OS Lab 33

Status	approved
checkbox	☑
class	OS
due date	@Apr 21, 2021

```
struct hostent {
    char    *h_name;        /* canonical name of host */
    char    **h_aliases;    /* alias list */
    int     h_addrtype;     /* host address type */
    int     h_length;       /* length of address */
    char    **h_addr_list;  /* list of addresses */
};
```

```
cc [flag... ]
file...–lnsl [library... ]
#include <netdb.h>
struct hostent *gethostbyname(
const char *name);
```

```
struct sockaddr_in {
  sa_family_t     sin_family;  /* семейство адресов */
  in_port_t       sin_port;    /* номер порта */
  struct in_addr  sin_addr;    /* адрес IPv4 */
  unsigned char   sin_zero[8]; /* заполнитель */
};
```

https://man7.org/linux/man-pages/man2/sigaction.2.html

Solaris 10	Sun SPARC	Прямой (big-endian)

Некоторые типы процессоров допускают возможность изменения порядка байтов, что вносит еще большую путаницу.

Чтобы не возникало путаницы с порядком байтов при обмене данными между разнородными компьютерными системами, сетевые протоколы жестко задают порядок байтов. Набор протоколов TCP/IP использует сетевой (прямой, big-endian) порядок байтов. Порядок байтов приобретает важность, когда приложения начинают обмениваться форматированными данными. При использовании протоколов TCP/IP адреса имеют сетевой порядок байтов, поэтому в приложениях иногда возникает необходимость преобразовать порядок байтов, поддерживаемый аппаратной архитектурой, в сетевой порядок байтов. Такое преобразование обычно производится, например, при выводе адреса в удобочитаемой форме.

Преобразования между сетевым и аппаратным порядком байтов производятся с помощью следующих четырех функций.

```
#include <arpa/inet.h>
uint32_t htonl(uint32_t hostint32);
```
Возвращает 32-разрядное целое с сетевым порядком байтов
```
uint16_t htons(uint16_t hostint16);
```
Возвращает 16-разрядное целое с сетевым порядком байтов
```
uint32_t ntohl(uint32_t netint32);
```
Возвращает 32-разрядное целое с аппаратным порядком байтов
```
uint16_t ntohs(uint16_t netint16);
```
Возвращает 16-разрядное целое с аппаратным порядком байтов

16.3.4. Присваивание адресов сокетам

Адрес, присваиваемый клиентскому сокету, не представляет для нас особого интереса, потому мы можем позволить системе выбирать адрес по умолчанию. Однако для сервера важно присвоить сокету предопределенный адрес, на который клиенты будут присылать запросы. Клиентам необходимо заранее знать требуемый адрес, чтобы войти в контакт с сервером, и самое простое решение заключается в том, чтобы зарезервировать адрес сервера в файле `/etc/services` или в службе имен.

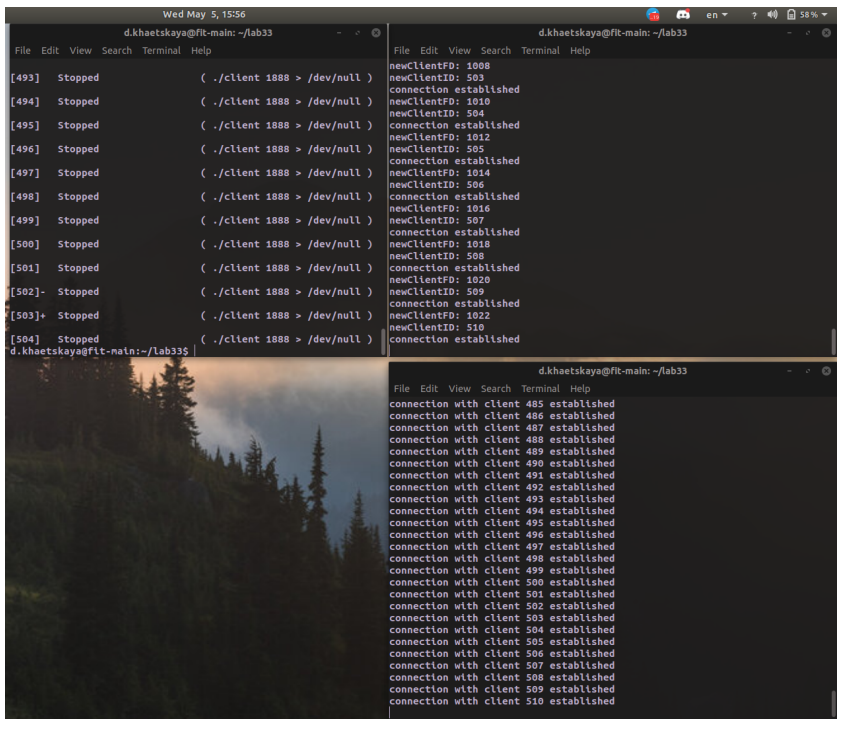
Присвоить адрес сокету можно с помощью функции `bind`.

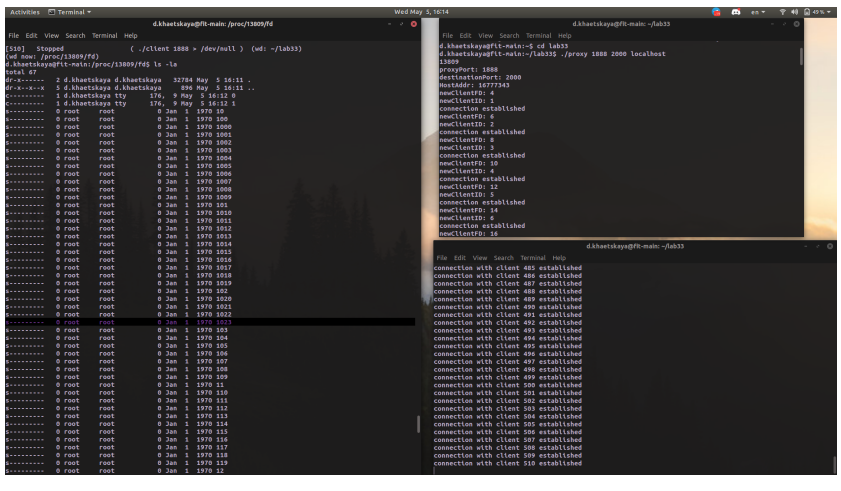
```
#include <sys/socket.h>
int bind(int sockfd, const struct sockaddr *addr, socklen_t len);
```
Возвращает 0 в случае успеха, –1 — в случае ошибки

Существует несколько ограничений, касающихся адресов:

- Указываемый адрес должен быть действительным адресом для машины, на которой выполняется процесс, — нельзя задать адрес, который принадлежит другой машине.
- Формат адреса должен совпадать с форматом, который поддерживается семейством адресов, указанным при создании сокета.
- Номер порта не может быть меньше 1024, если процесс не имеет соответствующих привилегий (например, привилегий суперпользователя).
- Обычно каждый конкретный адрес может быть связан только с одним сокетом, хотя некоторые протоколы допускают присвоение одного и того же адреса нескольким сокетам.

В домене Интернета имеется специальный IP-адрес `INADDR_ANY`, который соответствует адресам всех сетевых интерфейсов в системе. Это означает, что существует возможность принимать пакеты с любого сетевого интерфейса, установленного в системе. В следующем разделе мы увидим, что система сама может присвоить адрес сокету при обращении к функциям `connect` и `listen`.


Wed May 5, 15:56
d.khaetskaya@fit-main: ~/lab33
File Edit View Search Terminal Help
[493] Stopped (./client 1888 > /dev/null)
[494] Stopped (./client 1888 > /dev/null)
[495] Stopped (./client 1888 > /dev/null)
[496] Stopped (./client 1888 > /dev/null)
[497] Stopped (./client 1888 > /dev/null)
[498] Stopped (./client 1888 > /dev/null)
[499] Stopped (./client 1888 > /dev/null)
[500] Stopped (./client 1888 > /dev/null)
[501] Stopped (./client 1888 > /dev/null)
[502]- Stopped (./client 1888 > /dev/null)
[503]+ Stopped (./client 1888 > /dev/null)
[504] Stopped (./client 1888 > /dev/null)
d.khaetskaya@fit-main:~/lab33$
newClientFD: 1008
newClientID: 503
connection established
newClientFD: 1010
newClientID: 504
connection established
newClientFD: 1012
newClientID: 505
connection established
newClientFD: 1014
newClientID: 506
connection established
newClientFD: 1016
newClientID: 507
connection established
newClientFD: 1018
newClientID: 508
connection established
newClientFD: 1020
newClientID: 509
connection established
newClientFD: 1022
newClientID: 510
connection established
connection with client 485 established
connection with client 486 established
connection with client 487 established
connection with client 488 established
connection with client 489 established
connection with client 490 established
connection with client 491 established
connection with client 492 established
connection with client 493 established
connection with client 494 established
connection with client 495 established
connection with client 496 established
connection with client 497 established
connection with client 498 established
connection with client 499 established
connection with client 500 established
connection with client 501 established
connection with client 502 established
connection with client 503 established
connection with client 504 established
connection with client 505 established
connection with client 506 established
connection with client 507 established
connection with client 508 established
connection with client 509 established
connection with client 510 established


Activities Terminal
Wed May 5, 16:14
d.khaetskaya@fit-main: /proc/13809/fd
[510] Stopped (./client 1888 > /dev/null) (wd: ~/lab33)
(wd now: /proc/13809/fd)
d.khaetskaya@fit-main:/proc/13809/fd$ ls -la
total 67
d.khaetskaya@fit-main:~$ cd lab33
d.khaetskaya@fit-main:~/lab33$ ./proxy 1888 2000 localhost
13809
proxyPort: 1888
destinationPort: 2000
HostAddr: 16777343
newClientFD: 4
newClientID: 1
connection established
newClientFD: 6
newClientID: 2
connection established
newClientFD: 8
newClientID: 3
connection established
newClientFD: 10
newClientID: 4
connection established
newClientFD: 12
newClientID: 5
connection established
newClientFD: 14
newClientID: 6
connection established
newClientFD: 16
connection with client 485 established
connection with client 486 established
connection with client 487 established
connection with client 488 established
connection with client 489 established
connection with client 490 established
connection with client 491 established
connection with client 492 established
connection with client 493 established
connection with client 494 established
connection with client 495 established
connection with client 496 established
connection with client 497 established
connection with client 498 established
connection with client 499 established
connection with client 500 established
connection with client 501 established
connection with client 502 established
connection with client 503 established
connection with client 504 established
connection with client 505 established
connection with client 506 established
connection with client 507 established
connection with client 508 established
connection with client 509 established
connection with client 510 established