UNIX / Linux: 10 Netstat Command Examples

by SATHIYAMOORTHY *on* MARCH 29, 2010

Netstat command displays various network related information such as network connections, routing tables, interface statistics, masquerade connections, multicast memberships etc.,

In this article, let us review 10 practical unix **netstat command** examples.

1. List All Ports (both listening and non listening ports)

List all ports using netstat -a

```
# netstat -a | more

Active Internet connections (servers and established)

Proto Recv-Q Send-Q Local Address           Foreign Address         State

tcp        0      0 localhost:30037         *:*                     LISTEN

udp        0      0 *:bootpc                *:*



Active UNIX domain sockets (servers and established)

Proto RefCnt Flags       Type       State         I-Node   Path

unix  2      [ ACC ]     STREAM     LISTENING     6135     /tmp/.X11-unix/X0

unix  2      [ ACC ]     STREAM     LISTENING     5140     /var/run/acpid.socket
```

List all tcp ports using netstat -at

```
# netstat -at

Active Internet connections (servers and established)

Proto Recv-Q Send-Q Local Address           Foreign Address         State
```

```
tcp        0      0 localhost:30037         *:*                     LISTEN

tcp        0      0 localhost:ipp           *:*                     LISTEN

tcp        0      0 *:smtp                  *:*                     LISTEN

tcp6       0      0 localhost:ipp           [::]:*                  LISTEN
```

List all udp ports using netstat -au

```
# netstat -au

Active Internet connections (servers and established)

Proto Recv-Q Send-Q Local Address           Foreign Address         State

udp        0      0 *:bootpc                *:*

udp        0      0 *:49119                 *:*

udp        0      0 *:mdns                  *:*
```

2. List Sockets which are in Listening State

List only listening ports using netstat -l

```
# netstat -l

Active Internet connections (only servers)

Proto Recv-Q Send-Q Local Address           Foreign Address         State

tcp        0      0 localhost:ipp           *:*                     LISTEN

tcp6       0      0 localhost:ipp           [::]:*                  LISTEN

udp        0      0 *:49119                 *:*
```

List only listening TCP Ports using netstat -lt

```
# netstat -lt
```

```
Active Internet connections (only servers)

Proto Recv-Q Send-Q Local Address           Foreign Address         State

tcp        0      0 localhost:30037         *:*                     LISTEN

tcp        0      0 *:smtp                  *:*                     LISTEN

tcp6       0      0 localhost:ipp           [::]:*                  LISTEN
```

List only listening UDP Ports using netstat -lu

```
# netstat -lu

Active Internet connections (only servers)

Proto Recv-Q Send-Q Local Address           Foreign Address         State

udp        0      0 *:49119                 *:*

udp        0      0 *:mdns                  *:*
```

List only the listening UNIX Ports using netstat -lx

```
# netstat -lx

Active UNIX domain sockets (only servers)

Proto RefCnt Flags       Type       State         I-Node   Path

unix  2      [ ACC ]     STREAM     LISTENING     6294     private/maildrop

unix  2      [ ACC ]     STREAM     LISTENING     6203     public/cleanup

unix  2      [ ACC ]     STREAM     LISTENING     6302     private/ifmail

unix  2      [ ACC ]     STREAM     LISTENING     6306     private/bsmtp
```

3. Show the statistics for each protocol

Show statistics for all ports using netstat -s

```
# netstat -s

Ip:

    11150 total packets received

    1 with invalid addresses

    0 forwarded

    0 incoming packets discarded

    11149 incoming packets delivered

    11635 requests sent out

Icmp:

    0 ICMP messages received

    0 input ICMP message failed.

Tcp:

    582 active connections openings

    2 failed connection attempts

    25 connection resets received

Udp:

    1183 packets received

    4 packets to unknown port received.

.....
```

Show statistics for TCP (or) UDP ports using netstat -st (or) -su

```
# netstat -st



# netstat -su
```

4. Display PID and program names in netstat output using netstat -p

netstat -p option can be combined with any other netstat option. This will add the “PID/Program Name” to the netstat output. This is very useful while debugging to identify which program is running on a particular port.

```
# netstat -pt

Active Internet connections (w/o servers)

Proto Recv-Q Send-Q Local Address           Foreign Address         State
PID/Program name

tcp        1      0 ramesh-laptop.loc:47212 192.168.185.75:www        CLOSE_WAIT
2109/firefox

tcp        0      0 ramesh-laptop.loc:52750 lax:www ESTABLISHED 2109/firefox
```

5. Don’t resolve host, port and user name in netstat output

When you don’t want the name of the host, port or user to be displayed, use netstat -n option. This will display in numbers, instead of resolving the host name, port name, user name.

This also speeds up the output, as netstat is not performing any look-up.

```
# netstat -an
```

If you don’t want only any one of those three items (ports, or hosts, or users) to be resolved, use following commands.

```
# netsat -a --numeric-ports



# netsat -a --numeric-hosts



# netsat -a --numeric-users
```

6. Print netstat information continuously

netstat will print information continuously every few seconds.

```
# netstat -c

Active Internet connections (w/o servers)

Proto Recv-Q Send-Q Local Address           Foreign Address         State

tcp        0      0 ramesh-laptop.loc:36130 101-101-181-225.ama:www ESTABLISHED

tcp        1      1 ramesh-laptop.loc:52564 101.11.169.230:www      CLOSING

tcp        0      0 ramesh-laptop.loc:43758 server-101-101-43-2:www ESTABLISHED

tcp        1      1 ramesh-laptop.loc:42367 101.101.34.101:www      CLOSING

^C
```

7. Find the non supportive Address families in your system

```
netstat --verbose
```

At the end, you will have something like this.

```
        netstat: no support for `AF IPX' on this system.
```

```
        netstat: no support for `AF AX25' on this system.

        netstat: no support for `AF X25' on this system.

        netstat: no support for `AF NETROM' on this system.
```

8. Display the kernel routing information using netstat -r

```
# netstat -r

Kernel IP routing table

Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface

192.168.1.0     *               255.255.255.0   U         0 0          0 eth2

link-local      *               255.255.0.0     U         0 0          0 eth2

default         192.168.1.1     0.0.0.0         UG        0 0          0 eth2
```

Note: Use netstat -rn to display routes in numeric format without resolving for host-names.

9. Find out on which port a program is running

```
# netstat -ap | grep ssh

(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        1      0 dev-db:ssh           101.174.100.22:39213        CLOSE_WAIT  -

tcp        1      0 dev-db:ssh           101.174.100.22:57643        CLOSE_WAIT  -
```

Find out which process is using a particular port:

```
# netstat -an | grep ':80'
```

10. Show the list of network interfaces

```
# netstat -i

Kernel Interface table

Iface   MTU Met   RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg

eth0       1500 0         0      0      0 0             0      0      0      0 BMU

eth2       1500 0     26196      0      0 0         26883      6      0      0 BMRU

lo        16436 0         4      0      0 0             4      0      0      0 LRU
```

Display extended information on the interfaces (similar to ifconfig) using netstat -ie:

```
# netstat -ie

Kernel Interface table

eth0      Link encap:Ethernet  HWaddr 00:10:40:11:11:11

          UP BROADCAST MULTICAST  MTU:1500  Metric:1

          RX packets:0 errors:0 dropped:0 overruns:0 frame:0

          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0

          collisions:0 txqueuelen:1000

          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

          Memory:f6ae0000-f6b00000
```