

Socket Programming

There are three files, namely, bootpsvr.cpp, tcpclient.cpp, filesvr.cpp.

Part A

Bootpsvr.cpp

- Bootstrap server that accepts the serving information from different servers in the sequence - MSGTYPE:REGISTRN IP address, port number, servicename(png/jpeg/mp4/txt/ or any other file extension), and a service access token on UDP.
- To compile : g++ bootpsvr.cpp -o bs
- To run : ./bs

```
supriya@supriya-Super-Server:~/Dropbox/ACN/assgn3$ g++ bootpsvr.cpp -o bs
supriya@supriya-Super-Server:~/Dropbox/ACN/assgn3$ ./bs
Enter the ip address and port number
to listen the connections for
1234
Server listening.....
```

fileserver.cpp

- Firstly it registers to bootstrap server on UDP with the details given in step1.
- To compile : g++ fileserver.cpp -o fs
- To run : ./fs

```

supriya@supriya-Super-Server:~/Dropbox/ACN/assgn3$ ./fs
Enter the ip address and port number of bootpserver
1234
Enter details: MSGTYPE(REGISTRN,DISCOVERY), IP address, port number, servicename, service access token or simply type 'bye' to end
REGISTRN,192.168.126.137,1236,pdf,server1
sent to bootp REGISTRN,192.168.126.137,1236,pdf,server1
server1
message sent successfully
AT is -server1
All done closing socket now

```

Part B

fileserver.cpp

- Once it registers to the Bootstrap server successfully in Part A, it listens on TCP socket.

Server listening on TCP port -

```

supriya@supriya-Super-Server:~/Dropbox/ACN/assgn3$ ./fs
Enter the ip address and port number of bootpserver
1234
Enter details: MSGTYPE(REGISTRN,DISCOVERY), IP address, port number, servicename, service access token or simply type 'bye' to end
REGISTRN,192.168.126.137,1236,pdf,server1
sent to bootp REGISTRN,192.168.126.137,1236,pdf,server1
server1
message sent successfully
AT is -server1
All done closing socket now
Server server1
Enter the port no
1236
Server socket created successfully.
Listening.....

```

tcpclient.cpp

- client interacts with Bootstrap server on UDP and file servers on TCP. servicename, IP address, port number, and a service access token.
- To compile : g++ tcpclient.cpp -o client
- To run : ./client

- Client connects to bootpserver and get the information of all running servers.

```
supriya@supriya-Super-Server:~/Dropbox/ACN/assgn3$ ./client
Enter the ip address and port number of bootpserver
1234
Enter message to send to server in  or simply type 'bye' to end
DISCOVERY
message sent successfully
Message from server: 192.168.126.137,1236,pdf,server1
ip - 192.168.126.137
0 1236
1 pdf
2 server1
received server infos
Message from server: end
All done closing socket now
All done closing udp bootpsocket now
connecting to server 192.168.126.137 at port 1236
ACCESS TOKEN -server1 server type pdf
Server socket created successfully.
Connected to Server.
want to change the access token
```

- After that client sends the access token to server and server verifies and confirms that client is valid or not. Then the client sends the filename it wants from the server. Server sends the file and the client exists.

CONSOLE	TERMINAL	JUPYTER
<pre>Enter details: MSGTYPE(REGISTRN,DISCOVERY), IP address, port number, servicename, service access token or simply type 'b ye' to end REGISTRN,192.168.126.137,1236,pdf,server1 sent to bootp REGISTRN,192.168.126.137,1236,pdf,server1 server1 message sent successfully AT is -server1 All done closing socket now Server server1 Enter the port no 1236 Server socket created successfully. Listening,.... new client connected 127.0.0.1 port 1236 Client sent this access token-server1 Server AT-server1 client is requesting file-server.pdf Requested File is found Total file size: 110592 Bytes read: 4 add to existing sockets list 0 Host disconnected , ip 127.0.0.1 , port 53614</pre>	<pre>All done closing socket now All done closing udp bootpsocket now connecting to server 192.168.126.137 at port 1236 ACCESS TOKEN -server1 server type pdf Server socket created successfully. Connected to Server. want to change the access token no sending access token to server - server1 valid Server can send file of type pdf Enter the filename server.pdf filename server.pdf requesting file server.pdf server.pdf found creating client.pdf Receiving file size is - 110592 File successfully Received! All done closing socket now supriya@supriya-Super-Server:~/Dropbox/ACN/assgn3\$</pre>	

Server connects to multiple clients and sends them files parallelly.

```

supriya@supriya-Super-Server:~/Dropbox
x/ACN/assgn3$ g
++ bootserver.cpp -o bs
supriya@supriya-Super-Server:~/Dropbox
x/ACN/assgn3$ ./bs
Enter the ip address and port number
to listen the connections for
1234
Server listening.....
Message from client : REGISTRN,192.16
8.126.137,1236,pdf,server1
Message from client : DISCOVERY
ended sending server info
Message from client : DISCOVERY
ended sending server info
Message from client : DISCOVERY
ended sending server info
[]

Bootserver

0 1236
1 pdf
2 server1
received server infos
Message from server: end
All done closing socket now
connecting to server 192.168.126.137
at port 1236
ACCESS TOKEN -server1 server type pdf
Server socket created successfully.
Connected to Server.
want to change the access token
yes
Enter new access token
server2
sending access token to server - serv
er2
Invalid
Invalid client
All done closing socket now
supriya@supriya-Super-Server:~/Dropbo
x/ACN/assgn3$ []

Server

message sent successfully
AT is -server1
All done closing socket now
Server server1
Enter the port no
1236
Server socket created successfully.
Listening.....
new client connected 127.0.0.1 port 1
236
Client sent this access token-server1
Server AT-server1
client is requesting file-server.pdf
Requested File is found
Total file size: 110592
Bytes read: 4
add to existing sockets list 0
Host disconnected , ip 127.0.0.1 , po
rt 53614
new client connected 127.0.0.1 port 1
236
Client sent this access token-server2

Client 2

f
Server socket created successfully.
Connected to Server.
want to change the access token
no
sending access token to server - ser
ver1
valid
Server can send file of type pdf
Enter the filename
server.pdf
filename server.pdf
requesting file server.pdf server.
pdf
found
creating client.pdf
Receiving
file size is - 110592
File successfully Received!
All done closing socket now
supriya@supriya-Super-Server:~/Dropb
ox/ACN/assgn3$ []

```

Final execution

- A single BootStrap server, 4 different file servers for video/audio files, .pdf, .text files, image files respectively and two clients

```

supriya@supriya-Super-Server:~/Dropbox/ACN/assgn3$ ./fs
Enter the ip address and port number of bootserver
1234
Enter details: MSGTYPE(REGISTRN,DISCOVERY), IP address, port number, servicename, service access tok
en or simply type 'bye' to end
REGISTRN,192.168.126.137,1236,np4,server1
sent to bootp REGISTRN,192.168.126.137,1236,np4,server1
server1
message sent successfully
AT is -server1
All done closing socket now
Server server1
Enter the port no
1236
Server socket created successfully.
Listening.....
new client connected 127.0.0.1 port 1236
client sent this access token-server1 Server AT-server1
[]

Fileserver1

supriya@supriya-Super-Server:~/Dropbox/ACN/assgn3$ ./fs
Enter the ip address and port number of bootserver
1234
Enter details: MSGTYPE(REGISTRN,DISCOVERY), IP address, port number, servicename, service access tok
en or simply type 'bye' to end
REGISTRN,192.168.126.138,1237,pdf,server2
sent to bootp REGISTRN,192.168.126.138,1237,pdf,server2
server2
message sent successfully
AT is -server2
All done closing socket now
Server server2
Enter the port no
1237
Server socket created successfully.
Listening.....
[]

Fileserver2

supriya@supriya-Super-Server:~/Dropbox/ACN/assgn3$ ./fs
Enter the ip address and port number of bootserver
1234
Enter details: MSGTYPE(REGISTRN,DISCOVERY), IP address, port number, servicename, service access tok
en or simply type 'bye' to end
REGISTRN,192.168.126.139,1238,txt,server3
sent to bootp REGISTRN,192.168.126.139,1238,txt,server3
server3
message sent successfully
AT is -server3
All done closing socket now
Server server3
Enter the port no
1238
Server socket created successfully.
Listening.....
[]

Fileserver3

supriya@supriya-Super-Server:~/Dropbox/ACN/assgn3$ ./fs
Enter the ip address and port number of bootserver
1234
Enter details: MSGTYPE(REGISTRN,DISCOVERY), IP address, port number, servicename, service access tok
en or simply type 'bye' to end
REGISTRN,192.168.126.139,1239,png,server4
sent to bootp REGISTRN,192.168.126.139,1239,png,server4
server4
message sent successfully
AT is -server4
All done closing socket now
Server server4
Enter the port no
1239
Server socket created successfully.
Listening.....
[]

Fileserver4

```

```
Client1
2. server2
received server infos
Message from server: 192.168.126.139,1238,txt,server3,1239,png,server4
ip - 192.168.126.139
0 1238
1 txt
2 server3
3 1239
4 png
5 server4
received server infos
Message from server: end
all done closing socket now
connecting to server 192.168.126.137 at port 1236
ACCESS TOKEN -server1 server type mp4
Server socket created successfully.
Connected to server.
want to change the access token
no
sending access token to server - server1
valid
Server can send file of type mp4
Enter the filename

Client2
ip - 192.168.126.138
0 1237
1 pdf
2 server2
received server infos
Message from server: 192.168.126.139,1238,txt,server3,1239,png,server4
ip - 192.168.126.139
0 1238
1 txt
2 server3
3 1239
4 png
5 server4
received server infos
Message from server: end
all done closing socket now
connecting to server 192.168.126.137 at port 1236
ACCESS TOKEN -server1 server type mp4
Server socket created successfully.
Connected to server.
want to change the access token
no
sending access token to server - server1

Bootpsrver
supriya@supriya-Super-Server:~/Dropbox/ACH/assgn35 ./bs
Enter the ip address and port number to listen the connections for
1234
Server listening.....
Message from client : REGISTRN,192.168.126.137,1236,mp4,server1
Message from client : REGISTRN,192.168.126.138,1237,pdf,server2
Message from client : REGISTRN,192.168.126.139,1238,txt,server3
Message from client : REGISTRN,192.168.126.139,1239,png,server4
Message from client : DISCOVERY
ended sending server info
Message from client : DISCOVERY
ended sending server info
```

- Invalid client check.

```
Client sent this access token-server2
Server AT-server1
Invalid client
```

