

# BBR3 Performance Test Summary

Brian Tierney, FasterData LLC

ESnet Consultant

# Test tool: iperf3-mt-beta

## Test Harness config:

```
iterations = 10
src-cmd = numactl -N 6 iperf3 -c {dst} -C cubic -t 60 -4 -J
src-cmd = numactl -N 6 iperf3 -c {dst} -C bbr -t 60 -4 -J
src-cmd = numactl -N 6 iperf3 -c {dst} -C cubic -t 60 -4 -P 8 -J
src-cmd = numactl -N 6 iperf3 -c {dst} -C bbr -t 60 -4 -P 8 -J
src-cmd = numactl -N 6 iperf3 -c {dst} -C cubic -t 60 -4 -P 8 --fq-
rate 12G -J
src-cmd = numactl -N 6 iperf3 -c {dst} -C bbr -t 60 -4 -P 8 --fq-rate
12G -J
```

Testing harness available from: <https://github.com/esnet/testing-harness>

# lbl-dev-dtn1.es.net to star-dtn1.es.net

Dest Host	RTT (ms)	Streams	CC Alg	Pacing (Gbps)	Tput (Gbps)	Stddev (nvals)	RXMTs
star-dtn1.es.net	44	1	bbr	0.0	24.65	2.59 (10)	197207
star-dtn1.es.net	44	1	cubic	0.0	26.85	2.63 (10)	23821
star-dtn1.es.net	44	8	bbr	0.0	96.09	0.39 (10)	207756
star-dtn1.es.net	44	8	cubic	0.0	94.66	0.46 (10)	3747
star-dtn1.es.net	44	8	bbr	12.0	92.00	0.56 (10)	439
star-dtn1.es.net	44	8	cubic	12.0	95.22	0.12 (10)	1787

Things to note:

- Throughput is slightly better with CUBIC
- Way more retransmits with BBR

# lbl-dev-dtn1.es.net to cern773-dtn1.es.net

Dest Host	RTT (ms)	Streams	CC Alg	Pacing (Gbps)	Tput (Gbps)	Stddev (nvals)	RXMTs
cern773-dtn1.es.net	150	1	bbr	0.0	22.90	3.58 (10)	568340
cern773-dtn1.es.net	150	1	cubic	0.0	19.50	4.04 (10)	68051
cern773-dtn1.es.net	150	8	bbr	0.0	47.96	2.43 (10)	300157
cern773-dtn1.es.net	150	8	cubic	0.0	30.64	7.35 (10)	49778
cern773-dtn1.es.net	150	8	bbr	12.0	50.58	2.21 (10)	302178
cern773-dtn1.es.net	150	8	cubic	12.0	44.08	3.58 (10)	61393

Things to note:

- Throughput is better with BBR
- Stddev is better with BBR

# pscheduler with iperf3

- Current version of perfSONAR does not include iperf3-mt
  - All parallel stream tests are CPU limited on a single core
- Test Harness config:

```
iterations = 6
src-cmd pscheduler task--format=json throughput --client-cpu-affinity=28 --zero-copy --congestion=cubic -
--ip-version 4 --parallel 1 --duration PT60S --dest {dst}
src-cmd = pscheduler task --format=json throughput --client-cpu-affinity=28 --zero-copy --congestion=bbr
--ip-version 4 --parallel 1 --duration PT60S --dest {dst}
src-cmd = pscheduler task --format=json throughput --congestion=cubic --ip-version 4 --parallel 8 --
duration PT60S --dest {dst}
src-cmd = pscheduler task --format=json throughput --congestion=bbr --ip-version 4 --parallel 8 --
duration PT60S --dest {dst}
pacing = 12gbit
src-cmd = pscheduler task --format=json throughput --congestion=cubic --ip-version 4 --parallel 8 --
duration PT60S --dest {dst}
src-cmd = pscheduler task --priority 100 --format=json throughput --congestion=bbr --ip-version 4 --
parallel 8 --duration PT60S --dest {dst}
```

# lbl-dev-dtn1.es.net to ps02-b.farm.particle.cz

Dest Host	RTT (ms)	Streams	CC Alg	Pacing (Gbps)	Tput (Gbps)	Stddev (nvals)	RXMTs
ps02-b.farm.particle.cz	149	1	bbr	0.0	12.45	1.08 (6)	985391
ps02-b.farm.particle.cz	149	1	cubic	0.0	2.78	1.36 (6)	79310
ps02-b.farm.particle.cz	149	8	bbr	0.0	19.19	1.68 (6)	312219
ps02-b.farm.particle.cz	149	8	cubic	0.0	16.69	1.98 (6)	245647
ps02-b.farm.particle.cz	149	8	bbr	11.0	18.13	1.07 (2)	267929
ps02-b.farm.particle.cz	149	8	cubic	11.0	15.43	2.32 (6)	231081

Things to note:

- Single stream throughput is 5X better with BBR

# lbl-dev-dtn1.es.net to pygrid-sonar2.lancs.ac.uk

Dest Host	RTT (ms)	Streams	CC Alg	Pacing (Gbps)	Tput (Gbps)	Stddev (nvals)	RXMTs
pygrid-sonar2.lancs.ac.uk	147	1	bbr	0.0	5.61	0.84 (6)	19721
pygrid-sonar2.lancs.ac.uk	147	1	cubic	0.0	2.60	1.18 (6)	4966
pygrid-sonar2.lancs.ac.uk	147	8	bbr	0.0	20.26	0.90 (5)	525351
pygrid-sonar2.lancs.ac.uk	147	8	cubic	0.0	8.34	0.73 (5)	112758
pygrid-sonar2.lancs.ac.uk	147	8	bbr	11.0	18.19	1.32 (5)	685559
pygrid-sonar2.lancs.ac.uk	147	8	cubic	11.0	8.46	2.82 (6)	89142

Things to note:

- Throughput is considerably better with BBR
- Likely CPU bound on single stream iperf3 for 8 stream tests

# lbl-dev-dtn1.es.net to ps-london-bw.perf.ja.net

Dest Host	RTT (ms)	Streams	CC Alg	Pacing (Gbps)	Tput (Gbps)	Stddev (nvals)	RXMTs
ps-london-bw.perf.ja.net	141	1	bbr	0.0	3.69	0.01 (6)	0
ps-london-bw.perf.ja.net	141	1	cubic	0.0	3.75	0.00 (6)	0
ps-london-bw.perf.ja.net	141	8	bbr	0.0	26.70	1.69 (6)	20
ps-london-bw.perf.ja.net	141	8	cubic	0.0	18.04	0.24 (6)	0
ps-london-bw.perf.ja.net	141	8	bbr	11.0	26.38	2.59 (6)	0
ps-london-bw.perf.ja.net	141	8	cubic	11.0	24.02	4.76 (6)	0

Things to note:

- Single stream is bad for both
- BBR better with un-paced



# lbl-dev-dtn1.es.net to fiona.sce.pennren.net (10G)

Dest Host	RTT (ms)	Streams	CC Alg	Pacing (Gbps)	Tput (Gbps)	Stddev (nvals)	RXMTs
fiona.sce.pennren.net	71	1	bbr	0.0	1.82	0.00 (4)	0
fiona.sce.pennren.net	71	1	cubic	0.0	1.83	0.07 (6)	84
fiona.sce.pennren.net	71	8	bbr	0.0	9.19	0.02 (3)	4020
fiona.sce.pennren.net	71	8	cubic	0.0	9.34	0.02 (2)	7342
fiona.sce.pennren.net	71	8	bbr	1.0	8.98	0.50 (5)	2131
fiona.sce.pennren.net	71	8	cubic	1.0	9.19	0.40 (6)	3926

Things to note:

- Single stream is terrible. Why?
- Otherwise no difference

# lbl-dev-dtn1.es.net to ps01-l-v4.farm.particle.cz (1G host)

Dest Host	RTT (ms)	Streams	CC Alg	Pacing (Gbps)	Tput (Gbps)	Stddev (nvals)	RXMTs
ps01-l-v4.farm.particle.cz	149	1	bbr	0.0	0.91	0.00 (2)	1364
ps01-l-v4.farm.particle.cz	149	1	cubic	0.0	0.93	0.00 (3)	1575
ps01-l-v4.farm.particle.cz	149	8	bbr	0.0	0.93	0.00 (3)	23955
ps01-l-v4.farm.particle.cz	149	8	cubic	0.0	0.94	0.00 (2)	2138
ps01-l-v4.farm.particle.cz	149	8	bbr	12.0	0.93	0.00 (3)	20246
ps01-l-v4.farm.particle.cz	149	8	cubic	12.0	0.94	0.00 (3)	2197

Things to note:

- Parallel streams step on each other less than expected

pscheduler with iperf2

Running now....

# Conclusions

- Pacing helps less than expected
- BBR helps, but less than expected
- Most tests are receiver CPU limited
  - Single threaded iperf3
- Next Steps:
  - Iperf2 testing
  - More testing of 100G to 10G and 1G hosts
  - BBR/CUBIC competing flows
  - MTU size testing