Vipul Sharma B17069
Swapnil Rustagi B17104

System Practicum
CS307
Assignment 4
Report

Video link: **https://youtu.be/Zwg16L5MeZg**

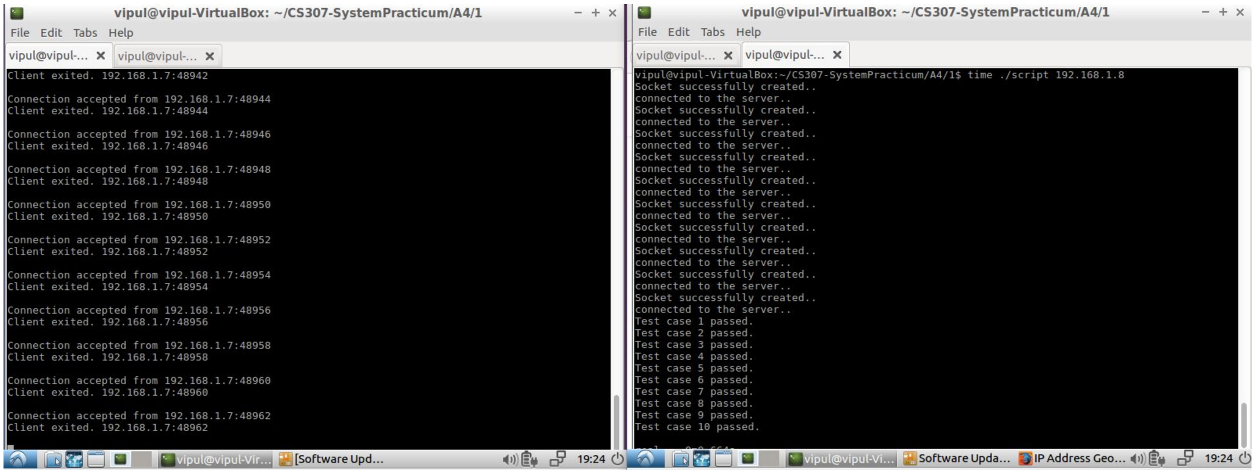
Sol 1.

Aim -

Write a client server application in which client sends a string to the server and the server sends back the string after changing cases of each alphabet (lower case to upper case and vice verse). Implement it over one PC utilizing two VMs (Virtual Machines), both act as client as well as server ie both VMs can send string to other VM and both can reply to incoming strings. The program should work for as many inputs as given don't show it for single input only. Write a script to test your application. The script should simulate large number of clients and cover different test cases and produce a proper report.

Procedure -

1. Compile using **make**. This will generate 3 executables - **script, server and client.**
2. If you just want to test manually on 2 VMs run ./server on your server VM and run ./client YOUR_SERVER_IP on your client VM.
3. Both can run simultaneously on both the VMs. Just take care of the server IP you put as the command line argument.
4. Send exit from the client to exit the program anytime.
5. To run automated script tests on the client, run ./script YOUR_SERVER_IP on your client VM. This script will perform these tasks-
    a. Generate random input strings in input/ directory.
    b. Generate corresponding outputs in expected_output/ directory.
    c. Run the test case using the client program by sending requests to the server and generate the output in output/ directory.
    d. Compare the test cases in output/ and expected_output/ directory to see if the cases pass or fail.



```
Client exited. 192.168.1.7:48942

Connection accepted from 192.168.1.7:48944
Client exited. 192.168.1.7:48944

Connection accepted from 192.168.1.7:48946
Client exited. 192.168.1.7:48946

Connection accepted from 192.168.1.7:48948
Client exited. 192.168.1.7:48948

Connection accepted from 192.168.1.7:48950
Client exited. 192.168.1.7:48950

Connection accepted from 192.168.1.7:48952
Client exited. 192.168.1.7:48952

Connection accepted from 192.168.1.7:48954
Client exited. 192.168.1.7:48954

Connection accepted from 192.168.1.7:48956
Client exited. 192.168.1.7:48956

Connection accepted from 192.168.1.7:48958
Client exited. 192.168.1.7:48958

Connection accepted from 192.168.1.7:48960
Client exited. 192.168.1.7:48960

Connection accepted from 192.168.1.7:48962
Client exited. 192.168.1.7:48962
```

```
vipul@vipul-VirtualBox:~/CS307-SystemPracticum/A4/1$ time ./script 192.168.1.8
Socket successfully created..
connected to the server..
Socket successfully created..
connected to the server..
Socket successfully created..
connected to the server..
Socket successfully created..
connected to the server..
Socket successfully created..
connected to the server..
Socket successfully created..
connected to the server..
Socket successfully created..
connected to the server..
Socket successfully created..
connected to the server..
Socket successfully created..
connected to the server..
Socket successfully created..
connected to the server..
Test case 1 passed.
Test case 2 passed.
Test case 3 passed.
Test case 4 passed.
Test case 5 passed.
Test case 6 passed.
Test case 7 passed.
Test case 8 passed.
Test case 9 passed.
Test case 10 passed.
```

Results -

All 10 test cases passed.

Total time taken for running the script = 0.664 s

Total characters processed = Number of characters per string x Number of strings per file x
Number of files = 1000 x 10 x 10
= 100000

Characters processed per second = 150602

Sol 2.

Aim-

Design a file transfer application in client-server architect which supports multiple clientssimultaneously. Client sends file name to the server which checks its local disk for thefile, if found it will send the file to the requesting client. Use sufficiently large files andcheck the difference between original file and the file that client has received.(use scpto copy the same file on client or store the original file on client beforehand for testing)Client can also ask server for its usage details (list of files client has downloaded sofar, size of data transferred etc). Usage details are client specific. Connection shouldterminate properly after completion of file transfer. If file is not found on server, servershould inform this to client. Think how will you differentiate between different types ofmessages (request for file, file not found,request for usage details etc). Make it a reliabletransfer. (no packet loss and in-order packet reception)

Implementation

Transmission Control Protocol used (SOCK_STREAM option in socket() system call) for reliable, in-order packet reception.

Procedure of communication -

Server obtains a socket and binds to port 3000, and listens for new connections. When a client tries to connect, “accept()” creates provides a file descriptor for the connection (connection is uniquely identified by source IP, source port, destination IP, destination port, and protocol), and a new thread is created and this file descriptor is passed on to the thread.

For implementing file transfer and usage statistics, the server first sends the size of the file/length of string having usage statistics. Then client knows how many bytes to expect.

Sol 3.

a) DNS - DNS or a Domain Name Server is a server that acts like a phonebook for the clients on the internet. Everytime we put in a domain name in a browser, the request actually goes to a DNS Server, which then provides the IP Address corresponding to that particular domain name.

DHCP - DHCP or Dynamic Host Configuration Protocol is a network management protocol used on IP Networks. A DHCP server dynamically assigns IP Addresses to nodes on a network. It allows the nodes to request IP Addresses from the router or the ISP, reducing the need for manual IP Addresses.

b) TCP - TCP or Transmission Control Protocol is a connection-oriented protocol, i.e. connection is first established before transmitting data. It is a reliable data delivery protocol. Packets arrive in order at the server. Retransmission of lost packets occurs. Used in HTTP, HTTPS, FTP, SMTP etc.

UDP - UDP or User Datagram Protocol is a datagram oriented protocol. Unlike TCP, there is no need to establish a connection in this case and also maintain and terminate it. It is preferred for broadcast or multicast transmissions. It is an unreliable data delivery protocol. No ordering of packets is required. No retransmission of lost packets occurs. Used in DNS, DHCP, VoIP etc.

c) The OSI Layer model was developed in the year 1984 and used in most modern day networks. In this 7 layer model, each layer has a separate function to perform and all these layers work together to transmit data from one node to another over a network. These layers are Physical, Data Link, Network, Transport, Session, Presentation and Application Layers.

d)

```
vipul@vipul:~/CS307-SystemPracticum/A4/1$ ifconfig
enp0s3    Link encap:Ethernet  HWaddr 08:00:27:60:ea:c1
          inet addr:192.168.1.7  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::dfc6:3a18:df69:8b58/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:87885 errors:0 dropped:5 overruns:0 frame:0
          TX packets:13605 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:30317696 (30.3 MB)  TX bytes:8611453 (8.6 MB)
```

IP Address from ifconfig - 192.168.1.7
IP Address from ip2location - 117.207.52.35

The reason for this difference is that there are a finite number of IPs in the world. Thus, there is a need for routers to which my computer is connected to. The IP address 192.168.1.7 is assigned to my computer by the router and the IP Address 117.207.52.35 is assigned to me by my ISP - BSNL. This same IP Address may be assigned to many people across North India as the location shown on ip2location is Ambala, Haryana.

e) 0.0.0.0 is used to designate an unknown or invalid node on a network. It has different uses in different contexts. For example, to send a request from a node to DHCP to get an IP Address assigned, a packet is sent to 0.0.0.0 along with the MAC Address of the node.

255.255.255.255 is the broadcast address which is a network address used to transmit to all devices connected on the network.

f)

```
vipul@vipul:~/CS307-SystemPracticum/A4/1$ traceroute harvard.edu
traceroute to harvard.edu (23.185.0.1), 30 hops max, 60 byte packets
 1  192.168.1.1 (192.168.1.1)  1.948 ms  1.920 ms  1.876 ms
 2  117.207.52.1 (117.207.52.1)  1.847 ms * *
 3  * * *
 4  218.248.107.134 (218.248.107.134)  2.848 ms * *
 5  * 117.216.207.202 (117.216.207.202)  9.703 ms *
 6  117.216.207.203 (117.216.207.203)  12.683 ms  12.500 ms *
 7  * * 115.110.78.173 (115.110.78.173)  12.446 ms
 8  * 172.23.183.161 (172.23.183.161)  30.739 ms *
 9  * * 172.28.132.245 (172.28.132.245)  32.192 ms
```

192.168.1.1 - Router IP Address
117.207.52.1 - ISP IP Address BSNL - Ambala
218.248.107.134 - ISP Node BSNL - Delhi
117.216.207.203 - ISP Node BSNL - Bengaluru
115.110.78.173 - Inter-country Network Node by TATA Communications - Delhi
172.23.183.161 - Private IP Address LAN
172.28.132.245 - Private IP Address LAN
23.185.0.1 - Public Proxy Server harvard.edu - California, USA