

LOAD TEST REPORT

DATE: 7/24/2014

TEST FROM : VIRGINIA

Query URL: http://wordpress-1055-3238-5443.cloudwaysapps.com/

Started at: Thu Jul 24 2014, 12:19:47 -04:00

Finished at: Thu Jul 24 2014, 12:19:47 -04:00

ANALYSIS

This rush generated **25,498** successful hits in **60 seconds** and we transferred **241.35 MB** of data in and out of your app. The average hit rate of **425/second** translates to about **36,717,120** hits/day.

The average response time was **338 ms**.

You've got bigger problems, though: **34.86%** of the users during this **rush** experienced timeouts or errors!

RESPONSE TIMES

FASTEST: **5ms**

SLOWEST: **683ms**

AVERAGE: **338ms**

TEST CONFIGURATION

REGION: **VIRGINIA**

DURATION: **60**SECONDS

LOAD: **1-2000**USERS

OTHER STATS

AVG. HITS: **425**/SEC

DATA TRANSFERRED:

241.35MB



HITS **65.14%** (25498)

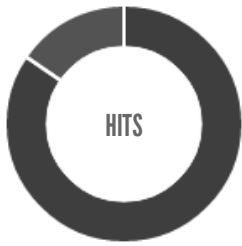
ERRORS **12.21%** (4780)

TIMEOUTS **22.65%** (8865)

HITS

This rush generated **25,498** 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 | 21565 |
| 500 | HTTP | Internal Server Error | 3933 |



HTTP 200 OK **85%** (21565)
HTTP 500 INTERNAL SE... **15%** (3933)

ERRORS

The first error happened at **22.5 seconds** into the test when the number of concurrent users was at **748**. 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 |
|------|------|--------------------|--------|
| 23 | TCP | Connection timeout | 4780 |

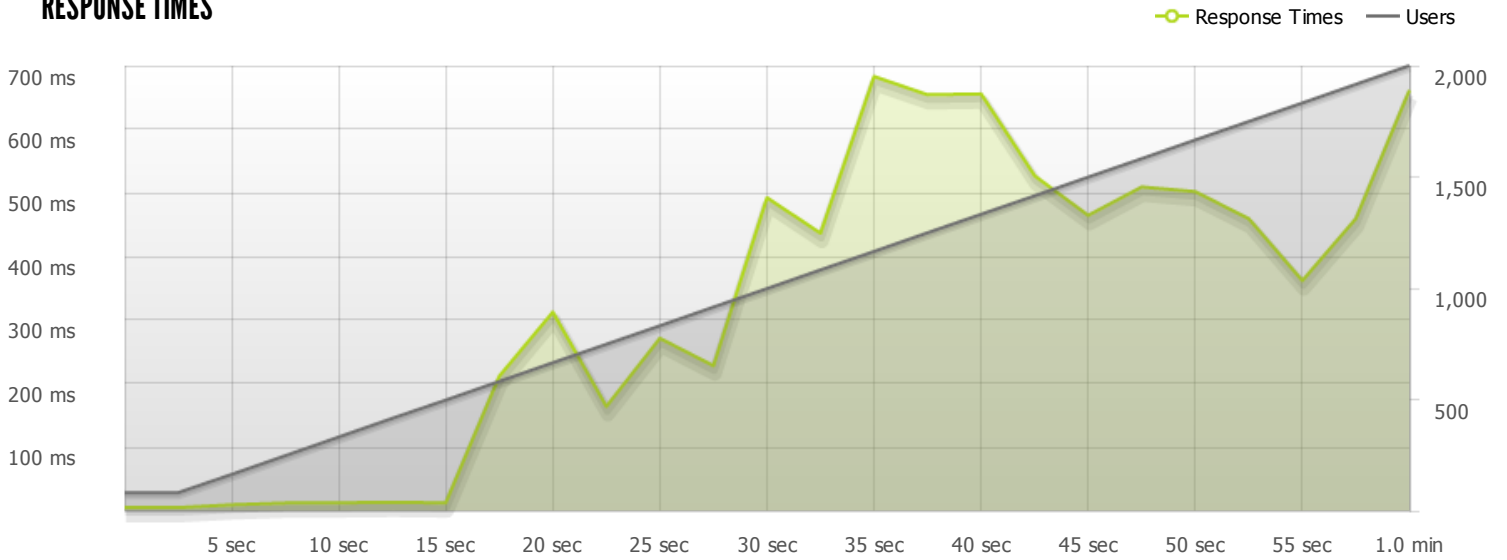


CONNECTION TIMEOUT **100%** (4780)

TIMEOUTS

The first timeout happened at **20 seconds** into the test when the number of concurrent users was at **665**. 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.

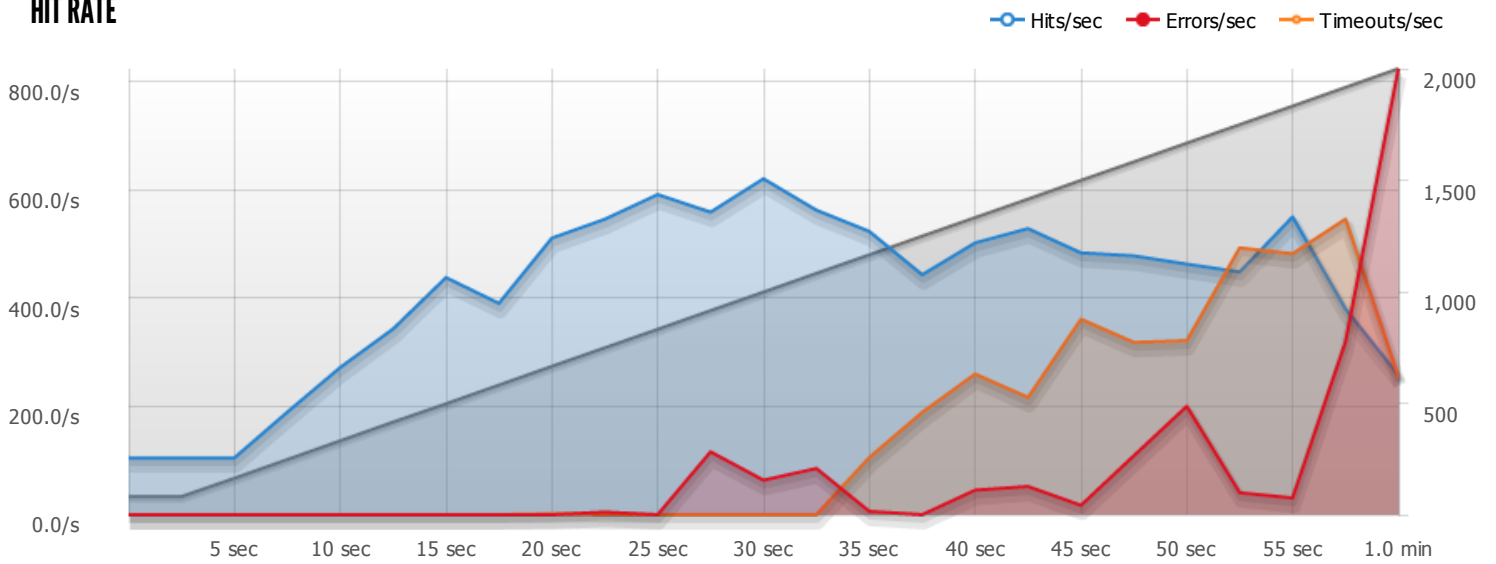
RESPONSE TIMES



STEP 1
Response Times

The max response time was: **683 ms @ 1165 users**

HIT RATE



STEP 1
Hits/sec Errors/sec Timeouts/sec

The max hit rate was: **620 hits per second**