COMPUTER NETWORKS

EXP 8

FILE TRANSFER PROTOCOL

SEPTEMBER 14, 2021

ROEHIT RANGANATHAN
RA1911033010017 | L2

Aim:

To implement FTP application, where the Client on establishing a connection with the Server sends the name of the file it wishes to access remotely. The Server then sends the contents of the file to the Client, where it is stored.

Procedure:

Server:

-> Include the necessary header files.

-> Create a socket using socket function with family AF_INET, type as SOCK_STREAM.

-> Initialize server address to 0 using the bzero function.

-> Assign the sin_family to AF_INET, sin_addr to INADDR_ANY, sin_port to dynamically

assigned port number.

-> Bind the local host address to socket using the bind function.

-> Listen on the socket for connection request from the client.

-> Accept connection request from the Client using accept function.

-> Within an infinite loop, receive the file name from the Client.

-> Open the file, read the file contents to a buffer and send the buffer to the Client.

Client:

-> Include the necessary header files.

-> Create a socket using socket function with family AF_INET, type as SOCK_STREAM.

-> Initialize server address to 0 using the bzero function.

-> Assign the sin_family to AF_INET.

-> Get the server IP address and the Port number from the console.

-> Using gethostbyname function assign it to a hostent structure, and assign it to sin_addr of

the server address structure.

-> Within an infinite loop, send the name of the file to be viewed to the Server.

-> Receive the file contents, store it in a file and print it on the console.

Code:

SERVER.C

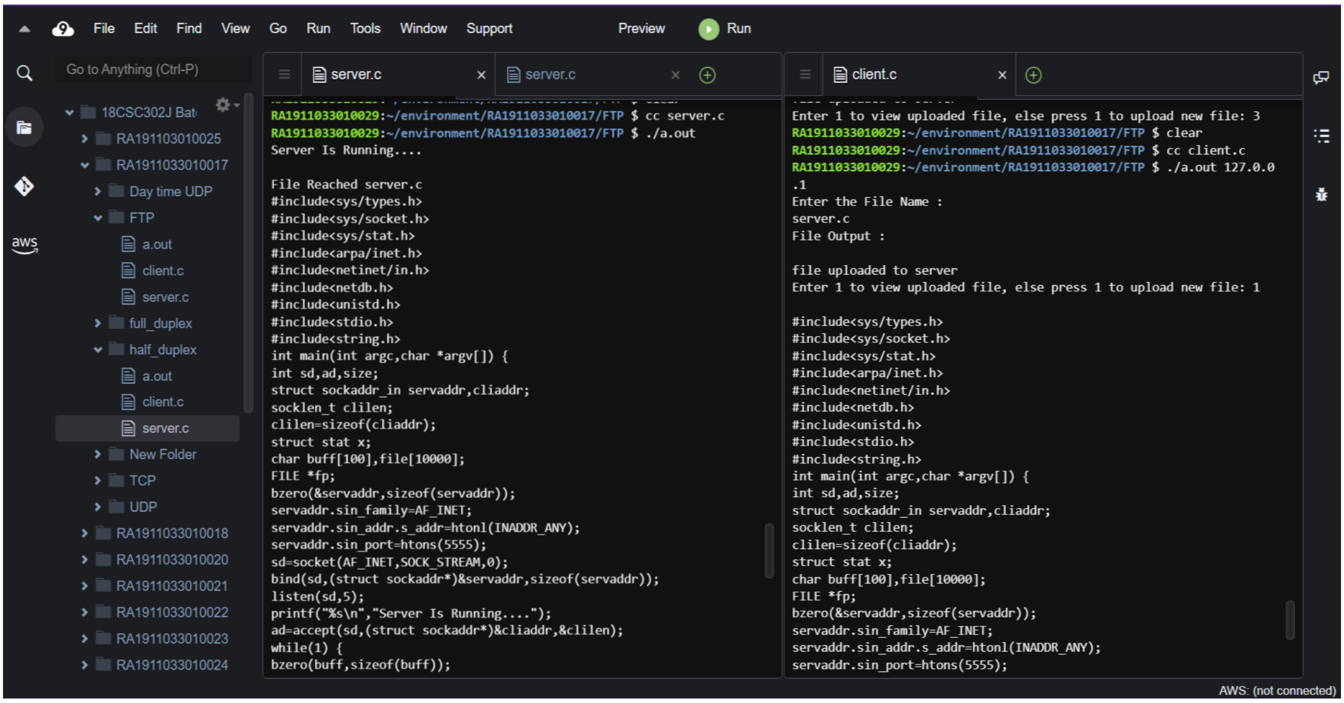
```
#include<sys/types.h>

#include<sys/socket.h>
```

```
#include<sys/stat.h>
#include<arpa/inet.h>
#include<netinet/in.h>
#include<netdb.h>
#include<unistd.h>
#include<stdio.h>
#include<string.h>
int main(int argc,char *argv[]) {
int sd,ad,size;
struct sockaddr_in servaddr,cliaddr;
socklen_t clilen;
clilen=sizeof(cliaddr);
struct stat x;
char buff[100],file[10000];
FILE *fp;
bzero(&servaddr,sizeof(servaddr));
servaddr.sin_family=AF_INET;
servaddr.sin_addr.s_addr=htonl(INADDR_ANY);
servaddr.sin_port=htons(5555);
sd=socket(AF_INET,SOCK_STREAM,0);
bind(sd,(struct sockaddr*)&servaddr,sizeof(servaddr));
listen(sd,5);
printf("%s\n","Server Is Running....");
ad=accept(sd,(struct sockaddr*)&cliaddr,&clilen);
while(1) {
bzero(buff,sizeof(buff));
bzero(file,sizeof(file));
recv(ad,buff,sizeof(buff),0);
printf("\nFile Reached %s",buff);
fp=fopen(buff,"r");
stat(buff,&x);
size=x.st_size;
fread(file,sizeof(file),1,fp);
printf("\n%s",file);
send(ad,file,sizeof(file),0);
}
}
```

CLIENT.C

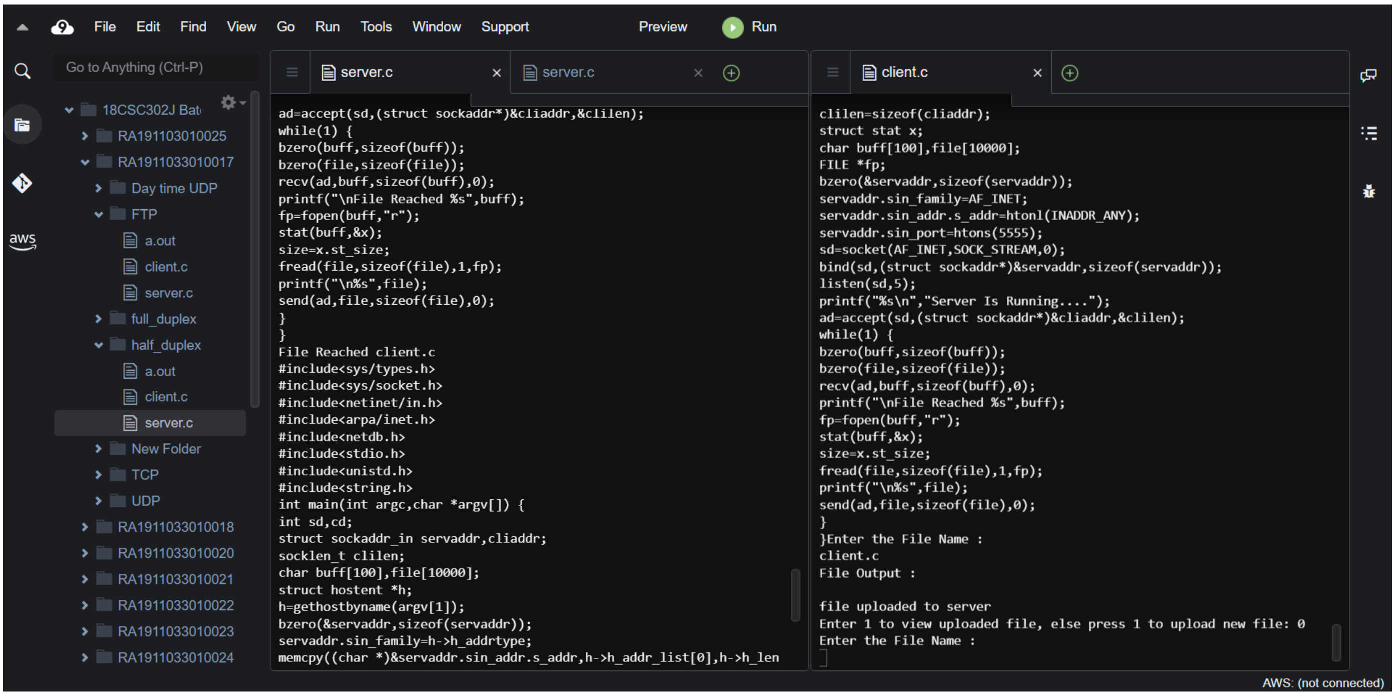
```
#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in.h>
#include<arpa/inet.h>
#include<netdb.h>
#include<stdio.h>
#include<unistd.h>
#include<string.h>
int main(int argc,char *argv[]) {
int sd,cd;
struct sockaddr_in servaddr,cliaddr;
socklen_t clilen;
char buff[100],file[10000];
struct hostent *h;
h=gethostbyname(argv[1]);
bzero(&servaddr,sizeof(servaddr));
servaddr.sin_family=h->h_addrtype;
memcpy((char *)&servaddr.sin_addr.s_addr,h->h_addr_list[0],h->h_length);
servaddr.sin_port=htons(5555);
sd=socket(AF_INET,SOCK_STREAM,0);
cd=connect(sd,(struct sockaddr*)&servaddr,sizeof(servaddr));
while(1)
{
printf("%s\n","Enter the File Name :");
scanf("%s",buff);
send(sd,buff,strlen(buff)+1,0);
printf("%s\n","File Output :");
recv(sd,file,sizeof(file),0);
printf("\nfile uploaded to server");
int val;
printf("\nEnter 1 to view uploaded file, else press 1 to upload new file: ");
scanf("%d",&val);
if(val == 1)
printf("\n%s",file);
else if(val == 0)
continue;
else
break;
}
return 0;
}
```

OUTPUT:



```
RA1911033010029:~/environment/RA1911033010017/FTP $ cc server.c
RA1911033010029:~/environment/RA1911033010017/FTP $ ./a.out
Server Is Running....

File Reached server.c
#include<sys/types.h>
#include<sys/socket.h>
#include<sys/stat.h>
#include<arpa/inet.h>
#include<netinet/in.h>
#include<netdb.h>
#include<unistd.h>
#include<stdio.h>
#include<string.h>
int main(int argc,char *argv[]) {
int sd,ad,size;
struct sockaddr_in servaddr,cliaddr;
socklen_t clilen;
clilen=sizeof(cliaddr);
struct stat x;
char buff[100],file[10000];
FILE *fp;
bzero(&servaddr,sizeof(servaddr));
servaddr.sin_family=AF_INET;
servaddr.sin_addr.s_addr=htonl(INADDR_ANY);
servaddr.sin_port=htons(5555);
sd=socket(AF_INET,SOCK_STREAM,0);
bind(sd,(struct sockaddr*)&servaddr,sizeof(servaddr));
listen(sd,5);
printf("%s\n","Server Is Running....");
ad=accept(sd,(struct sockaddr*)&cliaddr,&clilen);
while(1) {
bzero(buff,sizeof(buff));
```

```
Enter 1 to view uploaded file, else press 1 to upload new file: 3
RA1911033010029:~/environment/RA1911033010017/FTP $ clear
RA1911033010029:~/environment/RA1911033010017/FTP $ cc client.c
RA1911033010029:~/environment/RA1911033010017/FTP $ ./a.out 127.0.0
.1
Enter the File Name :
server.c
File Output :

file uploaded to server
Enter 1 to view uploaded file, else press 1 to upload new file: 1

#include<sys/types.h>
#include<sys/socket.h>
#include<sys/stat.h>
#include<arpa/inet.h>
#include<netinet/in.h>
#include<netdb.h>
#include<unistd.h>
#include<stdio.h>
#include<string.h>
int main(int argc,char *argv[]) {
int sd,ad,size;
struct sockaddr_in servaddr,cliaddr;
socklen_t clilen;
clilen=sizeof(cliaddr);
struct stat x;
char buff[100],file[10000];
FILE *fp;
bzero(&servaddr,sizeof(servaddr));
servaddr.sin_family=AF_INET;
servaddr.sin_addr.s_addr=htonl(INADDR_ANY);
servaddr.sin_port=htons(5555);
```



```
ad=accept(sd,(struct sockaddr*)&cliaddr,&clilen);
while(1) {
bzero(buff,sizeof(buff));
bzero(file,sizeof(file));
recv(ad,buff,sizeof(buff),0);
printf("\nFile Reached %s",buff);
fp=fopen(buff,"r");
stat(buff,&x);
size=x.st_size;
fread(file,sizeof(file),1,fp);
printf("\n%s",file);
send(ad,file,sizeof(file),0);
}
}
File Reached client.c
#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in.h>
#include<arpa/inet.h>
#include<netdb.h>
#include<stdio.h>
#include<unistd.h>
#include<string.h>
int main(int argc,char *argv[]) {
int sd,cd;
struct sockaddr_in servaddr,cliaddr;
socklen_t clilen;
char buff[100],file[10000];
struct hostent *h;
h=gethostbyname(argv[1]);
bzero(&servaddr,sizeof(servaddr));
servaddr.sin_family=h->h_addrtype;
memcpy((char *)&servaddr.sin_addr.s_addr,h->h_addr_list[0],h->h_len
```

```
clilen=sizeof(cliaddr);
struct stat x;
char buff[100],file[10000];
FILE *fp;
bzero(&servaddr,sizeof(servaddr));
servaddr.sin_family=AF_INET;
servaddr.sin_addr.s_addr=htonl(INADDR_ANY);
servaddr.sin_port=htons(5555);
sd=socket(AF_INET,SOCK_STREAM,0);
bind(sd,(struct sockaddr*)&servaddr,sizeof(servaddr));
listen(sd,5);
printf("%s\n","Server Is Running....");
ad=accept(sd,(struct sockaddr*)&cliaddr,&clilen);
while(1) {
bzero(buff,sizeof(buff));
bzero(file,sizeof(file));
recv(ad,buff,sizeof(buff),0);
printf("\nFile Reached %s",buff);
fp=fopen(buff,"r");
stat(buff,&x);
size=x.st_size;
fread(file,sizeof(file),1,fp);
printf("\n%s",file);
send(ad,file,sizeof(file),0);
}
}Enter the File Name :
client.c
File Output :

file uploaded to server
Enter 1 to view uploaded file, else press 1 to upload new file: 0
Enter the File Name :
```