# CN\_LAB

**Gourav Samantaray 1906555**

# WAP UDP socket to:

* 1. **Print the directory contents given by the client**

# Print the file contents given by the client (Both parts in single program)

/\*

\*\* A datagram sockets "server" demo

\*/

#include <stdio.h> #include <stdlib.h> #include <unistd.h> #include <errno.h> #include <string.h> #include <sys/types.h> #include <sys/socket.h> #include <netinet/in.h> #include <arpa/inet.h> #include<dirent.h>

#define MYPORT 4952 // the port users will be connecting to

#define MAXBUFLEN 200 int main()

{

int sockfd;

struct sockaddr\_in my\_addr; // my address information

struct sockaddr\_in their\_addr; // connector's address information

socklen\_t addr\_len; DIR \*d;

struct dirent \*dir; int numbytes;

char buf[MAXBUFLEN],buf1[MAXBUFLEN];

if ((sockfd = socket(AF\_INET, SOCK\_DGRAM, 0))

== -1) {

perror("socket"); exit(1);

}

my\_addr.sin\_family = AF\_INET; // host byte order

my\_addr.sin\_port = htons(MYPORT); // short, network byte order

my\_addr.sin\_addr.s\_addr = INADDR\_ANY; // automatically fill with my IP

//memset(my\_addr.sin\_zero, '\0', sizeof my\_addr.sin\_zero);

if (bind(sockfd, (struct sockaddr \*)&my\_addr, sizeof my\_addr) == -1) {

perror("bind"); exit(1);

}

addr\_len = sizeof their\_addr;

if ((numbytes = recvfrom(sockfd, buf, MAXBUFLEN-1 , 0,

(struct sockaddr \*)&their\_addr, &addr\_len))

== -1) {

perror("recvfrom"); exit(1);

}

char abc[10000]; strcpy(abc,"/home/gourav/CN\_LAB/"); strcat(abc,buf);

char mn[10]; strcpy(mn,"exit"); d= opendir(abc); if(d)

{

while((dir=readdir(d))!=NULL) printf("%s\n",dir->d\_name); closedir(d);

}

/\*printf("got packet

from %s\n",inet\_ntoa(their\_addr.sin\_addr)); printf("packet is %d bytes long\n",numbytes); buf[numbytes] = '\0';

printf("packet contains \"%s\"\n",buf);\*/ recvfrom(sockfd, buf1, MAXBUFLEN-1 , 0, (struct sockaddr \*)&their\_addr, &addr\_len); FILE \*fp;

char ch; fp=fopen(buf1,"r"); while(1)

{

ch=fgetc(fp);

if(ch==EOF) break; printf("%c",ch);

}

fclose(fp); close(sockfd); return 0;

}

/\*

\*\* A datagram "client" demo

\*/

#include <stdio.h> #include <stdlib.h> #include <unistd.h> #include <errno.h> #include <string.h> #include <sys/types.h> #include <sys/socket.h> #include <netinet/in.h> #include <arpa/inet.h> #include <netdb.h>

#define SERVERPORT 4952 // the port users will be connecting to

int main()

{

int sockfd;

struct sockaddr\_in their\_addr; // connector's address information

//struct hostent \*he; int numbytes;

char arg[30];

if ((sockfd = socket(AF\_INET, SOCK\_DGRAM, 0))

== -1) {

perror("socket"); exit(1);

}

their\_addr.sin\_family = AF\_INET; // host byte order

their\_addr.sin\_port = htons(SERVERPORT); // short, network byte order their\_addr.sin\_addr.s\_addr = inet\_addr("127.0.0.1");

//memset(their\_addr.sin\_zero, '\0', sizeof their\_addr.sin\_zero);

//addr\_len = sizeof their\_addr; printf("Enter the folder name\n"); scanf("%s",arg);

if ((numbytes = sendto(sockfd, arg, strlen(arg), 0,

(struct sockaddr \*)&their\_addr, sizeof their\_addr)) == -1) { perror("sendto");

exit(1);

}

/\*while(1)

{

recvfrom(sockfd, buf, MAXBUFLEN-1 , 0, (struct sockaddr \*)&their\_addr, &addr\_len); if(strcmp(buf,"exit")==0)

break; else

printf("%s\n",buf);

}\*/



//printf("sent %d bytes to %s\n", numbytes, inet\_ntoa(their\_addr.sin\_addr)); printf("\nEnter the file name with extension\n");

scanf("%s",arg);

sendto(sockfd, arg, strlen(arg), 0, (struct sockaddr \*)&their\_addr, sizeof their\_addr);

close(sockfd); return 0;

}