# Linux Commands

`arp -a`
Display the content of the ARP cache.

`arp –d` **`IPaddress`**
Deletes the entry with the IP address IPaddress.

`arp –s` **`IPaddress MACAddress`**
Adds a static entry to the ARP cache that is never overwritten by network events. The MAC address is entered as 6 hexadecimal bytes separated by colons.
Example: arp –s 10.0.1.12 00:02:2D:0D:68:C1

`ip –s –s neigh flush all`
command to clear the arp cache

`ip address add` **`IPAddr/xx`** `dev` **`interface`**
Adds IPAddr with prefix xx to interface. E.g., ip address add 128.10.1.10/24 dev eth0

`ip address del` **`IPAddr/xx`** `dev` **`interface`**
Deletes IPAddr with prefix xx on interface. E.g., ip address del 128.10.1.10/24 dev eth0

`ip address flush dev` **`interface`**
Deletes all statically assigned IP addresses for dev interface.

`ip address show dev` **`interface`**
Shows all assigned IP addresses for dev interface.

`netstat –i`
Displays a table with statistics of the currently configured network interfaces.

`netstat –rn`
Displays the kernel routing table. The *–n* option forces netstat to print the IP addresses. Without this option, netstat attempts to display the host names.

`netstat –an`
`netstat –tan`
`netstat -uan`
Displays the active network connections. The *–a* option display all active network connections, the *–ta* option displays only information on TCP connections, and the *–tu* option displays only information on UDP traffic. Omitting the *–n* option prints host names, instead of IP addresses.

`netstat –s`
Displays summary statistics for each protocol that is currently running on the host.

`ifconfig`
Displays the configuration parameters of all active interfaces.

`ifconfig` **`interface`**
Displays the configuration parameters of a single interface. For example, `ifconfig` **`eth0`** displays information on interface eth0.

`ifconfig` **`interface`** `down`
Disables the interface. For example: ifconfig eth0 down. No traffic is sent or received on a disabled interface.

`ifconfig` **`interface`** `up`
Enables an interface.

`ifconfig` **`interface IPAdr/xx`**
**`e.g.`** `ifconfig` **`eth0 10.0.1.8/24`**
Assigns interface *eth0* the IP address *10.0.1.8/24* and a broadcast address of *10.0.1.255*

`ifconfig eth0 mtu xxx`
Assigns MTU size xxx bytes to interface eth0

`sudo echo 1 > ‘/proc/sys/net/ipv4/ip_forward’`

```
   enables IPforwarding
sudo echo 0 > '/proc/sys/net/ipv4/ip_forward'
   disables IPforwarding
sysctl net.ipv4.ip_forward
   shows current status of ipforwarding

route add –net netaddress netmask mask gw gw_address
route add –net netaddress netmask mask dev iface
```

Adds a routing table entry for the network prefix identified by IP address **netaddress** and netmask **mask**. The next-hop is identified by IP address **gw_address** or by interface **iface**.

```
route add –host hostaddress gw gw_address
route add –host hostaddress dev iface
```

Adds a host route entry for IP address **hostaddress** with the next-hop identified by IP address **gw_address** or by interface **iface**

```
route add default gw gw_address
```

Sets the default route to IP address **gw_address**

```
route del –net netaddress netmask mask gw gw_address
route del –host hostaddress gw gw_address
route del default gw gw_address
```

Deletes an existing route from the routing table with specific arguments.

```
route -e
```

Displays the current routing table with extended fields. The command is identical to the `netstat –r` command.

```
ip route flush table main
```

Deletes all entries in the routing table on a PC. Please note that the local interface route(s) need to be added before adding any other static route entries after a flush table command. To add interface route(s) use the ifconfig interface down and ifconfig interface up (where e.g. interface = eth0)

```
ping IPaddr
```

Pings host with IP address IPaddr.

```
ping IPaddr –c num
```

Where num is the number of pings you want issued to destination IPaddr.

```
ping IPaddr –s num
```

Where num is the number of data bytes in the ICMP request message you want sent to destination IPaddr.

```
traceroute IPaddr –mxxx -qyyy
```

Command used to trace the route between an origin and a destination IP address IPaddr, where –m indicates the max TTL value and –q indicates the number of queries. E.g. m=2, and q=1.

```
ncat –l X -k -e "/bin/cat"
```

sets up a PC to listen on port X, default is for TCP, e.g., X=7

```
ncat –l X -k -u -e "/bin/cat"
```

sets up a PC to listen on port X, where -u use UDP instead of TCP, e.g., X=7

```
echoping –v  –s xxxx IPaddr
```

pings a host with IP address IPaddr using TCP (default), -v is for verbose, -s is packet size

```
echoping –v -u –s xxxx IPaddr
```

pings a host with IP address IPaddr, where -u indicates using UDP instead of TCP, -v is for verbose, -s is packet size