

LOAD TEST REPORT

DATE: 6/24/2014

TEST FROM : VIRGINIA

Query URL: http://198.199.81.108/

Started at: Tue Jun 24 2014, 12:10:43 -04:00

Finished at: Tue Jun 24 2014, 12:10:43 -04:00

ANALYSIS

This rush generated **1,050** successful hits in **60 seconds** and we transferred **8.01 MB** of data in and out of your app. The average hit rate of **18/second** translates to about **1,512,000** hits/day.

The average response time was **305 ms**.

RESPONSE TIMES	TEST CONFIGURATION	OTHER STATS
FASTEST: 90 ms	REGION: VIRGINIA	AVG. HITS: 18 /SEC
SLOWEST: 729 ms	DURATION: 60 SECONDS	DATA TRANSFERED: 8.01 MB
AVERAGE: 305 ms	LOAD: 1-50 USERS	



HITS **97.95%** (1050)
ERRORS **0.00%** (0)
TIMEOUTS **2.05%** (22)

HITS

This rush generated **1,050** 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	1050

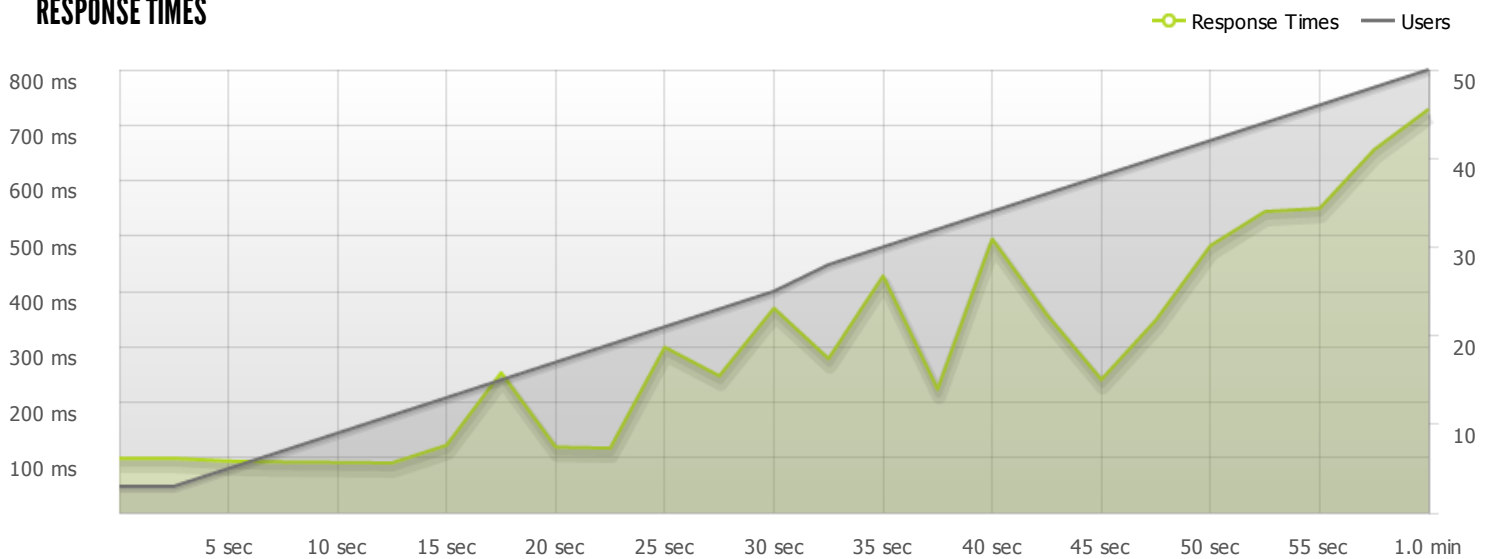


HTTP 200 OK **100%** (1050)

TIMEOUTS

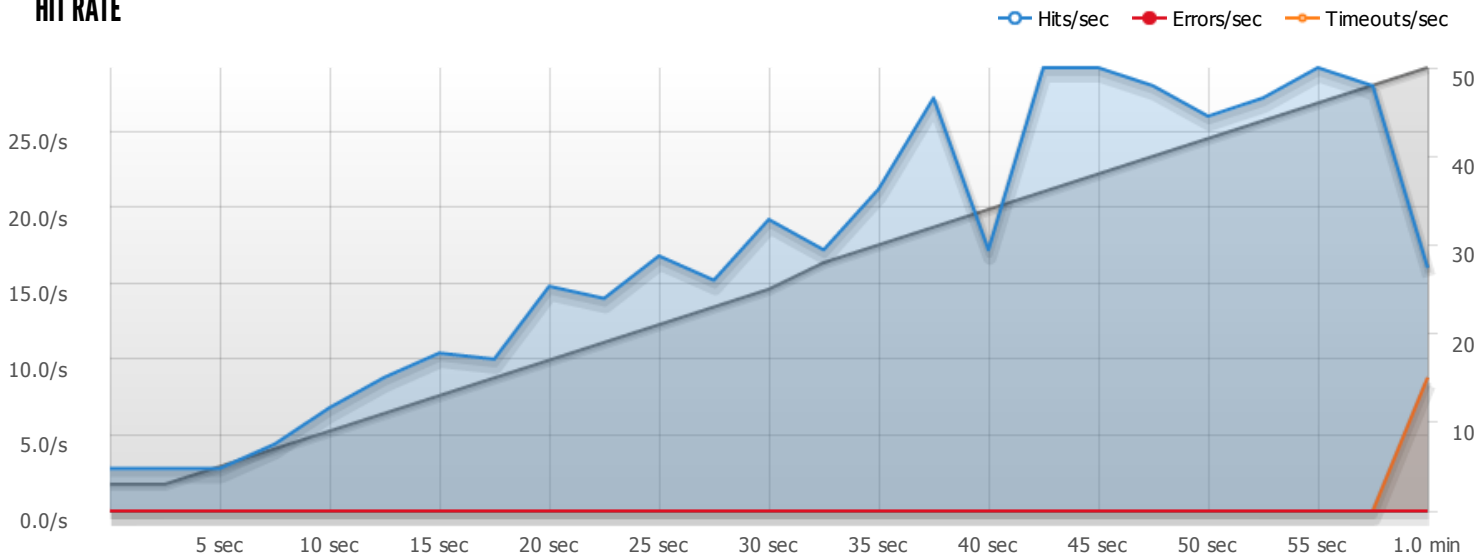
The first timeout happened at **60 seconds** into the test when the number of concurrent users was at **50**. Looks like you've been rushing with a timeout of **1000 ms**. Timeouts tend to increase with concurrency if you have lock contention of sorts. You might want to think about in-memory caching using [redis](#), [memcached](#) or [varnish](#) to return stale data for a period of time and asynchronously refresh this data.

RESPONSE TIMES



The max response time was: **728 ms @ 50 users**

HIT RATE



The max hit rate was: **29 hits per second**