***MAYANK SINHA RA1911003010386 G1***

## *Week 9: REMOTE COMMAND EXECUTION USING UDP*

## *Aim: To study remote command execution uing UDP.*

## *Code:*

Server program

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<unistd.h>
#include<netinet/in.h>
#include<arpa/inet.h>
#include<sys/types.h>
#include<sys/socket.h>
#include<errno.h>
int main()
{
int sd,acpt,len,bytes,port;
char send[50],receiv[50];
struct sockaddr_in serv,cli;
if((sd=socket(AF_INET,SOCK_STREAM,0))<0)
{
printf("Error in socket\n");
exit(0);
}
bzero(&serv,sizeof(serv));
printf("Enter the port number  :  ");
scanf("%d",&port);
serv.sin_family=AF_INET;
serv.sin_port=htons(port);
serv.sin_addr.s_addr=htonl(INADDR_ANY);
if(bind(sd,(struct sockaddr *)&serv,sizeof(serv))<0)
{
printf("Error in bind\n");
exit(0);
}
if(listen(sd,3)<0)
{
printf("Error in listen\n");
exit(0);
}
if((acpt=accept(sd,(struct sockaddr*)NULL,NULL))<0)
{
```

```
printf("\n\t Error in accept");
exit(0);
}
while(1)
{

bytes=recv(acpt,receiv,50,0);
receiv[bytes]='\0';
if(strcmp(receiv ,"end")==0)
{
close(acpt);
close(sd);
exit(0);
}
else
{
printf("Command received : %s",receiv);
system(receiv);
printf("\n");
}
}
}
```

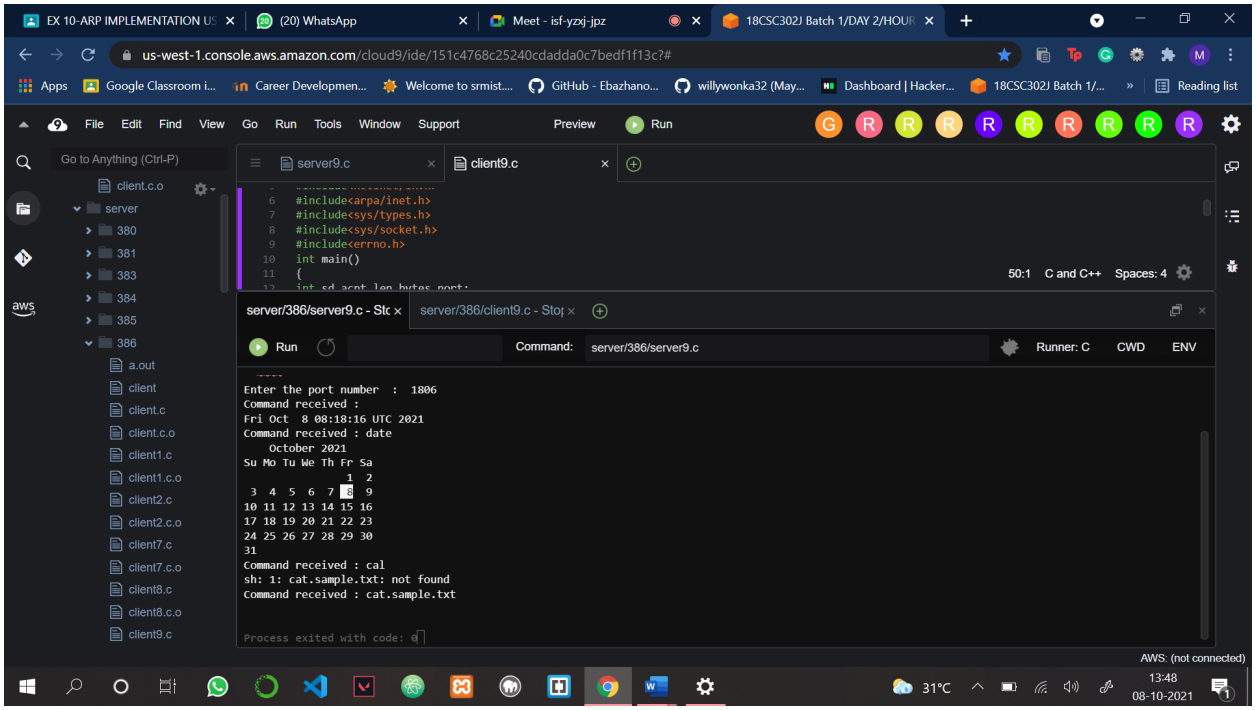
Client program

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<unistd.h>
#include<netinet/in.h>
#include<arpa/inet.h>
#include<sys/types.h>
#include<sys/socket.h>
#include<errno.h>
int main()
{
int sd,acpt,len,bytes,port;
char send1[50],receiv[50];
struct sockaddr_in serv,cli;
if((sd=socket(AF_INET,SOCK_STREAM,0))<0)
{
printf("Error in socket\n");
```

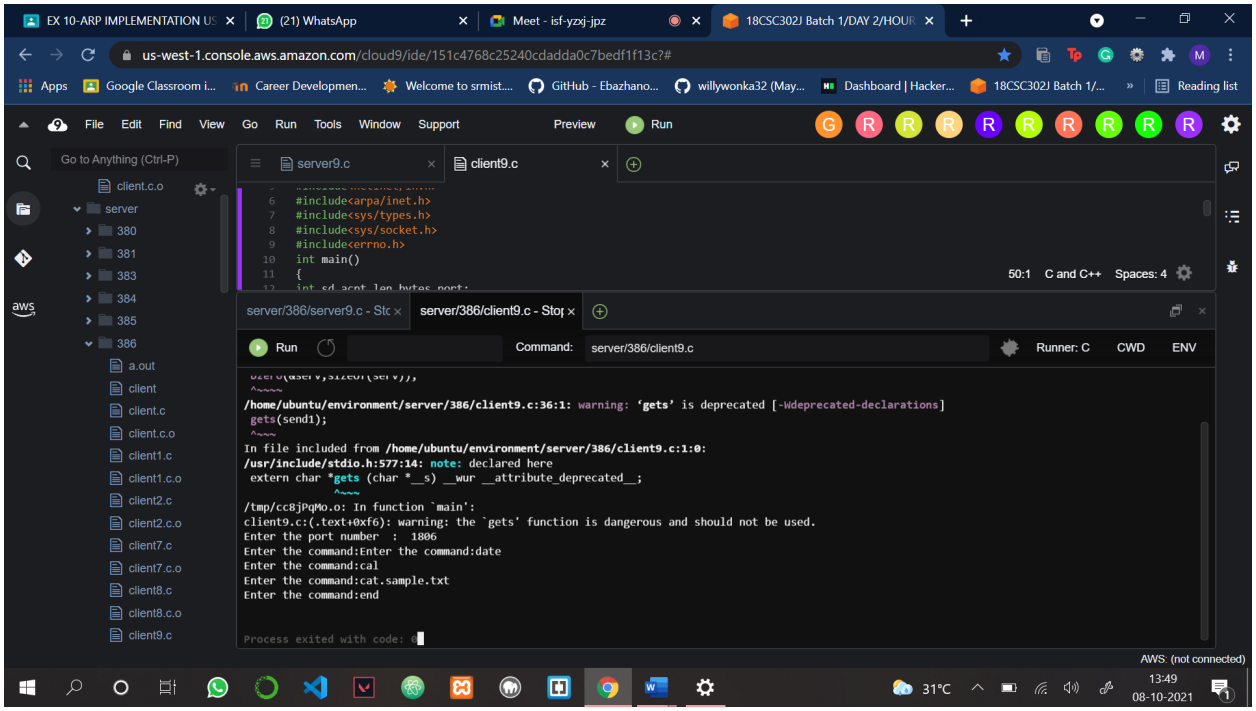
```
exit(0);
}
bzero(&serv,sizeof(serv));
printf("Enter the port number  :  ");
scanf("%d",&port);
serv.sin_family=AF_INET;
serv.sin_port=htons(port);
serv.sin_addr.s_addr=htonl(INADDR_ANY);
if(connect(sd,(struct sockaddr *)&serv,sizeof(serv))<0)
{
printf("Error in connection\n");
exit(0);
}

while(1)
{

printf("Enter the command:");
gets(send1);
if(strcmp(send1,"end")!=0)
{
send(sd,send1,50,0);
}
else
{
send(sd,send1,50,0);
close(sd);
break;
}

}
}
```

## ***OUTPUT:***

## ***server:***


server9.c
client9.c
#include<arpa/inet.h>
#include<sys/types.h>
#include<sys/socket.h>
#include<errno.h>
int main()
{
server/386/server9.c - Stc
server/386/client9.c - Stop
Run
Command: server/386/server9.c
Runner: C
CWD
ENV
Enter the port number : 1806
Command received :
Fri Oct 8 08:18:16 UTC 2021
Command received : date
October 2021
Su Mo Tu We Th Fr Sa
1 2
3 4 5 6 7 8 9
10 11 12 13 14 15 16
17 18 19 20 21 22 23
24 25 26 27 28 29 30
31
Command received : cal
sh: 1: cat.sample.txt: not found
Command received : cat.sample.txt
Process exited with code: 0
AWS: (not connected)
13:48
08-10-2021

## ***client:***


server9.c
client9.c
#include<arpa/inet.h>
#include<sys/types.h>
#include<sys/socket.h>
#include<errno.h>
int main()
{
server/386/server9.c - Stc
server/386/client9.c - Stop
Run
Command: server/386/client9.c
Runner: C
CWD
ENV
/home/ubuntu/environment/server/386/client9.c:36:1: warning: 'gets' is deprecated [-Wdeprecated-declarations]
gets(send1);
In file included from /home/ubuntu/environment/server/386/client9.c:1:0:
/usr/include/stdio.h:577:14: note: declared here
extern char *gets (char *__s) __wur __attribute_deprecated__;
/tmp/cc8jPqMo.o: In function `main':
client9.c:(.text+0xf6): warning: the `gets' function is dangerous and should not be used.
Enter the port number : 1806
Enter the command:Enter the command:date
Enter the command:cal
Enter the command:cat.sample.txt
Enter the command:end
Process exited with code: 0
AWS: (not connected)
13:49
08-10-2021