Debian configure the network manually

You can use [ip command](https://www.cyberciti.biz/faq/linux-ip-command-examples-usage-syntax/?utm_source=Linux_Unix_Command&utm_medium=faq&utm_campaign=nixcmd) or ifconfig command which is deprecated to configure IP address and other information on Debian Linux. Hence, I recommend using the ip command.

Task: Display the Current Network Configuration

Type the following command:  
$ ip address show  
Sample outputs:

1: lo: mtu 16436 qdisc noqueue

link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00

inet 127.0.0.1/8 scope host lo

inet6 ::1/128 scope host

valid\_lft forever preferred\_lft forever

2: eth0: mtu 1500 qdisc pfifo\_fast qlen 100

link/ether 00:19:d1:2a:ba:a8 brd ff:ff:ff:ff:ff:ff

inet 192.168.2.1/24 brd 192.168.2.255 scope global eth0

inet6 fe80::219:d1ff:fe2a:baa8/64 scope link

valid\_lft forever preferred\_lft forever

3: ra0: mtu 1500 qdisc pfifo\_fast qlen 1000

link/ether 00:17:9a:0a:f6:44 brd ff:ff:ff:ff:ff:ff

inet 192.168.1.106/24 brd 192.168.1.255 scope global ra0

inet6 fe80::217:9aff:fe0a:f644/64 scope link

valid\_lft forever preferred\_lft forever

4: ppp0: mtu 1496 qdisc pfifo\_fast qlen 3

link/ppp

inet 10.1.3.103 peer 10.0.31.18/32 scope global ppp0

You can also use **ifconfig -a** command, enter:  
$ ifconfig -a  
Sample outputs:

eth0 Link encap:Ethernet HWaddr 00:19:D1:2A:BA:A8

inet addr:192.168.2.1 Bcast:192.168.2.255 Mask:255.255.255.0

inet6 addr: fe80::219:d1ff:fe2a:baa8/64 Scope:Link

UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1

RX packets:15819 errors:0 dropped:0 overruns:0 frame:0

TX packets:27876 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:100

RX bytes:1695948 (1.6 MB) TX bytes:40399983 (38.5 MB)

Base address:0x1000 Memory:93180000-931a0000

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:11943 errors:0 dropped:0 overruns:0 frame:0

TX packets:11943 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:0

RX bytes:7024449 (6.6 MB) TX bytes:7024449 (6.6 MB)

ppp0 Link encap:Point-to-Point Protocol

inet addr:10.1.3.103 P-t-P:10.0.31.18 Mask:255.255.255.255

UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1496 Metric:1

RX packets:34922 errors:0 dropped:0 overruns:0 frame:0

TX packets:15764 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:3

RX bytes:50535608 (48.1 MB) TX bytes:1256881 (1.1 MB)

ra0 Link encap:Ethernet HWaddr 00:17:9A:0A:F6:44

inet addr:192.168.1.106 Bcast:192.168.1.255 Mask:255.255.255.0

inet6 addr: fe80::217:9aff:fe0a:f644/64 Scope:Link

UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1

RX packets:73809 errors:0 dropped:0 overruns:0 frame:0

TX packets:31332 errors:1 dropped:1 overruns:0 carrier:0

collisions:27 txqueuelen:1000

RX bytes:61373519 (58.5 MB) TX bytes:5007190 (4.7 MB)

Interrupt:20

The information is grouped by network interfaces. Every interface entry starts with a digit, called the interface index, with the interface name displayed after the interface index. In the above example, there are four interfaces:

* **lo** : Loopback interface, used to access local services such as proxy or webserver http://127.0.0.1/
* **eth0** : The first Ethernet interface connected to network switch or router
* **ra0** : The first wireless interface
* **ppp0** :The first point-to-point interface, used to connect via VPN or dial up service

Task: Show Network Device / Interface Statistics

Type the following [ip command](https://www.cyberciti.biz/faq/linux-ip-command-examples-usage-syntax/?utm_source=Linux_Unix_Command&utm_medium=faq&utm_campaign=nixcmd)$ ip -s link show interface-name  
$ ip -s link show eth0  
$ ip -s link show ppp0  
Sample outputs:

4: ppp0: mtu 1496 qdisc pfifo\_fast qlen 3

link/ppp

RX: bytes packets errors dropped overrun mcast

50537336 34946 0 0 0 0

TX: bytes packets errors dropped carrier collsns

1257745 15776 0 0 0 0

Change the Current Network Configuration On Debian Linux

You must login as the root to change current network settings.

Task: Assign an IP Address to a Device Interface

In the following example, the command assigns the IP address 192.168.1.10 to the device eth0. The network mask is 24 (255.255.255.0) bits long. The brd + option sets the broadcast address automatically as determined by the network mask:  
# ip address add 192.168.1.100/24 brd + dev eth0  
You can also use ifconfig command, enter:  
# ifconfig eth0 192.168.1.100 netmask 255.255.255.0 up

Task: Remove / Delete / Deactivate IP address from a device interface

To remove IP / delete device, enter:  
# ip address del 192.168.1.100 dev eth0  
OR  
# ifconfig eth0 down

Save Network Settings to a Configuration File

To change the current network configuration setting you’ll need to edit**/etc/network/interfaces** file using a text editor such as vi. This is the only way to save device setting to a configuration file so that system can remember changes after a reboot.

Task: Configure a Device Statically

Open /etc/network/interfaces file as the root user:  
# vi /etc/network/interfaces  
Let us assign static public routable (or private) IP address to eth0 interface, enter:  
auto eth0  
iface eth0 inet static  
address 192.168.2.1  
netmask 255.255.255.0  
gateway 192.1.2.254  
  
Save and close the file. Where,

* **auto eth0** : Identify the physical interfaces such as eth0, eth1 and so on
* **iface eth0 inet static** : This method used to define ethernet interfaces with statically allocated IPv4 addresses
* **address 192.168.2.1** : Static IP address
* **netmask 255.255.255.0** : Static netmask
* **gateway 192.168.1.254** : Static gatway/router IP address

Task: Configure a Device Dynamically with DHCP

Open the /etc/network/interfaces file as the root user:  
# vi /etc/network/interfaces  
Let us configure eth0 using DHCP. When the device is configured by using DHCP, you don’t need to set any options for the network address configuration in the file.  
auto eth0  
iface eth0 inet dhcp  
Save and close the file.  
Where,

* **auto eth0** : Identify the physical interfaces such as eth0, eth1 and so on
* **iface eth0 inet dhcp** : This method used to define ethernet interfaces with DHCP server allocated IPv4 addresses

Start and Stop Configured Interfaces

To apply changes to a configuration file, you need to stop and restart the corresponding interface  
# /etc/init.d/networking stop  
# /etc/init.d/networking start  
# /etc/init.d/networking restart  
You can also use following command to bring down or up the eth0. **Disables the device eth0**, enter:  
# ifdown eth0  
**Enables eth0** again, enter:  
# ifup eth0

Defining the (DNS) Nameservers

Edit [/etc/resolv.conf](https://bash.cyberciti.biz/guide/etc/resolv.conf?utm_source=Linux_Unix_File&utm_medium=faq&utm_campaign=nixfile), enter:  
# vi /etc/resolv.conf  
Update / add as follows:

### The IP addresses of nameservers ##

nameserver 8.8.8.8

nameserver 192.168.1.254

Save and close the file. Use [dig command](https://www.cyberciti.biz/faq/linux-unix-dig-command-examples-usage-syntax/?utm_source=Linux_Unix_Command&utm_medium=faq&utm_campaign=nixcmd) or [host command](https://www.cyberciti.biz/faq/linux-unix-host-command-examples-usage-syntax/?utm_source=Linux_Unix_Command&utm_medium=faq&utm_campaign=nixcmd) to verify DNS connectivity:  
$ host cyberciti.biz  
dig cyberciti.biz

Summing up

You learned about IP networking config options on a Debian Linux. For more info consult the following man pages using the [help command](https://bash.cyberciti.biz/guide/Help_command?utm_source=Linux_Unix_Command&utm_medium=faq&utm_campaign=nixcmd) or [man command](https://bash.cyberciti.biz/guide/Man_command?utm_source=Linux_Unix_Command&utm_medium=faq&utm_campaign=nixcmd):

**man** **ip**

**man** 5 interfaces

**man** 8 nmcli