



Architecture & Deployment

2025-2026 v0.1.0 on branch main Rev: d1f684699a511826485586b2ea383cf021631add

Unix Networking

Learn the basics of Unix networking and how to make TCP connections.

You will need

- A Unix CLI
- An Ubuntu server with a public IP address to connect to

Recommended reading

- [Unix Basics & Administration](#)
- [Unix Processes](#)

Table of contents

- [Presentation](#)
- [Useful commands](#)
 - [The ip command](#)
 - [The ping command](#)
 - [The ss command](#)
 - [The nc command](#)
- [References](#)

Useful commands

Useful commands for unix networking

The `ip` command

The `ip` command is used to manipulate and display IP network information:

```
$> ip address
1: `lo`: <LOOPBACK,UP,LOWER_UP> mtu 65536 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
        valid_lft forever preferred_lft forever
    inet6 `::1`/128 scope host
        valid_lft forever preferred_lft forever
2: `eth0`: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 ...
    link/ether 06:5f:44:85:36:92 brd ff:ff:ff:ff:ff:ff
    inet `172.31.39.219`/20 brd 172.31.47.255 scope global dynamic eth0
        valid_lft 2665sec preferred_lft 2665sec
    inet6 `fe80::45f:44ff:fe85:3692`/64 scope link
        valid_lft forever preferred_lft forever
```

In this sample output, there are **two network interfaces**:

- The virtual loopback interface (`lo`) through which applications can communicate on the computer itself without actually hitting the network
- A physical **E**thernet interface (`eth0`) which has the private IP address `172.31.39.219` (i.e. the computer's address in its local network)

The `ping` command

The `ping` command tests the reachability of a host on an IP network. It measures the **round-trip time** (`rtt`) for messages sent to a computer and echoed back. The name comes from active sonar terminology that sends a pulse of sound and listens for the echo to detect objects under water.

It uses the Internet Control Message Protocol (ICMP), a **network layer** protocol (OSI layer 3).

```
$> ping -c 1 google.com
PING `google.com` (`172.217.21.238`) 56(84) bytes of data.
64 bytes from 172.217.21.238: icmp_seq=1 ttl=53 time=`1.12 ms`

--- google.com ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 1.125/1.125/1.125/0.000 ms
```

In this example, you can see that the domain name `google.com` was translated to the public IP address `172.217.21.238` by the Domain Name System, and that the round-trip to that computer took about 1.12 milliseconds.

Tip

The `-c 1` (or **c**ount) option tells ping to send only one ping. Remove it to keep pinging once per second.

The `ss` command

The socket statistics (`ss`) command (or the older `netstat` command) displays information about the open network sockets on the computer.

A **socket** is the software representation of a network communication's endpoint. For a TCP connection in an IP network, it corresponds to a connection made on an IP address and port number.

```
$> ss -tlnp
State  Recv-Q  Send-Q  Local Address:Port  Peer Address:Port  Process
LISTEN  0        80      127.0.0.1:3306       0.0.0.0:*          mysqld...
LISTEN  0        128     127.0.0.53%lo:53     0.0.0.0:*          systemd-resolve...
LISTEN  0        128      0.0.0.0:22          0.0.0.0:*          sshd...
LISTEN  0        128      [::]:22             [::]:*            sshd...
```

The above command lists the processes listening for TCP connections. In this example, we can see that there is a MySQL database listening on port 3306, a DNS resolver on port 53, and an SSH server on port 22.

More information

On some systems, you may need to add the `-e` (extended) option to display process information. You can remove the `-n` (or `--numeric`) option to see service names (e.g. `ssh` instead of `22`). The other options are `-t` for **T**CP, `-l` to only display listening sockets, and `-p` to show the **p**rocesses.

The `nc` command

The netcat (`nc`) command can read from and write to network connections using TCP or UDP.

```
$> nc -zv -w 2 google.com 80
Connection to google.com 80 port [tcp/http] succeeded!
$> nc -zv -w 2 google.com 81
nc: connect to google.com port 81 (tcp) timed out: Operation now in progress
nc: connect to google.com port 81 (tcp) failed: Network is unreachable
$> nc -zv -w 2 google.com 443
Connection to google.com 443 port [tcp/http] succeeded!
```

For example, the above two commands check whether ports 80, 81 and 443 are open on the computer reached by resolving the domain name `google.com`.

More information

The `-z` (zero bytes) option tells netcat to close the connection as soon as it opens, the `-v` option enables more **v**erbose output, and the `-w 2` tells netcat to **w**ait at most 2 seconds before giving up.

References

- [Internet Protocol](#)
 - [IP address](#)
 - [IPv4 & IPv6](#)
 - [Subnetworks](#)
 - [Network Address Translation \(NAT\)](#)
 - [Ports](#)
 - [List of TCP and UDP port numbers](#)
- [Ops School Curriculum](#)
 - [Networking 101](#)
 - [Networking 202](#)
- [Unix Networking Basics for the Beginner](#)
- [Unix Top Networking Commands and What They Tell You](#)
- [What happens when you type google.com into your browser and press enter?](#)

↑ [Back to top](#)