_udp.c:96:24: warning: implicit declarat
   96 |  sin.sin_addr.s_addr = inet_addr ("
      |                        ^~~~~~~~~~
Source IP : 192.168.1.2
Packet Send. Length : 34
Source IP : 192.168.1.2
Packet Send. Length : 34
Source IP : 192.168.1.2
Packet Send. Length : 34
Source IP : 192.168.1.2
Packet Send. Length : 34
Source IP : 192.168.1.2
```

P. T. O

Part b

TCP Header captured by sniffer



```
############################################################
TCP : 1   UDP : 0   ICMP : 0   IGMP : 0   Others : 0   Total : 1
**********************TCP Packet*************************

Ethernet Header
   |-Destination Address : BC-62-D2-41-20-68
   |-Source Address      : 10-62-E5-8E-55-88
   |-Protocol            : 8

IP Header
   |-IP Version         : 4
   |-IP Header Length   : 5 DWORDS or 20 Bytes
   |-Type Of Service    : 0
   |-IP Total Length    : 46  Bytes(Size of Packet)
   |-Identification     : 49354
   |-TTL      : 255
   |-Protocol : 6
   |-Checksum : 13647
   |-Source IP        : 192.168.1.2
   |-Destination IP   : 1.2.3.4

TCP Header
   |-Source Port      : 1234
   |-Destination Port : 80
   |-Sequence Number    : 0
   |-Acknowledge Number : 0
   |-Header Length      : 5 DWORDS or 20 BYTES
   |-Urgent Flag          : 0
   |-Acknowledgement Flag : 0
   |-Push Flag            : 0
   |-Reset Flag           : 0
   |-Synchronise Flag     : 1
   |-Finish Flag          : 0
   |-Window         : 5840
   |-Checksum       : 59994
   |-Urgent Pointer : 0

                        DATA Dump
IP Header
printing data-size :20
    BC 62 D2 41 20 68 10 62 E5 8E 55 88 08 00 45 00      .b.A h.b..U...E.
    00 2E C0 CA                                          ....
TCP Header
printing data-size :20
    00 00 FF 06 35 4F C0 A8 01 02 01 02 03 04 04 D2      ....5O..........
    00 50 00 00                                          .P..
Data Payload
printing data-size :6
    43 4F 52 4F 4E 41                                    CORONA
```

P. T. O

UDP Header captured by sniffer