

COMPUTER NETWORKS PRACTICE ASSIGNMENT 4

Name: Vinayak Sethi

Roll No: COE18B061

File Transfer using TCP

The general idea of sending the file is opening the file in 'rb' mode in client side and storing the content of it in sendbuffer using fread and then send the character bit by bit using **send() function** and then in server side opening the file in 'wb' mode and receiving the file using **recv() function** and write the contents in file using fwrite function.

Filename: TCP_Client_ft.c

```
~/Documents/Computer Networking/File_Transfer/
File Edit Selection Find View Goto Tools Project Preferences Help
TCP_Client_ft.c x TCP_Server_ft.c x
1  #include<stdio.h>
2  #include<stdlib.h>
3  #include<sys/socket.h>
4  #include<sys/types.h>
5  #include<netinet/in.h>
6  #include<unistd.h>
7  #include<string.h>
8  #include<libgen.h>
9  #include<time.h>
10
11 #define bufsize 1024
12
13 int main()
14 {
15     int client_socket, sin_size, check;
16     char sendbuffer[bufsize];
17     struct sockaddr_in server_address;
18
19     //create a socket
20     client_socket = socket(AF_INET, SOCK_STREAM, 0);
21     if(client_socket == -1)
22     {
23         printf("\nSocket Creation Failure\n");
24         exit(EXIT_FAILURE);
25     }
26
27     //specify an address for the socket
28     server_address.sin_family = AF_INET;
29     server_address.sin_port = htons(9009);
30     server_address.sin_addr.s_addr = INADDR_ANY;
31
32     sin_size = sizeof(struct sockaddr_in);
33
34     //connect to server
35     if(connect(client_socket, (struct sockaddr *)&server_address, sin_size) == 0)
36     {
37         printf("Connect Successful\n");
38
39         char path[bufsize];
40         printf("Enter the complete path of the filename you wish to send : ");
41         scanf("%s", path);
42         char *filename = basename(path); //return the last component of a pathname
43         printf("%s\n", filename);
44
45         send(client_socket, filename, strlen(filename), 0);
46
47         FILE *fp = fopen(path, "rb");
48         if (fp == NULL)
```

```
TCP_Client_ft.c  x  TCP_Server_ft.c  x
48 {
49     printf("Cannot open the file\n");
50     exit(EXIT_FAILURE);
51 }
52
53 clock_t t;
54 t = clock();
55
56 while((check = fread(sendbuffer, 1, sizeof(sendbuffer), fp)) > 0)
57 {
58     send(client_socket, sendbuffer, check, 0);
59 }
60
61 fclose(fp);
62 t = clock() - t;
63 double time_taken = ((double)t)/CLOCKS_PER_SEC; //in seconds
64
65 printf("File sent successfully ...\n");
66 printf("File transfer took %f seconds to execute.\n", time_taken);
67
68 close(client_socket);
69 return 0;
70 }
71
72
```

Filename: TCP_Server_ft.c

```
~/Documents/Computer Networking/File_Transfer/TCP/
File Edit Selection Find View Goto Tools Project Preferences Help
TCP_Client_ft.c  x  TCP_Server_ft.c  x
1  #include<stdio.h>
2  #include<stdlib.h>
3  #include<sys/socket.h>
4  #include<sys/types.h>
5  #include<netinet/in.h>
6  #include<unistd.h>
7  #include<string.h>
8  #include<time.h>
9
10 #define buffsize 1024
11
12 int main()
13 {
14     int server_socket, client_socket, sin_size, check;
15     char buffer[buffsize];
16     struct sockaddr_in server_address, client_address;
17
18     //create a socket
19     server_socket = socket(AF_INET, SOCK_STREAM, 0);
20     if(server_socket == -1)
21     {
22         printf("\nSocket Creation Failure\n");
23         exit(EXIT_FAILURE);
24     }
25
26     //specify an address for the socket
27     server_address.sin_family = AF_INET;
28     server_address.sin_port = htons(9009);
29     server_address.sin_addr.s_addr = INADDR_ANY;
30
31     //bind with the client
32     if( bind(server_socket, (const struct sockaddr *)&server_address, sizeof(server_address)) < 0)
33     {
34         printf("Could not bind to Client\n");
35         exit(EXIT_FAILURE);
36     }
37
38     //listen to the incoming client request
39     if(listen(server_socket, 10) == 0)
40         printf("Listen successful\n");
```

```

41
42 //accept a connection request from client
43 sin_size = sizeof(struct sockaddr_in);
44 if((client_socket = accept(server_socket, (struct sockaddr *)&client_address,&sin_size)) > 0)
45     printf("Accept Successful\n");
46
47 char filename[buffsize];
48 memset (filename, '\0', sizeof(filename));
49 recv(client_socket, filename, sizeof(filename), 0);
50 printf("File received is : ");
51 printf("%s\n", filename);
52
53
54 clock_t t;
55 t = clock();
56
57 FILE *fp = fopen(filename, "wb");
58 if(fp!=NULL)
59 {
60     while((check = recv(client_socket,buffer, sizeof(buffer), 0)) > 0)
61     {
62         fwrite(buffer, 1, check, fp);
63     }
64     fclose(fp);
65 }
66
67 t = clock() - t;
68 double time_taken = ((double)t)/CLOCKS_PER_SEC; //in seconds
69
70 printf("File received successfully...\n");
71 printf("File received in %f seconds.\n", time_taken);
72
73 //close the socket
74 close(client_socket);
75 close(server_socket);
76
77 return 0;
78 }
79

```

Output:

```

vinayak@vinayak-Swift-SF315-52G: ~/Documents/Computer Networking/File_Transfer/TCP/Client
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/TCP/Client$ make TCP_Client_ft
make: 'TCP_Client_ft' is up to date.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/TCP/Client$ ./TCP_Client_ft
Connect Successful
Enter the complete path of the filename you wish to send : /home/vinayak/Documents/india-flag.gif
india-flag.gif
File sent successfully ...
File transfer took 0.005138 seconds to execute.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/TCP/Client$ ./TCP_Client_ft
Connect Successful
Enter the complete path of the filename you wish to send : TAG.mp3
TAG.mp3
File sent successfully ...
File transfer took 0.010705 seconds to execute.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/TCP/Client$ ./TCP_Client_ft
Connect Successful
Enter the complete path of the filename you wish to send : rauf_faik.mp4
rauf_faik.mp4
File sent successfully ...
File transfer took 0.061589 seconds to execute.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/TCP/Client$

vinayak@vinayak-Swift-SF315-52G: ~/Documents/Computer Networking/File_Transfer/TCP/Server
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/TCP/Server$ make TCP_Server_ft
make: 'TCP_Server_ft' is up to date.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/TCP/Server$ ./TCP_Server_ft
Listen successful
Accept Successful
File received is : india-flag.gif
File received successfully...
File received in 0.008587 seconds.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/TCP/Server$ ./TCP_Server_ft
Listen successful
Accept Successful
File received is : TAG.mp3
File received successfully...
File received in 0.013118 seconds.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/TCP/Server$ ls
india-flag.gif TAG.mp3 TCP_Server_ft TCP_Server_ft.c
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/TCP/Server$ ./TCP_Server_ft
Listen successful
Accept Successful
File received is : rauf faik.mp4
File received successfully...
File received in 0.063613 seconds.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/TCP/Server$ ls
india-flag.gif rauf_faik.mp4 TAG.mp3 TCP_Server_ft TCP_Server_ft.c
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/TCP/Server$

```

- Using **ls** we can verify server has received the file.

Latency Test:

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Computer Networking/File_Transf...
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/TC
P/Server$ sudo traceroute localhost
traceroute to localhost (127.0.0.1), 30 hops max, 60 byte packets
 1  localhost (127.0.0.1)  0.042 ms  0.012 ms  0.009 ms
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/TC
P/Server$ ping -c 6 localhost
PING localhost (127.0.0.1) 56(84) bytes of data.
64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 time=0.049 ms
64 bytes from localhost (127.0.0.1): icmp_seq=2 ttl=64 time=0.053 ms
64 bytes from localhost (127.0.0.1): icmp_seq=3 ttl=64 time=0.060 ms
64 bytes from localhost (127.0.0.1): icmp_seq=4 ttl=64 time=0.070 ms
64 bytes from localhost (127.0.0.1): icmp_seq=5 ttl=64 time=0.070 ms
64 bytes from localhost (127.0.0.1): icmp_seq=6 ttl=64 time=0.067 ms

--- localhost ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5105ms
rtt min/avg/max/mdev = 0.049/0.061/0.070/0.008 ms
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/TC
P/Server$
```

File Transfer using UDP

Same concept is used like TCP file transfer logic but in UDP there is chances of data loss, so to reduce data loss we have used sleep command because client keep sending data to server and server take time to write to file so to slow down this process sleep command is used. Also some functions are changed as per UDP protocol, like **sendto()** in client side and **recvfrom()** in server side is used.

Also **fgetc** command is used to read the content byte by byte and **fputc** command is used to write byte by byte to file in server side.

Filename: UDP_Client_ft.c


```
~/Documents/Computer Networking/File_Transfer/UDP/Client/UDP_Client_ft.c - Sublime Text (
File Edit Selection Find View Goto Tools Project Preferences Help

UDP_Client_ft.c x UDP_Server_ft.c x
1 #include<stdio.h>
2 #include<stdlib.h>
3 #include<sys/socket.h>
4 #include<sys/types.h>
5 #include<netinet/in.h>
6 #include<unistd.h>
7 #include<string.h>
8 #include<libgen.h>
9 #include<time.h>
10
11 #define buffsize 1024
12
13 int main()
14 {
15     int client_socket,check;
16     struct sockaddr_in server_address;
17
18     //create a socket
19     client_socket = socket(AF_INET,SOCK_DGRAM,0);
20     if(client_socket == -1)
21     {
22         printf("\nSocket Creation Failure\n");
23         exit(EXIT_FAILURE);
24     }
25
26     //specify an address for the socket
27     server_address.sin_family = AF_INET;
28     server_address.sin_port = htons(9009);
29     server_address.sin_addr.s_addr = INADDR_ANY;
30 }
```

```
UDP_Client_ft.c x UDP_Server_ft.c x
31     socklen_t length = sizeof(server_address);
32
33     sendto(client_socket,"Hello server", strlen("Hello server"),0,(struct sockaddr *)&server_address, sizeof(server_address));
34
35     char path[buffsize];
36     printf("Enter the complete path of the filename you wish to send : ");
37     scanf("%s",path);
38     char *filename = basename(path); //return the last component of a pathname
39     printf("%s\n", filename);
40
41     sendto(client_socket, filename, strlen(filename), 0,(struct sockaddr *)&server_address, sizeof(server_address));
42
43     FILE *fp = fopen(path,"rb");
44     if (fp == NULL)
45     {
46         printf("Cannot open the file\n");
47         exit(EXIT_FAILURE);
48     }
49
50     clock_t t;
51     t = clock();
52
53     while((check = fgetc(fp)) != EOF)
54     {
55         sendto(client_socket, &check, sizeof(check), 0,(struct sockaddr *)&server_address, sizeof(server_address));
56         sleep(0);
57     }
58
59     //send the EOF to signal file end
60     sendto(client_socket, &check, sizeof(check), 0,(struct sockaddr *)&server_address, sizeof(server_address));
61
62     fclose(fp);
63     t = clock() - t;
64     double time_taken = ((double)t)/CLOCKS_PER_SEC; //in seconds
65
66     printf("File sent successfully ...\n");
67     printf("File transfer took %f seconds to execute.\n", time_taken);
68
69     close(client_socket);
70     return 0;
71 }
```

Filename: UDP_Server_ft.c

```
~/Documents/Computer Networking/File_Transfer/UDP/Server/UDP_Server_ft.c - Sublime
File Edit Selection Find View Goto Tools Project Preferences Help

UDP_Client_ft.c x UDP_Server_ft.c x
1 #include<stdio.h>
2 #include<stdlib.h>
3 #include<sys/socket.h>
4 #include<sys/types.h>
5 #include<netinet/in.h>
6 #include<unistd.h>
7 #include<string.h>
8 #include<time.h>
9
10 #define buffsize 1024
11
12 int main()
13 {
14     int server_socket,check,store;
15     char buffer[buffsize];
16     struct sockaddr_in server_address, client_address;
17
18     //create a socket
19     server_socket = socket(AF_INET,SOCK_DGRAM,0);
20     if(server_socket == -1)
21     {
22         printf("\nSocket Creation Failure\n");
23         exit(EXIT_FAILURE);
24     }
25
26     //specify an address for the socket
27     server_address.sin_family = AF_INET;
28     server_address.sin_port = htons(9009);
29     server_address.sin_addr.s_addr = INADDR_ANY;
30 }
```

```
UDP_Client_ft.c x UDP_Server_ft.c x
31 //bind with the client
32 if( bind(server_socket, (const struct sockaddr *)&server_address, sizeof(server_address)) < 0)
33 {
34     printf("Could not bind to Client\n");
35     exit(EXIT_FAILURE);
36 }
37
38 socklen_t length = sizeof(server_address);
39 recvfrom(server_socket, buffer, sizeof(buffer),0,(struct sockaddr *)&client_address,&length);
40
41 char filename[buffsize];
42 memset (filename, '\0', sizeof(filename));
43 recvfrom(server_socket, filename, sizeof(filename), 0,(struct sockaddr *)&client_address,&length);
44 printf("File received is : ");
45 printf("%s\n", filename);
46
47 clock_t t;
48 t = clock();
49
50 FILE *fp = fopen(filename, "wb");
51 if(fp!=NULL)
52 {
53     while(store != EOF)
54     {
55         recvfrom(server_socket, &store, sizeof(store), 0,(struct sockaddr *)&client_address,&length);
56         fputc(store, fp);
57     }
58 }
59
60 fclose(fp);
61 t = clock() - t;
62 double time_taken = ((double)t)/CLOCKS_PER_SEC; //in seconds
63
64 printf("File received successfully...\n");
65 printf("File received in %f seconds.\n", time_taken);
66
67 //close the socket
68 close(server_socket);
69 return 0;
70 }
```

Output:

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Computer Networking/File_Transfer/UDP/Client
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/UDP/Client$ make UDP_Client ft
make: 'UDP_Client ft' is up to date.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/UDP/Client$ ./UDP_Client ft
Enter the complete path of the filename you wish to send : /home/vinayak/Documents/india-flag.gif
india-flag.gif
File sent successfully ...
File transfer took 13.511252 seconds to execute.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/UDP/Client$ ./UDP_Client ft
Enter the complete path of the filename you wish to send : TAG.mp3
TAG.mp3
File sent successfully ...
File transfer took 72.453478 seconds to execute.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/UDP/Client$ ./UDP_Client ft
Enter the complete path of the filename you wish to send : rauf_faik.mp4
rauf_faik.mp4
File sent successfully ...
File transfer took 142.087561 seconds to execute.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/UDP/Client$

vinayak@vinayak-Swift-SF315-52G: ~/Documents/Computer Networking/File_Transfer/UDP/Server
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/UDP/Server$ make UDP_Server ft
make: 'UDP_Server ft' is up to date.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/UDP/Server$ ./UDP_Server ft
File received is : india-flag.gif
File received successfully...
File received in 8.361433 seconds.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/UDP/Server$ ls
india-flag.gif  UDP_Server_ft  UDP_Server_ft.c
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/UDP/Server$ ./UDP_Server ft
File received is : TAG.mp3
File received successfully...
File received in 44.841575 seconds.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/UDP/Server$ ls
india-flag.gif  TAG.mp3  UDP_Server_ft  UDP_Server_ft.c
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/UDP/Server$ ./UDP_Server ft
File received is : rauf_faik.mp4
File received successfully...
File received in 87.785046 seconds.
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/UDP/Server$ ls
india-flag.gif  rauf_faik.mp4  TAG.mp3  UDP_Server_ft  UDP_Server_ft.c
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/UDP/Server$
```

- Using `ls` we can verify server has received the file.

Latency Test:

```
vinayak@vinayak-Swift-SF315-52G: ~/Documents/Computer Networking/File_Transf...
File Edit View Search Terminal Help
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/UDP/Server$ sudo traceroute localhost
traceroute to localhost (127.0.0.1), 30 hops max, 60 byte packets
 1 localhost (127.0.0.1)  0.034 ms  0.011 ms  0.010 ms
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/UDP/Server$ ping -c 6 localhost
PING localhost (127.0.0.1) 56(84) bytes of data.
64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 time=0.035 ms
64 bytes from localhost (127.0.0.1): icmp_seq=2 ttl=64 time=0.044 ms
64 bytes from localhost (127.0.0.1): icmp_seq=3 ttl=64 time=0.045 ms
64 bytes from localhost (127.0.0.1): icmp_seq=4 ttl=64 time=0.083 ms
64 bytes from localhost (127.0.0.1): icmp_seq=5 ttl=64 time=0.043 ms
64 bytes from localhost (127.0.0.1): icmp_seq=6 ttl=64 time=0.043 ms

--- localhost ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5097ms
rtt min/avg/max/mdev = 0.035/0.048/0.083/0.015 ms
vinayak@vinayak-Swift-SF315-52G:~/Documents/Computer Networking/File_Transfer/UDP/Server$
```