

网络

<https://ifconfig.me/> 查询自己的公网ip

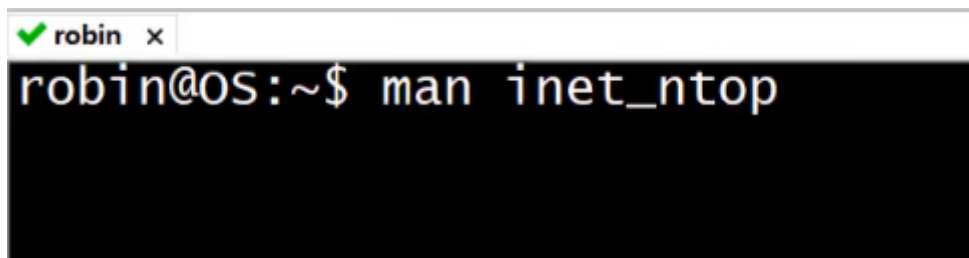
18.162.123.223 (远程服务器能ping通)

命令行输入:

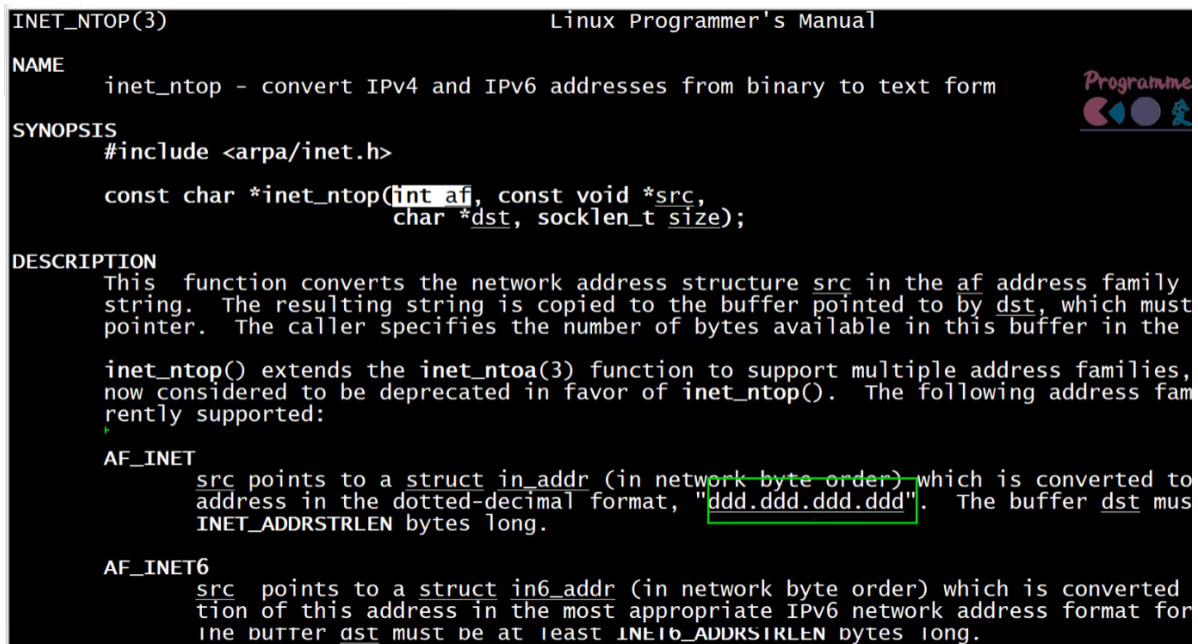
curl ifconfig.me

输出: 210.30.193.187 (ping不通, 可能原因是校园网是局域网)

linux:查看man文档



```
robin@os:~$ man inet_ntop
```



```
INET_NTOP(3)                                Linux Programmer's Manual

NAME
  inet_ntop - convert IPv4 and IPv6 addresses from binary to text form

SYNOPSIS
  #include <arpa/inet.h>

  const char *inet_ntop(int af, const void *src,
                        char *dst, socklen_t size);

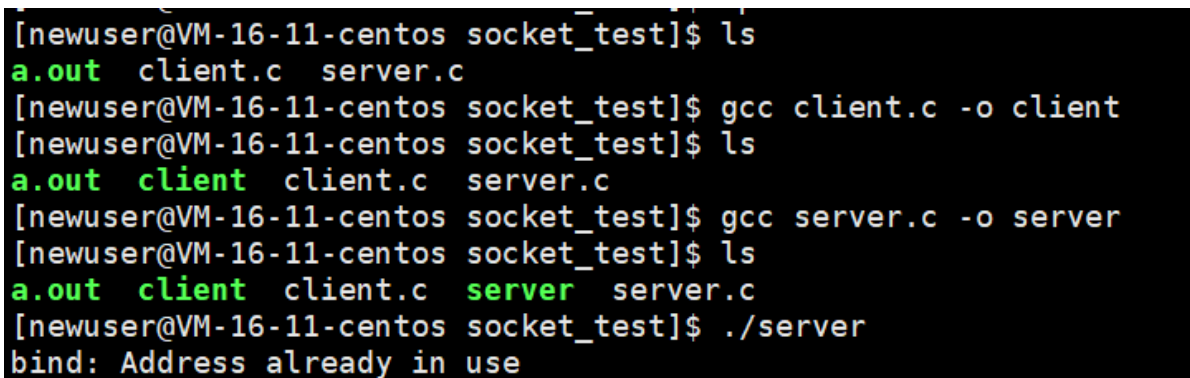
DESCRIPTION
  This function converts the network address structure src in the af address family string. The resulting string is copied to the buffer pointed to by dst, which must be a pointer. The caller specifies the number of bytes available in this buffer in the size argument.

  inet_ntop() extends the inet_ntoa(3) function to support multiple address families, now considered to be deprecated in favor of inet_ntop(). The following address families are currently supported:

  AF_INET
    src points to a struct in_addr (in network byte order) which is converted to address in the dotted-decimal format, "ddd.ddd.ddd.ddd". The buffer dst must be at least INET_ADDRSTRLEN bytes long.

  AF_INET6
    src points to a struct in6_addr (in network byte order) which is converted to address in the most appropriate IPv6 network address format for the address. The buffer dst must be at least INET6_ADDRSTRLEN bytes long.
```

编译运行, 报错: bind: Address already in use



```
[newuser@VM-16-11-centos socket_test]$ ls
a.out client.c server.c
[newuser@VM-16-11-centos socket_test]$ gcc client.c -o client
[newuser@VM-16-11-centos socket_test]$ ls
a.out client client.c server.c
[newuser@VM-16-11-centos socket_test]$ gcc server.c -o server
[newuser@VM-16-11-centos socket_test]$ ls
a.out client client.c server server.c
[newuser@VM-16-11-centos socket_test]$ ./server
bind: Address already in use
```

原因: 9999端口被占用

解决:

1、使用 netstat -tulpn 查看 端口使用情况

netstat -tulpn

2、找到被占用的端口 (确认该端口对应的服务是即将启用的服务)

netstat -tulpn | grep 9999

```
[newuser@VM-16-11-centos socket_test]$ netstat -tulpn | grep 9999
(Not all processes could be identified, non-owned process info
will not be shown, you would have to be root to see it all.)
tcp        0      0 0.0.0.0:9999 0.0.0.0:*        LISTEN      -
```

返回如下: 没有root权限, 信息展示不全

root后:

```
[newuser@VM-16-11-centos socket_test]$ su
Password:
[root@VM-16-11-centos socket_test]# netstat -tulpn | grep 9999
tcp        0      0 0.0.0.0:9999 0.0.0.0:*        LISTEN      8598/ChatServer.
out
```

3、释放对应的端口pid(8598)

```
[root@VM-16-11-centos socket_test]# kill -9 8598
```