

Практикум на ЭВМ client-server

Баев А.Ж.

Казахстанский филиал МГУ

25 ноября 2021

Клиент-сервер.

1. Сервер - тот кто ждет подключения

2. Клиент - тот кто подключается

Сервер

```
1  int port = 5000;
2  int server_socket, client_socket;
3  struct sockaddr_in client_address;
4  socklen_t size = sizeof client_address;
5  char data[12] = "Hello_world";
6
7  server_socket = init_socket(port);
8  for (;;) {
9      int client_socket = accept(
10         server_socket,
11         (struct sockaddr *) &client_address,
12         &size
13     );
14     write(client_socket, data, sizeof data);
15     close(client_socket);
16 }
17 close(server_socket);
```

Клиент

```
1 char ip[] = "192.168.0.1";
2 int port = 5000;
3 char data[12];
4
5 int server_socket = init_socket(ip, port); //TODO
6 read(server_socket, data, sizeof data);
7 close(server);
```

Сервер. Создание сокета

```
1  int server_socket;
2  server_socket = socket(AF_INET, SOCK_STREAM, 0);
3  if (server_socket < 0) {
4      perror("Fail: _open_socket");
5      exit(ERR_SOCKET);
6  }
7
8  int socket_option = 1;
9  setsockopt(
10     server_socket, SOL_SOCKET, SO_REUSEADDR,
11     &socket_option, socklen_t sizeof socket_option);
12  if (server_socket < 0) {
13      perror("Fail: _set_socket_options");
14      exit(ERR_SETSOCKETOPT);
15  }
```

Сервер. Привязка адреса

```
1 struct sockaddr_in server_address;  
2 server_address.sin_family = AF_INET;  
3 server_address.sin_port = htons(port);  
4 server_address.sin_addr.s_addr = INADDR_ANY;  
5 if (bind(server_socket,  
6         (struct sockaddr *) &server_address,  
7         (socklen_t) sizeof server_address) < 0) {  
8     perror("Fail: bind socket address");  
9     exit(ERR_BIND);  
10 }  
11  
12 if (listen(server_socket, 5) < 0) {  
13     perror("Fail: bind socket address");  
14     exit(ERR_LISTEN);  
15 }
```

Клиент подключение

```
1  int server_socket = socket(AF_INET, SOCK_STREAM, 0);
2  struct hostent *host = gethostbyname(ip);
3  struct sockaddr_in server_address;
4  if (server_socket < 0) {
5      perror("Fail: open socket");
6      exit(ERR_SOCKET);
7  }
8
9  server_address.sin_family = AF_INET;
10 server_address.sin_port = htons(port);
11 memcpy(&server_address.sin_addr,
12        host->h_addr_list[0],
13        (socklen_t) sizeof server_address.sin_addr);
14 if (connect(server_socket,
15             (struct sockaddr*) &server_address,
16             (socklen_t) sizeof server_address) < 0) {
17     perror("Fail: connect");
18     exit(ERR_CONNECT);
```