

Name : Sahil Dattatray Mohite

Div : SY-IT-A Batch : B1

Roll No. 30 PRN : 12010501

Computer Network – Lab Assignment 5

Unit : Stop&Wait(for errorfree channel and for Noisy Channel)

Question:

Implemet Stop & wait for Noiseless and Noisy channel using C sockets

Noiseless:

Server.c

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<stdbool.h>
#include<math.h>

#include <unistd.h>

#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in.h>

int main()
{

int socket_server , clientsocketfd, bindstatus;

socket_server = socket(AF_INET , SOCK_STREAM , 0);

struct sockaddr_in serveraddress , clientaddress;
serveraddress.sin_family = AF_INET;
serveraddress.sin_port = htons(9000);
serveraddress.sin_addr.s_addr = INADDR_ANY;

bindstatus = bind( socket_server ,
                  (struct sockaddr *)&serveraddress ,
                  sizeof(serveraddress)
                  );

if (bindstatus<0)
{
```

```

    printf("Binding Failed\n");
}
else
{
    printf("Binding is successful\n");
}

listen(socket_server , 10);
printf("Send reply to the client\n");

int cliaddlen = sizeof(clientaddress);
clientsocketfd = accept(socket_server ,
                        (struct sockaddr *)&clientaddress,
                        &cliaddlen );

int n;

read(clientsocketfd , &n, sizeof(n));

char server_msg[100] = "Received Sucessfully:";

int j = 0;
int k;
int time=0;
while(j<n)
{

    if(j==3)
    {
        sleep(2);
        break;
    }

    read(clientsocketfd , &k , sizeof(k));
    printf("\nPacket %d %s",k ,server_msg);
    write(clientsocketfd ,&k, sizeof(k));
    k++;
    printf("\n");
    j++;
}
sleep(2);

close(socket_server);
return 0;
}

```

Client.c

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<stdbool.h>
#include<math.h>

#include <unistd.h>

#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in.h>

int main()
{

int socket_client , serversocketfd;
struct sockaddr_in serveraddress;
struct hostent *server;

socket_client = socket(AF_INET , SOCK_STREAM , 0);

if(socket_client<0)
printf("Socket is NOT created:(\n");
else
printf("socket is created succesfully:\n");

serveraddress.sin_family = AF_INET;
serveraddress.sin_port = htons(9000);
serveraddress.sin_addr.s_addr= INADDR_ANY;

int connectionstatus = connect(socket_client,
                                (struct sockaddr *) &serveraddress,
                                sizeof(serveraddress));

if(connectionstatus == -1)
{
    printf("There was an error in the connection with server:( Try
again!\n");
}

    int packets[100] ;
    int n;
    printf("\n\n\tHow many packets you want to send? ");
    scanf("%d",&n);
```

```

    for (int i=0; i<=n;i++)
    {
        packets[i] = i;
    }
    printf("\n\tPackets to be send:\t");
    for (int i=0; i<n;i++)
    {
        printf("%d\t",packets[i]);
    }
    printf("\n");

printf("\n");
write(socket_client, &n , sizeof(n));
printf("\n");

    char send_msg[] = "Sent ...";
    char ack_msg[100] = "Received. Send Next";

    int i = 0;
    int k;
    int time=0;
    while(i<n)
    {
        printf("\nPacket %d %s",i , send_msg);
        sleep(2);

        if(i==3)
        {
            break;
        }
        write(socket_client, &i , sizeof(i));
        read(socket_client, &k , sizeof(i));
        printf("\nPacket %d %s",k ,ack_msg);
        time=time+2;
        i++;
        k++;

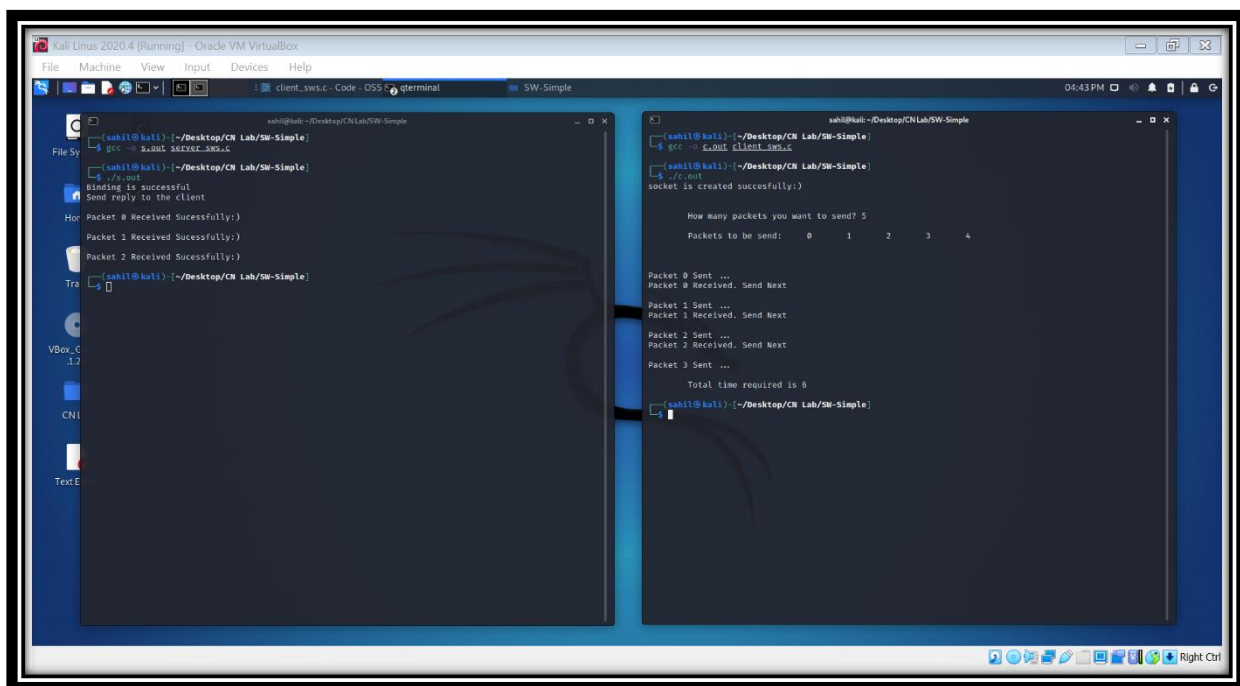
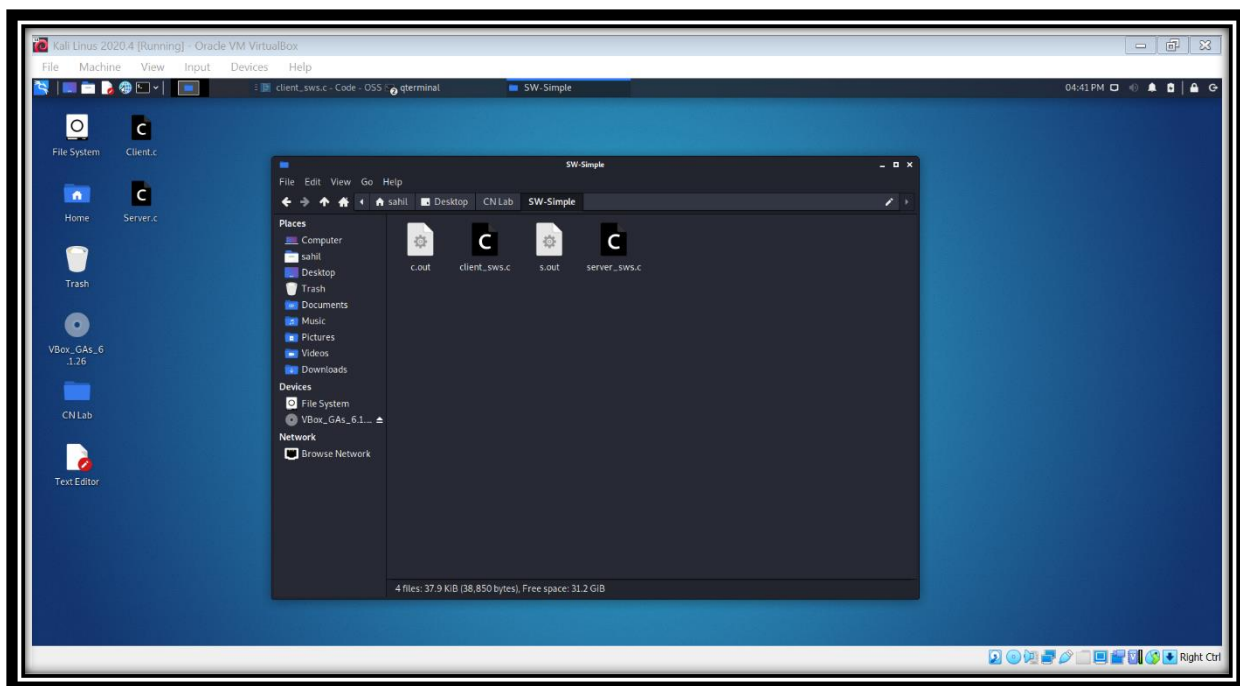
        printf("\n");
    }

printf("\n\n\tTotal time required is %d",time);

close(socket_client);
return 0;

}

```



Noisy:

Server.c

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<stdbool.h>
#include<math.h>

#include <unistd.h>

#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in.h>

int main()
{

int socket_server , clientsocketfd, bindstatus;

socket_server = socket(AF_INET , SOCK_STREAM , 0);

struct sockaddr_in serveraddress , clientaddress;
serveraddress.sin_family = AF_INET;
serveraddress.sin_port = htons(9000);
serveraddress.sin_addr.s_addr = INADDR_ANY;

bindstatus = bind( socket_server ,
                   (struct sockaddr *)&serveraddress ,
                   sizeof(serveraddress)
                   );

if (bindstatus<0)
{
    printf("Binding Failed\n");
}
else
{
    printf("Binding is successful\n");
}

listen(socket_server , 10);
printf("Send reply to the client\n");

int cliaddlen = sizeof(clientaddress);
clientsocketfd = accept(socket_server ,
                       (struct sockaddr *)&clientaddress,
                       &cliaddlen );
```

```
int n;

read(clientsocketfd , &n, sizeof(n));

char server_msg[100] = "Received Sucessfully:);

int j = 0;
int k;
int time=0;
while(j<n)
{

    if(j==3)
    {
        sleep(2);
        break;
    }

    read(clientsocketfd , &k , sizeof(k));
    printf("\nPacket %d %s",k ,server_msg);
    write(clientsocketfd ,&k, sizeof(k));
    k++;
    printf("\n");
    j++;
}
while(j<n)
{
    read(clientsocketfd , &k , sizeof(k));
    printf("\nPacket %d %s",k ,server_msg);
    write(clientsocketfd ,&k, sizeof(k));
    k++;
    printf("\n");
    j++;
}

close(socket_server);
return 0;
}
```

Client.c

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<stdbool.h>
#include<math.h>

#include <unistd.h>

#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in.h>

int main()
{

int socket_client , serversocketfd;
struct sockaddr_in serveraddress;
struct hostent *server;

socket_client = socket(AF_INET , SOCK_STREAM , 0);

if(socket_client<0)
printf("Socket is NOT created:(\n");
else
printf("socket is created succesfully:\n");

serveraddress.sin_family = AF_INET;
serveraddress.sin_port = htons(9000);
serveraddress.sin_addr.s_addr= INADDR_ANY;

int connectionstatus = connect(socket_client,
                                (struct sockaddr *) &serveraddress,
                                sizeof(serveraddress));

if(connectionstatus == -1)
{
    printf("There was an error in the connection with server:( Try
again!\n");
}

    int packets[100] ;
    int n;
    printf("\n\n\tHow many packets you want to send? ");
    scanf("%d",&n);
```



```

    for (int i=0; i<=n;i++)
    {
        packets[i] = i;
    }
    printf("\n\tPackets to be send:\t");
    for (int i=0; i<n;i++)
    {
        printf("%d\t",packets[i]);
    }
    printf("\n");

printf("\n");
write(socket_client, &n , sizeof(n));
printf("\n");

    char send_msg[] = "Sent ...";
    char ack_msg[100] = "Received. Send Next";
    int i = 0;
    int k;
    int time=0;

    while(i<n)
    {

        printf("\nPacket %d %s",i , send_msg);
        sleep(2);

        if(i==3)
        {
            break;
        }

        write(socket_client, &i , sizeof(i));
        read(socket_client, &k , sizeof(i));
        printf("\nPacket %d %s",k ,ack_msg);
        time=time+2;
        i++;
        k++;
        printf("\n");
    }

    printf("\n");
    sleep(2);

    while(i<n)
    {
        printf("\nPacket %d %s",i , send_msg);

```

```
sleep(2);
time=time+2;

write(socket_client, &i , sizeof(i));
read(socket_client, &k , sizeof(i));
printf("\nPacket %d %s",k ,ack_msg);

i++;
k++;
printf("\n");
}

printf("\n\n\tTotal time required is %d",time);
close(socket_client);
return 0;
}
```

