Lab1 Report

Name: Haoyang Zhang

Email: hzhang11@usc.edu

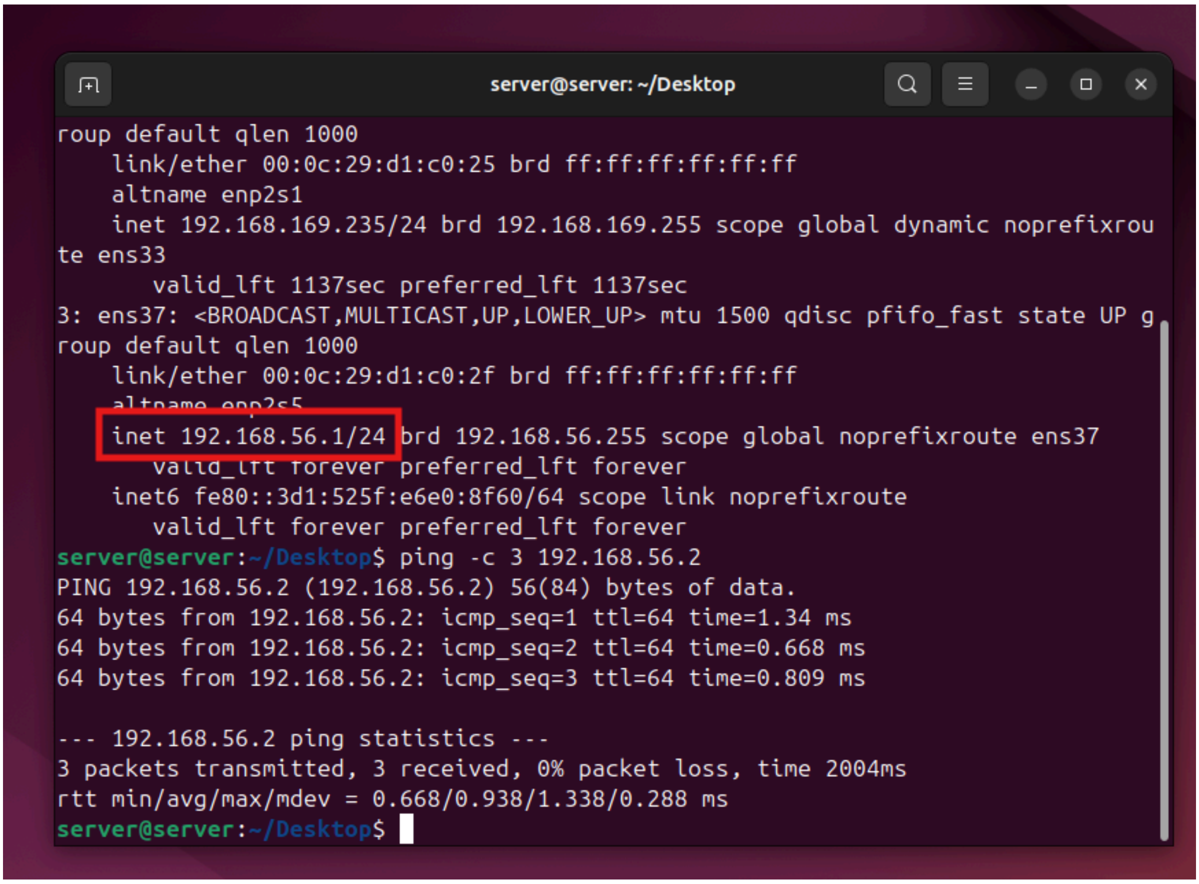
Remote repository address: https://github.com/hzhang2422/EE-533

Step 1. Set up visual machines and configure network

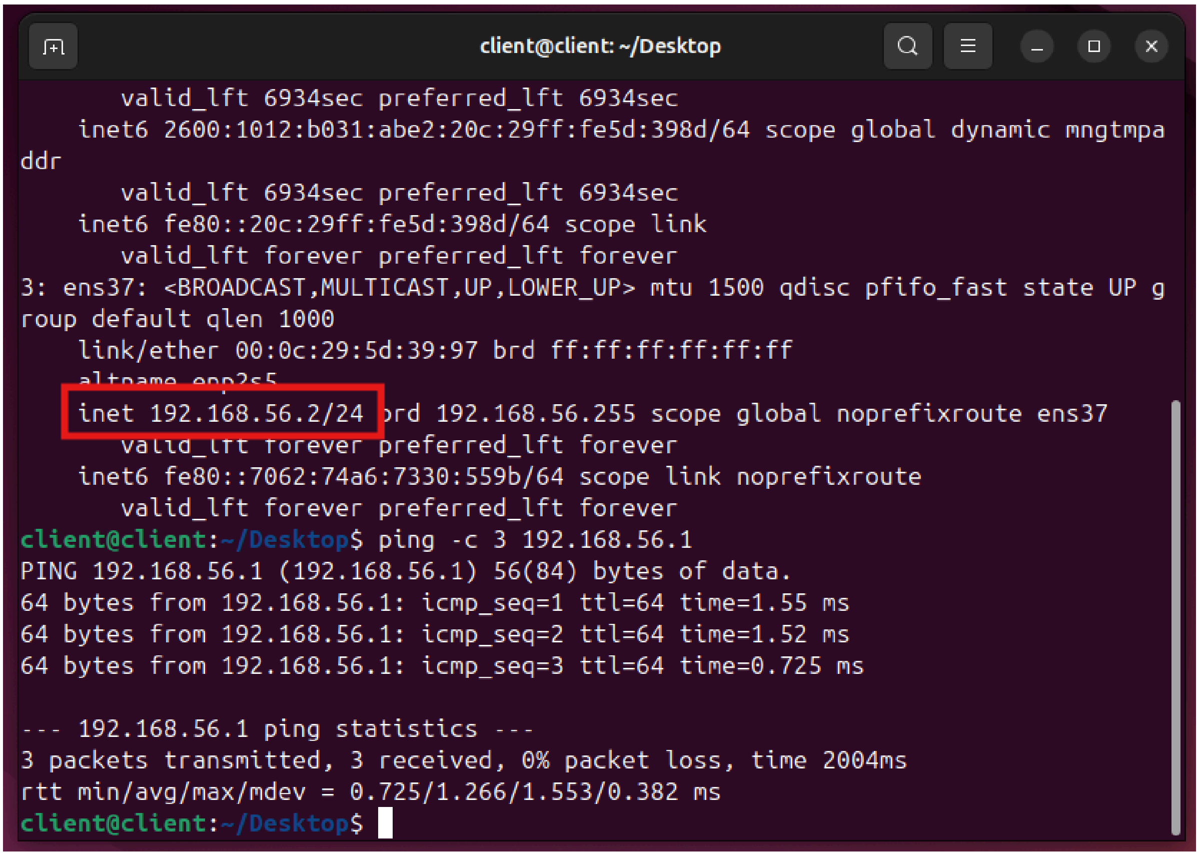
The ip address of server is 192.168.56.1 .

The ip address of client is 192.168.56.2 .

And the network between them is connected.



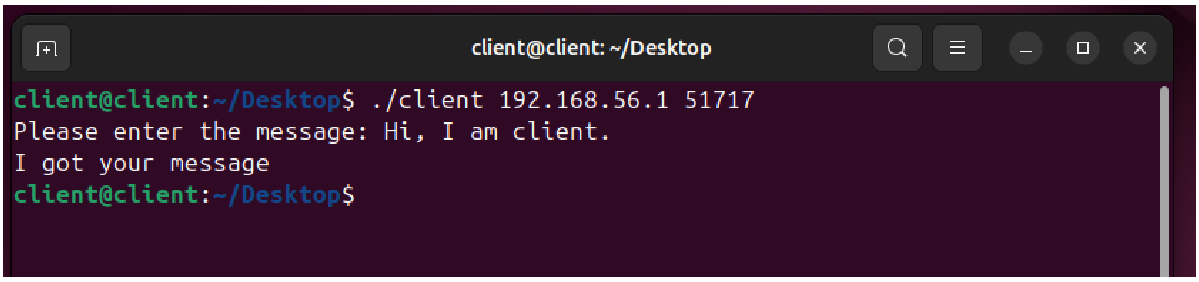
```
roup default qlen 1000
    link/ether 00:0c:29:d1:c0:25 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.169.235/24 brd 192.168.169.255 scope global dynamic noprefixrou
te ens33
       valid_lft 1137sec preferred_lft 1137sec
3: ens37: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP g
roup default qlen 1000
    link/ether 00:0c:29:d1:c0:2f brd ff:ff:ff:ff:ff:ff
    altname enp2s5
    inet 192.168.56.1/24 brd 192.168.56.255 scope global noprefixroute ens37
       valid_lft forever preferred_lft forever
    inet6 fe80::3d1:525f:e6e0:8f60/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
server@server:~/Desktop$ ping -c 3 192.168.56.2
PING 192.168.56.2 (192.168.56.2) 56(84) bytes of data.
64 bytes from 192.168.56.2: icmp_seq=1 ttl=64 time=1.34 ms
64 bytes from 192.168.56.2: icmp_seq=2 ttl=64 time=0.668 ms
64 bytes from 192.168.56.2: icmp_seq=3 ttl=64 time=0.809 ms

--- 192.168.56.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 0.668/0.938/1.338/0.288 ms
server@server:~/Desktop$ 
```



```
       valid_lft 6934sec preferred_lft 6934sec
    inet6 2600:1012:b031:abe2:20c:29ff:fe5d:398d/64 scope global dynamic mngtmpa
ddr
       valid_lft 6934sec preferred_lft 6934sec
    inet6 fe80::20c:29ff:fe5d:398d/64 scope link
       valid_lft forever preferred_lft forever
3: ens37: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP g
roup default qlen 1000
    link/ether 00:0c:29:5d:39:97 brd ff:ff:ff:ff:ff:ff
    altname enp2s5
    inet 192.168.56.2/24 brd 192.168.56.255 scope global noprefixroute ens37
       valid_lft forever preferred_lft forever
    inet6 fe80::7062:74a6:7330:559b/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
client@client:~/Desktop$ ping -c 3 192.168.56.1
PING 192.168.56.1 (192.168.56.1) 56(84) bytes of data.
64 bytes from 192.168.56.1: icmp_seq=1 ttl=64 time=1.55 ms
64 bytes from 192.168.56.1: icmp_seq=2 ttl=64 time=1.52 ms
64 bytes from 192.168.56.1: icmp_seq=3 ttl=64 time=0.725 ms

--- 192.168.56.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 0.725/1.266/1.553/0.382 ms
client@client:~/Desktop$
```

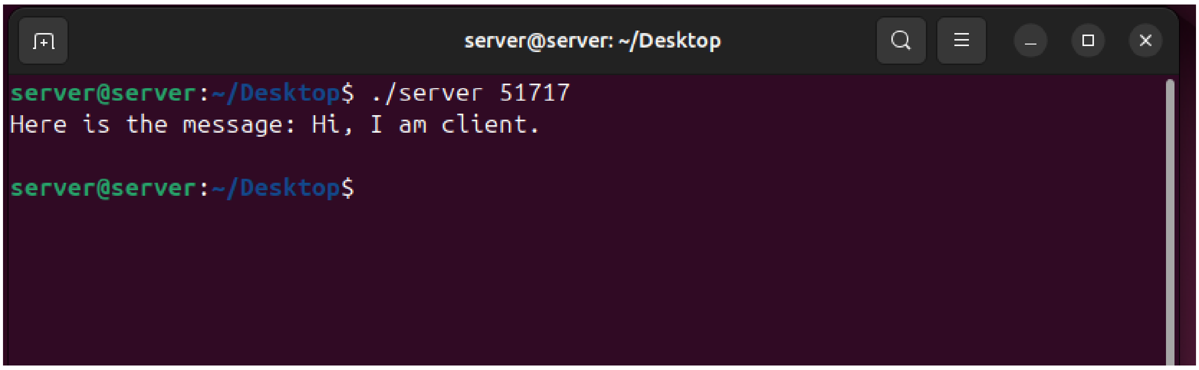
Step 2. Compile and test the sample code

Client:



```
client@client:~/Desktop$ ./client 192.168.56.1 51717
Please enter the message: Hi, I am client.
I got your message
client@client:~/Desktop$
```

Server:



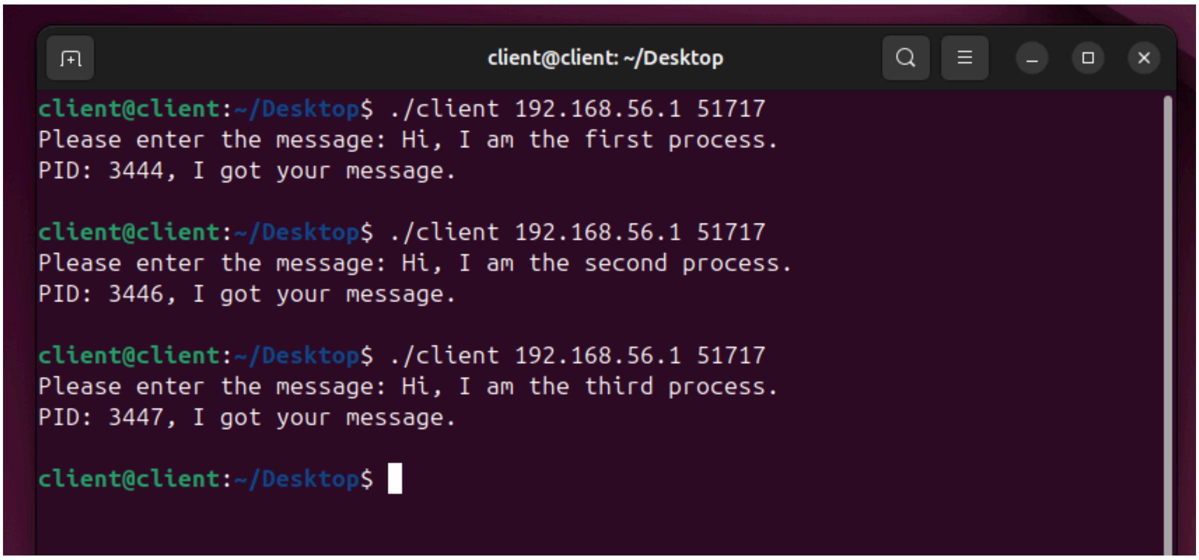
```
server@server:~/Desktop$ ./server 51717
Here is the message: Hi, I am client.

server@server:~/Desktop$
```

Step 3. Enhance the server code and test

Server should run indefinitely and should have the capability of handling a number of simultaneous connections, each in its own process.

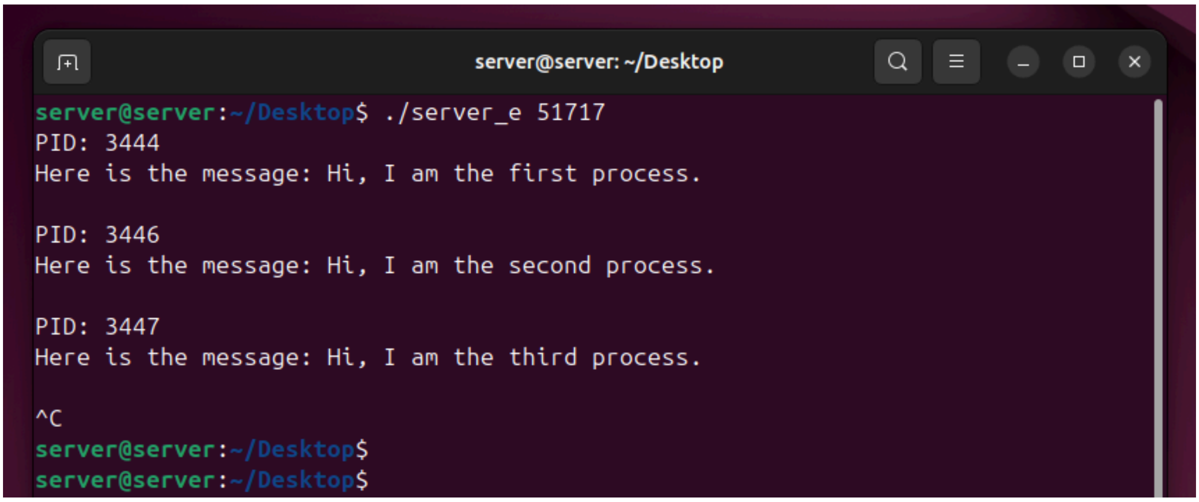
And I completed this part of the code according to the instructions to enhance the function and solve the zombie problem.

Client:



```
client@client:~/Desktop$ ./client 192.168.56.1 51717
Please enter the message: Hi, I am the first process.
PID: 3444, I got your message.

client@client:~/Desktop$ ./client 192.168.56.1 51717
Please enter the message: Hi, I am the second process.
PID: 3446, I got your message.

client@client:~/Desktop$ ./client 192.168.56.1 51717
Please enter the message: Hi, I am the third process.
PID: 3447, I got your message.

client@client:~/Desktop$ 
```

Server:

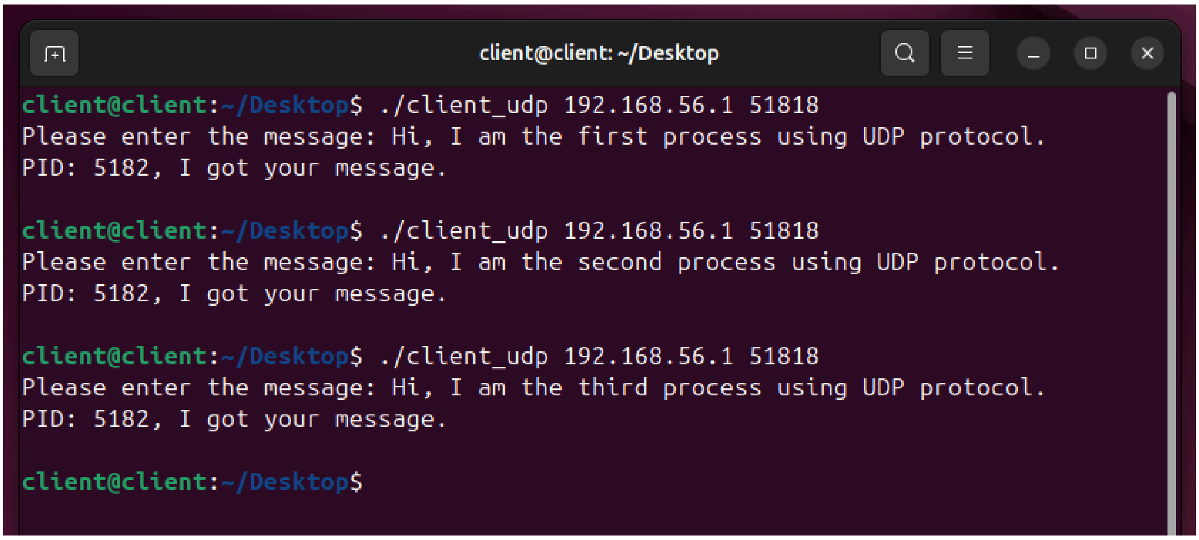


```
server@server:~/Desktop$ ./server_e 51717
PID: 3444
Here is the message: Hi, I am the first process.

PID: 3446
Here is the message: Hi, I am the second process.

PID: 3447
Here is the message: Hi, I am the third process.

^C
server@server:~/Desktop$
server@server:~/Desktop$
```

Step 4. Use datagram socket type and UDP protocol

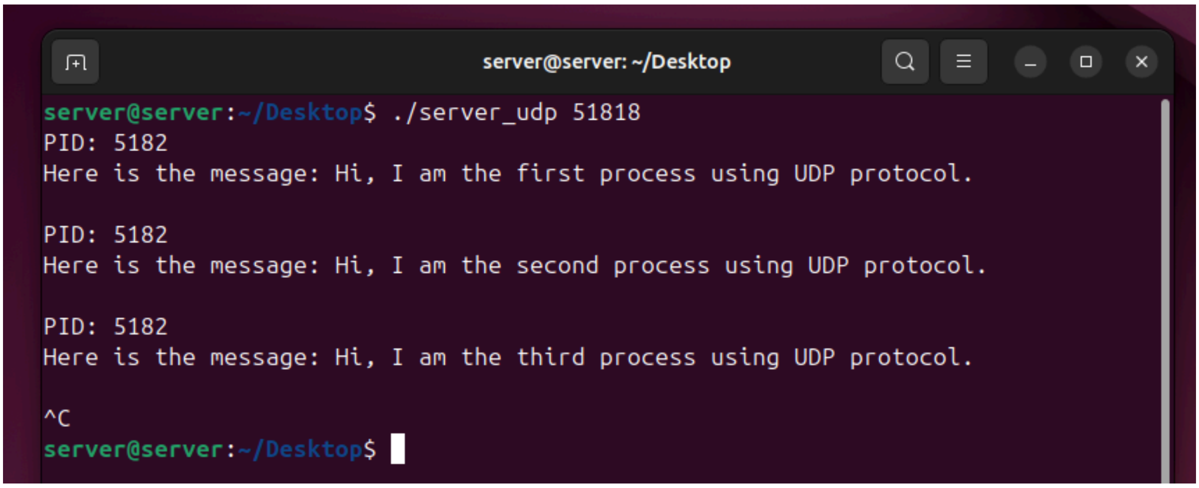
I replaced the write() and read() with sendto() and receivefrom() function according to the instructions and used the SOCK_DGRAM.

Client:



```
client@client:~/Desktop$ ./client_udp 192.168.56.1 51818
Please enter the message: Hi, I am the first process using UDP protocol.
PID: 5182, I got your message.

client@client:~/Desktop$ ./client_udp 192.168.56.1 51818
Please enter the message: Hi, I am the second process using UDP protocol.
PID: 5182, I got your message.

client@client:~/Desktop$ ./client_udp 192.168.56.1 51818
Please enter the message: Hi, I am the third process using UDP protocol.
PID: 5182, I got your message.

client@client:~/Desktop$
```

Server:



```
server@server:~/Desktop$ ./server_udp 51818
PID: 5182
Here is the message: Hi, I am the first process using UDP protocol.

PID: 5182
Here is the message: Hi, I am the second process using UDP protocol.

PID: 5182
Here is the message: Hi, I am the third process using UDP protocol.

^C
server@server:~/Desktop$
```