Netstat a fundo (parte 1)

Autor: Thiago Rodrigues <thirosantos at gmail.com>
Data: 03/11/2008

O comando netstat

Bem,

Vejo muito amigos e até eu mesmo usando o comando *netstat* para ver as conexões que um determinado host está abrindo com o mundo e vice e versa, porém poucos parâmetros desse comando são utilizados.

Vou exibir novos artigos falando em específico da tratativa de algumas saídas do netstat, principalmente a de rotas e a de interfaces, onde podemos pegar muitos indícios de erro.

Vou tentar ser o mais objetivo possível e espero ajudar o máximo:

 # netstat -a

Mostra todas as conexões do computador, incluindo todos os protocolos e sockets (TCP, UDP, RAW).

```
Conexões internet ativas (servidores e estabelecidas)

Proto Recv-Q Send-Q Endereço Local          Endereço Remoto         Estado
tcp        0      0 *:10050                 *:*                     OU
tcp        0      0 localhost:mysql         *:*                     OU
tcp        0      0 localhost:submission    *:*                     OU
tcp        0      0 *:netbios-ssn           *:*                     OU
tcp        0      0 *:pop3                  *:*                     OU
tcp        0      0 *:www                   *:*                     OU
tcp        0      0 *:ftp                   *:*                     OU
tcp6       0      0 [::]:ssh                [::]:*                  OU
tcp6       0    132 teste.com.br:ssh    adsl.com.br:4714   ESTABELECIDA
tcp6       0  11844 teste.com.br:ssh    adsl.com.br:2288   ESTABELECIDA
udp        0      0 volvo.nti.uf:netbios-ns *:*
udp        0      0 *:netbios-ns            *:*
udp        0      0 volvo.nti.u:netbios-dgm *:*
udp        0      0 *:netbios-dgm           *:*
udp        0      0 *:46237                 *:*
udp        0      0 localhost:snmp          *:*
udp        0      0 *:718                   *:*
udp        0      0 *:mdns                  *:*
udp        0      0 *:sunrpc                *:*
udp        0      0 *:35573                 *:*
Domain sockets UNIX ativos (servidores e estabelecidas)
Proto RefCnt Flags       Type       State         I-Node   Caminho
unix  2      [ ACC ]     STREAM     OUVINDO       14901    @ISCSIADM_ABSTRACT_NAMESPACE
unix  2      [ ACC ]     STREAM     OUVINDO       17267    @/var/run/dbus-z0P4AMwzz6
unix  2      [ ACC ]     STREAM     OUVINDO       17462    /var/run/gdm_socket
unix  2      [ ]         DGRAM                    6511     @/com/ubuntu/upstart
unix  2      [ ACC ]     STREAM     OUVINDO       17311    @/org/bluez/audio
unix  16     [ ]         DGRAM                    15719    /dev/log
unix  2      [ ACC ]     STREAM     OUVINDO       17510    /tmp/.X11-unix/X0
unix  2      [ ACC ]     STREAM     OUVINDO       16234    /var/run/atieventsd.socket
unix  2      [ ]         DGRAM                    6667     @/org/kernel/udev/udevd
unix  2      [ ACC ]     STREAM     OUVINDO       16662    @/var/run/hald/dbus-GQ71iqMReW
unix  2      [ ACC ]     STREAM     OUVINDO       17265    /var/run/sdp
unix  2      [ ACC ]     STREAM     OUVINDO       16007    /var/run/avahi-daemon/socket
```

As opções -t, -u, -w e -x exibem as atividades dos protocolos TCP, UDP, RAW ou Unix Socket respectivamente. Então a combinação pode variar, conforme abaixo:

- netstat -at (todas as conexões TCP)
- netstat -au (todas as conexões UDP)
- netstat -aw (todas as conexões RAW)
- netstat -ax (todas as conexões Unix Socket)
- netstat -aut (todas as conexões TCP e UDP)

E assim sucessivamente...

netstat -na

A opção acima é um dos mais interessantes ao meu ver, falando em especial do -n, que faz com que o comando não tente resolver nomes através de consulta ao DNS. Imagine um servidor onde temos milhares de conexões, se não usarmos o -n ficaríamos facilmente alguns bons minutos esperando o comando terminar por completo.

Pode-se combinar a vontade, como por exemplo: netstat -autn , netstat -axn etc.

netstat -r ou sem resolver nomes, netstat -nr

Exibe as rotas do seu computador, novamente, ao omitir a opção -n o comando tentará resolver todos os IPs para nome.

netstat -o

Mostra o temporizador da conexão, ou seja, a quanto tempo essa conexão está estabelecida, pode-se combinar a vontade: netstat -autno, netstat -axuo.

netstat -i

Exibe as informações de todas as interfaces ativas. Podemos ter estatísticas de erros de entrada/saída, assim com estatística de trafego.

netstat -c

Repete o comando ao final, muito útil para verificar o momento exato que uma conexão é estabelecia ou para ter noção do aumento de tráfego nas interfaces, ex.: netstat -ic , netstat -atnc.

netstat -e

Exibe uma lista mais completa. Deve ser combinado com as outras opções, como por exemplo o netstat -atne.

Com esse comando temos mais duas colunas, USER e INODE, ou seja, o usuário que subiu o processo que originou a abertura da porta e o INODE pertencente.

netstat -p

Exibe o daemon e o PID que estão ligados a essa porta, muito importante para detectarmos o daemon responsável.

netstat -s

Exibe as estatísticas dos protocolos, ou seja, quanto foi trafegado em cada protocolo. Podemos combinar para assim pegarmos a estatística de um determinado protocolo, ex.: netstat -st, netstat -su.

Acho que é isso, a principal idéia a ser passada com esse artigo, são as inúmeras combinações que podemos fazer para obter o resultado mais adequado.

Comentários são bem vindos.

Thiago Rodrigues - Miombo

Netstat (NETwork STATistics) is a command-line tool that provides information about your network configuration and activity.

- To display the routing table:

#netstat -rn

- -r: Kernel routing tables.
- -n: Shows numerical addresses instead of trying to determine hosts.

Kernel IP routing table

Destination	*Gateway*	*Genmask*	*Flags*	*MSS*	*Window*	*irtt*	*Iface*
192.168.1.0	*0.0.0.0*	*255.255.255.0*	*U*	*0*	*0*	*0*	*eth1*
0.0.0.0	*192.168.1.1*	*0.0.0.0*	*UG*	*0*	*0*	*0*	*eth1*

- To display the quick interfaces statistics:

#netstat -i

- -i: Interface

Kernel Interface table

Iface	*MTU*	*Met*	*RX-OK*	*RX-ERR*	*RX-DRP*	*RX-OVR*	*TX-OK*	*TX-ERR*	*TX-DRP*	*TX-OVR*	*FLG*
ath0	*1500*	*0*	*0*	*250*	*0*	*0*	*0*	*0*	*0*	*0*	*BMRU*
eth0	*1500*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*BMU*
eth1	*1500*	*0*	*1156*	*0*	*0*	*0*	*568*	*0*	*0*	*0*	*BMRU*
lo	*16436*	*0*	*225*	*0*	*0*	*0*	*225*	*0*	*0*	*0*	*LRU*

- To display the extended interfaces statistics:

#netstat -ie

- -i: Interface
- -e: Extended information

Kernel Interface table

eth0 Link encap:Ethernet HWaddr AA:00:11:22:33:44
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)
Interrupt:169

eth1 Link encap:Ethernet HWaddr AA:00:11:22:33:44
inet addr:192.168.1.101 Bcast:192.168.1.255 Mask:255.255.255.0
inet6 addr: fe80::a100:0aa:aa00:a01/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:1212 errors:0 dropped:0 overruns:0 frame:0
TX packets:580 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000

RX bytes:216479 (211.4 KiB) TX bytes:56987 (55.6 KiB)
Interrupt:201 Memory:dfcff000-dfcfffff

lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:16436 Metric:1
RX packets:238 errors:0 dropped:0 overruns:0 frame:0
TX packets:238 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:8688 (8.4 KiB) TX bytes:8688 (8.4 KiB)

Note that "netstat -ie" is equivalent to "ifconfig -a".

- To display all the opened network sockets:

#netstat -uta
- -u: UDP
- -t: TCP
- -a: All

Active Internet connections (servers and established)

Proto	*Recv-Q*	*Send-Q*	*Local Address*	*Foreign Address*	*State*
tcp	*0*	*0*	*localhost:48898*	**:**	*LISTEN*
tcp	*0*	*0*	*localhost:39524*	**:**	*LISTEN*
tcp	*0*	*0*	*localhost:mysql*	**:**	*LISTEN*
tcp	*0*	*0*	*localhost:ipp*	**:**	*LISTEN*
tcp	*0*	*0*	*192.168.1.101:49041*	*lm-in-f104.google.c:www*	*CLOSE_WAIT*
tcp	*0*	*0*	*localhost:39524*	*localhost:53920*	*ESTABLISHED*
tcp	*0*	*0*	*192.168.1.101:43706*	*fk-in-f104.google.c:www*	*ESTABLISHED*
tcp	*0*	*0*	*192.168.1.101:43704*	*fk-in-f104.google.c:www*	*ESTABLISHED*
tcp	*0*	*0*	*localhost:53920*	*localhost:39524*	*ESTABLISHED*
tcp6	*0*	*0*	**:www*	**:**	*LISTEN*
udp	*0*	*0*	**:bootpc*	**:**	

The listening state sockets are included in the output only if you specify the --listening (-l) or --all (-a) option.

The possible socket states are as follows:
(taken from the "man netstat" page)

ESTABLISHED:	The socket has an established connection.
SYN_SENT:	The socket is actively attempting to establish a connection.
SYN_RECV:	A connection request has been received from the network.
FIN_WAIT1:	The socket is closed, and the connection is shutting down.
FIN_WAIT2:	Connection is closed, and the socket is waiting for a shutdown from the remote end.
TIME_WAIT:	The socket is waiting after close to handle packets still in the network.
CLOSED:	The socket is not being used.

CLOSE_WAIT:	The remote end has shut down, waiting for the socket to close.
LAST_ACK:	The remote end has shut down, and the socket is closed. Waiting for acknowledgement.
LISTEN:	The socket is listening for incoming connections. Such sockets are not included in the output unless you specify the --listening (-l) or --all (-a) option.
CLOSING:	Both sockets are shut down but we still don't have all our data sent.
UNKNOWN:	The state of the socket is unknown.

- To display all the opened network sockets (extended informations):

#netstat -aute

- -a: All
- -u: UDP
- -t: TCP
- -e: Extended

Active Internet connections (servers and established)

Proto	*Recv-Q*	*Send-Q*	*Local Address*	*Foreign Address*	*State*	*User*	*Inode*
			localhost:48898	*:*	*LISTEN*	*hplip*	*12383*
tcp	*0*	*0*	*localhost:39524*	*:*	*LISTEN*	*hplip*	*12321*
tcp	*0*	*0*	*localhost:mysql*	*:*	*LISTEN*	*mysql*	*12635*
tcp	*0*	*0*	*localhost:ipp*	*localhost:53920*	*ESTABLISHED*	*root*	*12447*
tcp	*0*	*0*	*localhost:39524*	*localhost:39524*	*ESTABLISHED*	*hplip*	*12324*
tcp	*0*	*0*	*localhost:53920*	*lm-in-f147.google.c:www*	*ESTABLISHED*	*hplip*	*12389*
tcp	*0*	*0*	*192.168.1.101:42745*	*:*	*LISTEN*	*po*	*15781*
tcp6	*0*	*0*	*:www*	*:*	*LISTEN*	*root*	*13141*
udp	*0*	*0*	*:bootpc*	*:*		*dhcp*	*14513*

- To display all the listening state sockets:

#netstat -lt

- -t: TCP
- -l: Listening state sockets

Active Internet connections (only servers)

Proto	*Recv-Q*	*Send-Q*	*Local Address*	*Foreign Address*	*State*
tcp	*0*	*0*	*localhost:48898*	*:*	*LISTEN*
tcp	*0*	*0*	*localhost:39524*	*:*	*LISTEN*
tcp	*0*	*0*	*localhost:mysql*	*:*	*LISTEN*
tcp	*0*	*0*	*localhost:ipp*	*:*	*LISTEN*
tcp6	*0*	*0*	*:www*	*:*	*LISTEN*

- To display the summary statistics for each protocol

#netstat -s

- -s: Summary statistics for each protocol.

Ip:
604 total packets received
1 with invalid addresses
0 forwarded
0 incoming packets discarded
485 incoming packets delivered
507 requests sent out
Icmp:
0 ICMP messages received
0 input ICMP message failed.
ICMP input histogram:
0 ICMP messages sent
0 ICMP messages failed
ICMP output histogram:
Tcp:
21 active connections openings
4 passive connection openings
0 failed connection attempts
0 connection resets received
3 connections established
351 segments received
388 segments send out
0 segments retransmited
0 bad segments received.
2 resets sent
Udp:
119 packets received
0 packets to unknown port received.
0 packet receive errors
119 packets sent
TcpExt:
5 TCP sockets finished time wait in fast timer
21 delayed acks sent
Quick ack mode was activated 10 times
31 packets directly queued to recvmsg prequeue.
15765 of bytes directly received from prequeue
105 packet headers predicted
17 packets header predicted and directly queued to user
36 acknowledgments not containing data received
11 predicted acknowledgments
0 TCP data loss events

10 basic examples of linux netstat command

Netstat

Netstat is a command line utility that can be used to list out all the network (socket) connections on a system. It lists out all the tcp, udp socket connections and the unix socket connections.

Apart from connected sockets it can also list listening sockets that are waiting for incoming connections. So by verifying an open port 80 you can confirm if a web server is running on the system or not. This makes it a very useful tool for network and system administrators.

In this tutorial we shall be checking out few examples of how to use netstat to find information about network connections and open ports on a system.

Here is a quick intro to netstat from the man pages

```
netstat - Print network connections, routing tables, interface statistics,
masquerade connections, and multicast memberships
```

1. List out all connections

The first and most simple command is to list out all the current connections. Simply run the netstat command with the a option.

```
$ netstat -a

Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 enlightened:domain      *:*                     LISTEN
tcp        0      0 localhost:ipp           *:*                     LISTEN
tcp        0      0 enlightened.local:54750 li240-5.members.li:http ESTABLISHED
tcp        0      0 enlightened.local:49980 del01s07-in-f14.1:https ESTABLISHED
tcp6       0      0 ip6-localhost:ipp       [::]:*                  LISTEN
udp        0      0 enlightened:domain      *:*
udp        0      0 *:bootpc                *:*
udp        0      0 enlightened.local:ntp   *:*
udp        0      0 localhost:ntp           *:*
udp        0      0 *:ntp                   *:*
udp        0      0 *:58570                 *:*
udp        0      0 *:mdns                  *:*
udp        0      0 *:49459                 *:*
udp6       0      0 fe80::216:36ff:fef8:ntp [::]:*
udp6       0      0 ip6-localhost:ntp       [::]:*
udp6       0      0 [::]:ntp                [::]:*
udp6       0      0 [::]:mdns               [::]:*
udp6       0      0 [::]:63811              [::]:*
udp6       0      0 [::]:54952              [::]:*
Active UNIX domain sockets (servers and established)
Proto RefCnt Flags       Type       State         I-Node   Path
unix  2      [ ACC ]     STREAM     LISTENING     12403    @/tmp/dbus-IDgfj3UGXX
unix  2      [ ACC ]     STREAM     LISTENING     40202
@/dbus-vfs-daemon/socket-6nUC6CCx
```

The above command shows all connections from different protocols like tcp, udp and unix sockets. However this is not quite useful. Administrators often want to pick out specific connections based on protocols or port numbers for example.

2. List only TCP or UDP connections

To list out only tcp connections use the t options.

```
$ netstat -at
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 enlightened:domain      *:*                     LISTEN
tcp        0      0 localhost:ipp           *:*                     LISTEN
tcp        0      0 enlightened.local:36310 del01s07-in-f24.1:https ESTABLISHED
tcp        0      0 enlightened.local:45038 a96-17-181-10.depl:http ESTABLISHED
tcp        0      0 enlightened.local:37892 ABTS-North-Static-:http ESTABLISHED
.....
```

Similarly to list out only udp connections use the u option.

```
$ netstat -au
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
udp        0      0 *:34660                 *:*
udp        0      0 enlightened:domain      *:*
udp        0      0 *:bootpc                *:*
udp        0      0 enlightened.local:ntp   *:*
udp        0      0 localhost:ntp           *:*
udp        0      0 *:ntp                   *:*
udp6       0      0 fe80::216:36ff:fef8:ntp [::]:*
udp6       0      0 ip6-localhost:ntp       [::]:*
udp6       0      0 [::]:ntp                [::]:*
```

The above output shows both ipv4 and ipv6 connections.

3. Disable reverse dns lookup for faster output

By default, the netstat command tries to find out the hostname of each ip address in the connection by doing a reverse dns lookup. This slows down the output. If you do not need to know the host name and just the ip address is sufficient then suppress the hostname lookup with the n option.

```
$ netstat -ant
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.1.1:53            0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN
tcp        0      0 192.168.1.2:49058       173.255.230.5:80        ESTABLISHED
tcp        0      0 192.168.1.2:33324       173.194.36.117:443      ESTABLISHED
tcp6       0      0 ::1:631                 :::*                    LISTEN
```

The above command shows ALL TCP connections with NO dns resolution. Got it ? Good.

4. List out only listening connections

Any network daemon/service keeps an open port to listen for incoming connections. These too are like socket connections and are listed out by netstat. To view only listening ports use the l options.

```
$ netstat -tnl
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.1.1:53            0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN
tcp6       0      0 ::1:631                 :::*                    LISTEN
```

Now we can see only listening tcp ports/connections. If you want to see all listening ports, remove the t option. If you want to see only listening udp ports use the u option instead of t.
Make sure to remove the 'a' option, otherwise all connections would get listed and not just the listening connections.

5. Get process name/pid and user id

When viewing the open/listening ports and connections, its often useful to know the process name/pid which has opened that port or connection. For example the Apache httpd server opens port 80. So if you want to check whether any http server is running or not, or which http server is running, apache or nginx, then track down the process name.

The process details are made available by the 'p' option.

```
~$ sudo netstat -nlpt
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
PID/Program name
tcp        0      0 127.0.1.1:53            0.0.0.0:*               LISTEN
1144/dnsmasq
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN
661/cupsd
tcp6       0      0 ::1:631                 :::*                    LISTEN
661/cupsd
```

When using the p option, netstat must be run with root privileges, otherwise it cannot detect the pids of processes running with root privileges and most services like http and ftp often run with root privileges.

Along with process name/pid its even more useful to get the username/uid owning that particular process. Use the e option along with the p option to get the username too.

```
$ sudo netstat -ltpe
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
User       Inode       PID/Program name
tcp        0      0 enlightened:domain      *:*                     LISTEN
root       11090       1144/dnsmasq
tcp        0      0 localhost:ipp           *:*                     LISTEN
root       9755        661/cupsd
tcp6       0      0 ip6-localhost:ipp       [::]:*                  LISTEN
root       9754        661/cupsd
```

The above example lists out Listening connections of Tcp type with Process information and Extended information.
The extended information contains the username and inode of the process. This is a useful command for network administrators.

Note - If you use the n option with the e option, the uid would be listed and not the username.

6. Print statistics

The netstat command can also print out network statistics like total number of packets received and transmitted by protocol type and so on.

To list out statistics of all packet types

```
$ netstat -s
Ip:
    32797 total packets received
    0 forwarded
    0 incoming packets discarded
    32795 incoming packets delivered
    29115 requests sent out
    60 outgoing packets dropped
Icmp:
    125 ICMP messages received
    0 input ICMP message failed.
    ICMP input histogram:
        destination unreachable: 125
    125 ICMP messages sent
    0 ICMP messages failed
    ICMP output histogram:
        destination unreachable: 125
... OUTPUT TRUNCATED ...
```

To print out statistics of only select protocols like TCP or UDP use the corresponding options like t and u along with the s option. Simple!

7. Display kernel routing information

The kernel routing information can be printed with the r option. It is the same output as given by the route command. We also use the n option to disable the hostname lookup.

```
$ netstat -rn
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         192.168.1.1     0.0.0.0         UG        0 0          0 eth0
192.168.1.0     0.0.0.0         255.255.255.0   U         0 0          0 eth0
```

8. Print network interfaces

The netstat command can also print out the information about the network interfaces. The i option does the task.

```
$ 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     31611      0      0 0         27503      0      0      0
BMRU
lo        65536 0      2913      0      0 0          2913      0      0      0
LRU
```

The above output contains information in a very raw format. To get a more human friendly version of the output use the e option along with i.

```
$ netstat -ie
Kernel Interface table
eth0      Link encap:Ethernet  HWaddr 00:16:36:f8:b2:64
          inet addr:192.168.1.2  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::216:36ff:fef8:b264/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:31682 errors:0 dropped:0 overruns:0 frame:0
          TX packets:27573 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:29637117 (29.6 MB)  TX bytes:4590583 (4.5 MB)
          Interrupt:18 Memory:da000000-da020000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:2921 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2921 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:305297 (305.2 KB)  TX bytes:305297 (305.2 KB)
```

The above output is similar to the output shown by the ifconfig command.

9. Get netstat output continuously

Netstat can output connection information continuously with the c option.

```
$ netstat -ct
```

The above command will output tcp connections continuously.

10. Display multicast group information

The g option will display the multicast group information for IPv4 and IPv6 protocols.

```
$ netstat -g
IPv6/IPv4 Group Memberships
Interface       RefCnt Group
--------------- ------ ---------------------
lo              1      all-systems.mcast.net
eth0            1      224.0.0.251
eth0            1      all-systems.mcast.net
lo              1      ip6-allnodes
lo              1      ff01::1
eth0            1      ff02::fb
eth0            1      ff02::1:fff8:b264
eth0            1      ip6-allnodes
eth0            1      ff01::1
wlan0           1      ip6-allnodes
wlan0           1      ff01::1
```

More examples of netstat command

Okay, we covered the basic examples of netstat command above. Now its time to do some geek stuff with style.

Print active connections

Active socket connections are in "ESTABLISHED" state. So to get all current active connections use netstat with grep as follows

```
$ netstat -atnp | grep ESTA
(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      0 192.168.1.2:49156       173.255.230.5:80        ESTABLISHED
1691/chrome
tcp        0      0 192.168.1.2:33324       173.194.36.117:443      ESTABLISHED
1691/chrome
```

To watch a continous list of active connections, use the watch command along with netstat and grep

```
$ watch -d -n0 "netstat -atnp | grep ESTA"
```

Check if a service is running

If you want to check if a server like http,smtp or ntp is running or not, use grep again.

```
$ sudo netstat -aple | grep ntp
udp        0      0 enlightened.local:ntp   *:*
root       17430      1789/ntpd
udp        0      0 localhost:ntp           *:*
root       17429      1789/ntpd
udp        0      0 *:ntp                   *:*
root       17422      1789/ntpd
udp6       0      0 fe80::216:36ff:fef8:ntp [::]:*
root       17432      1789/ntpd
udp6       0      0 ip6-localhost:ntp       [::]:*
root       17431      1789/ntpd
udp6       0      0 [::]:ntp                [::]:*
root       17423      1789/ntpd
unix  2      [ ]         DGRAM                    17418    1789/ntpd
```

So we found that ntp server is running. Grep for http or smtp or whatever you are looking for.

Well, that was most of what netstat is used for. If you are looking for more advanced information or want to dig deeper, read up the netstat manual (man netstat).

And do leave your feedback and suggestions in the comments box below.

Netstat Tutorial

Netstat is a command-line utility to view of active ports on your machine and their status. This helps user to understand which ports are open, closed, or listening for incoming connections. The information provided by netstat conveys an accurate assumption of how vulnerable PC might be to attacks on various ports.

Common attacks may include port 21 (ftp) and port 23 (telnet). A hacker can connect to these ports to obtain view of the directory structure, download and upload files, and, if the password is compromised, connect to the host with complete control.

Netstat examines both basic TCP and UDP connections. Netstat has ability to filter between TCP and UDP. Netstat can select a particular protocol, including IP, ICMP, TCPv6 and UDPv6, etc.

Netstat displays protocol statistics and current TCP/IP network connections using the following command switches:

```
NETSTAT [-a] [-e] [-n] [-s] [-p proto] [-r] [interval]

  -a            Displays all connections and listening ports.
  -e            Displays Ethernet statistics. This may be combined with the -s
                option.
  -n            Displays addresses and port numbers in numerical form.
  -o            Displays the owning process ID associated with each connection.
  -p proto      Shows connections for the protocol specified by proto; proto
                may be any of: TCP, UDP, TCPv6, or UDPv6.  If used with the -s
                option to display per-protocol statistics, proto may be any of:
                IP, IPv6, ICMP, ICMPv6, TCP, TCPv6, UDP, or UDPv6.
  -r            Displays the routing table.
  -s            Displays per-protocol statistics.  By default, statistics are
                shown for IP, IPv6, ICMP, ICMPv6, TCP, TCPv6, UDP, and UDPv6;
                the -p option may be used to specify a subset of the default.
  interval      Redisplays selected statistics, pausing interval seconds
                between each display.  Press CTRL+C to stop redisplaying
                statistics.  If omitted, netstat will print the current
                configuration information once.
```

Netstat Switch Summary

A	The first switch, a, is used as the syntax below: `netstat -a` This command lists all active connections including listening ports.
E	The e switch lists the statistics of the internet connection, including how many packets were sent, recieved or how many bytes were recieved.
N	The n switch lists all connections and remote computers in numerical form, this being in IP form. For example if you would like to view the server IP in numerical form, use the n switch to transform the web address of to the

	corresponding IP.
O	This switch lists active connections, combined with its PID (Process Identification Number).
P	The p switch gives the user the ability to filter through protocols including TCP, UDP, IP, ICMP, TCPv6, UDPv6, IPv6 and ICMPv6.
R	The r switch lists information for your ethernet card, netmask, gateway, network destination, etc. For example, `netstat -nr` analyzes the routing table.
S	The s switch prints to the screen statistics for each protocol, including those in the p switch. This switch can be combined with the p switch in order to display specific statistics for each specified protocol: `netstat -ps TCP` The above command lists the statistics for the TCP protocol, plus its active connections. This query can be narrowed down to an even more specific or broader range of connections, as descibed below.
Interval	The interval switch allows you to give your computer a specific time, or interval, between the netstat probings of active connections. For example, `netstat -an 20` lists all connections (switch a) in numeric form (switch n) and spaces each netstat command 20 seconds (interval (20)). Command returns a list of connections every 20 seconds.

Using Multiple Switches

The user can specify multiple switches on the command line. To combine multiple switched either of the following syntaxes will work and yield the same result:

```
netstat -an
netstat -a -n
```

There is no limit on how many switches you use, as long as the switches are compatible with each other. For example, using the n switch with the r switch yields results of a standard r switch.

Netstat Output

Netstat with no arguments gives a generic look at what ports are open on the system. User can identify which protocol is in use along with the ports, local PC name, TCP/IP network connections, foreign address, local address and the status of each connection.

The characters under the title "Proto" indicate the protocol type, in this case the only connections present include TCP which means that you and the remote host are communicating via TCP.

The local address specifies the name of your computer on the network along with the port number that you are using to recieve connections, which is randomly generated.

The foreign address lists the remote host's name and the port they are using to initiate the connection.

The state of the connection indicates exactly what it says, the state of the connection between a remote system and yours. Possible states of a connection are as follows:

```
ESTABLISHED - Both hosts are connected.
CLOSING     - The remote host has agreed to close its connection.
LISTENING   - Your computer is waiting to handle an incoming connection.
SYN_RCVD    - A remote host has asked for you to start a connection.
SYN_SENT    - Your computer has accepted to start a connection.
LAST_ACK    - Your computer needs to obliterate (i.e. erase from memory)
              the packets before closing the connection.
TIMED_WAIT  - See above.
CLOSE_WAIT  - The remote host is closing its connection with your computer.
FIN_WAIT 1  - A client is closing its connection.
FIN_WAIT 2  - Both hosts have agreed to close the connection.
```

Other network utilities and resources

See also: Foundstone network security utilities. For example, **fport** utility identifies open ports and running applications, associated with those ports.

A set of animated tutorials is available at `www.grayhatplayground.com`, a website developed by prof. Rick Leinecker and his students at Rockingham Community College, NC.

20 Netstat Commands for Linux Network Management

by Ravi Saive | Published: August 8, 2012

netstat (**network statistics**) is a command line tool for monitoring network connections both incoming and outgoing as well as viewing routing tables, interface statistics etc. **netstat** is available on all Unix-like Operating Systems and also available on **Windows OS** as well. It is very useful in terms of network troubleshooting and performance measurement. **netstat** is one of the most basic network service debugging tools, telling you what ports are open and whether any programs are listening on ports.

This tool is very important and much useful for Linux network administrators as well as system administrators to monitor and troubleshoot their network related problems and determine network traffic performance. This article shows usages of **netstat** command with their examples which may be useful in daily operation.

You might also be interested in following article

1. 35 Practical Examples of Linux Find Command

1. Listing all the LISTENING Ports of TCP and UDP connections

Listing all ports (both TCP and UDP) using **netstat** -a option.

```
# netstat -a | more
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address               Foreign Address
State
tcp        0      0 *:sunrpc                    *:*
LISTEN
tcp        0     52 192.168.0.2:ssh             192.168.0.1:egs
ESTABLISHED
tcp        1      0 192.168.0.2:59292           www.gov.com:http
CLOSE_WAIT
tcp        0      0 localhost:smtp              *:*
LISTEN
tcp        0      0 *:59482                     *:*
LISTEN
udp        0      0 *:35036                     *:*
udp        0      0 *:npmp-local                *:*
Active UNIX domain sockets (servers and established)
Proto RefCnt Flags       Type       State         I-Node Path
unix  2      [ ACC ]     STREAM     LISTENING     16972  /tmp/orbit-root/linc-
76b-0-6fa08790553d6
unix  2      [ ACC ]     STREAM     LISTENING     17149  /tmp/orbit-root/linc-
794-0-7058d584166d2
unix  2      [ ACC ]     STREAM     LISTENING     17161  /tmp/orbit-root/linc-
792-0-546fe905321cc
unix  2      [ ACC ]     STREAM     LISTENING     15938  /tmp/orbit-root/linc-
74b-0-415135cb6aeab
```

2. Listing TCP Ports connections

Listing only **TCP** (**Transmission Control Protocol**) port connections using **netstat -at**.

```
# netstat -at
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address               Foreign Address
State
tcp        0      0 *:ssh                       *:*
LISTEN
tcp        0      0 localhost:ipp               *:*
LISTEN
tcp        0      0 localhost:smtp              *:*
LISTEN
tcp        0     52 192.168.0.2:ssh             192.168.0.1:egs
ESTABLISHED
tcp        1      0 192.168.0.2:59292           www.gov.com:http
CLOSE_WAIT
```

3. Listing UDP Ports connections

Listing only **UDP** (**User Datagram Protocol**) port connections using **netstat -au**.

```
# netstat -au
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address               Foreign Address
State
udp        0      0 *:35036                     *:*
udp        0      0 *:npmp-local                *:*
udp        0      0 *:mdns                      *:*
```

4. Listing all LISTENING Connections

Listing all active listening ports connections with **netstat -l**.

```
# netstat -l
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address               Foreign Address
State
tcp        0      0 *:sunrpc                    *:*
LISTEN
tcp        0      0 *:58642                     *:*
LISTEN
tcp        0      0 *:ssh                       *:*
LISTEN
udp        0      0 *:35036                     *:*
udp        0      0 *:npmp-local                *:*
Active UNIX domain sockets (only servers)
Proto RefCnt Flags       Type       State         I-Node Path
unix  2      [ ACC ]     STREAM     LISTENING     16972  /tmp/orbit-root/linc-
76b-0-6fa08790553d6
unix  2      [ ACC ]     STREAM     LISTENING     17149  /tmp/orbit-root/linc-
794-0-7058d584166d2
unix  2      [ ACC ]     STREAM     LISTENING     17161  /tmp/orbit-root/linc-
792-0-546fe905321cc
unix  2      [ ACC ]     STREAM     LISTENING     15938  /tmp/orbit-root/linc-
74b-0-415135cb6aeab
```

5. Listing all TCP Listening Ports

Listing all active listening TCP ports by using option **netstat -lt**.

```
# netstat -lt
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address               Foreign Address
State
tcp        0      0 *:dctp                      *:*
LISTEN
tcp        0      0 *:mysql                     *:*
LISTEN
tcp        0      0 *:sunrpc                    *:*
LISTEN
tcp        0      0 *:munin                     *:*
LISTEN
tcp        0      0 *:ftp                       *:*
LISTEN
tcp        0      0 localhost.localdomain:ipp   *:*
LISTEN
tcp        0      0 localhost.localdomain:smtp  *:*
LISTEN
tcp        0      0 *:http                      *:*
LISTEN
tcp        0      0 *:ssh                       *:*
LISTEN
tcp        0      0 *:https                     *:*
LISTEN
```

6. Listing all UDP Listening Ports

Listing all active listening UDP ports by using option **netstat -lu**.

```
# netstat -lu
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address               Foreign Address
State
udp        0      0 *:39578                     *:*
udp        0      0 *:meregister                *:*
udp        0      0 *:vpps-qua                  *:*
udp        0      0 *:openvpn                   *:*
udp        0      0 *:mdns                      *:*
udp        0      0 *:sunrpc                    *:*
udp        0      0 *:ipp                       *:*
udp        0      0 *:60222                     *:*
udp        0      0 *:mdns                      *:*
```

7. Listing all UNIX Listening Ports

Listing all active UNIX listening 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     4171
@ISCSIADM_ABSTRACT_NAMESPACE
unix  2      [ ACC ]     STREAM     LISTENING     5767   /var/run/cups/cups.sock
unix  2      [ ACC ]     STREAM     LISTENING     7082   @/tmp/fam-root-
unix  2      [ ACC ]     STREAM     LISTENING     6157   /dev/gpmctl
unix  2      [ ACC ]     STREAM     LISTENING     6215   @/var/run/hald/dbus-
IcefTIUkHm
unix  2      [ ACC ]     STREAM     LISTENING     6038   /tmp/.font-unix/fs7100
unix  2      [ ACC ]     STREAM     LISTENING     6175
/var/run/avahi-daemon/socket
unix  2      [ ACC ]     STREAM     LISTENING     4157
@ISCSID_UIP_ABSTRACT_NAMESPACE
```

```
unix  2      [ ACC ]     STREAM     LISTENING     60835836
/var/lib/mysql/mysql.sock
unix  2      [ ACC ]     STREAM     LISTENING     4645   /var/run/audispd_events
unix  2      [ ACC ]     STREAM     LISTENING     5136
/var/run/dbus/system_bus_socket
unix  2      [ ACC ]     STREAM     LISTENING     6216   @/var/run/hald/dbus-
wsUBI30V2I
unix  2      [ ACC ]     STREAM     LISTENING     5517   /var/run/acpid.socket
unix  2      [ ACC ]     STREAM     LISTENING     5531   /var/run/pcscd.comm
```

8. Showing Statistics by Protocol

Displays statistics by protocol. By default, statistics are shown for the TCP, UDP, ICMP, and IP protocols. The -s parameter can be used to specify a set of protocols.

```
# netstat -s
Ip:
2461 total packets received
0 forwarded
0 incoming packets discarded
2431 incoming packets delivered
2049 requests sent out
Icmp:
0 ICMP messages received
0 input ICMP message failed.
ICMP input histogram:
1 ICMP messages sent
0 ICMP messages failed
ICMP output histogram:
destination unreachable: 1
Tcp:
159 active connections openings
1 passive connection openings
4 failed connection attempts
0 connection resets received
1 connections established
2191 segments received
1745 segments send out
24 segments retransmited
0 bad segments received.
4 resets sent
Udp:
243 packets received
1 packets to unknown port received.
0 packet receive errors
281 packets sent
```

9. Showing Statistics by TCP Protocol

Showing statistics of only TCP protocol by using option **netstat -st**.

```
# netstat -st
Tcp:
2805201 active connections openings
1597466 passive connection openings
1522484 failed connection attempts
37806 connection resets received
1 connections established
57718706 segments received
64280042 segments send out
3135688 segments retransmited
74 bad segments received.
```

```
17580 resets sent
```

10. Showing Statistics by UDP Protocol

```
# netstat -su
Udp:
1774823 packets received
901848 packets to unknown port received.
0 packet receive errors
2968722 packets sent
```

11. Displaying Service name with PID

Displaying service name with their PID number, using option **netstat -tp** will display “PID/Program Name”.

```
# netstat -tp
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address               Foreign Address
State       PID/Program name
tcp        0      0 192.168.0.2:ssh            192.168.0.1:egs
ESTABLISHED 2179/sshd
tcp        1      0 192.168.0.2:59292          www.gov.com:http
CLOSE_WAIT  1939/clock-applet
```

12. Displaying Promiscuous Mode

Displaying Promiscuous mode with -ac switch, netstat print the selected information or refresh screen every five second. Default screen refresh in every second.

```
# netstat -ac 5 | grep tcp
tcp        0      0 *:sunrpc                   *:*
LISTEN
tcp        0      0 *:58642                    *:*
LISTEN
tcp        0      0 *:ssh                      *:*
LISTEN
tcp        0      0 localhost:ipp              *:*
LISTEN
tcp        0      0 localhost:smtp             *:*
LISTEN
tcp        1      0 192.168.0.2:59447          www.gov.com:http
CLOSE_WAIT
tcp        0     52 192.168.0.2:ssh            192.168.0.1:egs
ESTABLISHED
tcp        0      0 *:sunrpc                   *:*
LISTEN
tcp        0      0 *:ssh                      *:*
LISTEN
tcp        0      0 localhost:ipp              *:*
LISTEN
tcp        0      0 localhost:smtp             *:*
LISTEN
tcp        0      0 *:59482                    *:*
LISTEN
```

13. Displaying Kernel IP routing

Display Kernel IP routing table with netstat and route command.

```
# netstat -r
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
192.168.0.0     *               255.255.255.0   U         0 0          0 eth0
link-local      *               255.255.0.0     U         0 0          0 eth0
default         192.168.0.1     0.0.0.0         UG        0 0          0 eth0
```

14. Showing Network Interface Transactions

Showing network interface packet transactions including both transferring and receiving packets with MTU size.

```
# 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     4459      0      0      0     4057      0      0      0
BMRU
lo        16436   0        8      0      0      0        8      0      0      0
LRU
```

15. Showing Kernel Interface Table

Showing Kernel interface table, similar to **ifconfig** command.

```
# netstat -ie
Kernel Interface table
eth0      Link encap:Ethernet  HWaddr 00:0C:29:B4:DA:21
inet addr:192.168.0.2  Bcast:192.168.0.255  Mask:255.255.255.0
inet6 addr: fe80::20c:29ff:feb4:da21/64 Scope:Link
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:4486 errors:0 dropped:0 overruns:0 frame:0
TX packets:4077 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:2720253 (2.5 MiB)  TX bytes:1161745 (1.1 MiB)
Interrupt:18 Base address:0x2000
lo        Link encap:Local Loopback
inet addr:127.0.0.1  Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING  MTU:16436  Metric:1
RX packets:8 errors:0 dropped:0 overruns:0 frame:0
TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:480 (480.0 b)  TX bytes:480 (480.0 b)
```

16. Displaying IPv4 and IPv6 Information

Displays multicast group membership information for both IPv4 and IPv6.

```
# netstat -g
IPv6/IPv4 Group Memberships
Interface       RefCnt Group
--------------- ------ ---------------------
lo              1      all-systems.mcast.net
eth0            1      224.0.0.251
eth0            1      all-systems.mcast.net
lo              1      ff02::1
eth0            1      ff02::202
eth0            1      ff02::1:ffb4:da21
eth0            1      ff02::1
```

17. Print Netstat Information Continuously

To get netstat information every few second, then use the following command, it will print netstat information continuously, say every few seconds.

```
# netstat -c
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address               Foreign Address
State
tcp        0      0 tecmint.com:http   sg2nlhg007.shr.prod.s:36944 TIME_WAIT
tcp        0      0 tecmint.com:http   sg2nlhg010.shr.prod.s:42110 TIME_WAIT
tcp        0    132 tecmint.com:ssh    115.113.134.3.static-:64662 ESTABLISHED
tcp        0      0 tecmint.com:http   crawl-66-249-71-240.g:41166 TIME_WAIT
tcp        0      0 localhost.localdomain:54823 localhost.localdomain:smtp
TIME_WAIT
tcp        0      0 localhost.localdomain:54822 localhost.localdomain:smtp
TIME_WAIT
tcp        0      0 tecmint.com:http   sg2nlhg010.shr.prod.s:42091 TIME_WAIT
tcp        0      0 tecmint.com:http   sg2nlhg007.shr.prod.s:36998 TIME_WAIT
```

18. Finding non supportive Address

Finding un-configured address families with some useful information.

```
# netstat --verbose
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.
```

19. Finding Listening Programs

Find out how many listening programs running on a port.

```
# netstat -ap | grep http
tcp        0      0 *:http                      *:*
LISTEN      9056/httpd
tcp        0      0 *:https                     *:*
LISTEN      9056/httpd
tcp        0      0 tecmint.com:http   sg2nlhg008.shr.prod.s:35248 TIME_WAIT   -
tcp        0      0 tecmint.com:http   sg2nlhg007.shr.prod.s:57783 TIME_WAIT   -
tcp        0      0 tecmint.com:http   sg2nlhg007.shr.prod.s:57769 TIME_WAIT   -
tcp        0      0 tecmint.com:http   sg2nlhg008.shr.prod.s:35270 TIME_WAIT   -
tcp        0      0 tecmint.com:http   sg2nlhg009.shr.prod.s:41637 TIME_WAIT   -
tcp        0      0 tecmint.com:http   sg2nlhg009.shr.prod.s:41614 TIME_WAIT   -
unix  2      [ ]         STREAM     CONNECTED     88586726 10394/httpd
```

20. Displaying RAW Network Statistics

```
# netstat --statistics --raw
Ip:
62175683 total packets received
52970 with invalid addresses
0 forwarded
Icmp:
875519 ICMP messages received
destination unreachable: 901671
echo request: 8
echo replies: 16253
IcmpMsg:
InType0: 83
IpExt:
```

```
InMcastPkts: 117
```

That’s it, If you are looking for more information and options about netstat command, refer netstat manual docs or use **man netstat** command to know all the information. If we’ve missed anything in the list, please inform us using our comment section below. So, we could keep updating this list based on your comments.