gping Ping with a graph

START HERE

`gping <host>`
Graph ping latency

`gping -b 120 <host>`
Longer history

`gping <a> <b>`
Compare hosts

`gping -s <host>`
Plain dot graph

`gping --cmd "<cmd>"`
Graph command time

`gping -c red,blue <a> <b>`
Set colors

`gping -n 1 <host>`
Slower updates

`gping aws:<region>`
Ping a cloud region

Targets hosts or commands → `every interval` → **Ping / run** measure time → `keep buffer` → **Live graph** one line each

CONCEPTS

Each target gets its own colored line, so several hosts share one graph for comparison.

With --cmd the targets are commands, and the graph shows how long each run takes.

PING

`gping <host>`
Graph latency to a host

`gping <host1> <host2>`
Compare several hosts

`gping aws:<region>`
Ping a cloud region, e.g. aws:eu-west-1

`gping -4 <host>`
Resolve to IPv4

`gping -6 <host>`
Resolve to IPv6

PING (cont.)

`gping -i <interface> <host>`
Ping through an interface
Handy to compare Wi-Fi and Ethernet paths.

`gping --clear <host>`
Wipe the graph on exit

COMMAND TIMING

`gping --cmd "<command>"`
Graph how long a command takes

`gping --cmd "<cmd1>" "<cmd2>"`
Compare commands

`gping --cmd -n 2 "<command>"`
Run the command every 2 seconds
Default interval is 0.5 s for commands, 0.2 s for ping.

TIMING AND HISTORY

`gping -n <seconds> <host>`
Set update interval (0.5 allowed)

`gping -b <seconds> <host>`
Seconds of history shown (default 30)

APPEARANCE

`gping -s <host>`
Dots instead of braille
Use when the terminal font lacks braille glyphs.

`gping -c <color> <host>`
Line color: name or #RRGGBB

`gping -c red,blue <a> <b>`
Colors match targets in order

`gping --vertical-margin <n> <host>`
Space above and below the graph

APPEARANCE (cont.)

`gping --horizontal-margin <n> <host>`
Space left and right of the graph

HELP

`gping --help`
List all options

`gping --version`
Show installed version

RECIPES

Compare DNS resolvers

```
gping 1.1.1.1 8.8.8.8 9.9.9.9
```

Pin a network path

```
gping -i wlan0 1.1.1.1
# swap in eth0 and compare the two runs
```

Watch a web request

```
gping --cmd "curl -so /dev/null https://example.com"
```

Long, readable capture

```
gping -n 1 -b 300 -s -c light-green,yellow <a> <b>
```