



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«МИРЭА – Российский технологический университет»
РТУ МИРЭА

Институт информационных технологий
Кафедра вычислительной техники

Отчет по лабораторной работе № 7
по дисциплине
«Инструментальные средства разработки вычислительных
систем»

Тема работы:
«Клиент–серверное взаимодействие»

Выполнил: студент группы ИВБО-02-19

К. Ю. Денисов

Принял:

И. Р. Сон

Москва 2022

Цель работы

Цель работы практическое освоение механизма сокетов. Построении TCP-соединений для межпроцессного взаимодействия программ Клиента и Сервера в модели "Клиент-сервер"

Задание

Написать программы, демонстрирующие работу программ Клиента и Сервера в модели "Клиент-сервер".

Ход работы

Создадим два файла tcp-server.c и tcp-client.c, в которых опишем программы, реализующие сервер и клиент, взаимодействующие по протоколу TCP.

Исходный файл соответствующих файлов приведены в листингах 1 и 2.

Листинг 1 — tcp-server.c

```
1 #include<stdio.h>
2 #include<sys/types.h>
3 #include<sys/socket.h>
4 #include<netinet/in.h>
5 #include <unistd.h>
6 #include <stdlib.h>
7 int main()
8 {
9     int sockfd, clientfd;
10    struct sockaddr_in serverAddr, cliAddr;
11    socklen_t addr_size;
12    int len = sizeof(cliAddr);
13    char serverMessage[256] = "Hello from server\n";
14    char clientResponse[1000];
15
16    //create a socket file descriptor
17    sockfd = socket(PF_INET, SOCK_STREAM, 0);
18
19    //fill the serverAddr structure
20    serverAddr.sin_family = AF_INET;
21    serverAddr.sin_port = htons(9002);
22    serverAddr.sin_addr.s_addr = INADDR_ANY;
```

```

24 // bind the address
if(bind(sockfd, (struct sockaddr *) &serverAddr, sizeof(serverAddr))!=0)
26 {
    perror("server.bind()");
28     exit(1);
    }

30 // listen
32 if(listen(sockfd,5)==0)
    {

34 //accept the connection
36 clientfd = accept(sockfd, (struct sockaddr *) &cliAddr, &len);

38 // send the data
if (send(clientfd,serverMessage,sizeof(serverMessage),0)==-1)
40 {
    perror("server.send()");
42     exit(1);
    }
44 if (recv(clientfd,&clientResponse,sizeof(clientResponse),0)==-1)
    {
46     perror("server.recv()");
        exit(1);
48     }
    printf("Received data from client : %s\n", clientResponse);

50     close(sockfd);
52 }
    return 0;
54 }

```

Листинг 2 — tcp-client.c

```

#include <stdio.h>
2 #include <stdlib.h>

4 #include <sys/types.h>
#include <sys/socket.h>

6 #include <netinet/in.h>
8 #include <unistd.h>
#define SIZE 1000

10 //main functions
12 int main()
    {
14 int sockfd = socket(AF_INET, SOCK_STREAM, 0);
    // server address

```

```

16 struct sockaddr_in serverAddress;
   serverAddress.sin_family = AF_INET;
18 serverAddress.sin_port = htons(9002);
   serverAddress.sin_addr.s_addr = INADDR_ANY;
20
   // communicates with listen
22 if (connect(sockfd, (struct sockaddr *)&serverAddress, sizeof(serverAddress))
       !=0)
   {
24 perror("client.connect()");
   exit(1);
26 }


28 char serverResponse[SIZE];
   char clientMessage[SIZE]="Message from client";
30 if (recv(sockfd, &serverResponse, sizeof(serverResponse), 0)==-1)
   {
32 perror("client.recv()");
   exit(1);
34 }
   printf("Received data from server : %s", serverResponse);
36 if (send(sockfd, clientMessage, sizeof(serverResponse), 0)==-1)
   {
38 perror("client.send()");
   exit(1);
40 }

42 //closing the socket
   close(sockfd);
44 return 0;
   }

```

Выполним сборку программ с помощью команды `$ gcc tcp-server.c -o tcp-server` и `$ gcc tcp-client.c -o tcp-client`.

Для того, чтобы продемонстрировать работу программ, запустим два исполняемых файла в разных сессиях терминала. Сначала следует запустить программу-сервер (см. рисунок 1).



```
denilai:client-server$ ./tcp-server
Received data from client : Message from client
denilai:client-server$

denilai:client-server$ ./tcp-client
Received data from server : Hello from server
denilai:client-server$
```

Рисунок 1 — Демонстрация работы клиента и сервера

Вывод

В ходе настоящей лабораторной работы были созданы программы для демонстрации работу Клиента и Сервера в модели "Клиент-сервер".