

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

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 ТСР.

Исходный файл соответствующих файлов приведены в листингах 1 и 2.

```
#include<stdio.h>
 #include<sys/types.h>
 #include<sys/socket.h>
 #include<netinet/in.h>
  #include <unistd.h>
 #include <stdlib.h>
  int main()
8 {
    int sockfd, clientfd;
   struct sockaddr_in serverAddr, cliAddr;
   socklen_t addr_size;
   int len = sizeof(cliAddr);
   char serverMessage[256] = "Hello from server\n";
   char clientResponse[1000];
   //create a socket file descriptor
    sockfd = socket(PF_INET, SOCK_STREAM, 0);
18
   //fill the serverAddr structure
   serverAddr.sin family = AF INET;
   serverAddr.sin_port = htons(9002);
   serverAddr.sin_addr.s_addr = INADDR_ANY;
```

```
// bind the address
    if(bind(sockfd, (struct sockaddr *) &serverAddr, sizeof(serverAddr))!=0)
      perror("server.bind()");
      exit(1);
28
    }
30
    // listen
    if(listen(sockfd,5)==0)
34
      //accept the connection
      clientfd = accept(sockfd, (struct sockaddr *) &cliAddr, &len);
36
      // send the data
      if (send(clientfd, serverMessage, sizeof(serverMessage), 0) ==-1)
40
        perror("server.send()");
        exit(1);
42
      if (recv(clientfd, &clientResponse, sizeof(clientResponse), 0) ==-1)
        perror("server.recv()");
46
        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>
#include <stdlib.h>

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

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

//main functions
int main()
{
   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;
 // communicates with listen
22 if (connect(sockfd, (struct sockaddr *)&serverAddress, sizeof(serverAddress))
24 perror("client.connect()");
 exit(1);
26 }
28 char serverResponse[SIZE];
 char clientMessage[SIZE]="Message from client";
if (recv(sockfd, &serverResponse, sizeof(serverResponse), 0)==-1)
perror("client.recv()");
 exit(1);
 printf("Received data from server : %s", serverResponse);
36 if (send(sockfd, clientMessage, sizeof(serverResponse), 0)==-1)
perror("client.send()");
 exit(1);
40 }
42 //closing the socket
 close(sockfd);
44 return 0;
```

Выполним сборку программ с помощью команды \$ gcc 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).

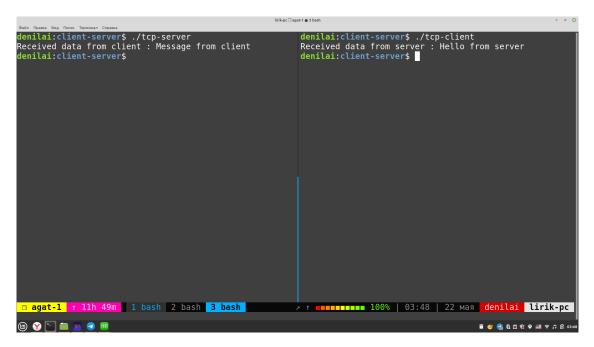


Рисунок 1 — Демонстрация работы клиента и сервера

Вывод

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