

Networking (Core)

Linux Commands Course · Section 12

Network Interfaces – ip a

Show all network interfaces and their IP addresses:

```
ip a
```

Example output:

```
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500
    inet 192.168.1.10/24 brd 192.168.1.255 scope global eth0
```

- `eth0, wlan0` – interface names
- `inet` – IPv4 address
- `inet6` – IPv6 address
- `state` – interface status (UP/DOWN)

Bring an interface up or down (root required):

```
sudo ip link set eth0 up
```

Routing Table – ip r

View the system routing table:

```
ip r
```

Example output:

```
default via 192.168.1.1 dev eth0
192.168.1.0/24 dev eth0 proto kernel scope link src 192.168.1.10
```

- default via → default gateway
- dev eth0 → which interface is used
- src → local source IP

Add or delete temporary routes:

```
sudo ip route add 10.10.0.0/16 via 192.168.1.1
sudo ip route del 10.10.0.0/16
```

Active Connections – ss (modern tool)

ss (socket statistics) shows open ports and connections.

```
ss -tulpn
```

- t → TCP
- u → UDP
- l → listening sockets
- p → show process using port
- n → show numeric addresses

Example:

```
Netid State  Recv-Q Send-Q Local Address:Port  Peer Address:Port
Process
tcp    LISTEN  0        128      0.0.0.0:22          0.0.0.0:*
users:(("sshd",pid=745,fd=3))
```

Legacy tool (if available):

Connectivity – ping

Test reachability of a host.

```
ping 8.8.8.8
```

Send a limited number of packets:

```
ping -c 4 example.com
```

Interrupt anytime with **Ctrl+C**.

Tracing Network Path – traceroute / tracepath

Show each hop between you and a destination.

```
traceroute example.com
```

If not installed, try:

```
tracepath example.com
```

Output shows latency at each hop – useful for debugging routing or latency issues.

DNS Lookups – dig and host

Query DNS records with `dig`

```
dig example.com
```

Show only the IP address:

```
dig +short example.com
```

Query specific record types:

```
dig example.com MX  
dig example.com NS
```

Simple lookup with `host`

HTTP & File Transfers – curl and wget



Fetch a URL or API data:

```
curl https://example.com
```

Save output to a file:

```
curl -o page.html https://example.com
```

Show headers only:

```
curl -I https://example.com
```

Send JSON data to an API:

Remote Access – ssh

Securely log into another machine:

```
ssh user@192.168.1.50
```

Use a key file instead of a password:

```
ssh -i ~/.ssh/id_rsa user@host
```

Exit remote session with **exit** or **Ctrl+D**.

Copy files securely using SSH:

```
scp report.txt user@192.168.1.50:/home/user/
```

Copy entire directories recursively:

Legacy Tool – telnet

Used for basic connectivity testing (not secure).

```
telnet example.com 80
```

If it connects, the port is open.

Use only for debugging – not for remote login.

NetworkManager CLI – nmcli

nmcli manages network connections on systems using **NetworkManager**.

List all connections:

```
nmcli connection show
```

Show active interfaces:

```
nmcli device status
```

Bring a connection up or down:

```
sudo nmcli connection up "Wired connection 1"  
sudo nmcli connection down "Wired connection 1"
```

View details for a specific interface:

Recap

- `ip a, ip r` – view interfaces and routes
- `ss -tulpn` – active sockets and ports
- `ping, traceroute, tracepath` – connectivity testing
- `dig, host` – DNS queries
- `curl, wget` – HTTP and file transfers
- `ssh, scp, telnet` – remote access and copy
- `nmcli` – manage connections via NetworkManager

These tools form the backbone of network troubleshooting and configuration.
