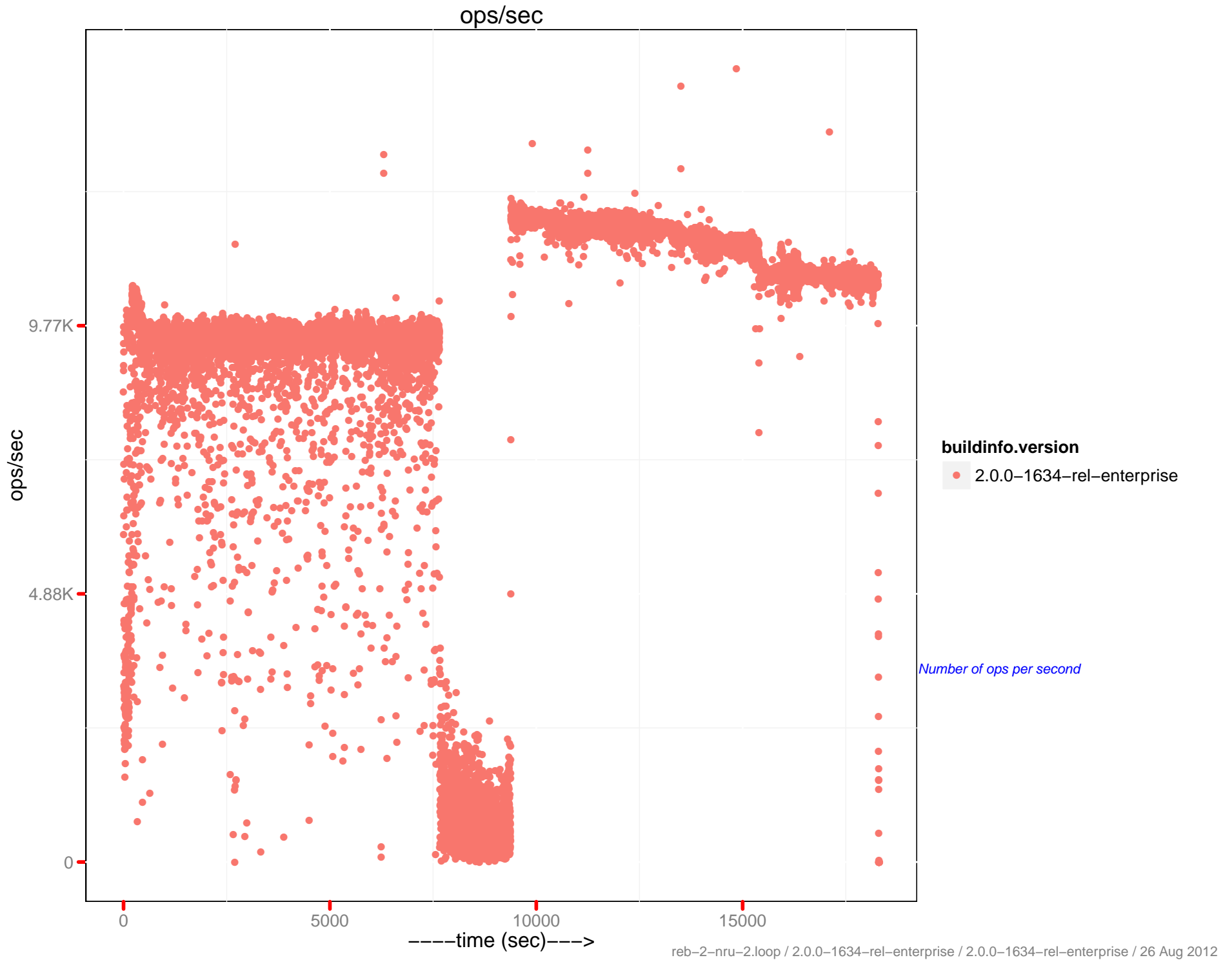
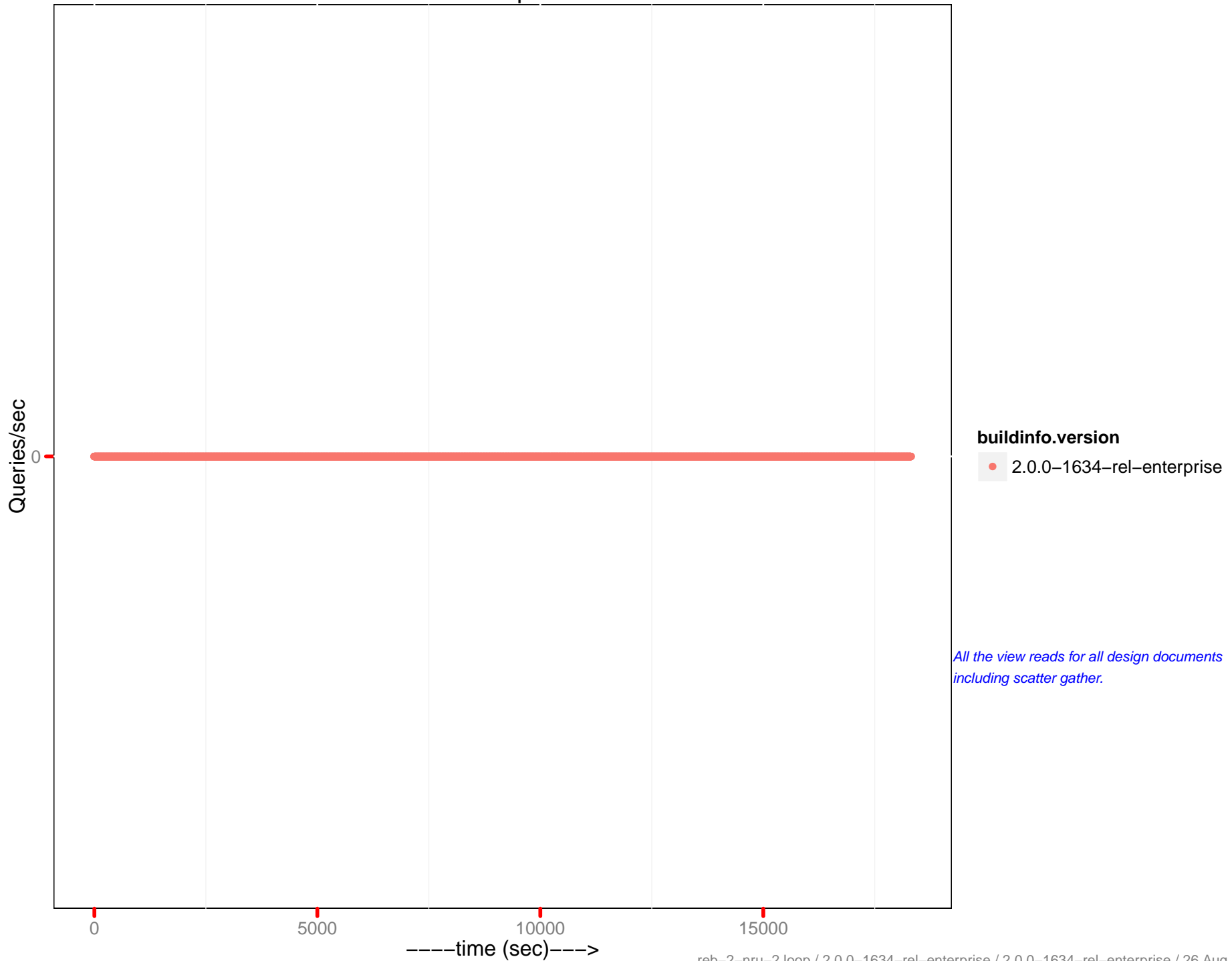


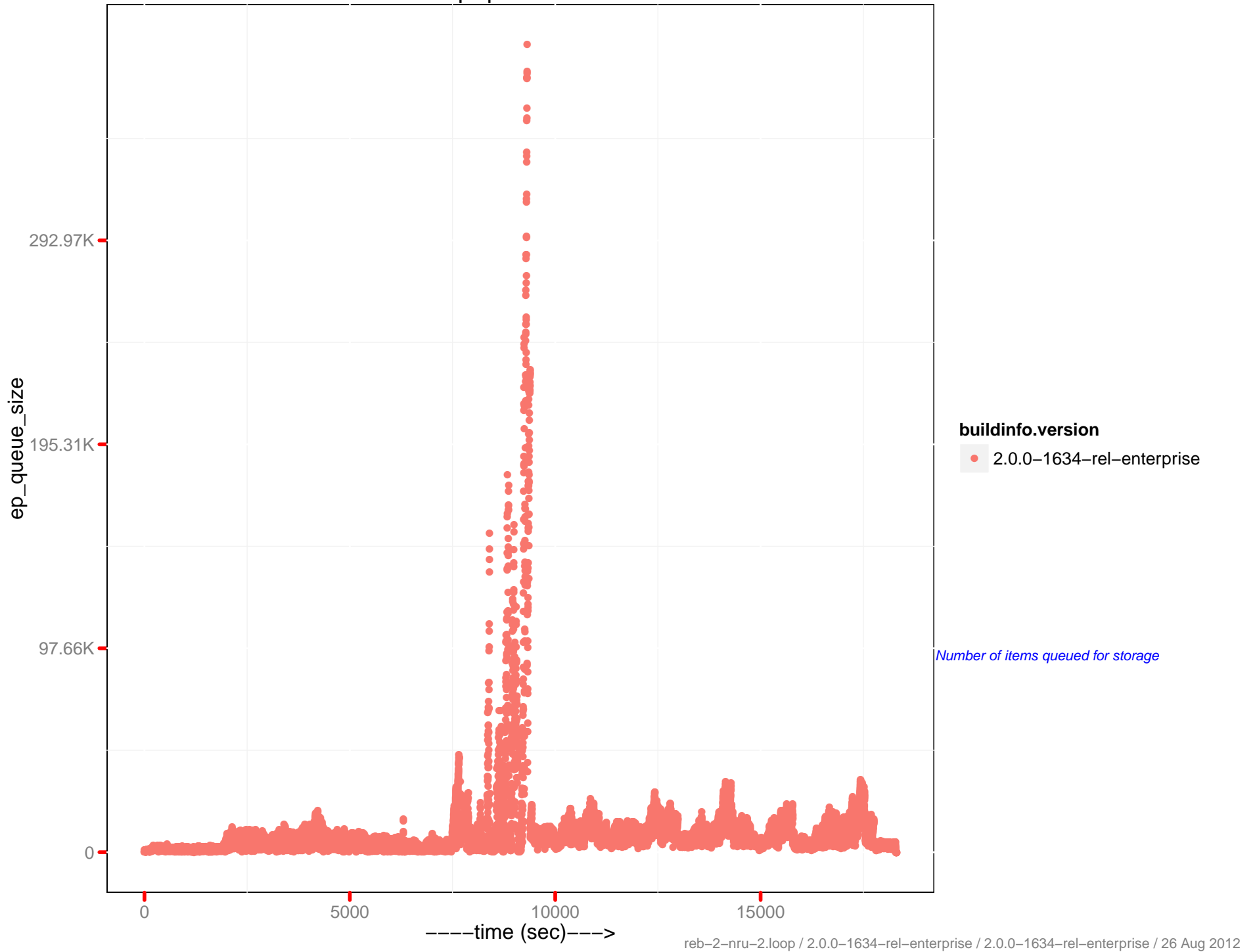
	2.0.0 – 1634	2.0.0 – 1634
<i>Runtime (in hr)</i>	5.09	NA
<i>Avg. Drain Rate</i>	2.54K	NANA
<i>Peak Disk (GB)</i>	64.94	NA
<i>Peak Memory (GB)</i>	16.88	NA
<i>Avg. OPS</i>	9.37K	NANA
<i>Avg. mem memcached (GB)</i>	16.36	NA
<i>Avg. mem beam.smp (MB)</i>	456.2	NA
<i>Avg. CPU rate (%)</i>	15.68	NA
<i>Latency-get (90th) (ms)</i>	1	NA
<i>Latency-get (95th) (ms)</i>	1.87	NA
<i>Latency-get (99th) (ms)</i>	6.11	NA
<i>Latency-set (90th) (ms)</i>	1.24	NA
<i>Latency-set (95th) (ms)</i>	2.03	NA
<i>Latency-set (99th) (ms)</i>	4.3	NA
<i>Latency-query (80th) (ms)</i>	NA	NA
<i>Latency-query (90th) (ms)</i>	NA	NA
<i>Latency-query (95th) (ms)</i>	NA	NA
<i>Latency-query (99th) (ms)</i>	NA	NA
<i>Latency-query (99.9th) (ms)</i>	NA	NA
<i>Avg. QPS</i>	0	NA
<i>Rebalance Time (sec)</i>	1740.92	NA
<i>Testrunner Version</i>	1b0aaac	NA



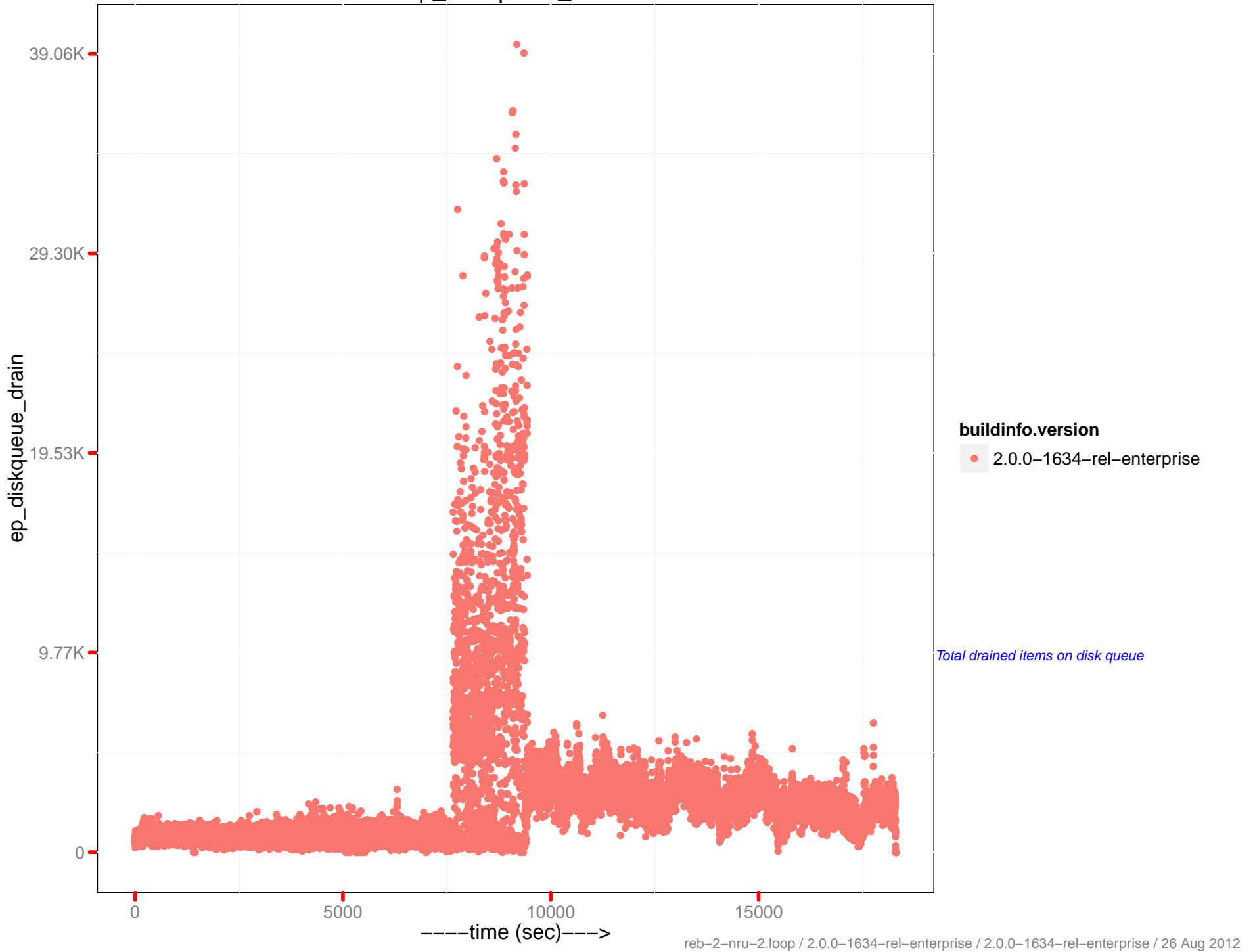
View read per sec.



ep queue size

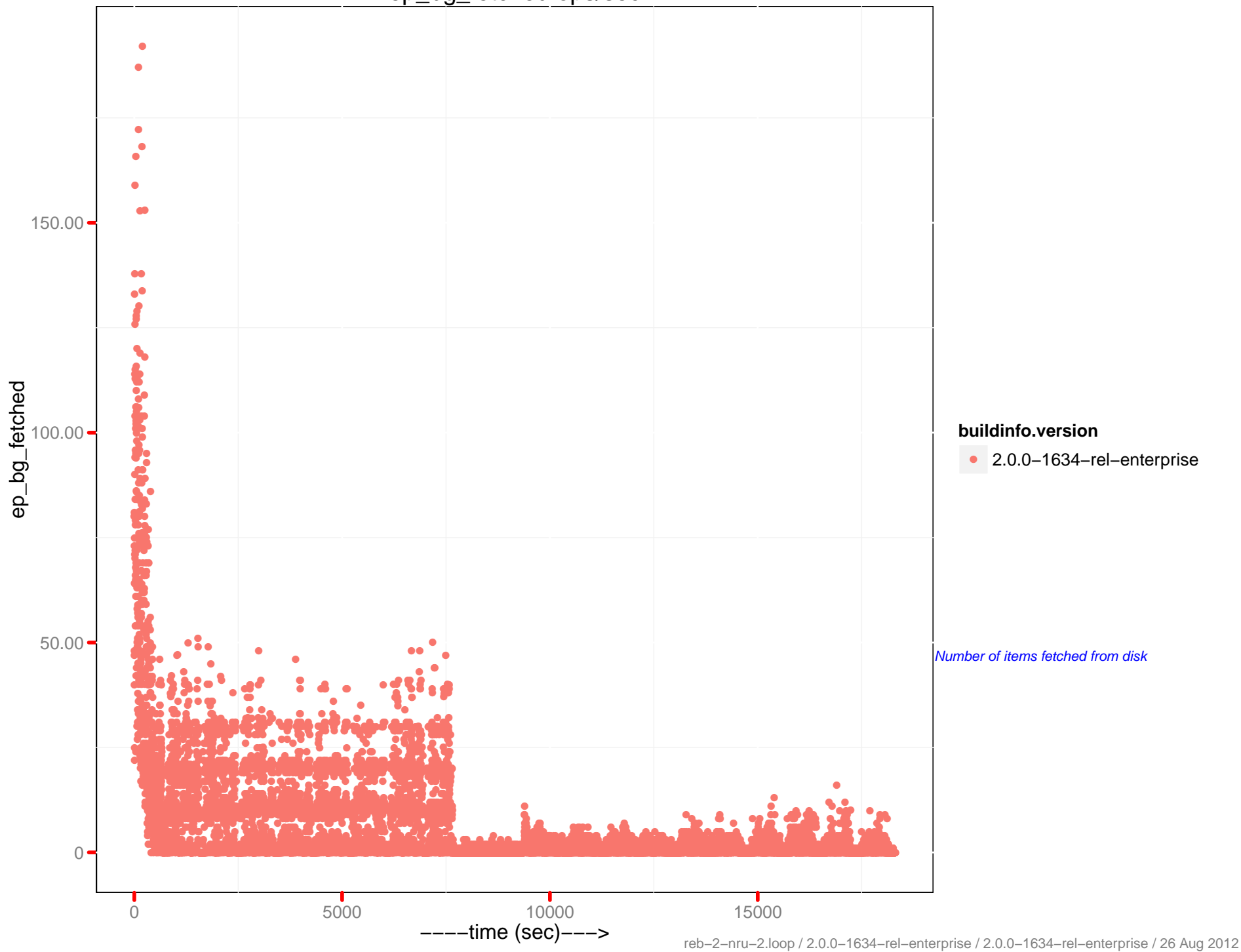


ep_diskqueue_drain

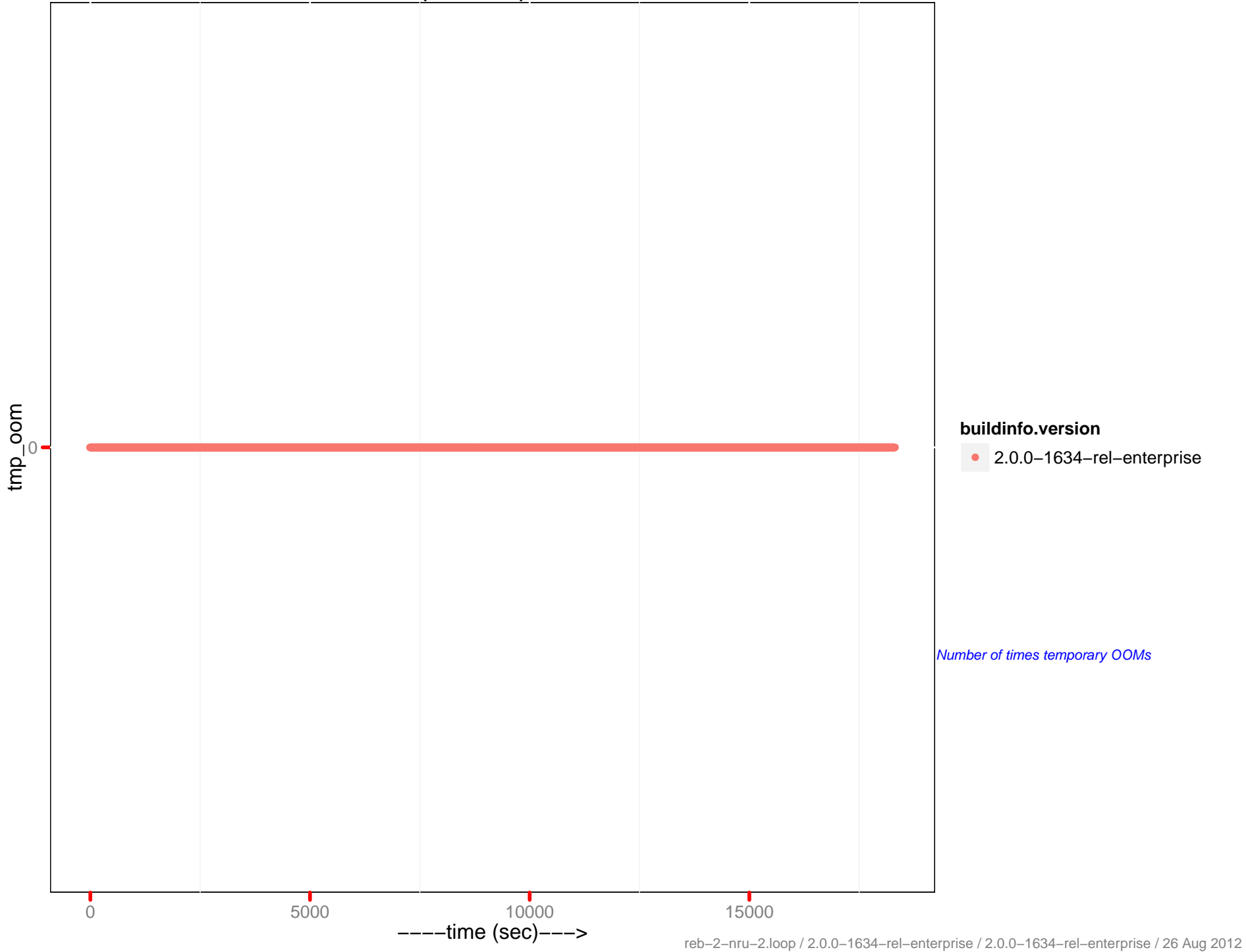


Total drained items on disk queue

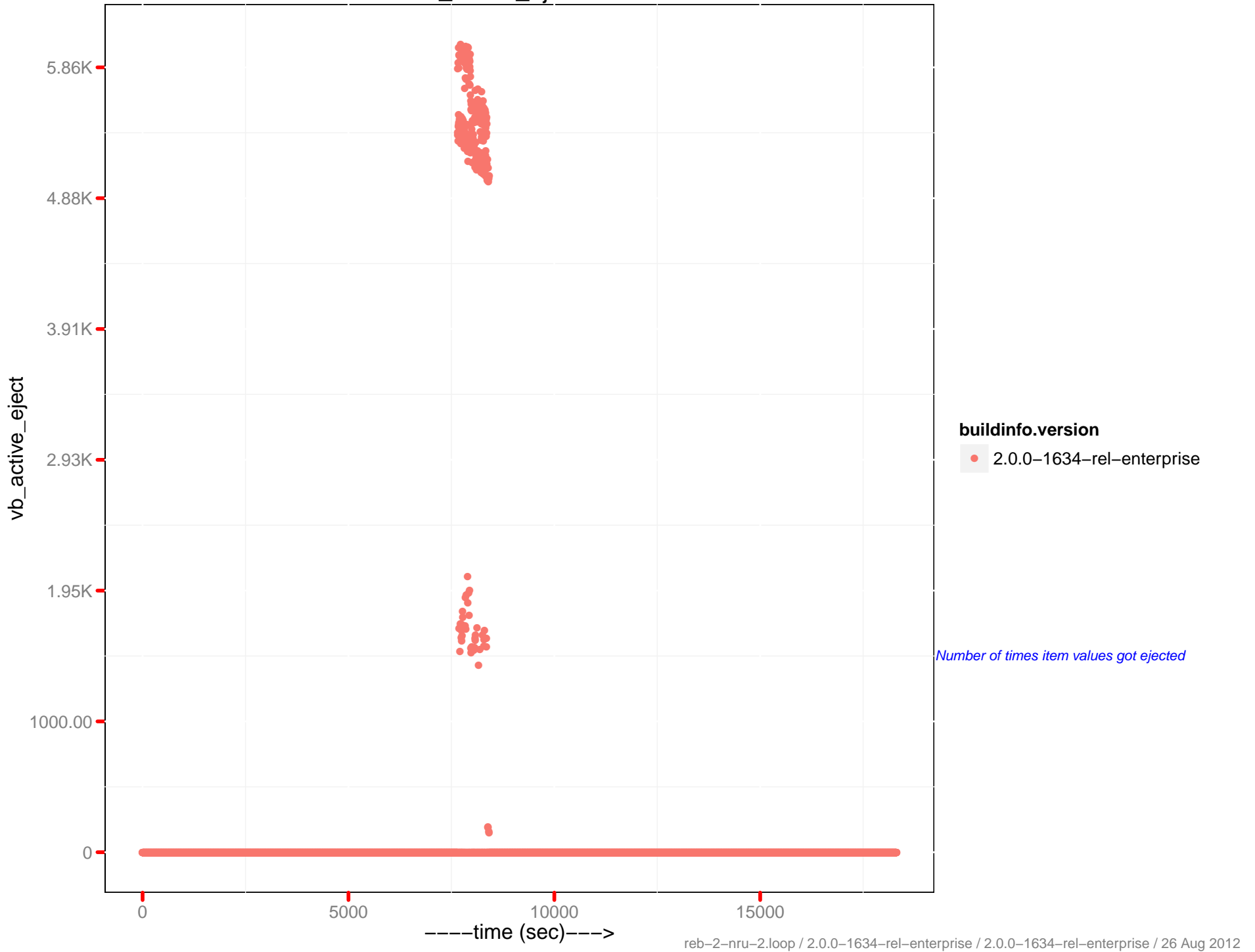
ep_bg_fetched ops/sec



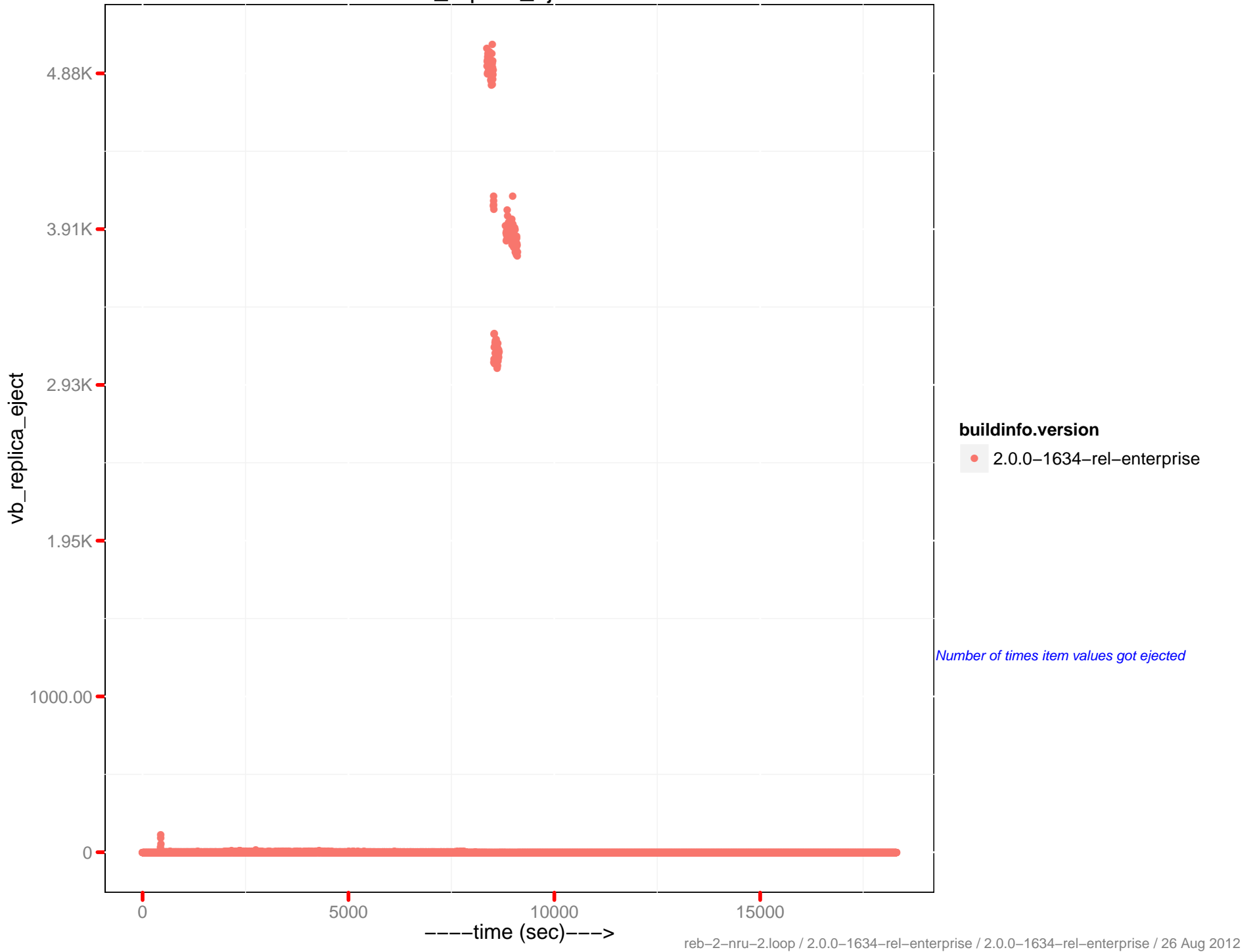
tmp_oom ops/sec



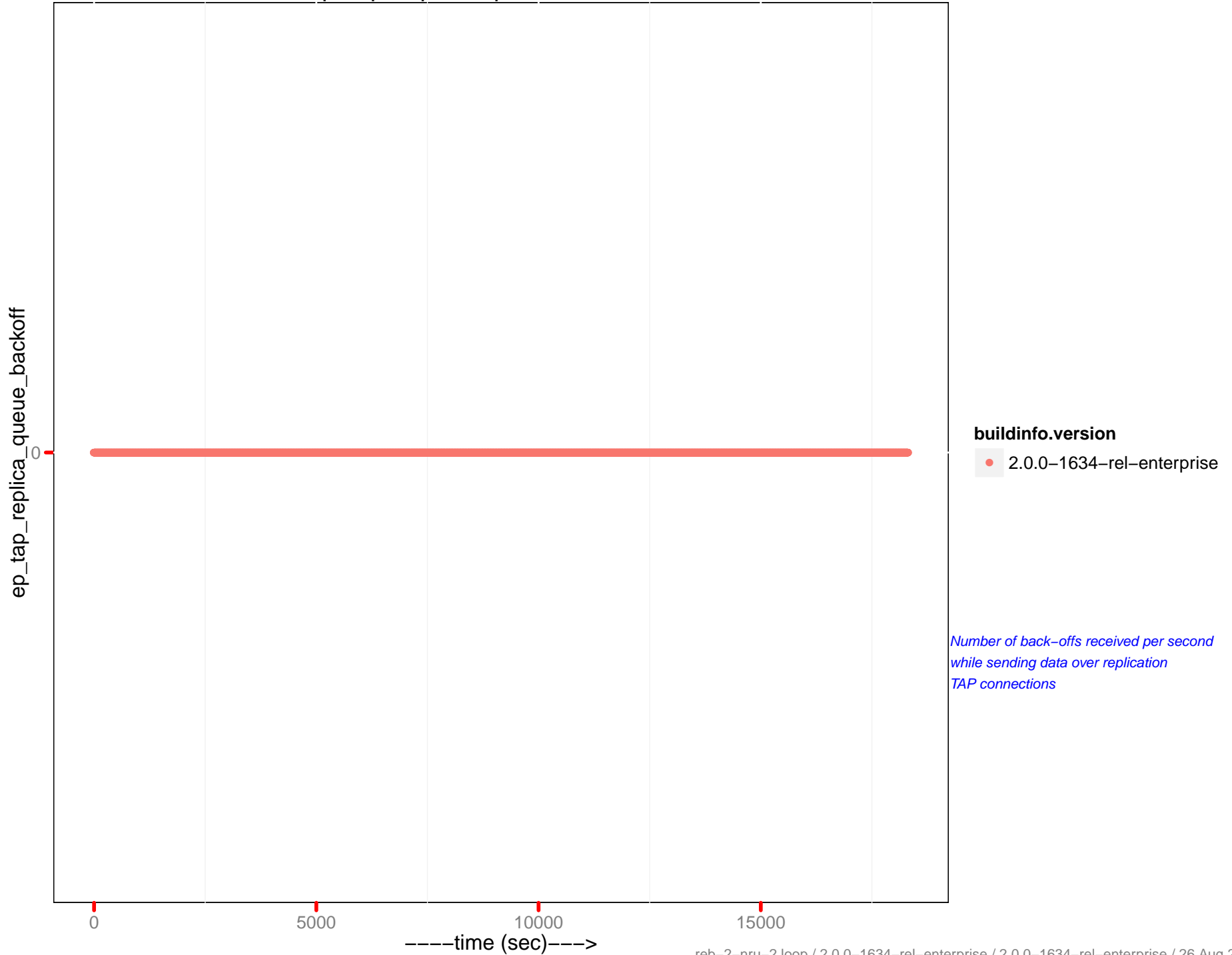
vb_active_eject/sec



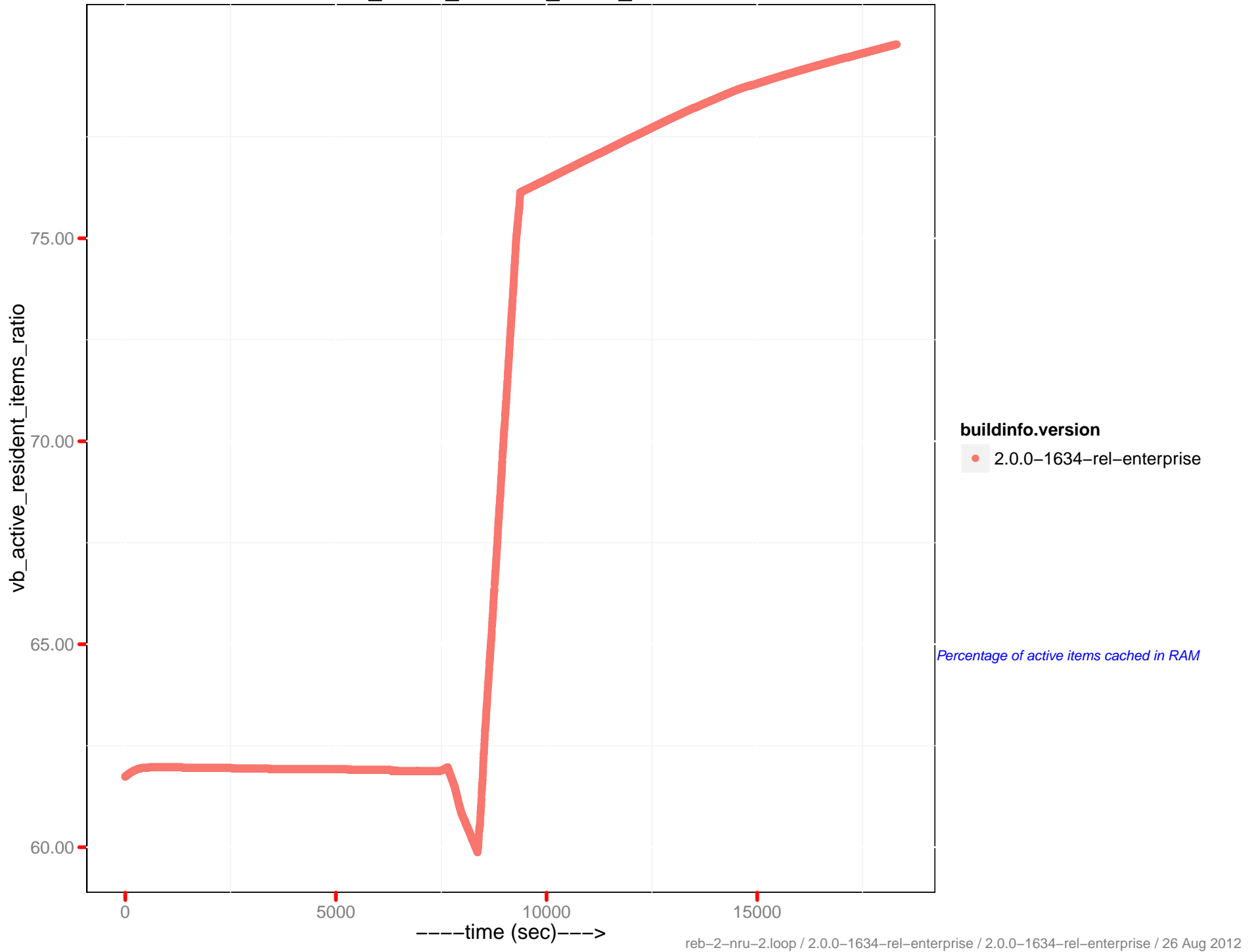
vb_replica_eject/sec



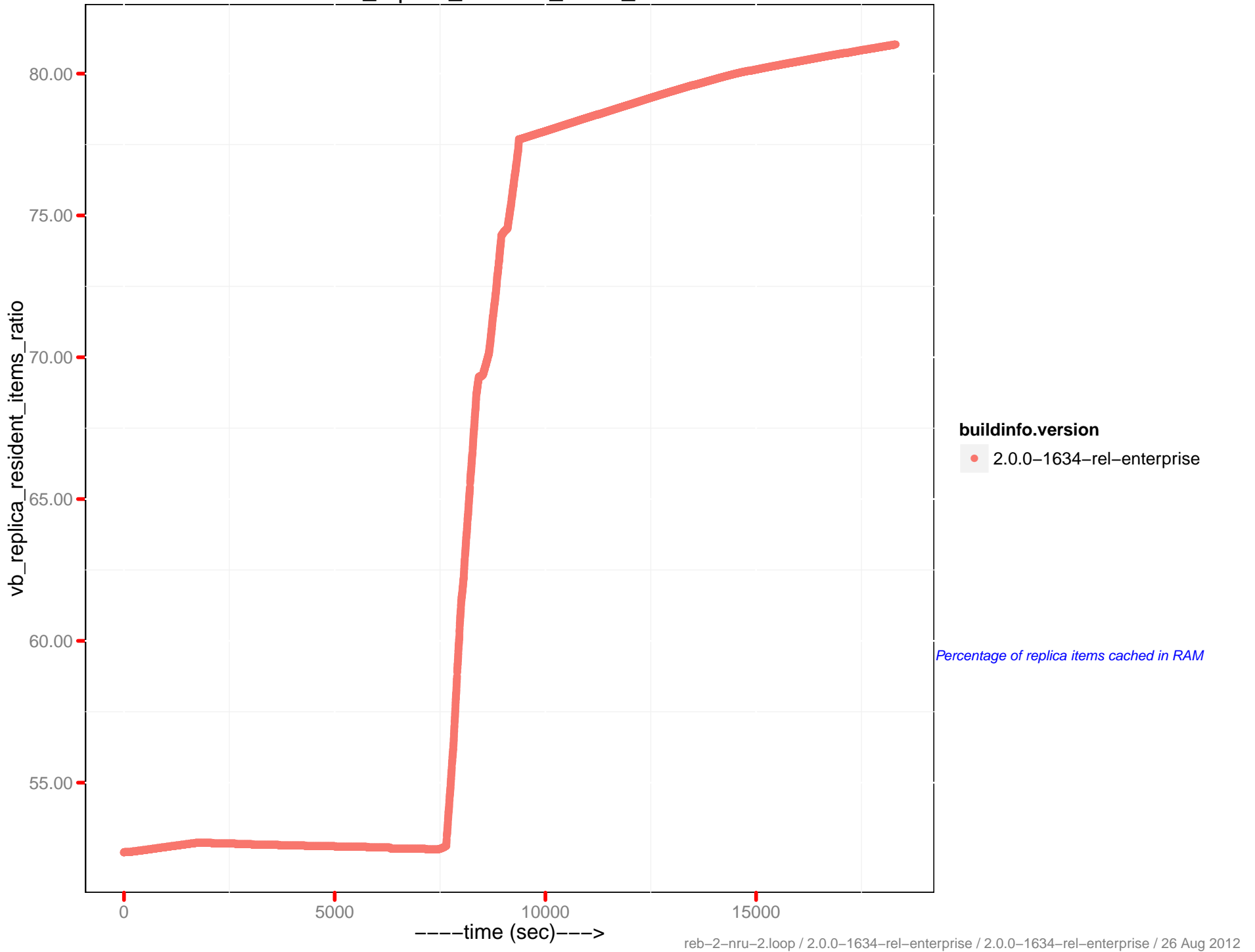
ep_tap_replica_queue_backoff/sec



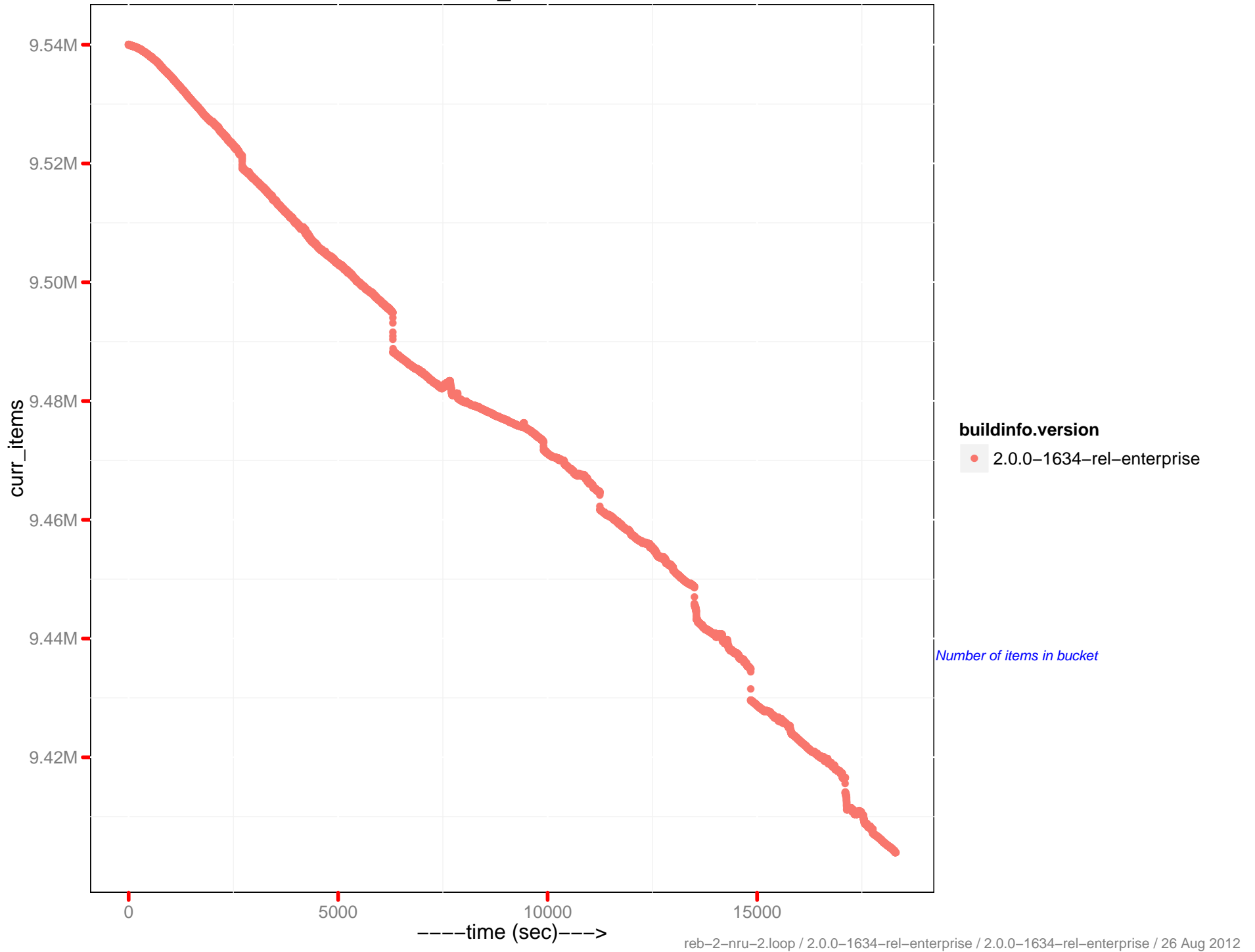
vb_active_resident_items_ratio



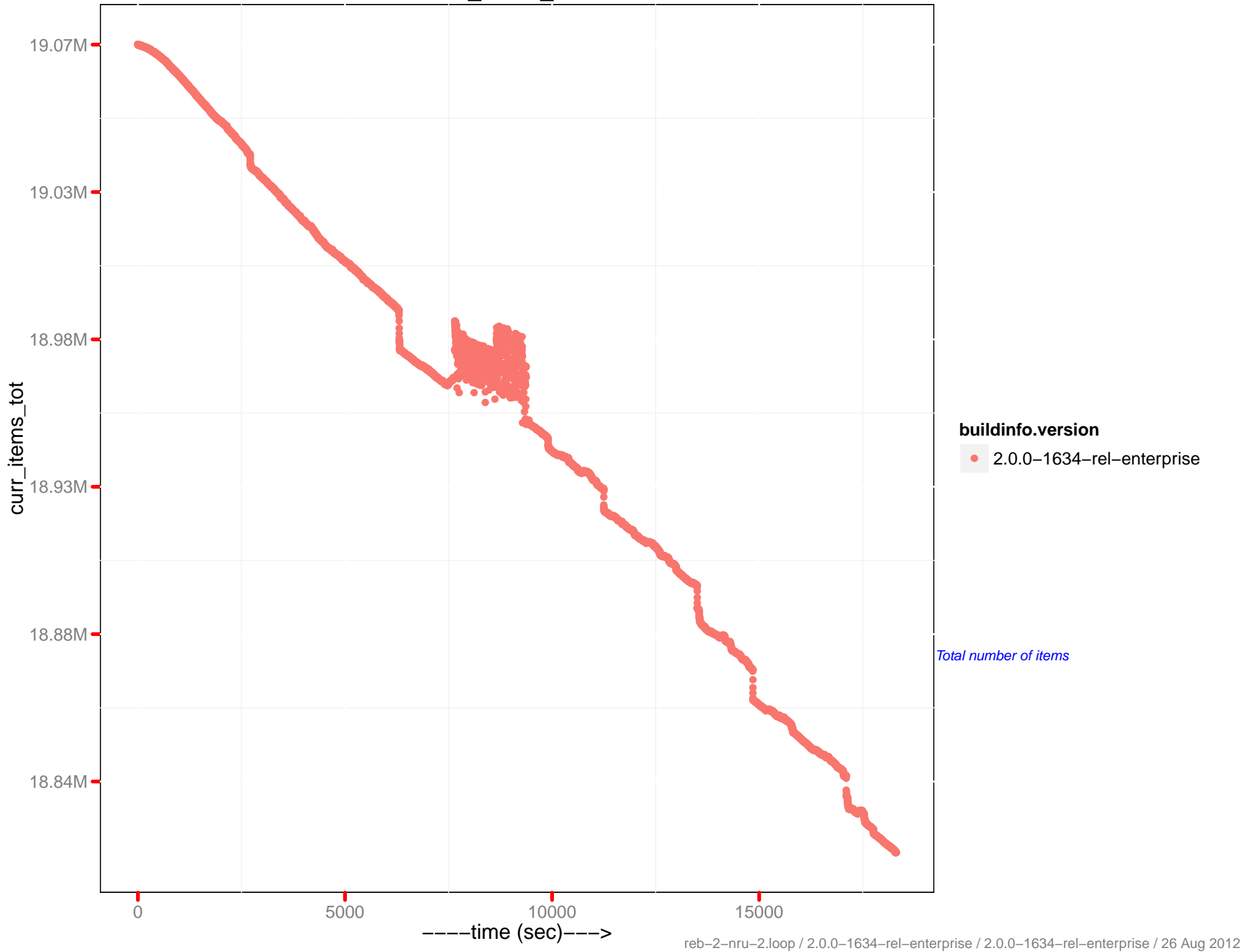
vb_replica_resident_items_ratio



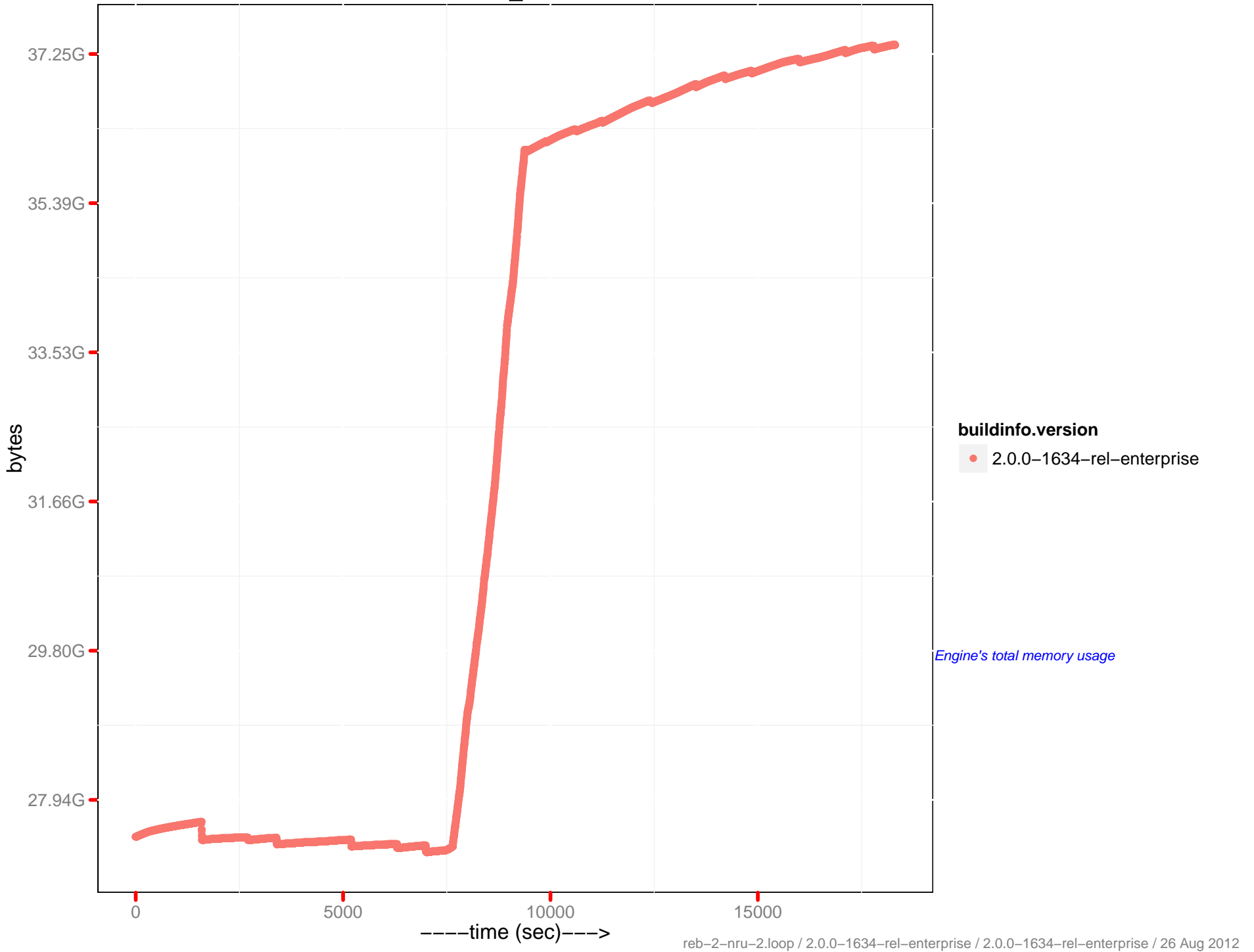
curr_items



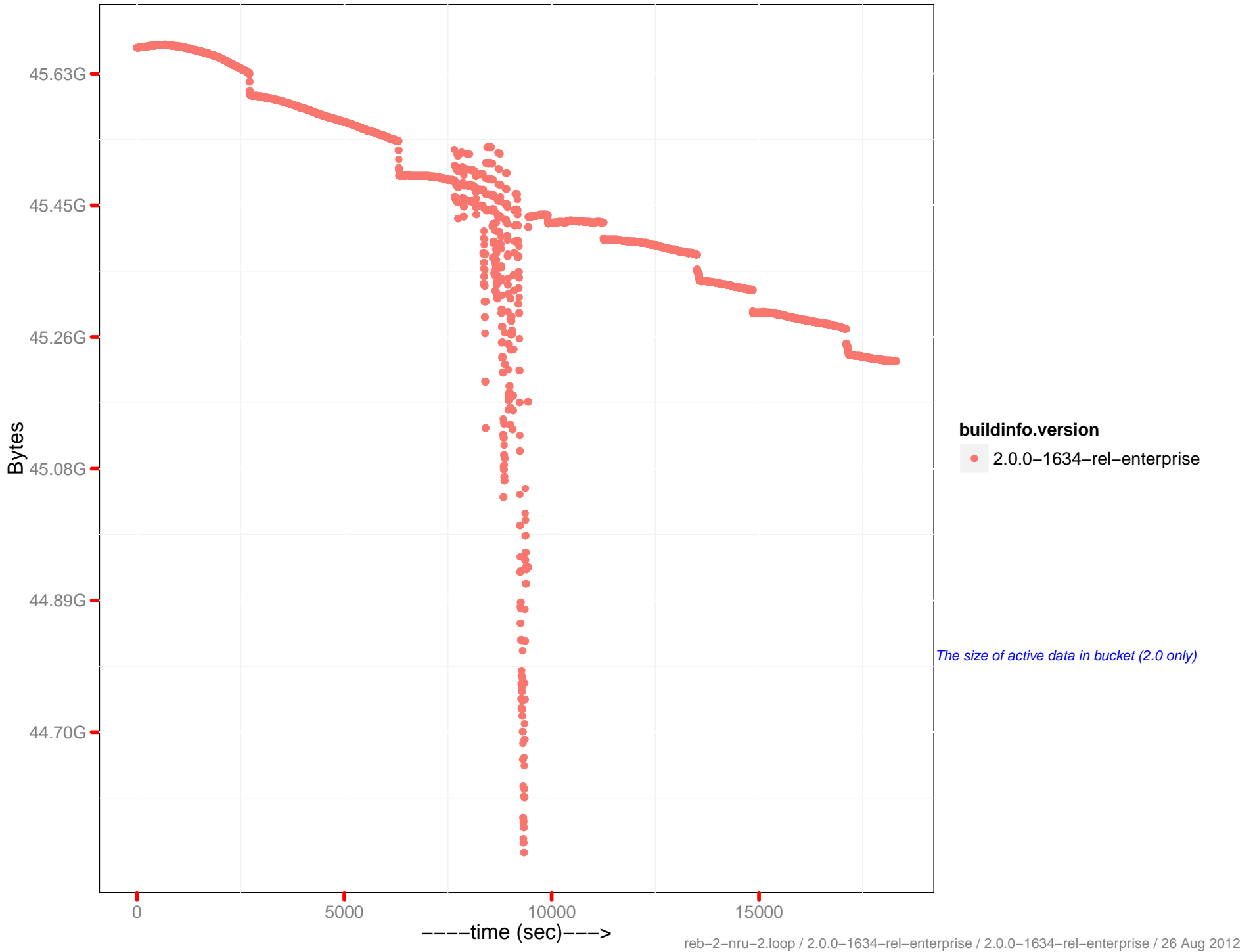
cur_items_total



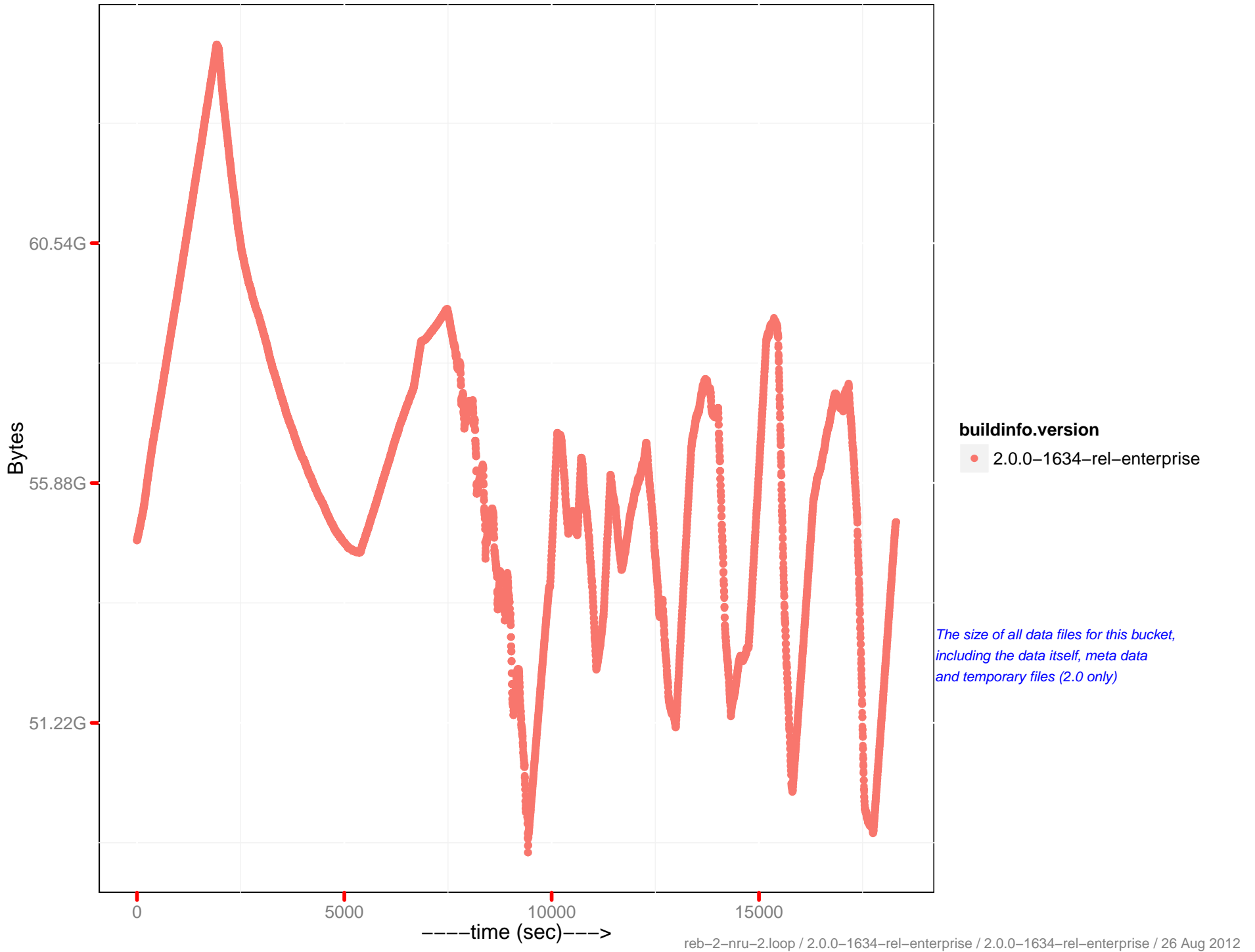
mem_used



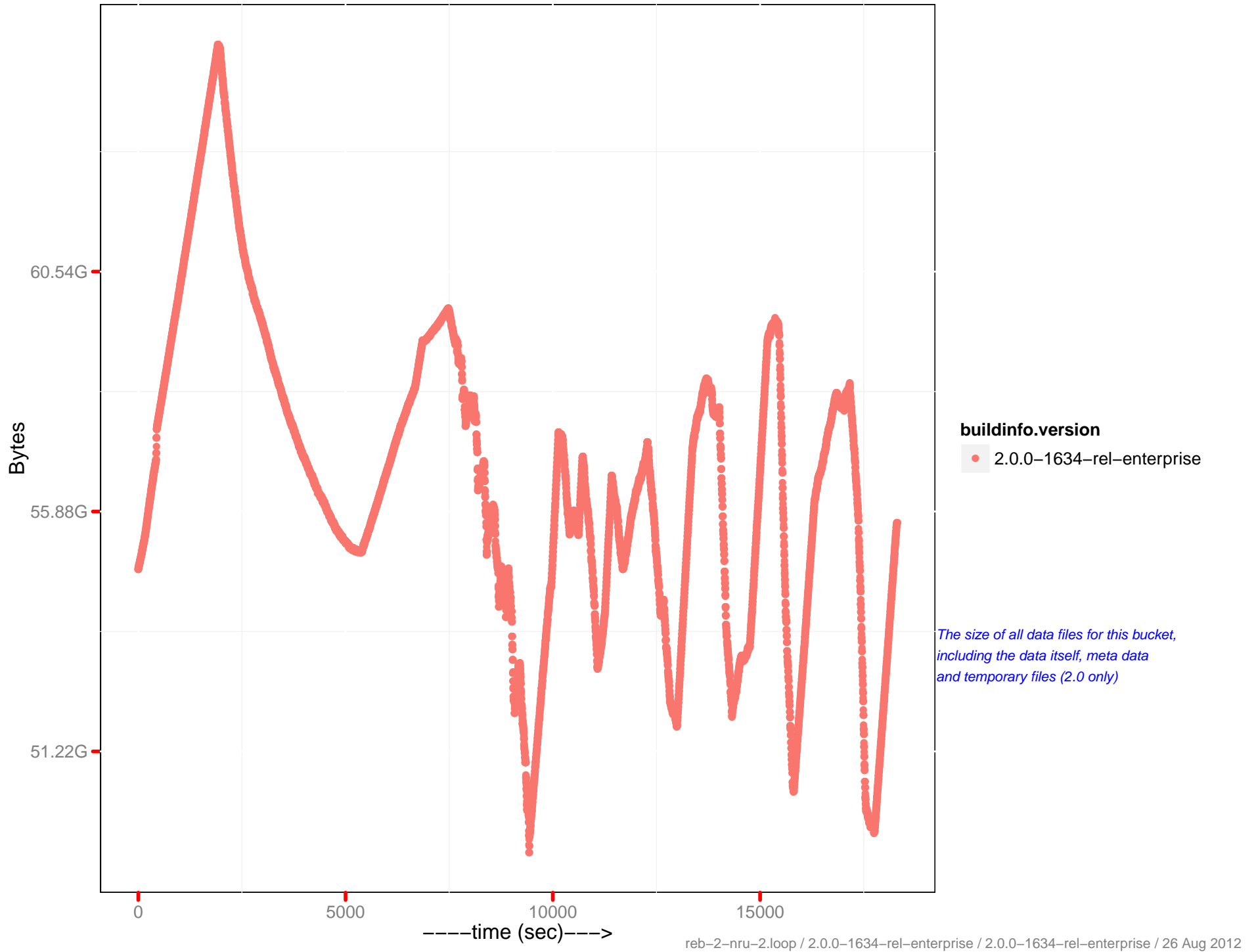
Docs data size



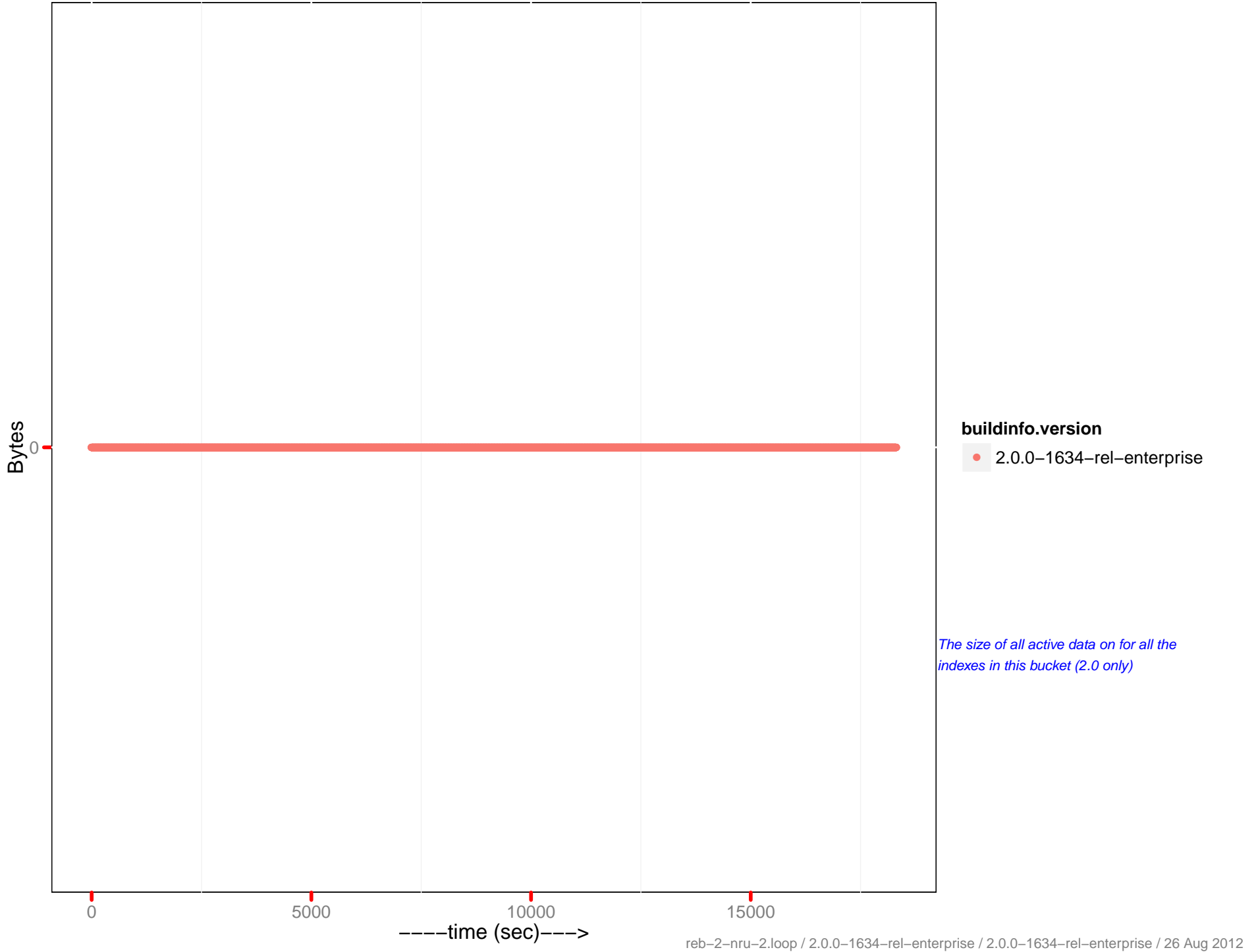
Docs disk size



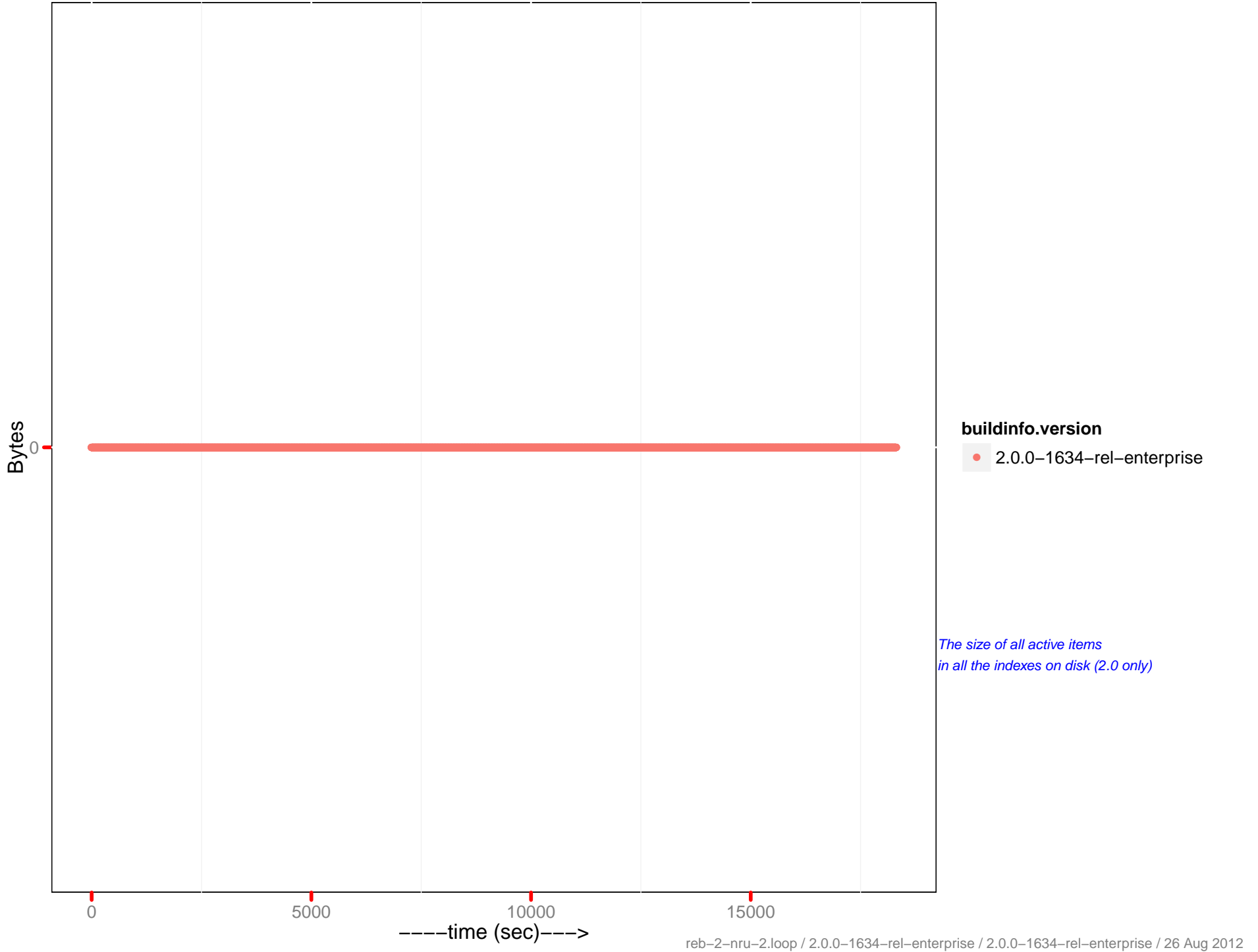
Docs actual disk size



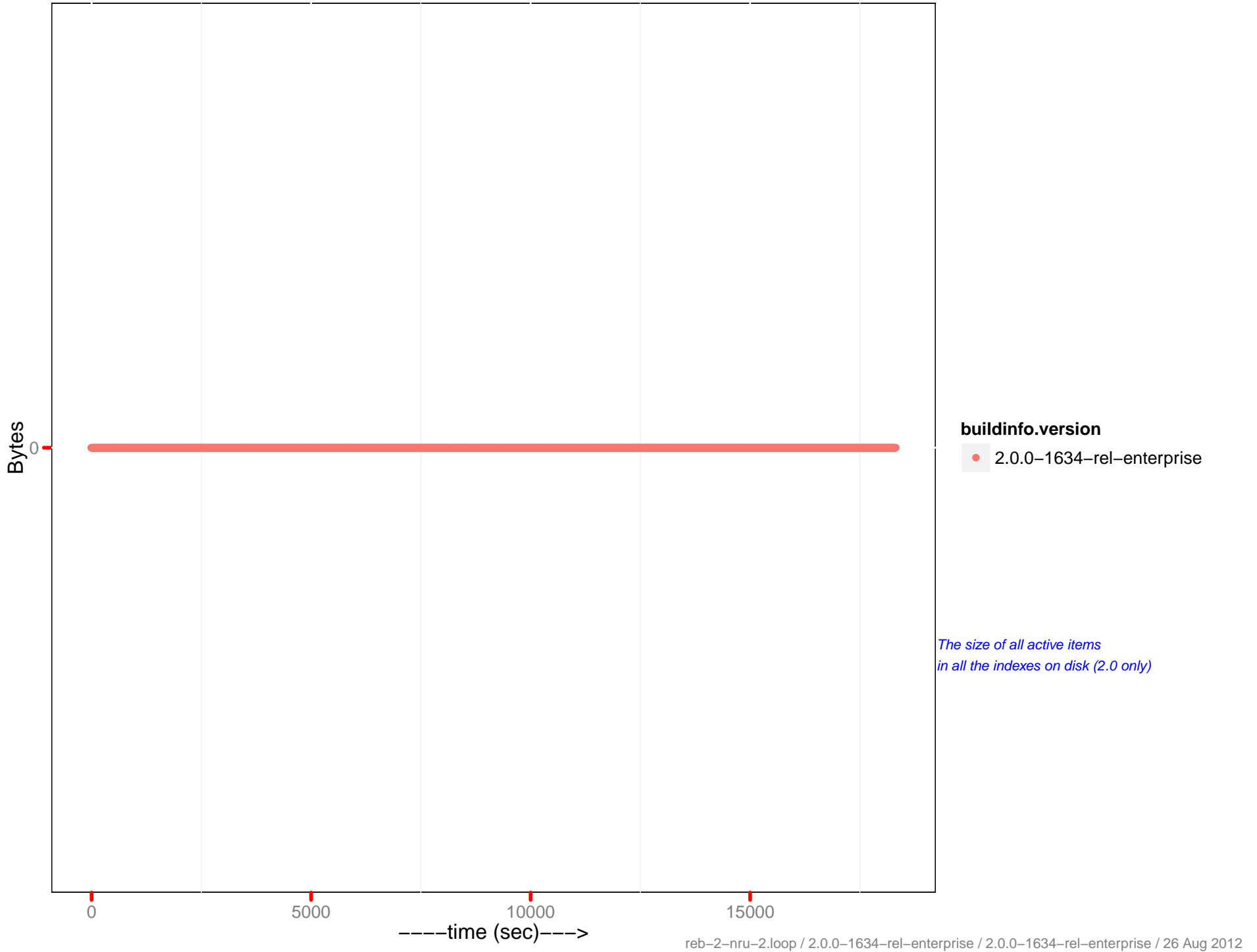
Views data size



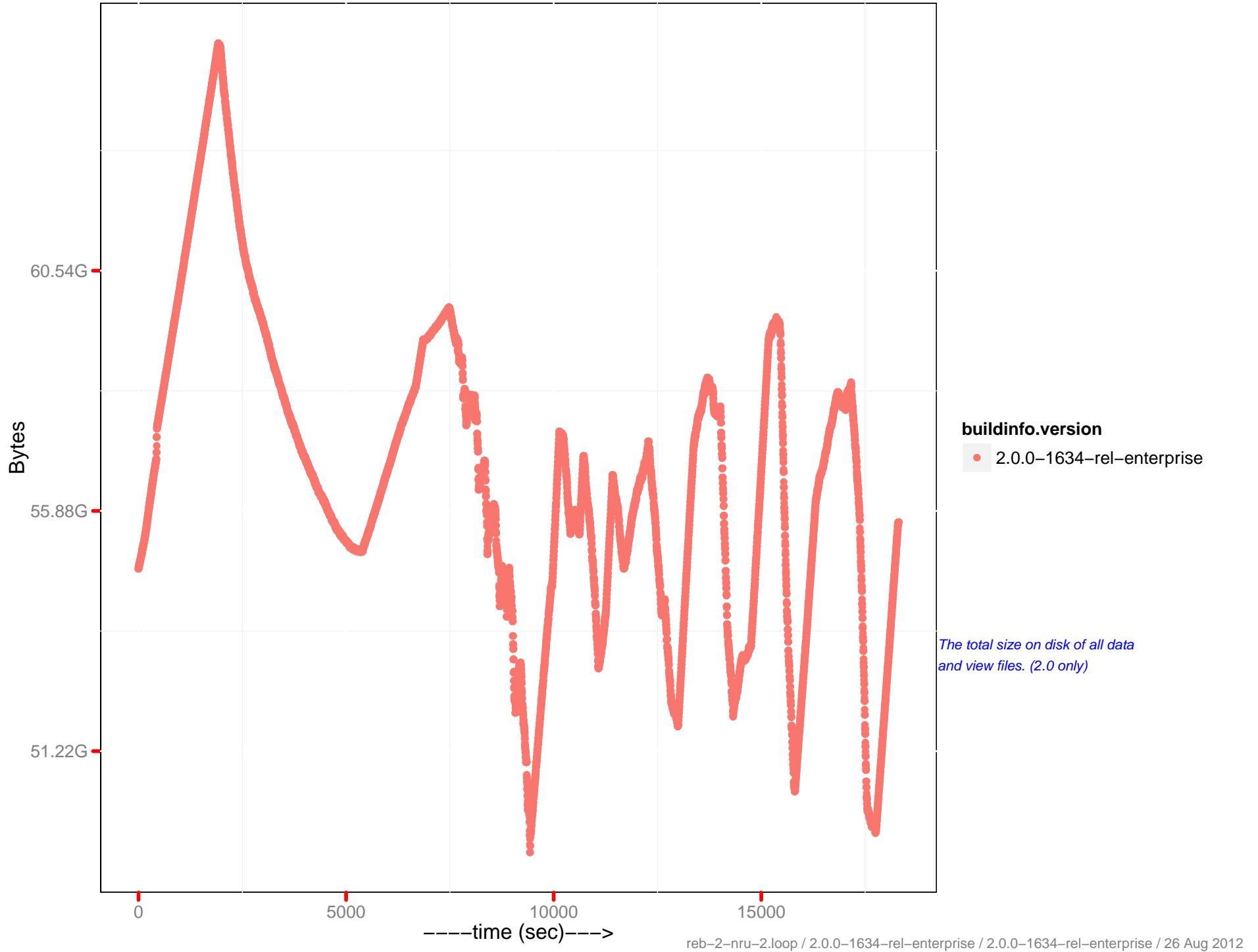
Views disk size



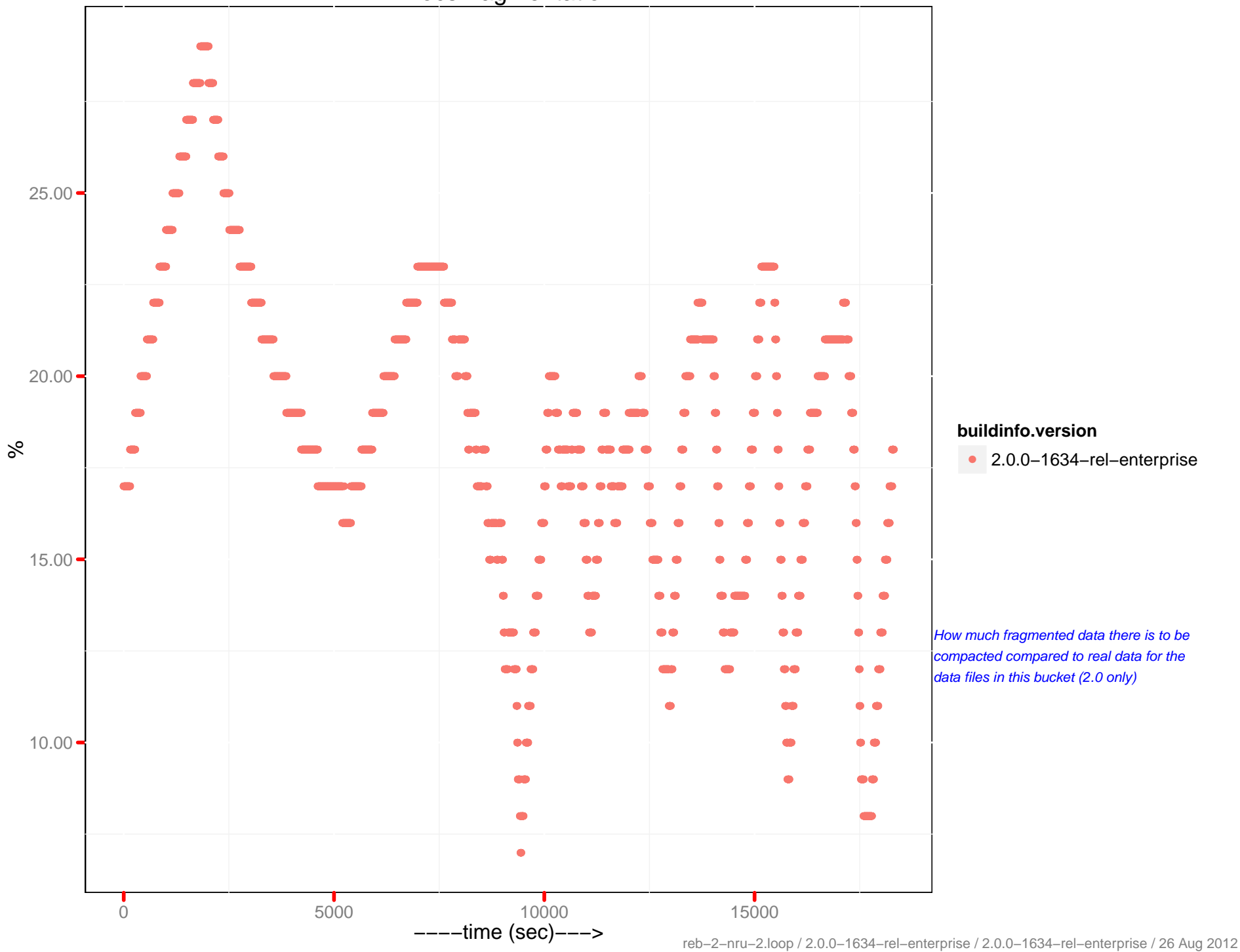
Views actual disk size



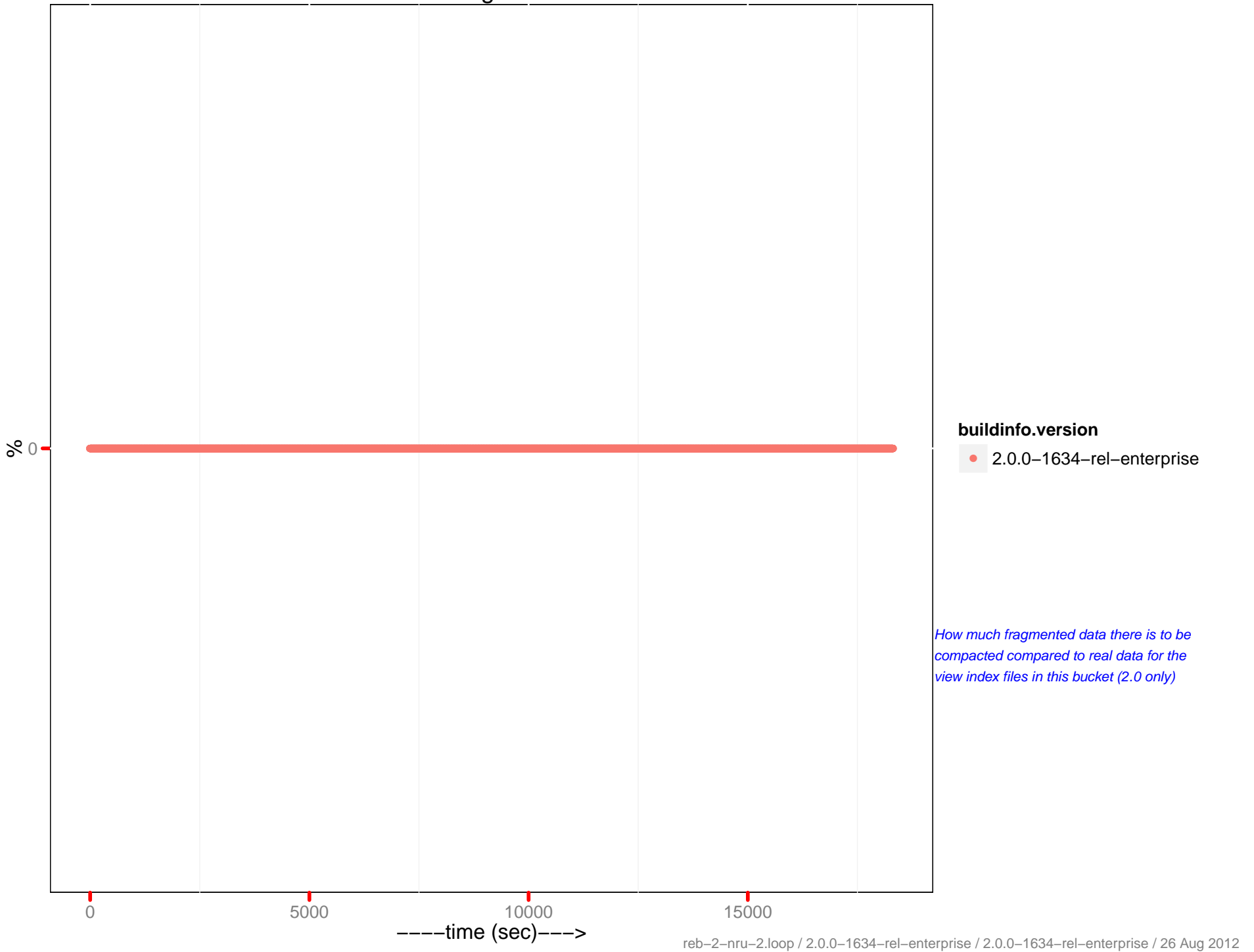
Total disk size



Docs fragmentation



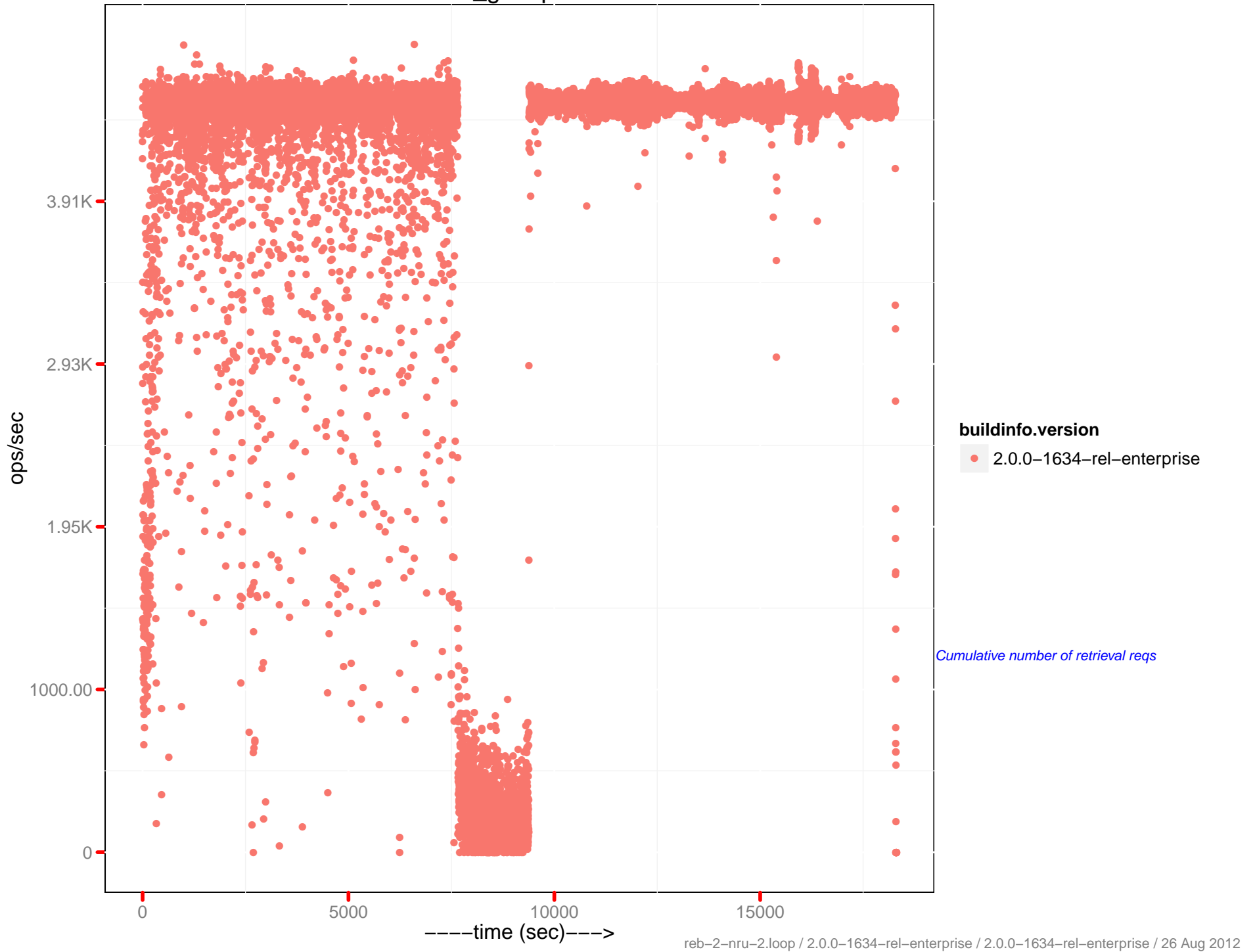
Views fragmentation



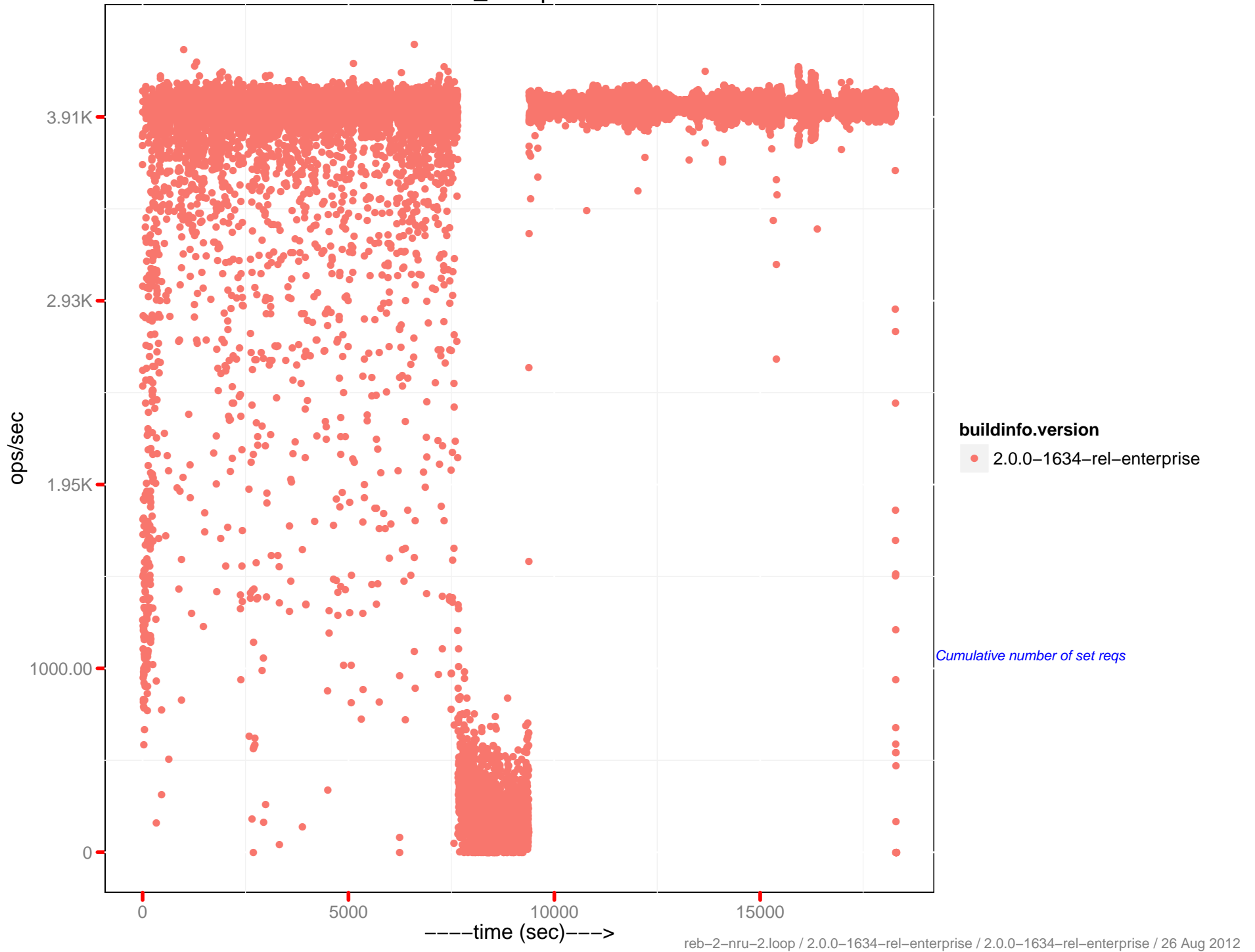
buildinfo.version
● 2.0.0-1634-rel-enterprise

How much fragmented data there is to be compacted compared to real data for the view index files in this bucket (2.0 only)

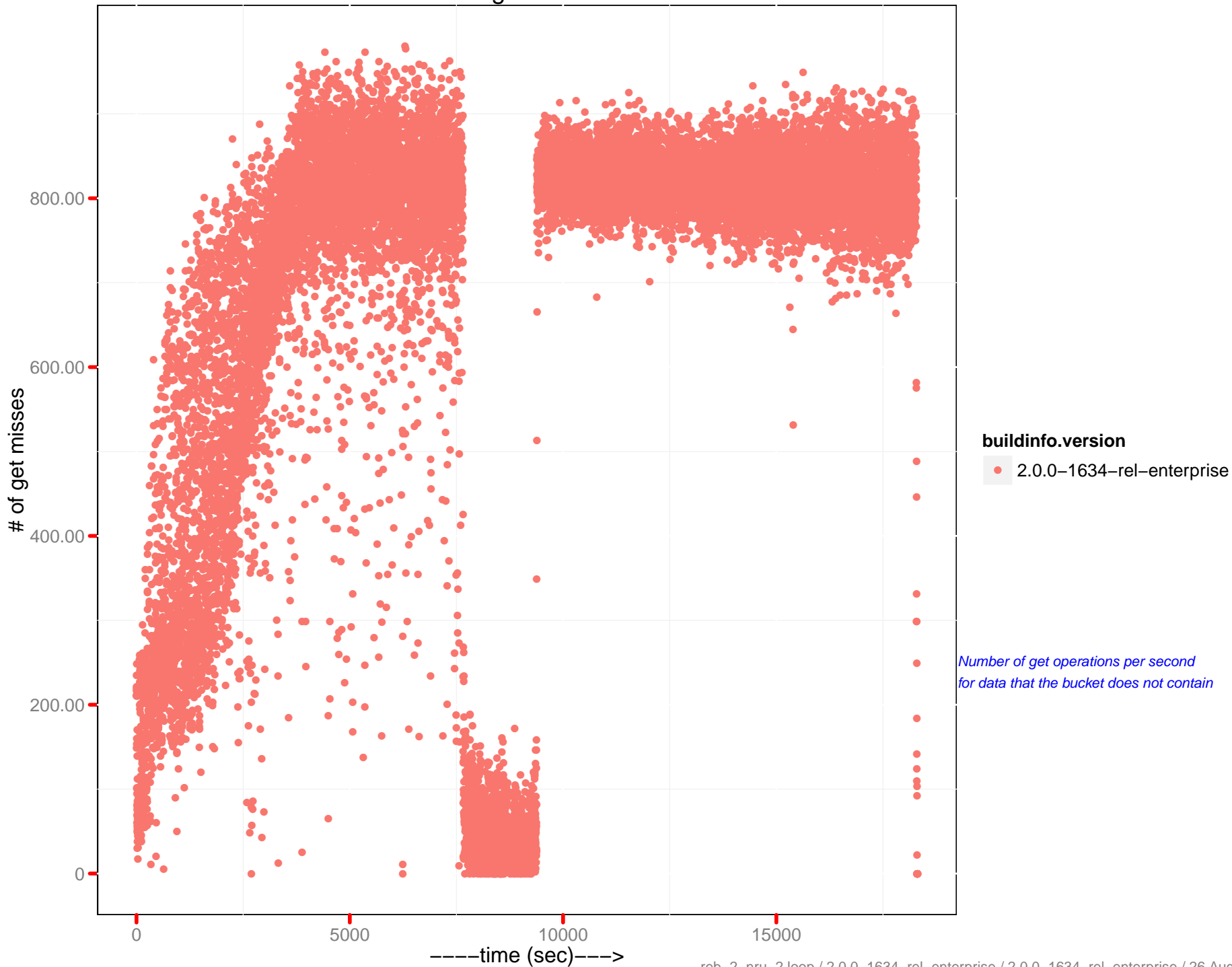
cmd_get ops/sec



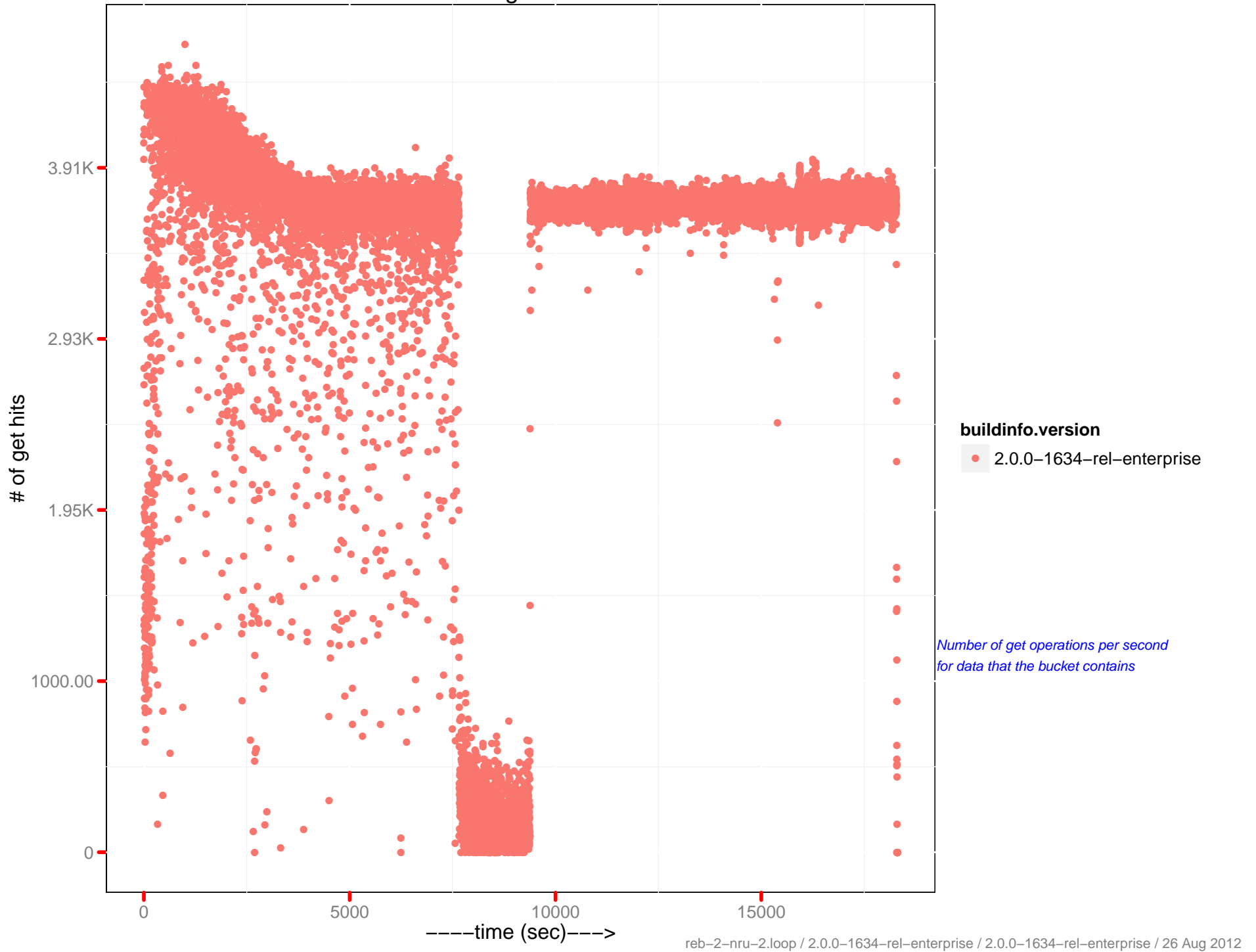
cmd_set ops/sec



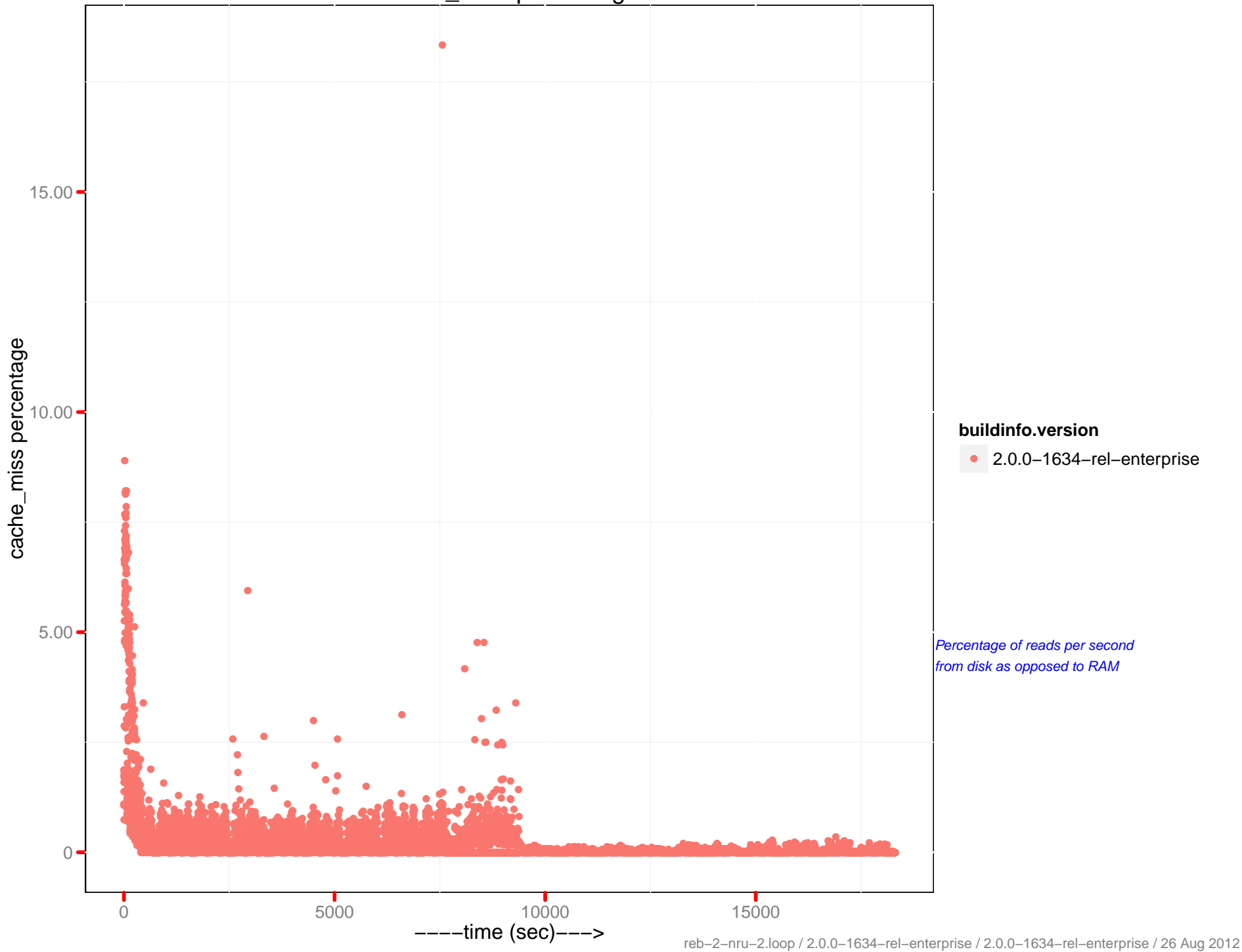
of get misses



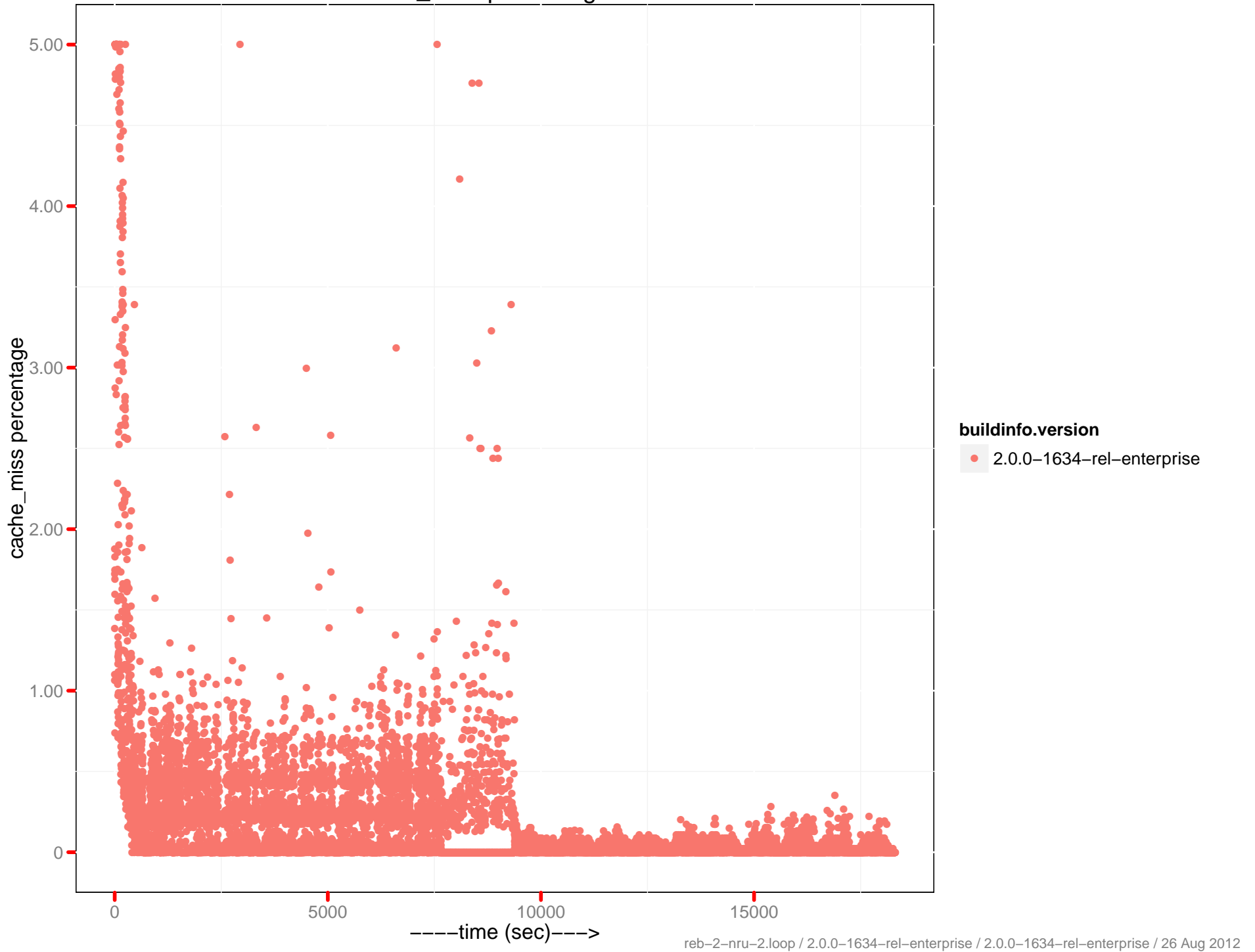
of get hits



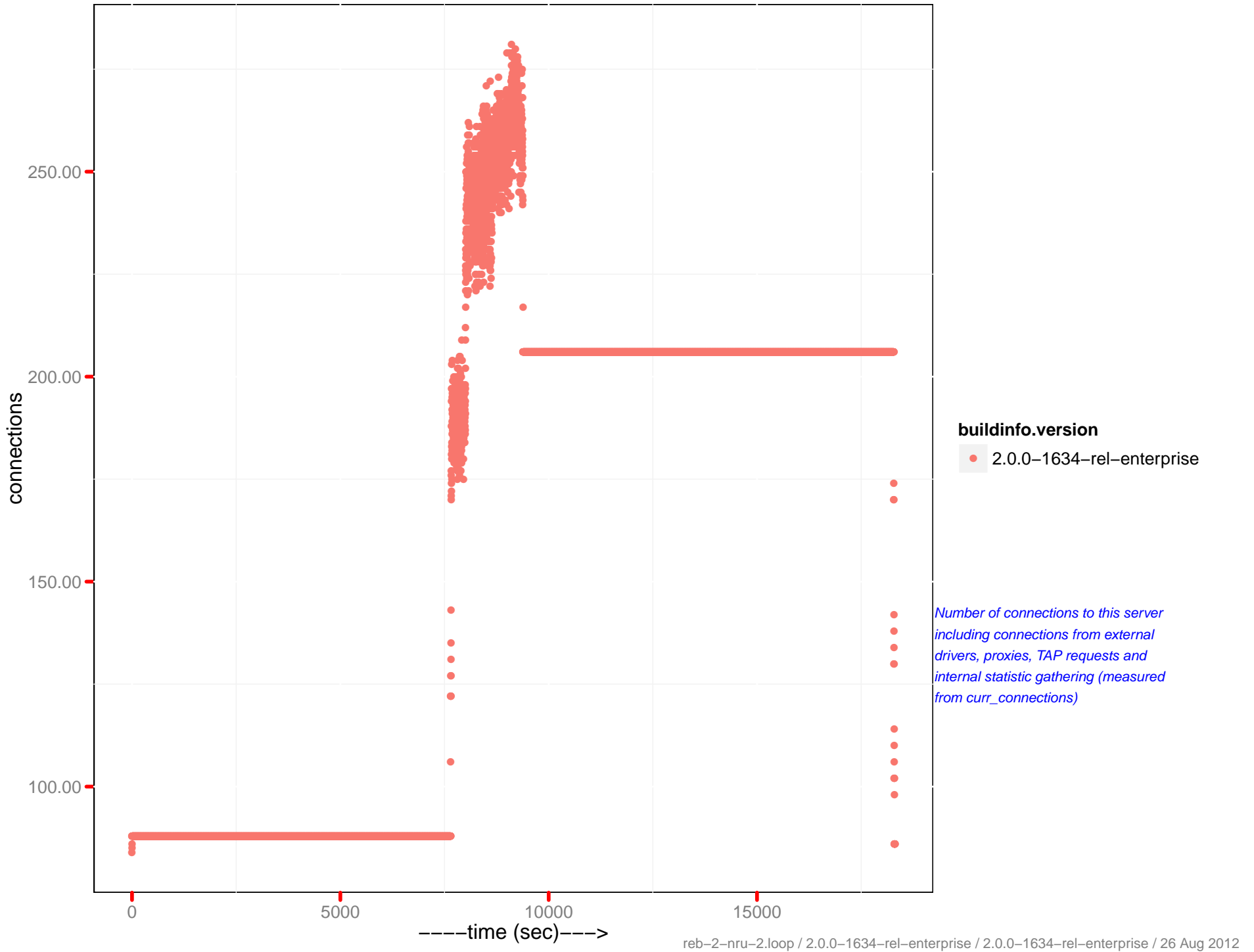
cache_miss percentage



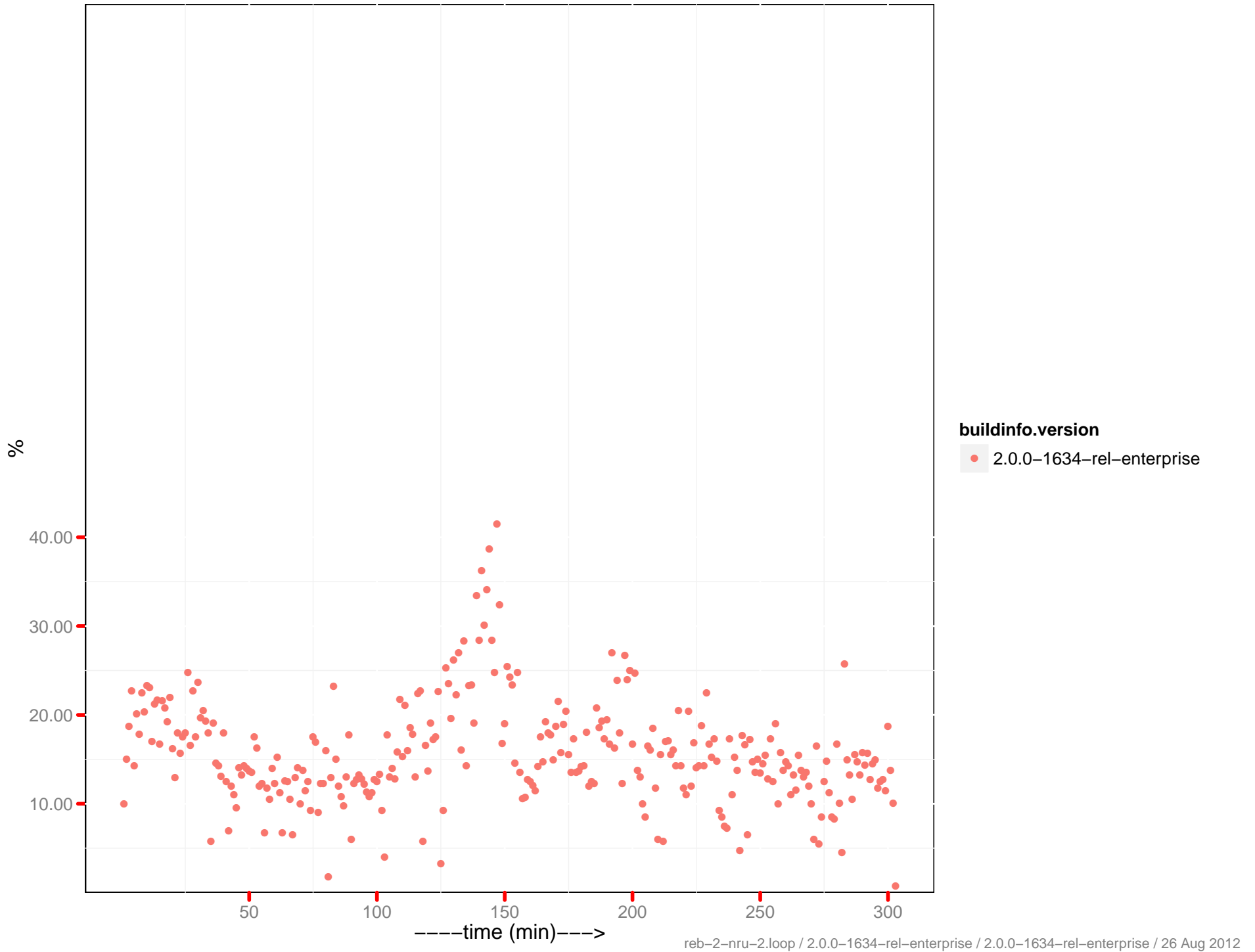
cache_miss percentage 0-5



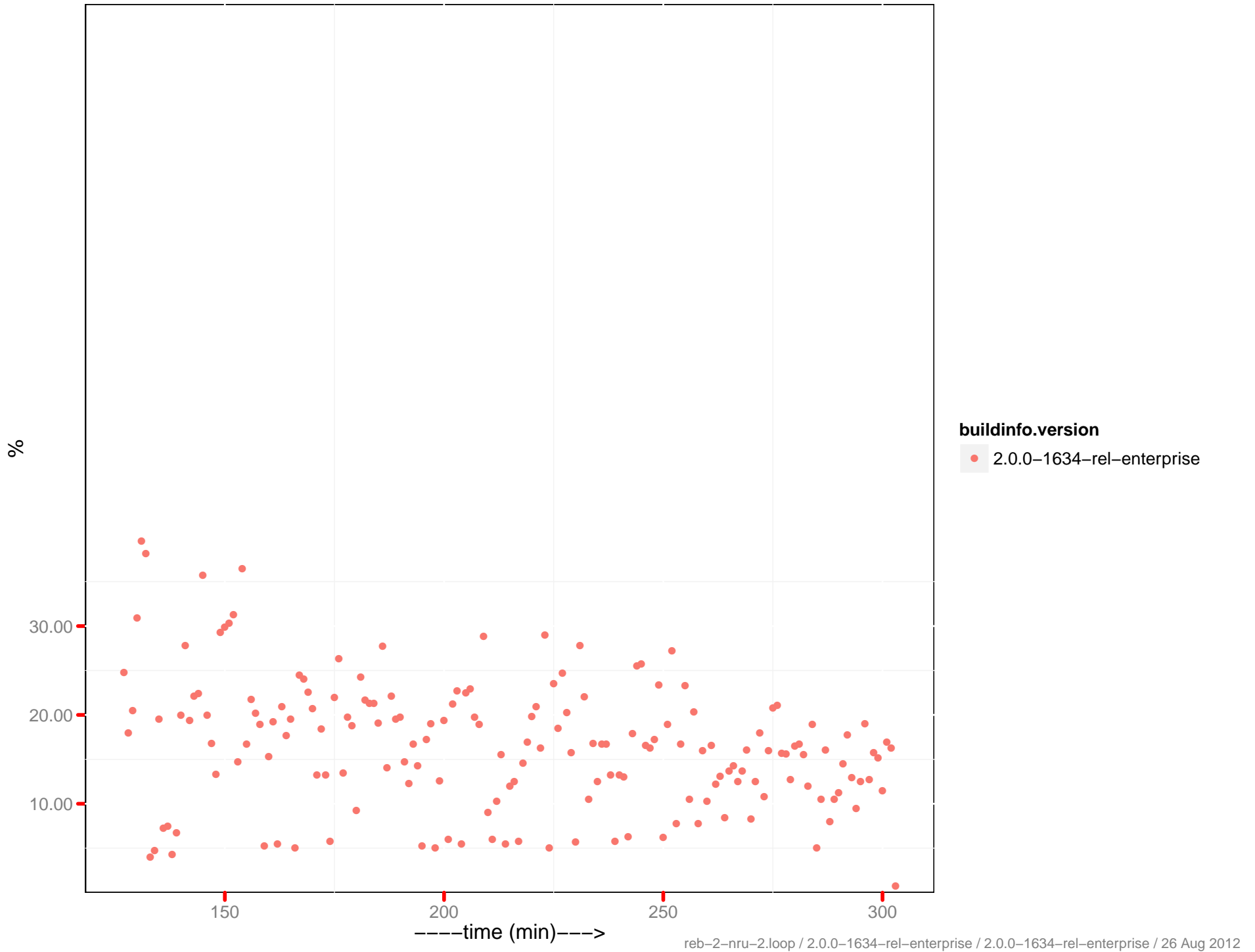
Number of connections



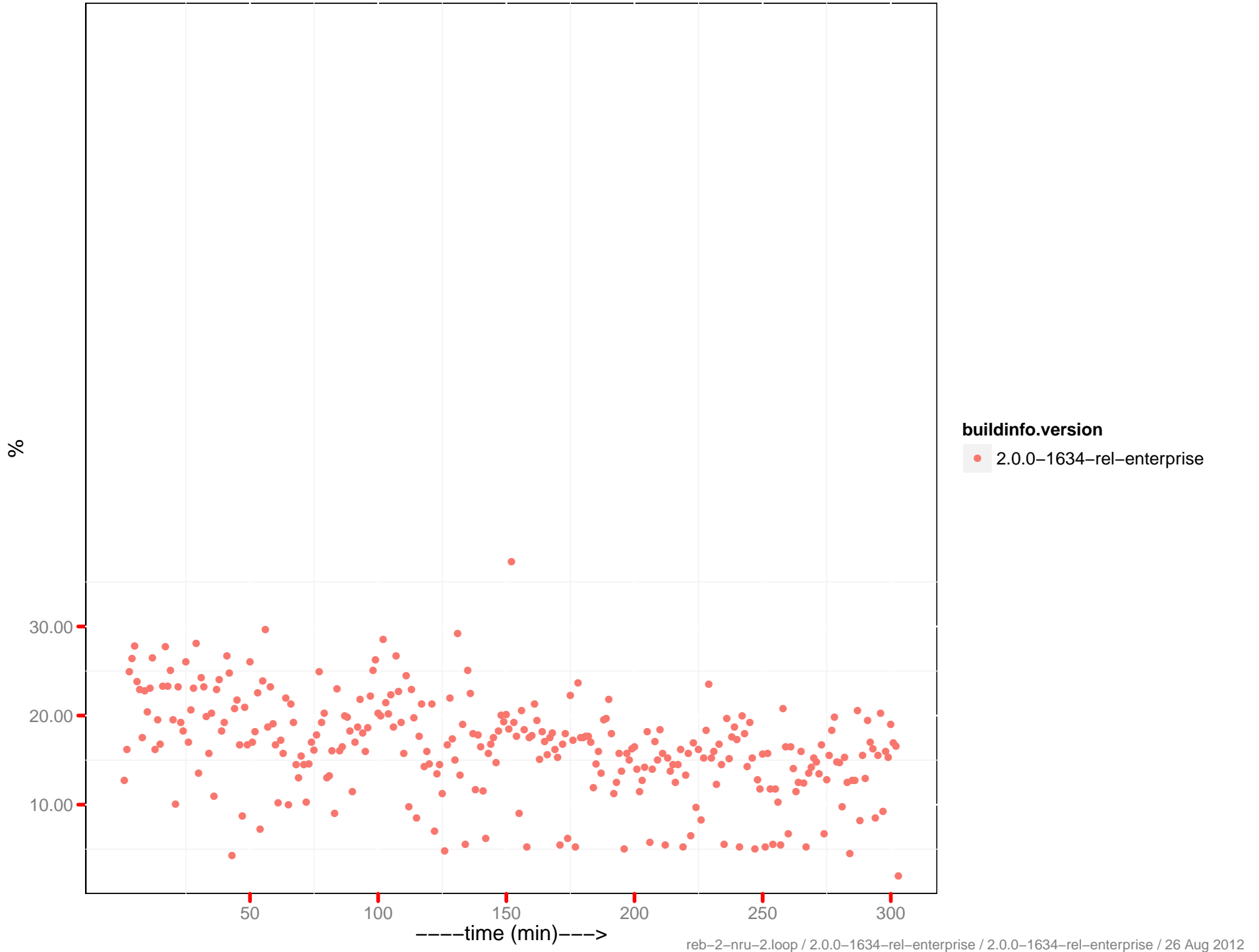
CPU utilization – 10.2.1.65:8091



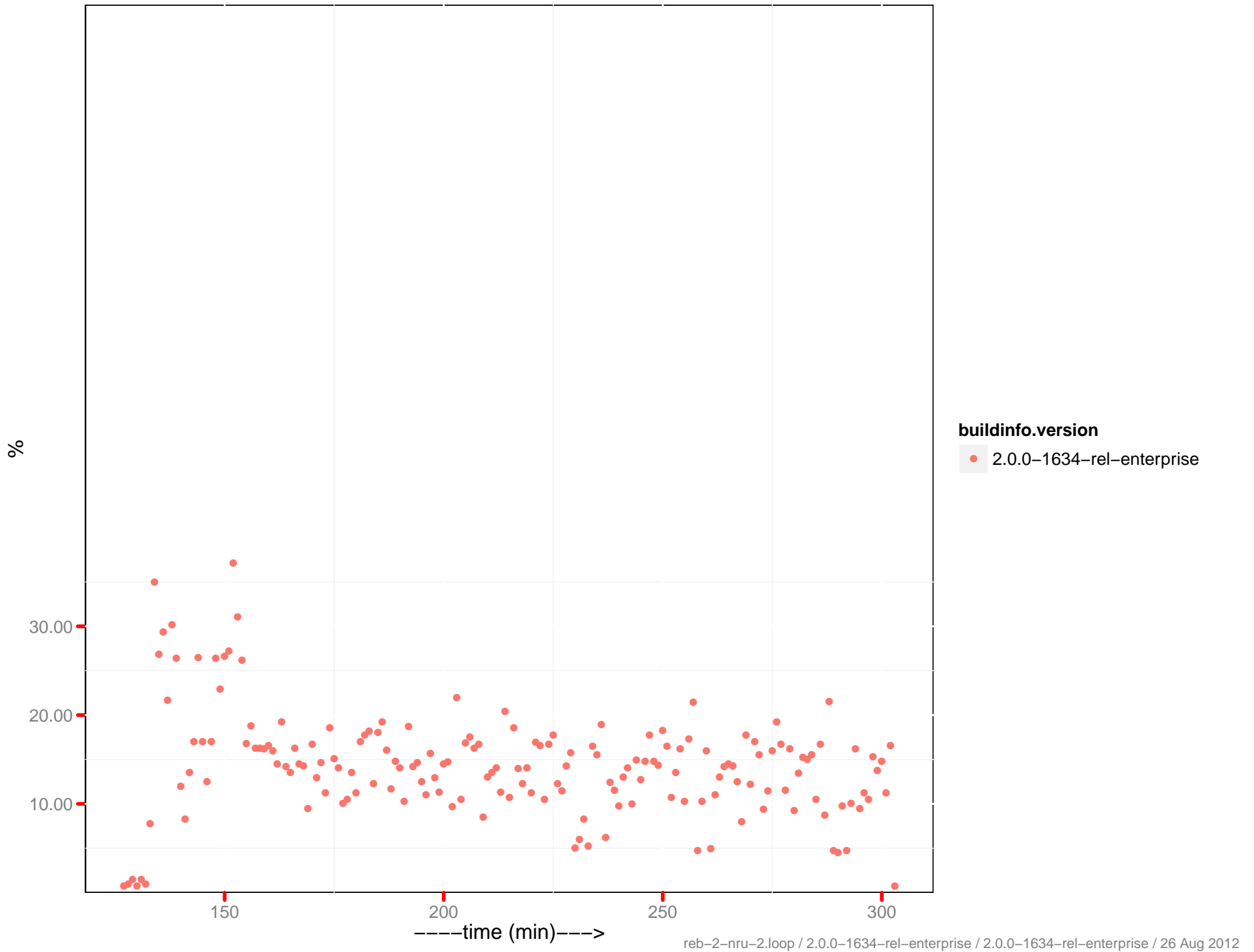
CPU utilization – 10.2.1.66:8091



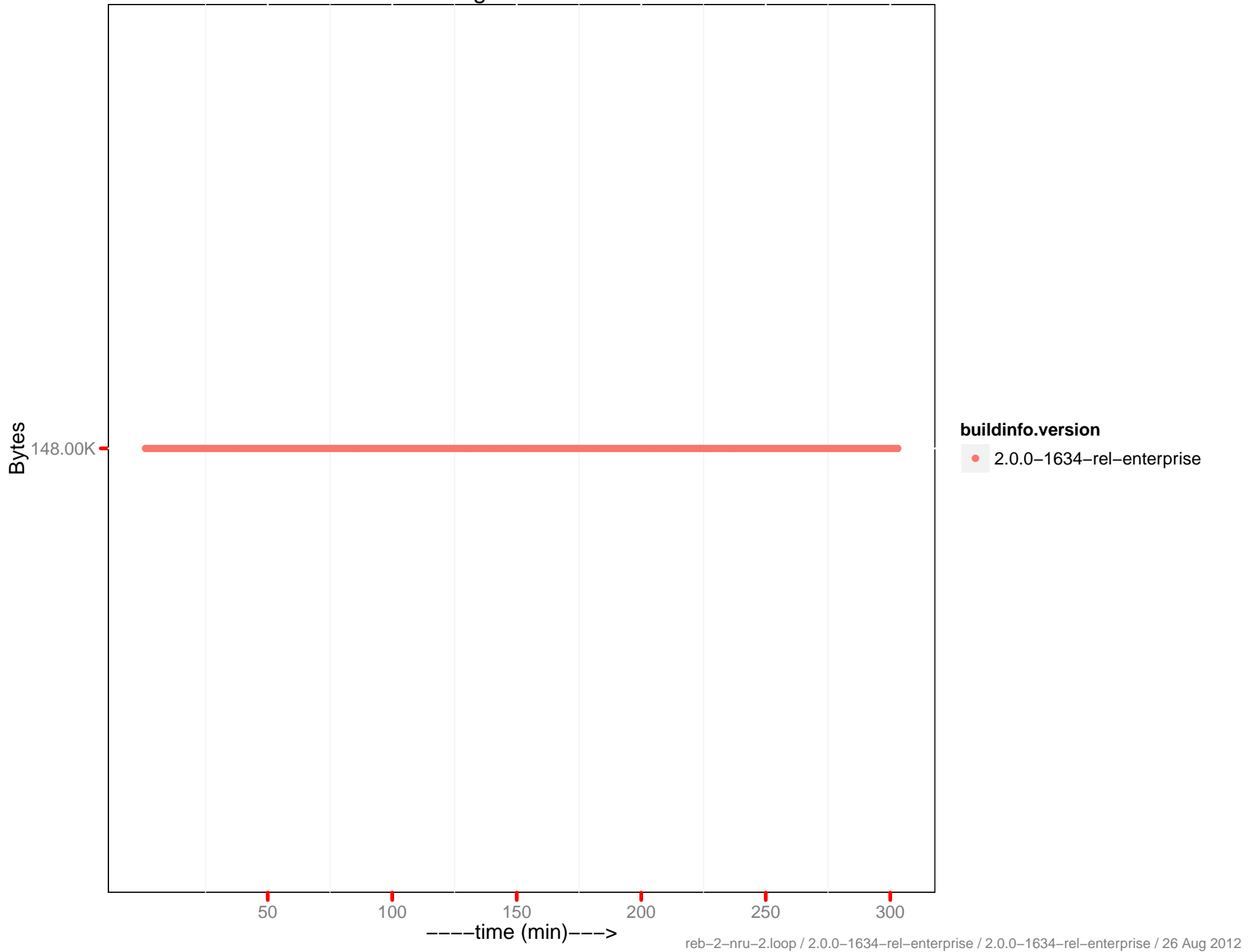
CPU utilization – 10.2.1.67:8091



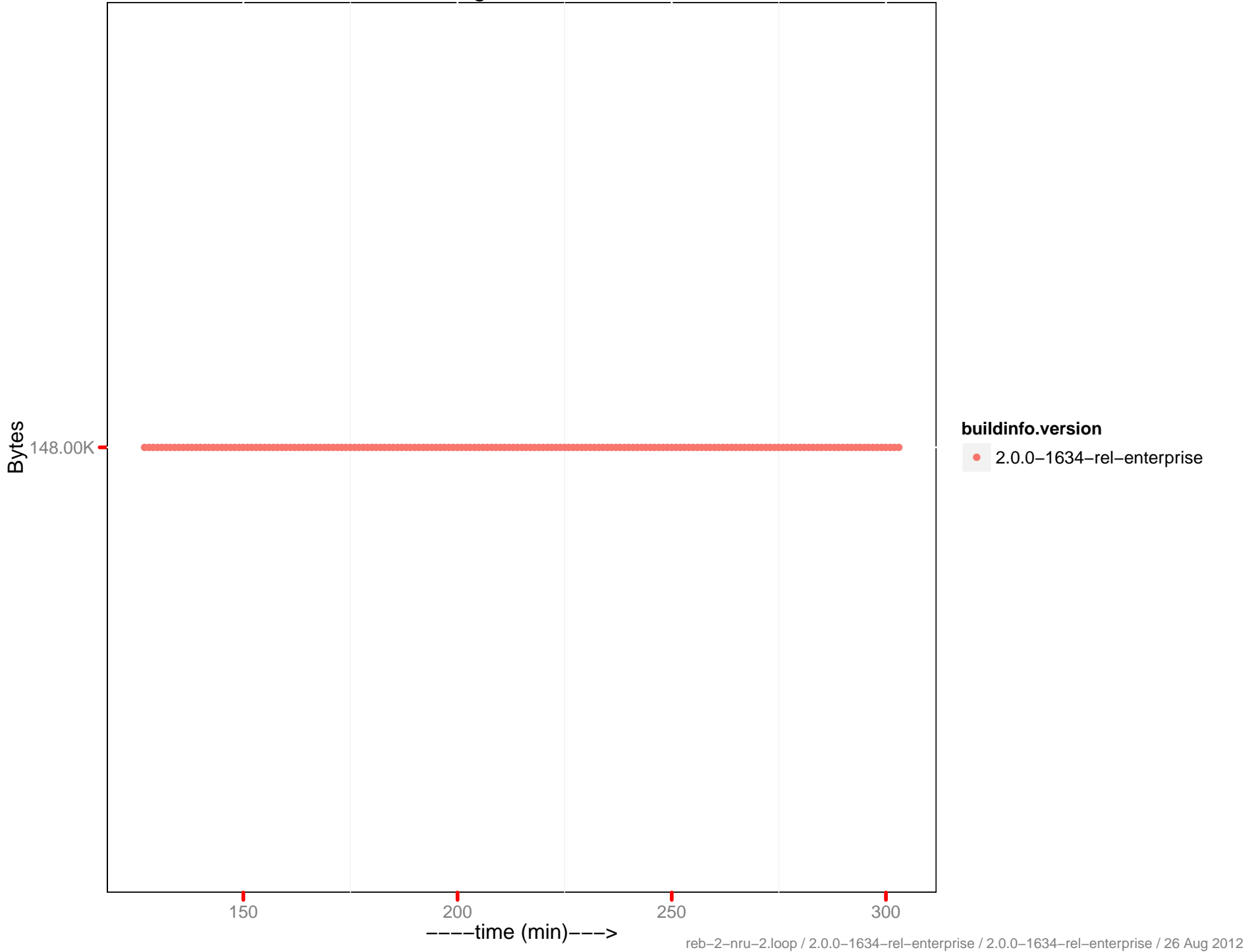
CPU utilization – 10.2.1.68:8091



SWAP Usage - 10.2.1.65:8091

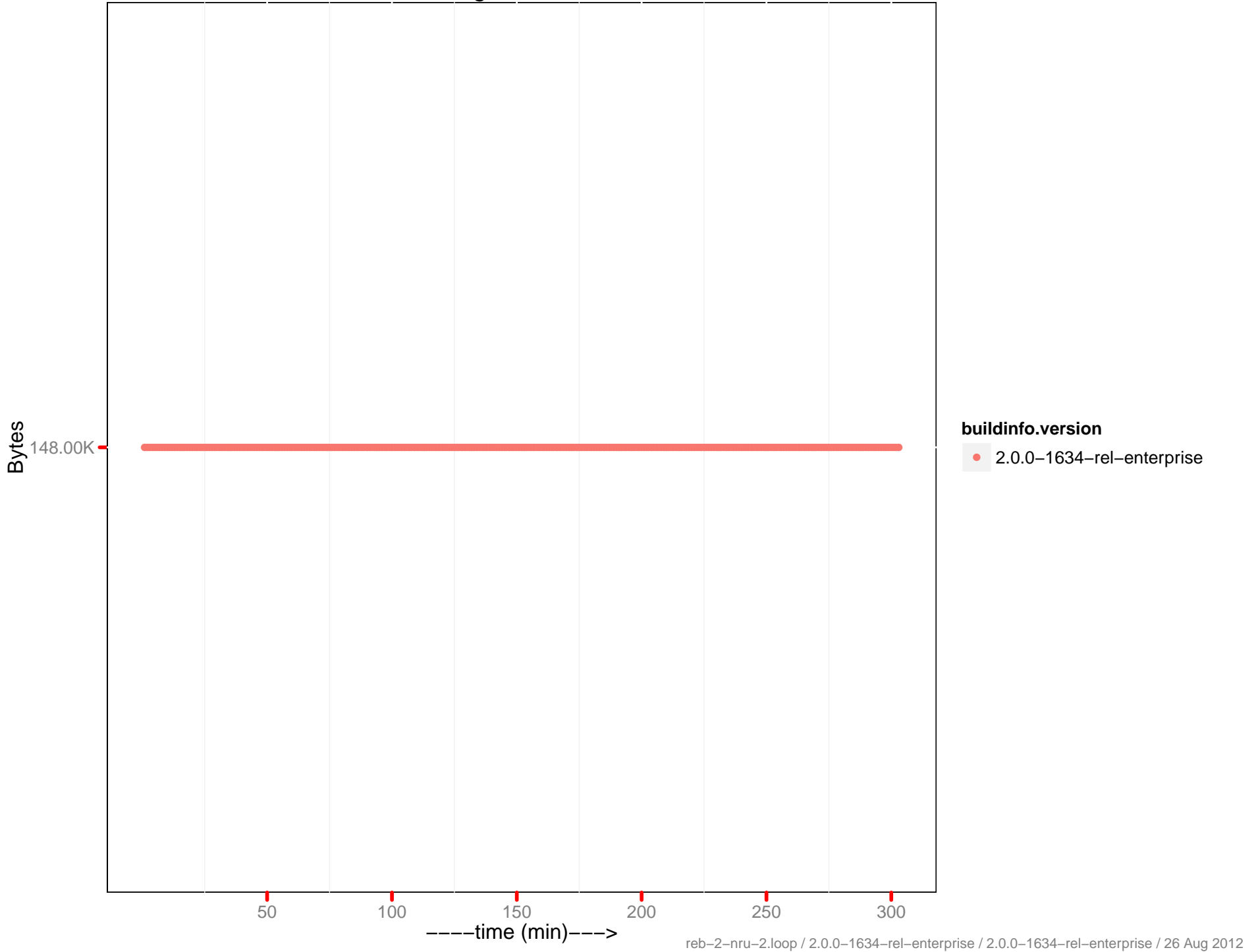


SWAP Usage - 10.2.1.66:8091

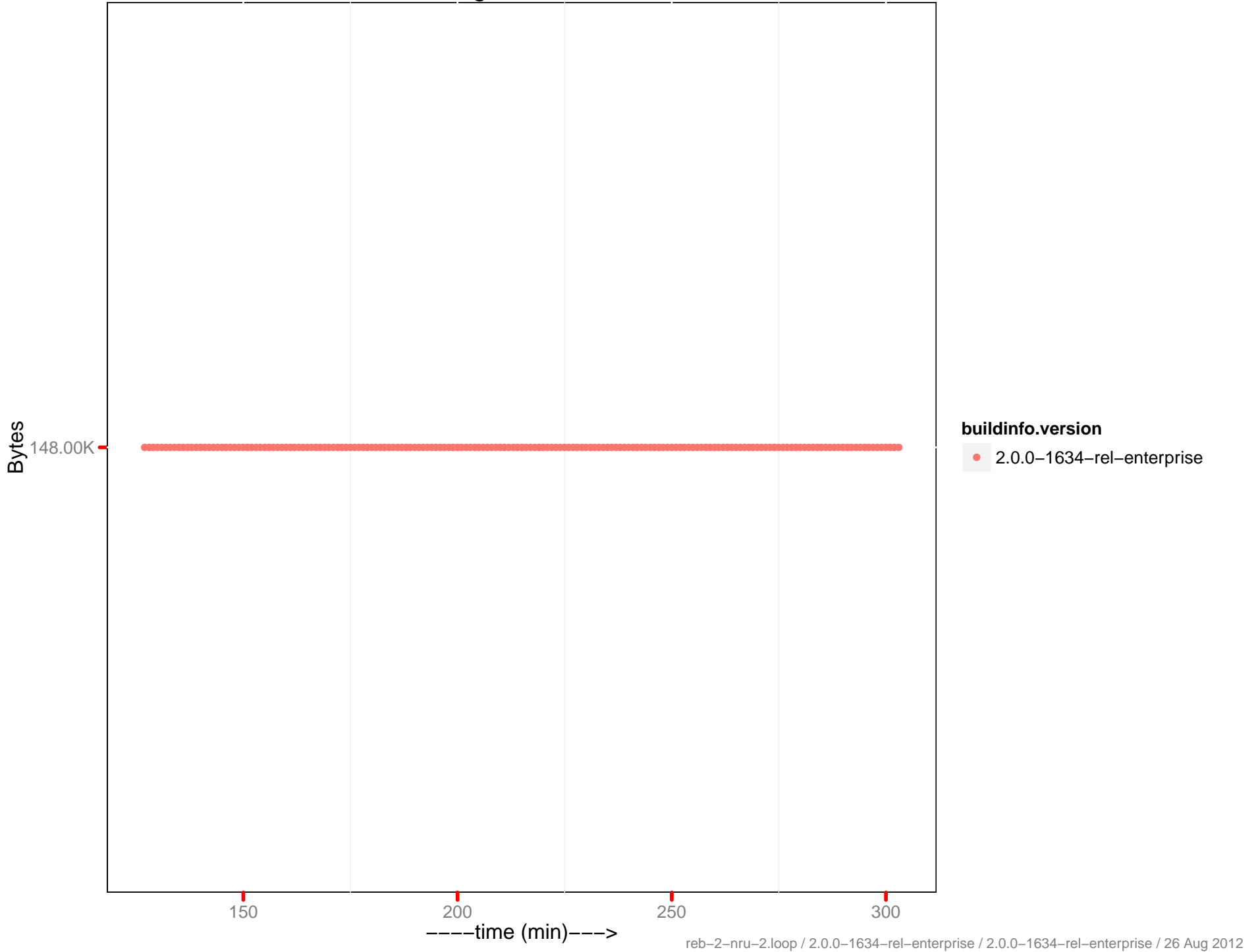


buildinfo.version
• 2.0.0-1634-rel-enterprise

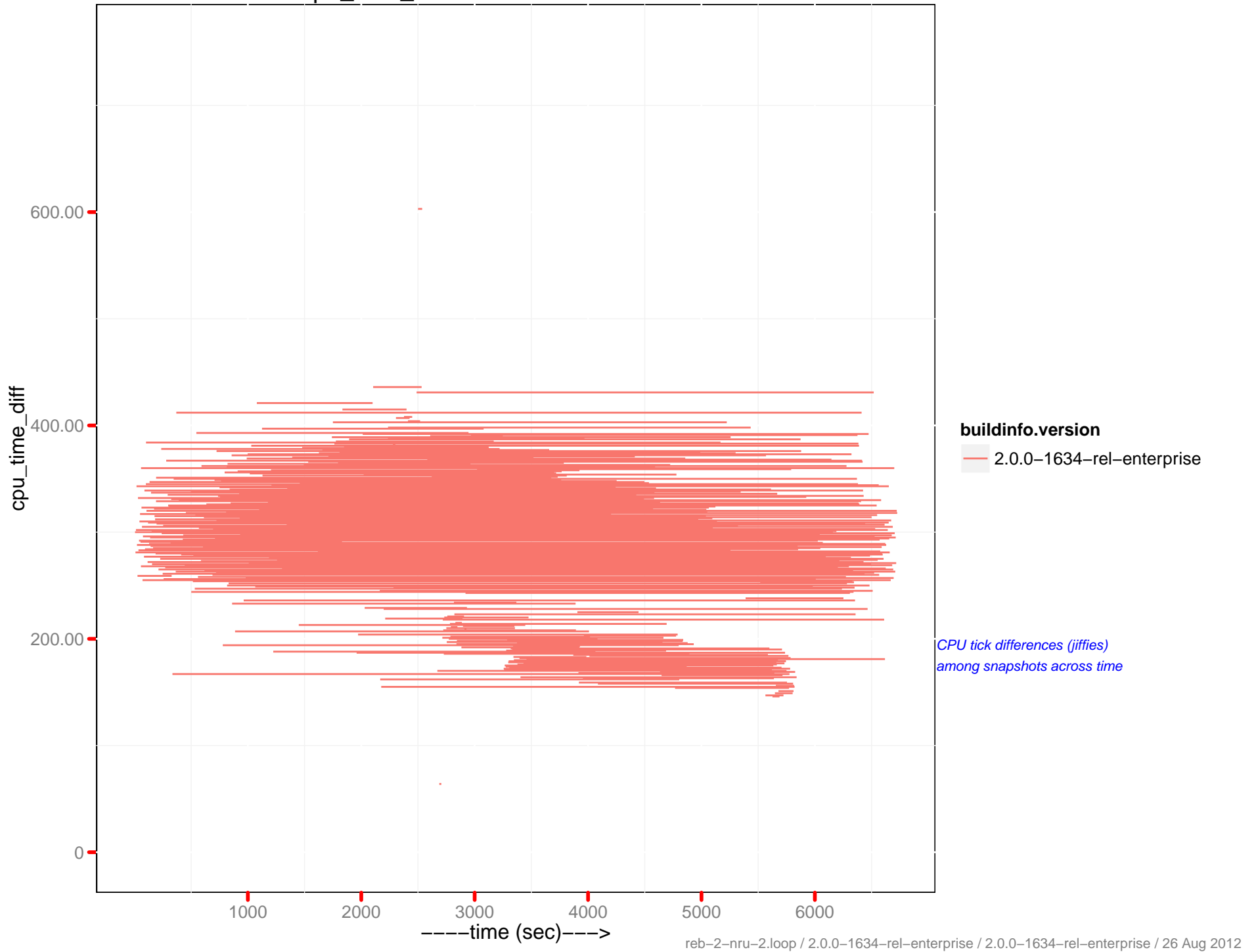
SWAP Usage - 10.2.1.67:8091



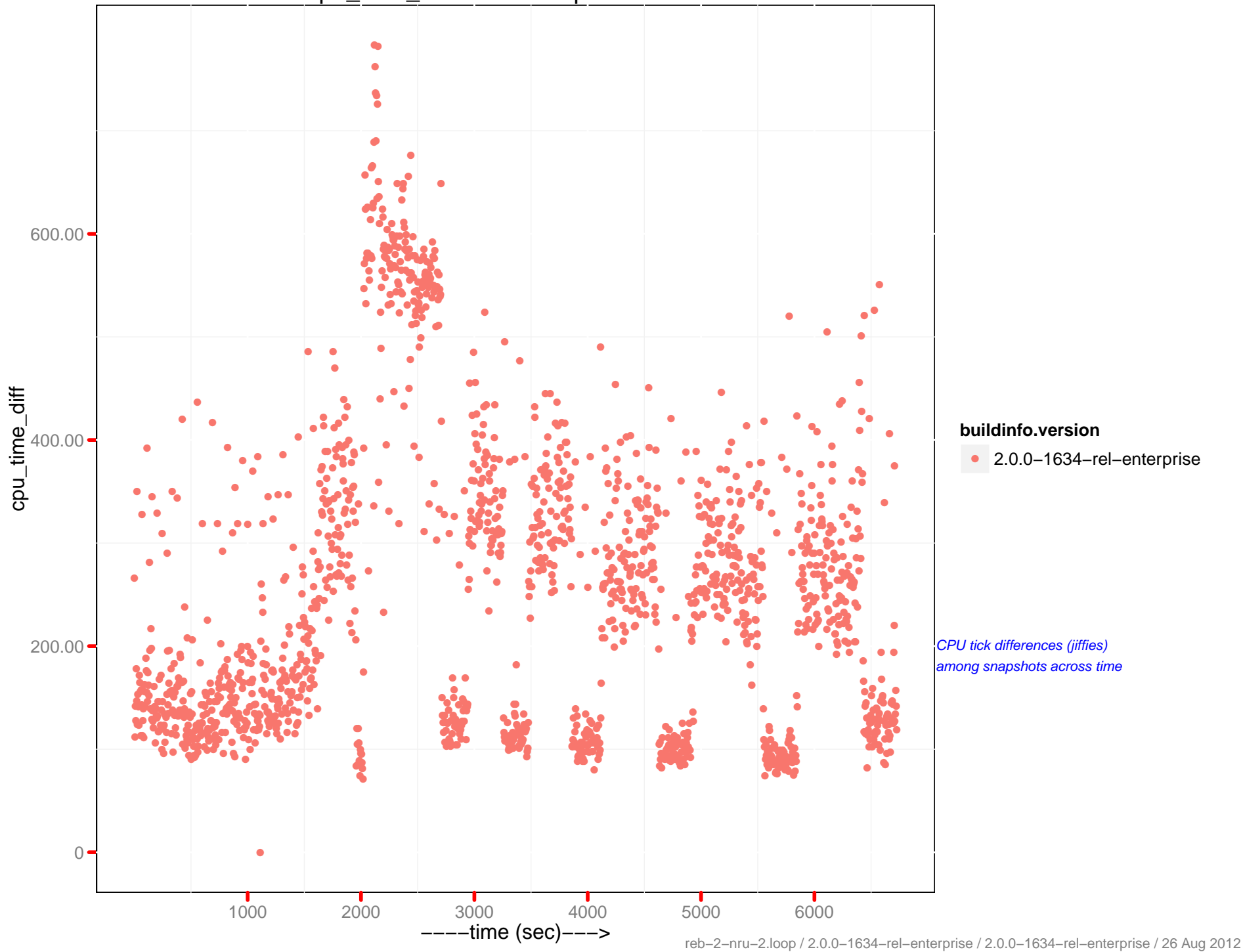
SWAP Usage - 10.2.1.68:8091



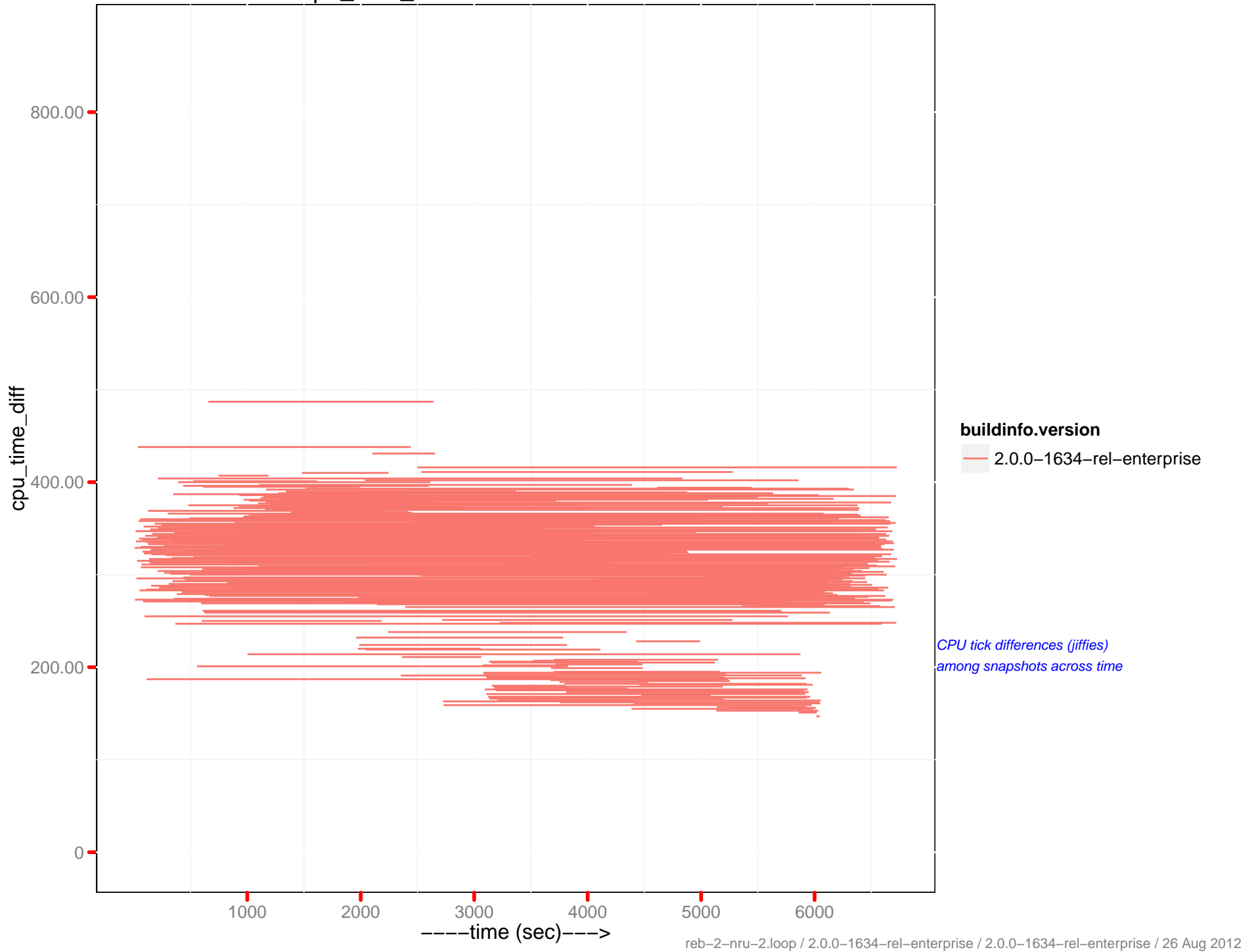
cpu_time_diff: memcached - 10.2.1.65



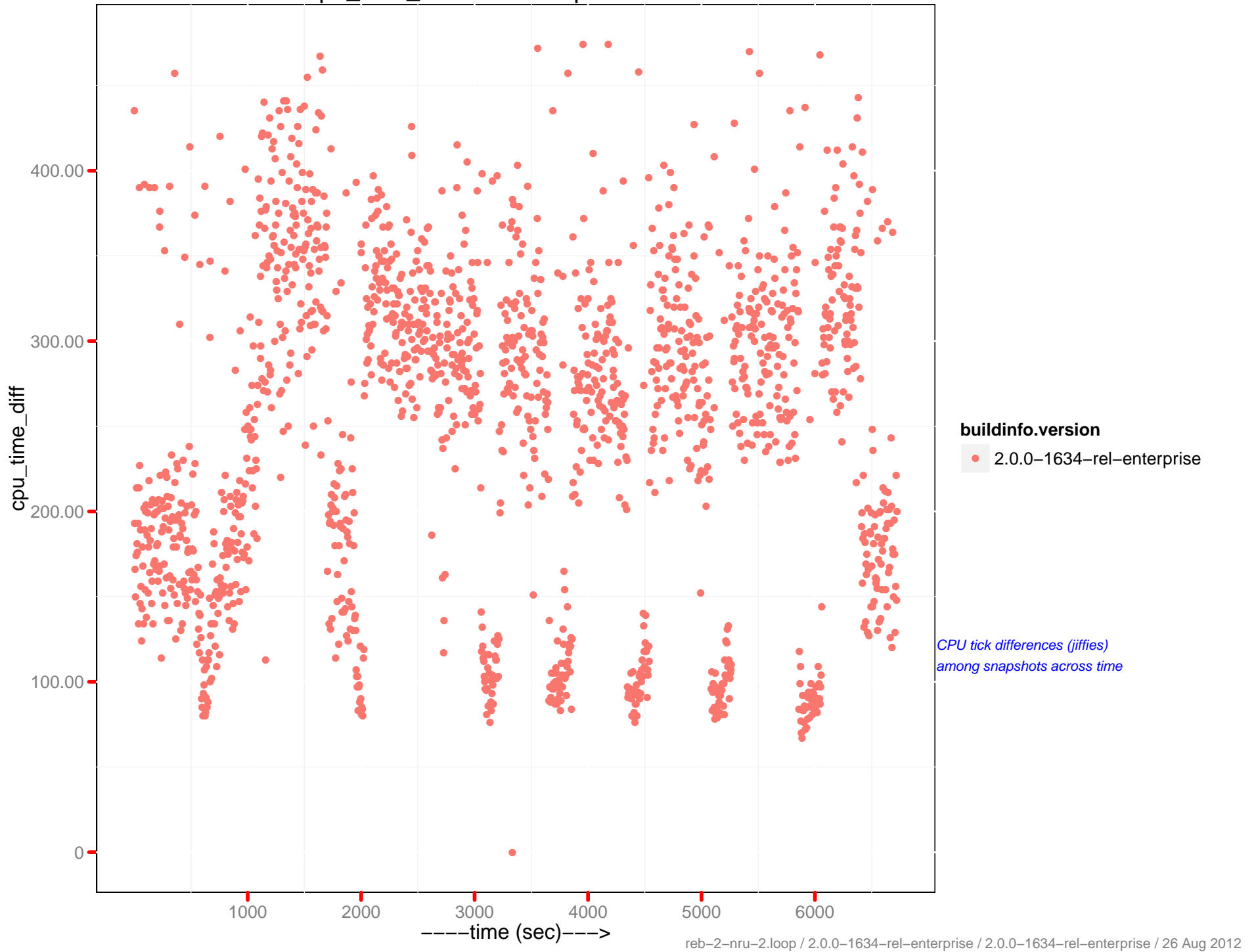
cpu_time_diff : beam.smp - 10.2.1.65



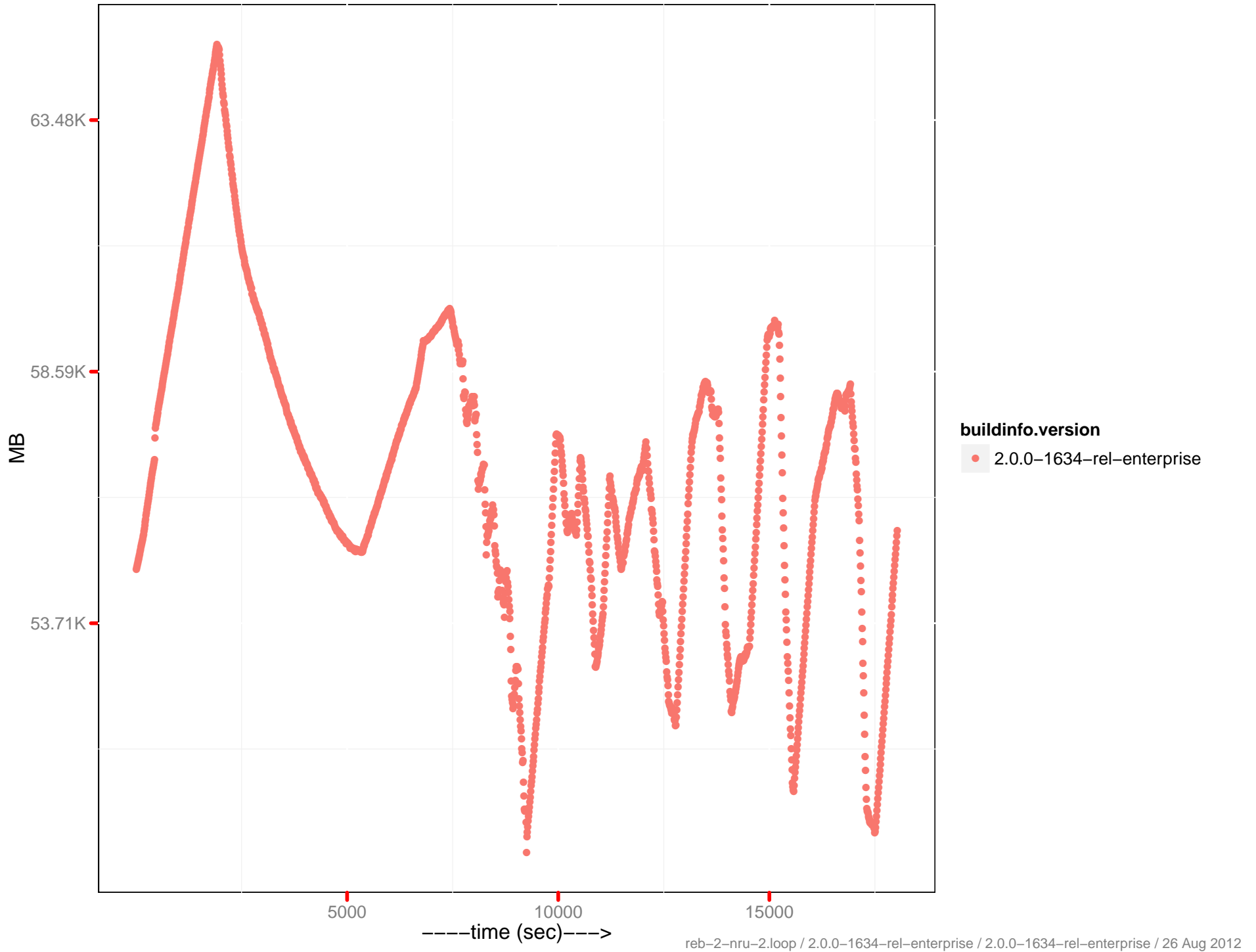
cpu_time_diff: memcached - 10.2.1.67



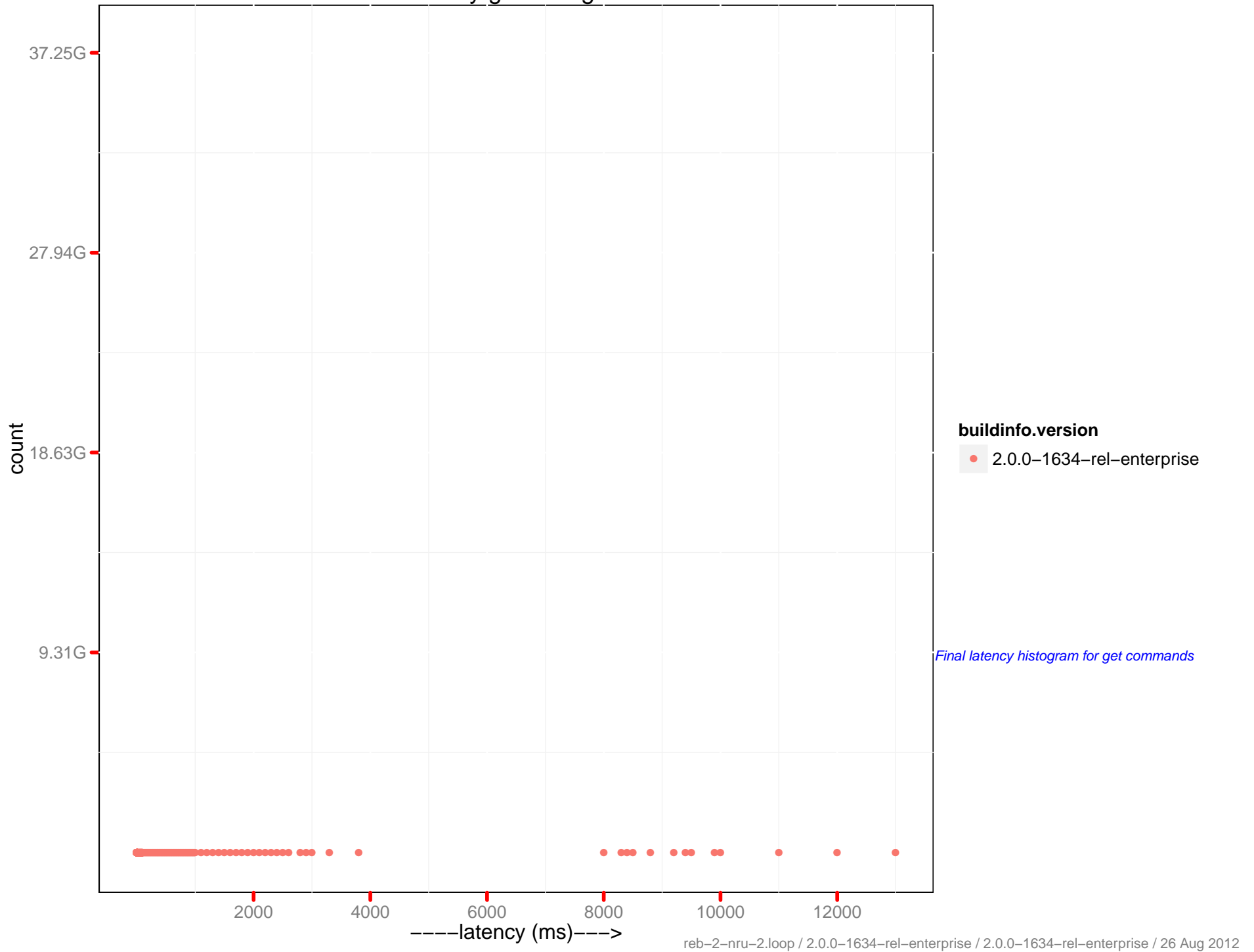
cpu_time_diff : beam.smp - 10.2.1.67



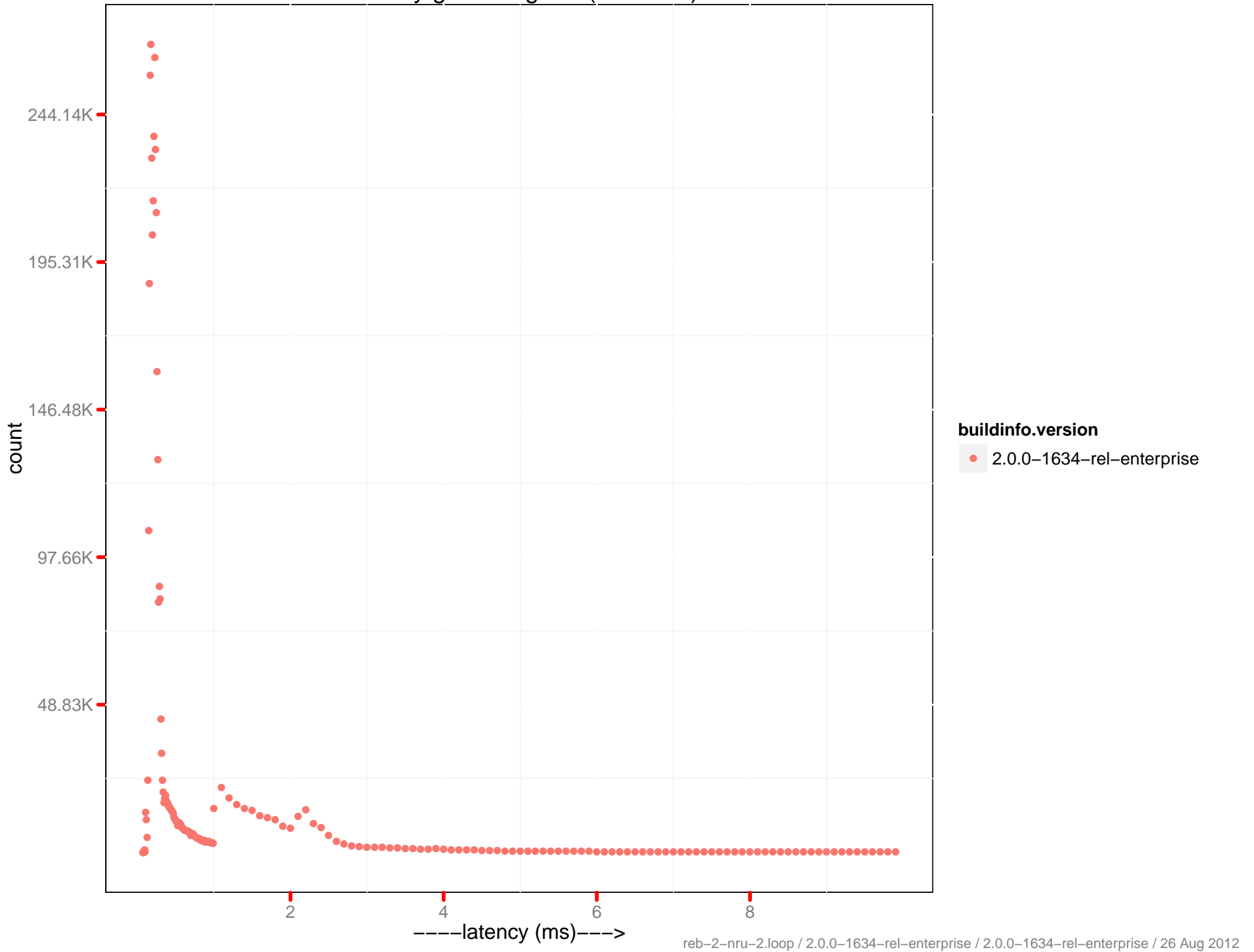
Data disk size



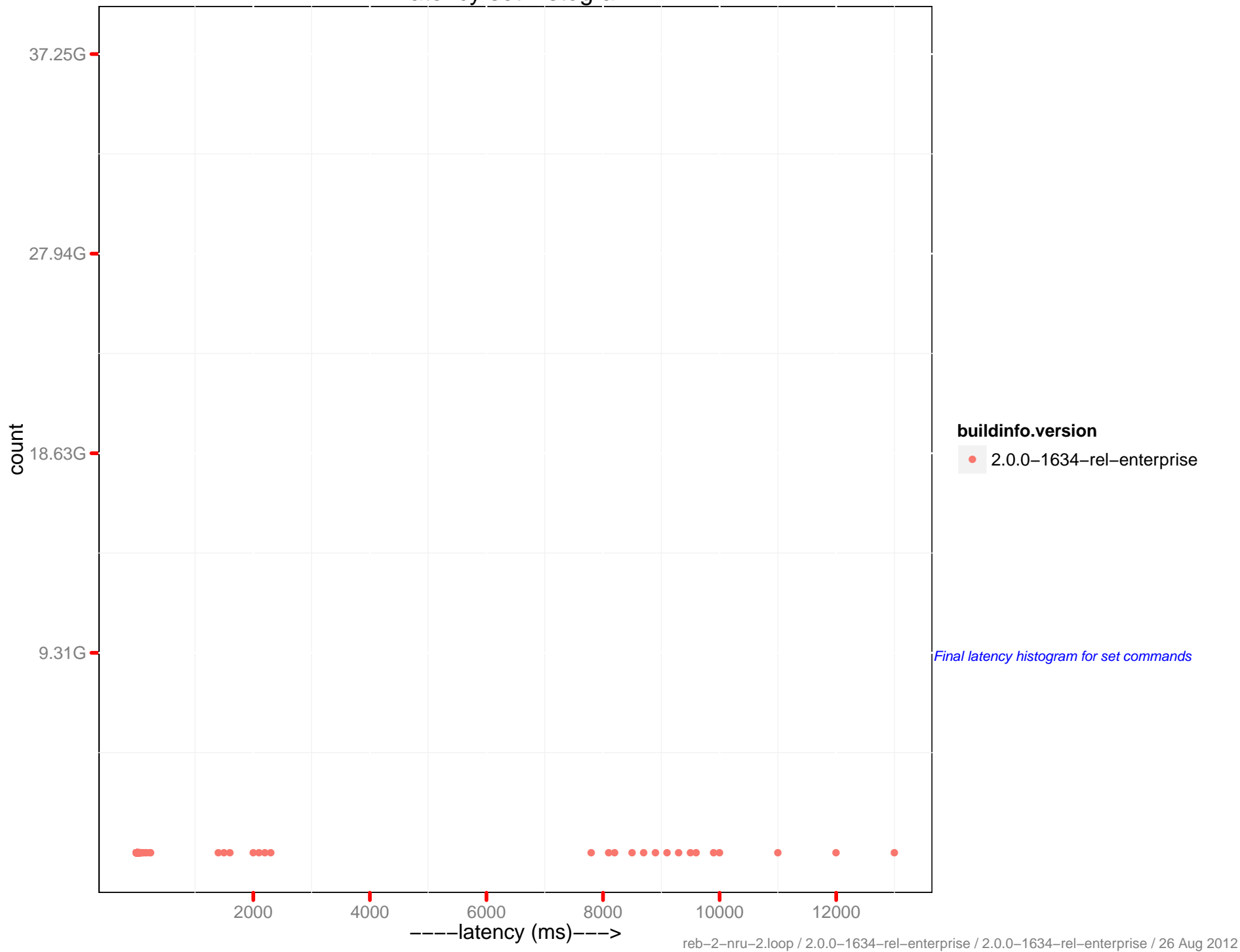
Latency get histogram



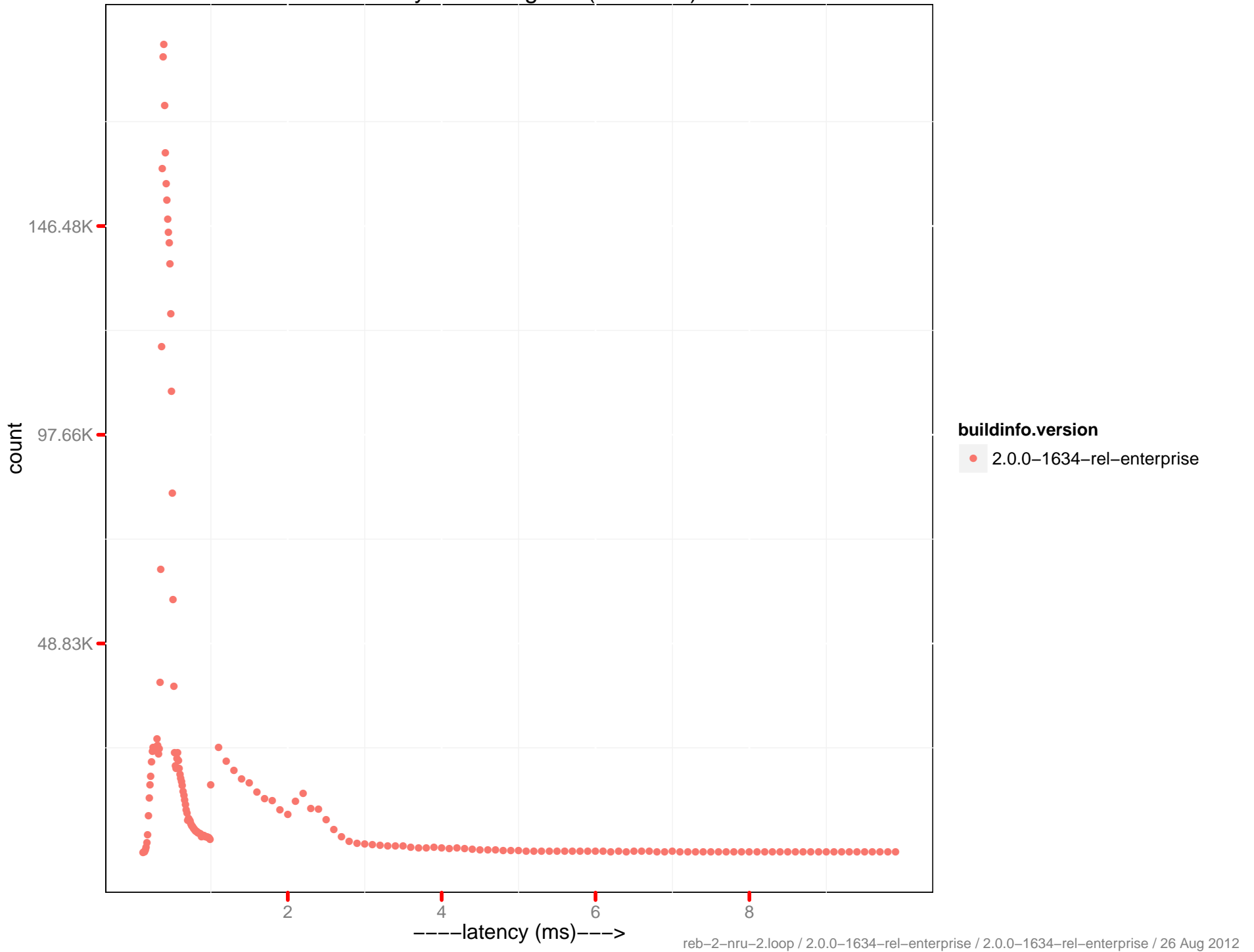
Latency get histogram (0–10 ms)



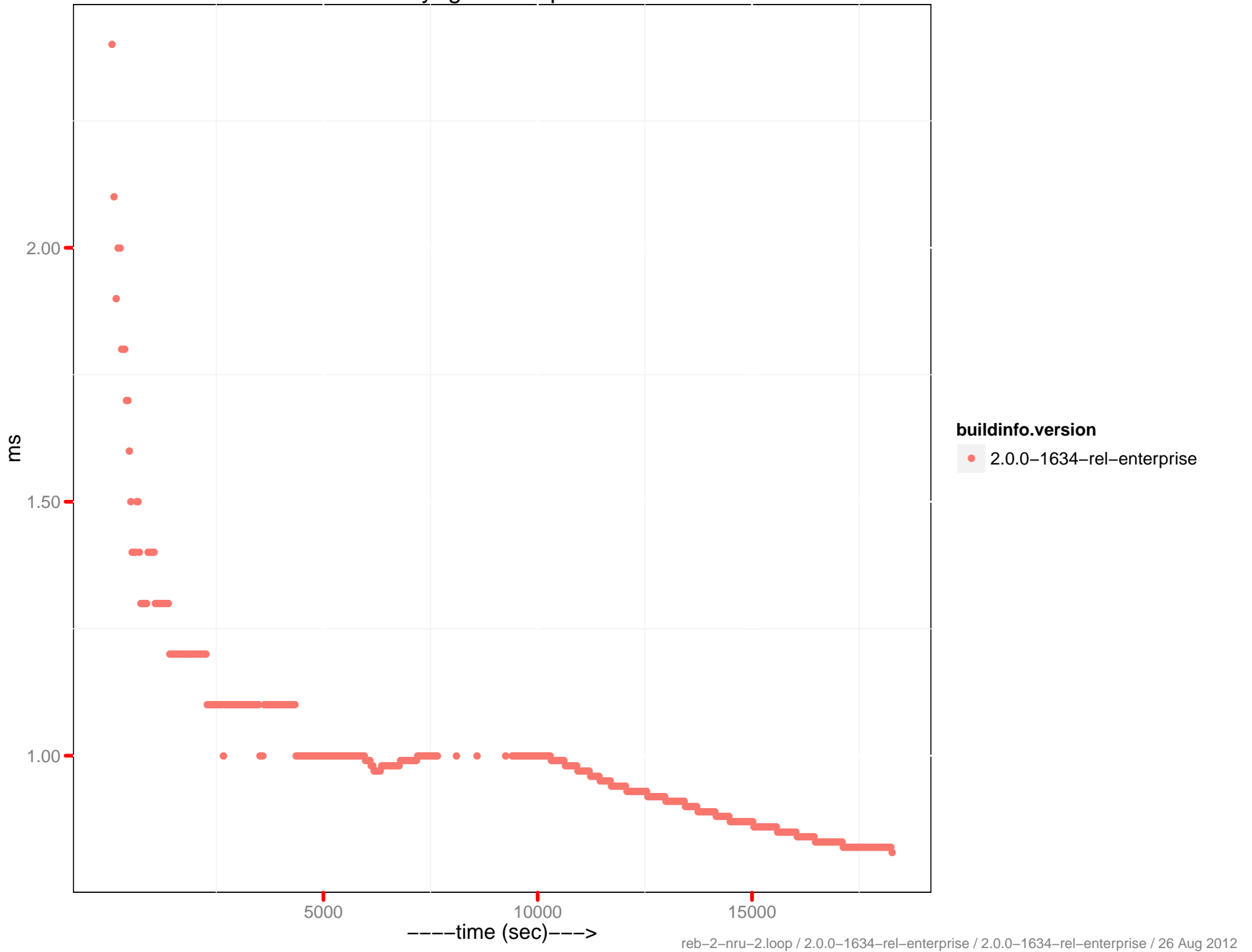
Latency set histogram



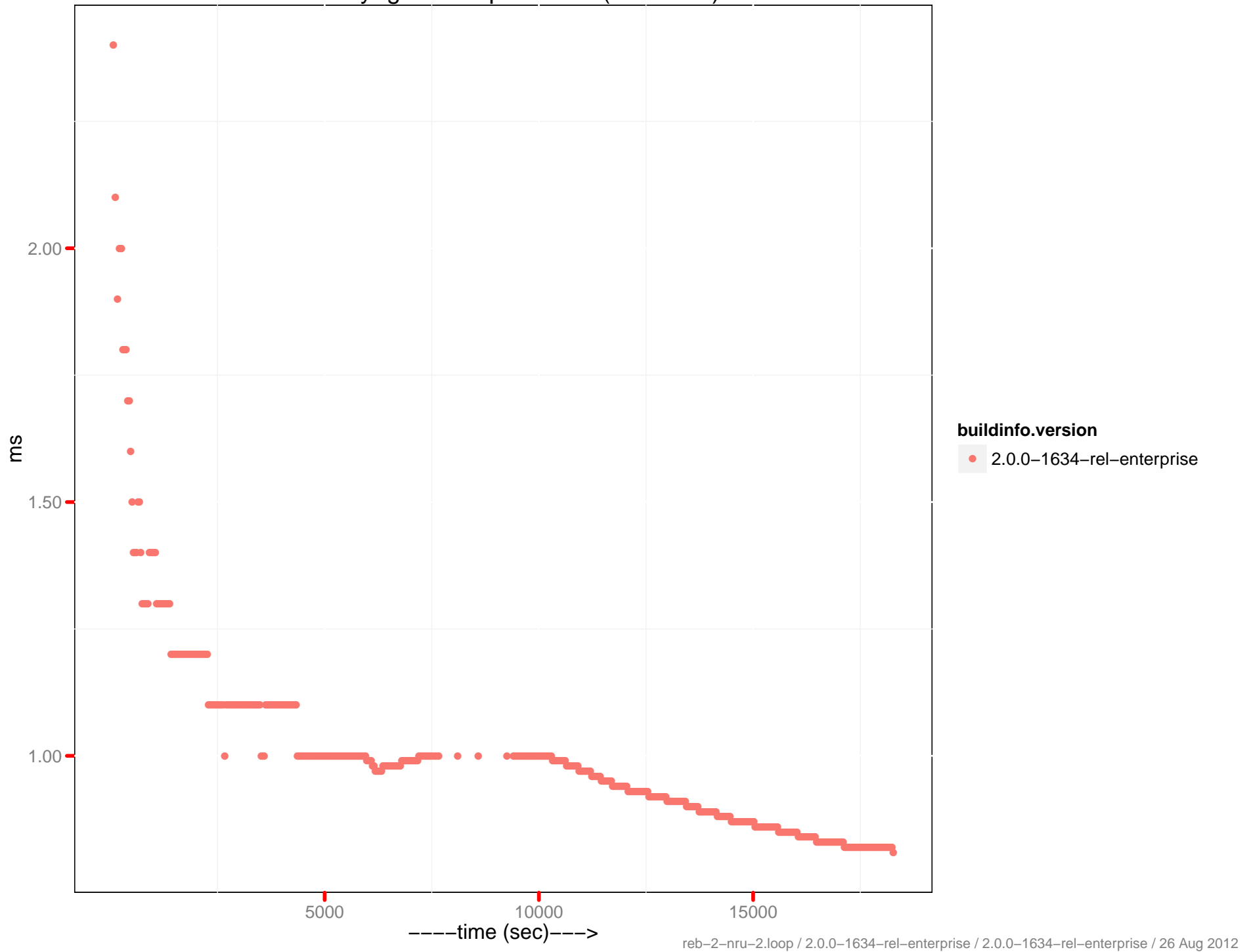
Latency set histogram (0–10 ms)



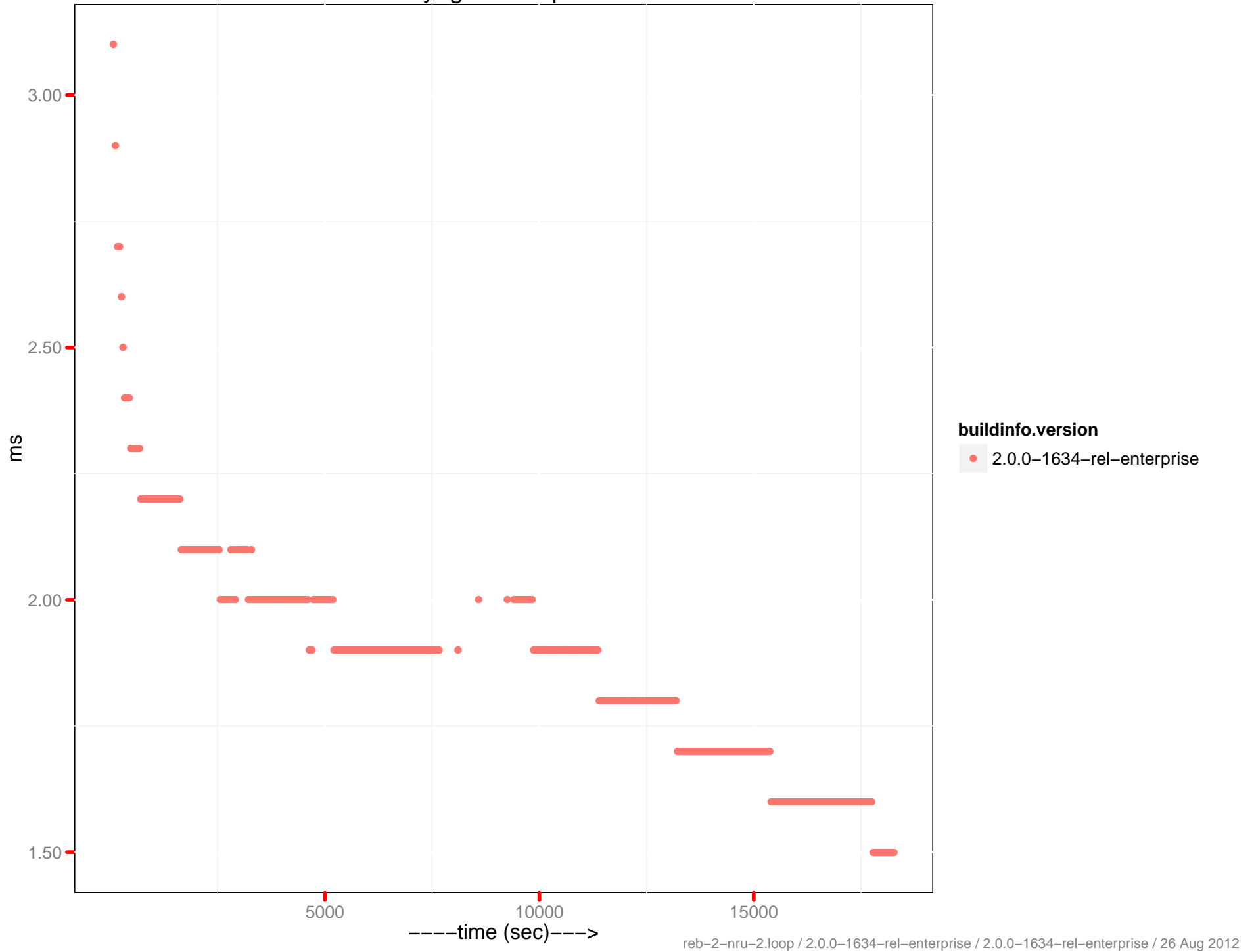
Latency-get 90th percentile



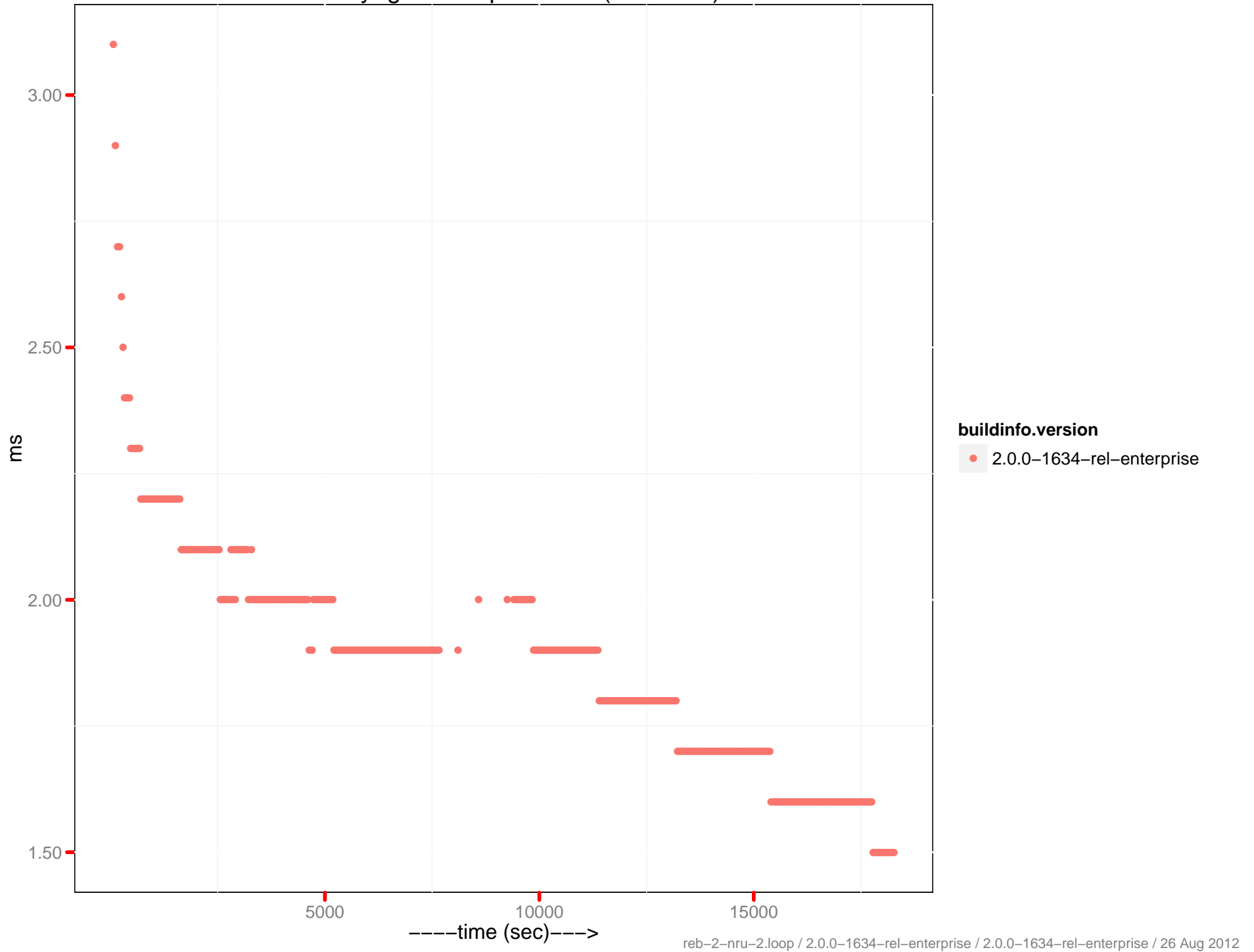
Latency-get 90th percentile (0 - 10ms)



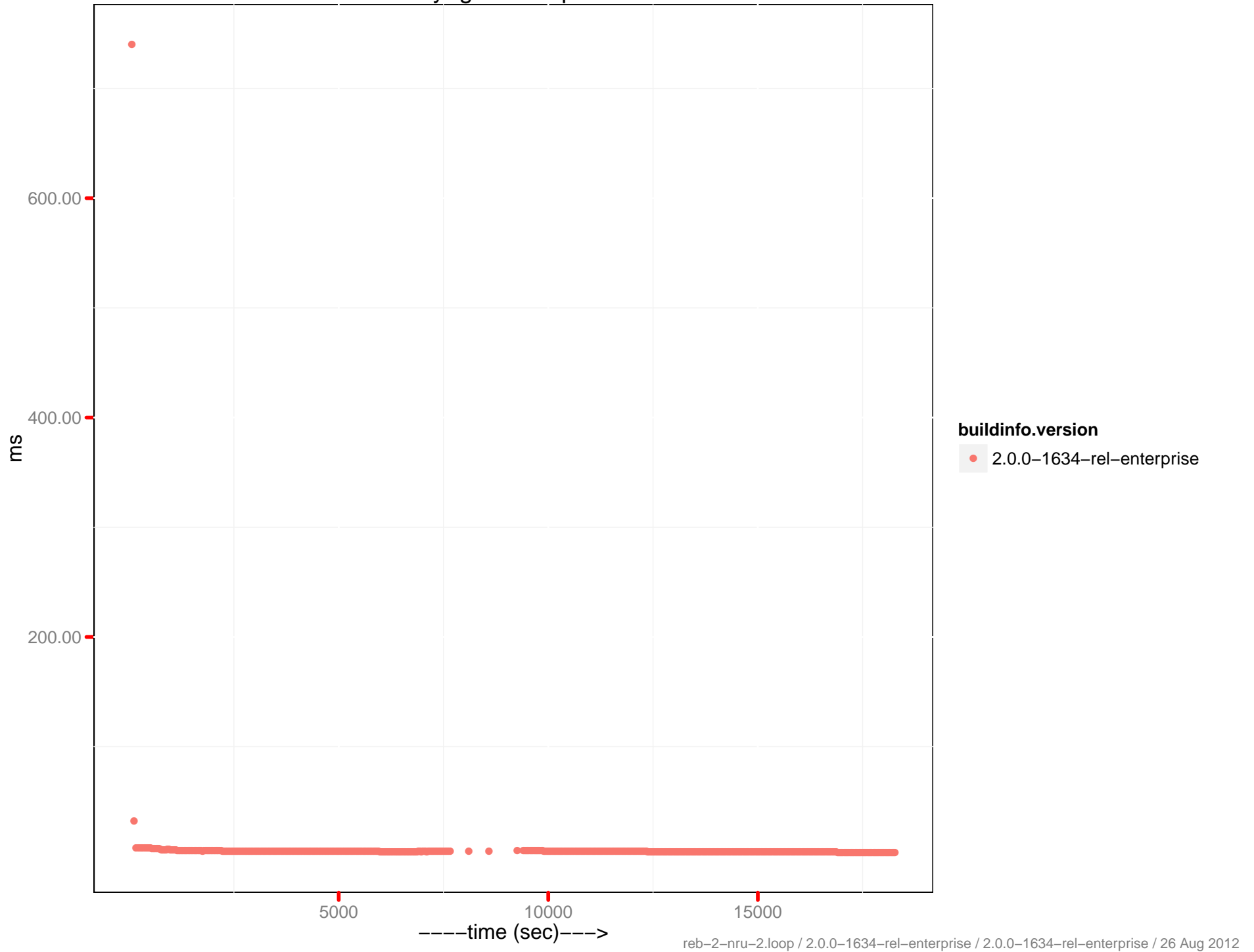
Latency-get 95th percentile



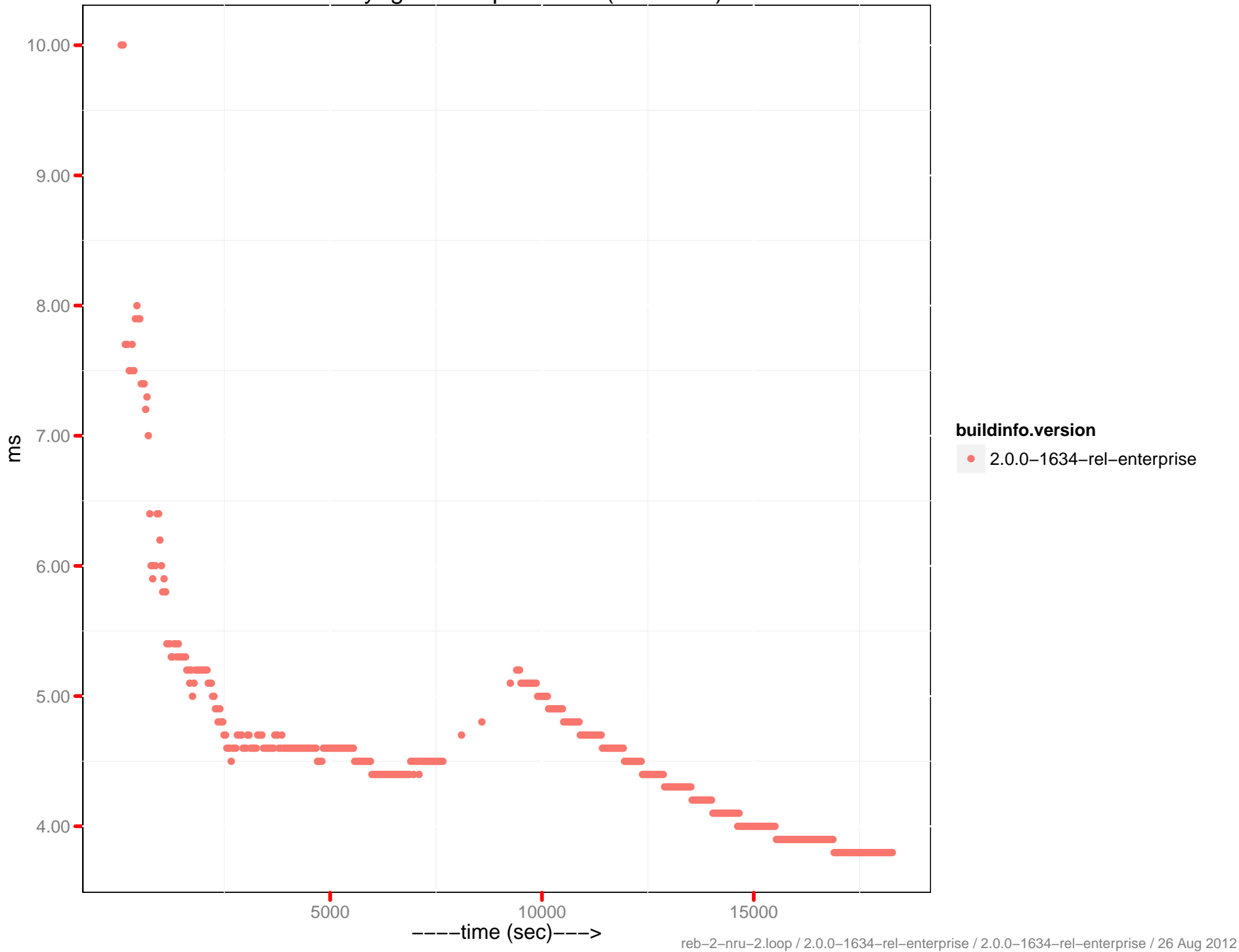
Latency-get 95th percentile (0 - 10ms)



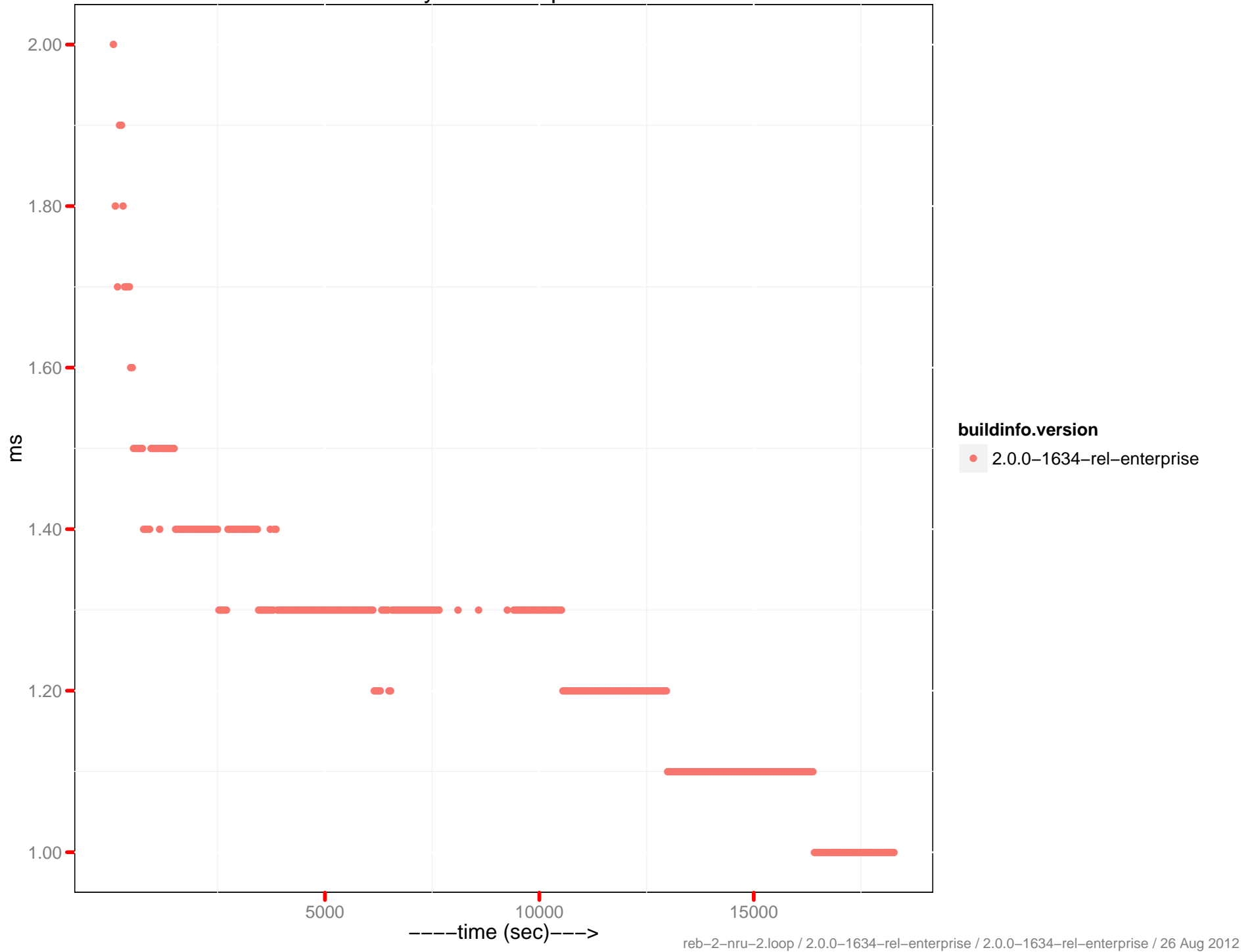
Latency-get 99th percentile



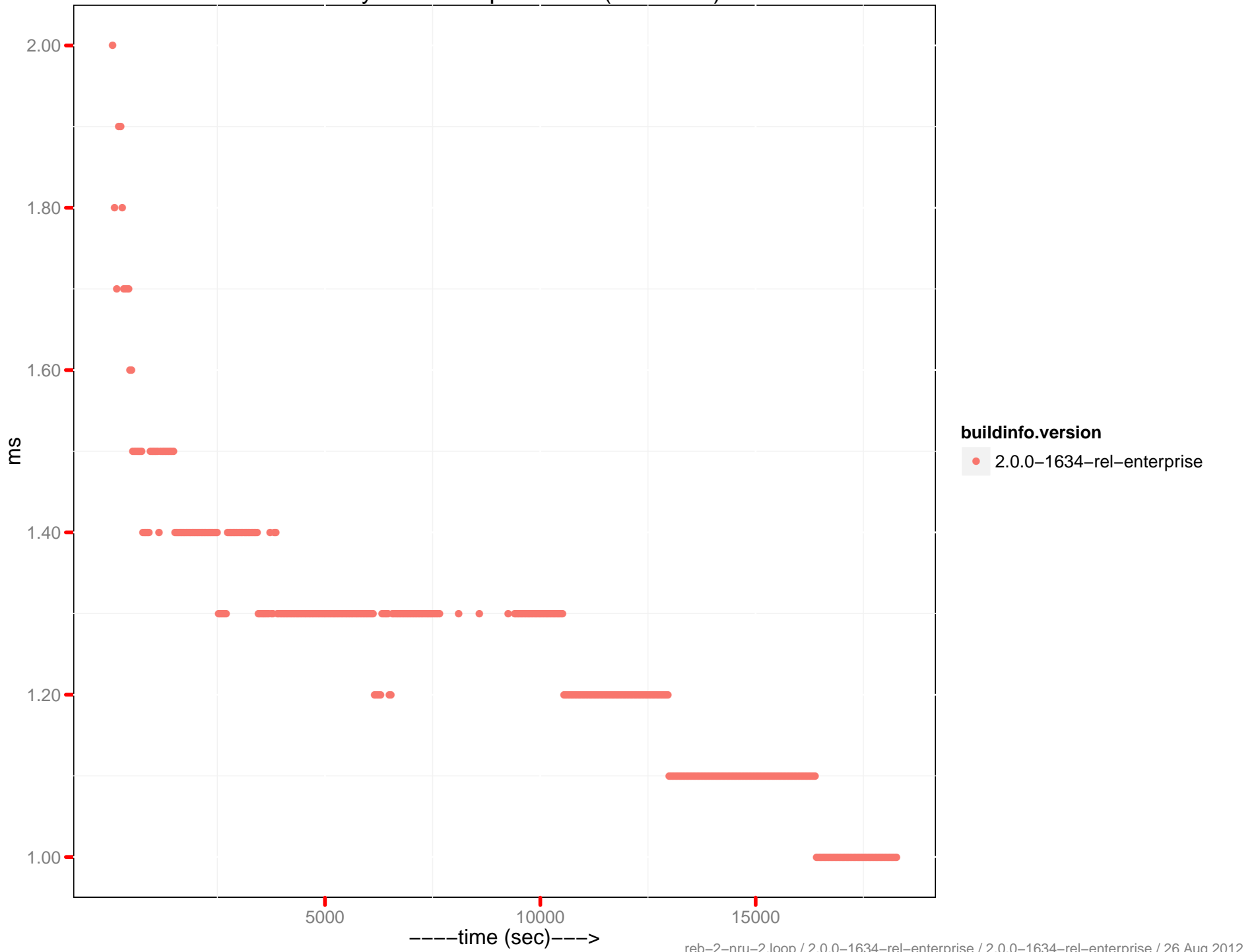
Latency-get 99th percentile (0 - 10ms)



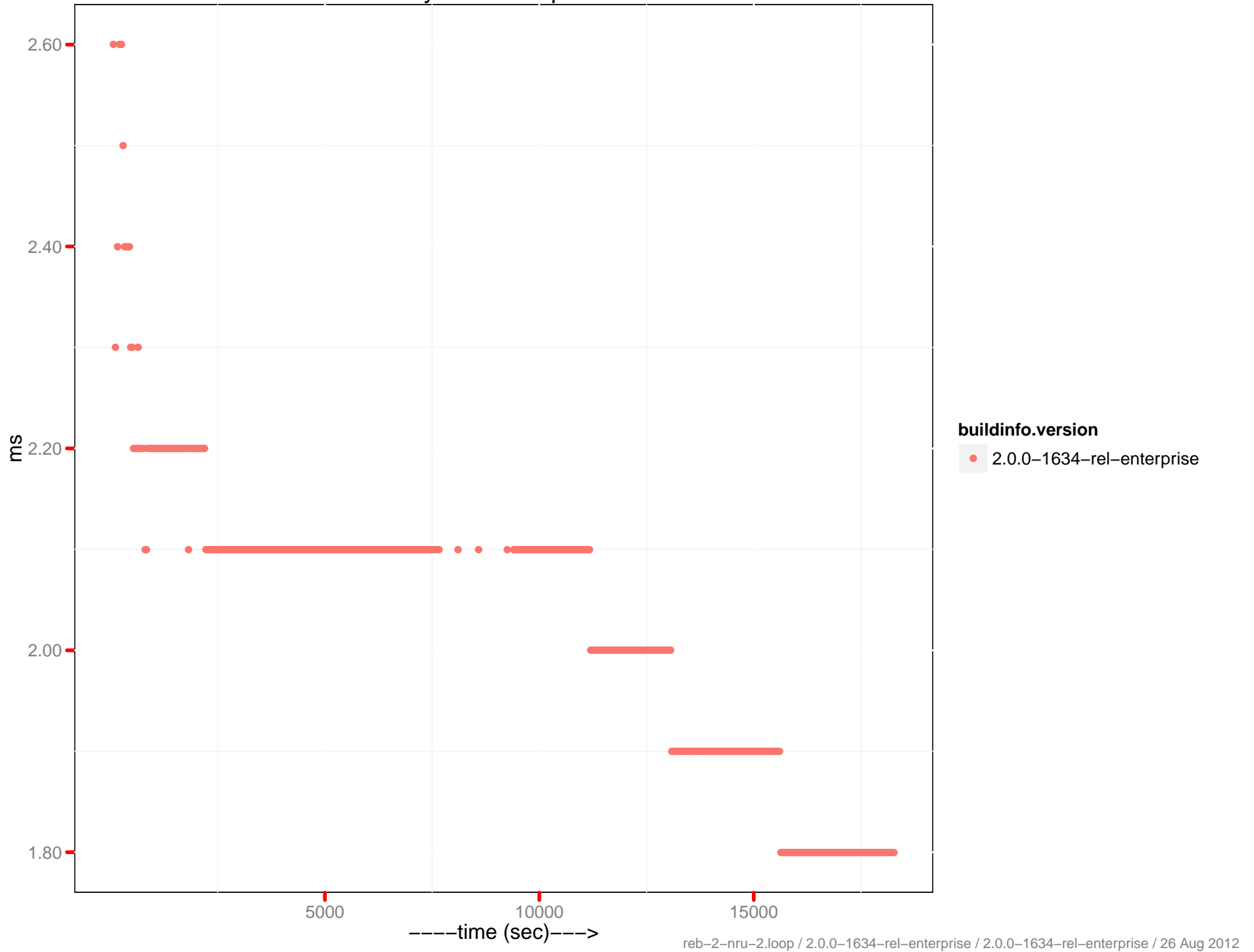
Latency-set 90th percentile



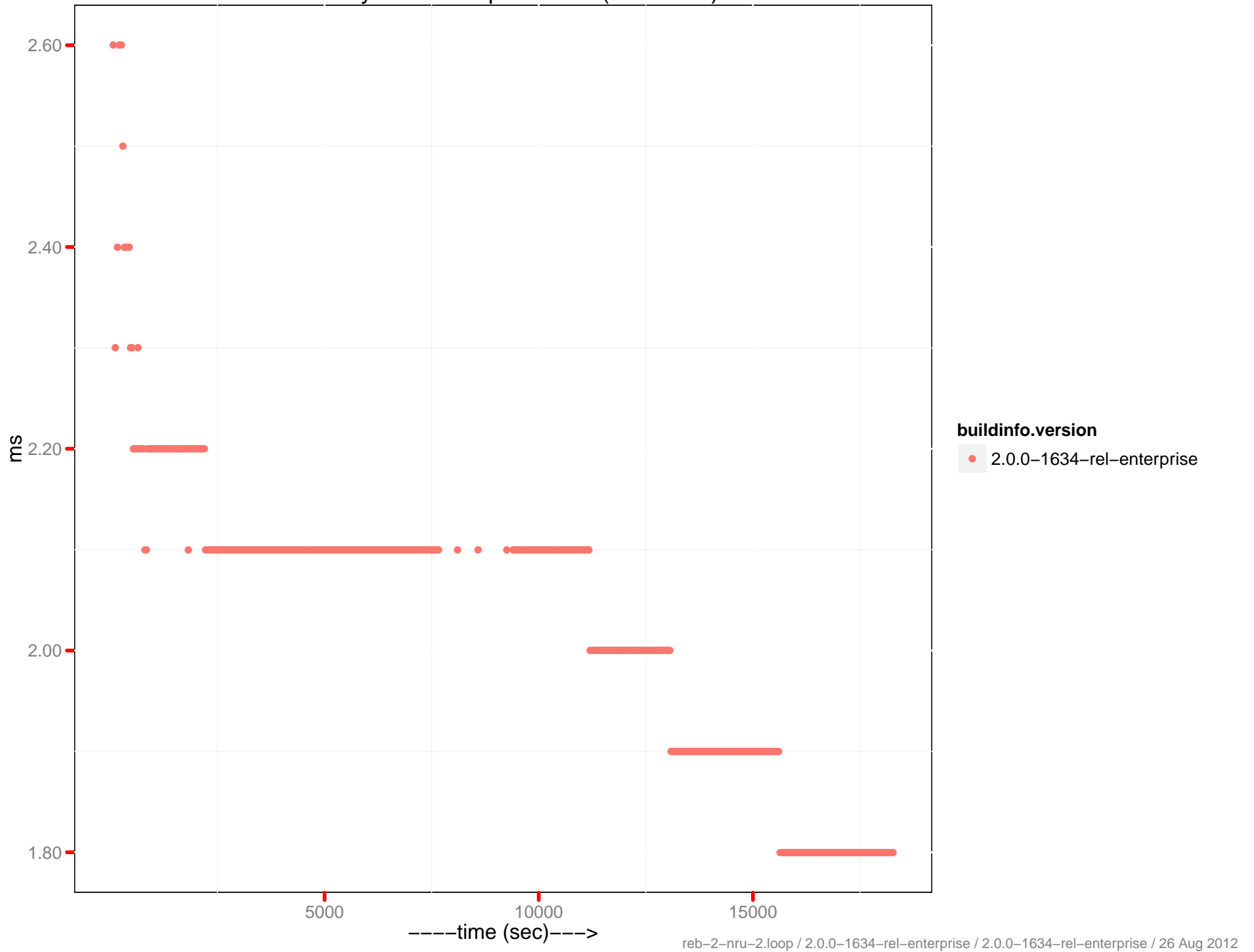
Latency-set 90th percentile (0 – 10ms)



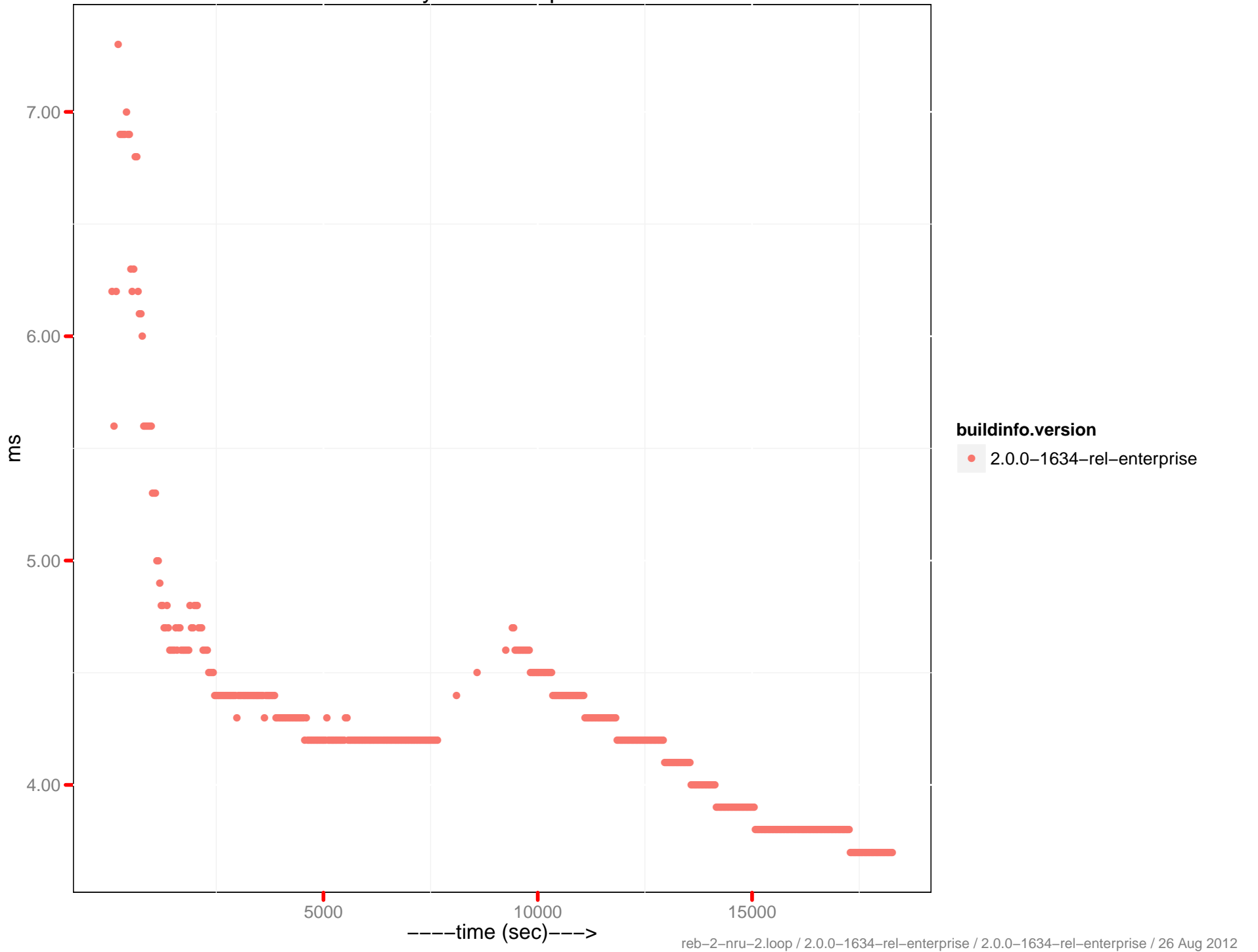
Latency-set 95th percentile



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

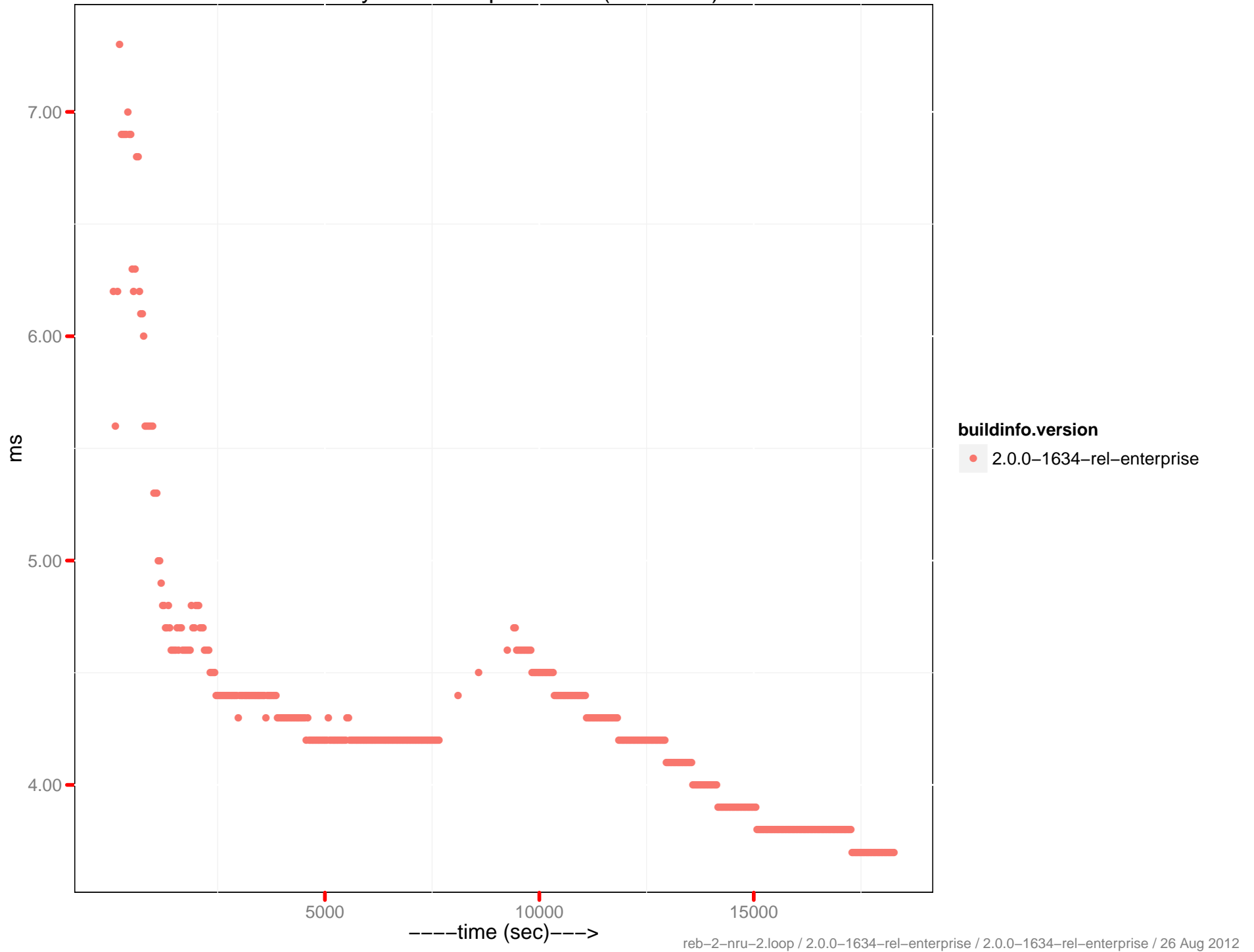


Latency-set 99th percentile

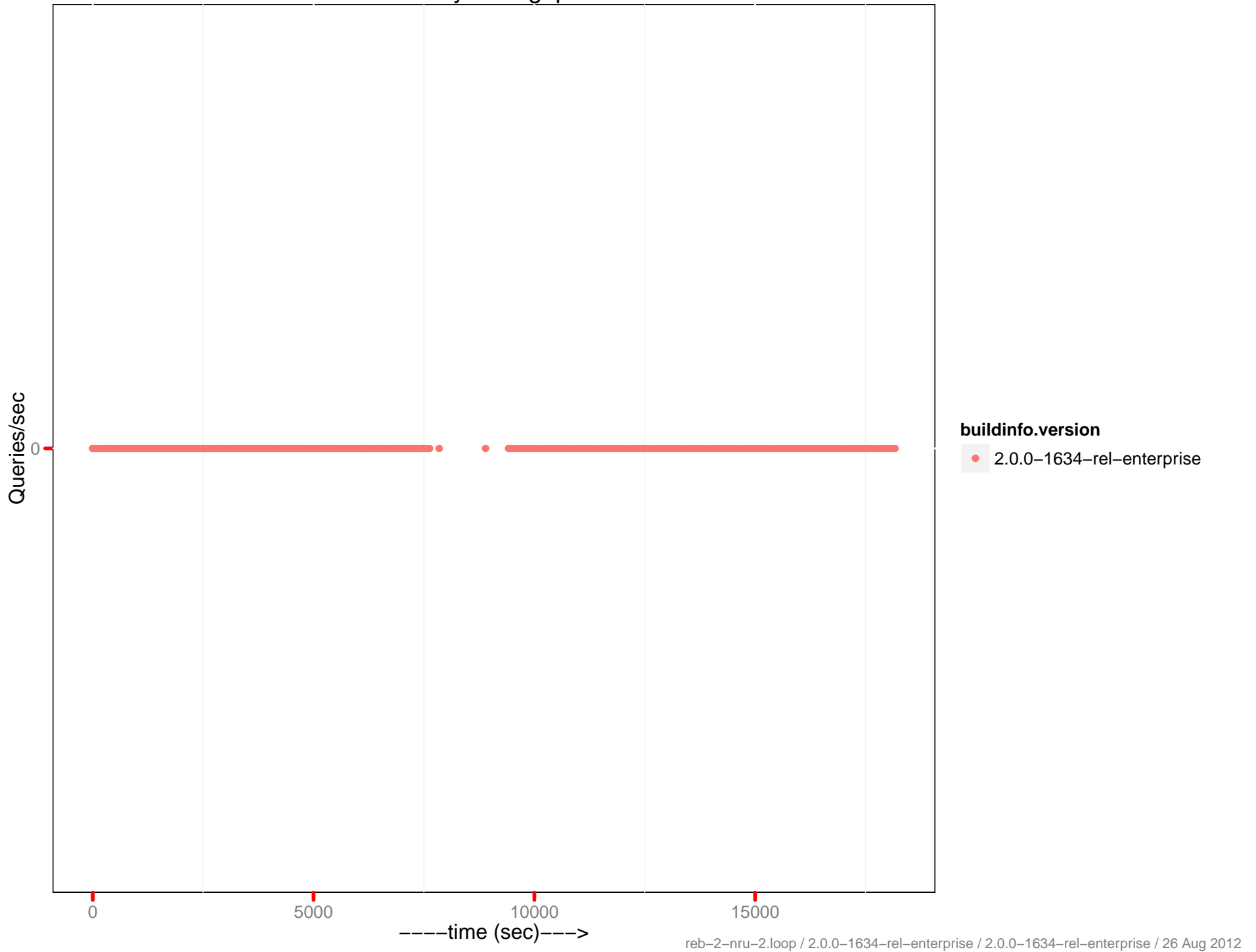


buildinfo.version
2.0.0-1634-rel-enterprise

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



Query throughput



```
reb-2-nru-2.conf
# rebalance mixed 10M load, 1M hot reload, 6M access creates
#
performance.eperf.EPerfClient.test_eperf_rebalance

params:

# general
batch=50
kind=nonjson
mem_quota=20000
db_compaction=30

# load phase
hot_init_items=1000000
items=10000000

# access phase
# Read:Insert:Update:Delete Ratio = 50:4:40:6.
ratio_sets=0.5
ratio_misses=0.05
ratio_creates=0.08
ratio_deletes=0.13
ratio_hot=0.05
ratio_hot_gets=0.99
ratio_hot_sets=0.99
ratio_expirations=0.03
max_creates=6000000

# rebalance
nru_task=1
nru_reb_delay=7200
reb_max_retries=5
num_nodes_after=3

# control (defaults: pytests/performance/perf_defaults.py)
load_wait_until_drained=1
load_wait_until_repl=1
loop_wait_until_drained=0
mcsoda_heartbeat=3
mcsoda_max_ops_sec=300
tear_down=1
tear_down_proxy=1
tear_down_bucket=0
tear_down_cluster=1
tear_down_on_setup=0
```

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

[servers]
1:10.2.1.65
2:10.2.1.66
3:10.2.1.67
4:10.2.1.68

[clients]
1:10.2.1.60

[membase]
rest_username:Administrator
rest_password:password

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