traceroute and mtr Cheatsheet

By Dejan Panovski • Updated on Nov 16, 2026

Quick reference for traceroute, tracepath, and mtr: UDP, ICMP, and TCP probes, hop limits, output codes, mtr reports, and Windows tracert equivalents

The `traceroute` command shows the path packets take to a host and the round-trip time to each hop. This cheatsheet covers probe methods, hop and timing options, output codes, `tracepath`, and `mtr` for continuous loss and latency reports.

Basic Syntax

Core `traceroute` command forms.

`traceroute host`	Trace the route with UDP probes
`traceroute -n host`	Show IP addresses only, skip reverse DNS
`traceroute -4 host`	Force IPv4
`traceroute -6 host`	Force IPv6
`traceroute host 1400`	Set the full packet length to 1400 bytes (default 60 for IPv4, 80 for IPv6)
`traceroute --version`	Check whether `traceroute` is installed

Install

Package names for `traceroute` and `mtr`.

`sudo apt install traceroute`	Install on Ubuntu, Debian, and derivatives
`sudo dnf install traceroute`	Install on Fedora, RHEL, and derivatives
`sudo apt install mtr-tiny`	Install `mtr` without the GTK interface
`sudo dnf install mtr`	Install `mtr` on Fedora, RHEL, and derivatives
`tracepath host`	Often preinstalled with iputils; no root needed

Probe Methods

Switch protocols when the default UDP probes are filtered.

`traceroute host`	UDP to port 33434, one port higher for each probe
`traceroute -I host`	ICMP echo, the same method as Windows `tracert`
`sudo traceroute -T host`	TCP SYN to port 80; requires root
`sudo traceroute -T -p 443 host`	TCP SYN to port 443
`traceroute -U host`	UDP to the fixed port 53
`traceroute -U -p 123 host`	UDP to the fixed port 123
`traceroute -p 40000 host`	Start the default UDP probes at port 40000

Hops and Timing

Control how far the trace goes and how long it waits.

`traceroute -m 15 host`	Stop after 15 hops (default 30)
`traceroute -f 5 host`	Start at hop 5 and skip the local hops
`traceroute -q 1 host`	Send one probe per hop (default 3)
`traceroute -w 2 host`	Wait up to 2 seconds for a reply (default 5)
`traceroute -N 1 host`	Send probes one at a time (default 16 at once)
`traceroute -z 0.5 host`	Pause 0.5 seconds between probes; values above 10 are milliseconds

Source and Path Details

Pick the outgoing interface and collect extra data per hop.

`traceroute -i eth0 host`	Send probes through `eth0`
`traceroute -s 192.168.1.10 host`	Use 192.168.1.10 as the source address
`traceroute --mtu host`	Discover the path MTU, shown as `F=1500`
`traceroute -F host`	Set the Do Not Fragment bit
`traceroute -A host`	Show the AS number for each hop
`traceroute -e host`	Show ICMP extensions such as MPLS labels

Reading the Output

What the symbols after each hop mean.

`* * *`	No reply within the wait time; the hop may filter or rate-limit probes
`* 12.3 ms *`	Some probes lost or rate-limited at that hop
`Several IPs on one hop`	Probes took different equal-cost paths
`!H`	Host unreachable
`!N`	Network unreachable
`!P`	Protocol unreachable
`!X`	Communication administratively prohibited, usually a firewall
`!F`	Fragmentation needed

tracepath

Trace without root and report the path MTU.

`tracepath host`	Trace with UDP and discover the MTU at each hop
`tracepath -n host`	Skip reverse DNS
`tracepath -b host`	Show both hostnames and IP addresses
`tracepath -m 15 host`	Stop after 15 hops (default 30)
`tracepath -p 40000 host`	Set the initial destination port
`tracepath -6 host`	Force IPv6

mtr Live View

Combine `ping` and `traceroute` in a continuously updating screen.

`mtr host`	Show live loss and latency for every hop
`mtr -n host`	Skip reverse DNS
`mtr -b host`	Show both hostnames and IP addresses
`mtr -T -P 443 host`	Use TCP SYN to port 443 instead of ICMP
`mtr -u host`	Use UDP instead of ICMP
`mtr -i 2 host`	Send a probe every 2 seconds (default 1)
`mtr -z host`	Show the AS number for each hop

mtr Reports

Run a fixed number of cycles and print the results.

`mtr -r host`	Run 10 cycles, print a report, and exit
`mtr -r -c 100 host`	Run 100 cycles for steadier loss figures
`mtr -rw host`	Wide report with full hostnames
`mtr -rwn -c 50 host > mtr.txt`	Save a report to send to your ISP
`mtr -j host`	Print the report as JSON
`mtr -C host`	Print the report as CSV

mtr Keys

Keys for the interactive `mtr` screen.

`q`	Quit
`p`	Pause; press Space to resume
`d`	Switch the display mode
`n`	Toggle DNS resolution
`r`	Reset all counters
`j`	Toggle between latency and jitter columns
`z`	Toggle AS number lookups
`? or h`	Show help

Windows tracert Equivalents

Linux commands for the common `tracert` options.

`traceroute -I host`	ICMP probes, like `tracert host`
`traceroute -I -n host`	ICMP probes without reverse DNS, like `tracert /d host`
`traceroute -I -m 15 host`	ICMP probes with a 15-hop limit, like `tracert /h 15 host`
`traceroute -I -w 3 host`	ICMP probes with a 3-second timeout, like `tracert /w 3000 host`
`traceroute -I -4 host`	ICMP probes over IPv4, like `tracert /4 host`
`mtr -r host`	Per-hop loss statistics, similar to `pathping host`

Troubleshooting

Quick checks for common `traceroute` and `mtr` issues.

`traceroute: command not found`	Install the `traceroute` package or use `tracepath`
`You do not have enough privileges to use this traceroute method.`	`-T` needs root, and `-I` may need it too; rerun with `sudo`
`Every hop shows * * *`	Probes or replies may be filtered, or the destination may be unreachable; try `-I` or `sudo traceroute -T -p 443 host`
`Replies stop after one hop`	Later devices may filter probes; test the service port with `-T -p`
`High latency on one middle hop only`	The router deprioritizes replies; ignore it unless the delay reaches the destination
`Loss in mtr on a middle hop only`	ICMP rate limiting; real loss continues through to the last hop

Related Guides

Use these guides for the full network troubleshooting workflow.

traceroute Command in Linux	Full guide with an output walkthrough
ping cheatsheet	Reachability and latency checks
ip cheatsheet	Interfaces, addresses, and routes
tcpdump cheatsheet	Capture the probes and replies on the wire
nmap cheatsheet	Check whether a port is open, closed, or filtered