iperf3 Network throughput tester

CONCEPTS

> iperf3 needs a server at one end and a client at the other; it measures the path between them, not either machine alone. Test only networks and hosts you are authorised to use.

Server iperf3 -s, port 5201
↓ control
Client iperf3 -c host
↓ data streams
Interval reports every 1 s
↓ summary
Sender / receiver final totals

> Default direction is client to server. -R reverses it, so the client receives; --bidir does both at once. Open TCP and UDP port 5201 in any firewall in between.

SERVER

`iperf3 -s`
Listen for tests (port 5201)

`iperf3 -s -p <port>`
Listen on another port

`iperf3 -s -D`
Run as a background daemon

RUNNING A TEST

`iperf3 -c <host>`
Run a TCP test

`iperf3 -c <host> -p <port>`
Connect to another port

`iperf3 -c <host> -t <sec>`
Set duration (default 10 s)

`iperf3 -c <host> -O <sec>`
Skip the first seconds in stats

DIRECTION & STREAMS

`iperf3 -c <host> -R`
Reverse: server sends to client

DIRECTION & STREAMS (cont.)

`iperf3 -c <host> --bidir`
Send and receive at the same time

`iperf3 -c <host> -P <n>`
Run n parallel streams

UDP & BITRATE

`iperf3 -c <host> -u`
Run a UDP test (1 Mbit/s default)

`iperf3 -c <host> -u -b <rate>`
UDP at a target rate, e.g. 100M

`iperf3 -c <host> -b <rate>`
Cap a TCP test's rate

`iperf3 -c <host> -u -R`
UDP loss and jitter, download

TUNING

`iperf3 -c <host> -w <size>`
Set socket buffer or window

OUTPUT

`iperf3 -c <host> -J`
Output as JSON

`iperf3 -c <host> -i <sec>`
Report interval (0 turns it off)

`iperf3 -c <host> -f <unit>`
Units: k, m, g, t (bits) or K, M, G, T (bytes)

`iperf3 -c <host> --get-server-output`
Also print the server's view

READING RESULTS

Output columns

Column	Meaning
`Transfer`	Data moved in the interval
`Bitrate`	Throughput in that interval
`Retr`	TCP retransmits (sender)
`Cwnd`	Congestion window (sender)
`Jitter`	UDP delay variation
`Lost/Total`	UDP datagrams lost / sent