Load Test Report

Date: 7/27/2016

Test from : virginia

Query URL: http://2016pressable50.reviewsignal.com/

Started at: Wed Jul 27 2016, 04:44:10 -04:00

Finished at: Wed Jul 27 2016, 04:45:10 -04:00

Test link: https://www.blitz.io/to#/play

Analysis

This rush generated **77,850** successful hits in **60 seconds** and we transferred **1.38 GB** of data in and out of your app. The average hit rate of **1,298/second** translates to about **112,104,000** hits/day.

The average response time was **132 ms**.

You've got bigger problems, though: **0.02%** of the users during this **rush** experienced timeouts or errors!

Response Times	Test Configuration	Other Stats
Fastest: **131** ms	Region: **virginia**	Avg. Hits: **1,298** /sec
Slowest: **135** ms	Duration: **60** seconds	Transfered: **11.32**MB
Average: **132** ms	Load: **1-3000** users	Received: **1,401.79**MB

Hits **99.98%** (77850)

Errors **0.01%** (11)

Timeouts **0.00%** (1)

Hits

This rush generated **77,850** successful hits. The number of hits includes all the responses listed below. For example, if you only want **HTTP 200 OK** responses to count as Hits, then you can specify **--status 200** in your rush.

Code	Type	Description	Amount
200	HTTP	OK	77850

HITS

HTTP 200 OK **100%** (77850)

Errors

The first error happened at **10 seconds** into the test when the number of concurrent users was at **496**. Errors are usually caused by resource exhaustion issues, like running out of file descriptors or the connection pool size being too small (for SQL databases).

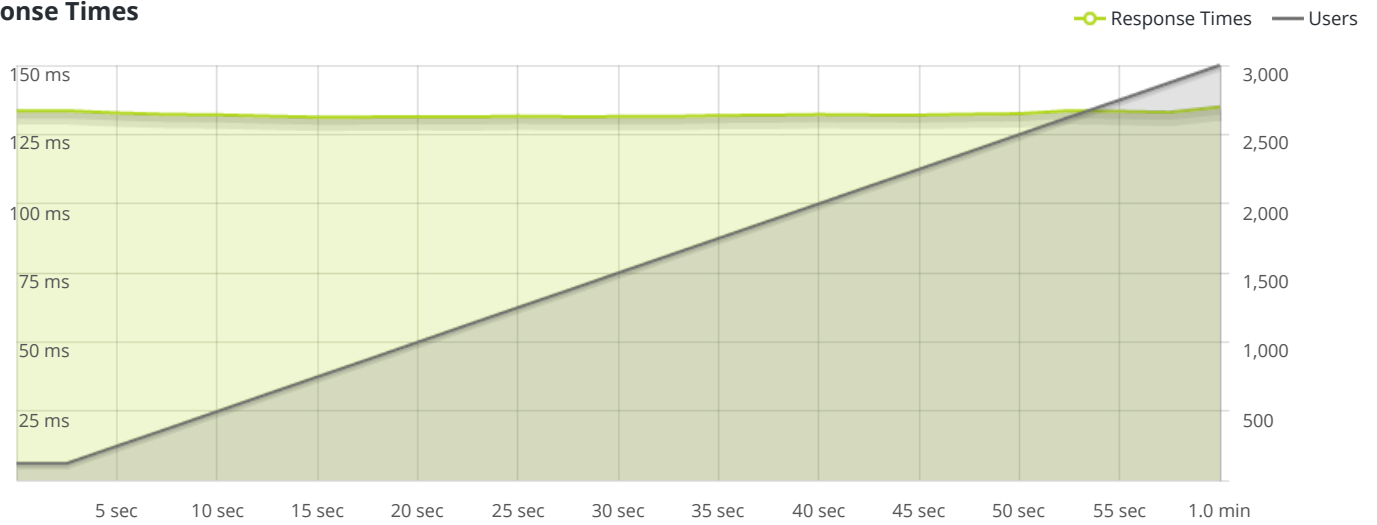| Code	Type	Description	Amount
17	TCP	Connection reset	8
23	TCP	Connection timeout	3

ERRORS

Connection reset **73%** (8)

Connection timeo... **27%** (3)

Timeouts

The first timeout happened at **52.5 seconds** into the test when the number of concurrent users was at **2625**. Looks like you've been rushing with a timeout of **1000 ms**. Timeouts tend to increase with concurrency if you have lock contention of sorts. You might want to think about in-memory caching using redis, memcached or varnish to return stale data for a period of time and asynchronously refresh this data.

Response Times
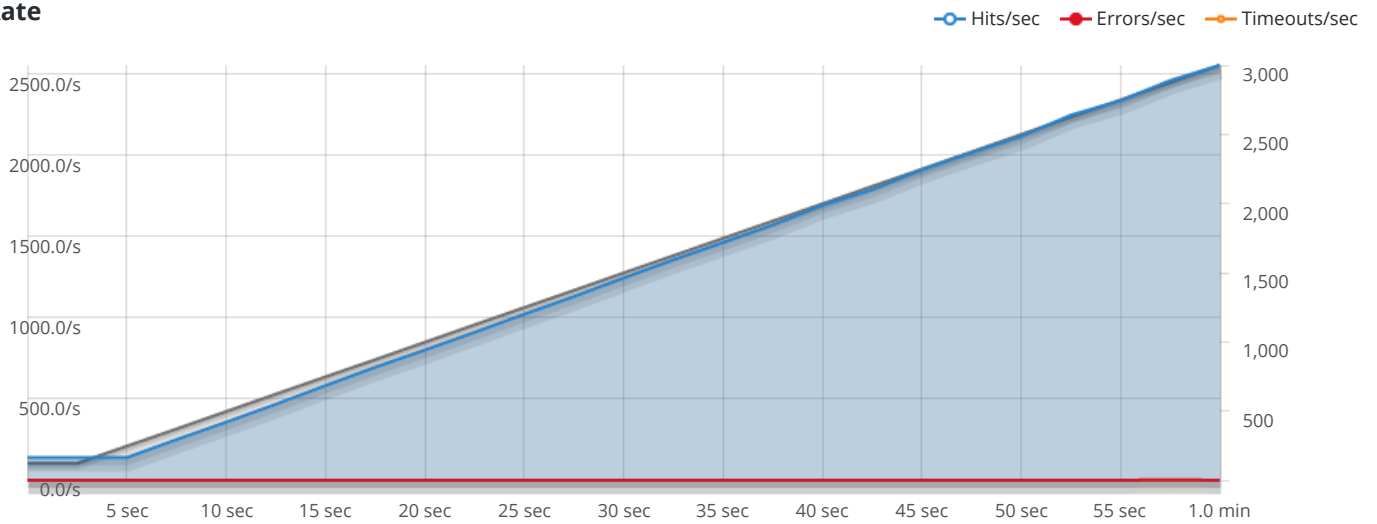
150 ms   3,000

125 ms   2,500

100 ms   2,000

75 ms   1,500

50 ms   1,000

25 ms   500

5 sec   10 sec   15 sec   20 sec   25 sec   30 sec   35 sec   40 sec   45 sec   50 sec   55 sec   1.0 min

STEP 1
Response Times

The max response time was: **134 ms @ 3000 users**

Hit Rate

Hits/sec   Errors/sec   Timeouts/sec

2500.0/s   3,000

    2,500

2000.0/s   2,000

1500.0/s   1,500

1000.0/s   1,000

500.0/s   500

0.0/s

5 sec   10 sec   15 sec   20 sec   25 sec   30 sec   35 sec   40 sec   45 sec   50 sec   55 sec   1.0 min

STEP 1
Hits/sec   Errors/sec   Timeouts/sec

The max hit rate was: **2,555 hits per second**