June 6, 2019

Stress-testing report

# Test content

This test is a stress- testing which focus on a simple containerized app-server with two HTTP restful API.

## 

During this test, both of the URI will be tested with designed scenarios and required conditions.

# testers & Testing time

Tester Name: Daniel Xu

Testing Time: 6/6/2019

Testing Location: Lab 1, R block, The University of Waikato.

# TEST METHODS

This test will use Apache open source test tool, JMeter.

# test target

Collect the results from JMeter and using the knowledge from Queueing theory, Markov chains and computer systems to explain the results.

# test environment

## Environment

## OS: Linux

## Machine: DELL D18M

## Scenario Design

### Scenario One

Repeatedly check if the number “2147483647” is a prime.

Threads(users): 50

Repeat Timing: 60s

### Scenario Two

First check all the number between 1 to 100 is a prime.

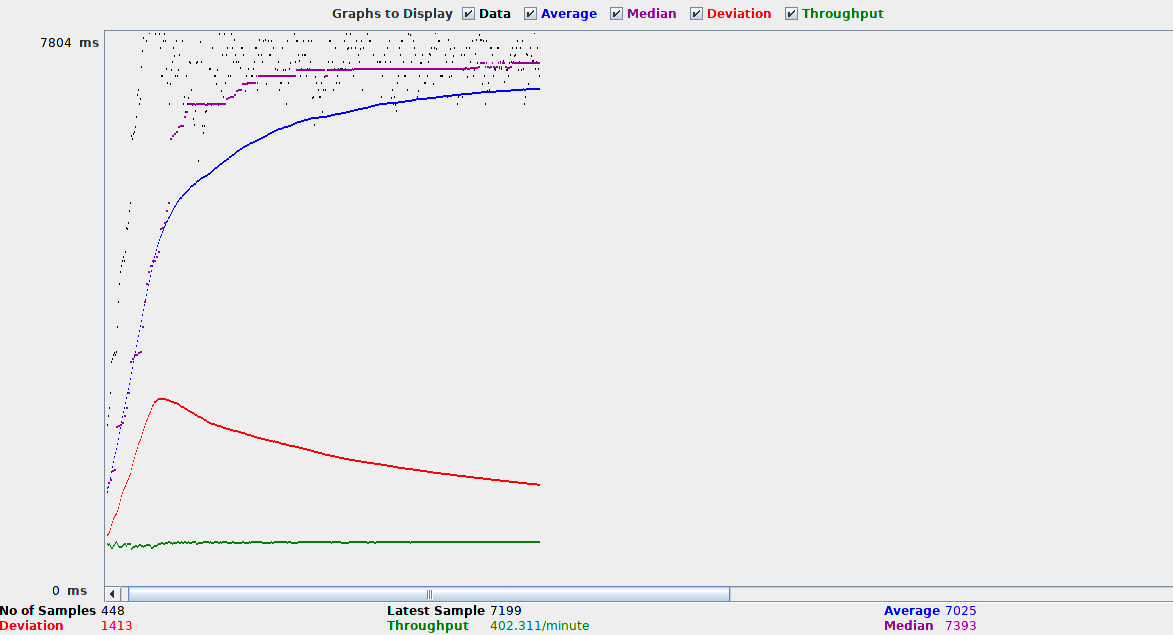
Then repeatedly run primeStored URI.

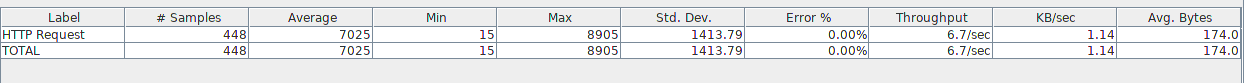
Threads(users): 50

Repeat Timing: 60s

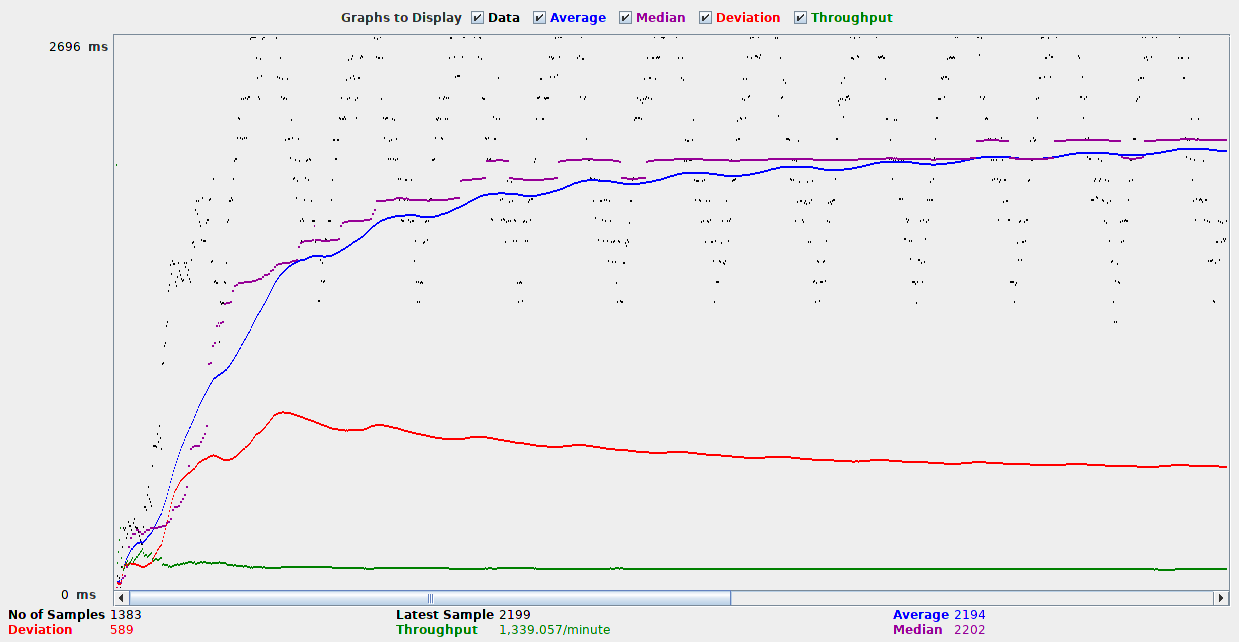
# Performance test results and analysis

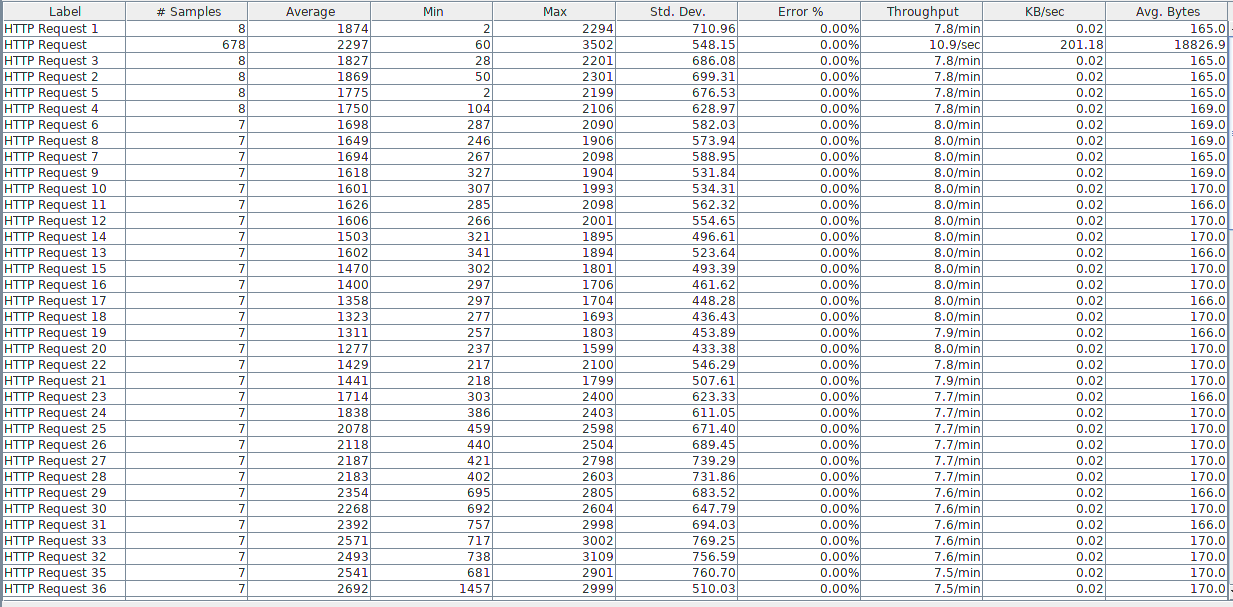
#### **Scenario One**

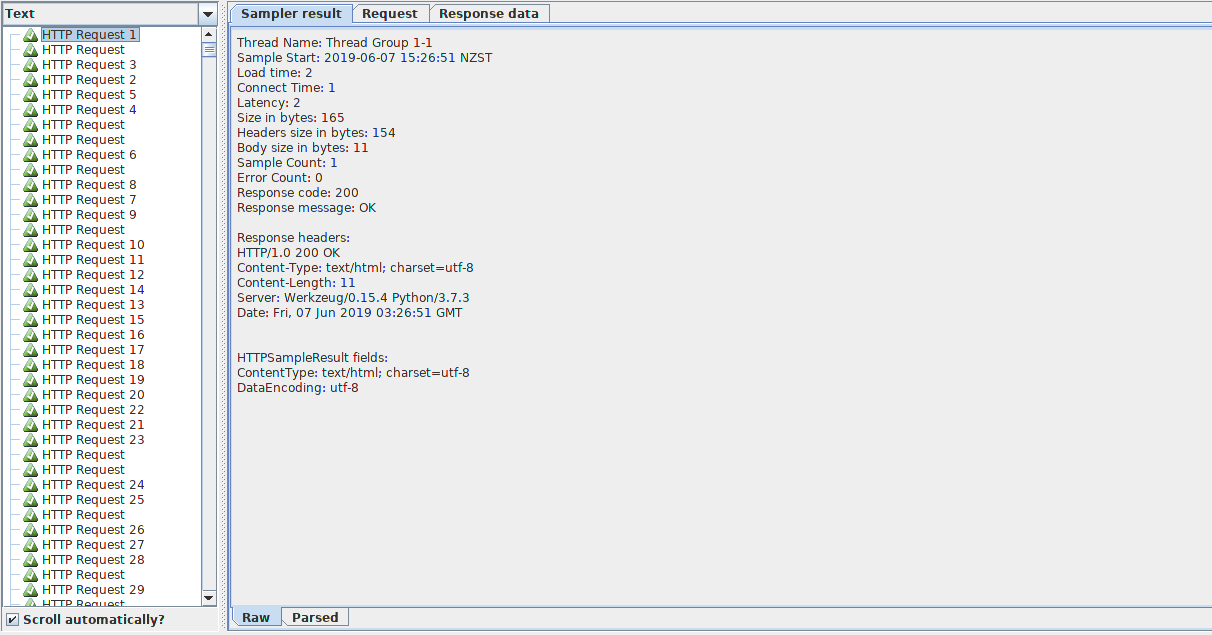




#### **Scenario Two**





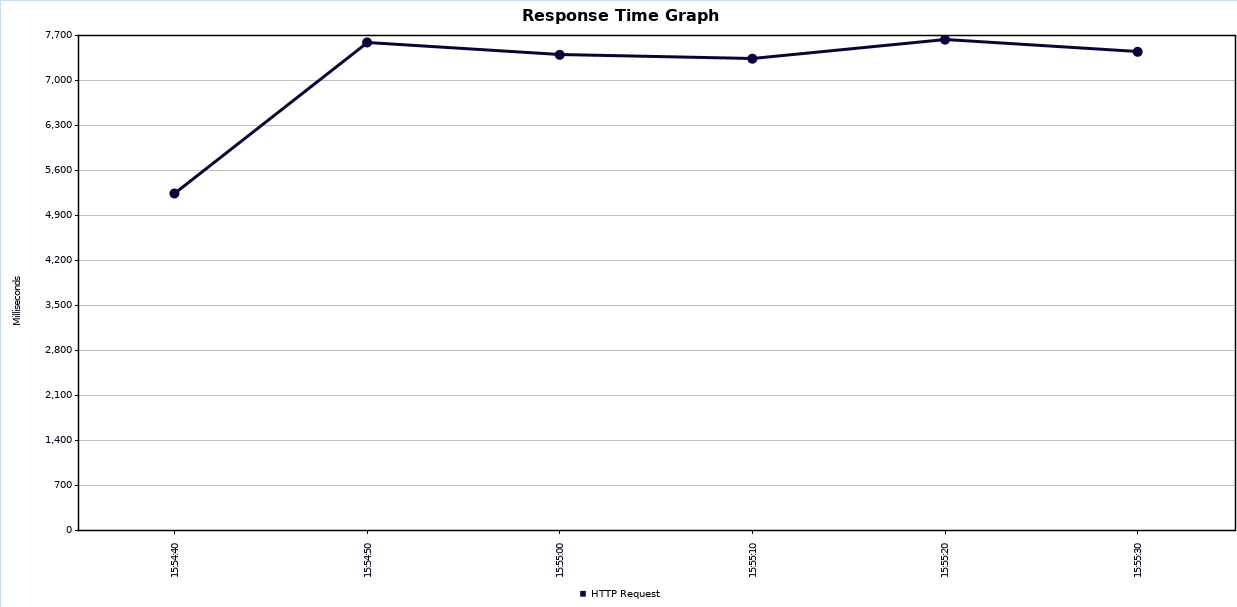


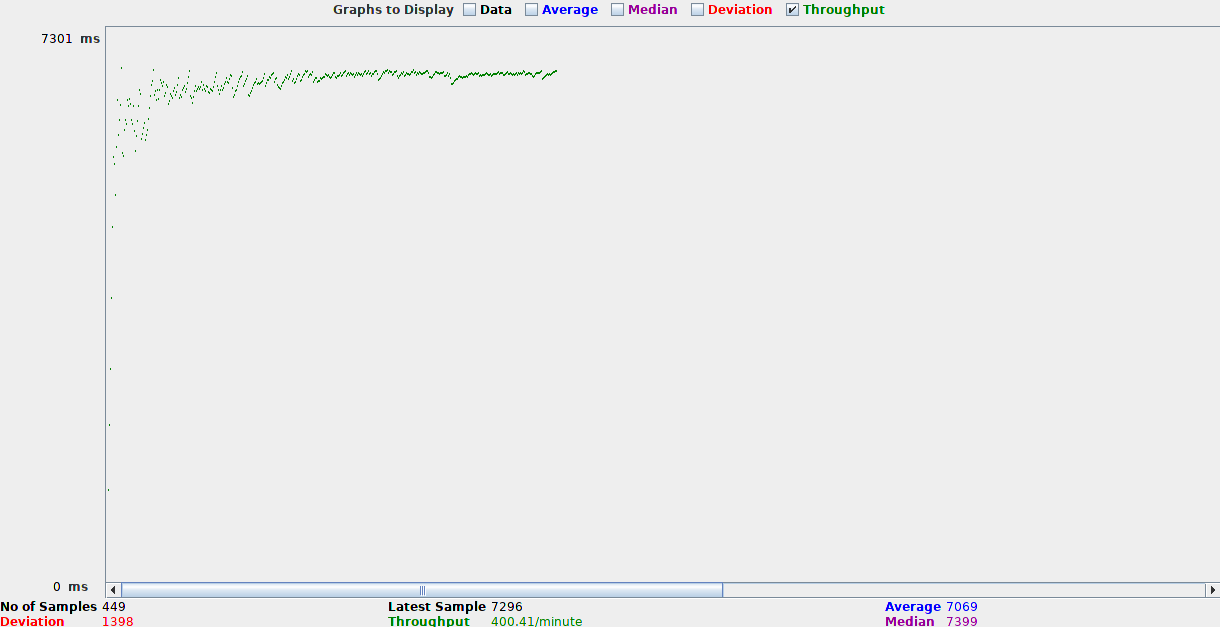
#### **Response-Time and Throughput data**

##### Three different CPU limits

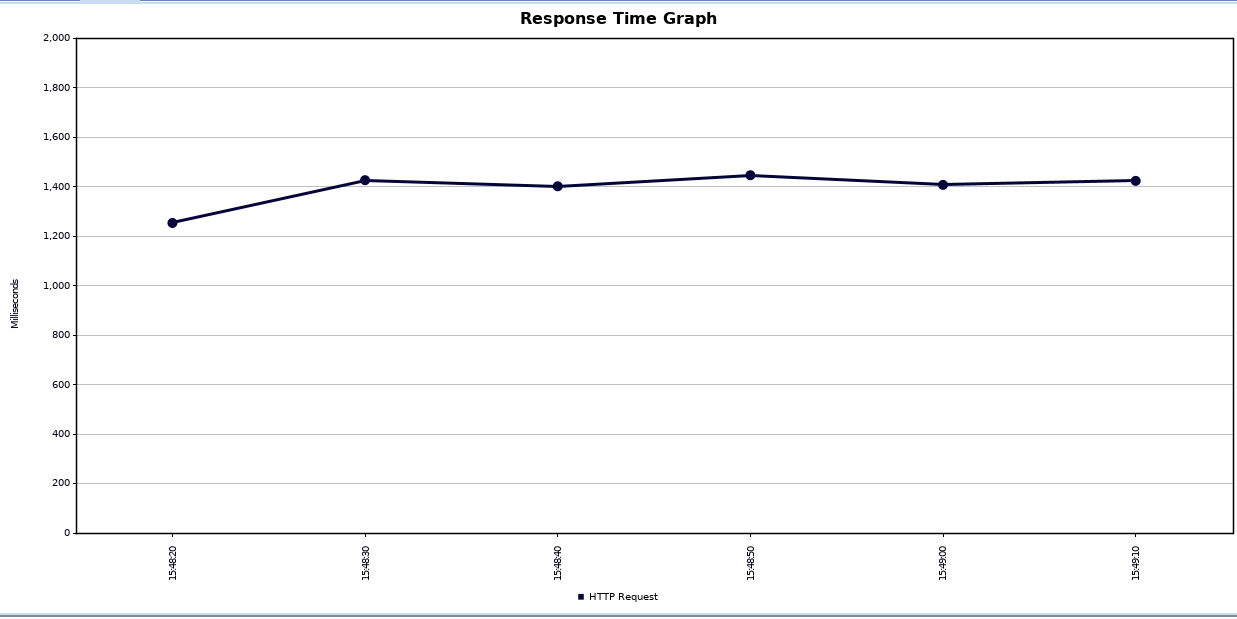
##### To test the response time on first scenario.

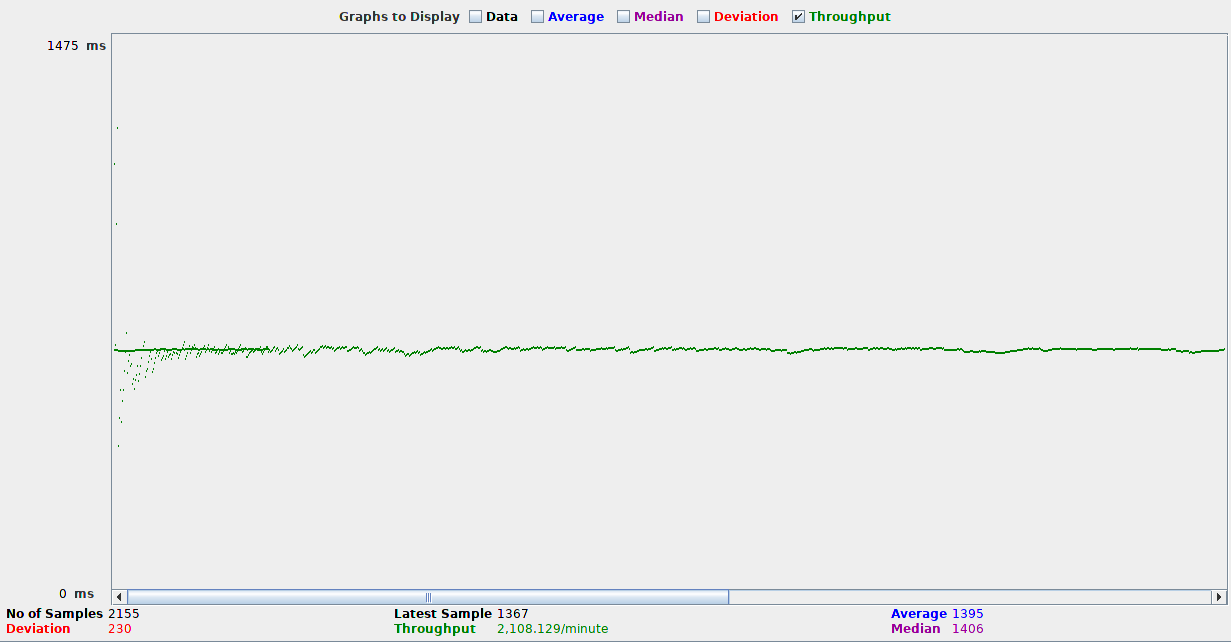
### When CPU : 0.1



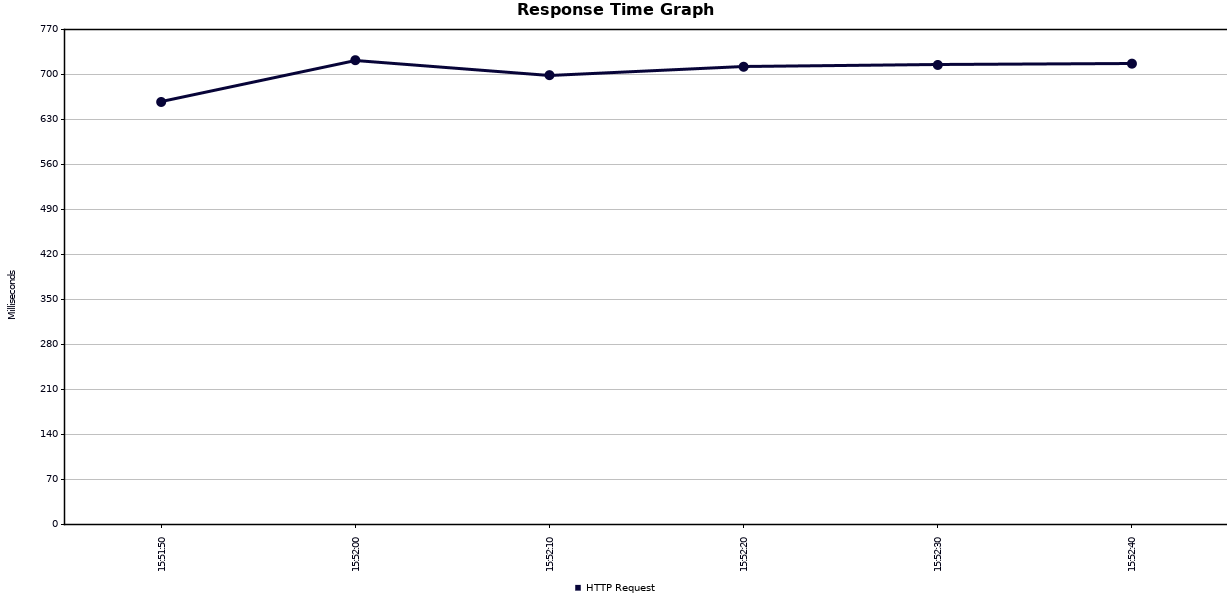


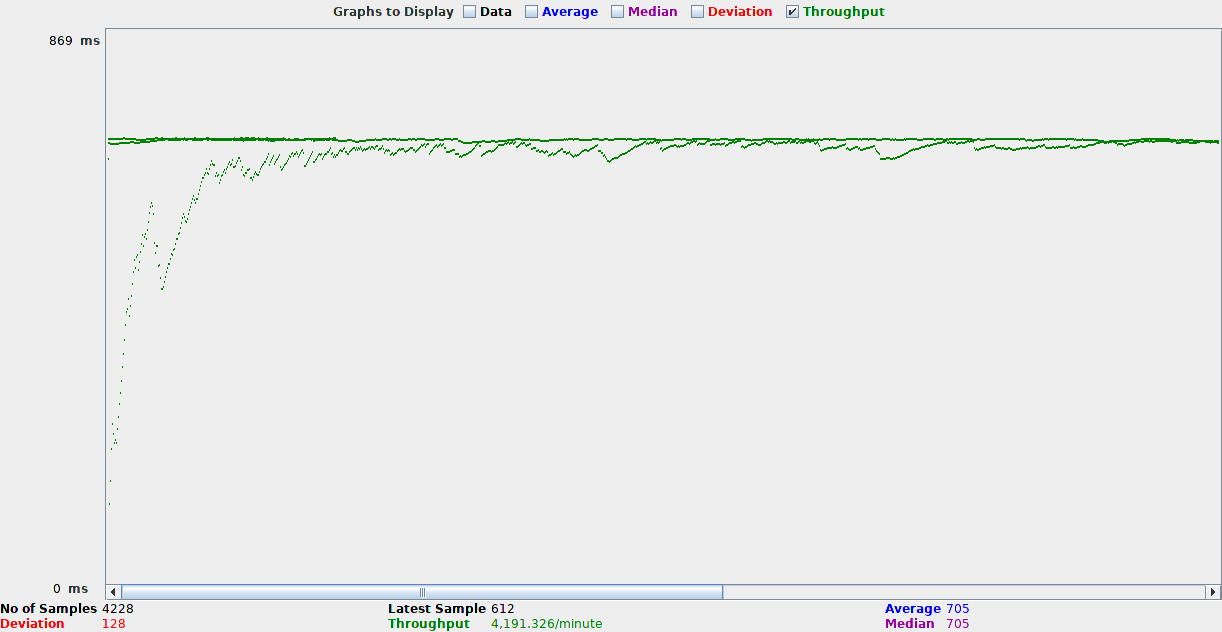
### When CPU : 0.5





### When CPU : 1



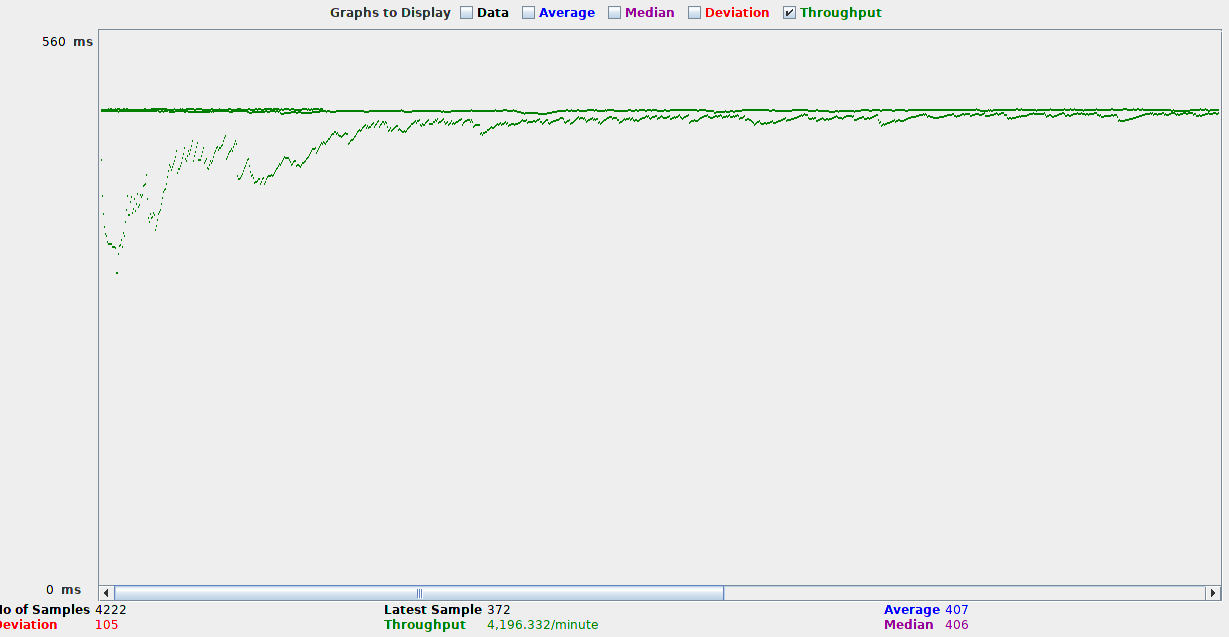


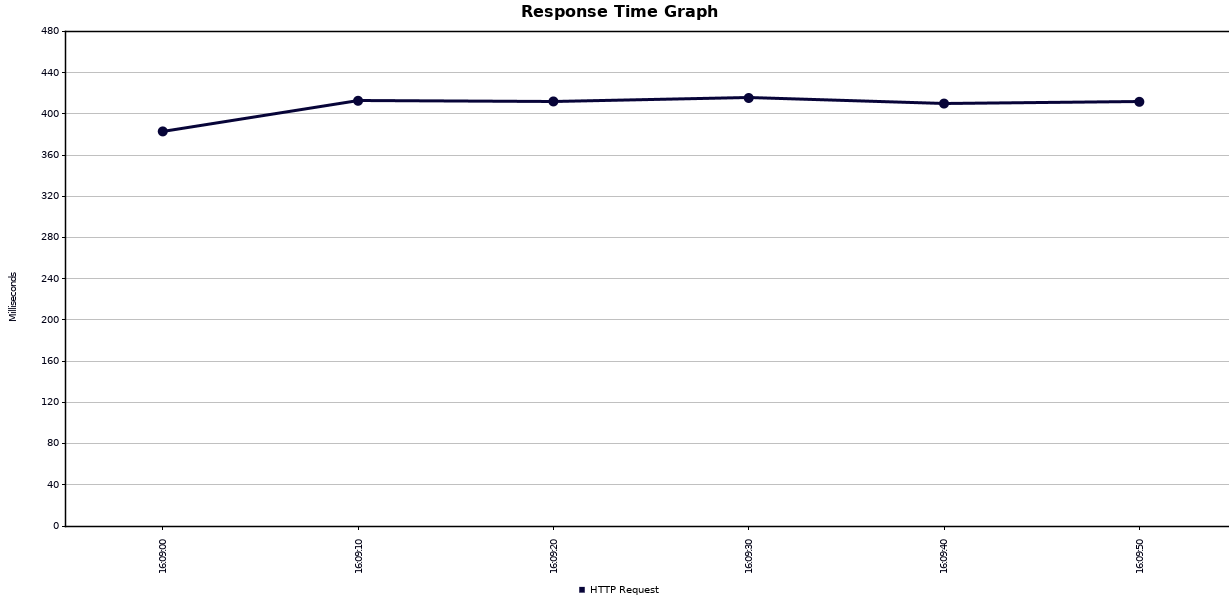
So we can see from the graph above, the more powerful the cpu is, the system will respond faster and throughput is much more.

##### Three different timer delays in JMeter

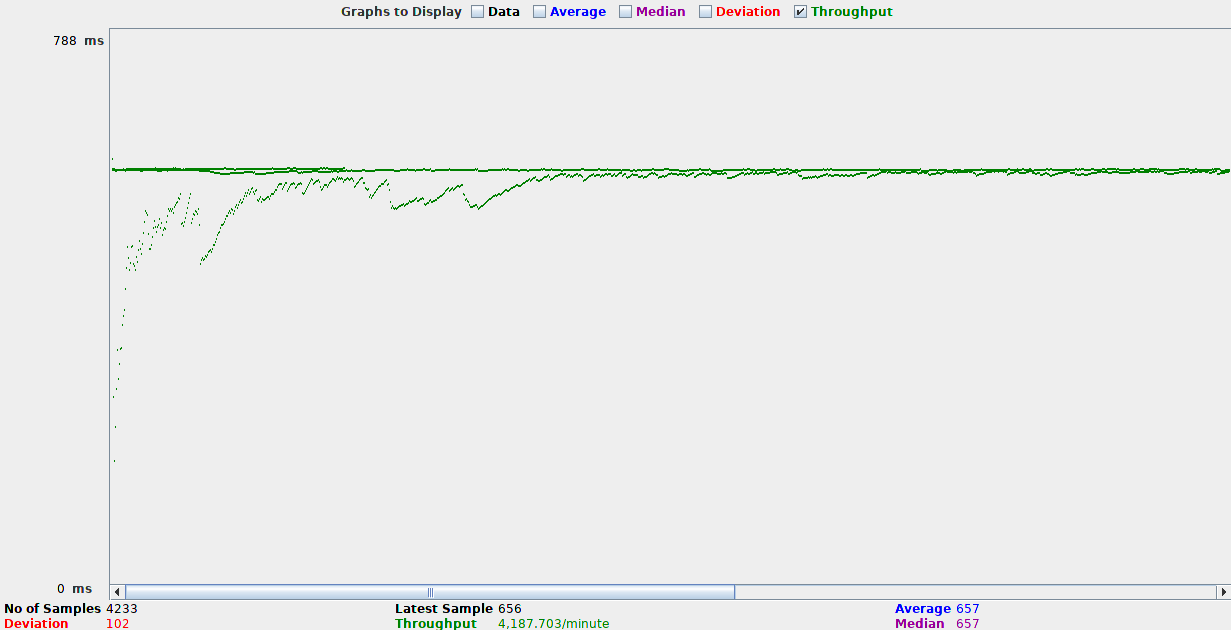
##### To test the response time on first scenario.

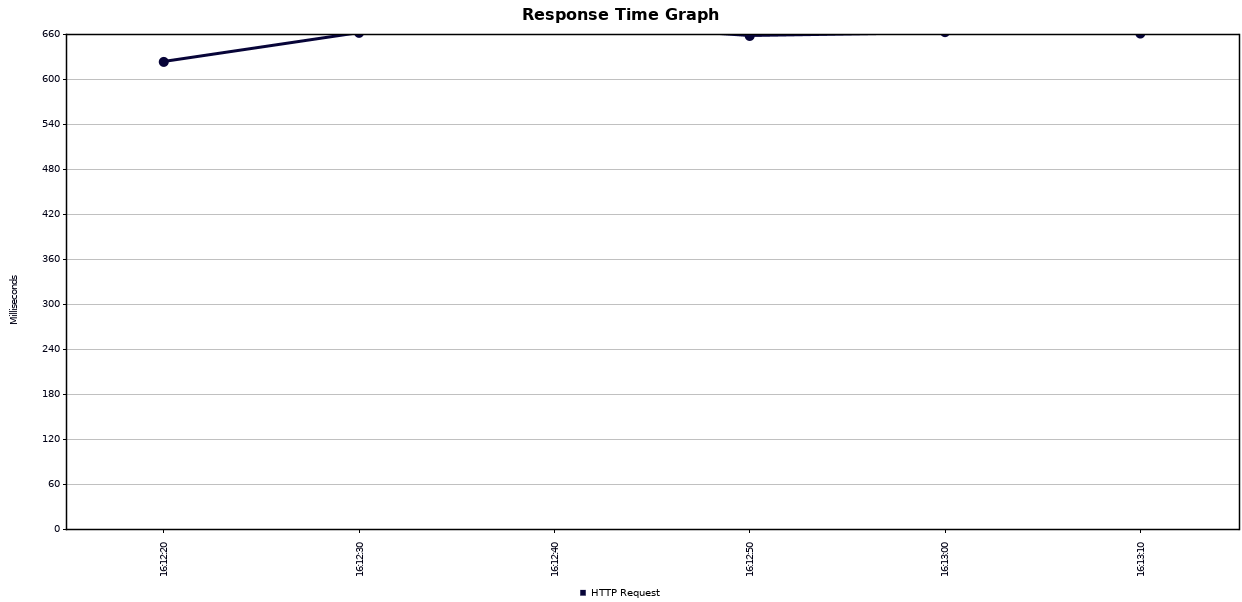
###### Constant Timer



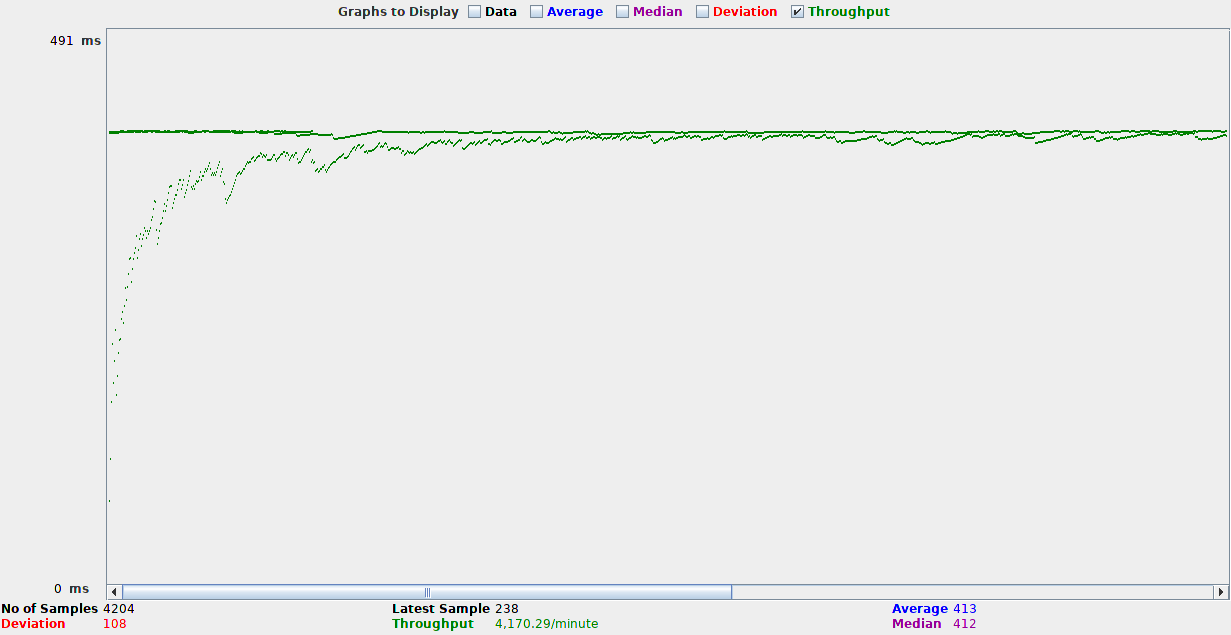


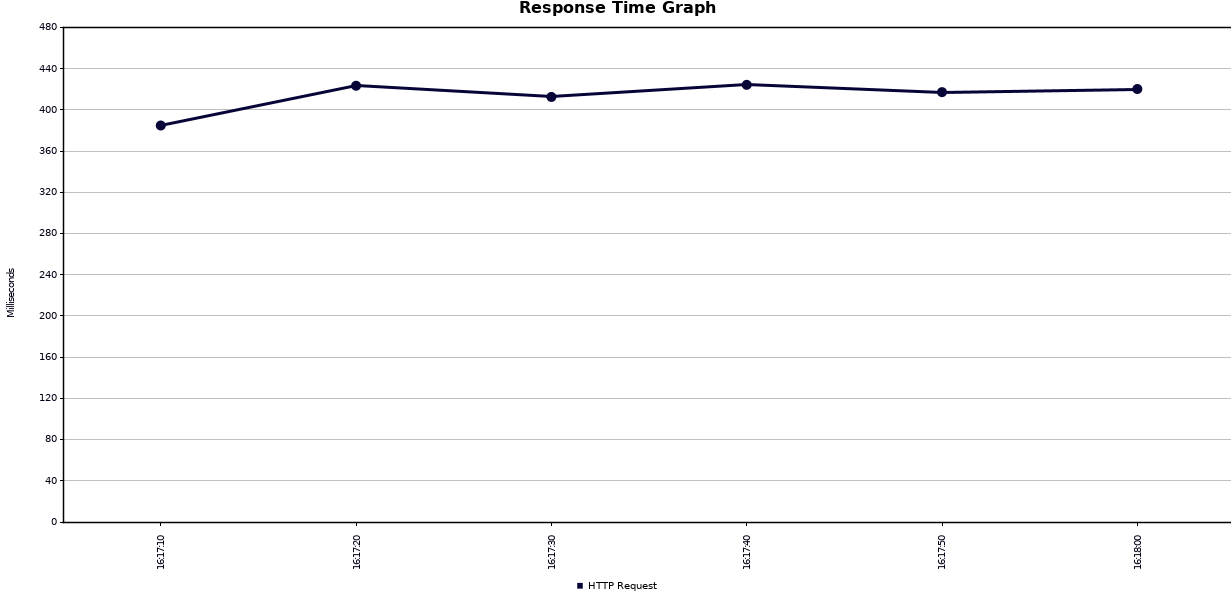
###### Uniform Random Timer





###### Gaussian Random Timer





# REview

During the Stress-test, the system did not have adverse reactions during the concurrent use and repeated operation from the user, and the system responded well. Under special designed conditions, the system response time was satisfactory and the stability was reliable.

Under pressure, the speed did not decrease, there was no increase in waiting time, and the user experience was not affected.