Basics of Network configuration files and Networking Commands in Linux

Yadhukrishnan M

January 17, 2018

1 Basic Networking Commands

1.1 ip

The new and recommended alternative (to the popular ifconfig command)for examining a network configuration on Debian Linux is ip command. It is used to show and manipulate routing, devices, policy routing and tunnels.

ip syntax:

```
ip [ OPTIONS ] OBJECT { COMMAND | help }
```

where OBJECT may be:

```
{ link | addr | addrlabel | route | rule | neigh | ntable | tunnel |
tuntap maddr | mroute | mrule | monitor | xfrm | netns | l2tp | tcp_metrics }
```

and OPTIONS may be:

```
{ -V[ersion] | -s[tatistics] | -r[esolve] | -f[amily]
{ inet | inet6 | ipx | dnet | link } | -o[neline] }
```

1.2 ping

Short for Packet InterNet Groper, ping is a utility used to verify whether or not a network data packet is capable of being distributed to an address without errors. The ping utility is commonly used to check for network errors.

```
ping [ OPTIONS ] destination
```

Sample output:

```
$ ping google.com
PING google.com (172.217.160.142) 56(84) bytes of data.
64 bytes from maa03s29-in-f14.1e100.net (172.217.160.142): icmp_seq=1 ttl=52 tim
64 bytes from maa03s29-in-f14.1e100.net (172.217.160.142): icmp_seq=2 ttl=52 tim
^C
--- google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 14.031/14.227/14.424/0.229 ms
```

1.3 traceroute

The traceroute command will attempt to provide a list of all the routers your connections cross when reaching out to a remote system. The output also provides some information on how long each segment of the path takes, thus giving you some notion of the quality of a connection.

```
traceroute [ OPTIONS ] host
```

Sample output:

```
$ traceroute google.com

traceroute to google.com (172.217.160.142), 30 hops max, 60 byte packets
 1  gateway (192.168.0.1)  1.358 ms  3.470 ms  3.463 ms
 2  172.31.34.3 (172.31.34.3)  6.680 ms  6.686 ms  6.674 ms
 3  172.31.207.142 (172.31.207.142)  15.920 ms  16.742 ms  17.182 ms
 4  172.31.90.126 (172.31.90.126)  22.002 ms  22.016 ms  22.010 ms
 5  172.31.10.66 (172.31.10.66)  20.404 ms  21.110 ms  21.124 ms
 6  10.93.12.5 (10.93.12.5)  18.289 ms  13.914 ms  14.029 ms
 7  10.93.12.6 (10.93.12.6)  18.357 ms  18.395 ms  18.382 ms
 8  172.31.110.123 (172.31.110.123)  15.817 ms  16.571 ms  19.205 ms
 9  172.31.10.78 (172.31.10.78)  18.277 ms  18.286 ms  18.296 ms
10  112.133.203.182 (112.133.203.182)  19.709 ms  20.504 ms  21.147 ms
11  72.14.233.204 (72.14.233.204)  21.150 ms  21.144 ms  21.497 ms
12  209.85.241.197 (209.85.241.197)  17.974 ms  18.061 ms  18.367 ms
13  maa03s29-in-f14.1e100.net (172.217.160.142)  14.142 ms  14.746 ms  15.182 ms
```

1.4 netstat

The netstat command is used to print network connections, routing tables, interface statistics, masquerade connections, and multicast memberships.

netstat ("network statistics") is a command-line tool that displays network connections (both incoming and outgoing), routing tables, and a number of network interface (network interface controller or software-defined network interface) and network protocol statistics. It is available on Unix-like operating systems including OS X, Linux, Solaris, and BSD, and on Windows NT-based operating systems including Windows XP, Windows Vista, Windows 7 and Windows 8.

It is used for finding problems in the network and to determine the amount of traffic on the network as a performance measurement.

```
netstat [address_family_options] [--tcp|-t] [--udp|-u] [--raw|-w]
        [--listening|-l] [--all|-a] [--numeric|-n] [--numeric-hosts]
        [--numeric-ports] [--numeric-users] [--symbolic|-N]
        [--extend|-e[--extend|-e]] [--timers|-o] [--program|-p]
        [--verbose|-v] [--continuous|-c]
```

Sample output:

```
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 debian:36292           klecker2.snt.utwen:http TIME_WAIT
tcp        0      0 debian:44176           ws203-233-252-122.:http ESTABLISHED
tcp        0      0 debian:44180           ws203-233-252-122.:http ESTABLISHED
tcp        0      0 debian:36140           ocsp.comodoca.com:http  TIME_WAIT
tcp        0      0 debian:48060           maa03s21-in-f10.1:https ESTABLISHED
tcp        0      0 debian:58094           ec2-46-51-218-82.:https ESTABLISHED
tcp        0      0 debian:59890           maa03s29-in-f3.1e:https ESTABLISHED
tcp        0      0 debian:44178           ws203-233-252-122.:http ESTABLISHED
tcp        0      0 debian:58116           ec2-46-51-218-82.:https ESTABLISHED
tcp        0      0 debian:37800           104.27.6.18:https       ESTABLISHED
tcp        0      0 debian:40972           ec2-52-77-181-198:https ESTABLISHED
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags       Type       State         I-Node   Path
unix  8      [ ]         DGRAM                    11556    /run/systemd/journal/
unix  18     [ ]         DGRAM                    11573    /run/systemd/journal/
unix  2      [ ]         DGRAM                    11578    /run/systemd/journal/
unix  2      [ ]         DGRAM                    91235    /run/wpa_supplicant/w
...
```

1.5 nslookup

The nslookup command is used to query Internet name servers interactively for information. nslookup, which stands for "name server lookup", is a useful tool for finding out information about a named domain. By default, nslookup will translate a domain name to an IP address (or vice versa).

```
$ nslookup google.com
Server: 8.8.8.8
Address: 8.8.8.8#53

Non-authoritative answer:
Name: google.com
Address: 172.217.160.142
```

1.6 whois

The whois protocol returns information about registered domain names, including the name servers they are configured to work with. While most of the information concerns the registration of the domain, it can be helpful to see that the name servers are returned correctly.

```
whois [domain-name]
```

Sample output:

```
$ whois google.com
  Domain Name: GOOGLE.COM
  Registry Domain ID: 2138514_DOMAIN_COM-VRSN
  Registrar WHOIS Server: whois.markmonitor.com
  Registrar URL: http://www.markmonitor.com
  Updated Date: 2011-07-20T16:55:31Z
  Creation Date: 1997-09-15T04:00:00Z
  Registry Expiry Date: 2020-09-14T04:00:00Z
  Registrar: MarkMonitor Inc.
  Registrar IANA ID: 292
  Registrar Abuse Contact Email: abusecomplaints@markmonitor.com
  Registrar Abuse Contact Phone: +1.2083895740
  Domain Status: clientDeleteProhibited https://icann.org/epp#clientDeleteProhib
  Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferPr
  Domain Status: clientUpdateProhibited https://icann.org/epp#clientUpdateProhib
  Domain Status: serverDeleteProhibited https://icann.org/epp#serverDeleteProhib
```

```
Domain Status: serverTransferProhibited https://icann.org/epp#serverTransferPr
Domain Status: serverUpdateProhibited https://icann.org/epp#serverUpdateProhib
Name Server: NS1.GOOGLE.COM
Name Server: NS2.GOOGLE.COM
Name Server: NS3.GOOGLE.COM
Name Server: NS4.GOOGLE.COM
DNSSEC: unsigned
```

1.7 arp

Arp is used to translate IP addresses into Ethernet addresses. Root can add and delete arp entries. Deleting them can be useful if an arp entry is malformed or just wrong. Arp entries explicitly added by root are permanent they can also be by proxy. The arp table is stored in the kernel and manipulated dynamically. Arp entries are cached and will time out and are deleted normally in 20 minutes.

```
arp -a : prints arp table
arp s <ip_address> <mac_address> [pub] to add an entry in the table
arp a d to delete all the entries in the ARP table
```

1.8 host

host is used to map names to IP addresses. It is a very quick and simple utility without a lot of functions.

```
$ host cet.ac.in
cet.ac.in has address 103.10.168.12
cet.ac.in mail is handled by 10 ASPMX3.GOOGLEMAIL.COM.
cet.ac.in mail is handled by 1 ASPMX.L.GOOGLE.COM.
cet.ac.in mail is handled by 5 ALT2.ASPMX.L.GOOGLE.COM.
cet.ac.in mail is handled by 5 ALT1.ASPMX.L.GOOGLE.COM.
cet.ac.in mail is handled by 10 ASPMX2.GOOGLEMAIL.COM.
```

1.9 dig

The meanest dog in the pound, the domain information groper, dig for short, is the go-to program for finding DNS information. dig can grab just about

anything from a DNS server including reverse lookups, A, CNAME, MX, SP, and TXT records. dig has many command line options and if you're not familiar with it you should read through it's extensive man page.

1.10 finger

finger will retrieve information about the specified user. You give finger a username or an email address and it will try to contact the necessary server and retrieve the username, office, telephone number, and other pieces of information.

1.11 telnet

telnet allows you to log in to a computer, just as if you were sitting at the terminal. Once your username and password are verified, you are given a shell prompt. From here, you can do anything requiring a text console. Compose email, read newsgroups, move files around, and so on. If you are running X and you telnet to another machine, you can run X programs on the remote computer and display them on yours.

2 Important files

- /etc/hosts - names to ip addresses
- /etc/networks - network names to ip addresses
- /etc/protocols protocol names to protocol numbers
- /etc/services - tcp/udp service names to port numbers