UCS1511 - COMPUTER NETWORKS

Simulation of Domain Name Server using UDP

REG NO : 205001085 **EX.NO : 10**
NAME : SABARIVASAN
DATE : 19.10.22

OBJECTIVE :

To be proficient in developing an application to simulate domain name server using socket programming in C

CODE :

CLIENT :

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

#define PORT     8080
#define MAXLINE 1024

int main() {
```

```
  int sockfd;
  char buffer[MAXLINE];
  //char *hello = "Hello from client";
char name[50];
  struct sockaddr_in     servaddr;


  if ( (sockfd = socket(AF_INET, SOCK_DGRAM, 0)) < 0 ) {
        perror("socket creation failed");
        exit(EXIT_FAILURE);
  }

  memset(&servaddr, 0, sizeof(servaddr));

  servaddr.sin_family = AF_INET;
  servaddr.sin_port = htons(PORT);
  servaddr.sin_addr.s_addr = INADDR_ANY;

  int n, len;
  printf("\nEnter the server name: ");
scanf(" %s",name);

  sendto(sockfd, (const char *)name, strlen(name),
        MSG_CONFIRM, (const struct sockaddr *) &servaddr,
              sizeof(servaddr));
  printf("Request Sent!\n");

  n = recvfrom(sockfd, (char *)buffer, MAXLINE,
                        MSG_WAITALL, (struct sockaddr *) &servaddr,
                        &len);
  buffer[n] = '\0';
  printf("Server : %s\n", buffer);
```

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

SERVER :

```
// Server side implementation of UDP client-server model
#include "dns.h"
#include <arpa/inet.h>
#include <netinet/in.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/socket.h>
#include <sys/types.h>
#include <unistd.h>

#define PORT 8080
#define MAXLINE 1024

int main() {

    dns values[15];
      int no = 3;

  // printf("\nEnter the no of websites: ");
  // scanf("%d", &no);
  /*for (int i = 0; i < no; i++) {
        values[i].n = 1;

        printf("\nEnter the name of website: ");
        scanf(" %s", values[i].name);
```

```
        printf("\nEnter the ip: ");
        scanf(" %s", values[i].ips[0]);

        // strcpy(values[i].name,w_name);
        // strcpy(values[i].ips[0],ip);
}*/
strcpy(values[0].name, "www.yahoo.com");
strcpy(values[0].ips[0], "10.2.45.67");
values[0].n = 1;

strcpy(values[1].name, "www.annauniv.edu");
strcpy(values[1].ips[0], "197.34.53.122");
values[1].n = 1;

strcpy(values[2].name, "www.google.com");
strcpy(values[2].ips[0], "142.89.78.66");
values[2].n = 1;

 int sockfd;
char buffer[MAXLINE];
char *retIP = (char *)malloc(sizeof(char) * 500);
struct sockaddr_in servaddr, cliaddr;

if ((sockfd = socket(AF_INET, SOCK_DGRAM, 0)) < 0) {
        perror("socket creation failed");
        exit(EXIT_FAILURE);
}

memset(&servaddr, 0, sizeof(servaddr));
memset(&cliaddr, 0, sizeof(cliaddr));
```

```
  servaddr.sin_family = AF_INET;
  servaddr.sin_addr.s_addr = INADDR_ANY;
  servaddr.sin_port = htons(PORT);

  if (bind(sockfd, (const struct sockaddr *)&servaddr, sizeof(servaddr))
< 0) {
        perror("bind failed");
        exit(EXIT_FAILURE);
  }

  int len, n, ch;

  len = sizeof(cliaddr);

  while (1) {
        printf("\n1.Modify\n2.Insert new\n3.Display\n4.Direct to
Listen\n5.Exit\nEnter your choice: ");
        scanf("%d", &ch);
        switch(ch)
        {
         case 1:    if(!append(values, no))
                      printf("\nInvalid server name!");
                display(values,no);
                break;
         case 2: no++;
                values[no-1].n = 1;
                printf("\nEnter the name of website: ");
                scanf(" %s", values[no-1].name);
                int k=0;
                while(!isValidIP(values[no-1].ips[0]))
           {
                if(k++)
```

```
                    printf("\nInvalid IP! Enter again\n");
                printf("\nEnter the ip: ");
                scanf(" %s", values[no-1].ips[0]);
            }
    case 3: display(values,no);
         break;
    case 4: break;
    case 5: return 0;
    default: printf("Invalid choice!\n");
        }
        printf("\n\nListening...\n");
        n = recvfrom(sockfd, (char *)buffer, MAXLINE, MSG_WAITALL,
                (struct sockaddr *)&cliaddr, &len);
        buffer[n] = '\0';
        printf("Client : %s\n", buffer);
        strcpy(retIP, search(values, no, buffer));
        sendto(sockfd, (const char *)retIP, strlen(retIP),
MSG_CONFIRM,
        (const struct sockaddr *)&cliaddr, len);
        printf("Reply Sent\n");
  }
  return 0;
}
```

OUTPUT :

```
root@spl16:~/Desktop/NetworksLAB/10-DNS/sabari ( works )# gcc server.c -o s
root@spl16:~/Desktop/NetworksLAB/10-DNS/sabari ( works )# ./s

1.Modify
2.Insert new
3.Display
4.Direct to Listen
5.Exit
Enter your choice: 3

Website name: www.yahoo.com
10.2.45.67
Website name: www.annauniv.edu
197.34.53.122
Website name: www.google.com
142.89.78.66

Listening...
3
Client : www.yahoo.com
Reply Sent

1.Modify
2.Insert new
3.Display
4.Direct to Listen
5.Exit
Enter your choice:
Website name: www.yahoo.com
10.2.45.67
Website name: www.annauniv.edu
197.34.53.122
Website name: www.google.com
142.89.78.66

Listening...
Client : www.yahoo.com
Reply Sent

1.Modify
2.Insert new
3.Display
4.Direct to Listen
5.Exit
Enter your choice: 4


Listening...
```

```
5.Exit
Enter your choice: 1

Enter the website name: www.yahoo.com

Enter the ip: 10.10.10.10

Website name: www.yahoo.com
10.2.45.67
10.10.10.10
Website name: www.annauniv.edu
197.34.53.122
Website name: www.google.com
142.89.78.66

Listening...
Client : www.yahoo.com
Reply Sent

1.Modify
2.Insert new
3.Display
4.Direct to Listen
5.Exit
Enter your choice: 2

Enter the name of website: sabari.com

Enter the ip: 10.20.30.40

Website name: www.yahoo.com
10.2.45.67
10.10.10.10
Website name: www.annauniv.edu
197.34.53.122
Website name: www.google.com
142.89.78.66
Website name: sabari.com
10.20.30.40

Listening...
Client : sabar.com
Reply Sent

1.Modify
2.Insert new
3.Display
4.Direct to Listen
```

```
1.Modify
2.Insert new
3.Display
4.Direct to Listen
5.Exit
Enter your choice: 4


Listening...
Client : sabari.com
Reply Sent

1.Modify
2.Insert new
3.Display
4.Direct to Listen
5.Exit
Enter your choice: 5
root@spl16:~/Desktop/NetworksLAB/10-DNS/sabari ( works )#
```

```
1.Modify
2.Insert new
3.Display
4.Direct to Listen
5.Exit
Enter your choice: 4


Listening...
Client : sabari.com
Reply Sent

1.Modify
2.Insert new
3.Display
4.Direct to Listen
5.Exit
Enter your choice: 5
root@spl16:~/Desktop/NetworksLAB/10-DNS/sabari ( works )#
```