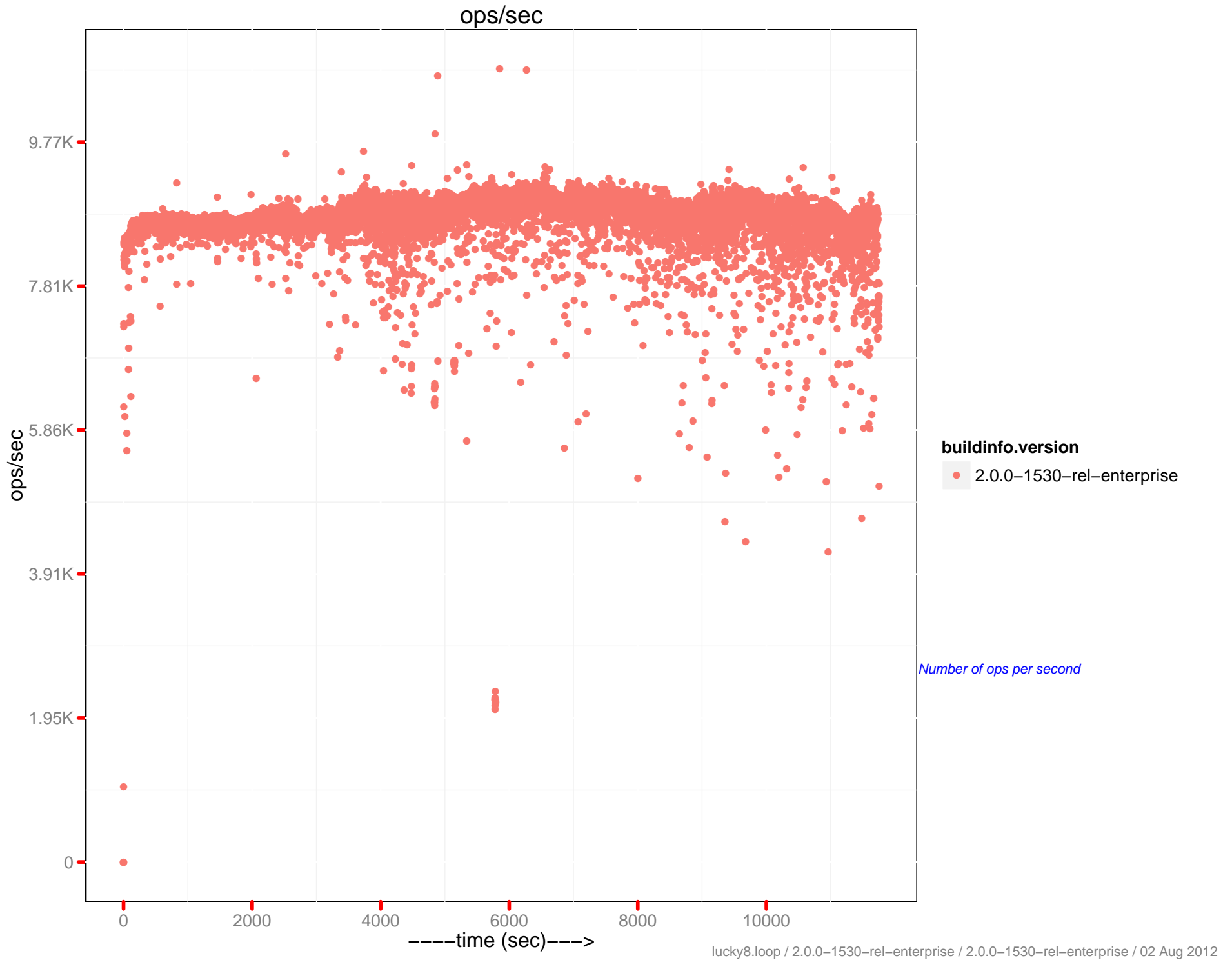
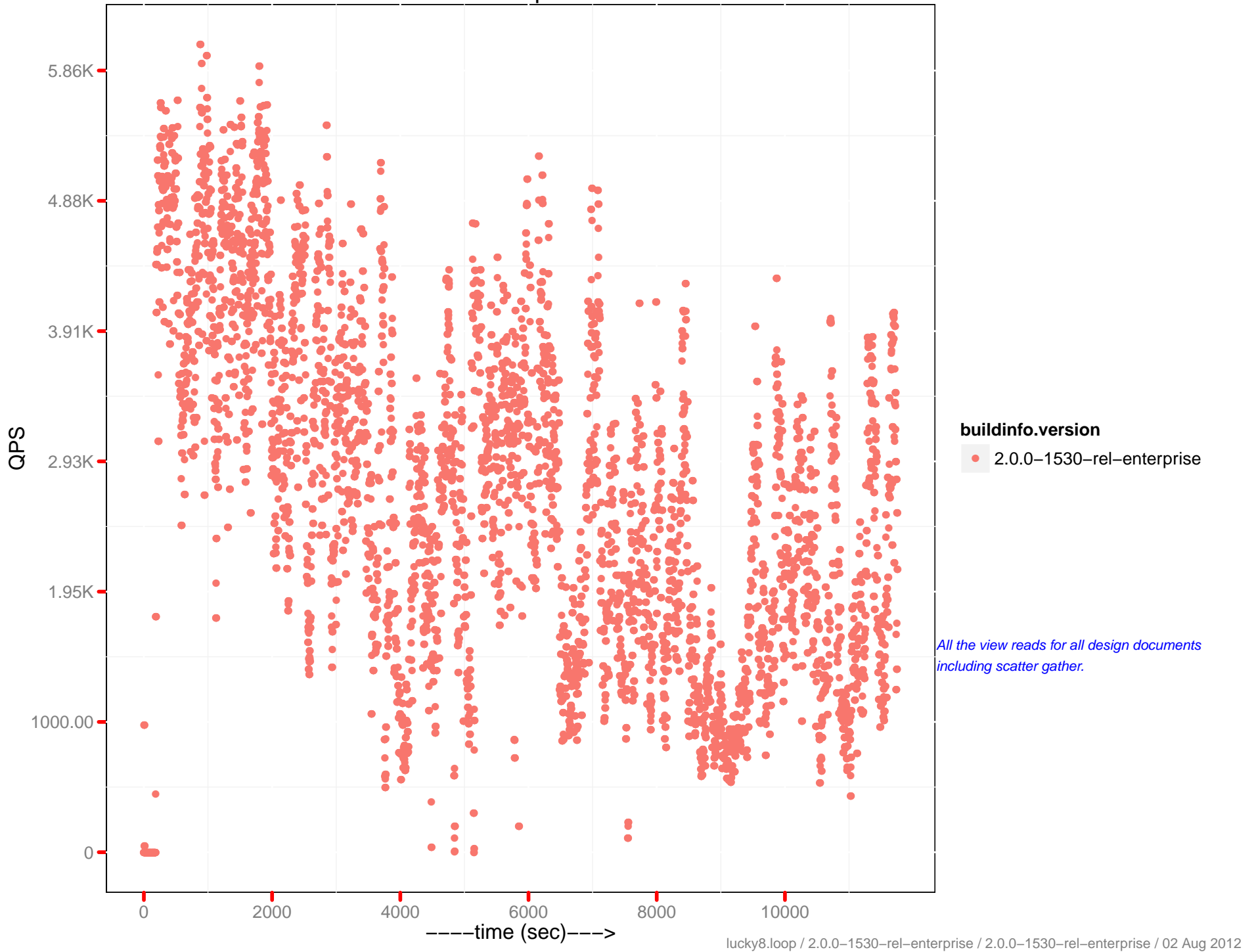


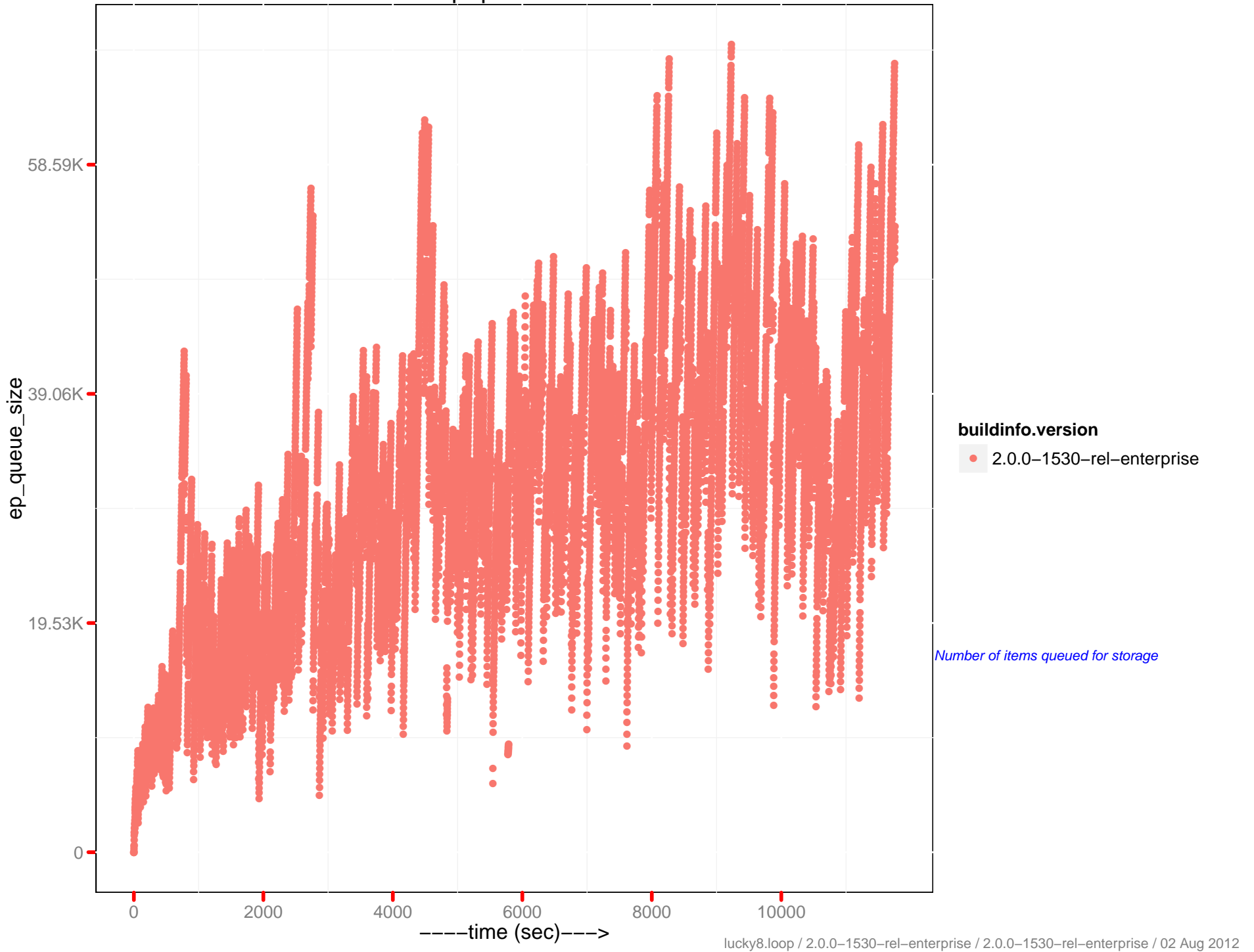
	2.0.0 – 1530	2.0.0 – 1530
<i>Runtime (in hr)</i>	3.27	NA
<i>Avg. Drain Rate</i>	423.33	NANA
<i>Peak Disk (GB)</i>	49.96	NA
<i>Peak Memory (GB)</i>	10.88	NA
<i>Avg. OPS</i>	8.73K	NANA
<i>Avg. mem memcached (GB)</i>	10.21	NA
<i>Avg. mem beam.smp (MB)</i>	554.92	NA
<i>Latency-get (90th) (ms)</i>	0.3	NA
<i>Latency-get (95th) (ms)</i>	0.53	NA
<i>Latency-get (99th) (ms)</i>	2.77	NA
<i>Latency-set (90th) (ms)</i>	0.47	NA
<i>Latency-set (95th) (ms)</i>	0.67	NA
<i>Latency-set (99th) (ms)</i>	2.99	NA
<i>Latency-query (80th) (ms)</i>	6.91	NA
<i>Latency-query (90th) (ms)</i>	9.95	NA
<i>Latency-query (95th) (ms)</i>	21.32	NA
<i>Latency-query (99th) (ms)</i>	81.62	NA
<i>Latency-query (99.9th) (ms)</i>	325.17	NA
<i>Avg. QPS</i>	670.57	NA
<i>Rebalance Time (sec)</i>	0	NA
<i>Testrunner Version</i>	799f985	NA



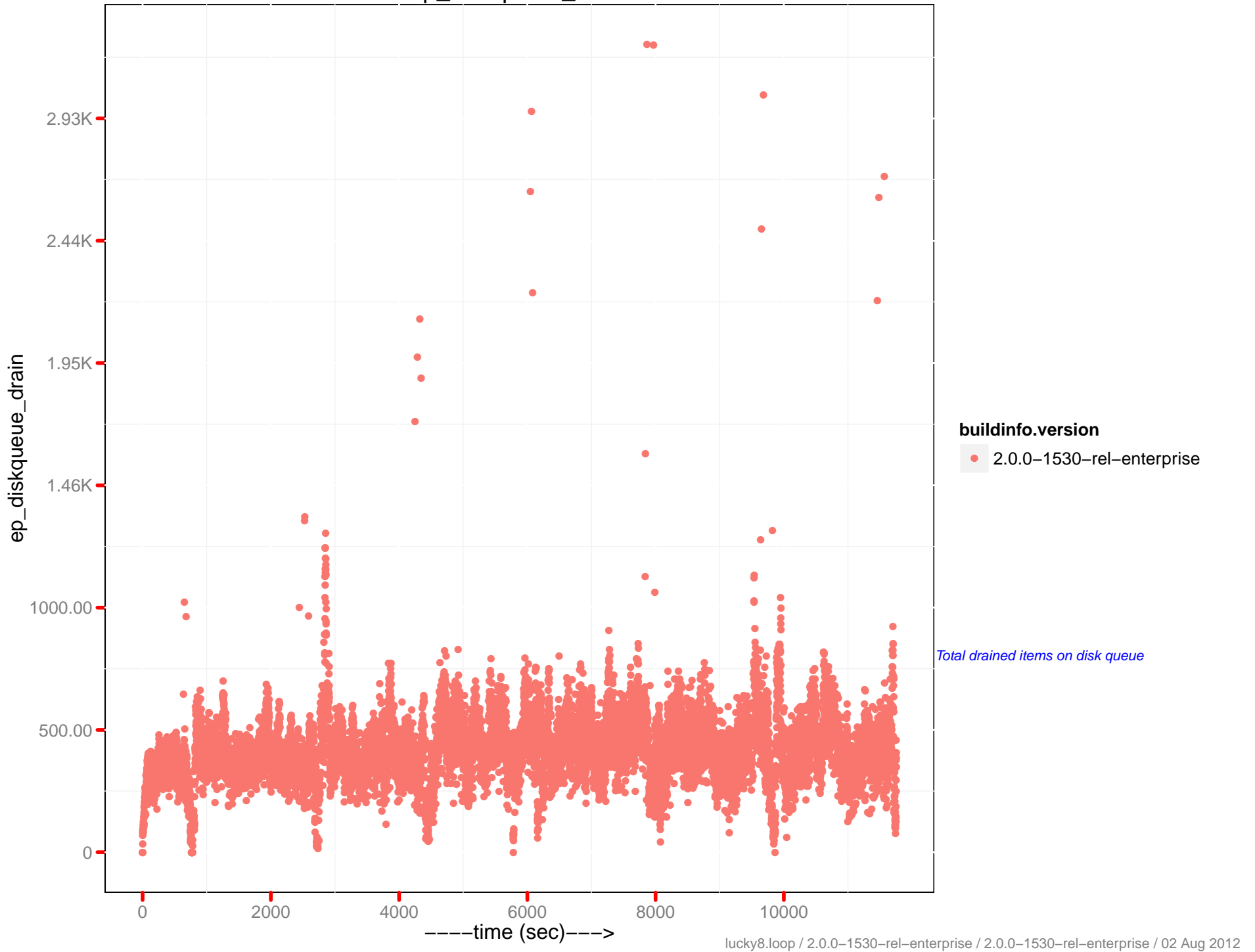
View read per sec.



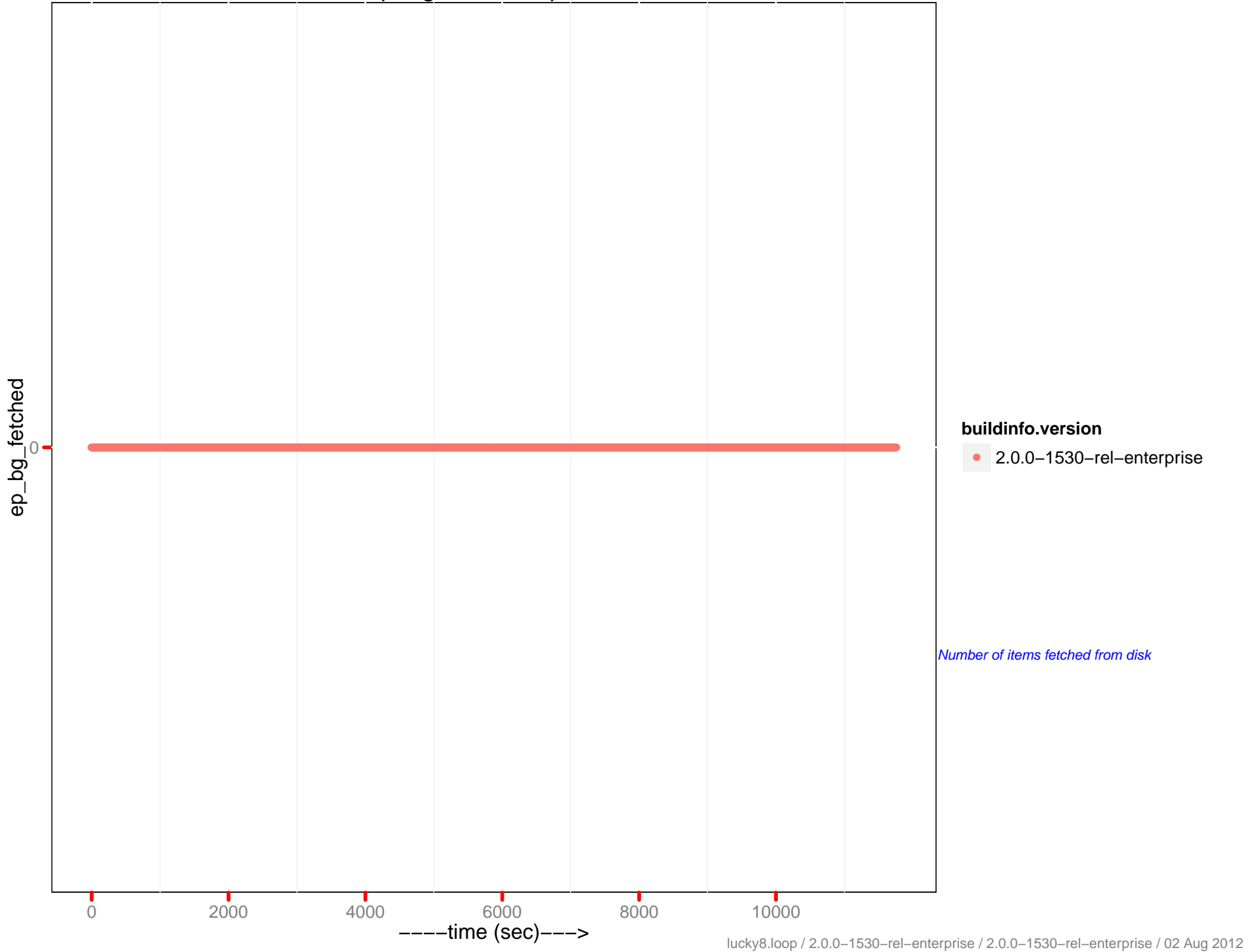
ep queue size



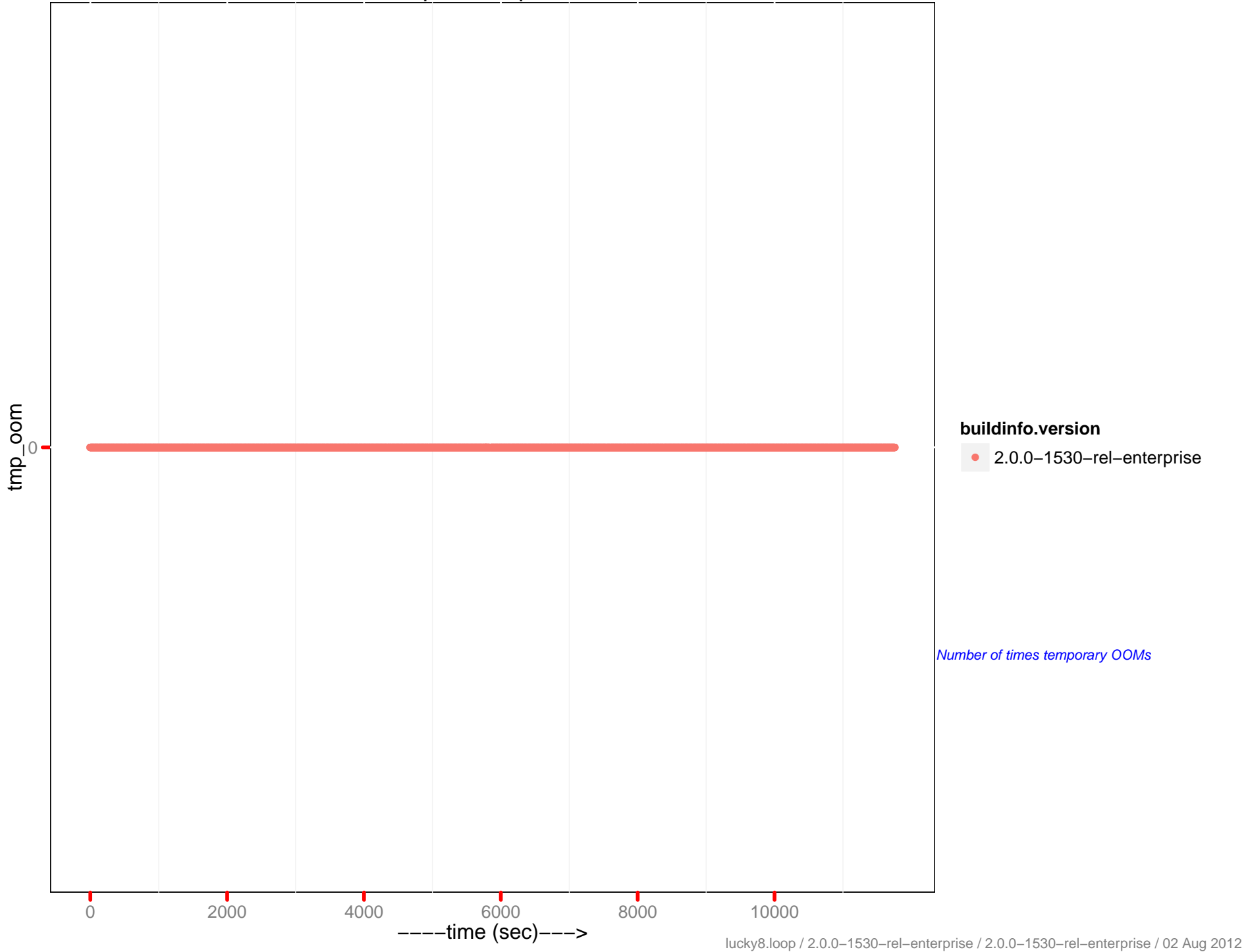
ep_diskqueue_drain



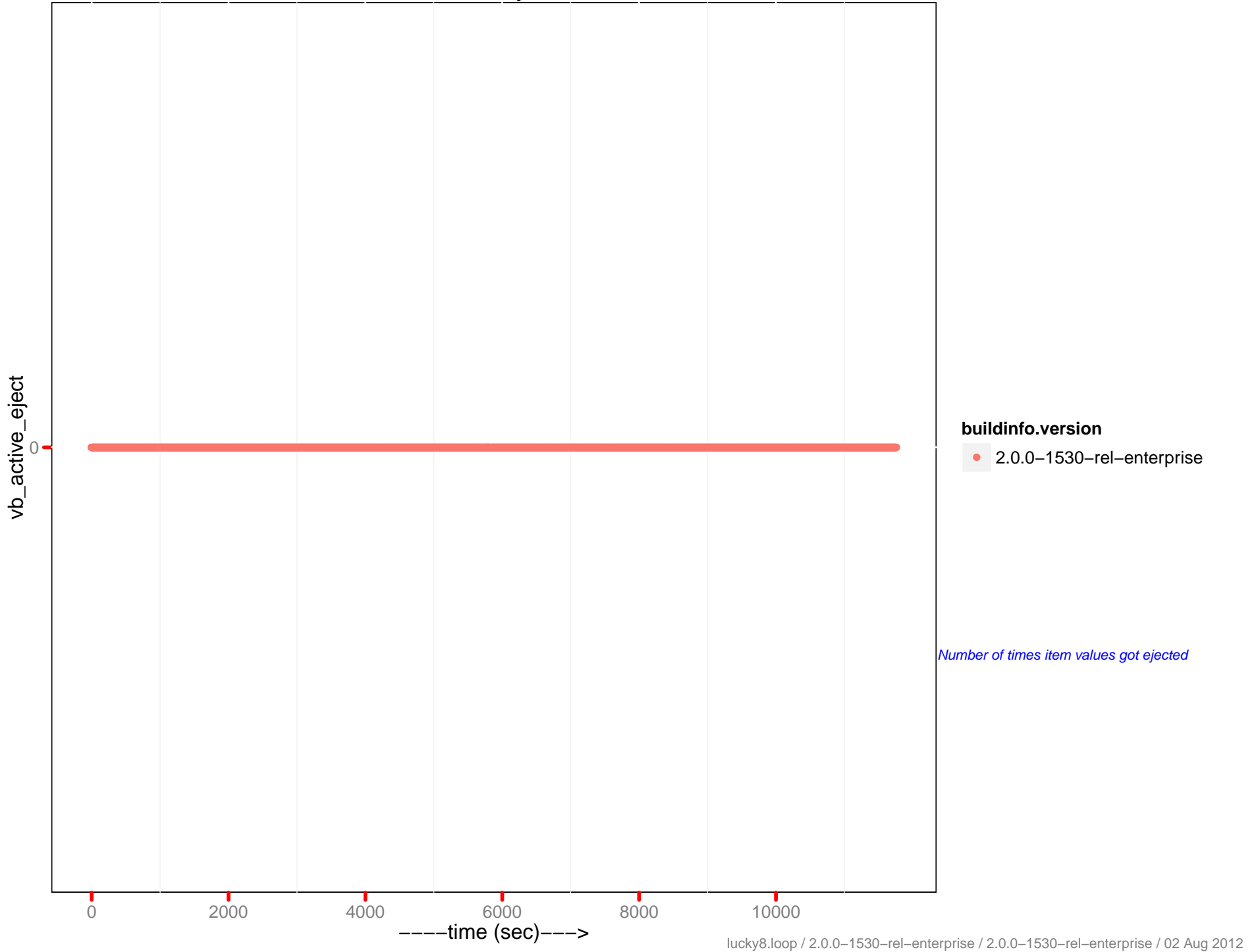
ep_bg_fetched ops/sec



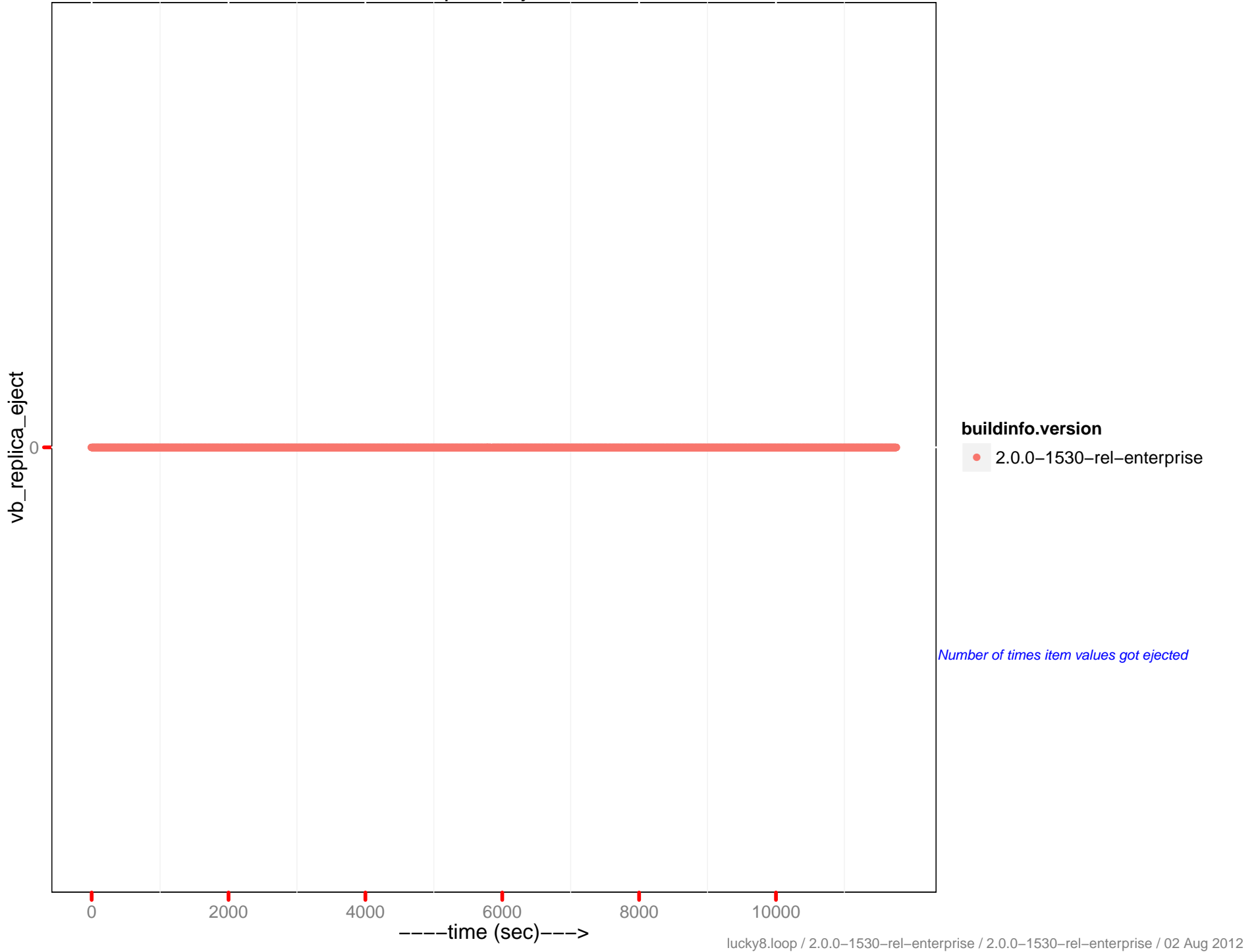
tmp_oom ops/sec



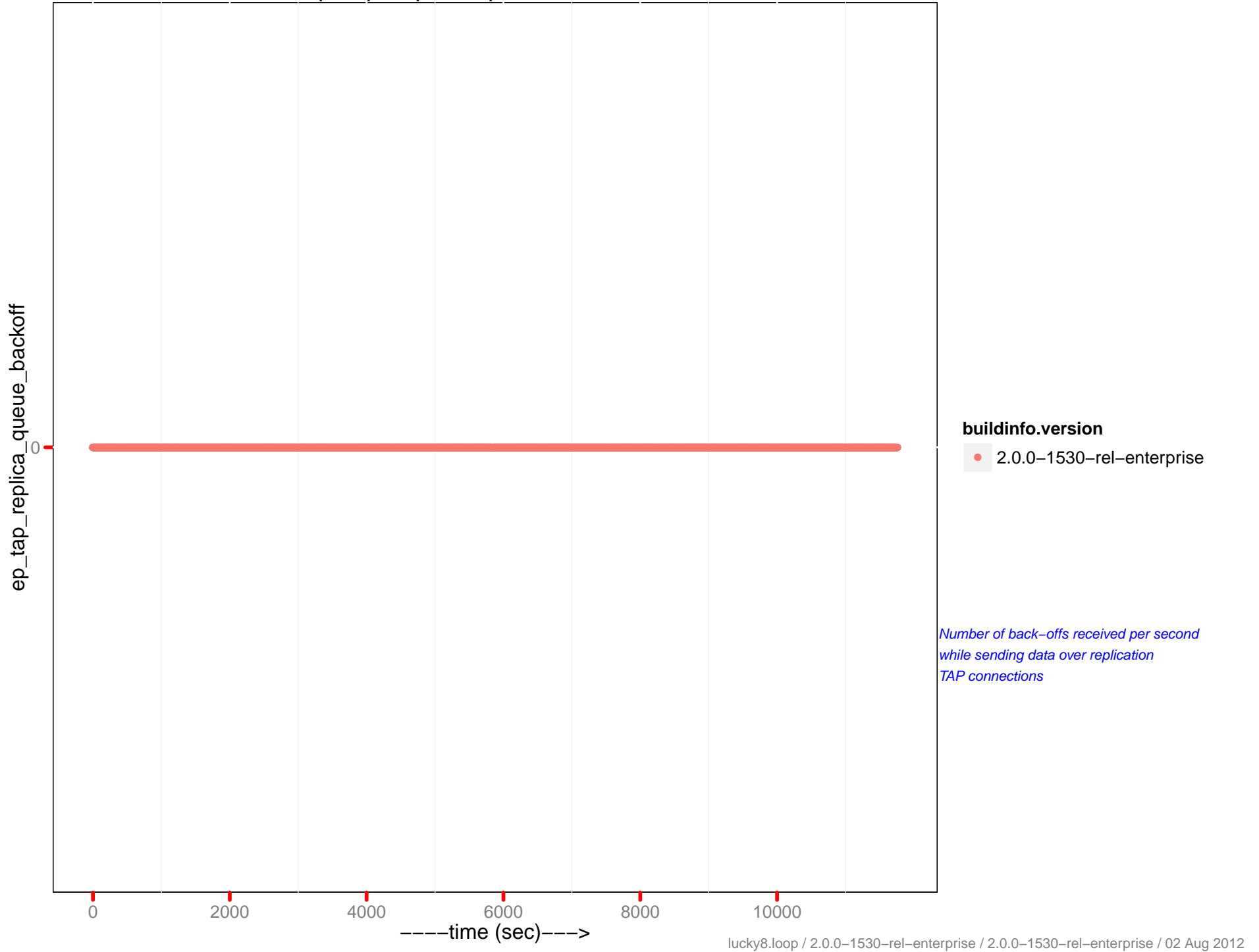
vb_active_eject/sec



vb_replica_eject/sec



ep_tap_replica_queue_backoff/sec

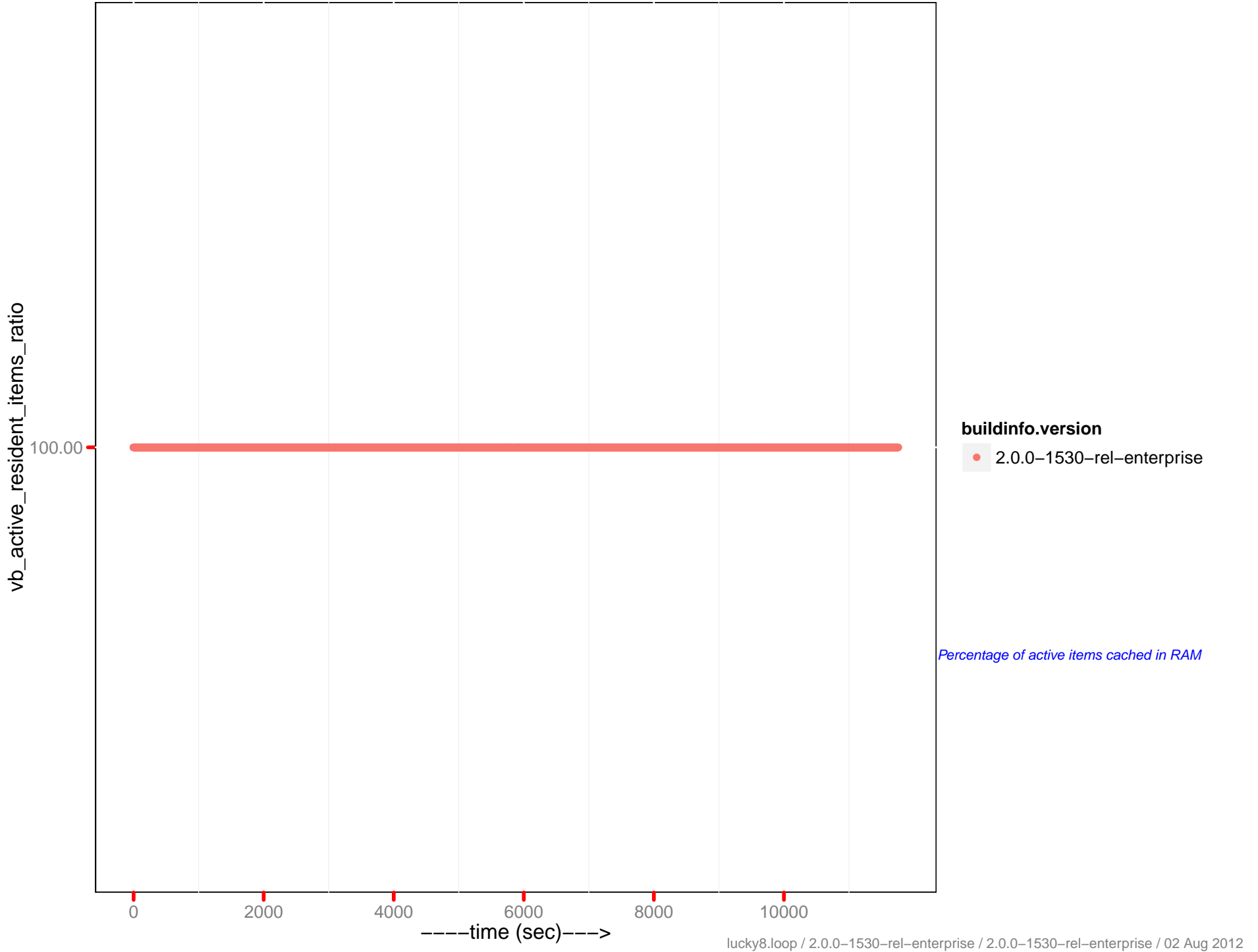


buildinfo.version

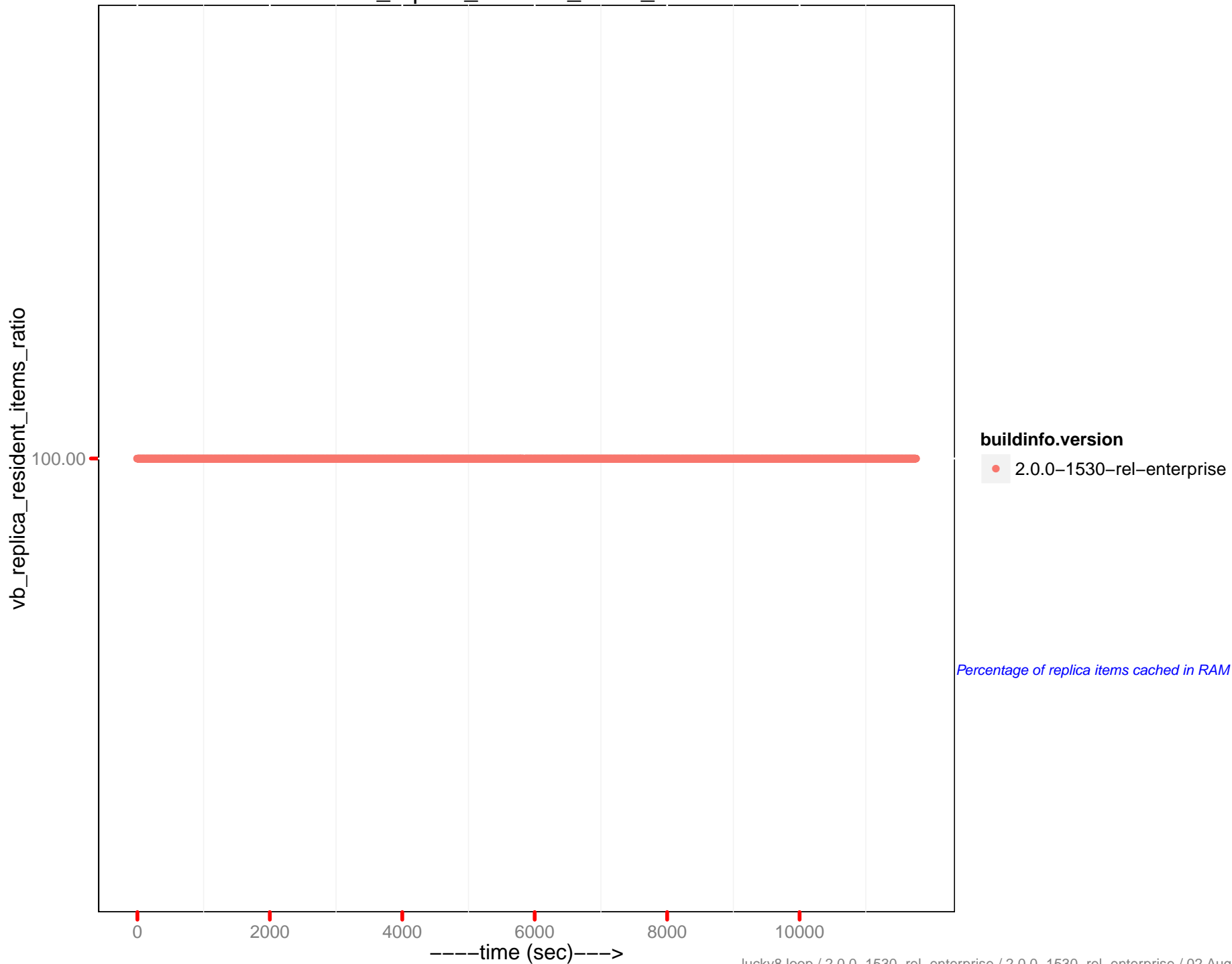
• 2.0.0-1530-rel-enterprise

*Number of back-offs received per second
while sending data over replication
TAP connections*

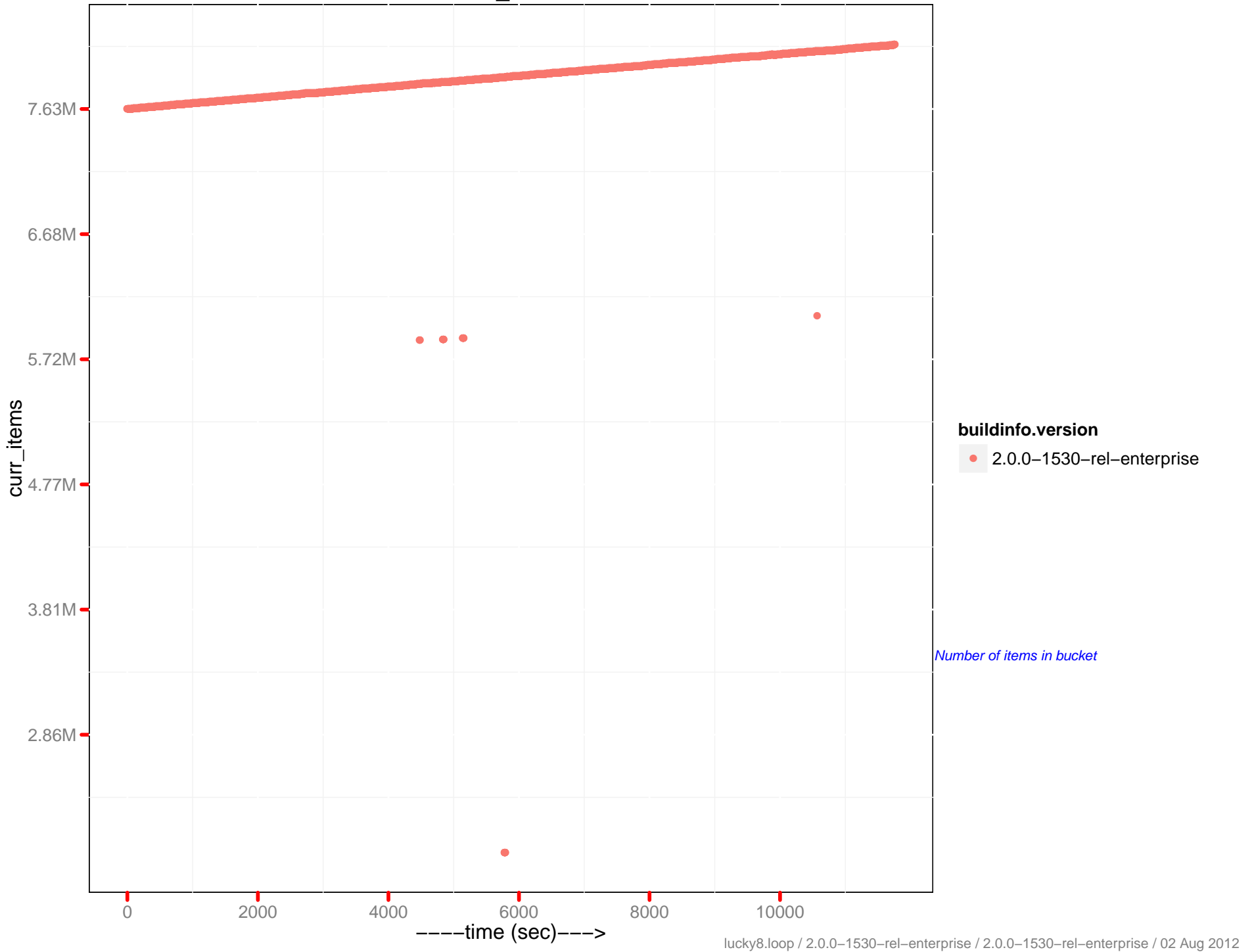
vb_active_resident_items_ratio



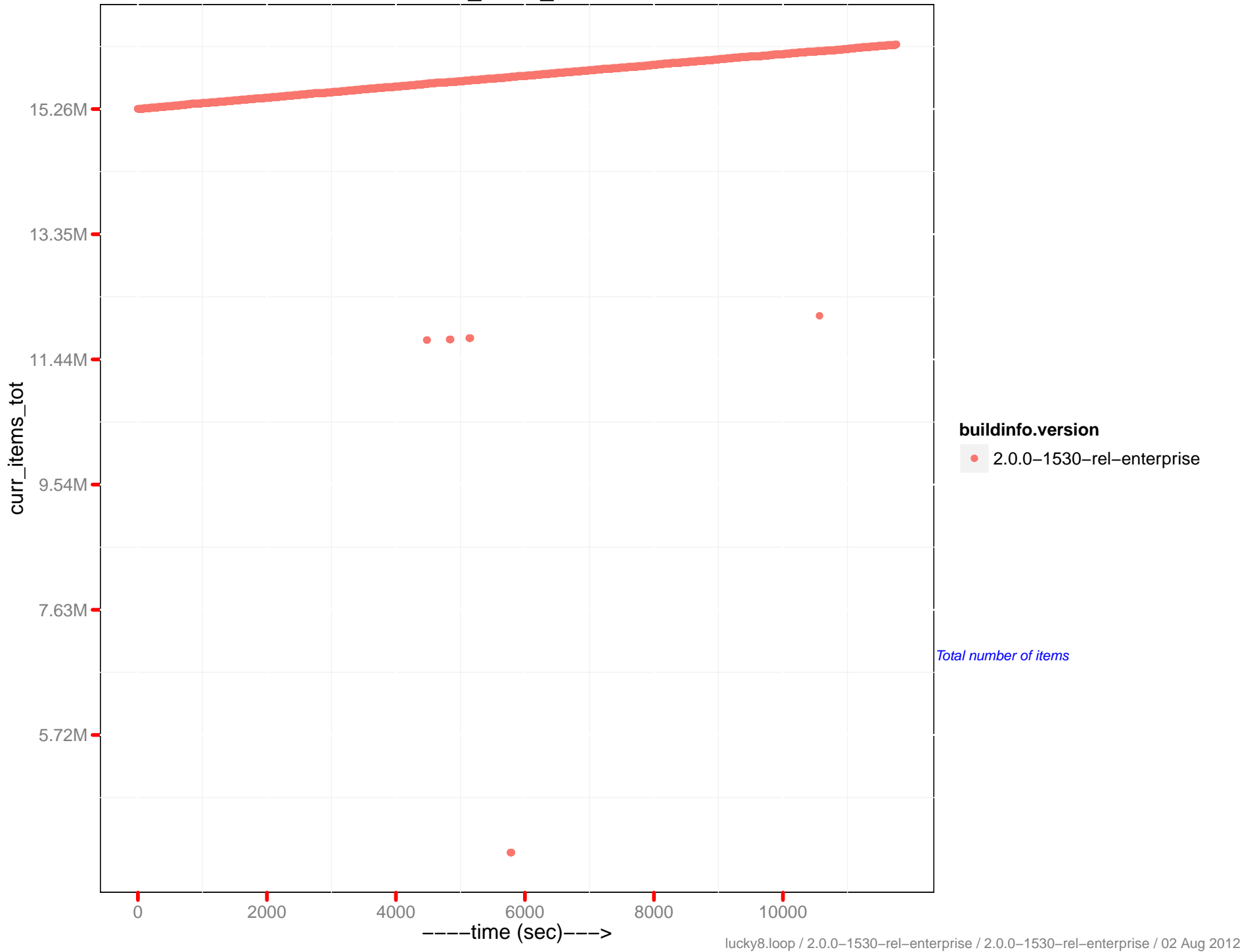
vb_replica_resident_items_ratio



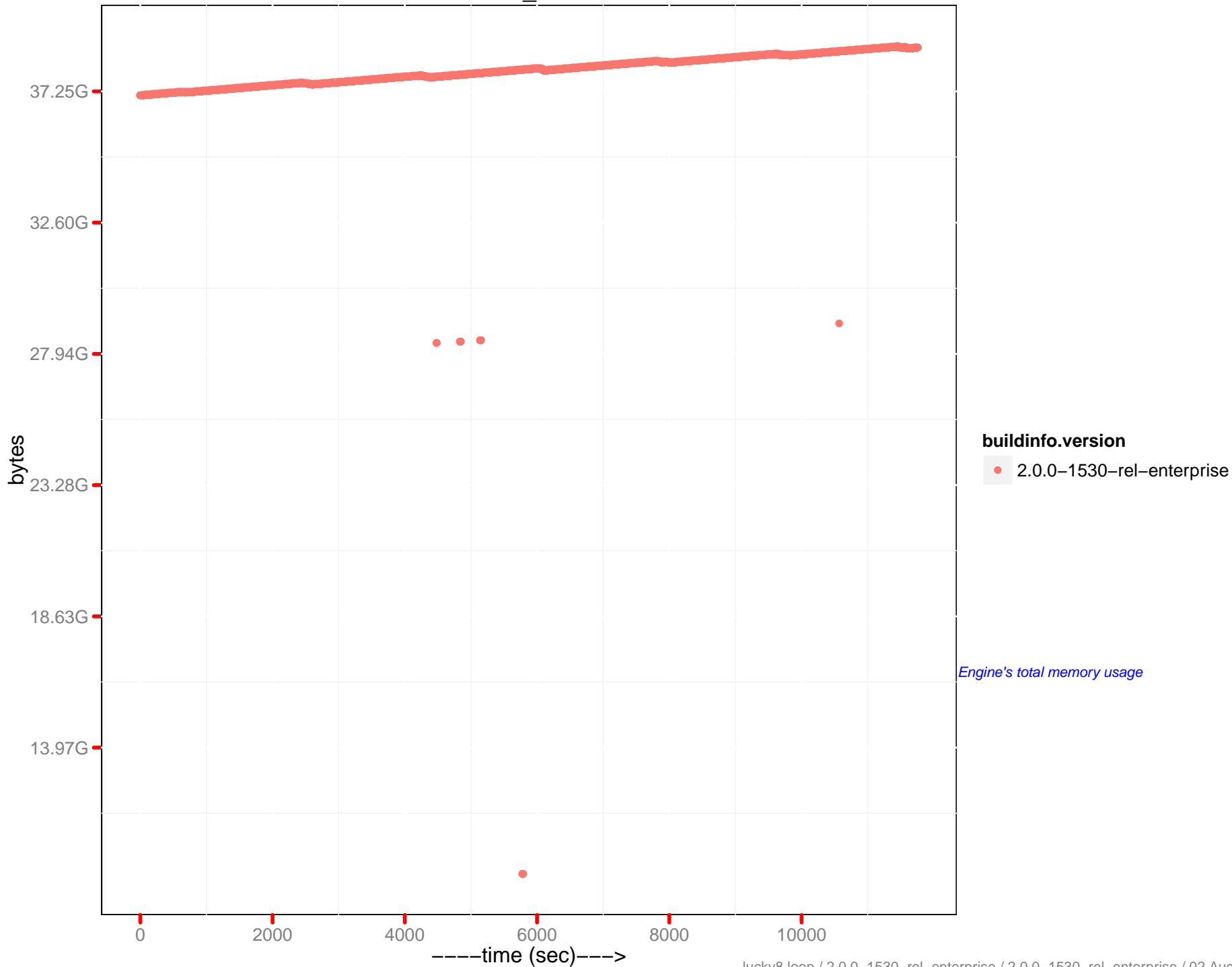
curr_items



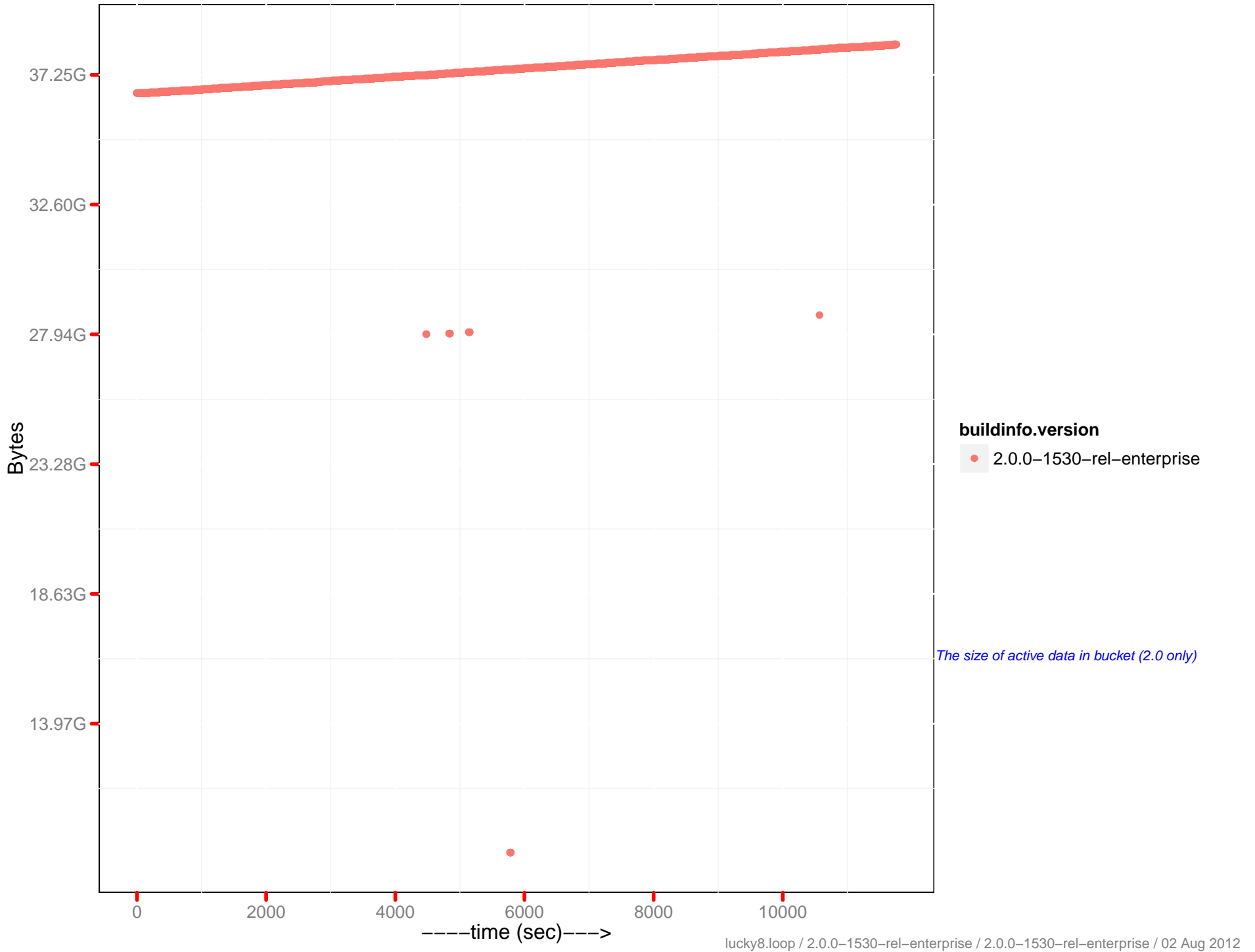
cur_items_total



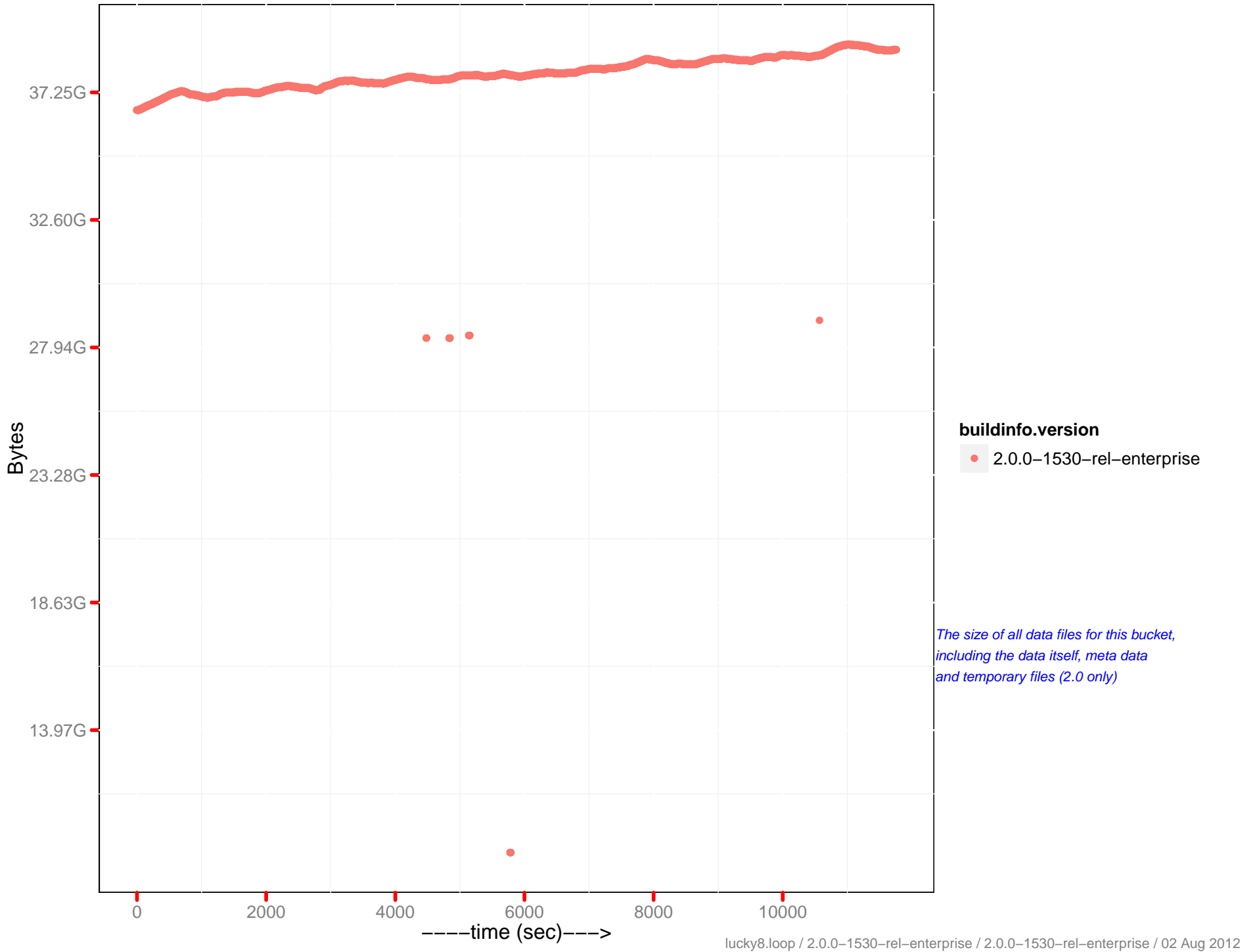
mem_used



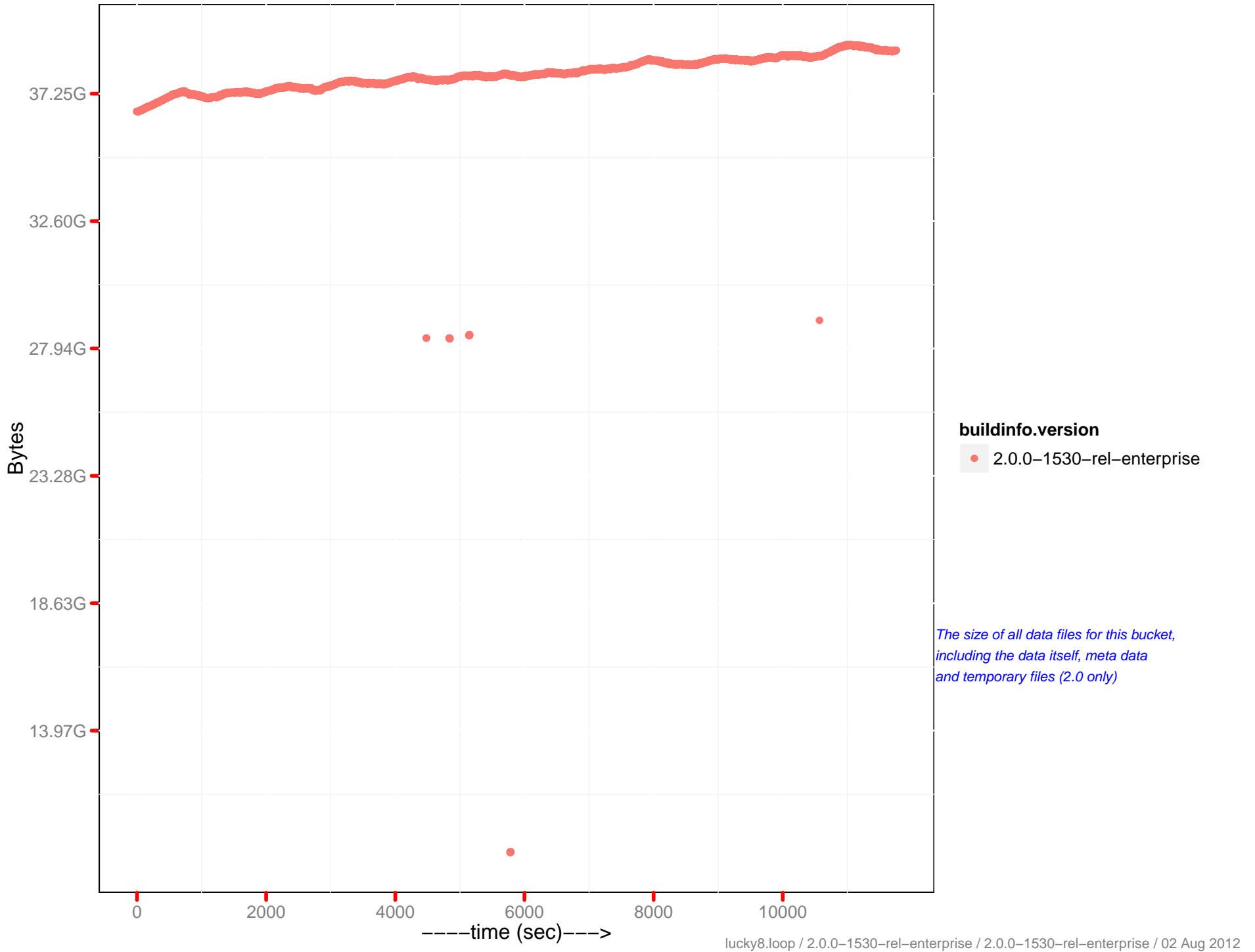
Docs data size



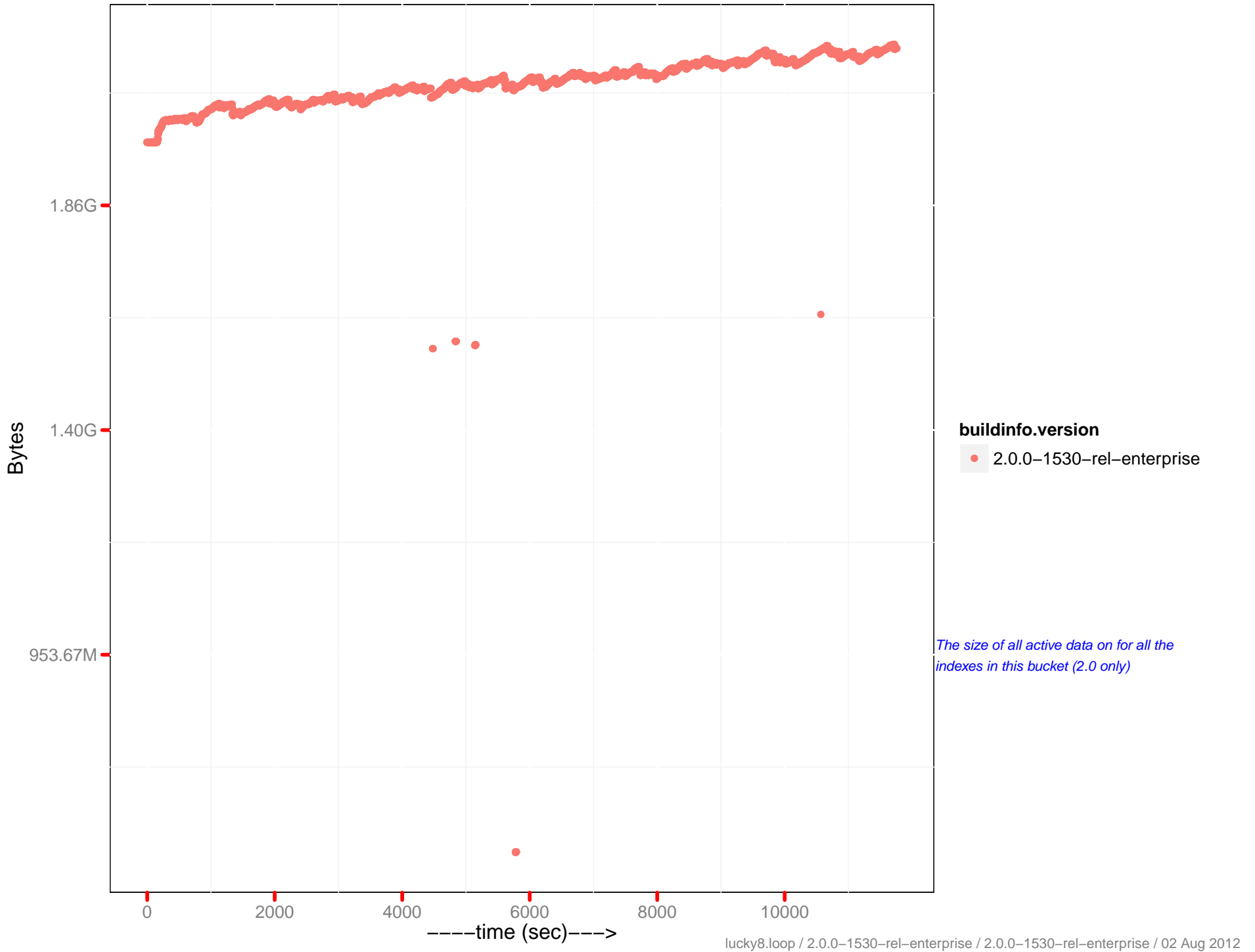
Docs disk size



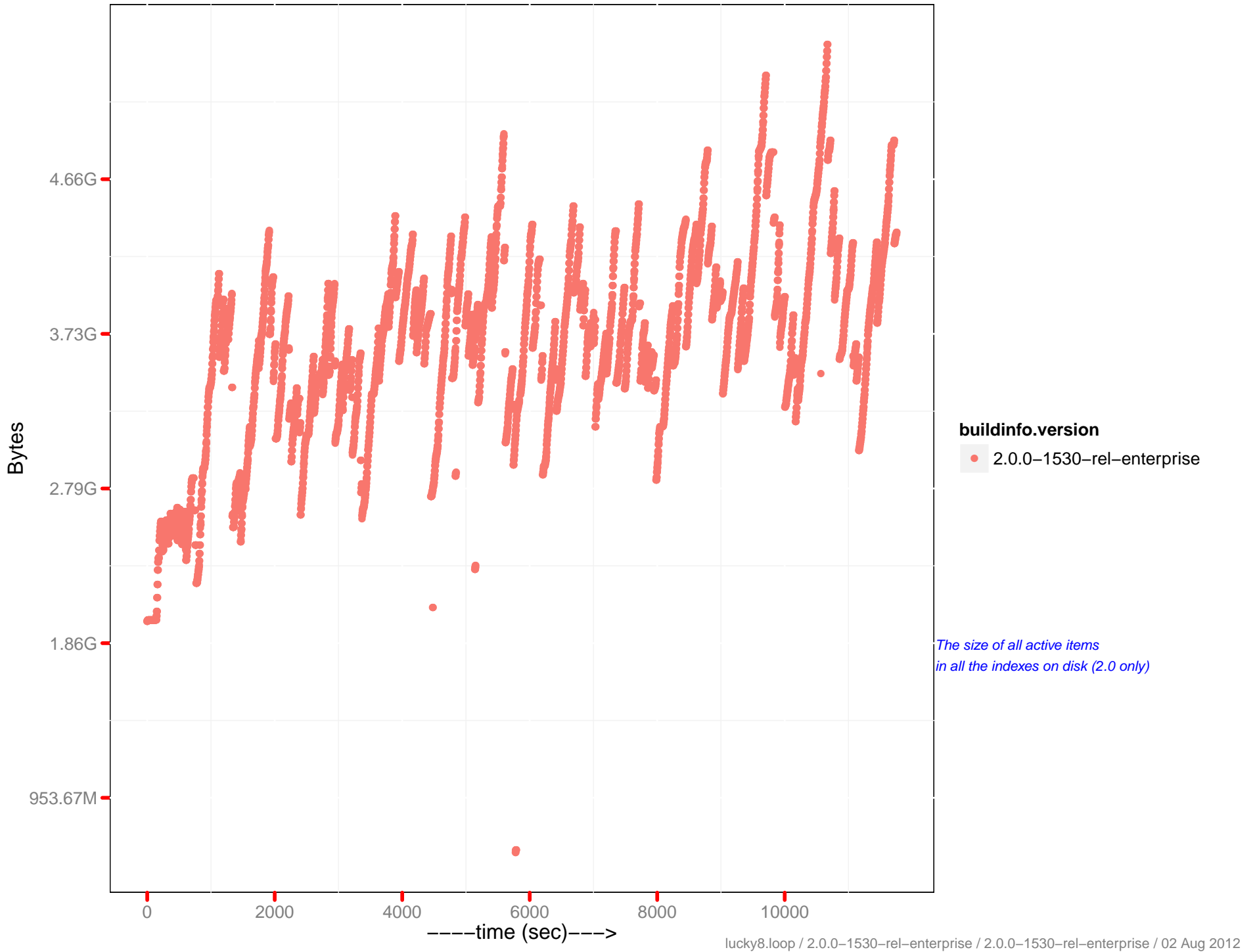
Docs actual disk size



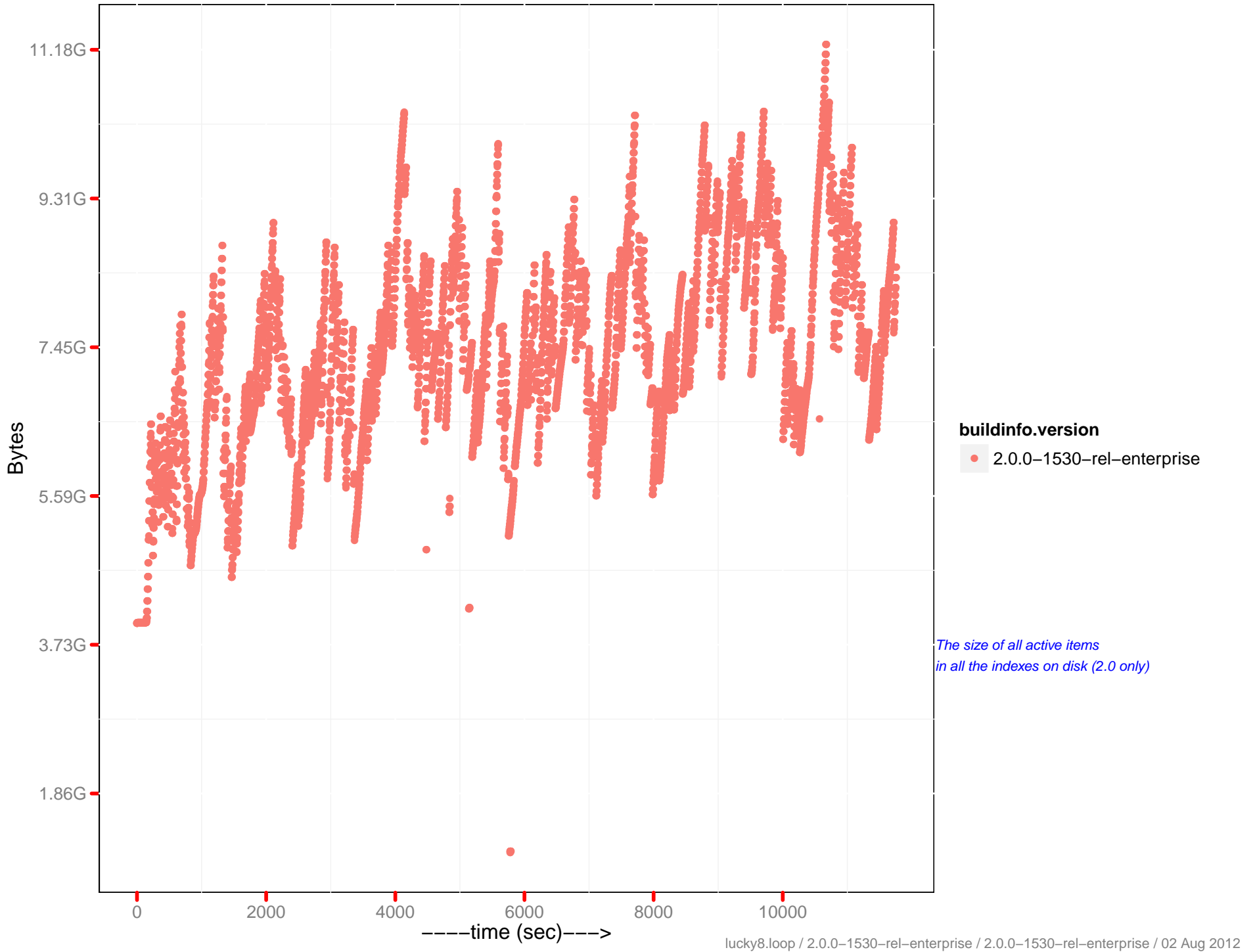
Views data size



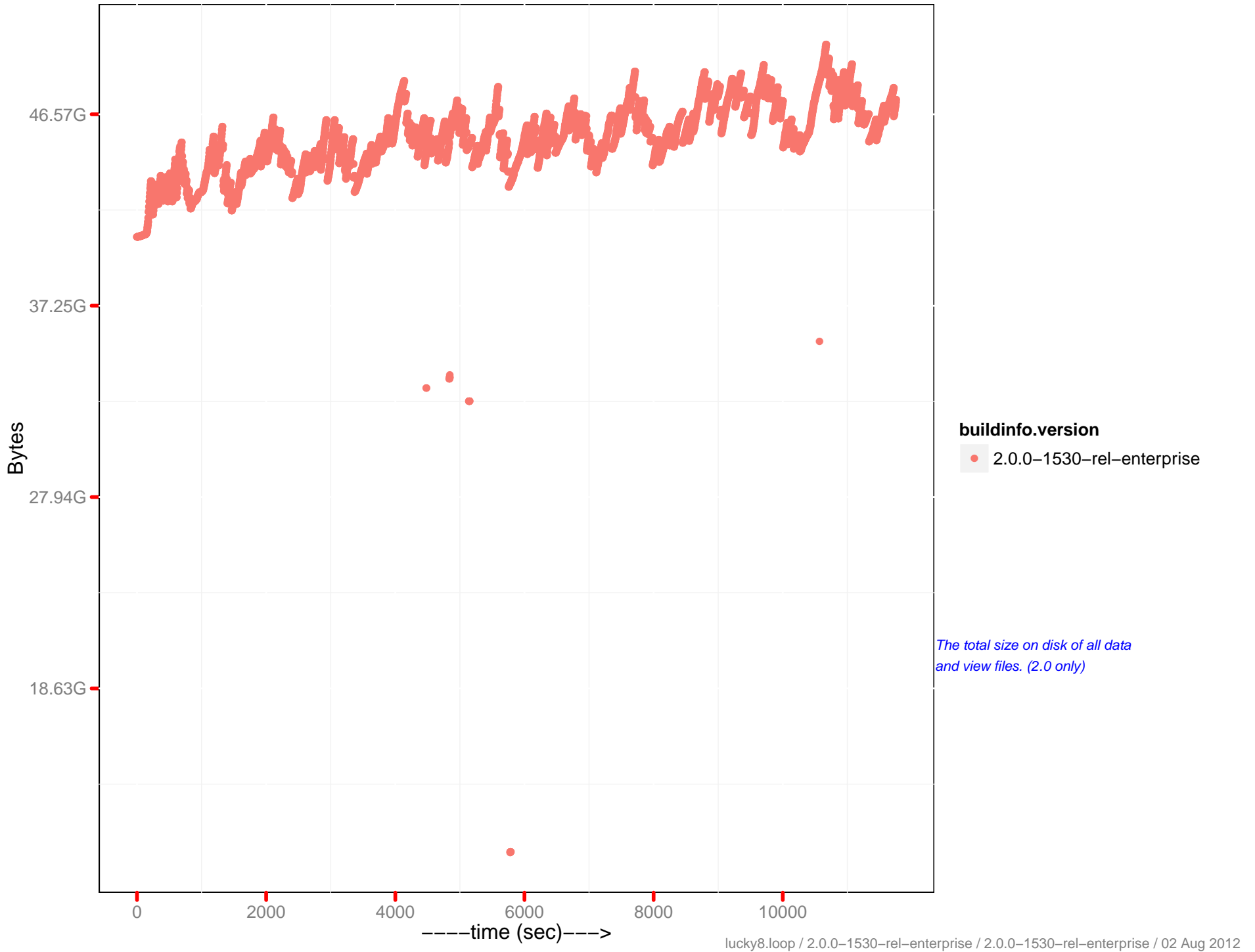
Views disk size



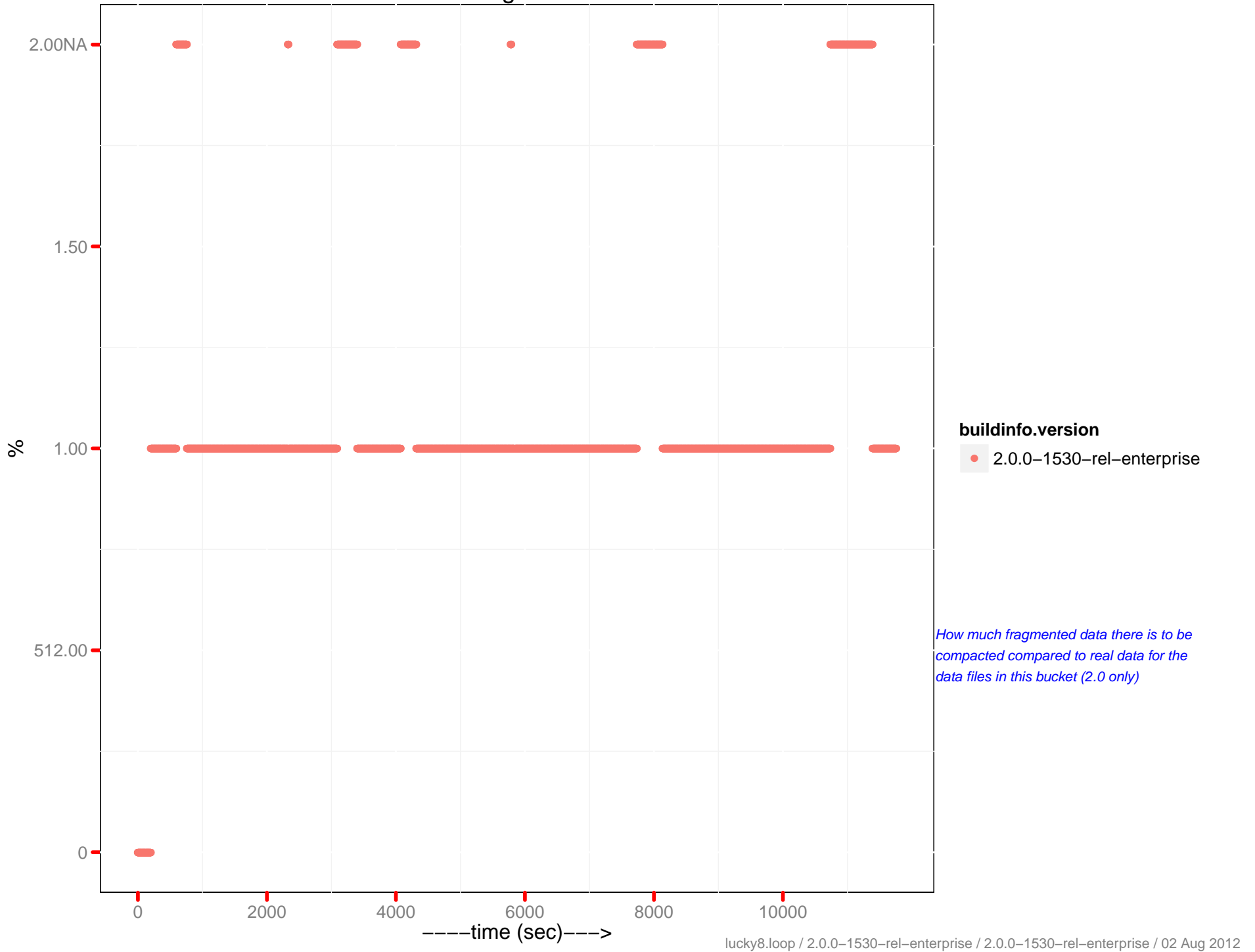
Views actual disk size



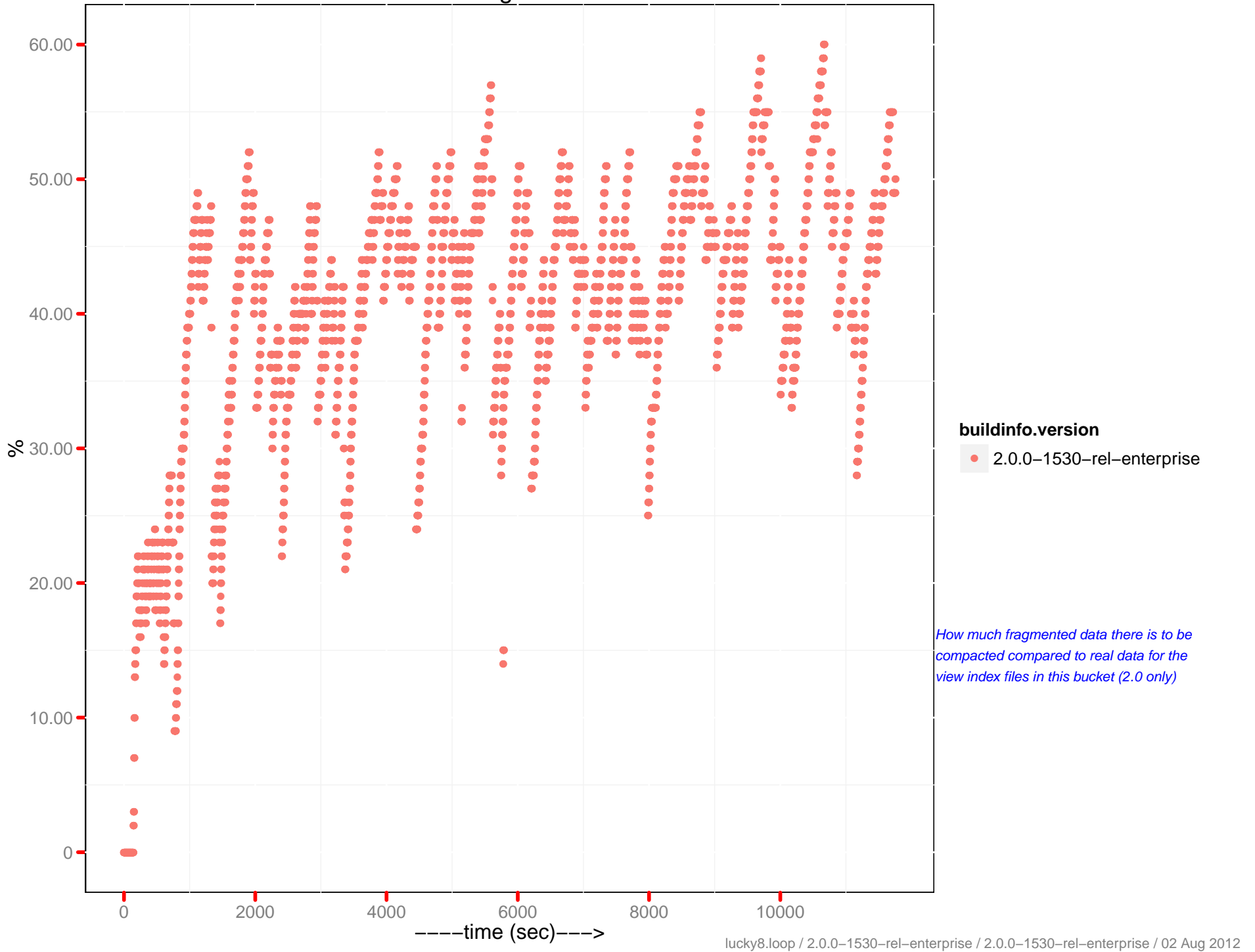
Total disk size



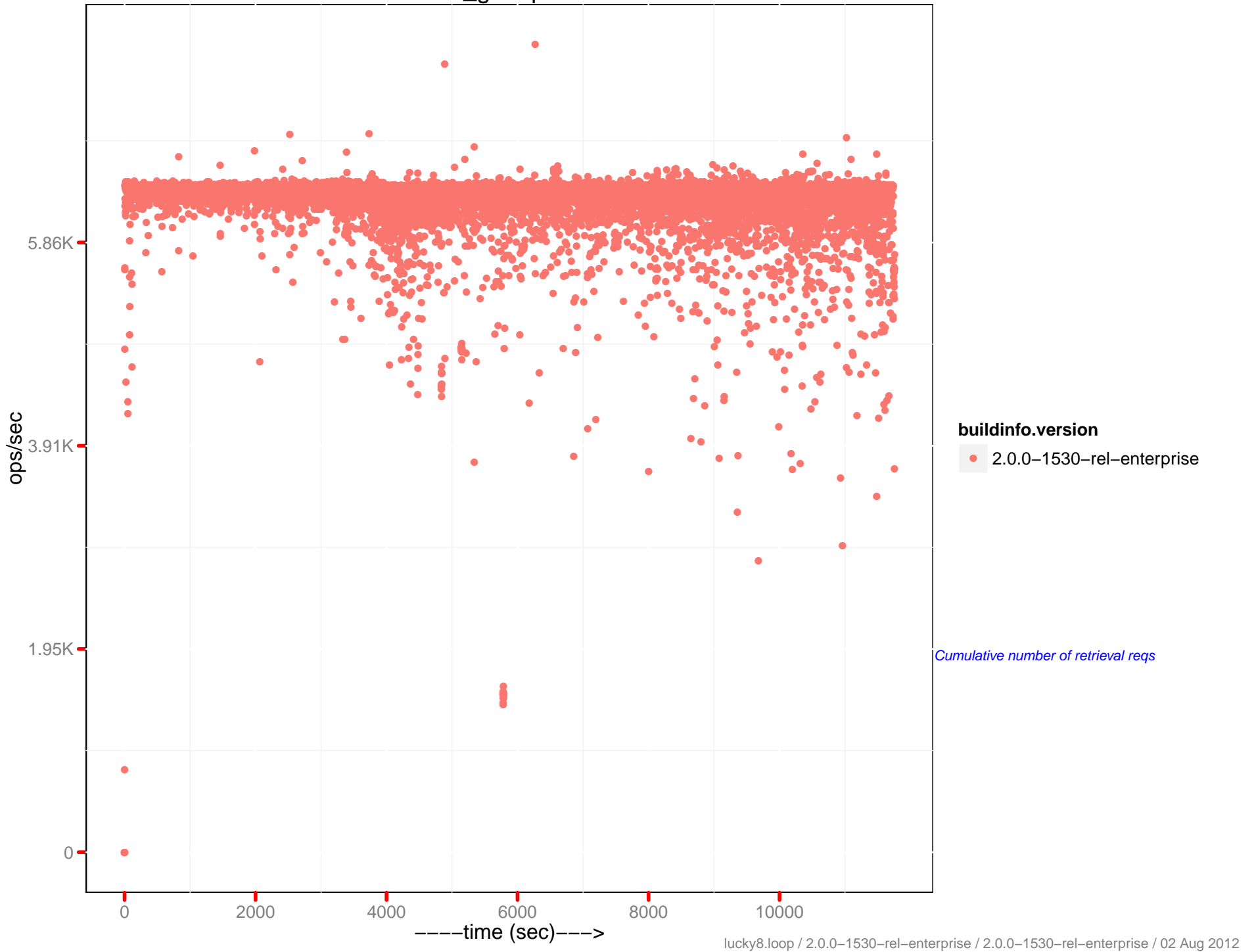
Docs fragmentation



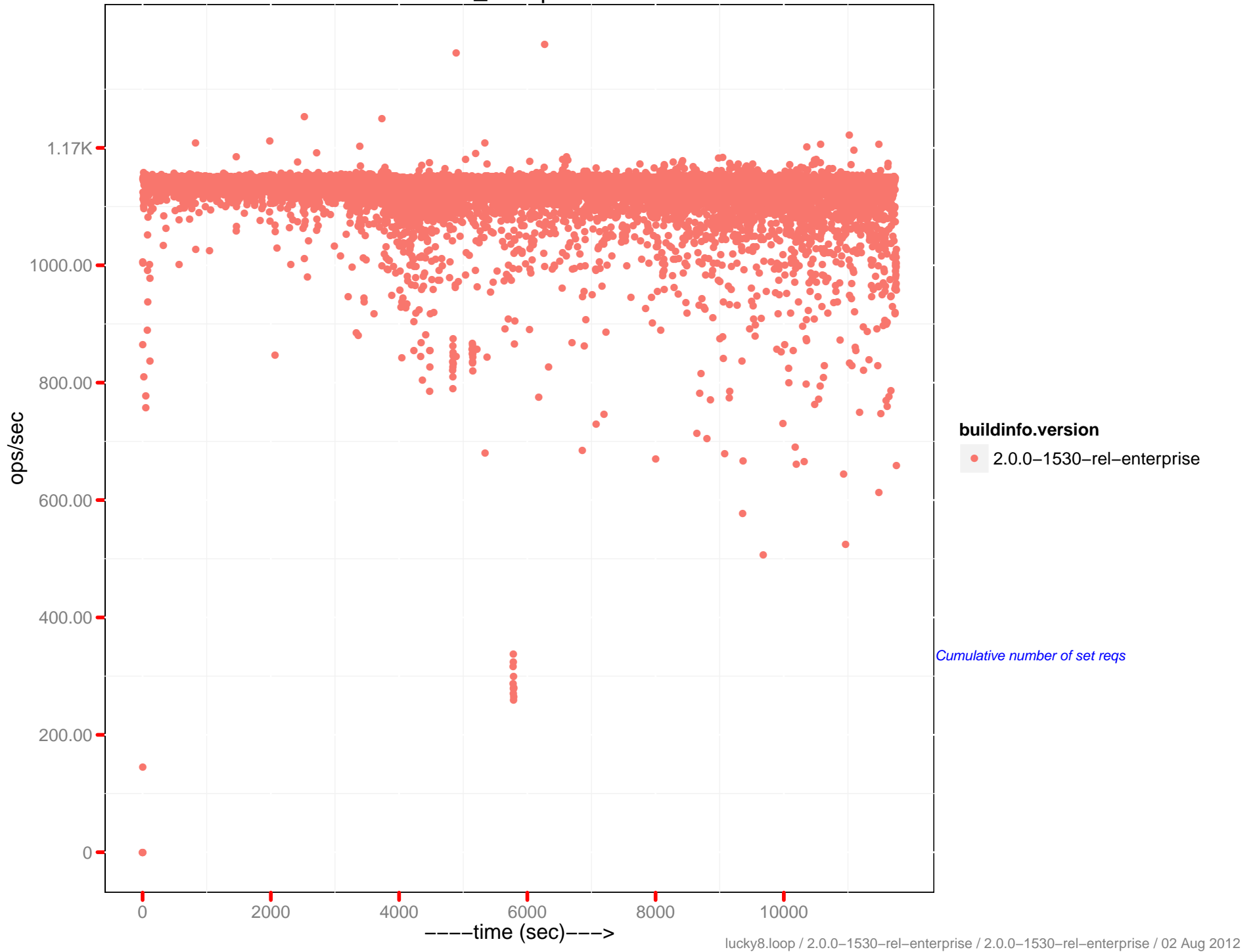
Views fragmentation



cmd_get ops/sec



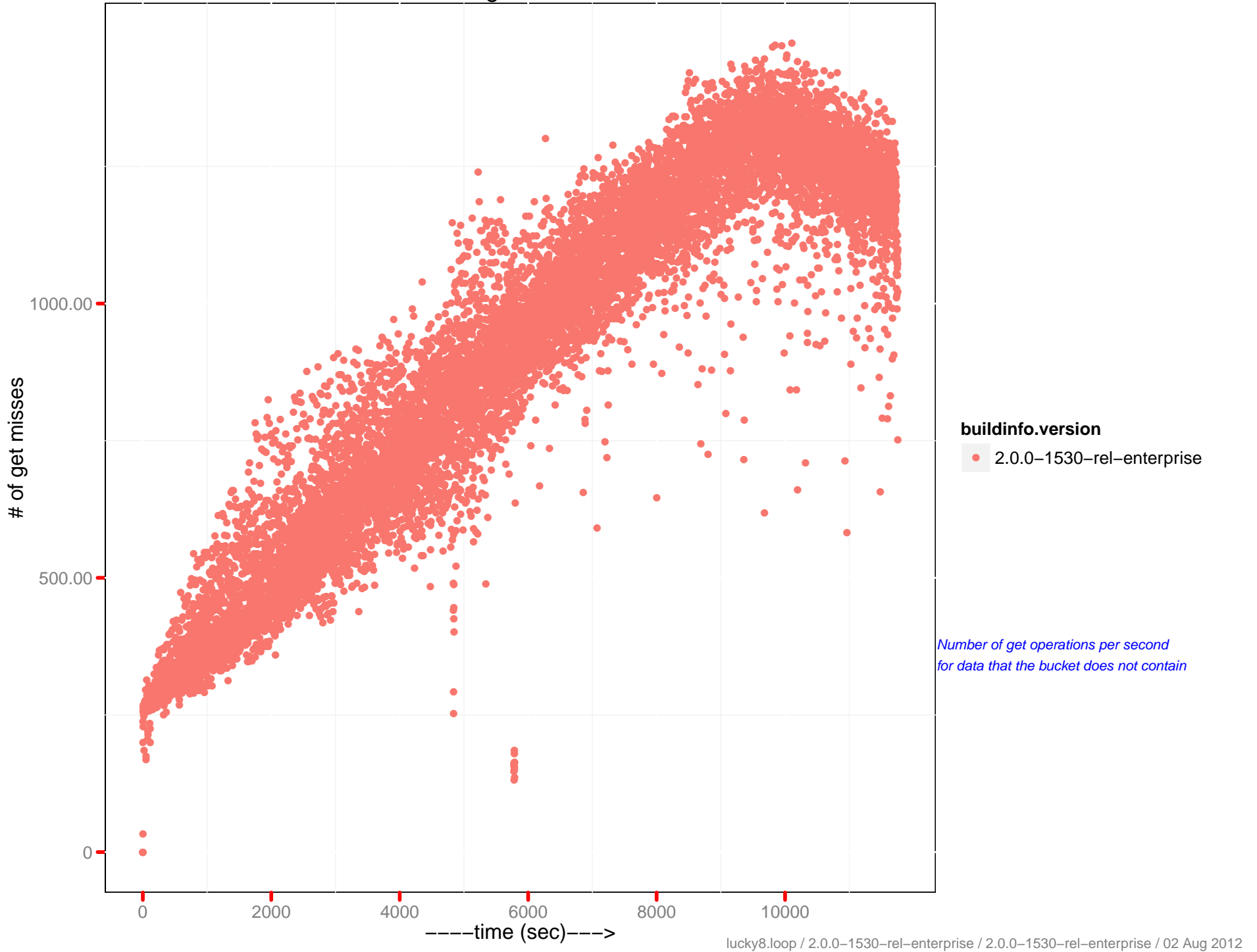
cmd_set ops/sec



buildinfo.version
● 2.0.0-1530-rel-enterprise

Cumulative number of set reqs

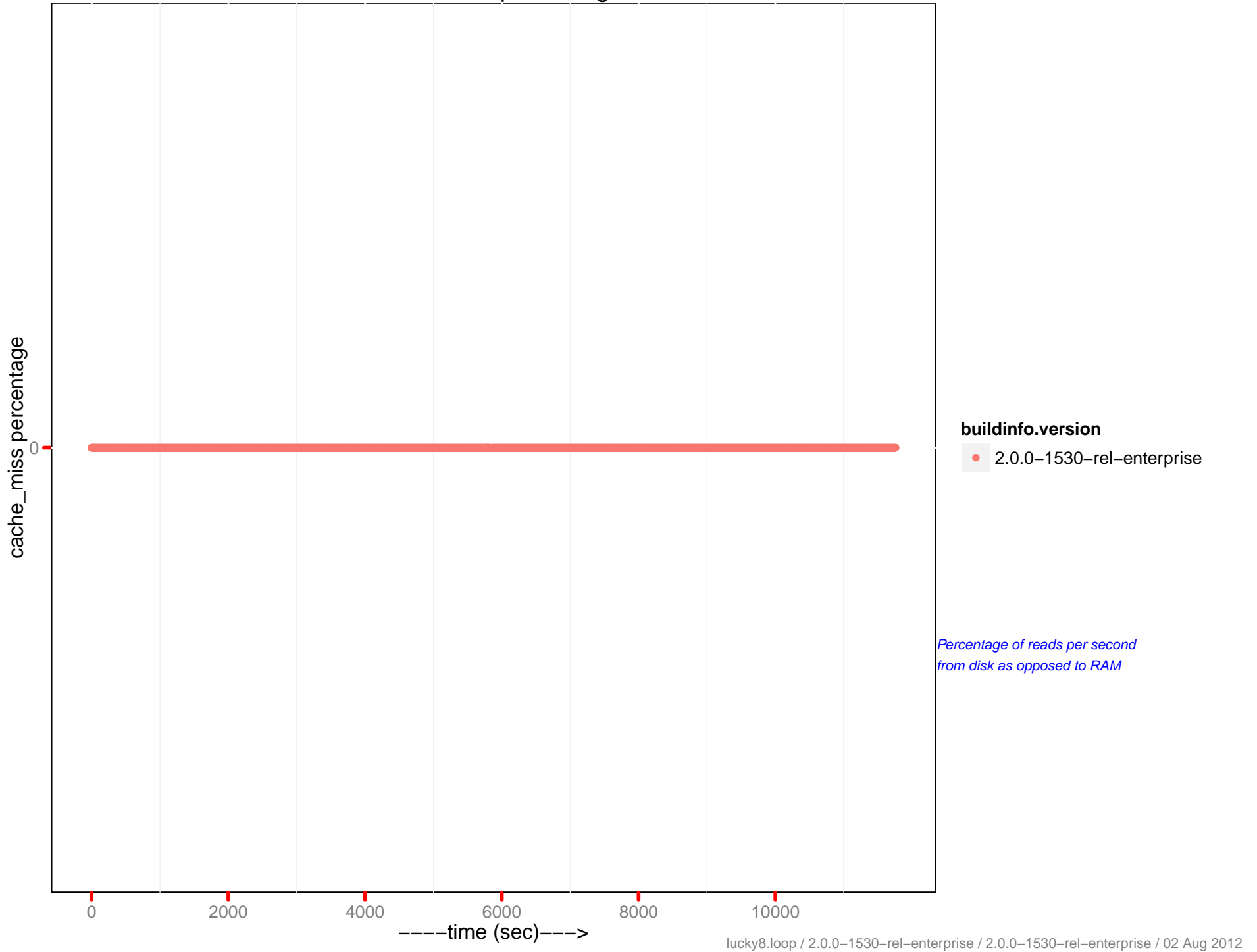
of get misses



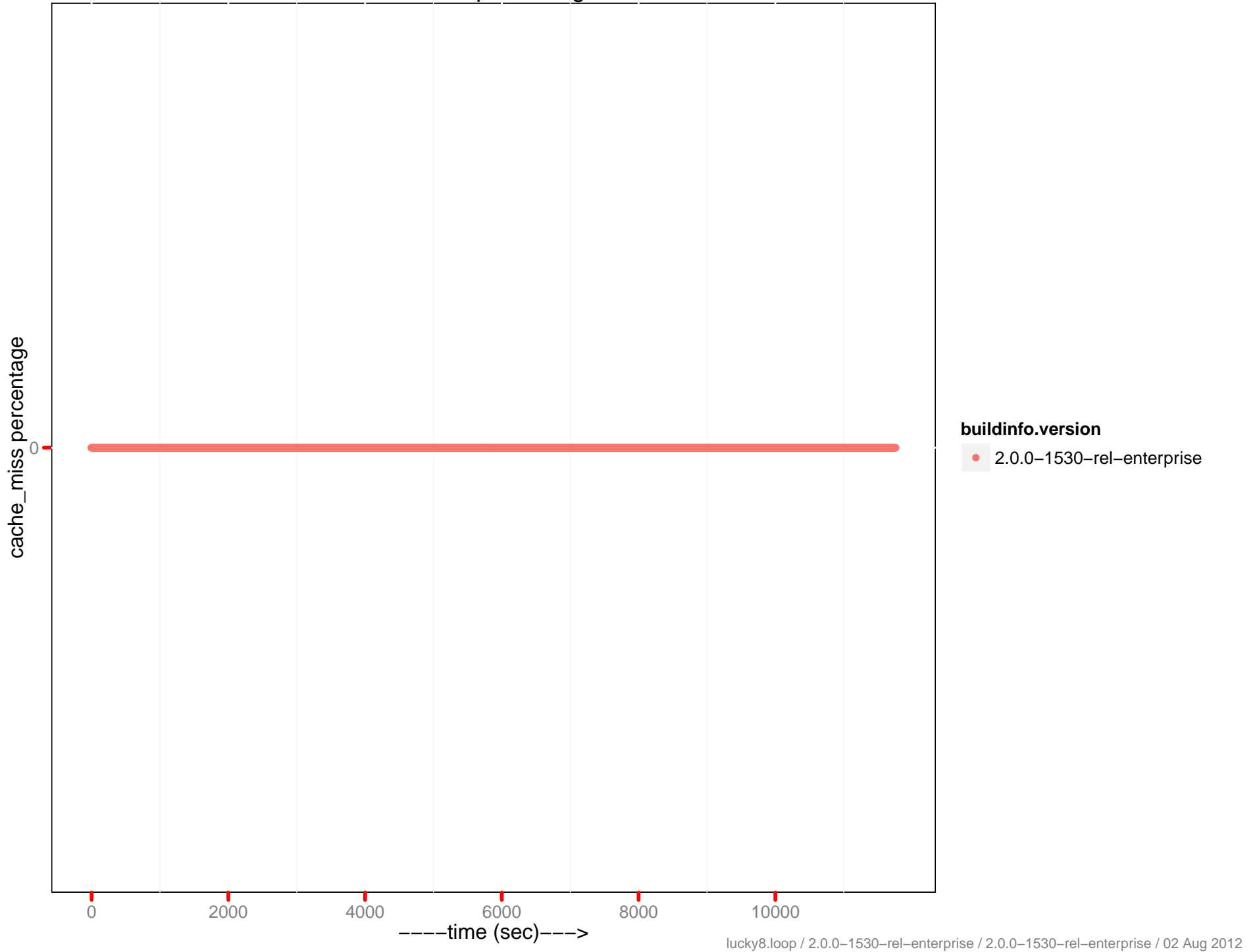
of get hits



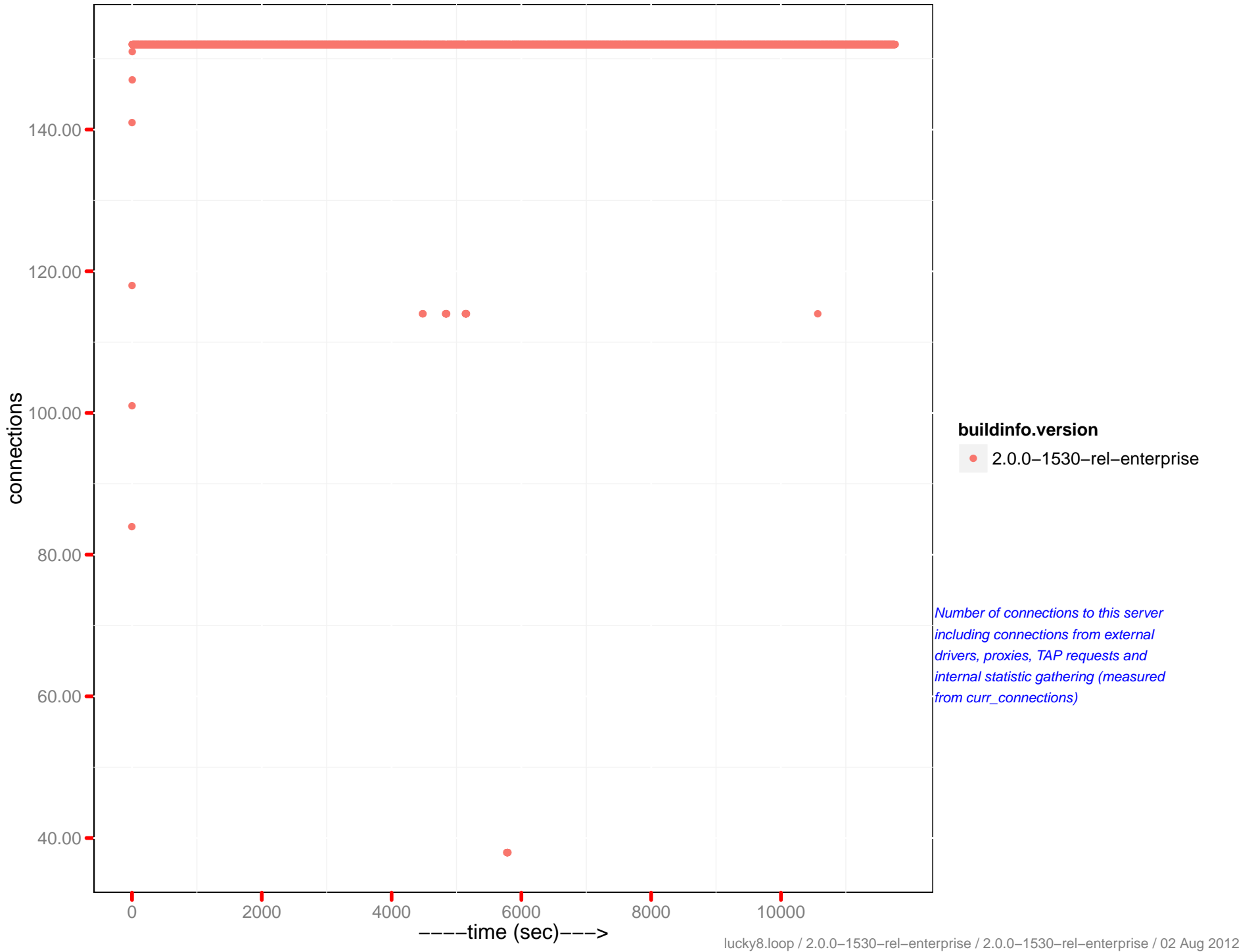
cache_miss percentage



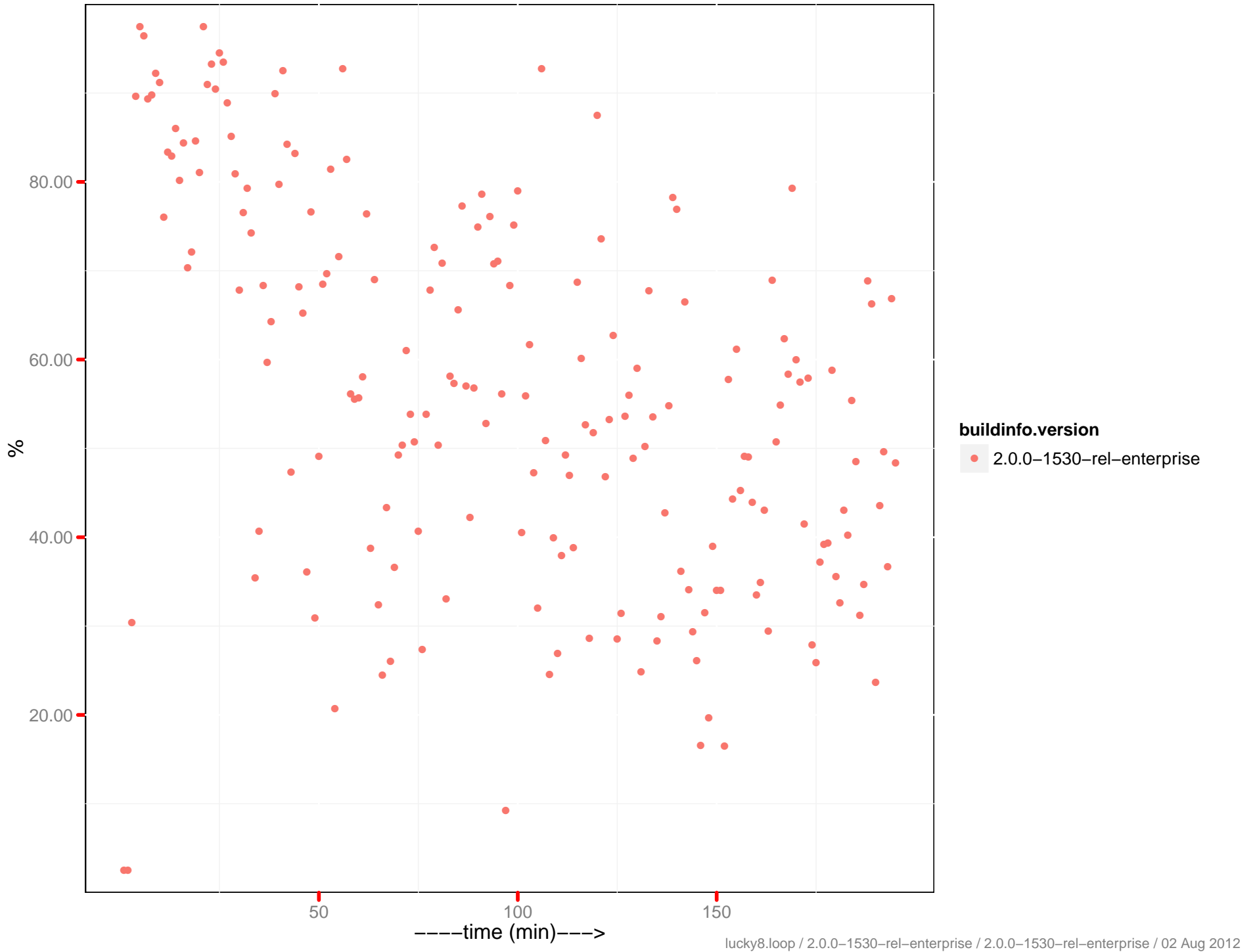
cache_miss percentage 0-5



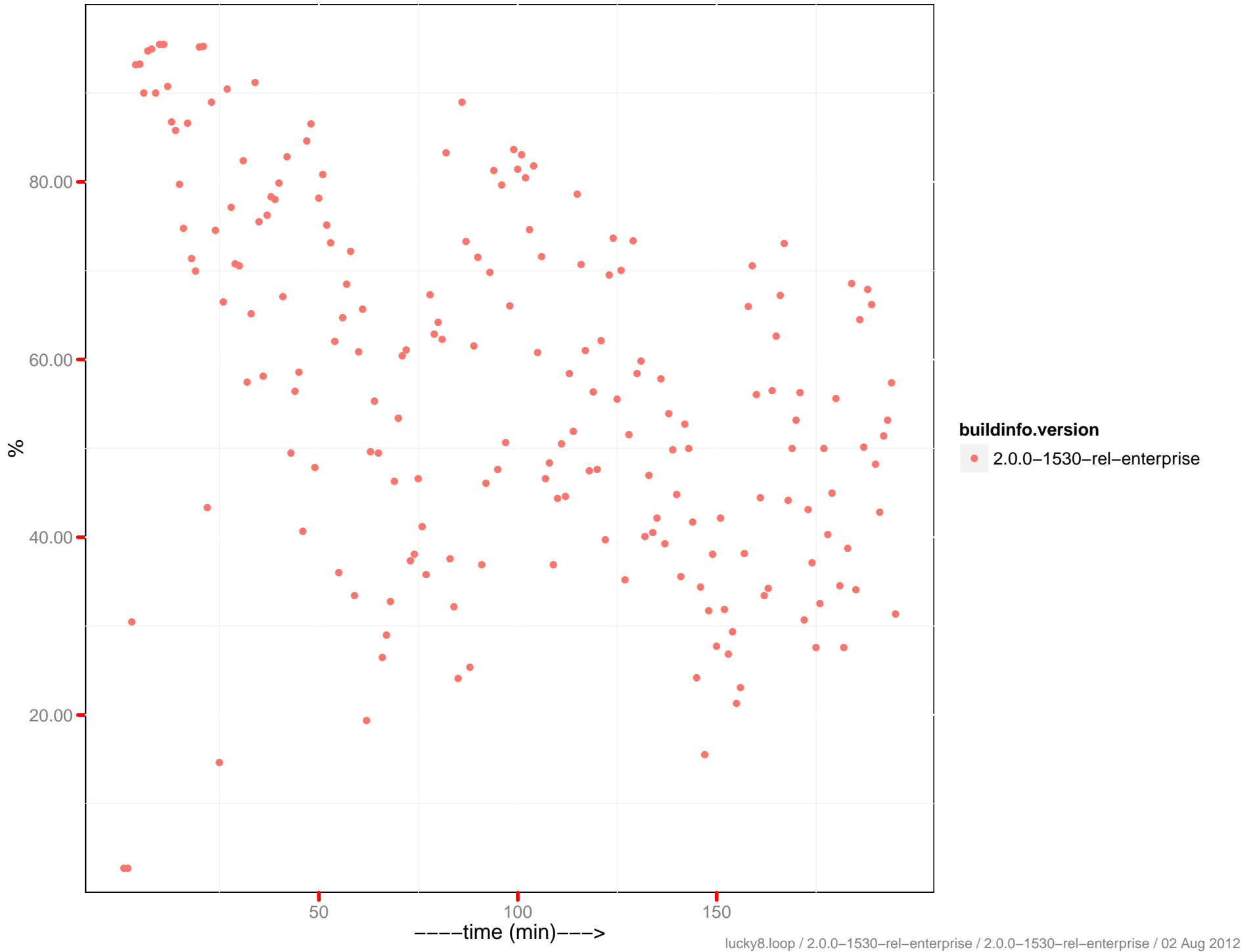
Number of connections



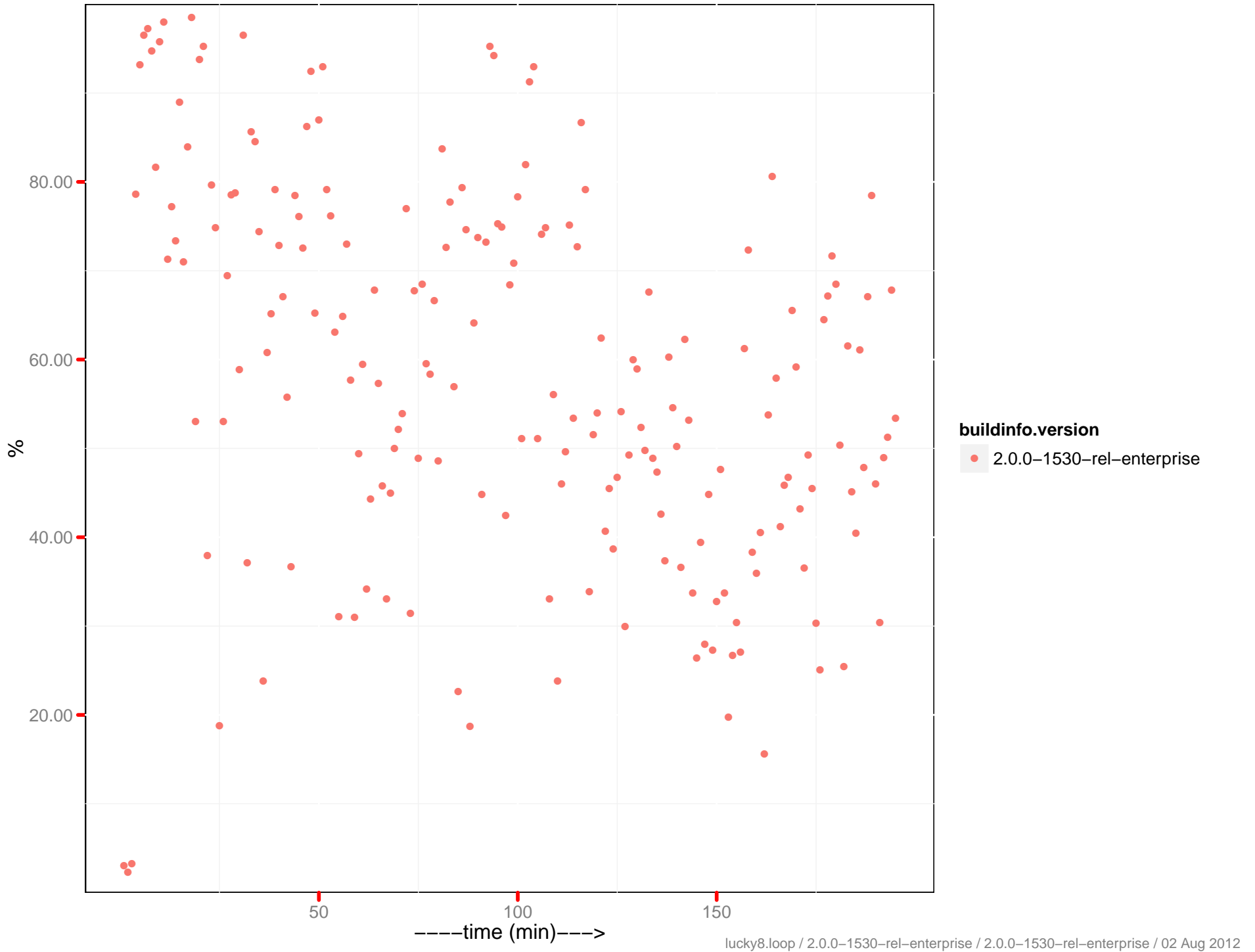
CPU utilization – 10.2.1.61:8091



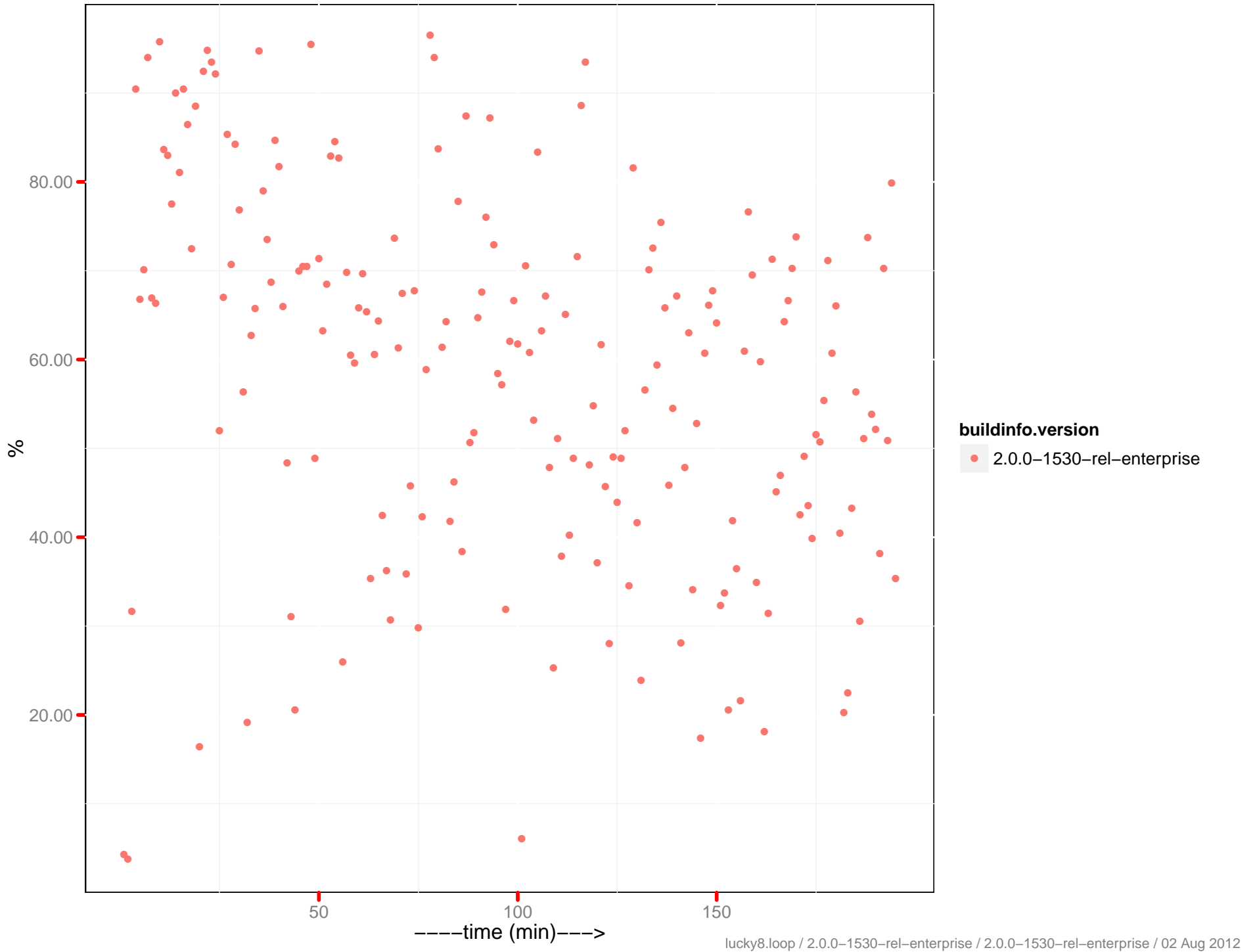
CPU utilization – 10.2.1.62:8091



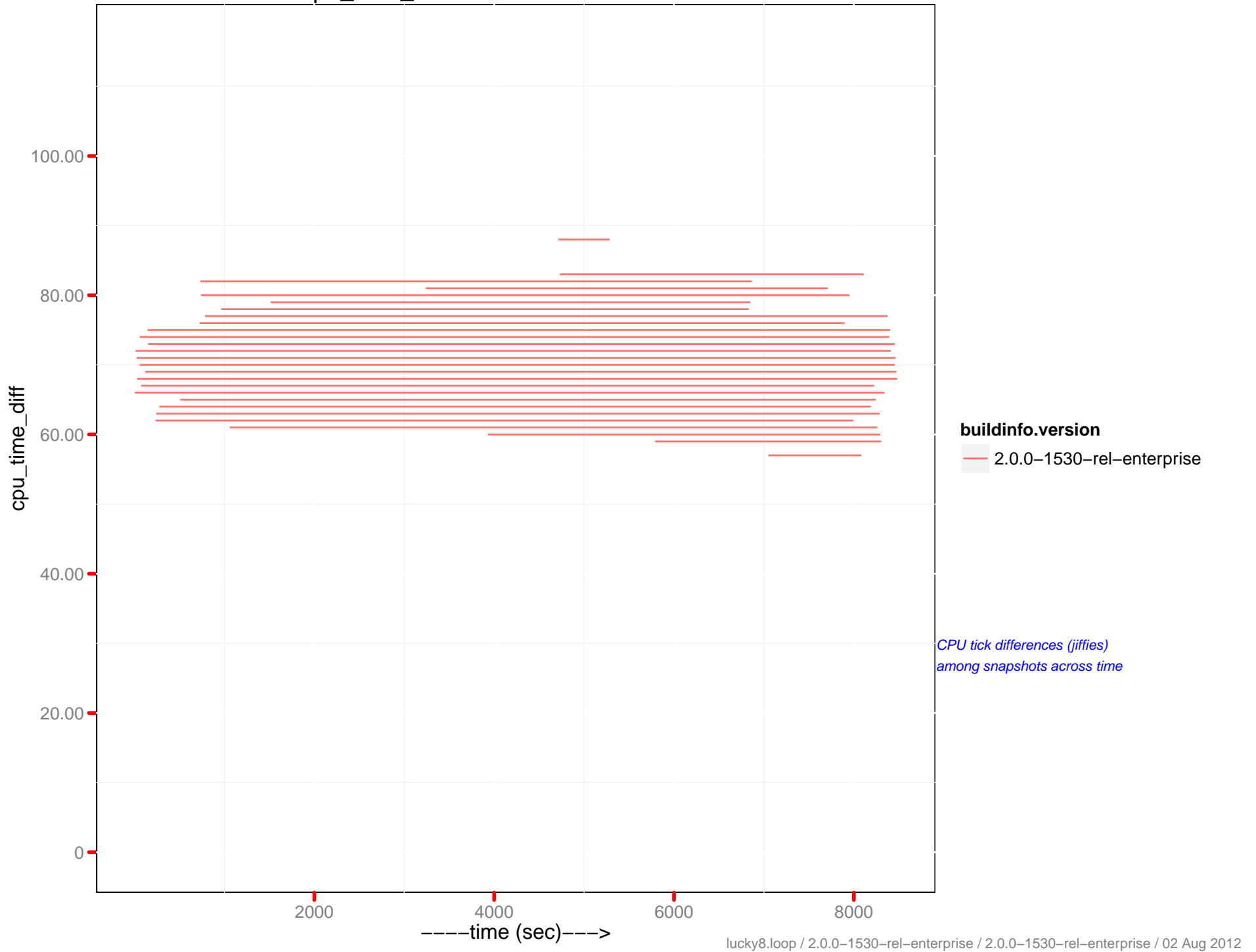
CPU utilization – 10.2.1.63:8091



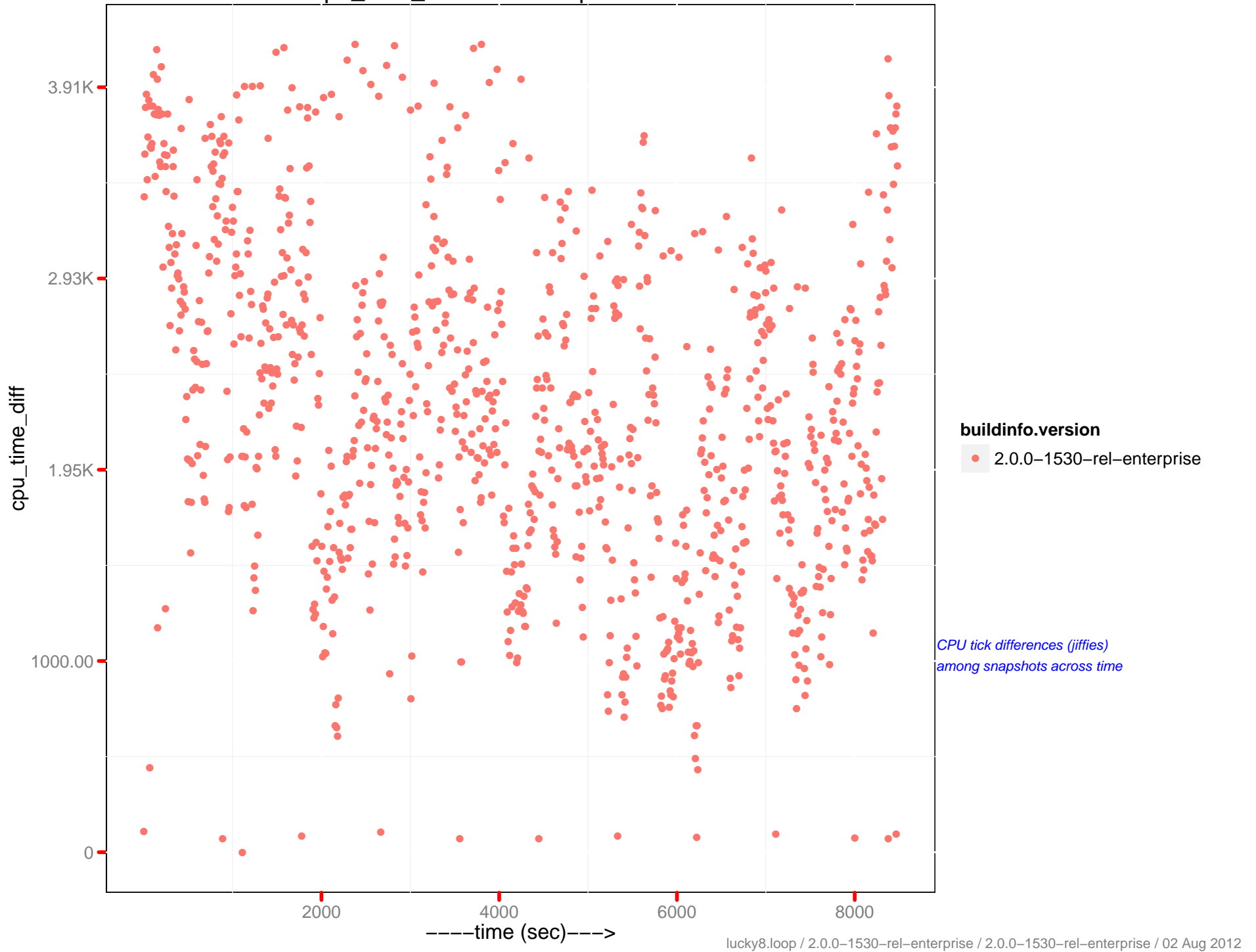
CPU utilization – 10.2.1.64:8091



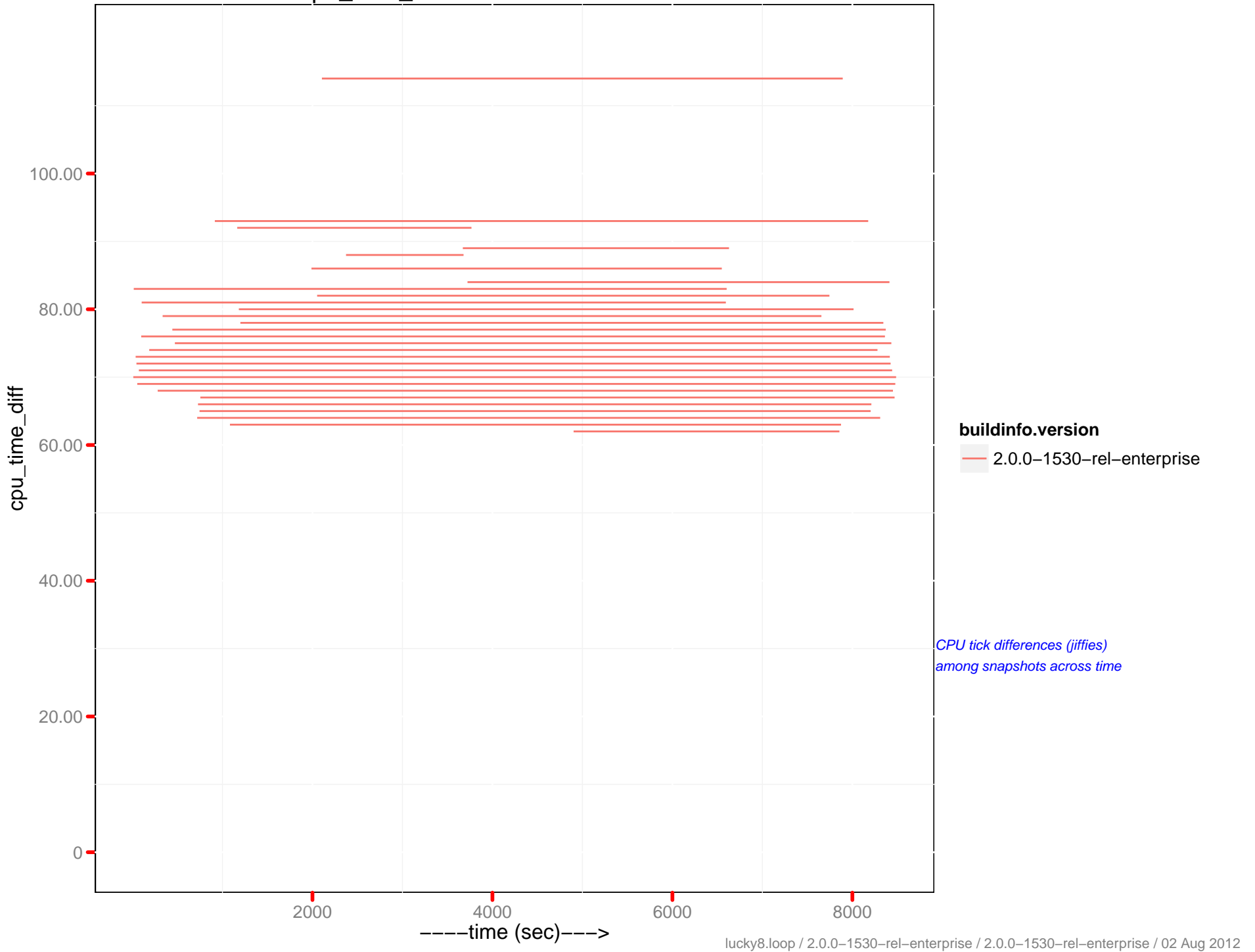
cpu_time_diff: memcached – 10.2.1.61



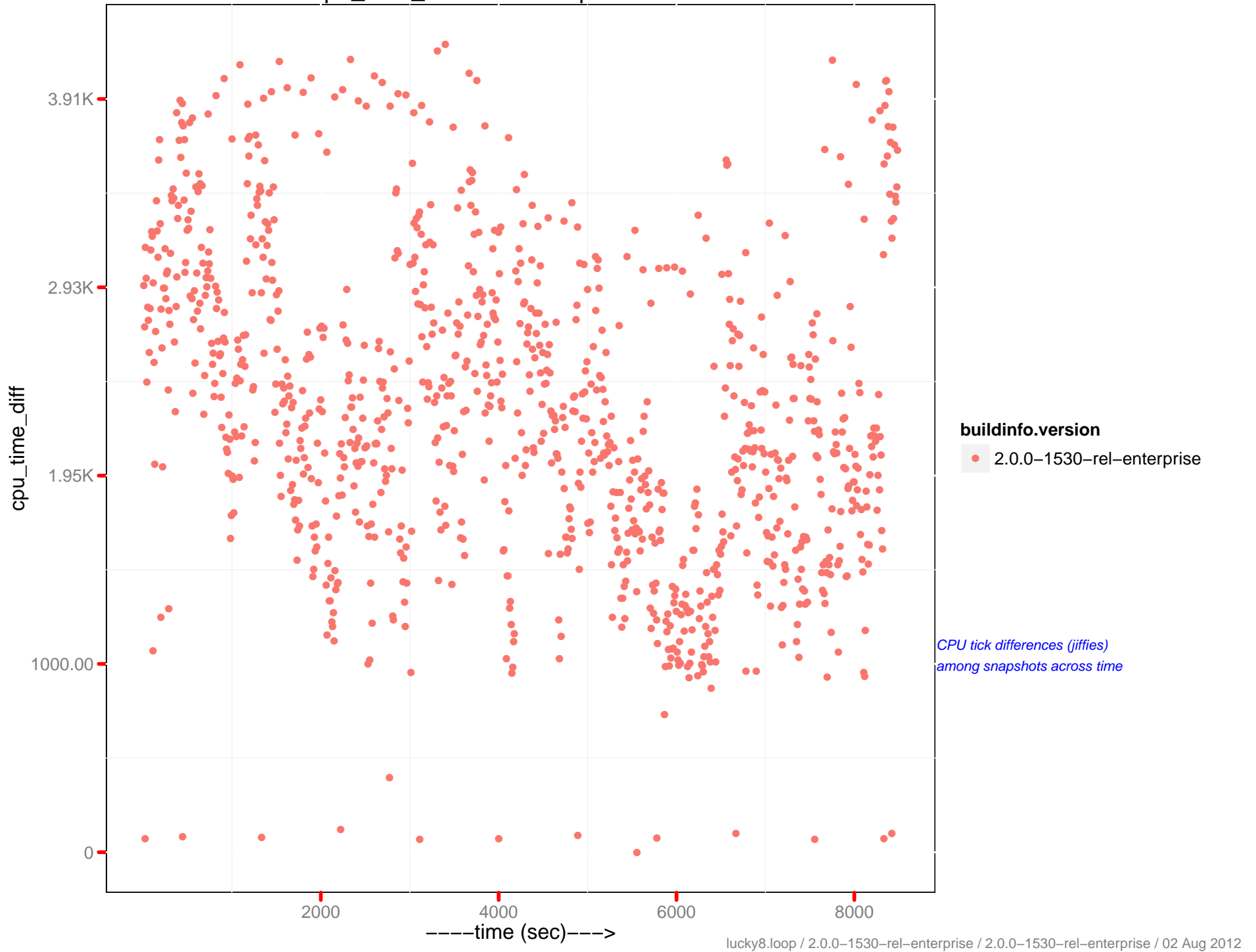
cpu_time_diff : beam.smp - 10.2.1.61



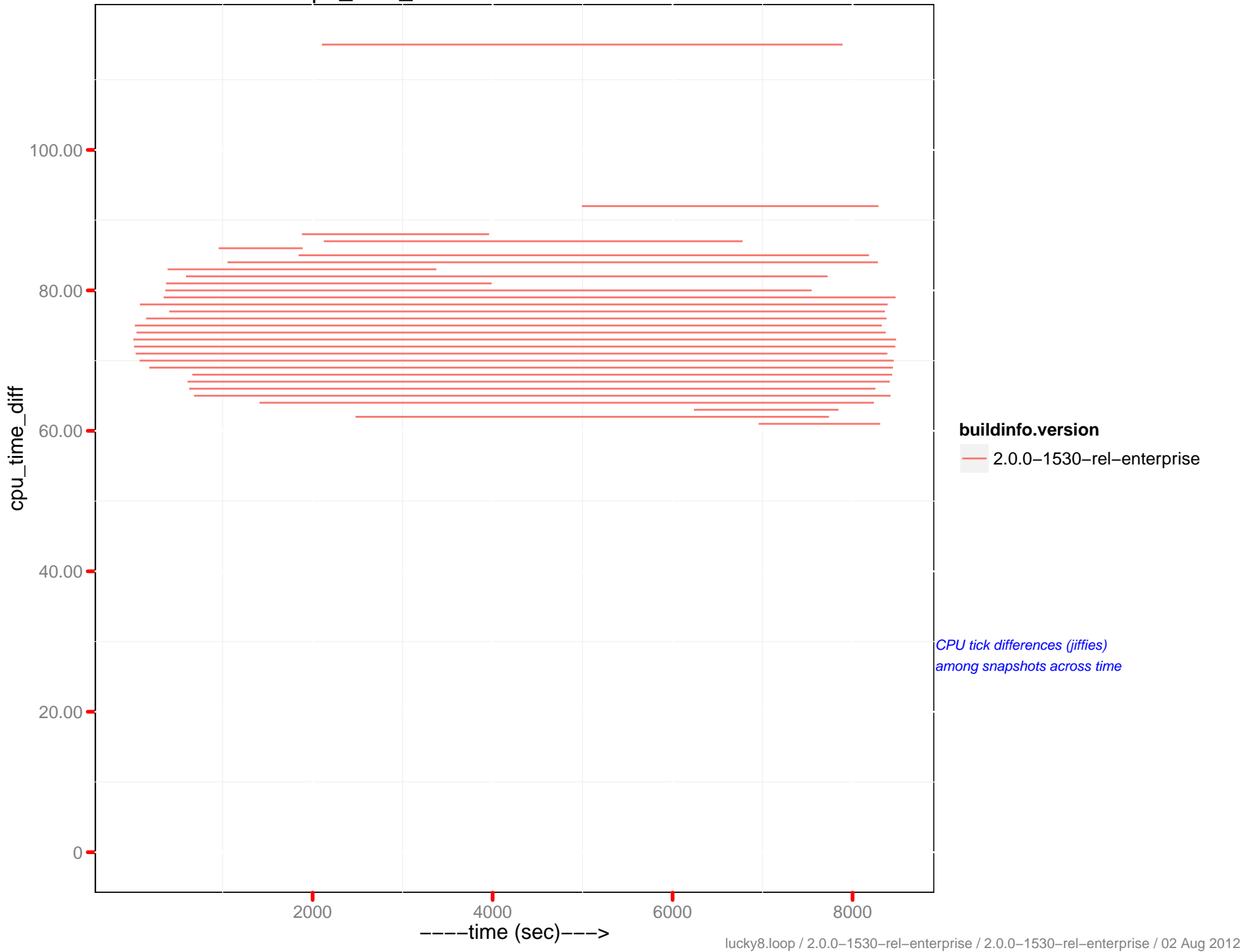
cpu_time_diff: memcached - 10.2.1.62



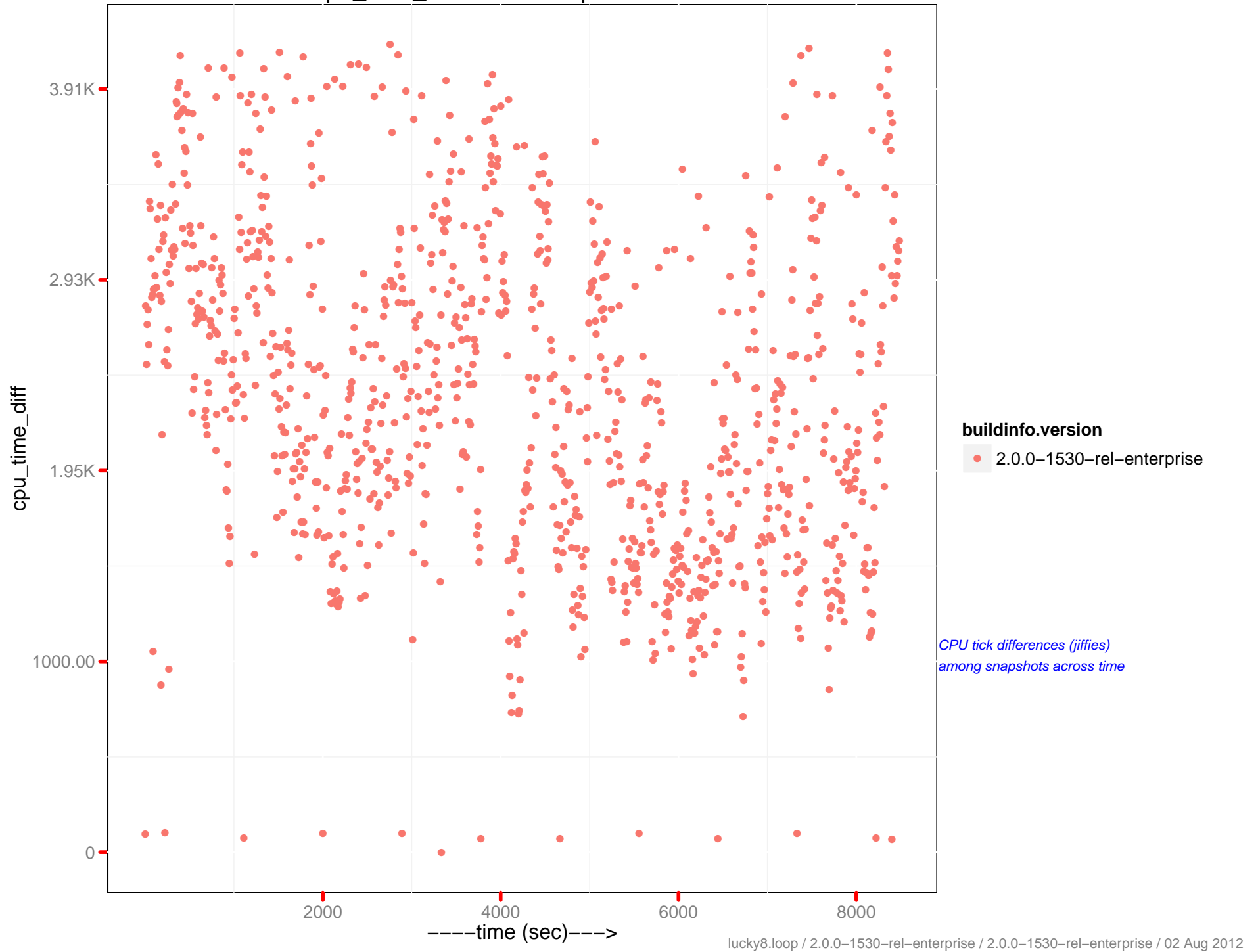
cpu_time_diff : beam.smp - 10.2.1.62



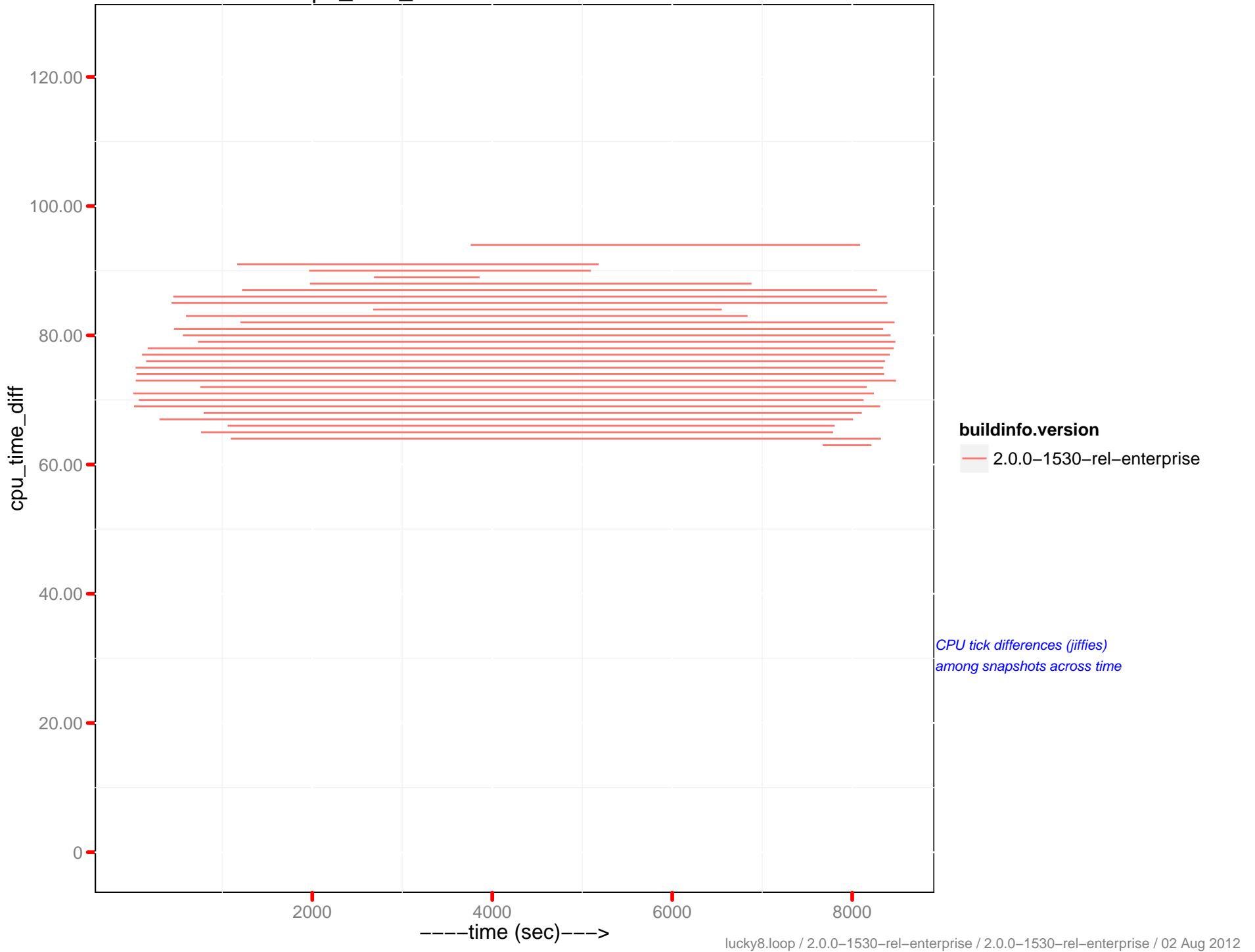
cpu_time_diff: memcached - 10.2.1.63



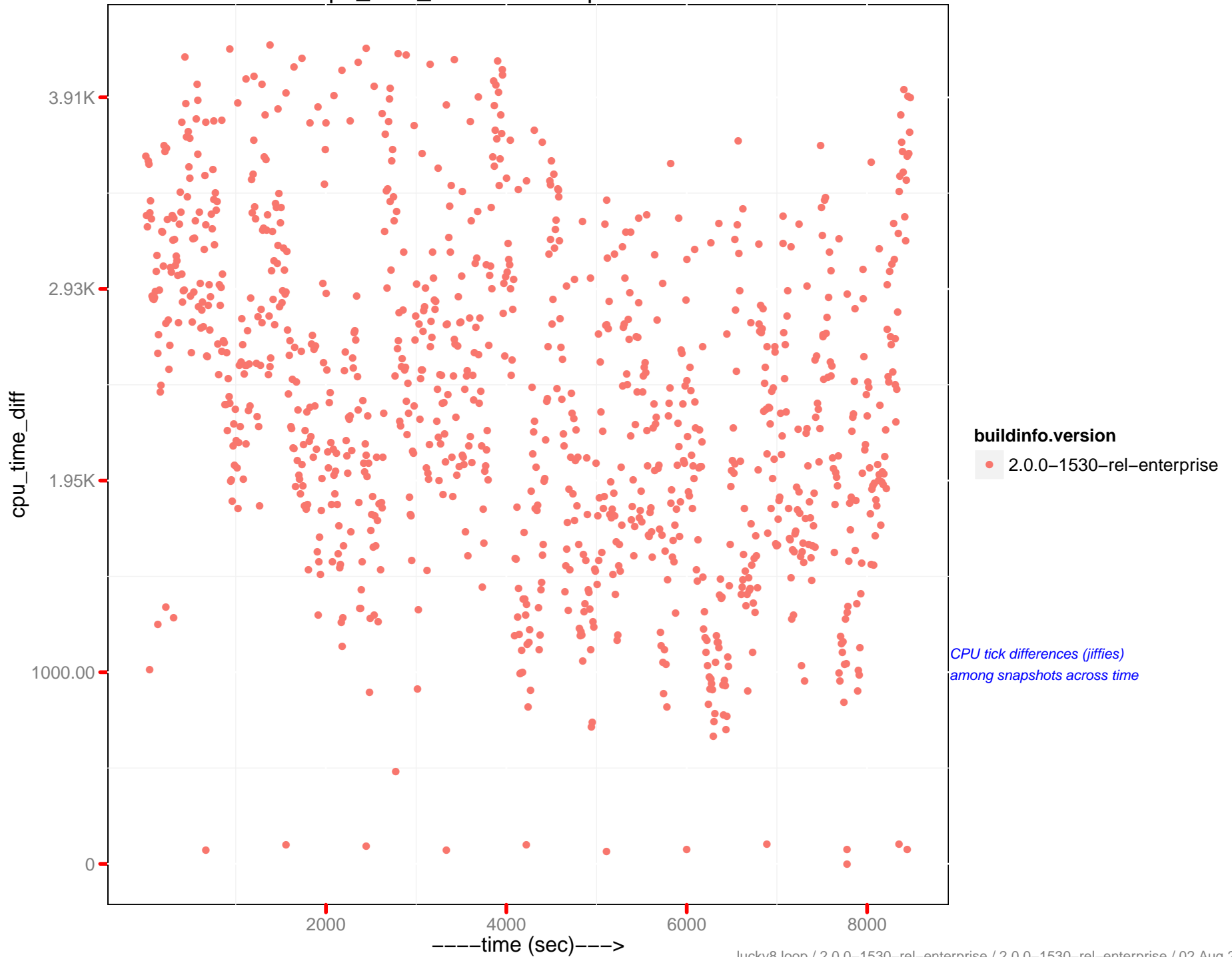
cpu_time_diff : beam.smp - 10.2.1.63



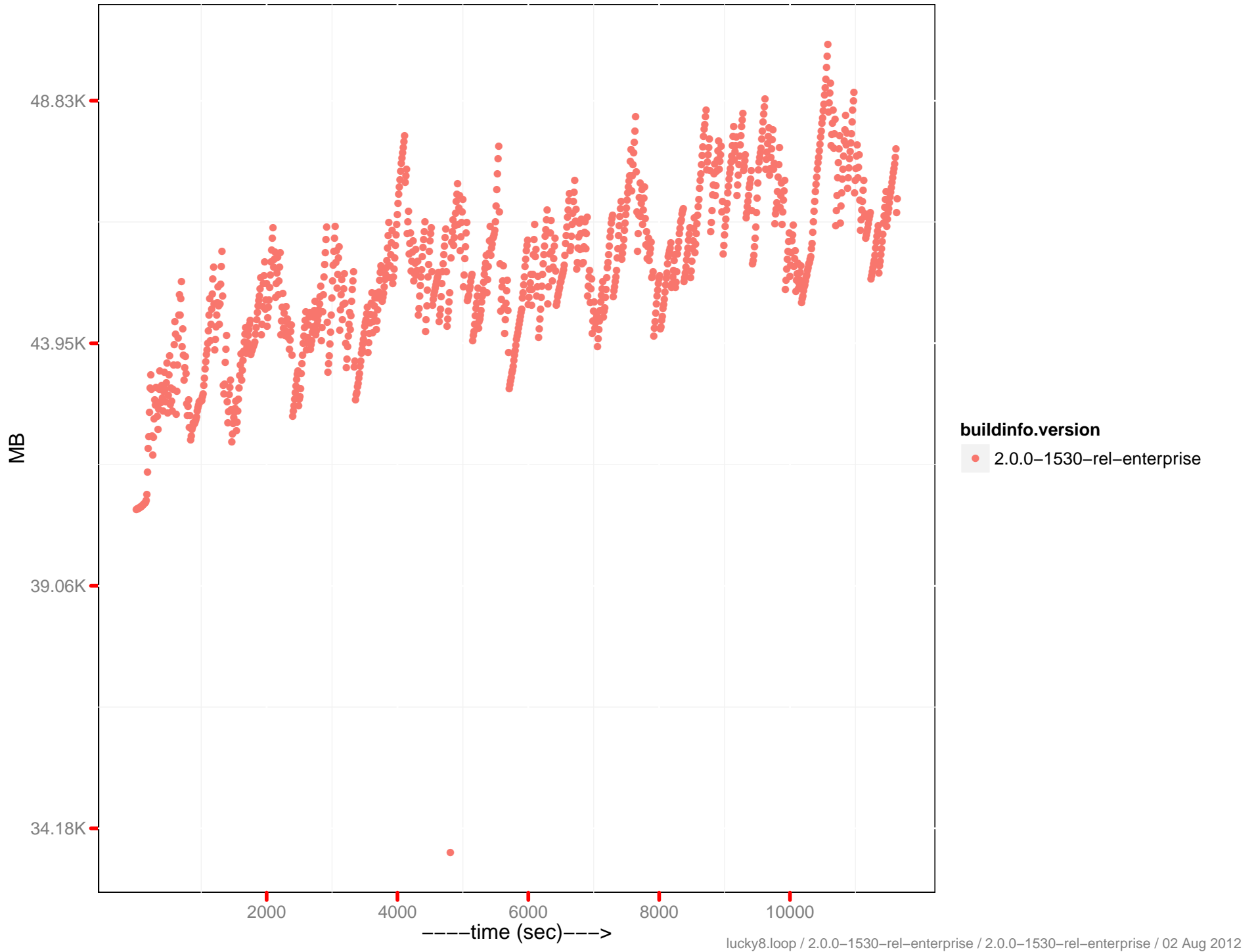
cpu_time_diff: memcached - 10.2.1.64



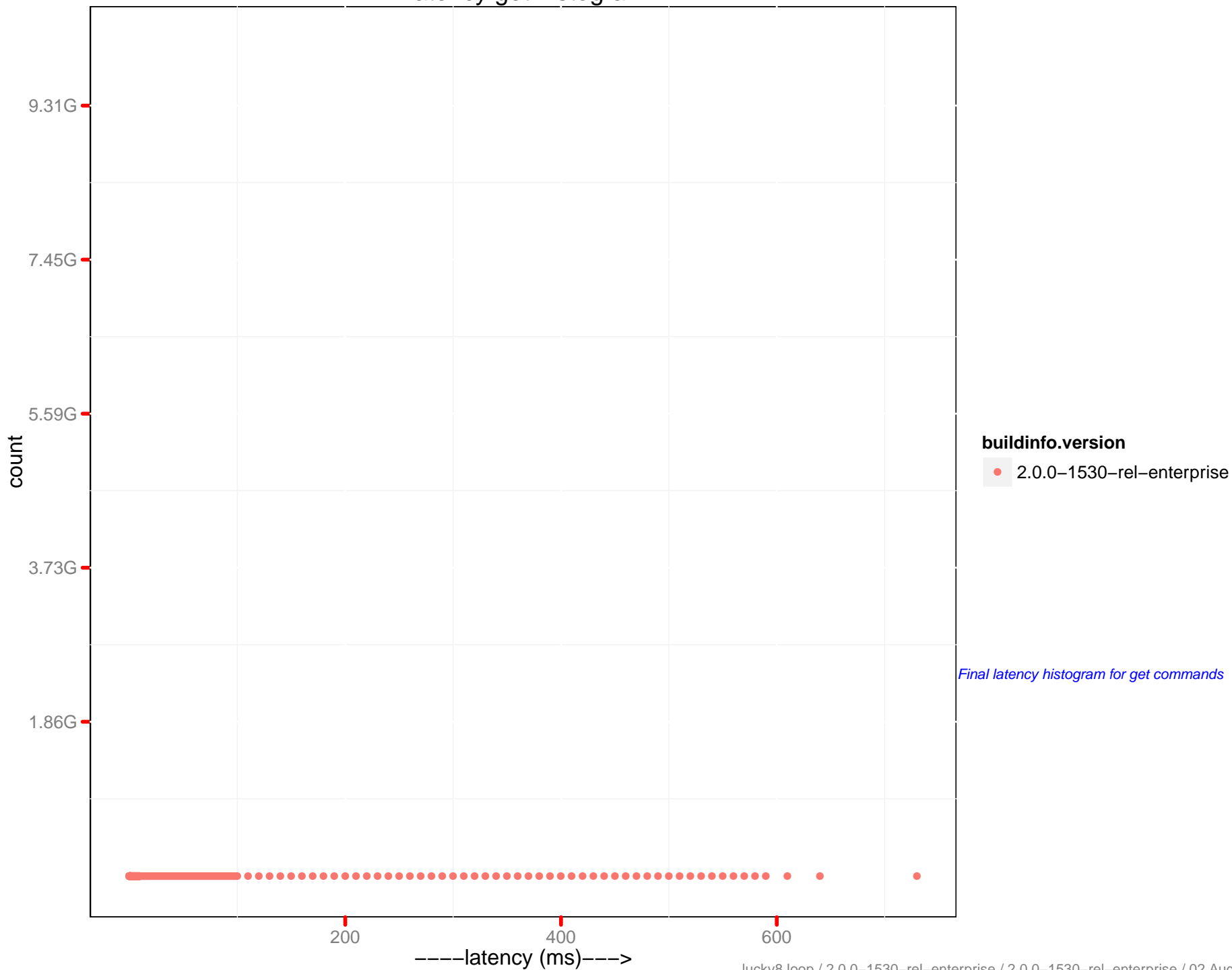
cpu_time_diff : beam.smp - 10.2.1.64



Data disk size

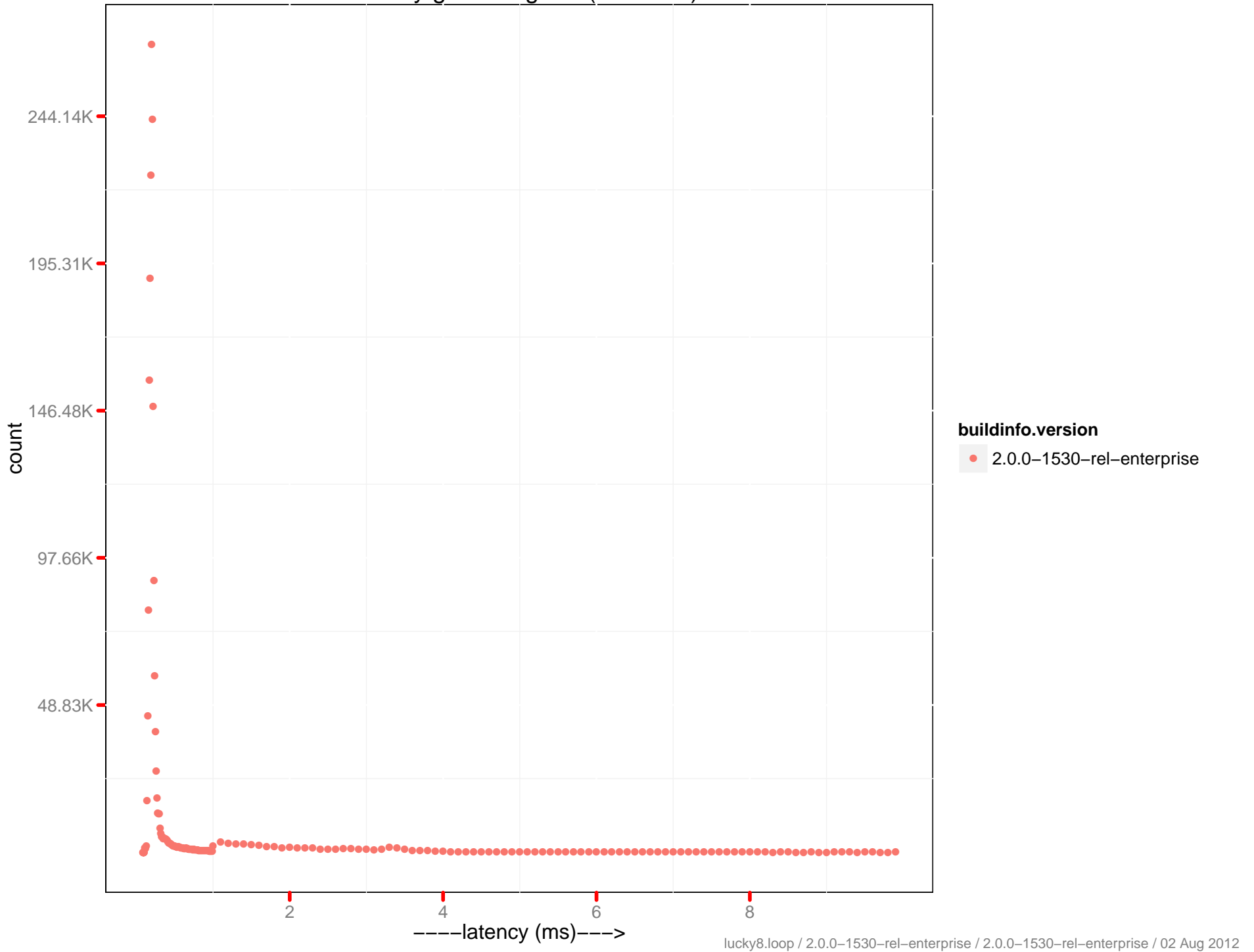


Latency get histogram



Final latency histogram for get commands

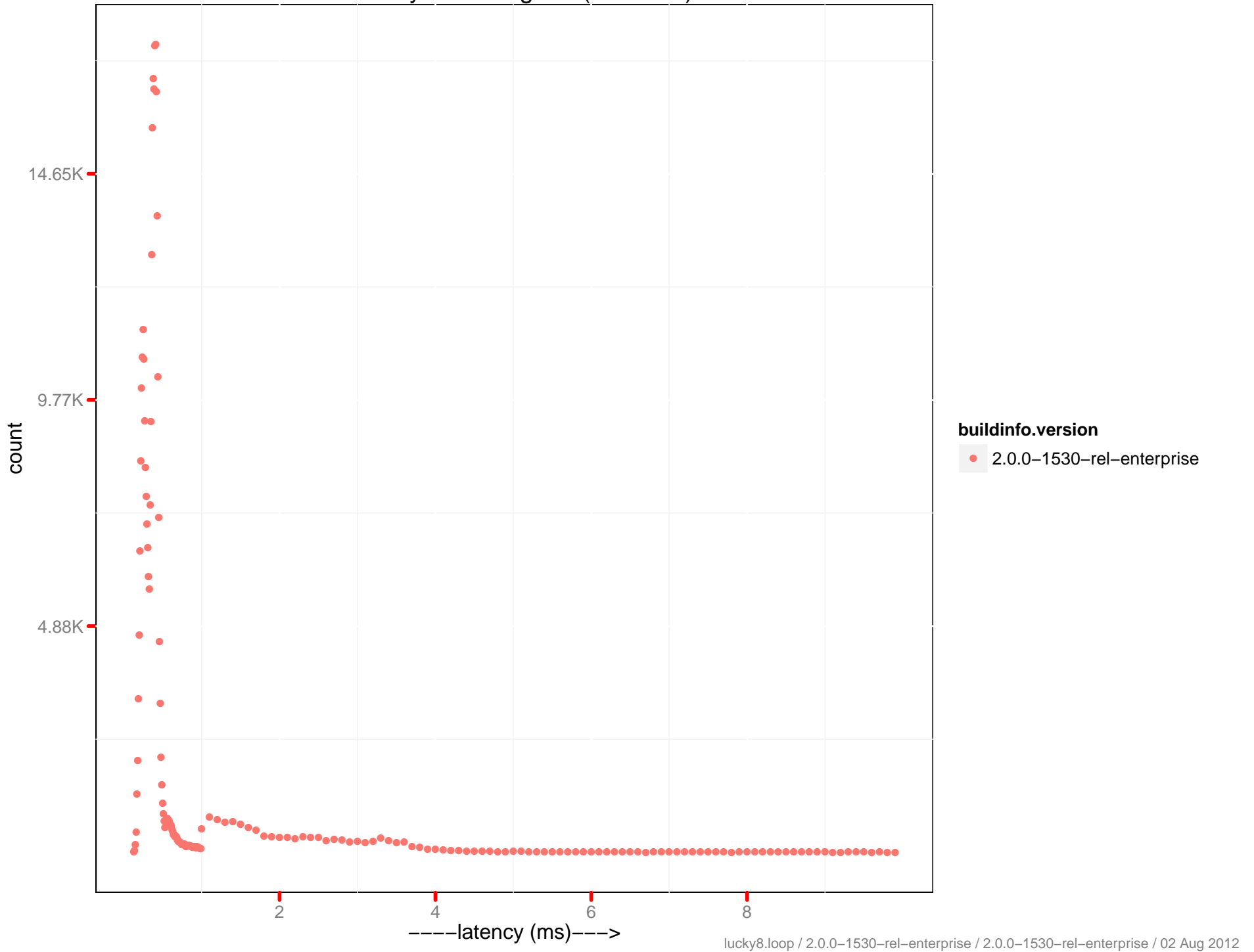
Latency get histogram (0–10 ms)



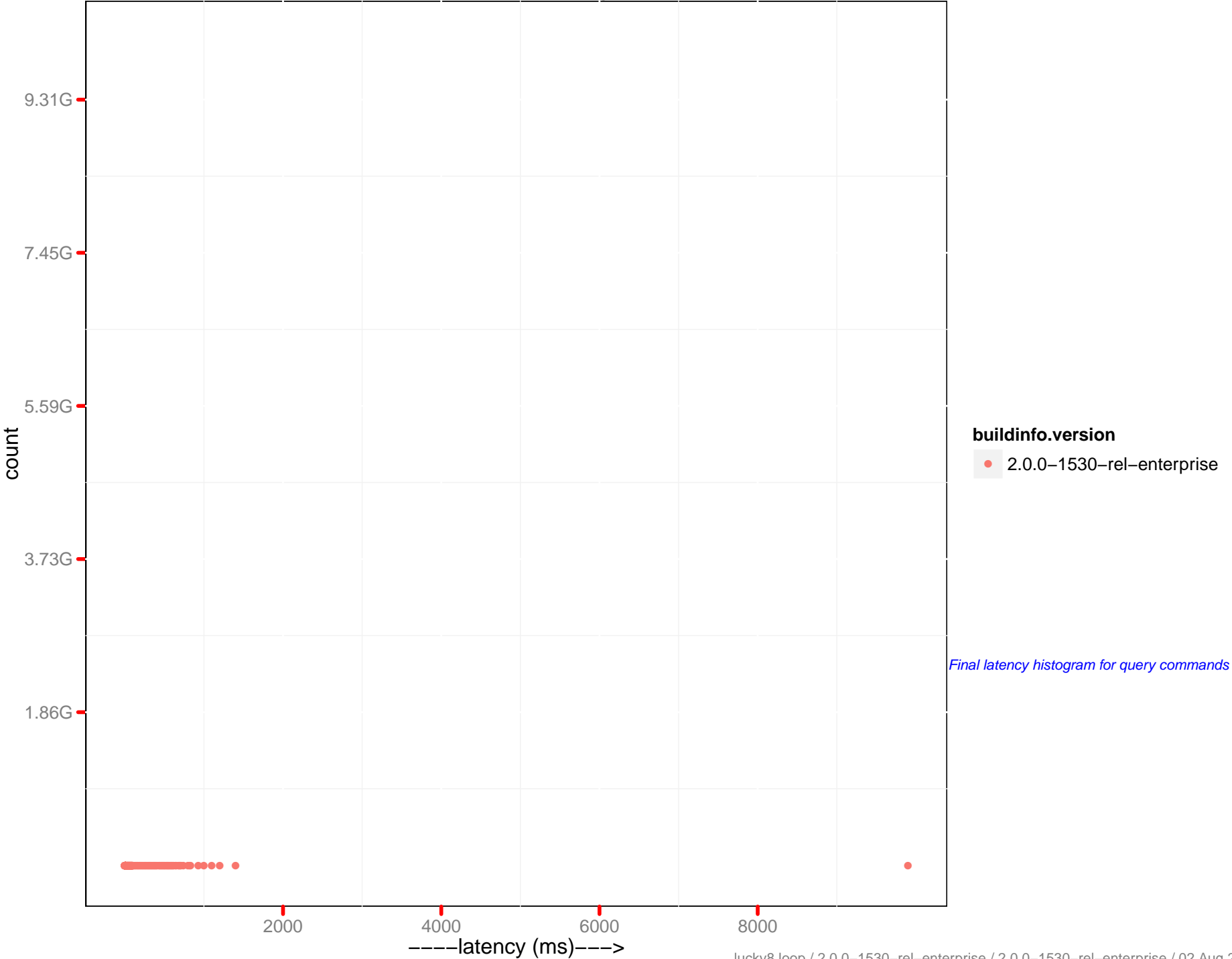
Latency set histogram



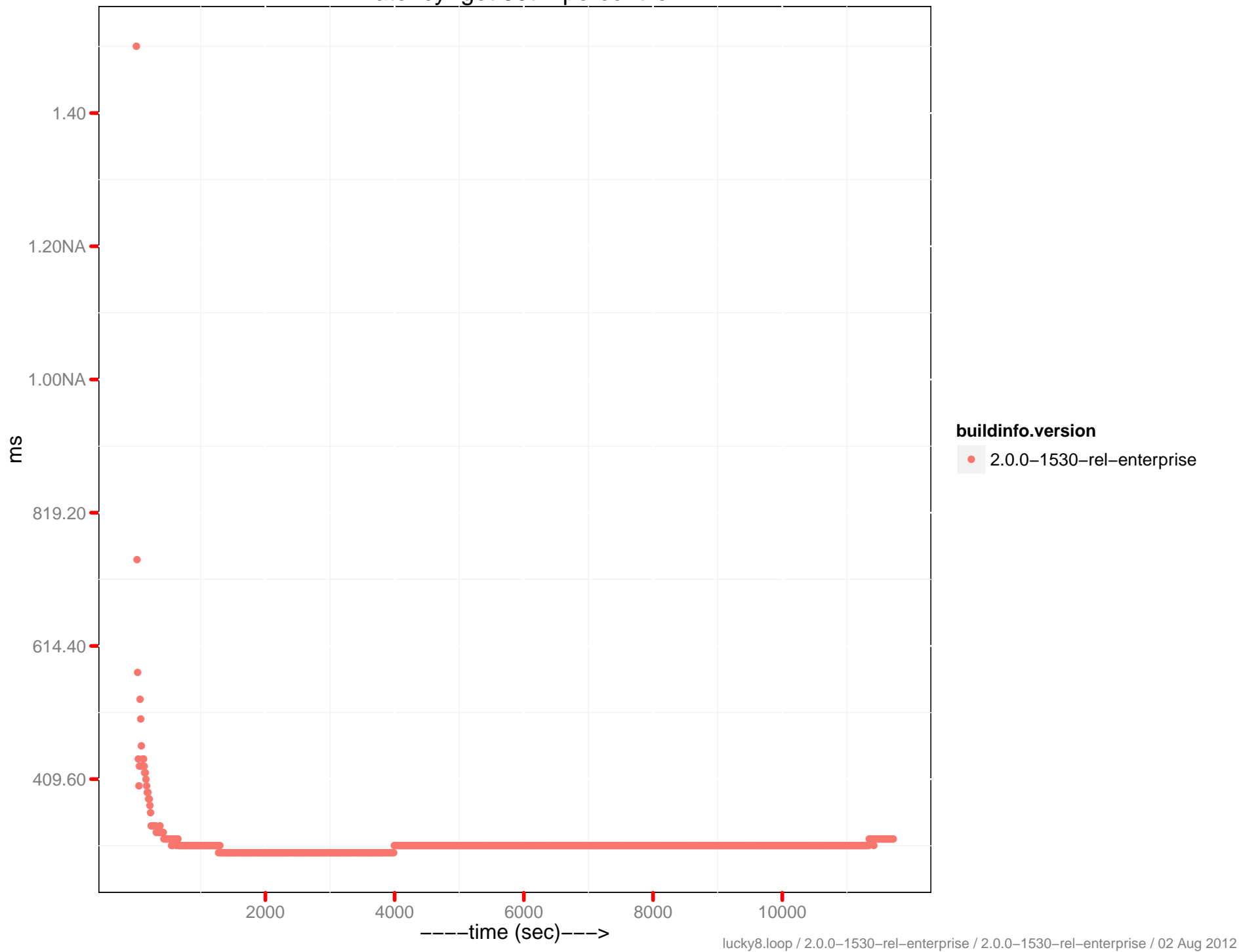
Latency set histogram (0-10 ms)



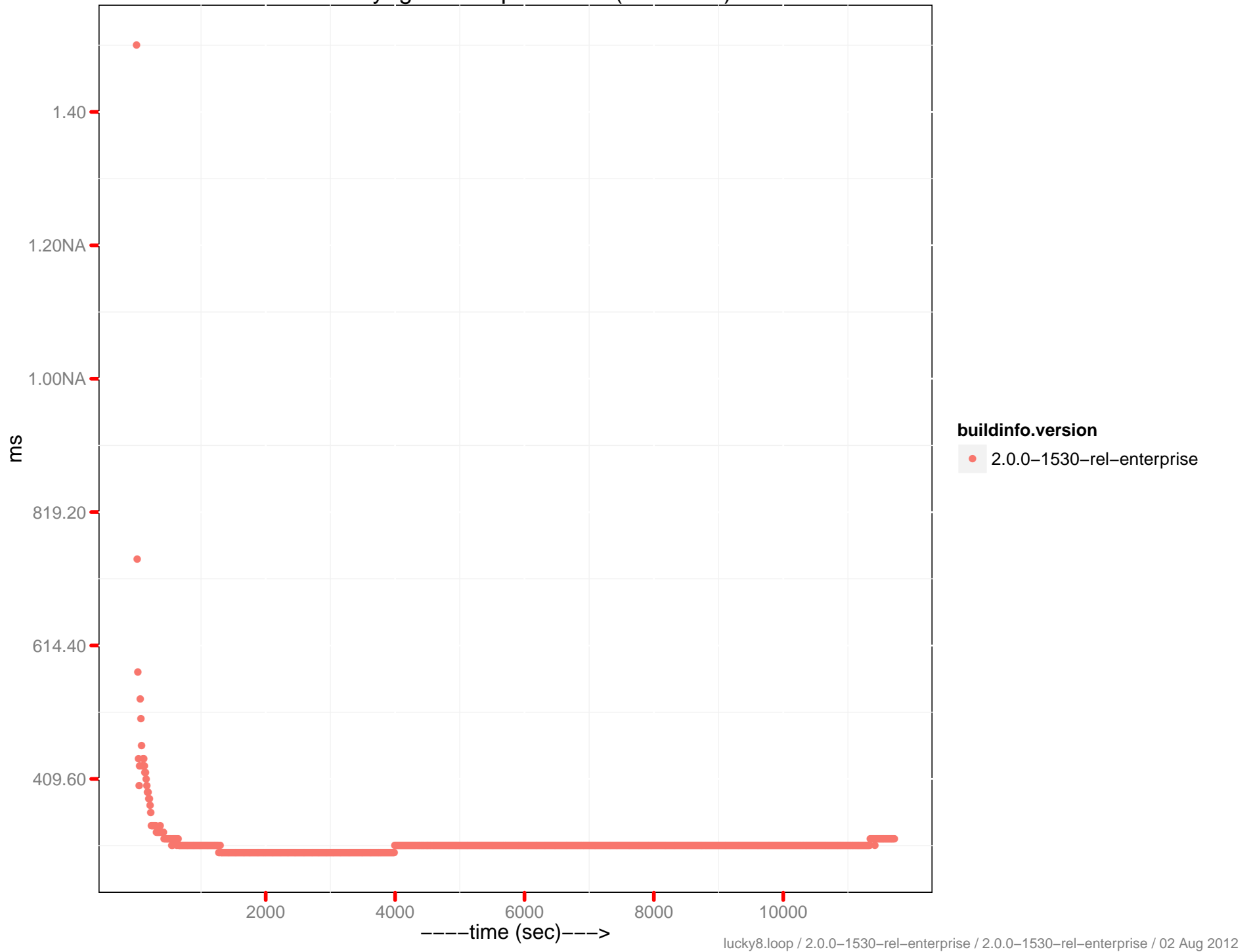
Latency query histogram



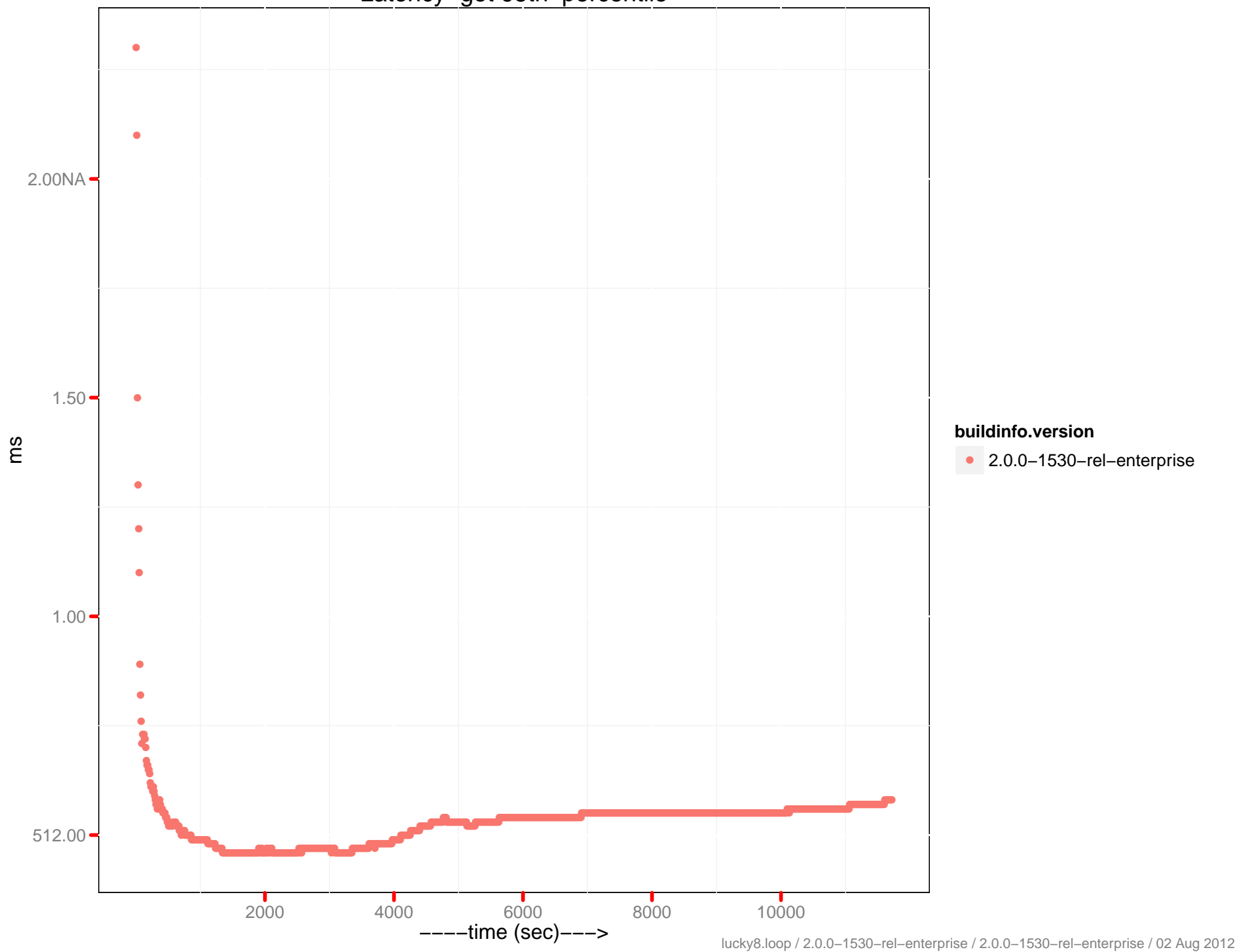
Latency-get 90th percentile



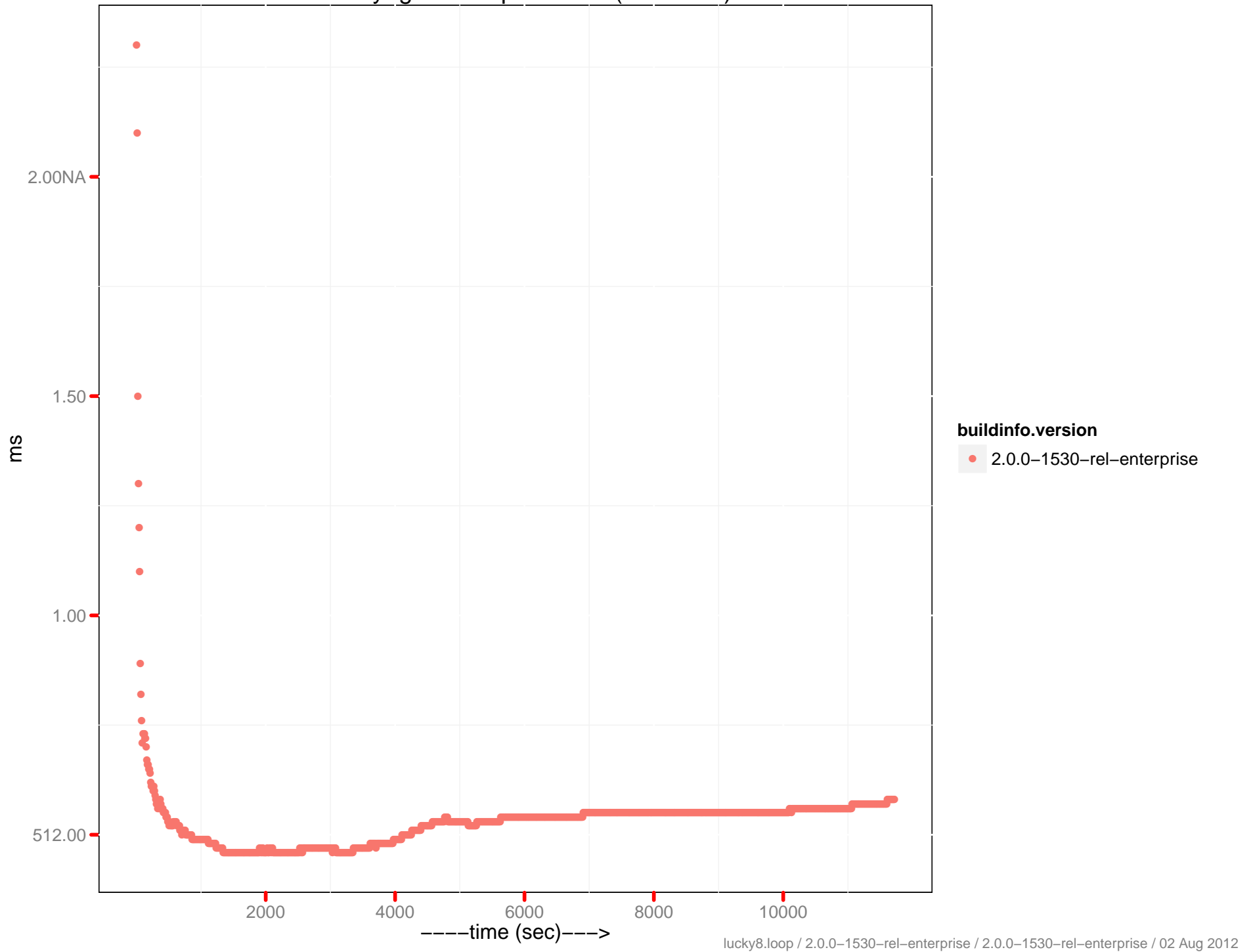
Latency-get 90th percentile (0 - 10ms)



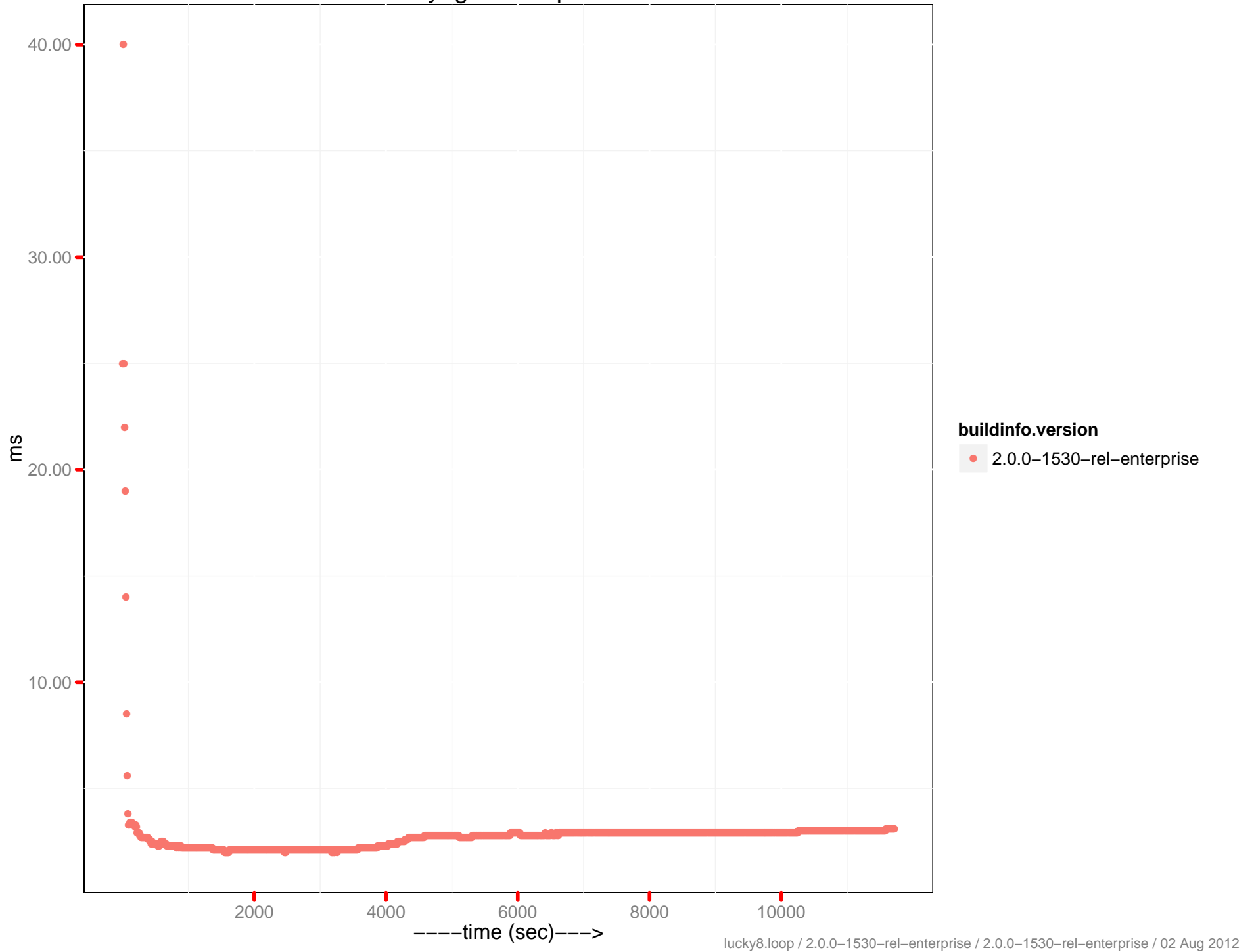
Latency-get 95th percentile



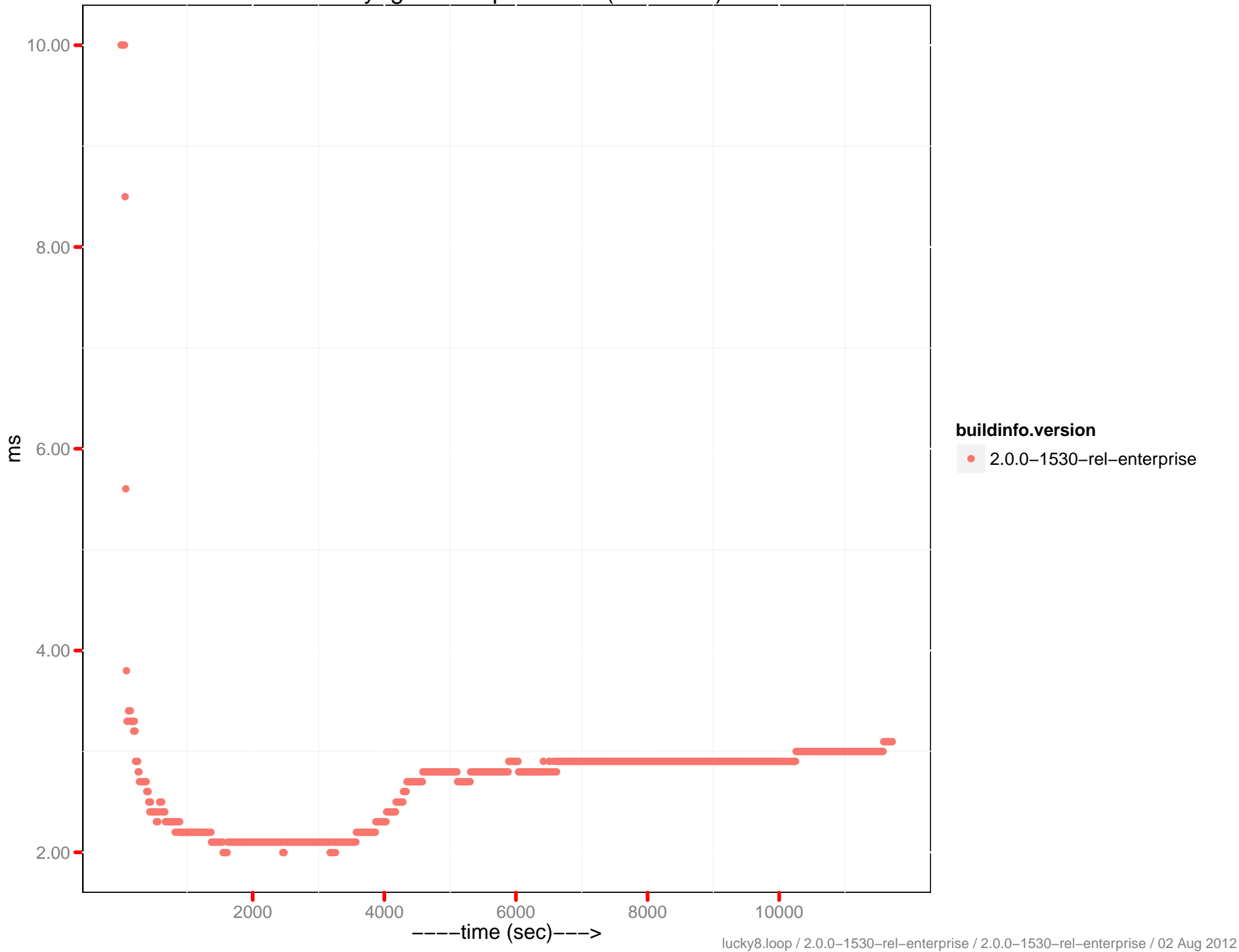
Latency-get 95th percentile (0 - 10ms)



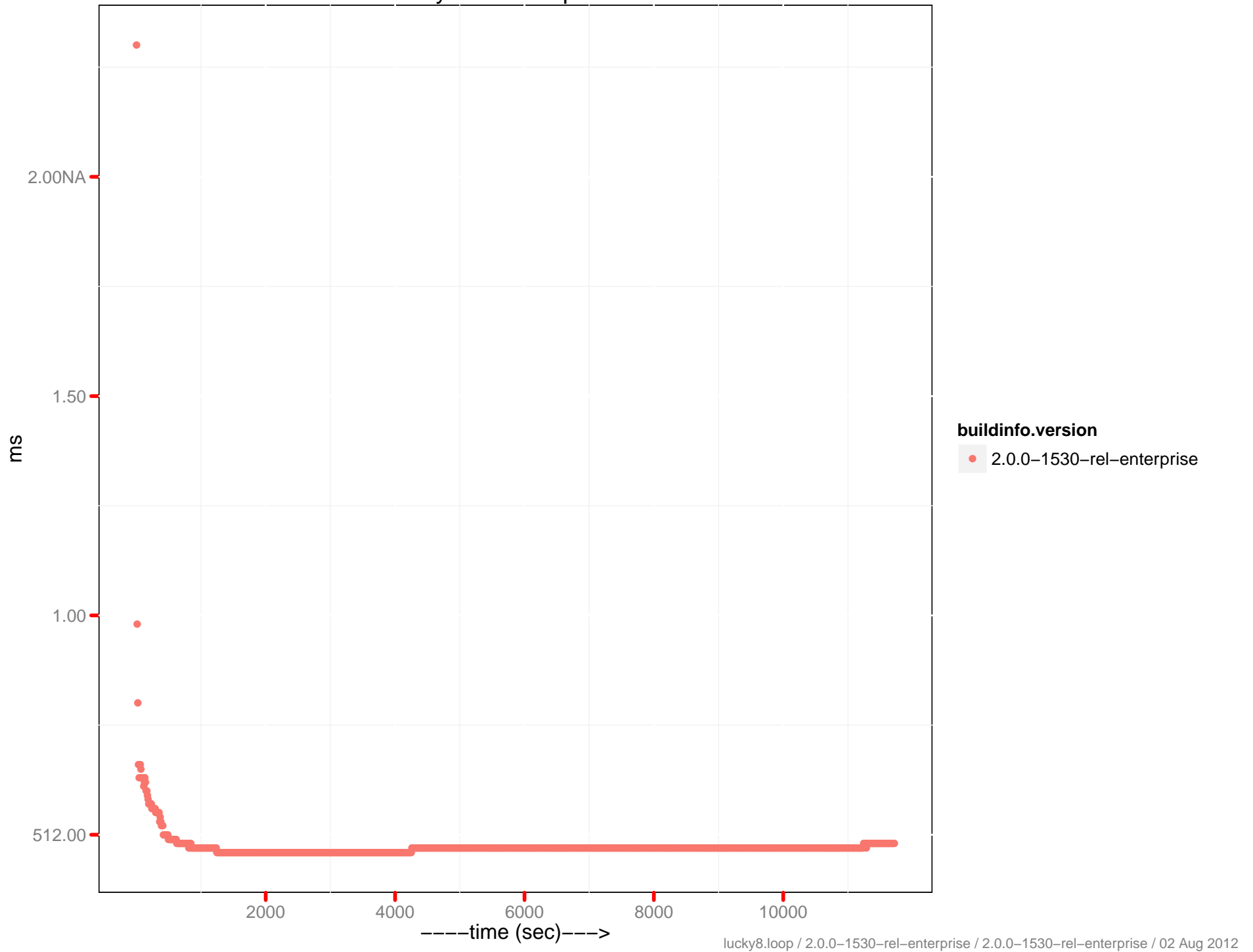
Latency-get 99th percentile



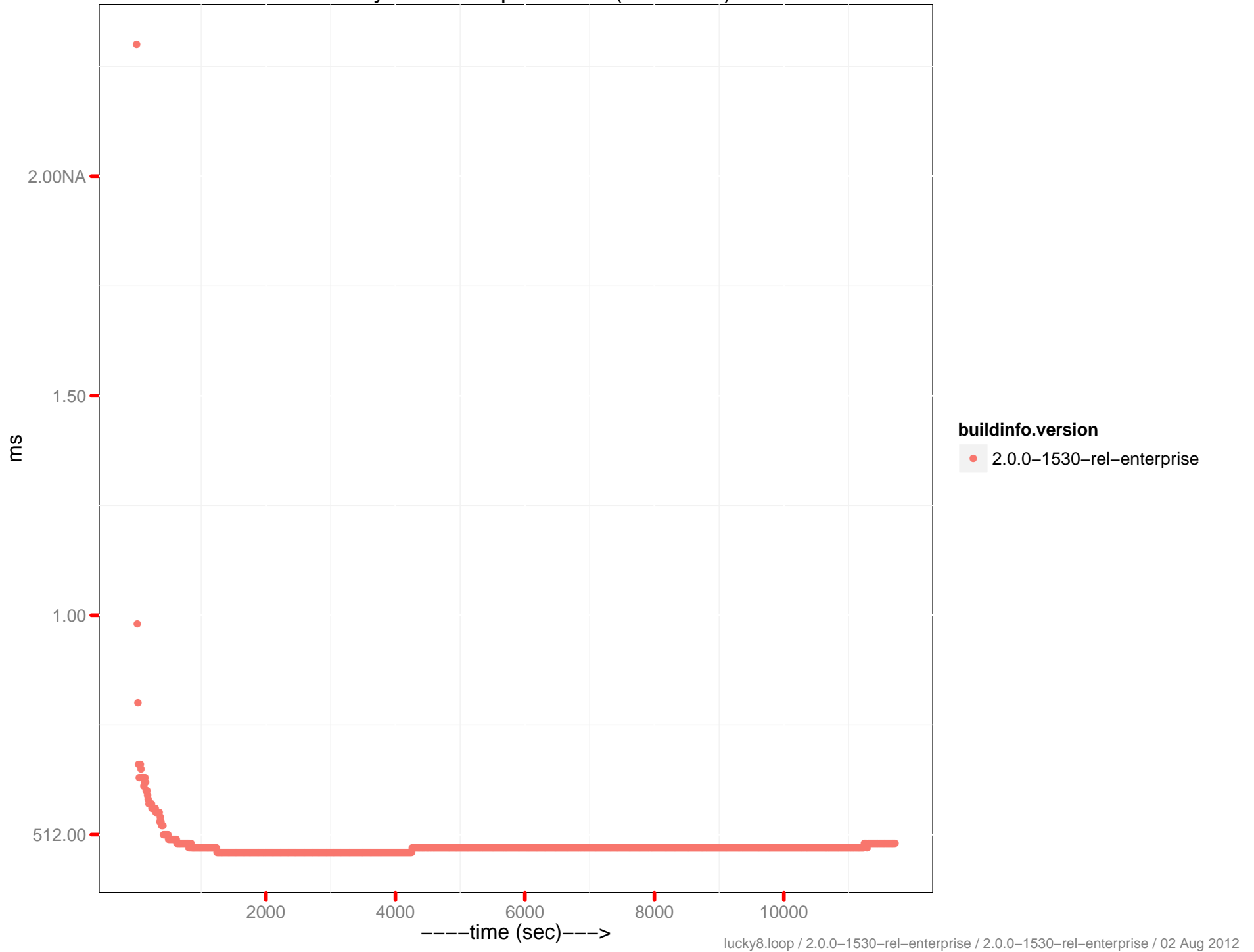
Latency-get 99th percentile (0 - 10ms)



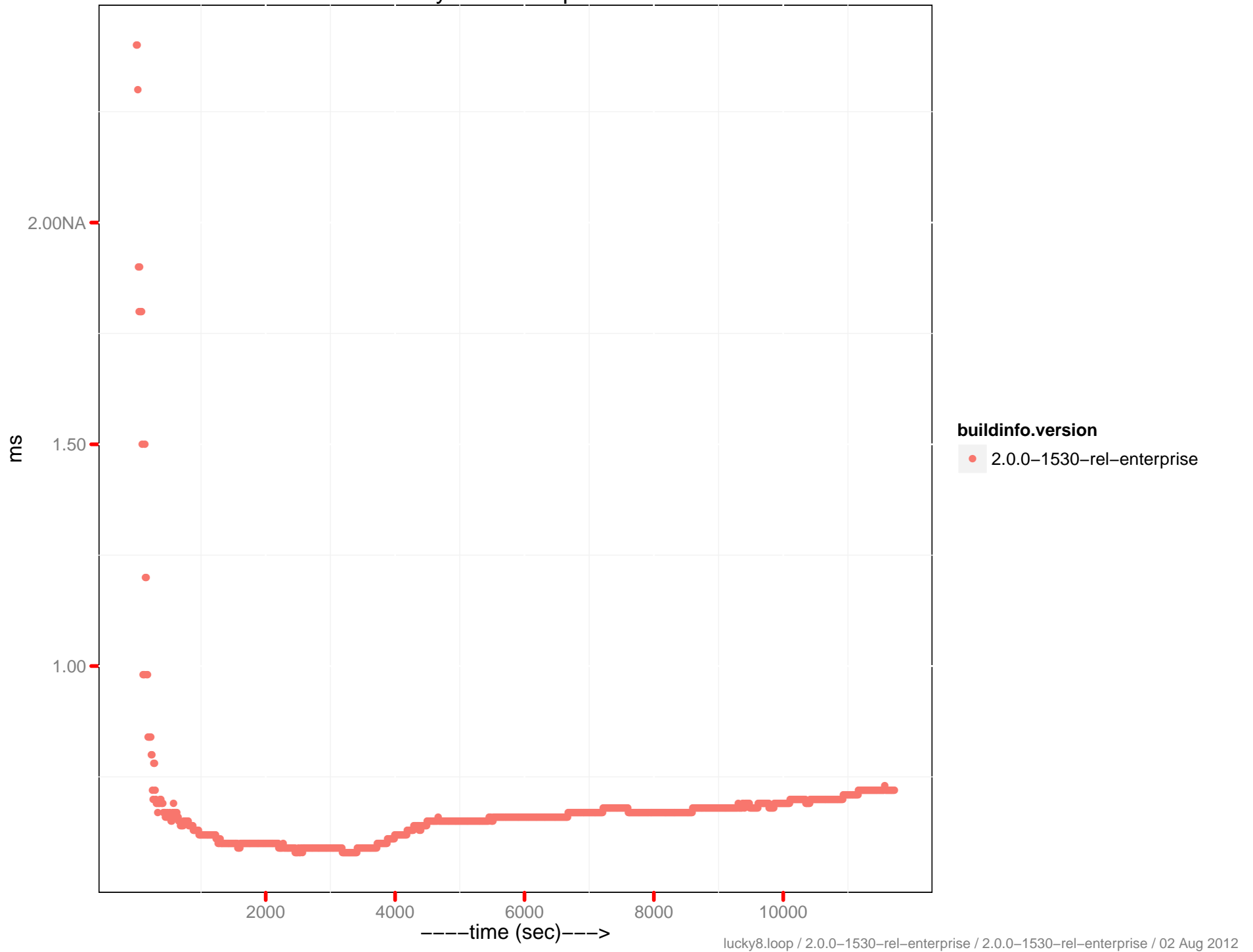
Latency-set 90th percentile



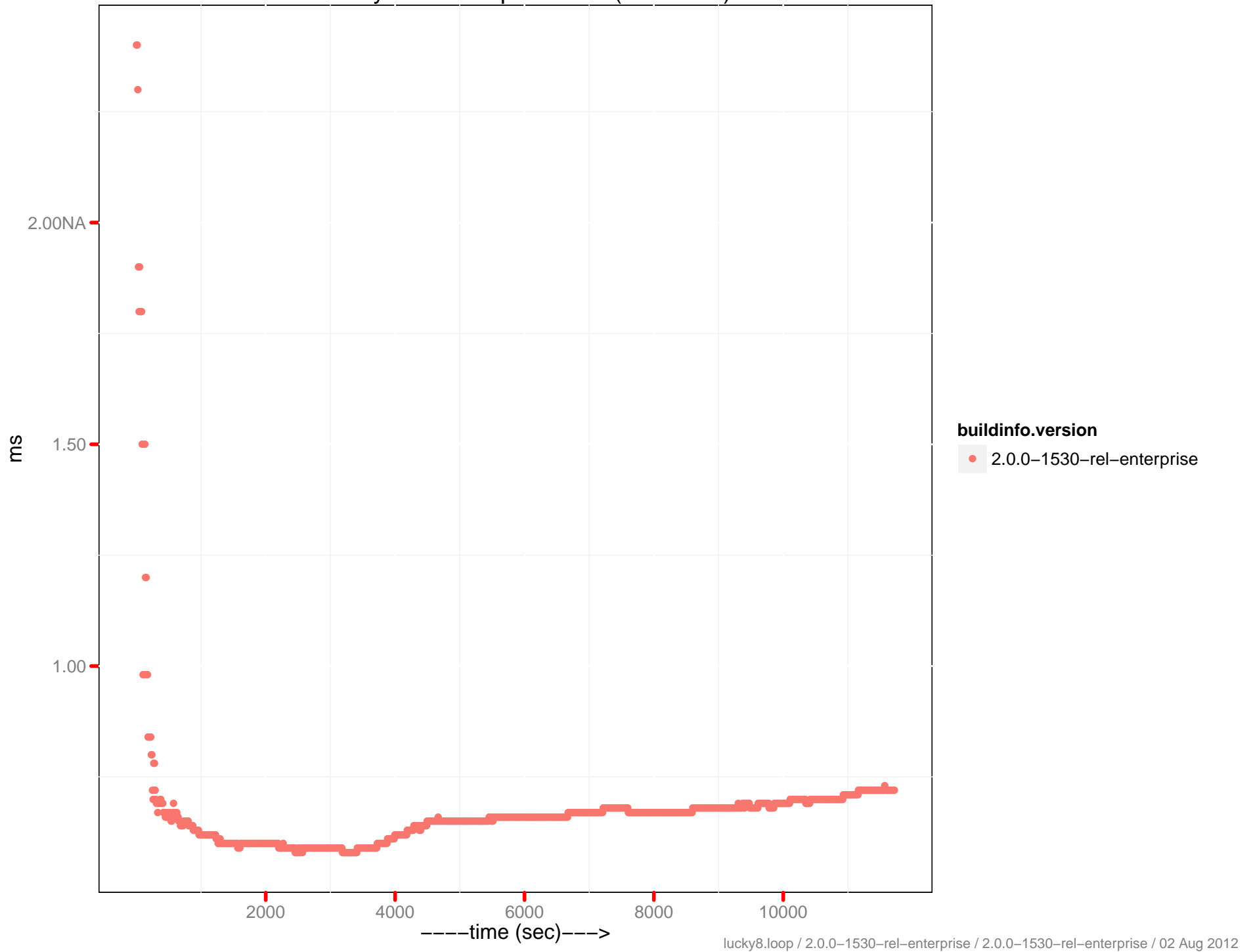
Latency-set 90th percentile (0 - 10ms)



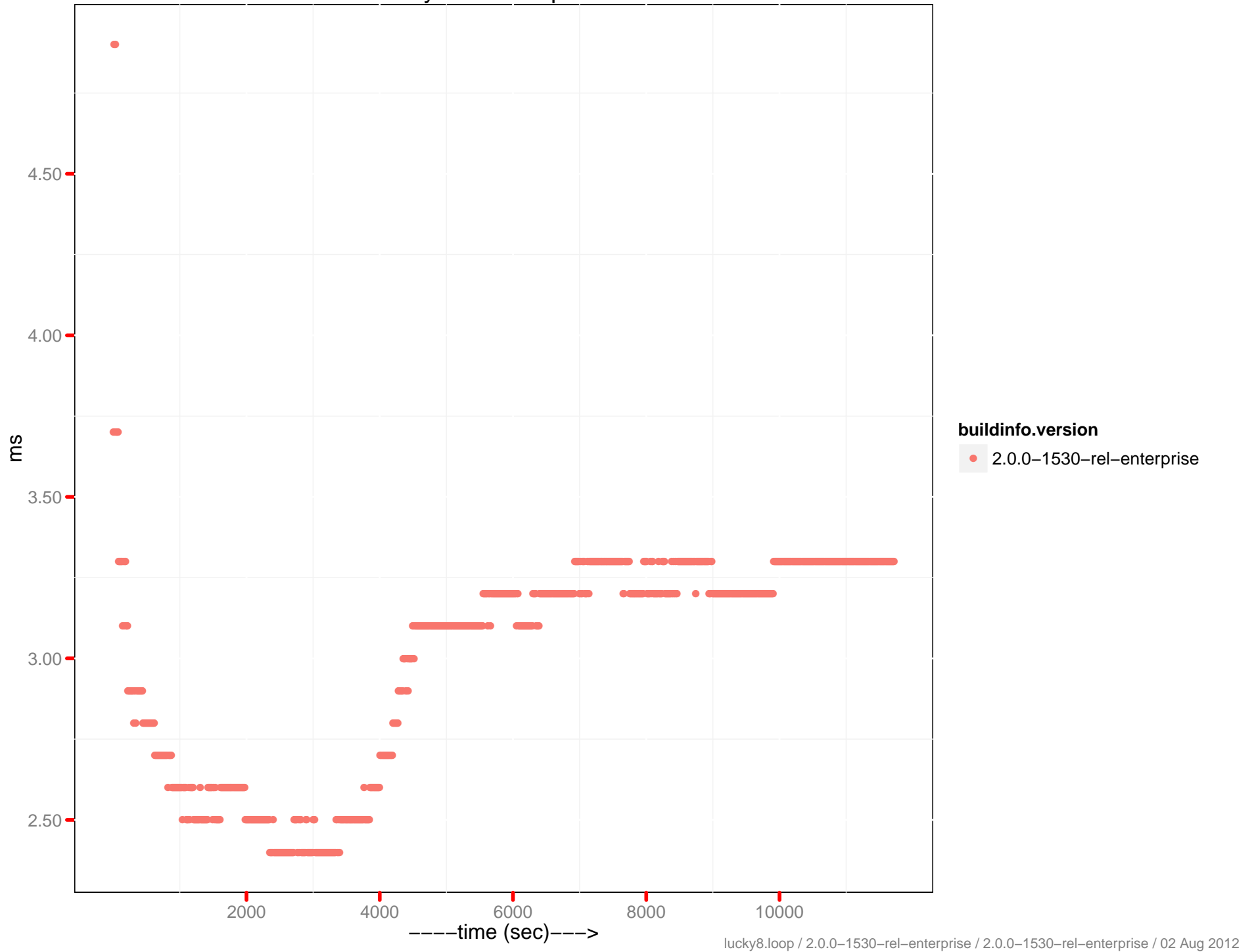
Latency-set 95th percentile



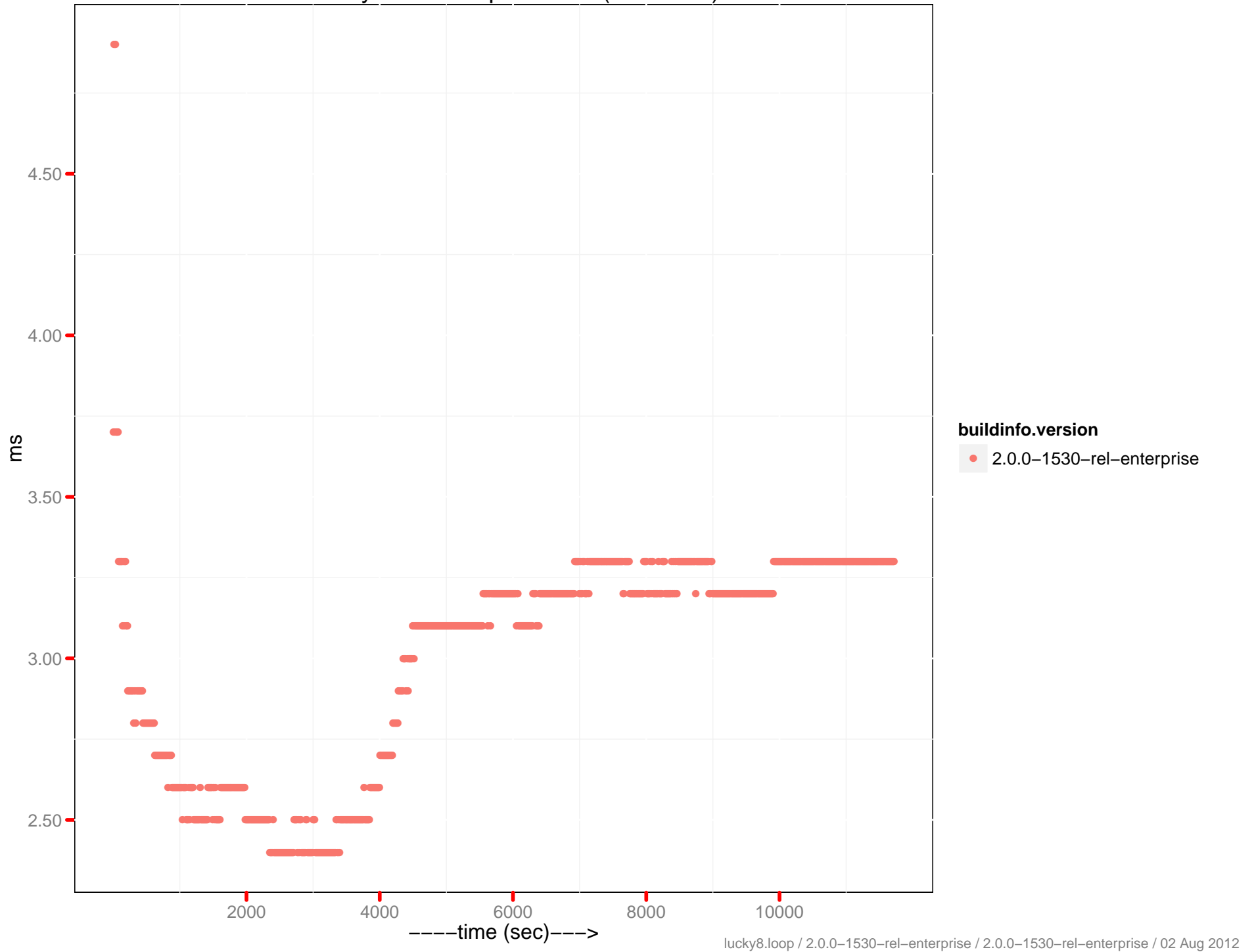
Latency-set 95th percentile (0 - 10ms)



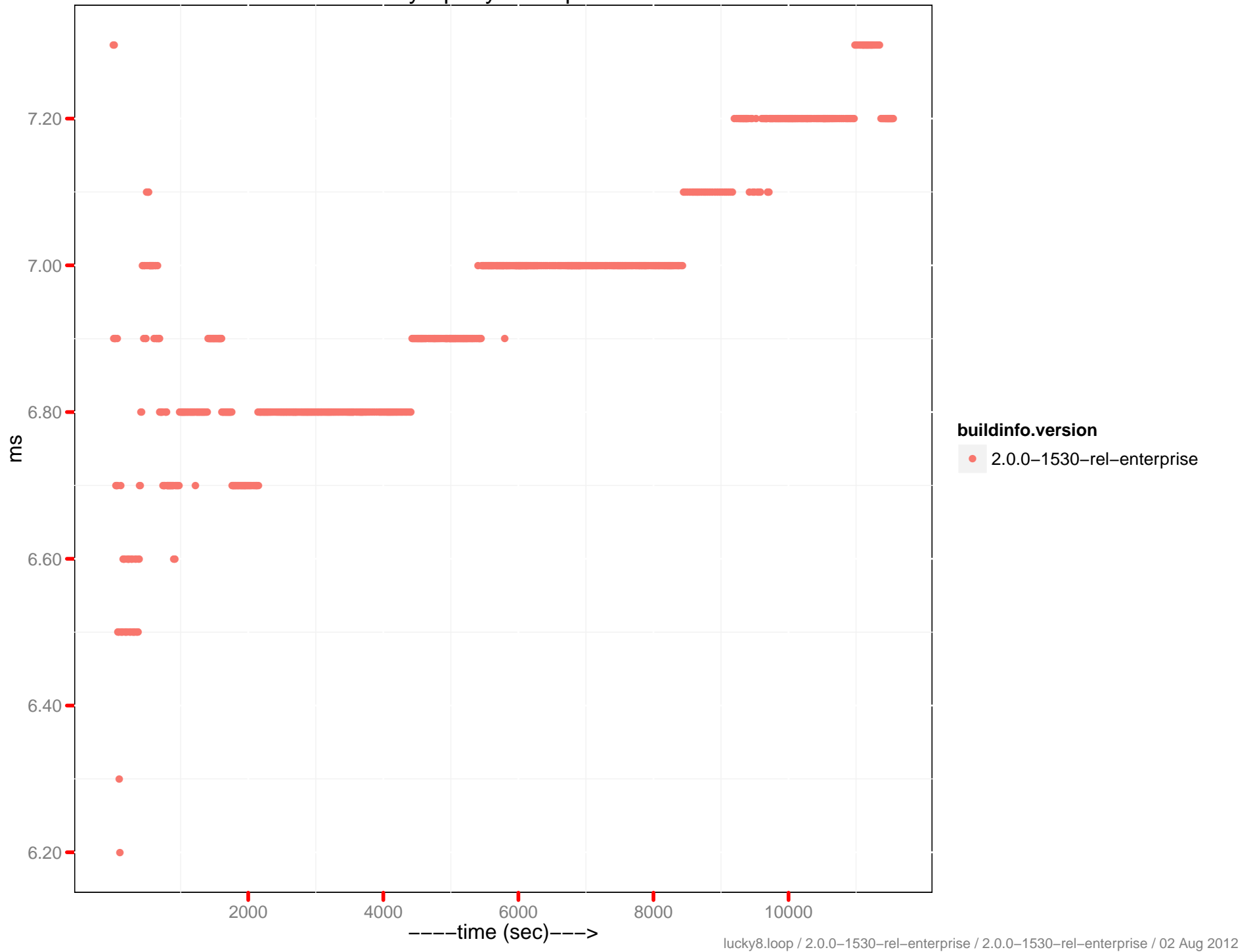
Latency-set 99th percentile



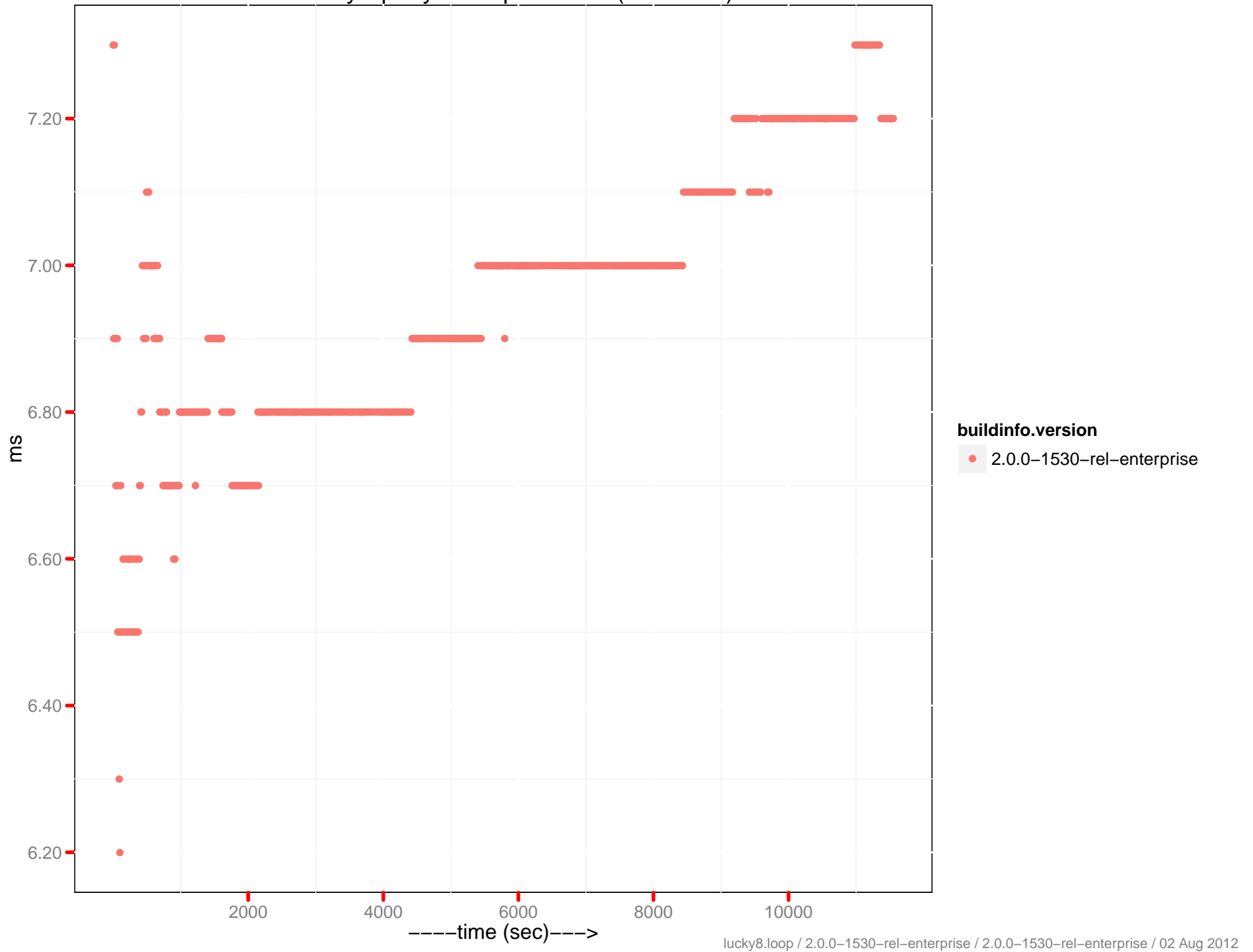
Latency-set 99th percentile (0 - 10ms)



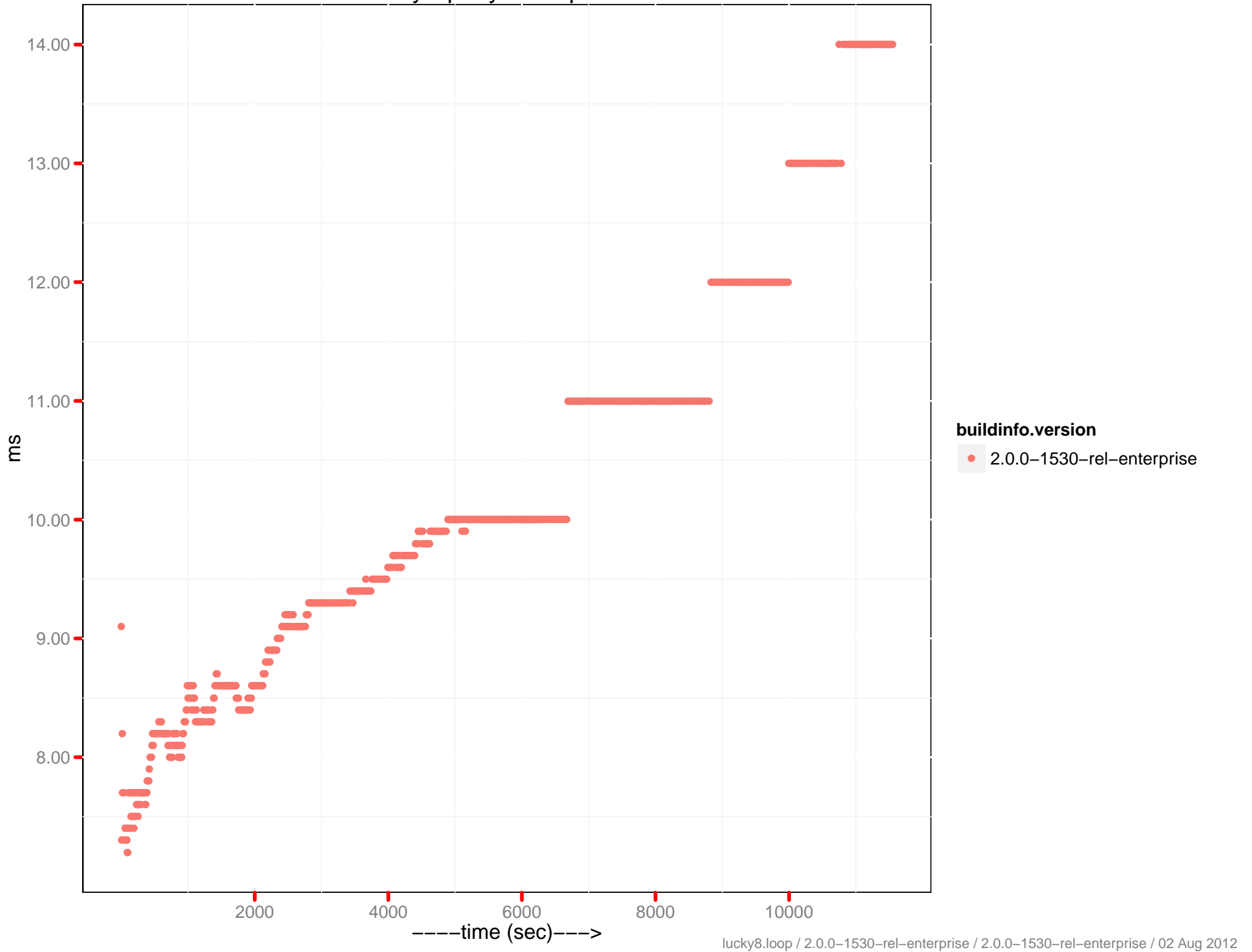
Latency-query 80th percentile



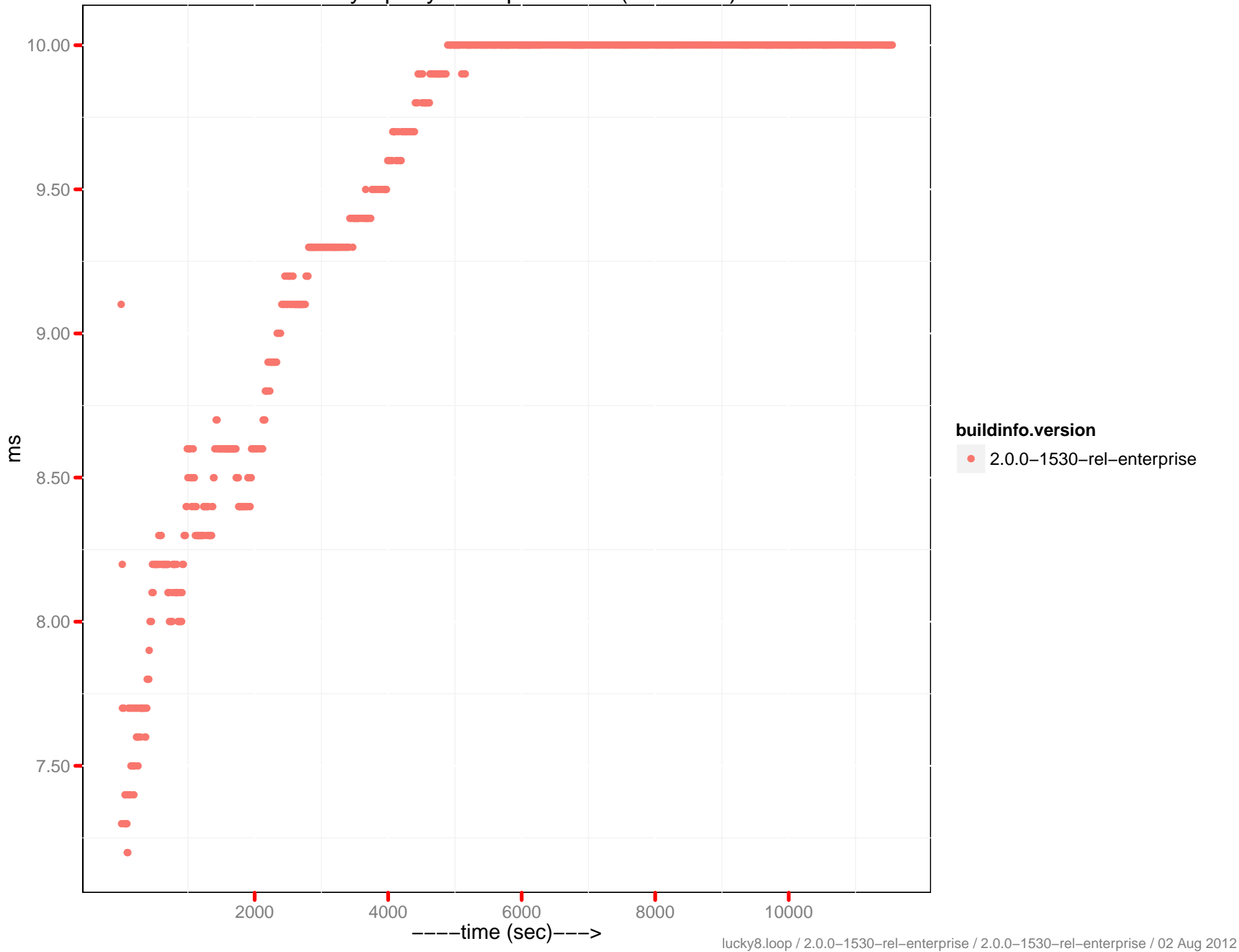
Latency-query 80th percentile (0 - 10ms)



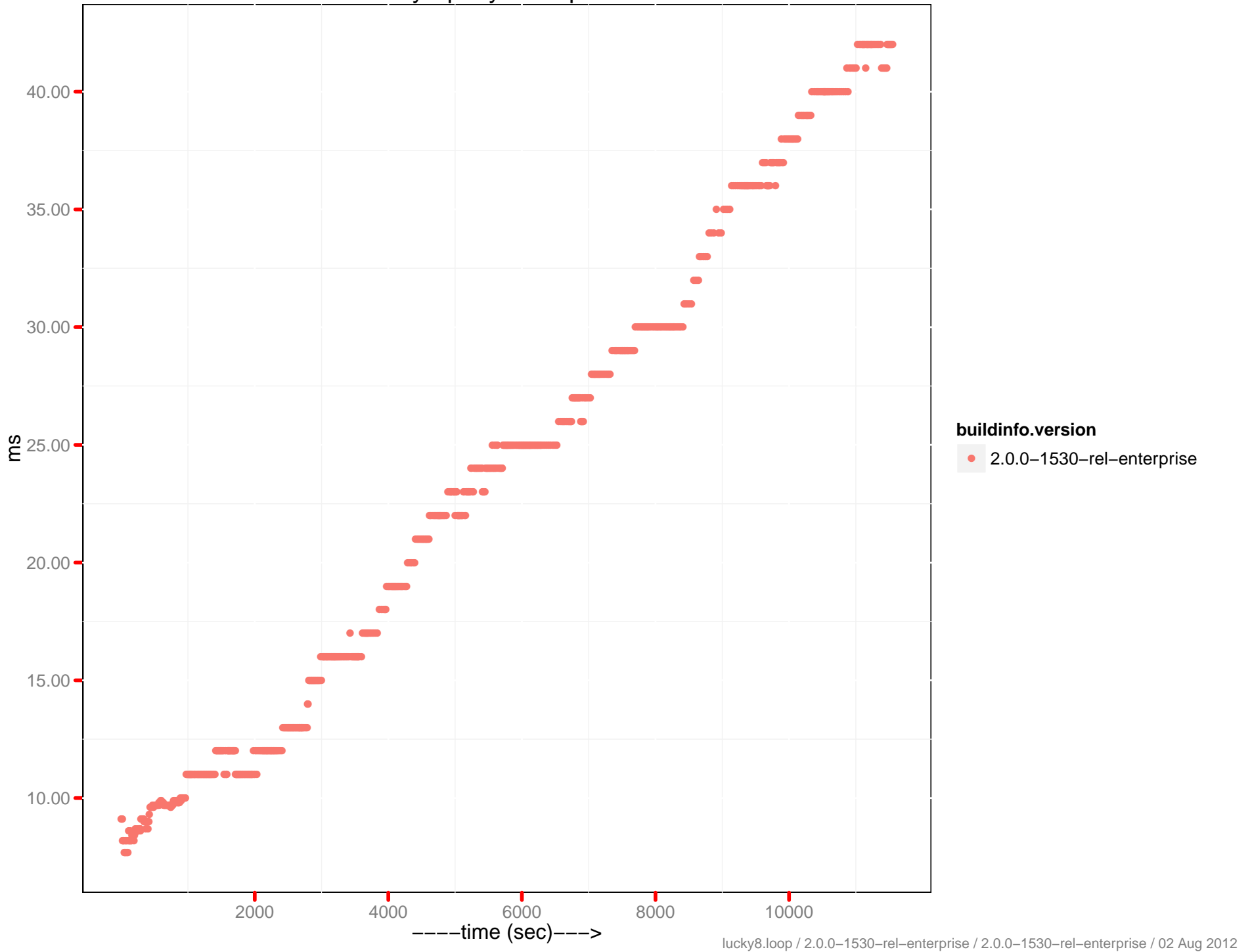
Latency-query 90th percentile



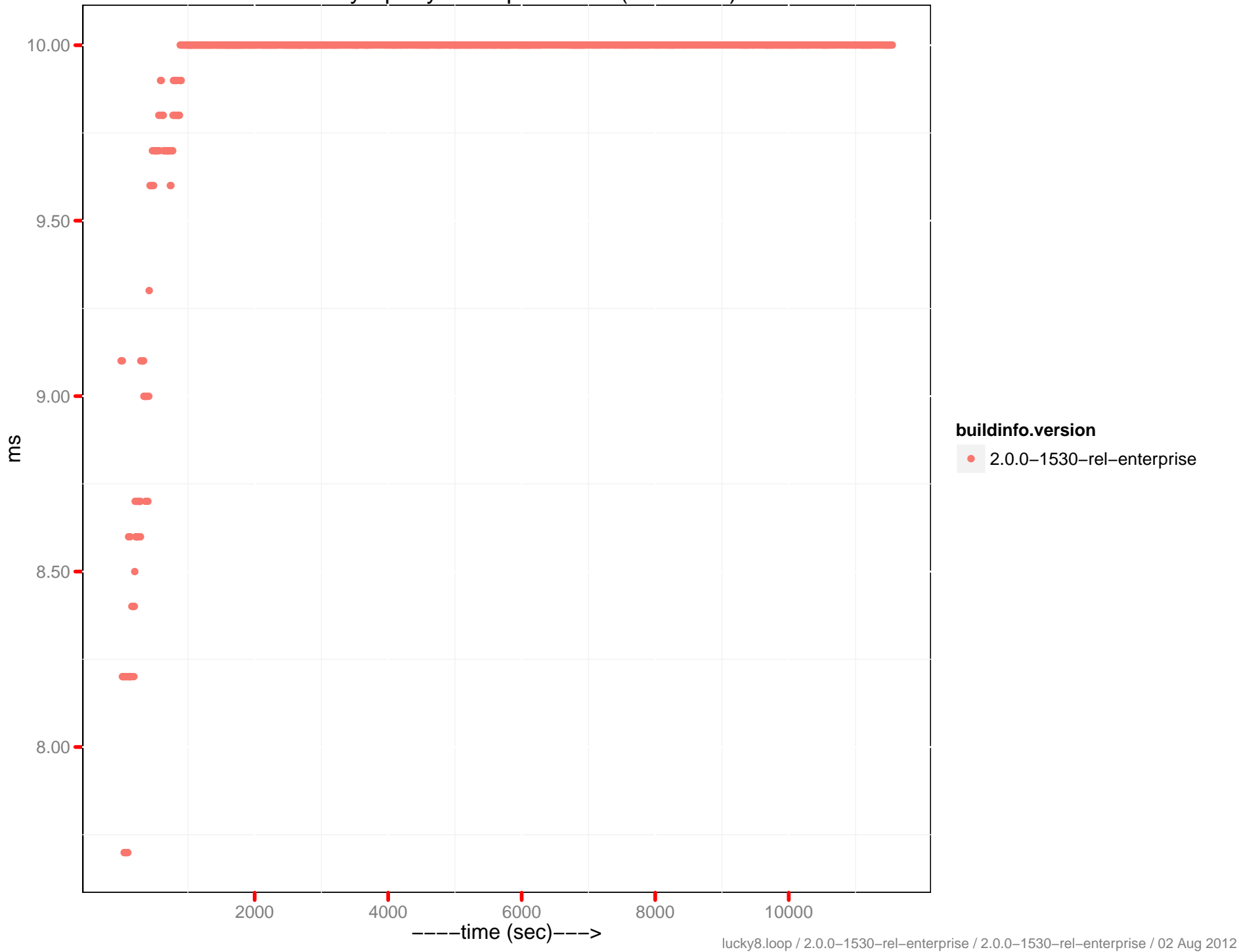
Latency-query 90th percentile (0 - 10ms)



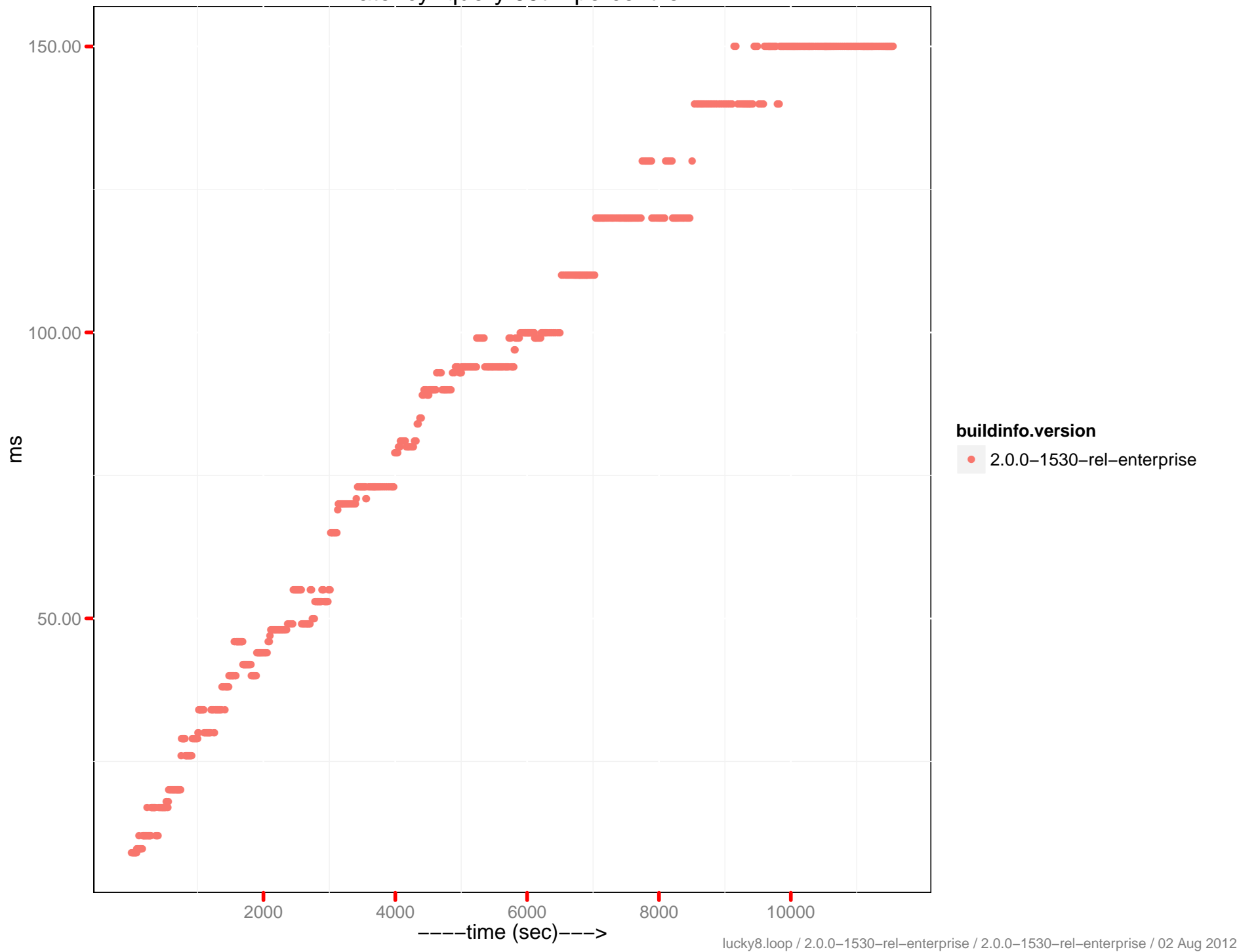
Latency-query 95th percentile



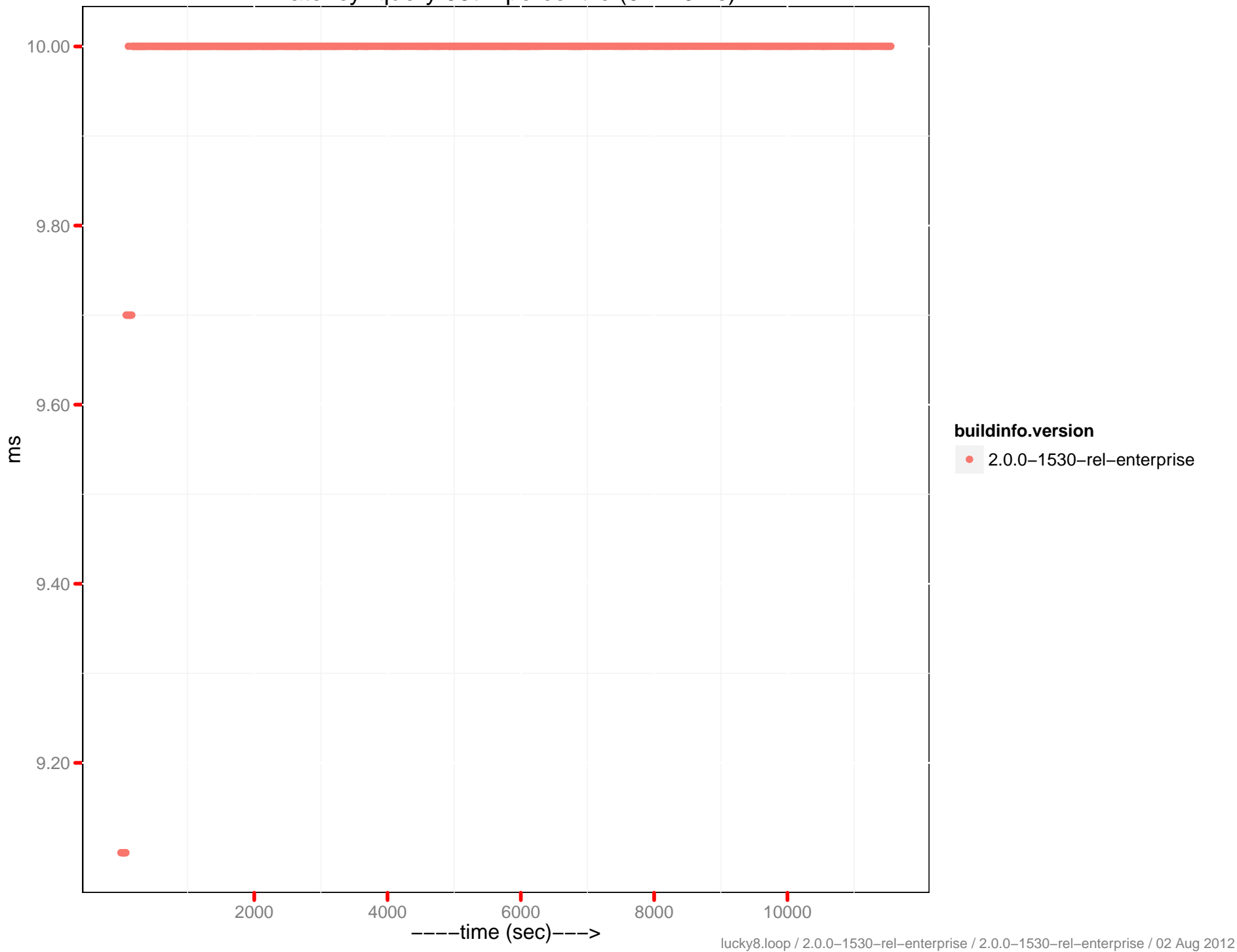
Latency-query 95th percentile (0 - 10ms)



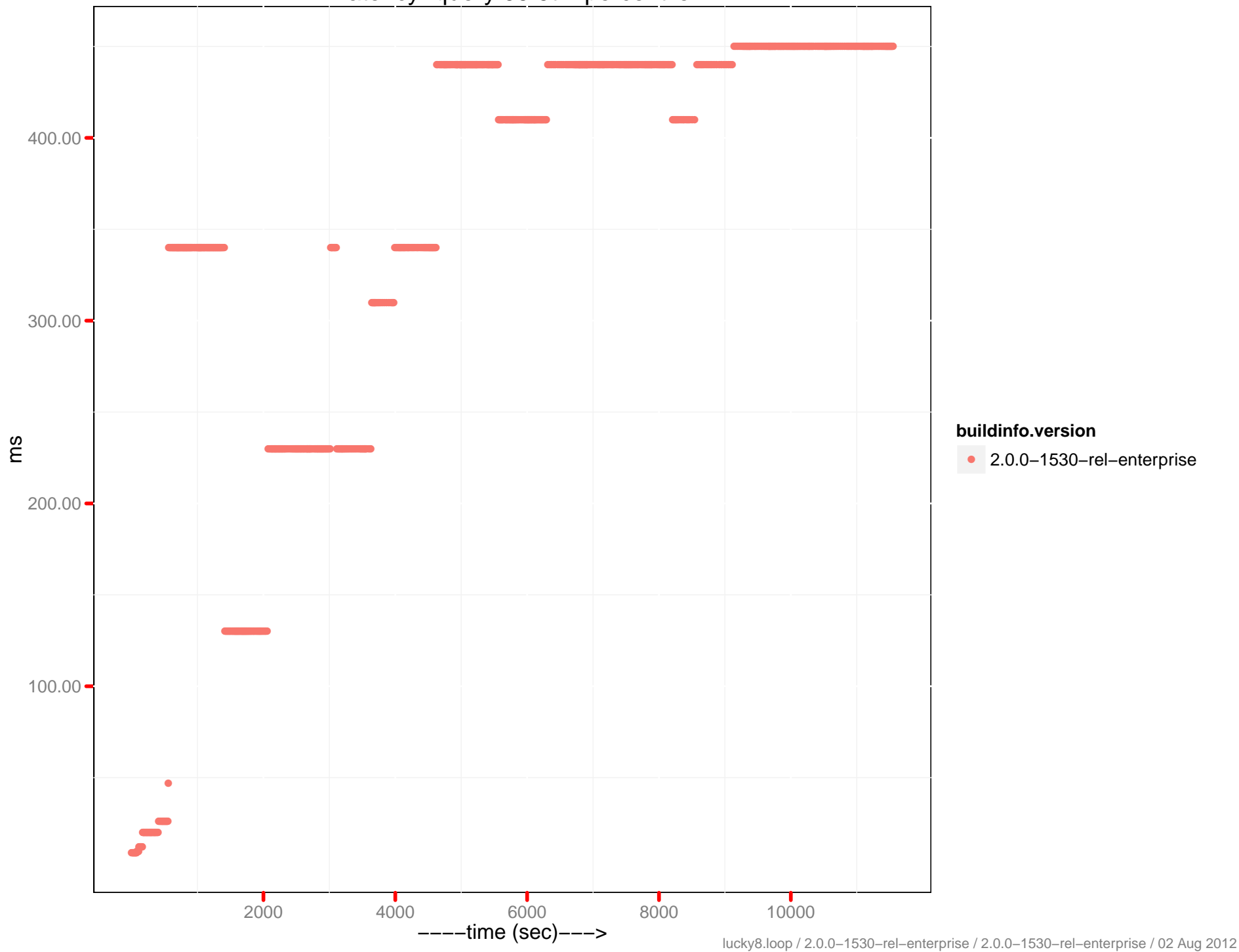
Latency-query 99th percentile



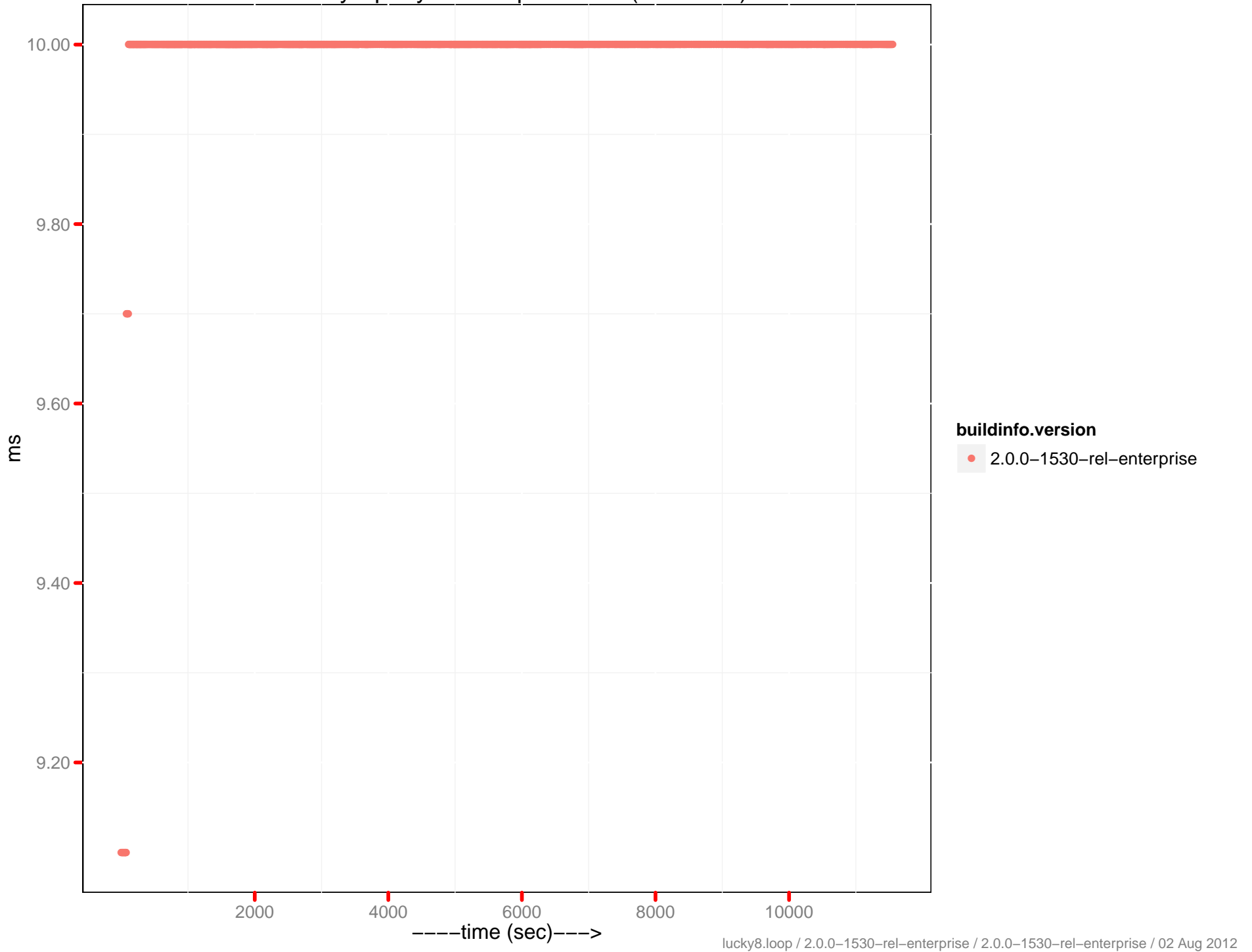
Latency-query 99th percentile (0 - 10ms)



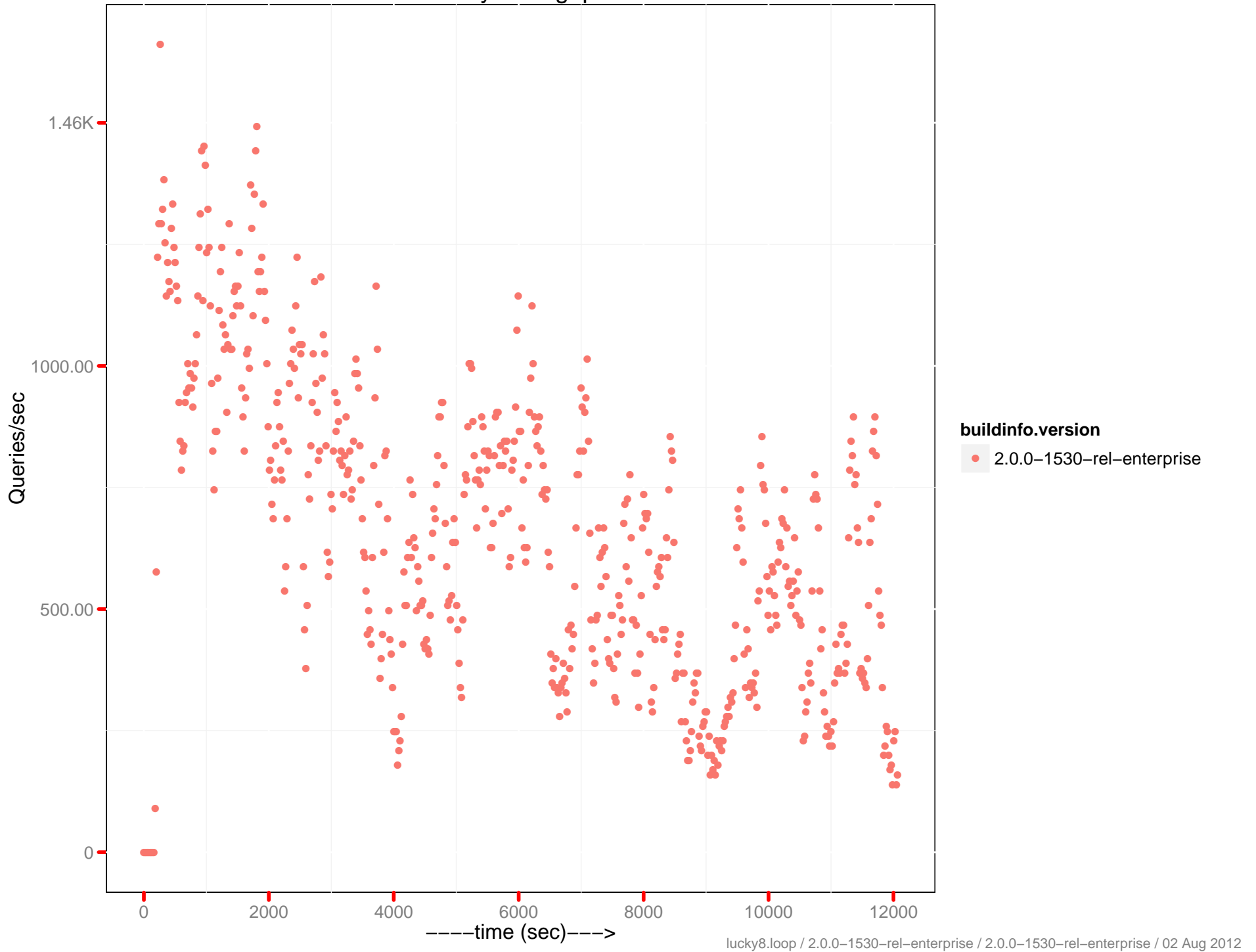
Latency-query 99.9th percentile



Latency-query 99.9th percentile (0 - 10ms)



Query throughput




```
lucky8.conf
# "lucky 8" view performance test:
# 8K ops/sec (background, cluster-wide)
# 80% reads, 20% write (12% updates/deletes, 8% inserts)
# 8M dataset
# Stop after 8M total queries
# 1 ddoc with 8 views

performance.ipperf.MultiClientTests.test_vperf2

params:

# general
batch=50
kind=json
mem_quota=16000
loglevel=error

# load phase
items=8000000

# access phase
ratio_sets=0.2
ratio_misses=0.04
ratio_creates=0.40
ratio_deletes=0.50
ratio_hot=0.2
ratio_hot_gets=0.95
ratio_hot_sets=0.95
ratio_expirations=0.0
bg_max_ops_per_sec=1000
fg_max_ops=8000000
total_clients=8

# control (defaults: pytests/performance/perf_defaults.py)
load_wait_until_drained=1
loop_wait_until_drained=0
mcsoda_heartbeat=3
tear_down=1
tear_down_proxy=1
tear_down_bucket=0
tear_down_cluster=1
tear_down_on_setup=0
```

```
terra.ini
[global]
username:root
password:couchbase
port:8091
data_path:/data

[servers]
1:10.2.1.61
2:10.2.1.62
3:10.2.1.63
4:10.2.1.64

[clients]
1:10.2.1.59

[membase]
rest_username:Administrator
rest_password:password

[dashboard]
1:dashboard.hq.couchbase.com:80
```