-: CONTENTS :-

Login Screen...200 User --Error %	0.00%
Login Screen ….400 User --Error %	0.29%
Contact list page...200 User --Error %	3.31%
Contact List ….400 User --Error %	16.46%
Design Report and assign Contact 400 user	16.06%
View reply on assigned reports 400 user	12.39%
Response Report with 400 User	10.53%
Archive Report with 400 user	10.11%
Submit a Report with 400 User	13.67%
Login Screen...1000 User --Error %	4.04%
Contact List ….1000 User --Error %	39.87%
Create Contact --1000 User --Error %	49.45%
Design your report and assign to contacts	16.06%
Submit a report with 500 user-Error%	24.31%
My contacts with 500 user -Error %	17.26%
Response report: With 500 User --Error%	23.35%
Archive report: With 500 User-Error%	20.84%
Track Response report: With 500 User-Error%	19.46%
View reply from contacts: with 500 User-Error%	17.49%
Design your report and assign to contacts: with 500 User-Error%	22.36%

Project: Make my report

Testing: Performance Testing

Tool: J Meter

Performed by: Om shaswat acharya

Loop Count: **Loop Count** is used to specify the number of times to execute the Performance Test.

Thread Count: Thread Count defines the number of Users you want to simulate for the execution.

Ramp-Up Time: Ramp-Up is the amount of time Jmeter should take to get all the threads sent for the execution. Ramp-Up should be sufficient enough to avoid unnecessary and large work load from the beginning of the test execution.

For example, if Thread Count is 10 and the ramp-up period is 100 seconds, then Jmeter will take 100 seconds to get all 10 Threads up and running. First thread will be sent on 0th second (beginning of the execution) and then each thread will start after 10 seconds (100/10)

Loop Count: Using Loop Count you can specify the number of times to execute the Performance Test. You can select check-box "forever", it will keep on executing same Test script in loop until you manually stop the execution. There is also an option to manually define the loop count.

Now let's look at some examples to understand Thread Count + Ramp-Up Time + Loop Count combination.

Transactions per Second. This graph shows the number of transactions per second for each sampler. It counts for each seconds the number of finished transactions.

Hits per second/Throughput vs. Concurrent Users. Throughput is a measure of how many units of work are being processed. In the case of load testing, this is usually **hits per second**, also known as **requests per second**. Concurrent users are the number of users engaged with the app or site at a given time.

Response time is the **time** a system or functional unit takes to react to a given input.

1: Let's implement one scenario: With 200 User.

1:1 for Login Screen:

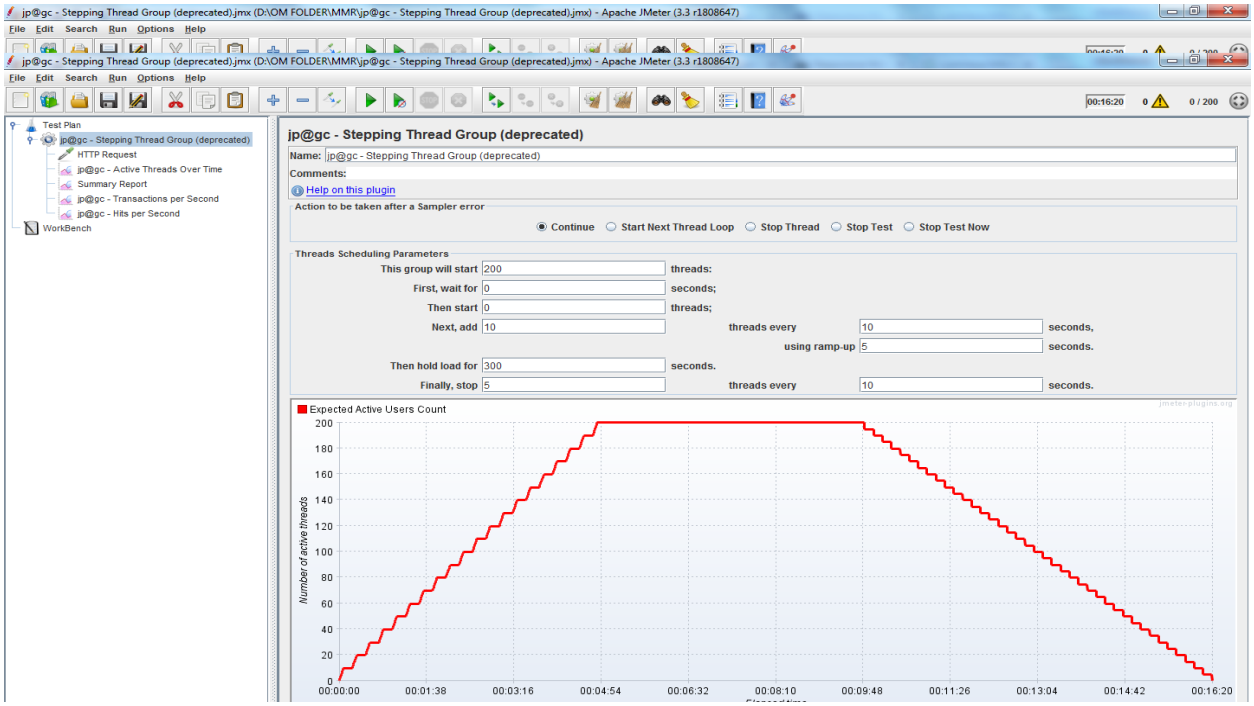
- Every 10 seconds 10 users will be added, until we reach 200 users. The first step is 1-10, the second 11-21, etc., because we defined 0 threads to run at the beginning.

- It will take each of the steps (with 10 users each) 10 seconds to complete. After that J-Meter waits 5 seconds before starting the next step.

- After reaching 200 threads all of them will continue running and hitting the server together for 5 minutes.

- At the end, 5 threads will stop every 10 seconds.

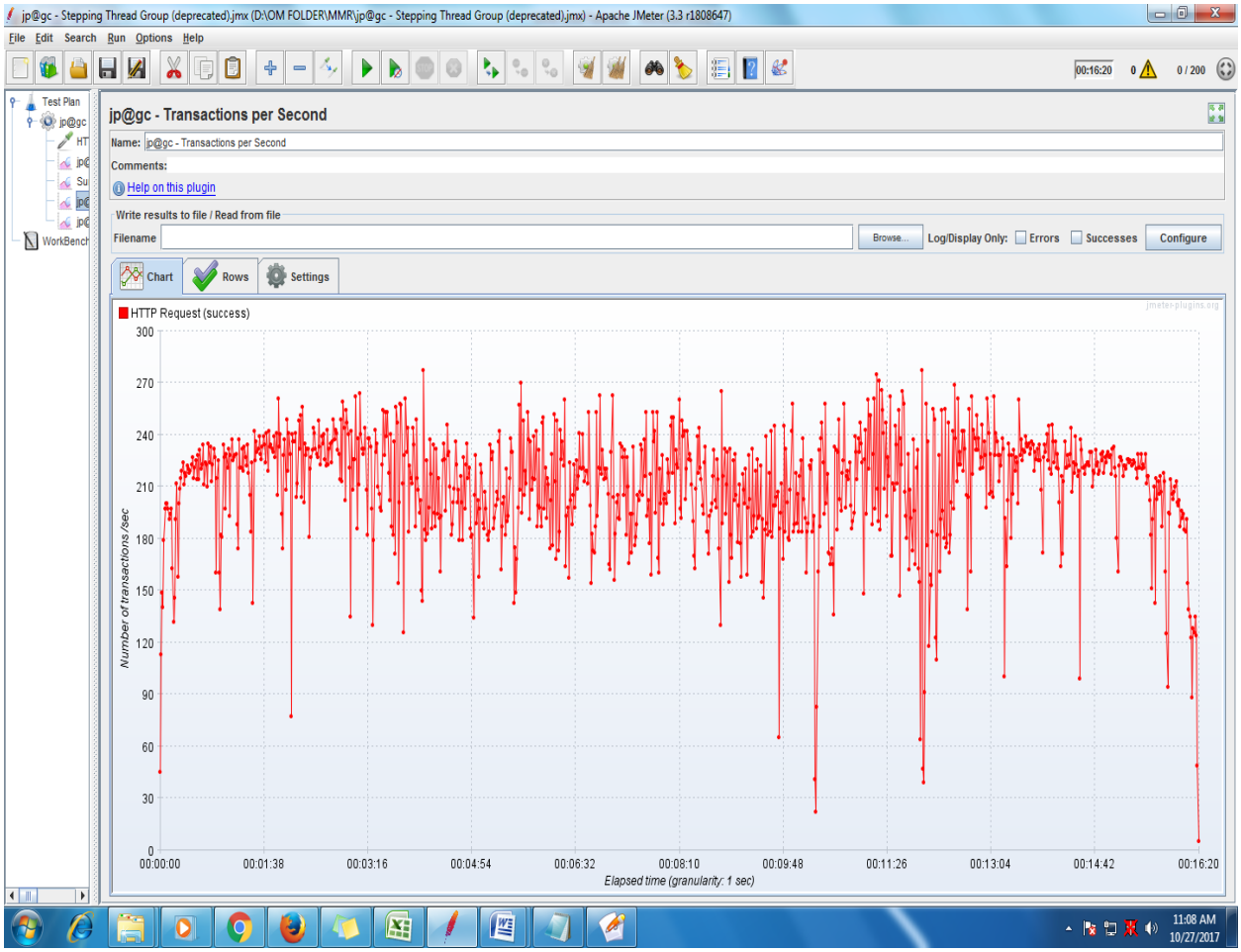
- **200 threads as target load**
- **0 seconds waiting after the test starts**
- **0 threads run at the immediate beginning of the test**
- **10 threads are added every 10 seconds with a ramp-up (or step transition time) of 5 seconds**
- **The target load is held for 300 seconds (5 minute)**
- **Finally, 5 threads are stopped every 10 seconds.**



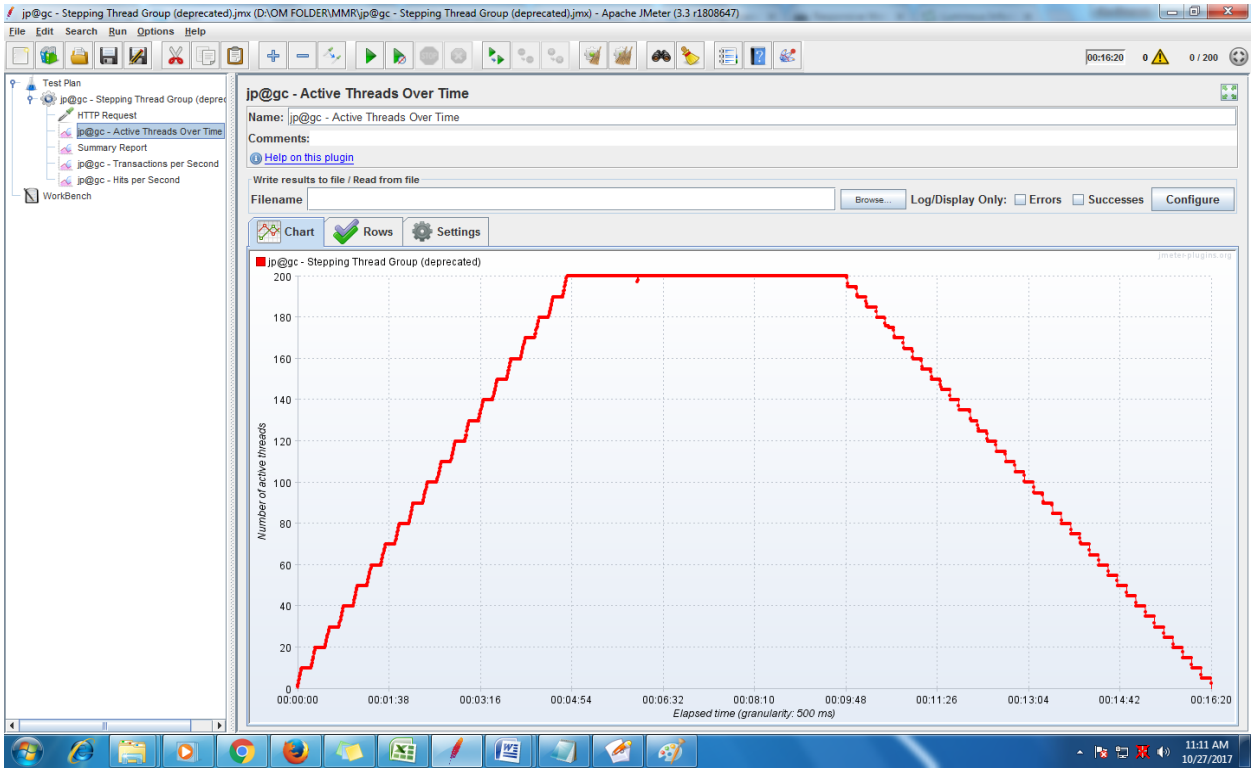
1.1.2 Summary Report:

Home page. Login Screen										
Lable	Sample	Average	Min	Max	Std.Dev	Error%	Throughout	Received KB/sec	Sent KB/sec	Avg. Bytes.
HTTP Request	206509	620	20	10113	464.01	0.00%	210.6/sec	4317.56	63.36	20988.7
TOTAL	206509	620	20	10113	464.01	0.00%	210.6/sec	4317.56	63.36	20988.7

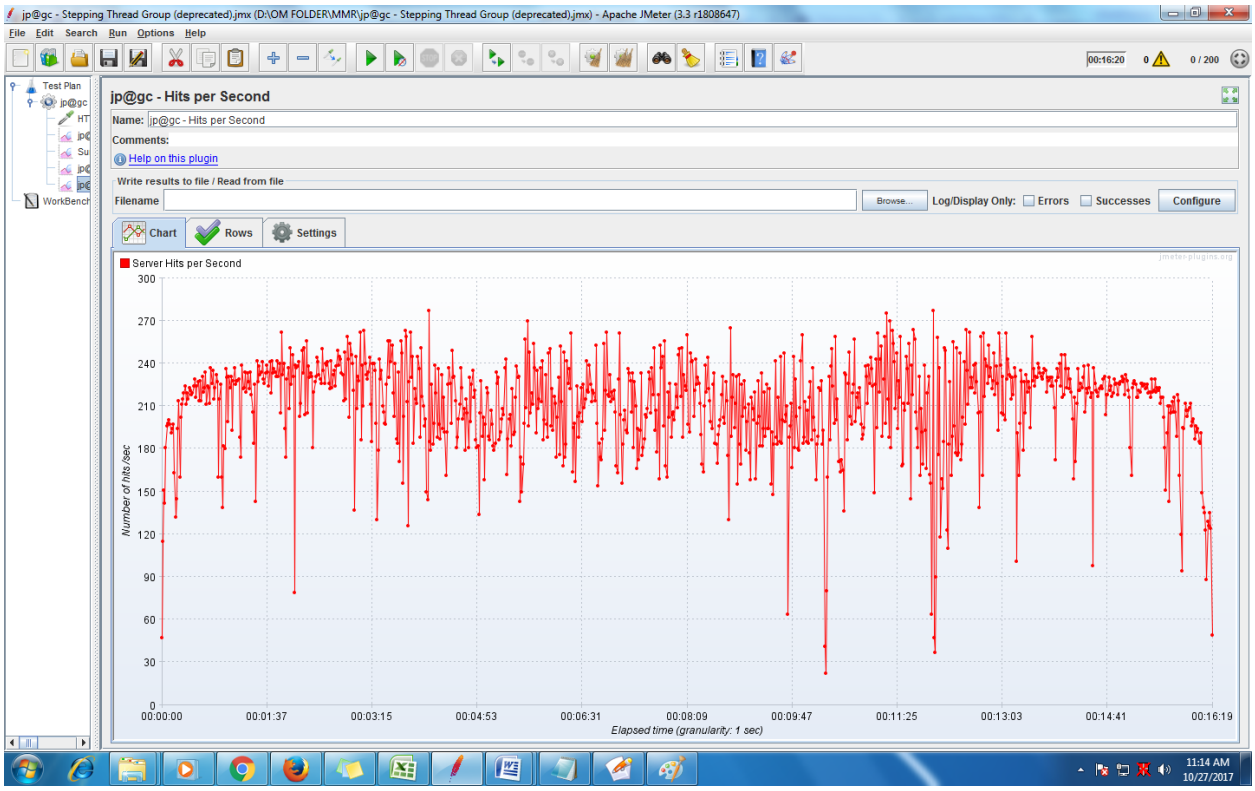
1.1.3 Transactions Per second:



1.1.4 Load test: Active threads over time:



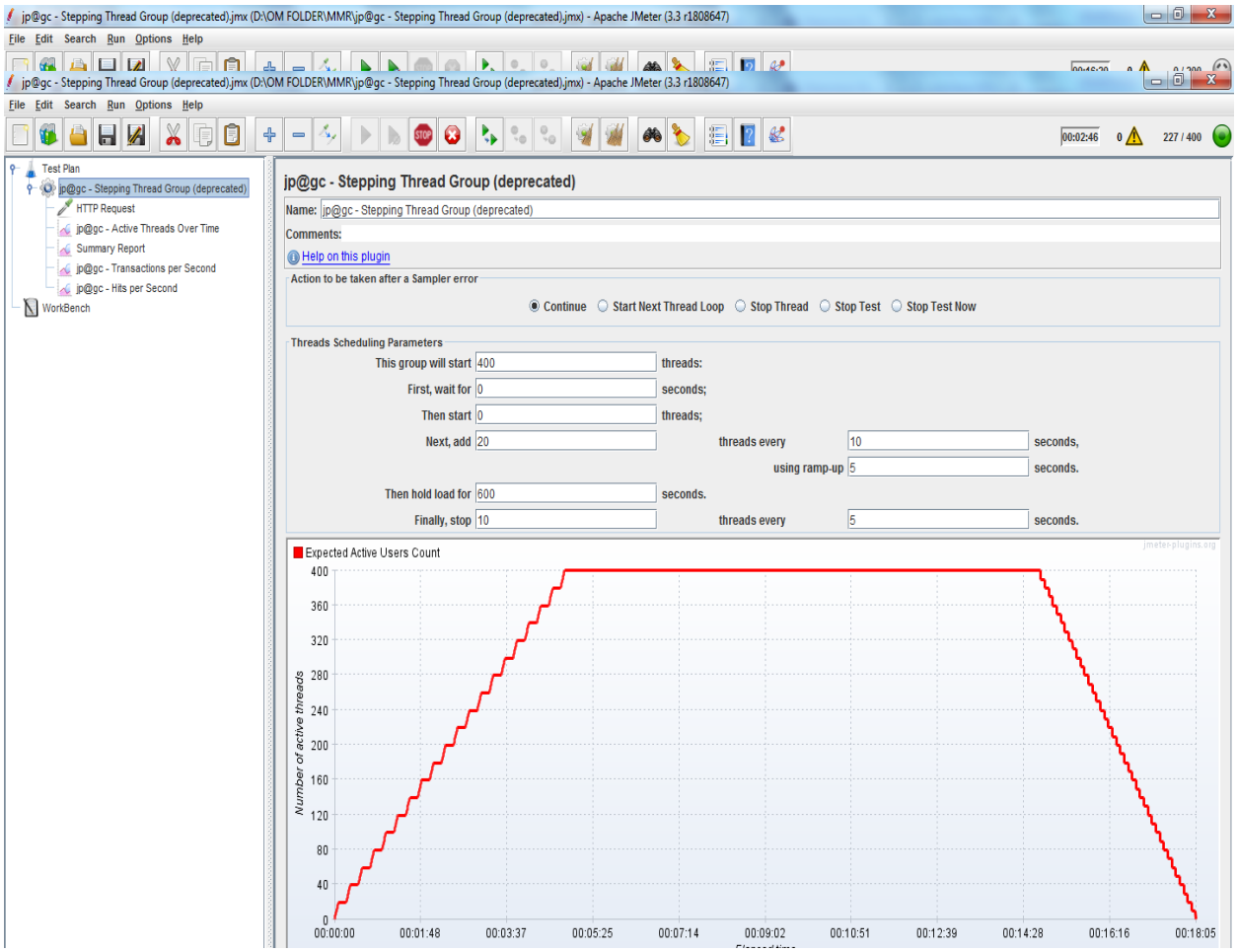
1.1.5 Hits per second:



2. Let's implement another scenario: With 400 User.

2.1 For Login screen:

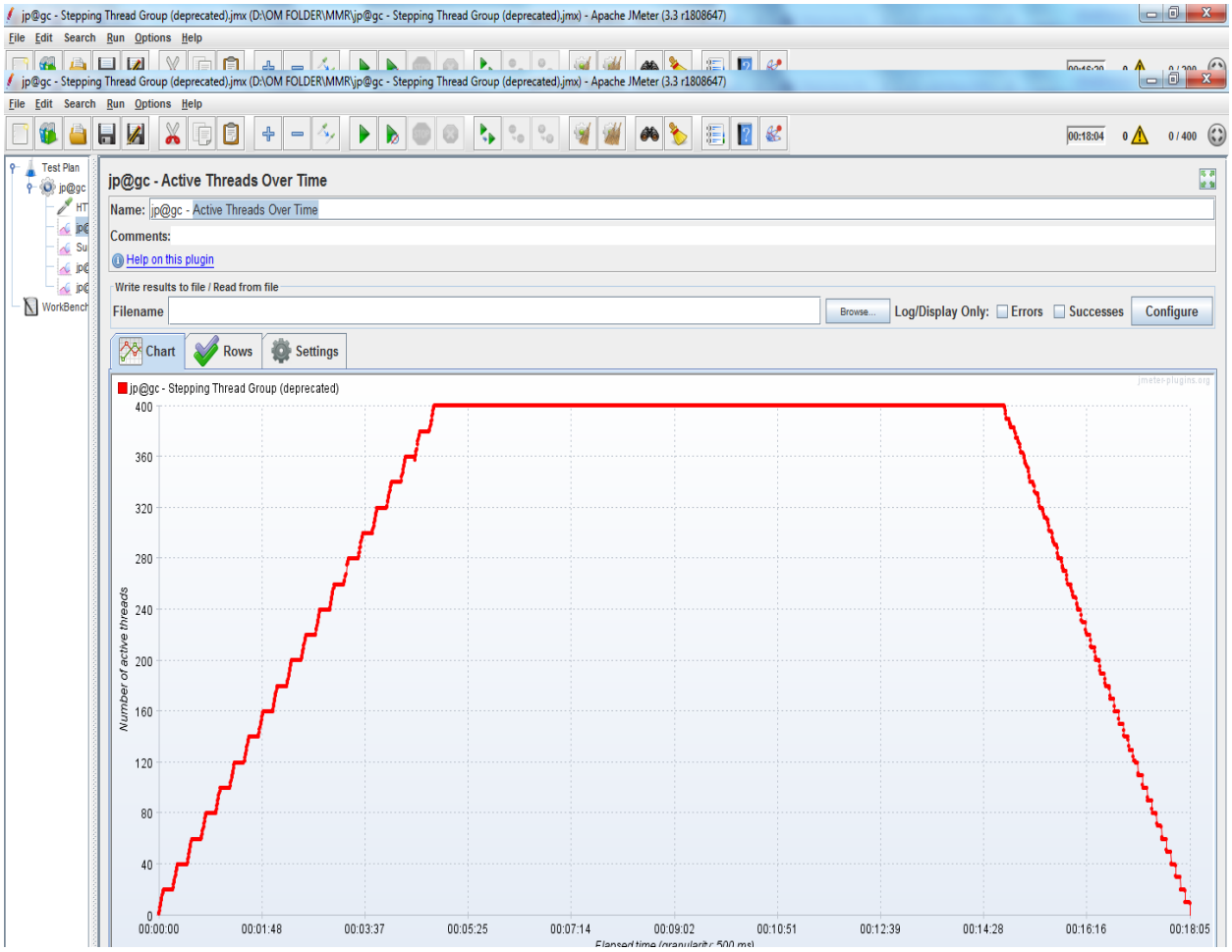
- **400 threads as target load**
- **0 seconds waiting after the test starts**
- **0 threads run at the immediate beginning of the test**
- **20 threads are added every 10 seconds with a ramp-up (or step transition time) of 5 seconds**
- **The target load is held for 600 seconds (10 minute)**
- **Finally, 10 threads are stopped every 5 seconds.**



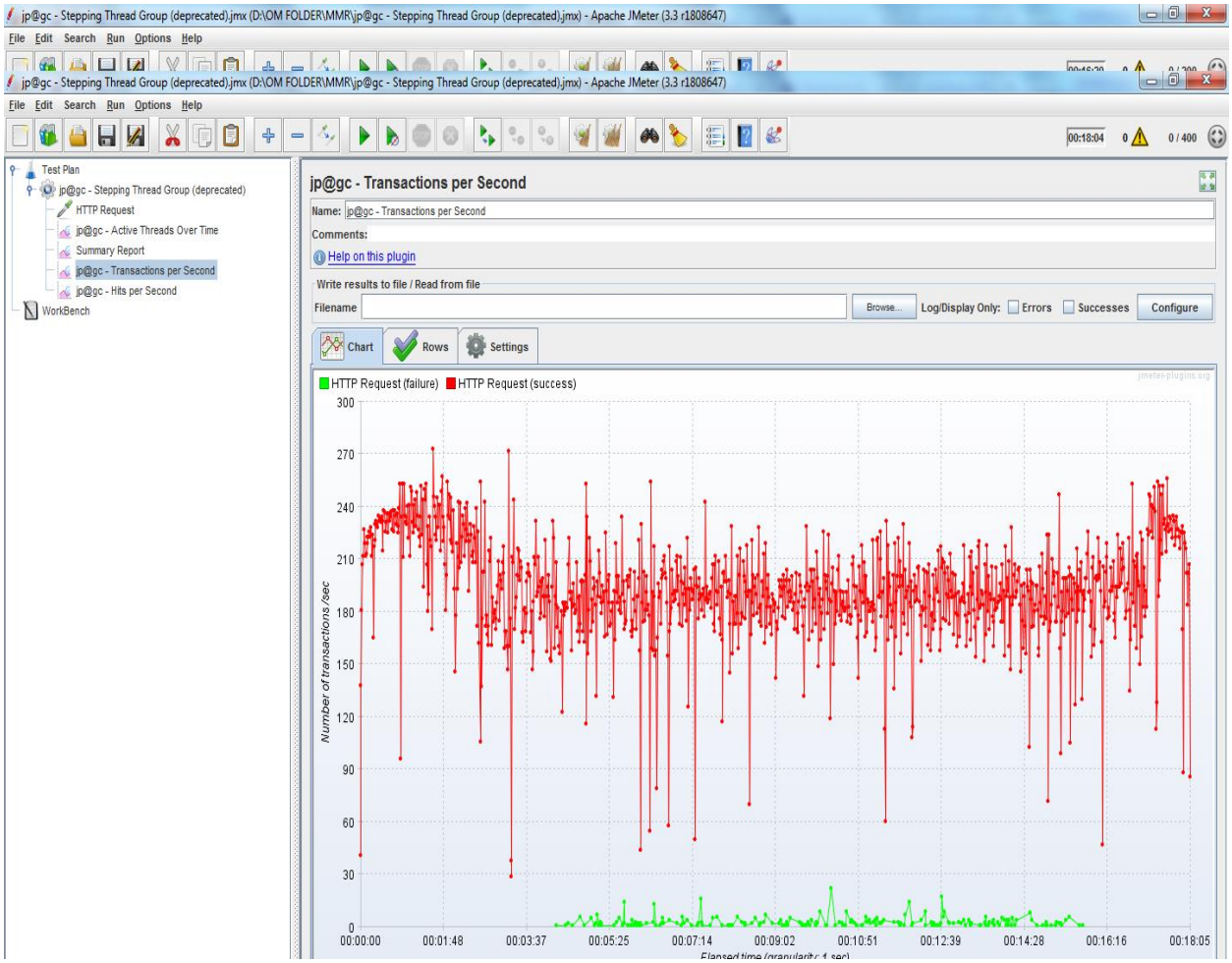
2.1.2 Summary Report:

Home page. Login Screen										
Lable	Sample	Average	Min	Max	Std.Dev	Error%	Throughout	Received KB/sec	Sent KB/sec	Avg. Bytes.
HTTP Request	210842	1599	21	49746	1999.573	0.29%	195/sec	3972.68	58.28	20934.8
TOTAL	210842	1599	21	49746	1999.573	0.29%	195/sec	3972.68	58.28	20934.8

2.1.3 Active Threads over Time:



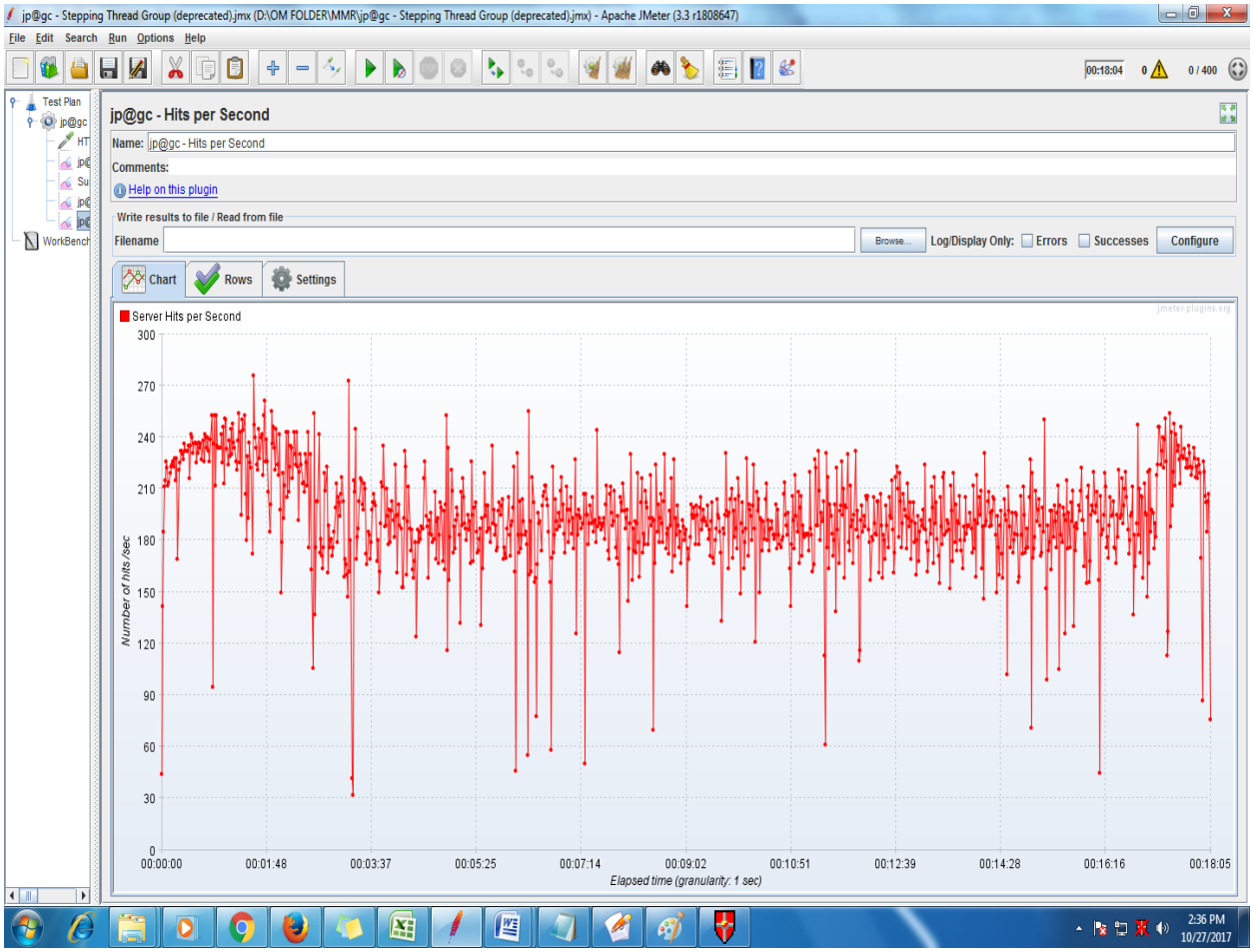
2.1.4 Transactions per second:



***Green color HTTP Request Failure

***Red Color Http Request Success

2.1.5 Hits per second:



3. Let's implement another scenario: With 200 User.

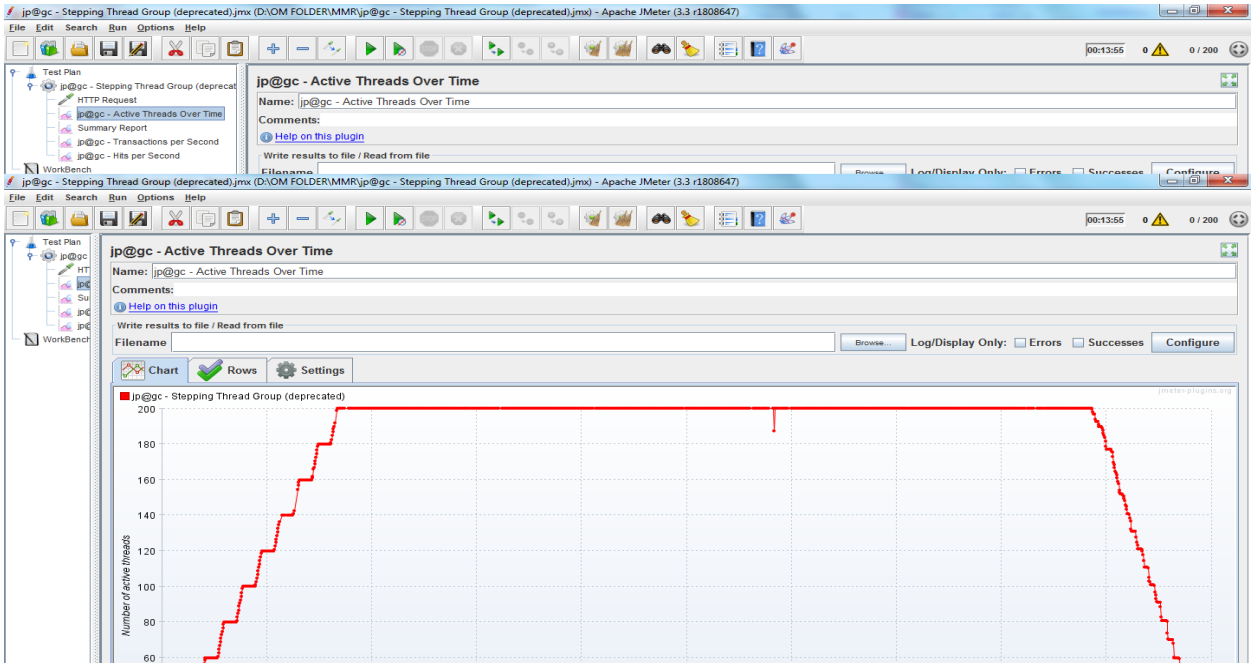
3.1 For Contact List page:

- 200 threads as target load
- 0 seconds waiting after the test starts
- 0 threads run at the immediate beginning of the test
- 20 threads are added every 10 seconds with a ramp-up (or step transition time) of 5 seconds
- The target load is held for 600 seconds (10 minute)
- Finally, 10 threads are stopped every 5 seconds.

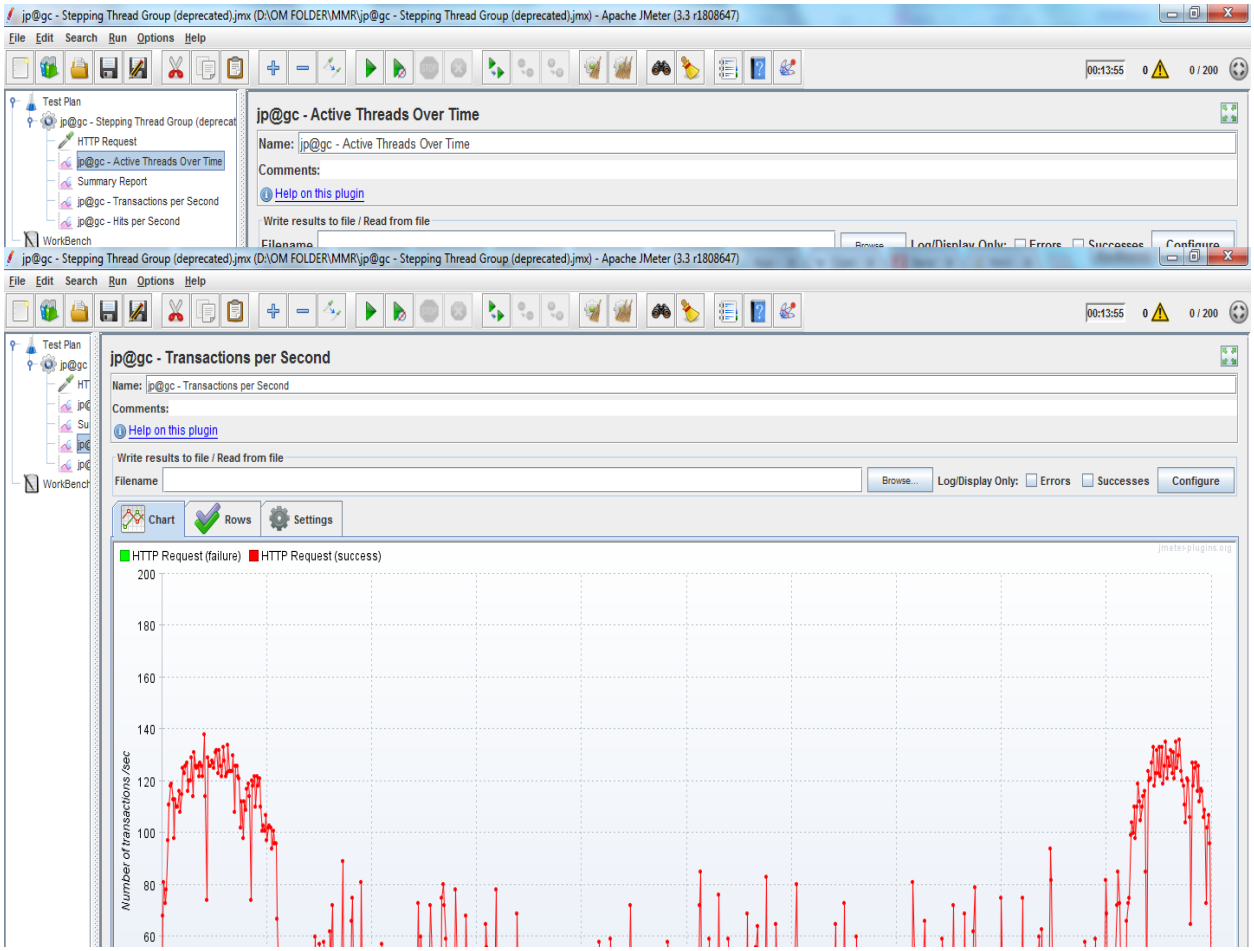
Summary Report:

Contact List										
Lable	Sample	Average	Min	Max	Std.Dev	Error%	Throughout	Received KB/sec	Sent KB/sec	Avg. Bytes.
HTTP Request	35383	4065	32	115292	7009.44	3.31%	42.37343	584.512	18.74	14125.4
TOTAL	35383	4065	32	115292	7009.44	3.31%	42.37343	584.512	18.74	14125.4

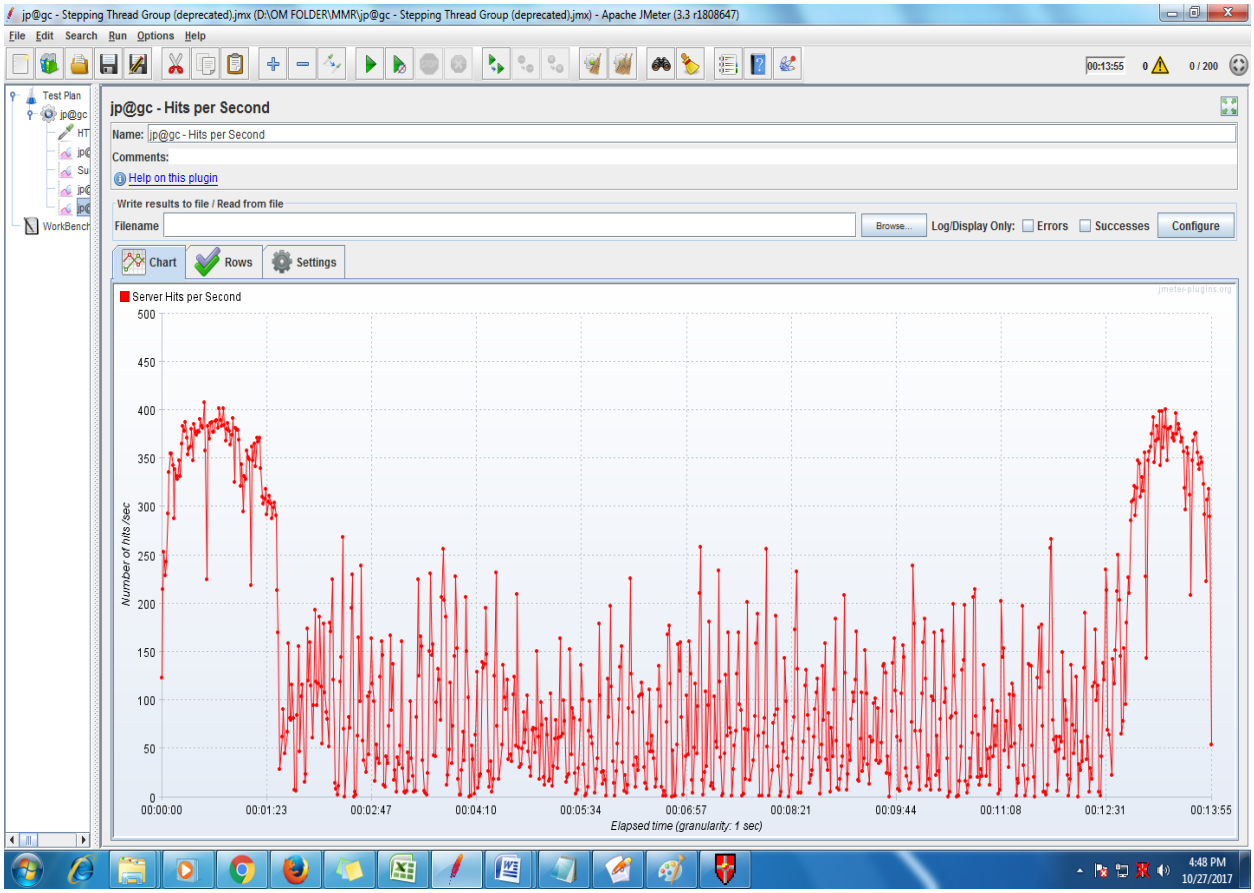
Active threads over time:



Transactions per second:



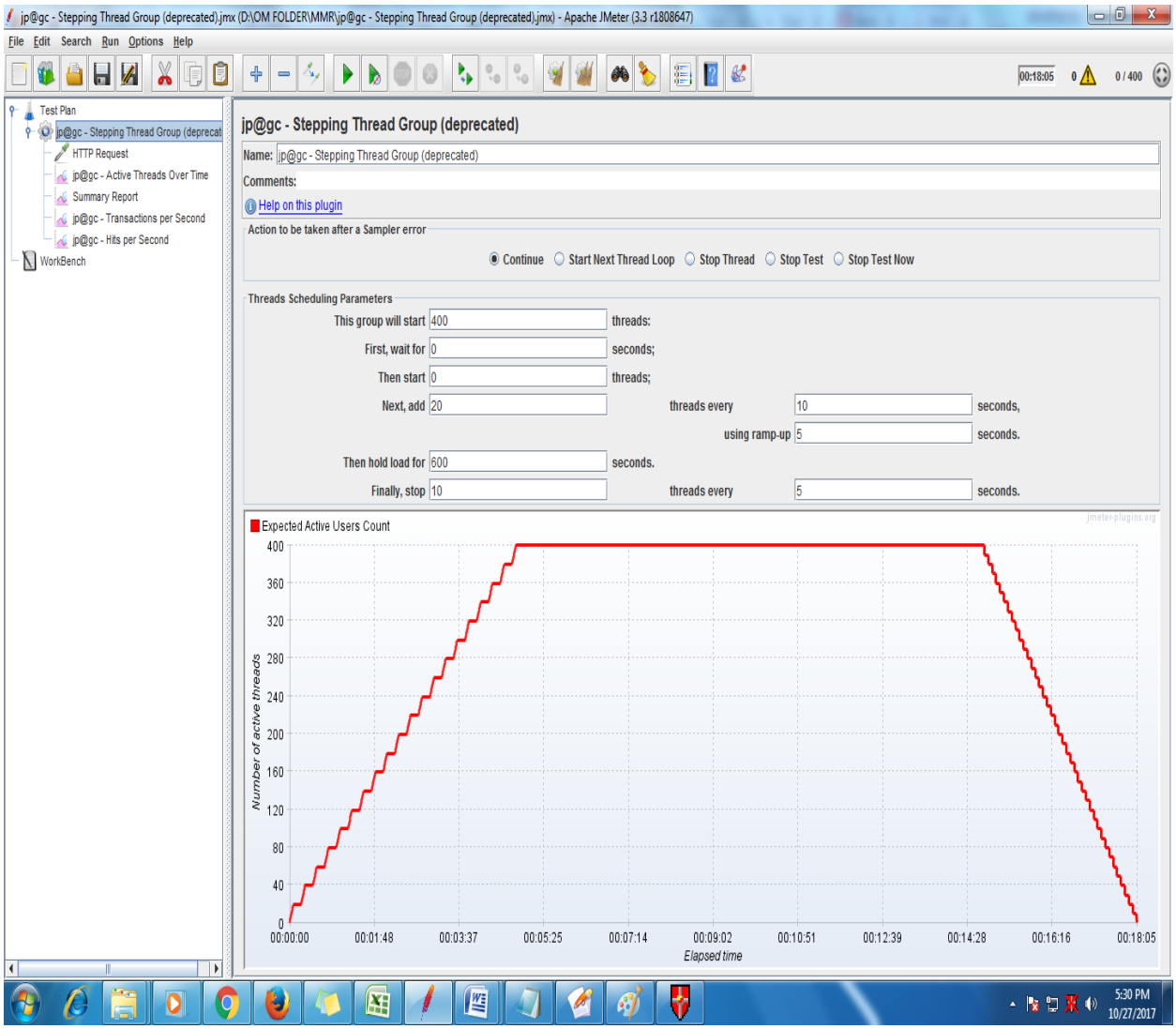
Hits per second:



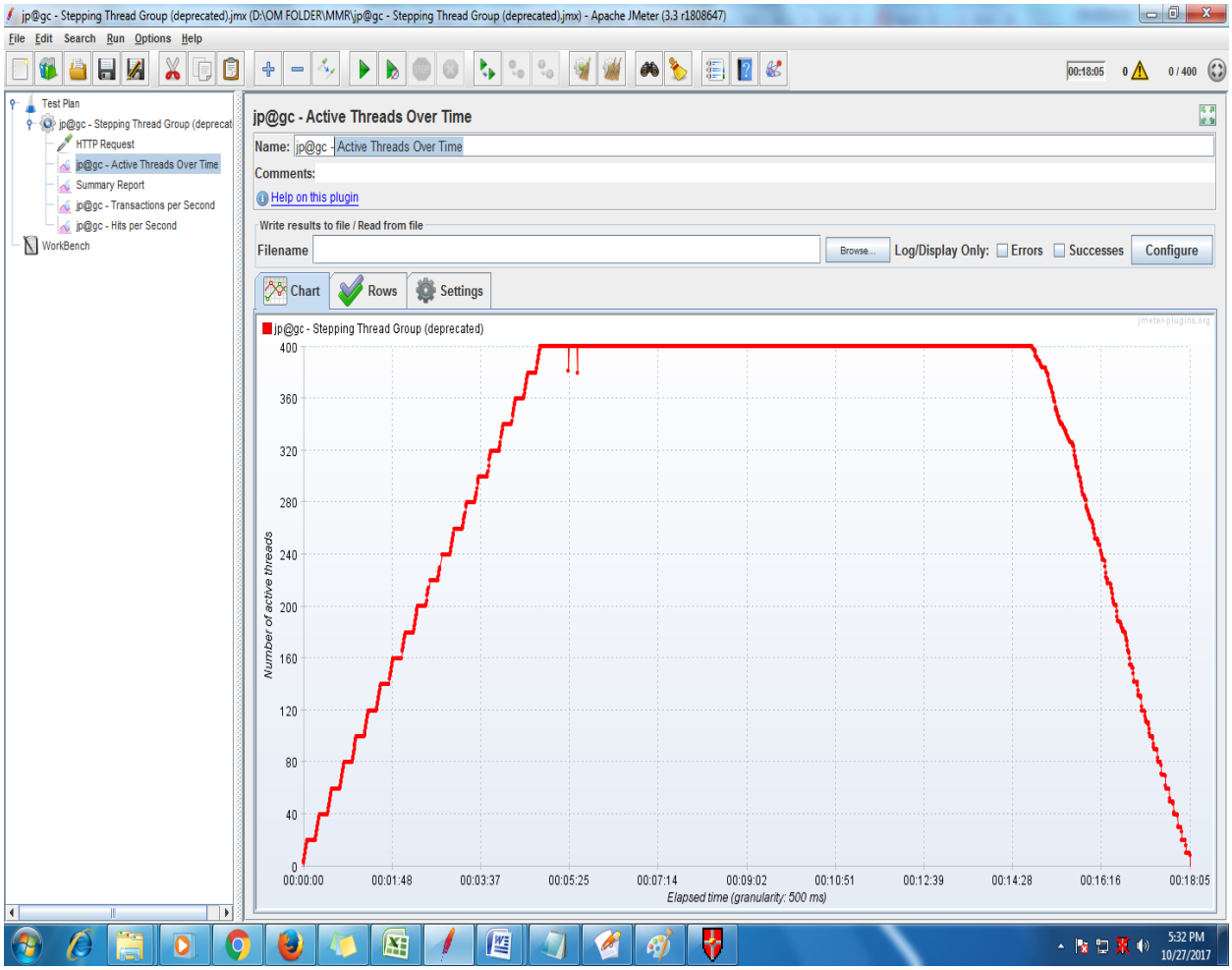
4. Let's implement another scenario: With 400 User.

4.1 For Contact List page:

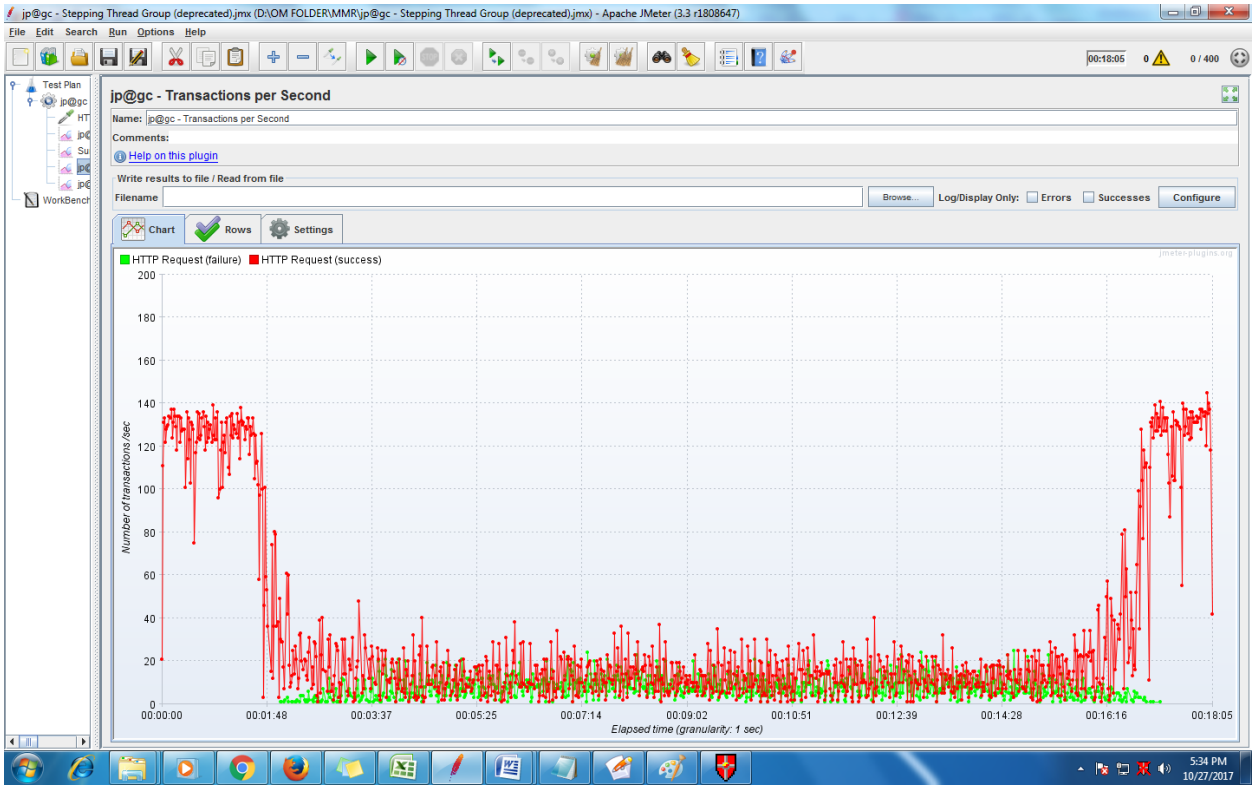
- 400 threads as target load
- 0 seconds waiting after the test starts
- 0 threads run at the immediate beginning of the test
- 20 threads are added every 10 seconds with a ramp-up (or step transition time) of 5 seconds
- The target load is held for 600 seconds (10 minute)
- Finally, 10 threads are stopped every 5 seconds.



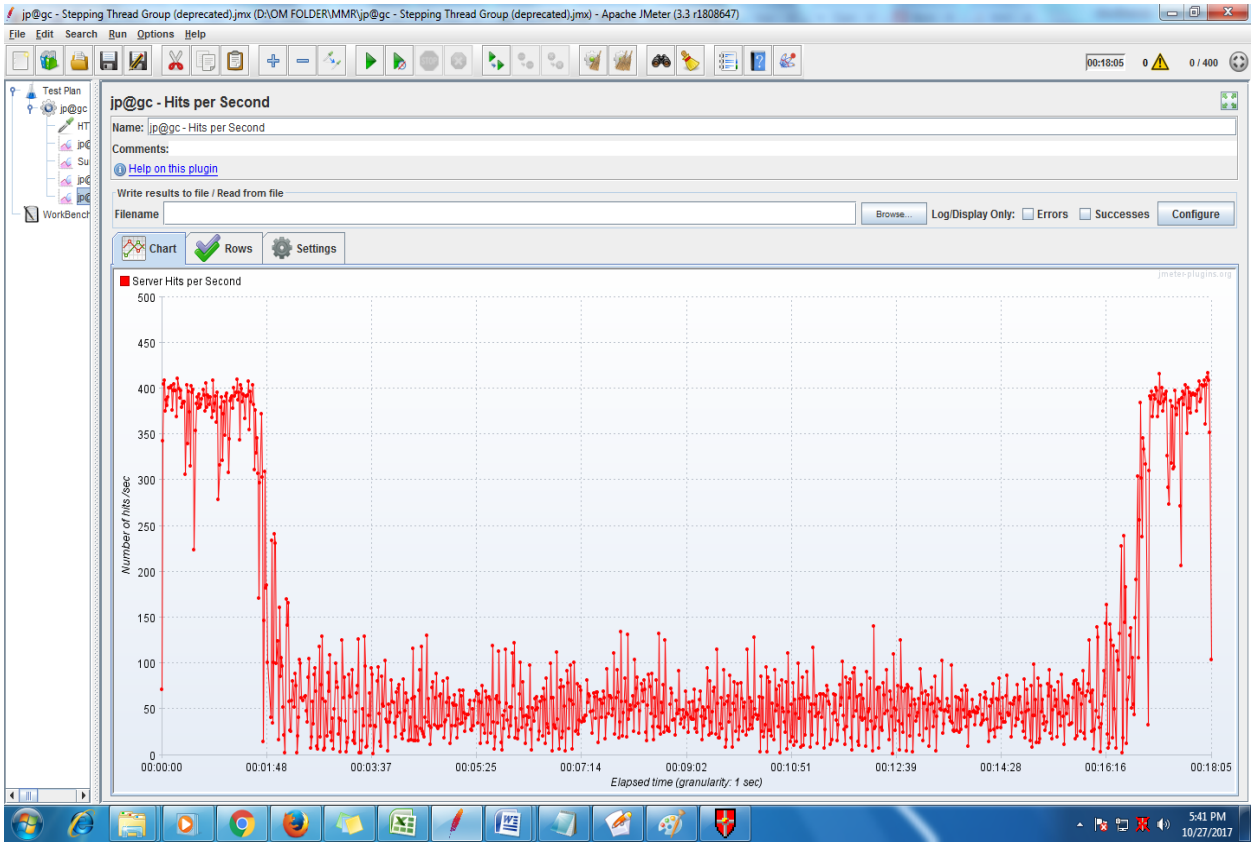
Active Threads over Time:



Transactions Per second:



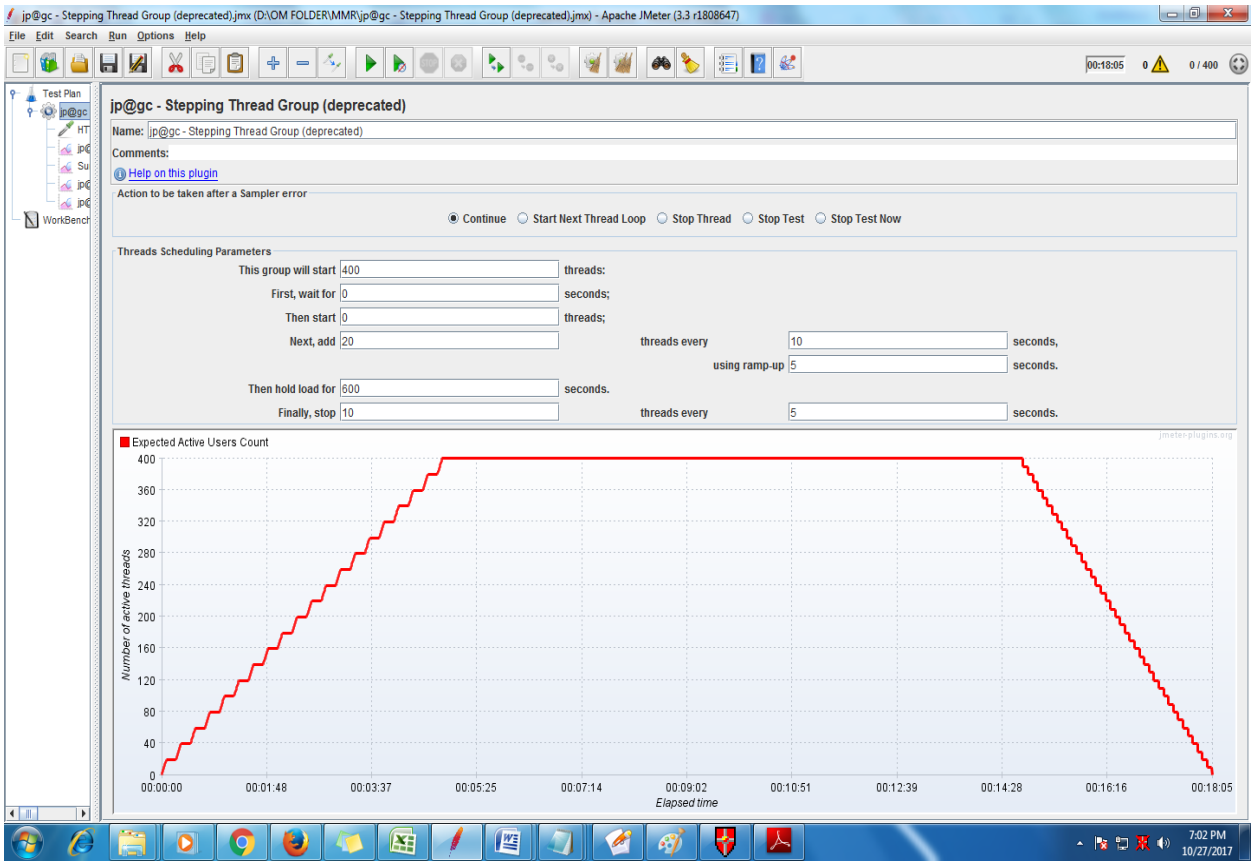
Hits per second:



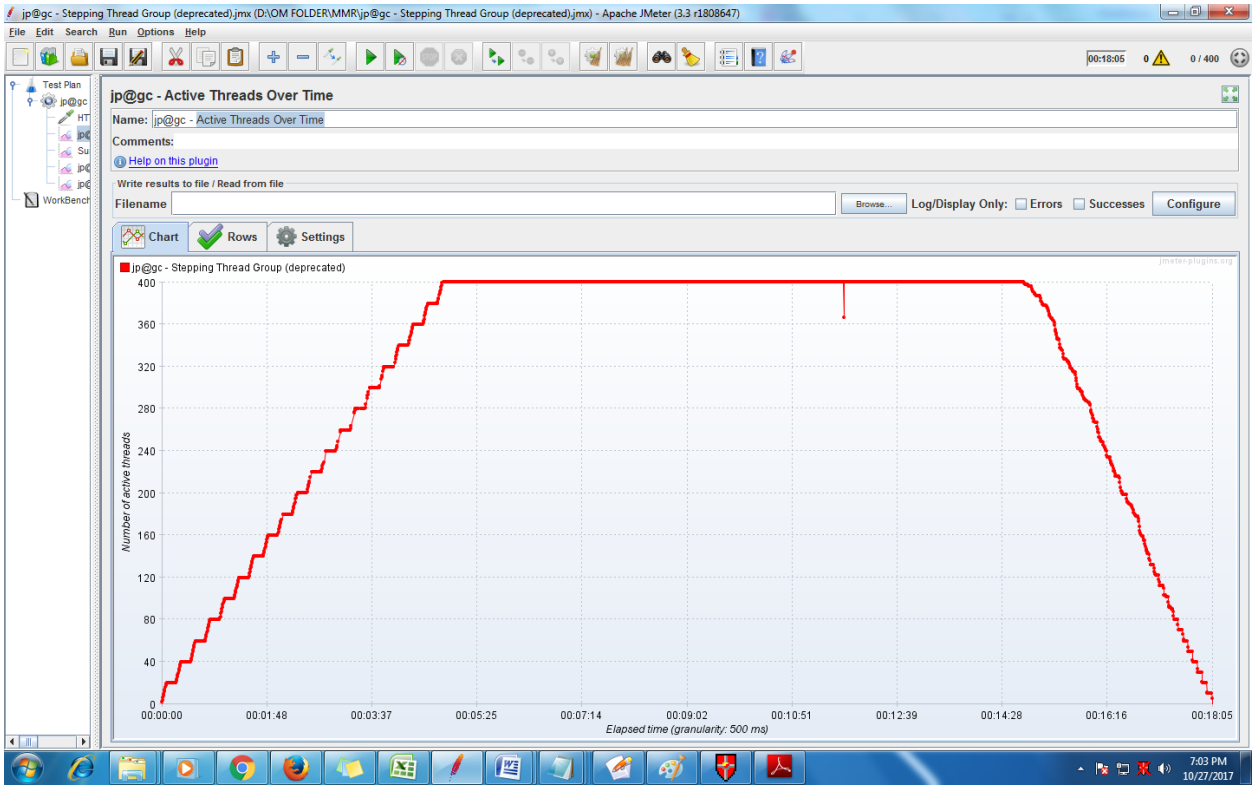
5: Let's implement another scenario: With 400 User.

Design your report and assign to contacts page

- 400 threads as target load
- 0 seconds waiting after the test starts
- 0 threads run at the immediate beginning of the test
- 20 threads are added every 10 seconds with a ramp-up (or step transition time) of 5 seconds
- The target load is held for 600 seconds (10 minute)
- Finally, 10 threads are stopped every 5 seconds.



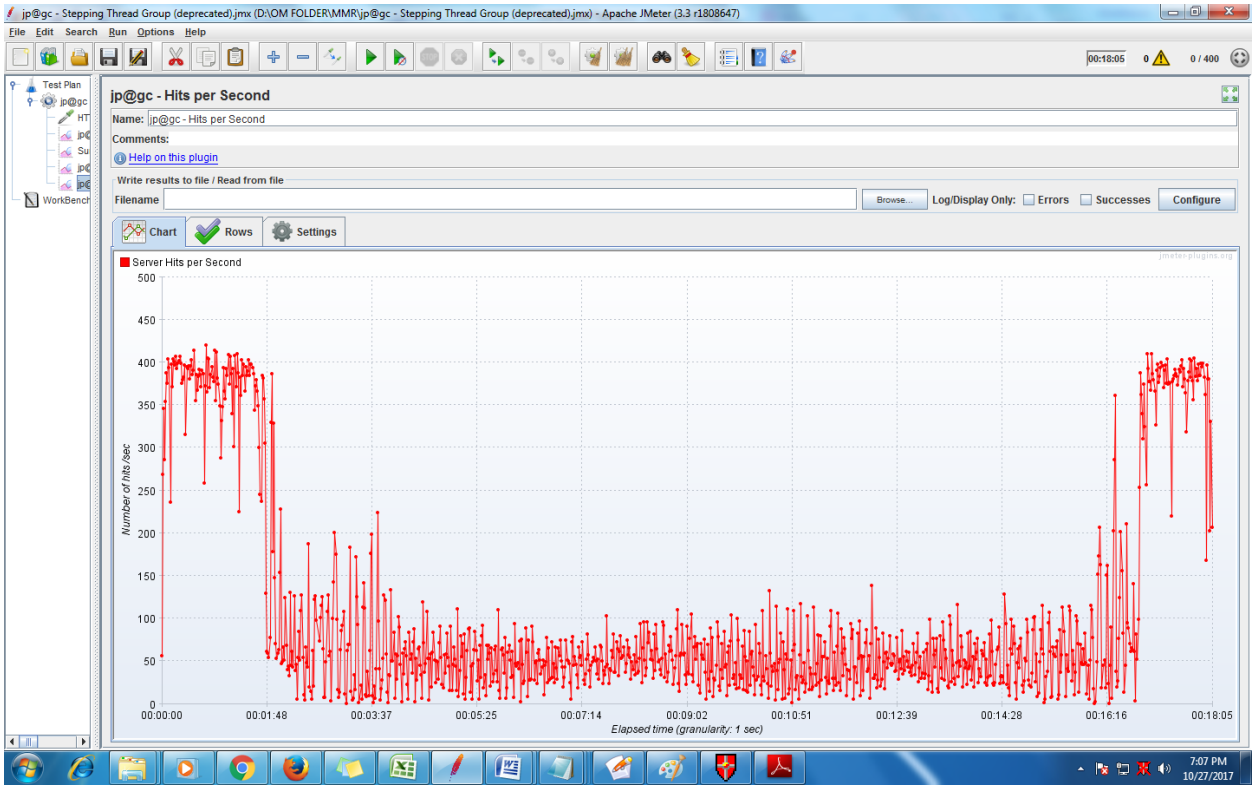
Active Threads over Time:



Summary Report:

Design your report and assign to contacts										
Lable	Sample	Average	Min	Max	Std.Dev	Error%	Throughout	Received KB/sec	Sent KB/sec	Avg. Bytes.
HTTP Request	42356	8018	30	144542	12094.25	16.06%	39.0402	479.87	15.14	12586.7
TOTAL	42356	8018	30	144542	12094.25	16.06%	39.0402	479.87	15.14	12586.7

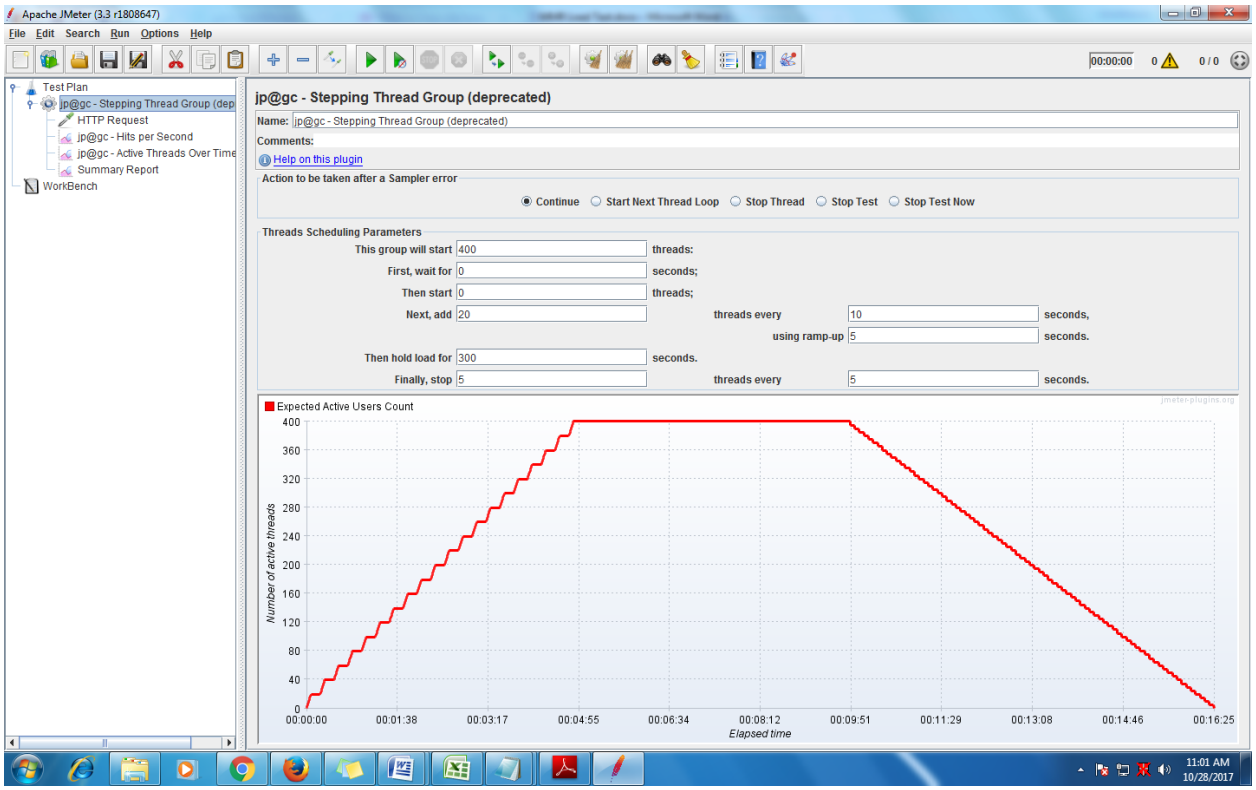
Hits per second:



6. Let's implement another scenario: With 400 User.

6.1 View reply on assigned reports:

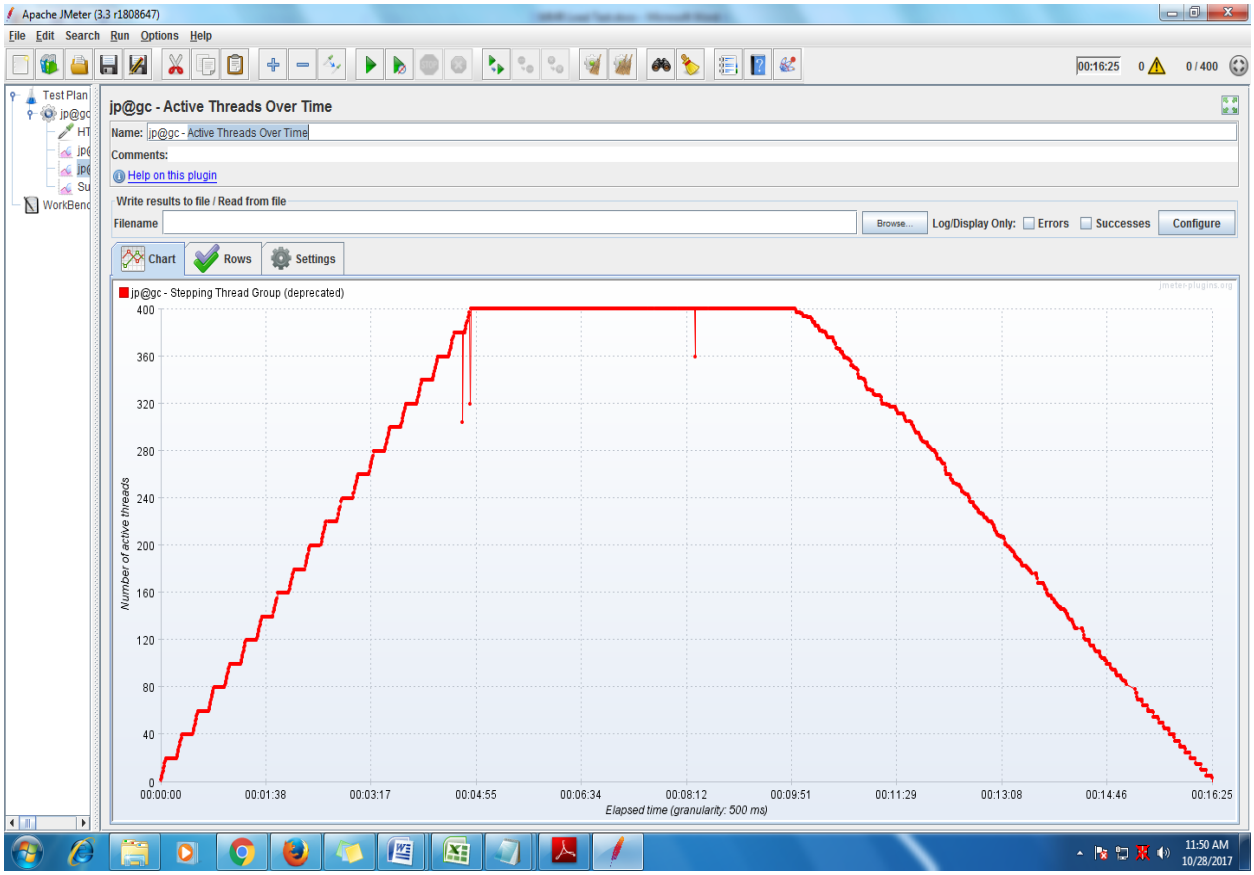
- 400 threads as target load
- 0 seconds waiting after the test starts
- 0 threads run at the immediate beginning of the test
- 20 threads are added every 10 seconds with a ramp-up (or step transition time) of 5 seconds
- The target load is held for 300 seconds (5 minute)
- Finally, 5 threads are stopped every 5 seconds.



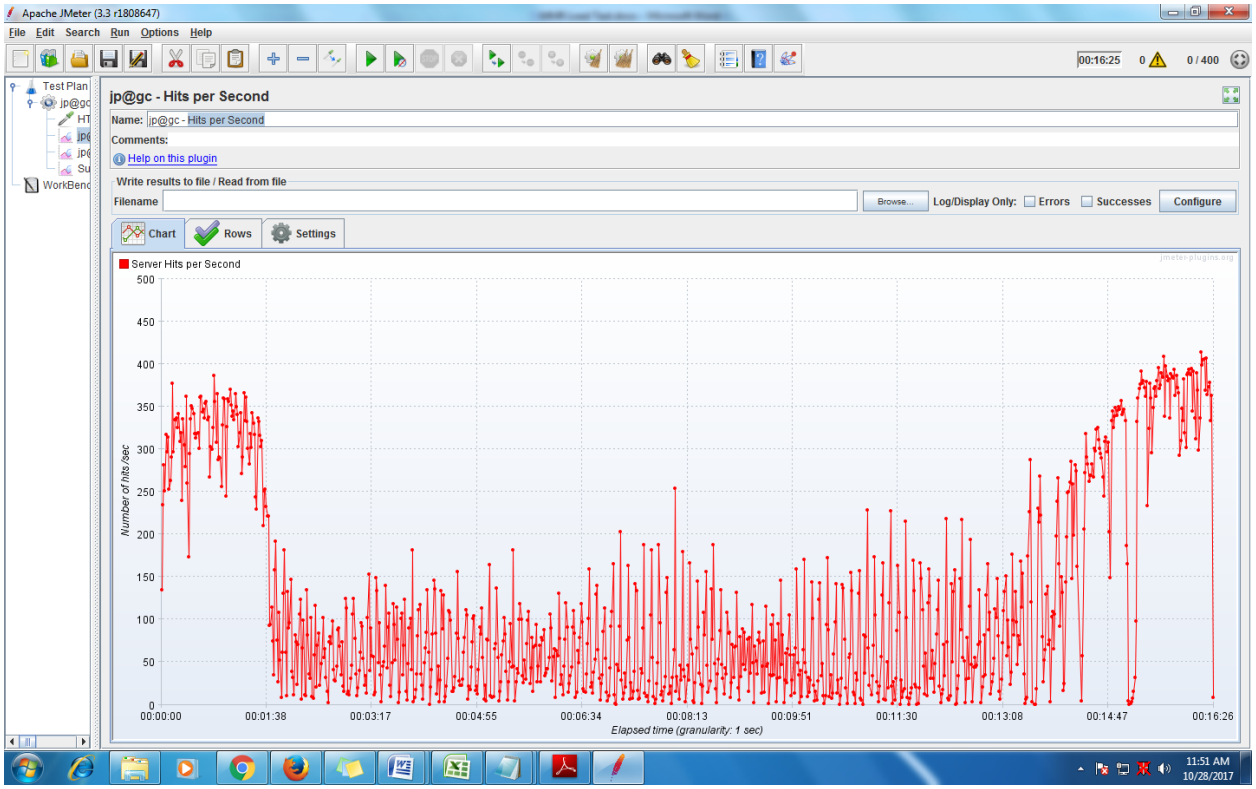
Summary Report:

View reply on assigned reports										
Lable	Sample	Average	Min	Max	Std.Dev	Error%	Throughout	Received KB/sec	Sent KB/sec	Avg. Bytes.
HTTP Request	42407	6130	0	132356	10683.06	12.39%	43.1/sec	547.705	17.39	13026.1
TOTAL	42407	6130	0	132356	10683.06	12.39%	43.1/sec	547.705	17.39	13026.1

Active Threads over Time:



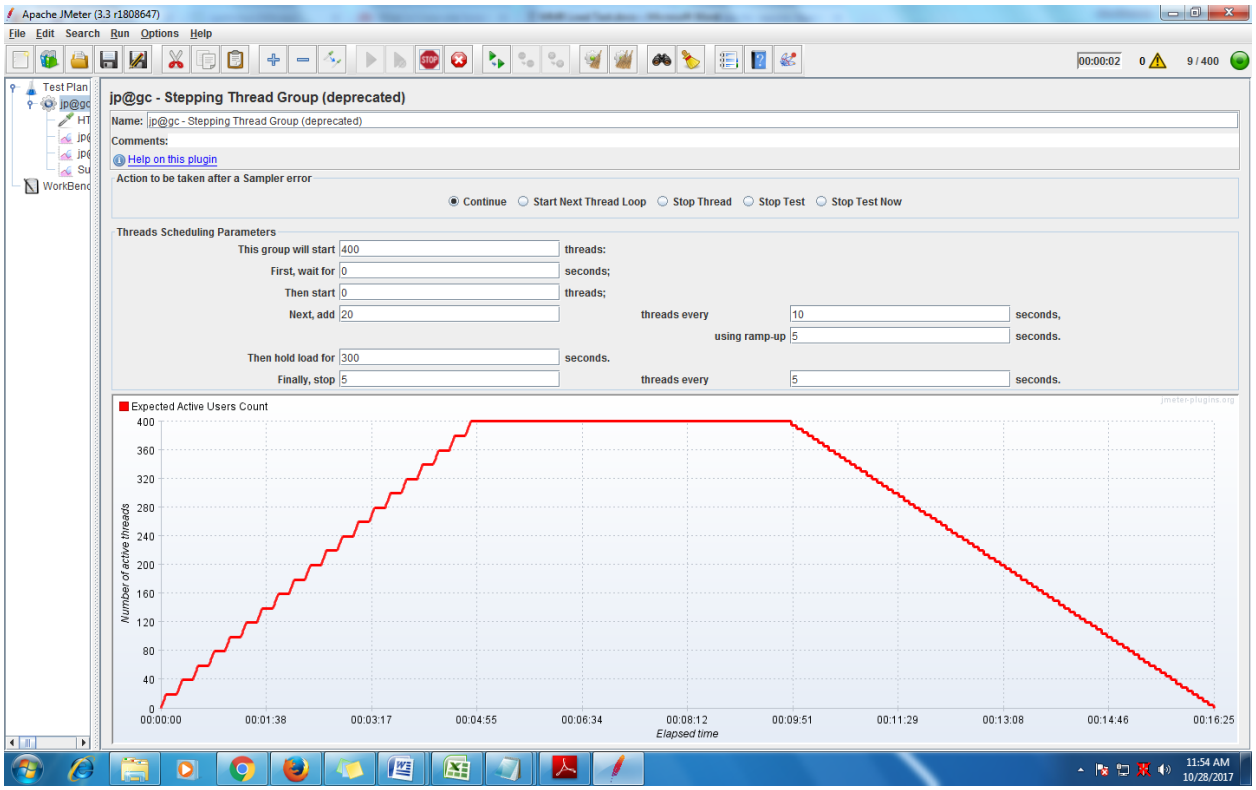
Hits per second:



7. Let's implement another scenario: With 400 User.

7.1 Response report:

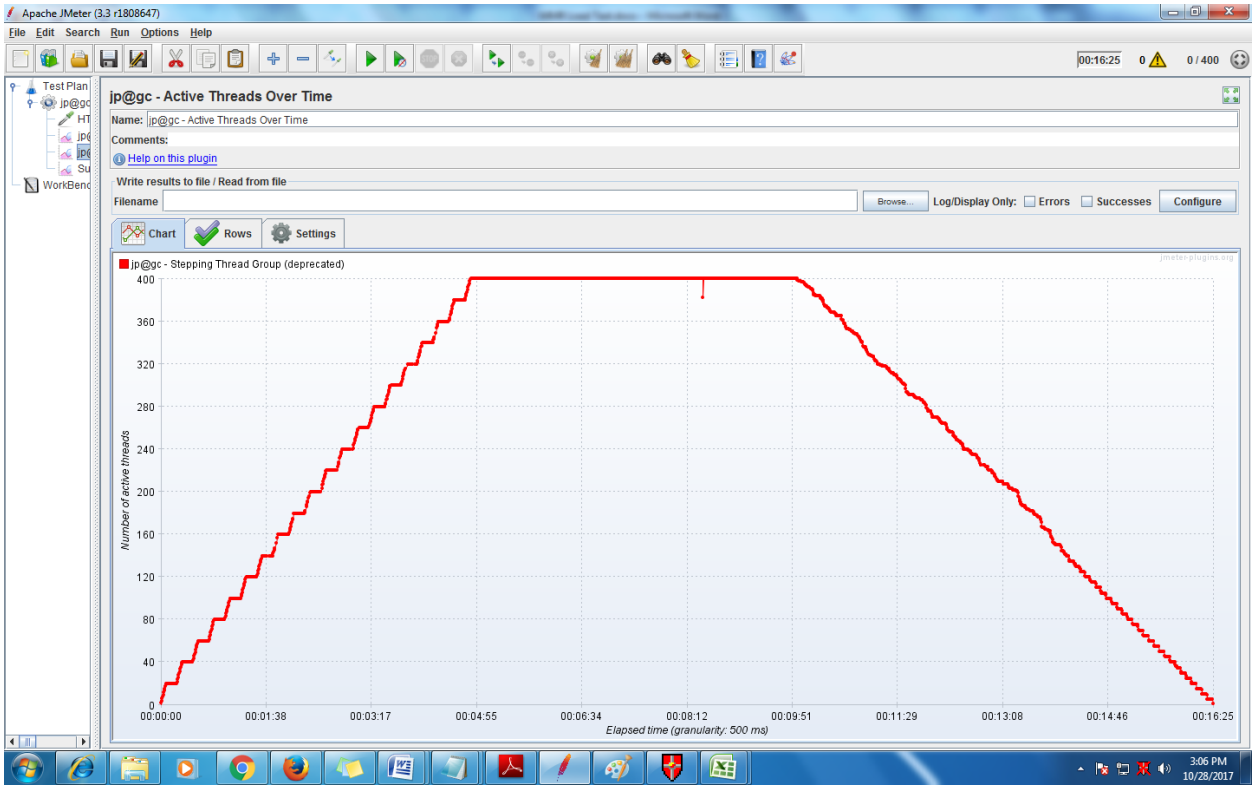
- 400 threads as target load
- 0 seconds waiting after the test starts
- 0 threads run at the immediate beginning of the test
- 20 threads are added every 10 seconds with a ramp-up (or step transition time) of 5 seconds
- The target load is held for 300 seconds (5 minute)
- Finally, 5 threads are stopped every 5 seconds.



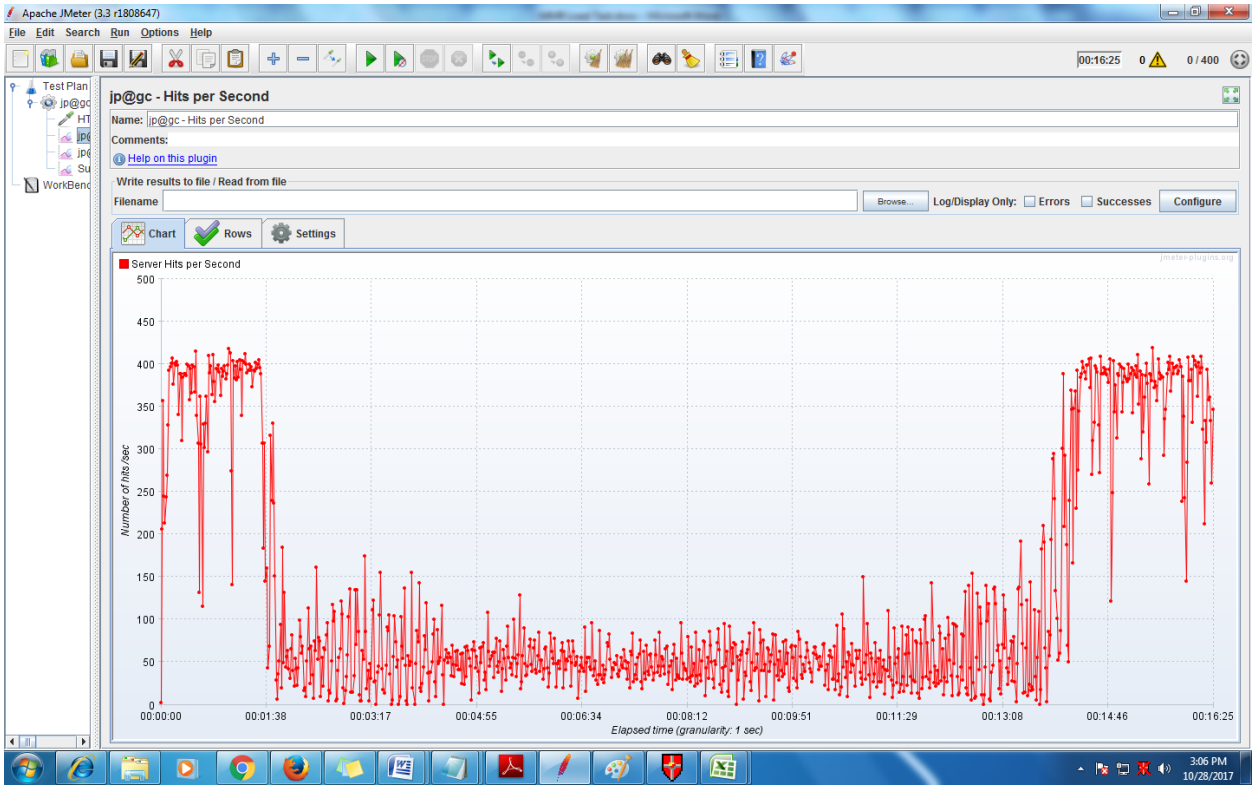
Summary Report:

Response reports										
Lable	Sample	Average	Min	Max	Std.Dev	Error%	Throughout	Received KB/sec	Sent KB/sec	Avg. Bytes.
HTTP Request	44483	5830	31	150369	10410.49	10.53%	45.15	584.41	18.53	13.2
Total	44483	5830	31	150369	10410.49	10.53%	45.15	584.41	18.53	13.2

Active Threads Over time:



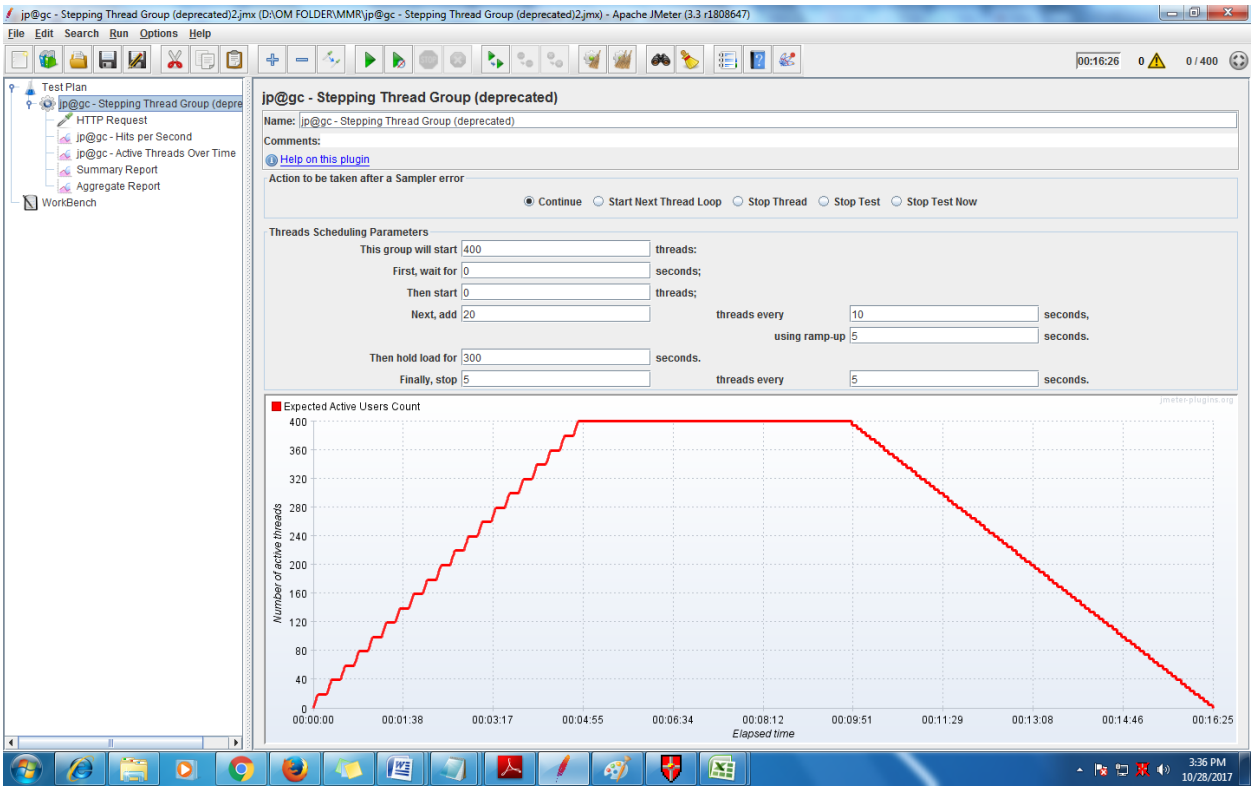
Hits per second:



8. Let's implement another scenario: With 400 User.

8.1 Archive report:

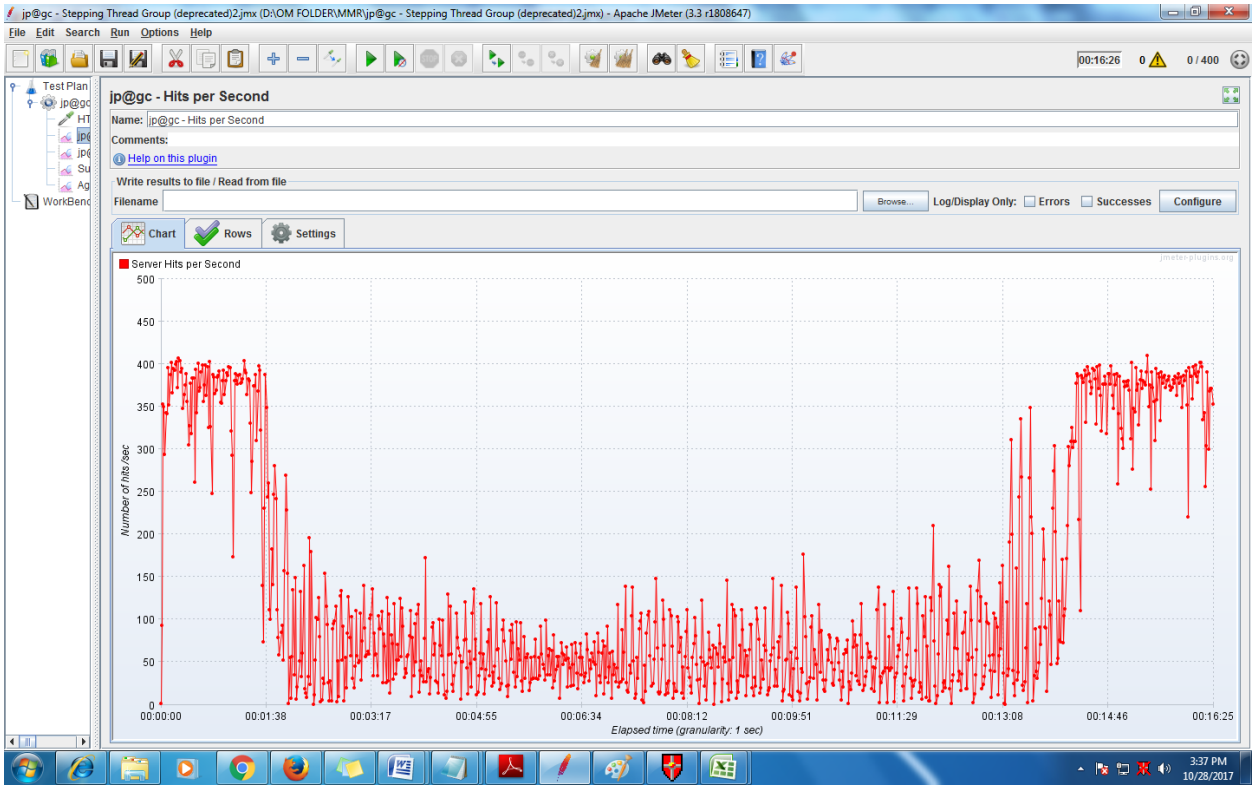
- **400 threads as target load**
- **0 seconds waiting after the test starts**
- **0 threads run at the immediate beginning of the test**
- **20 threads are added every 10 seconds with a ramp-up (or step transition time) of 5 seconds**
- **The target load is held for 300 seconds (5 minute)**
- **Finally, 5 threads are stopped every 5 seconds.**



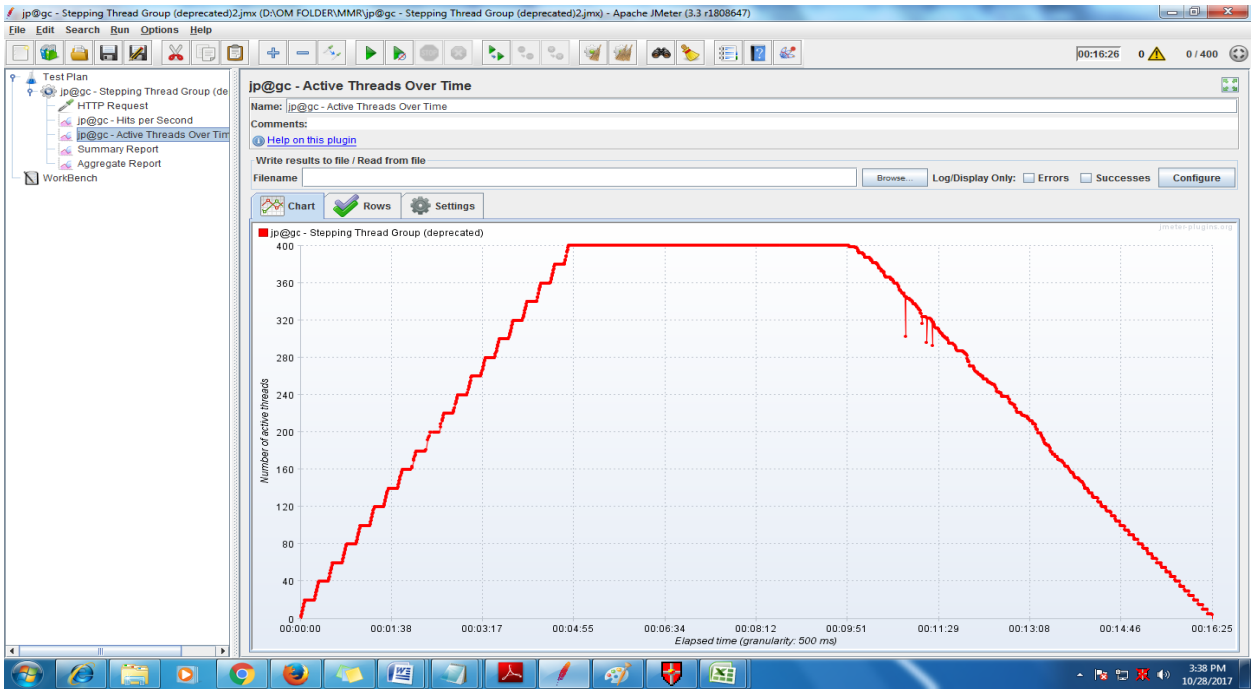
Summery Report:

Archive reports										
Lable	Sample	Average	Min	Max	Std.Dev	Error%	Throughout	Received KB/sec	Sent KB/sec	Avg. Bytes.
HTTP Request	46300	5609	32	150221	10213.08	10.11%	47.00412	610.7	19.36	13304.3
Total	46300	5609	32	150221	10213.08	10.11%	47.00412	610.7	19.36	13304.3

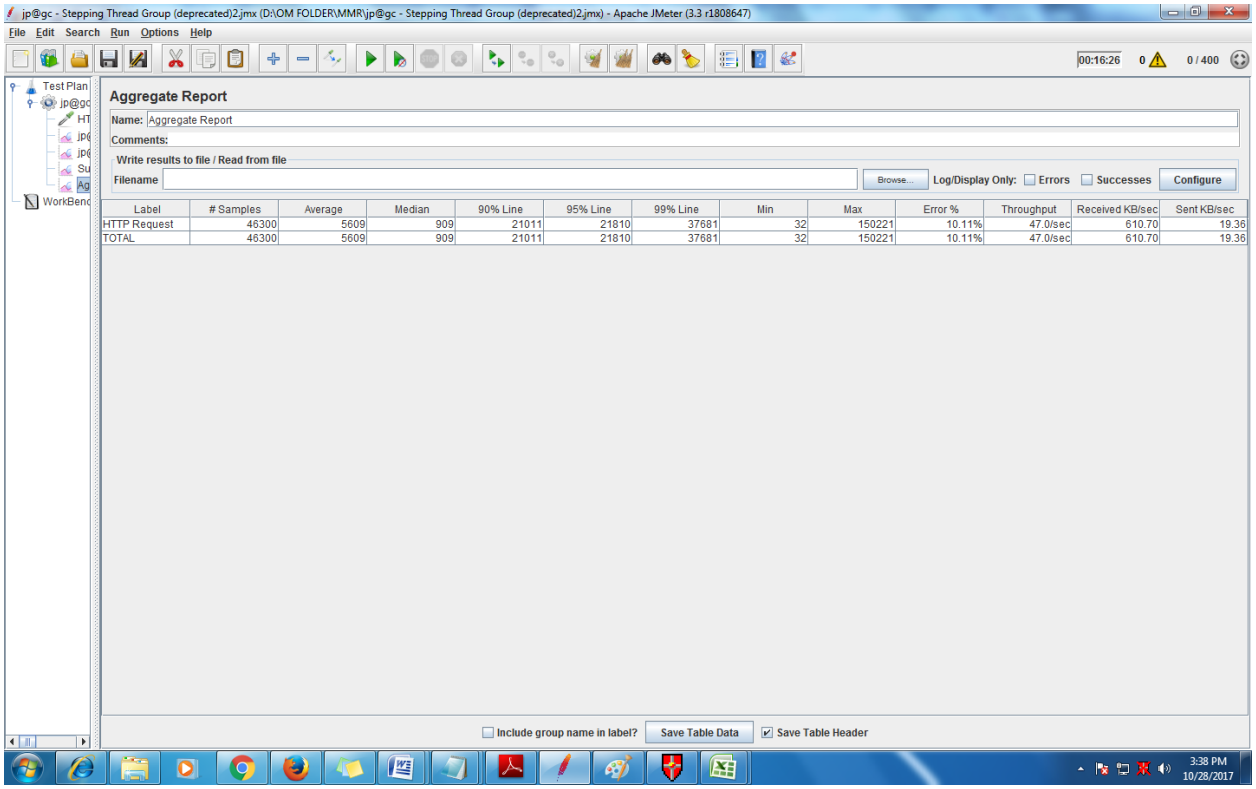
Hits per second:



Active threads Over time:



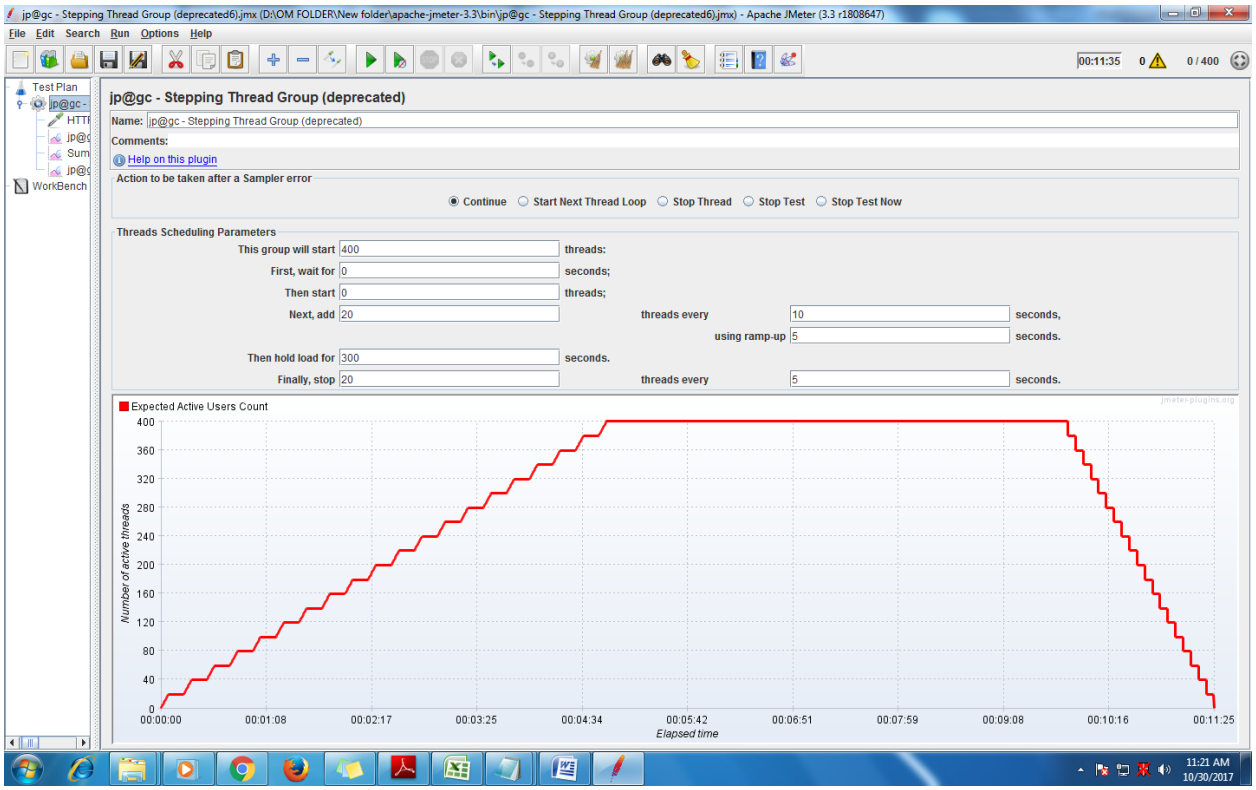
Aggregate report:



9. Let's implement another scenario: With 400 User.

9.1 Submit a report:

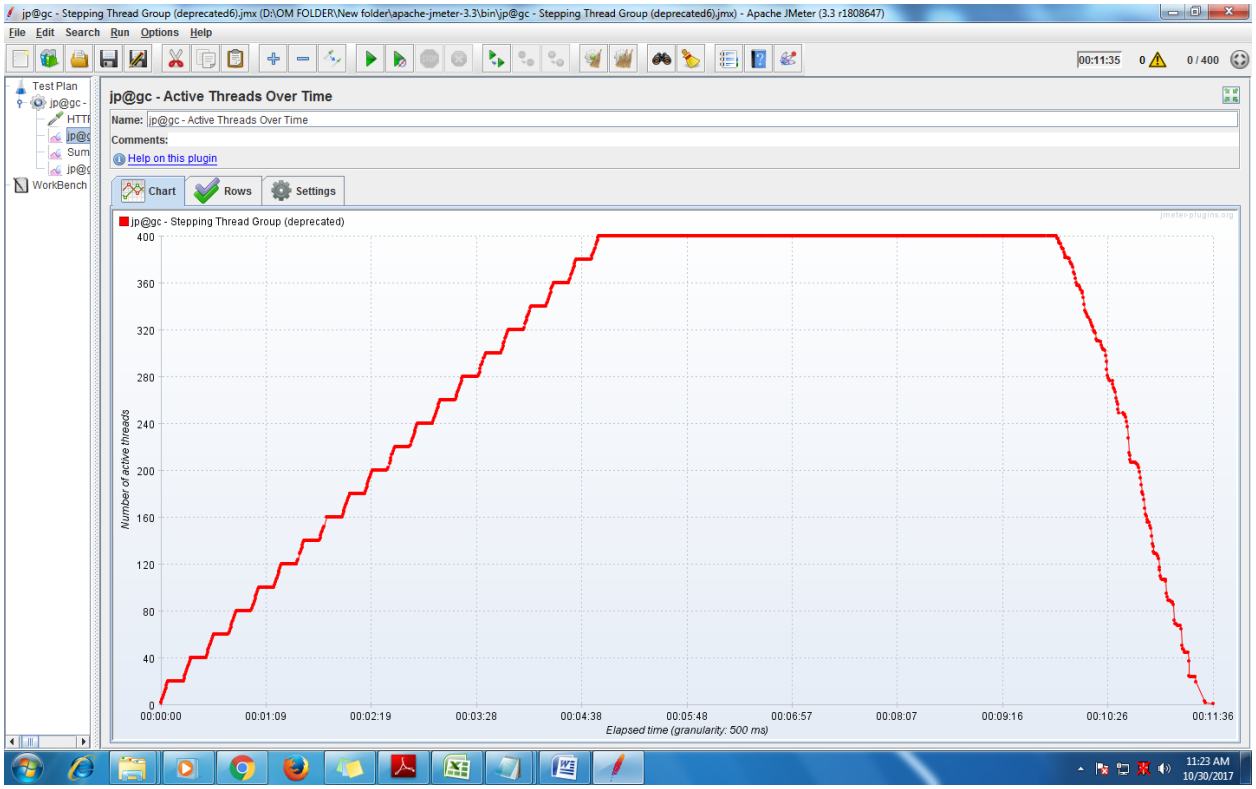
- **400 threads as target load**
- **0 seconds waiting after the test starts**
- **0 threads run at the immediate beginning of the test**
- **20 threads are added every 10 seconds with a ramp-up (or step transition time) of 5 seconds**
- **The target load is held for 300 seconds (5 minute)**
- **Finally, 20 threads are stopped every 5 seconds.**



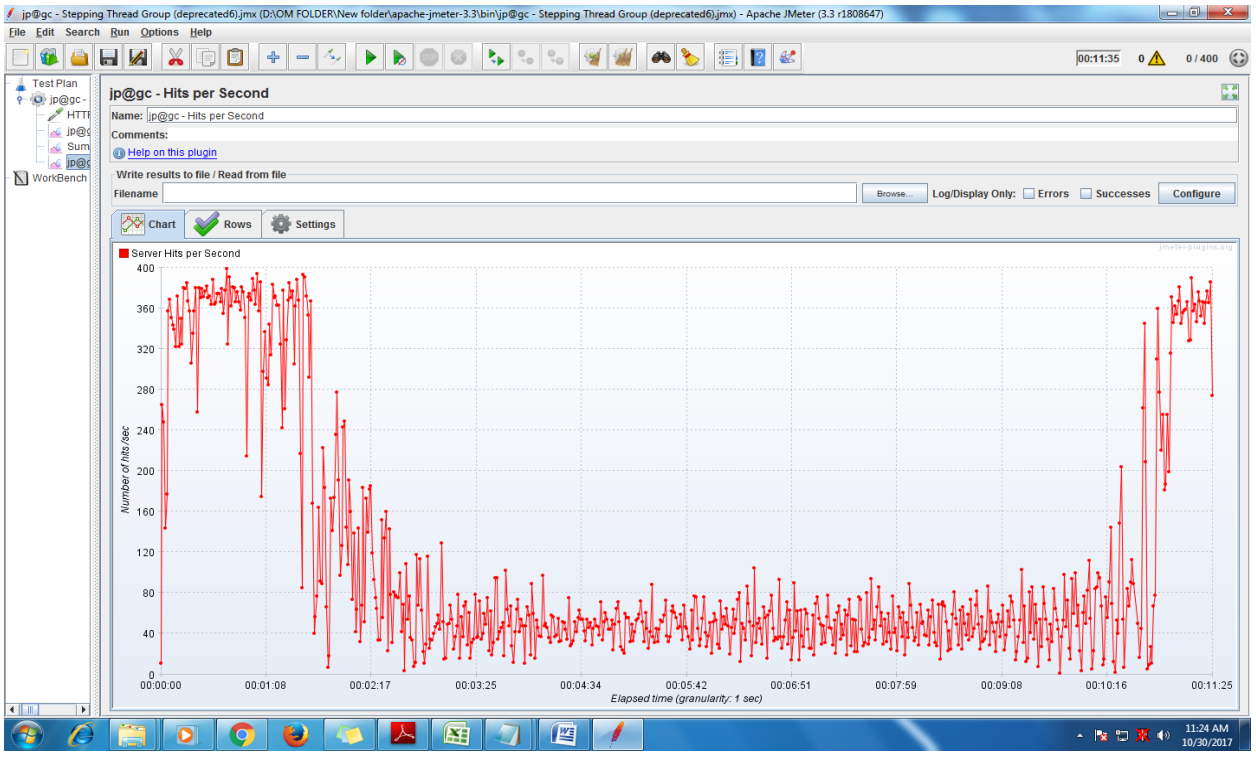
Summary Report:

Submit a report										
Lable	Sample	Average	Min	Max	Std.Dev	Error%	Throughout	Received KB/sec	Sent KB/sec	Avg. Bytes.
HTTP Request	27728	7198	33	126269	11499.23	13.67%	39.8/sec	500.827	16.02	12871.6
Total	27728	7198	33	126269	11499.23	13.67%	39.8/sec	500.827	16.02	12871.6

Active threads over time:



Hits per second:



10. Let's implement one scenario: With 1000 User.

10.1 For Login Page:

Let's implement at the following scenario:

- **1,000 threads as target load**
- **0 seconds waiting after the test starts**
- **0 threads run at the immediate beginning of the test**
- **100 threads are added every 30 seconds with a ramp-up (or step transition time) of 5 seconds**
- **The target load is held for 60 seconds (1 minute)**
- **Finally, 5 threads are stopped every 3 seconds.**

This means that:

- The test begins immediately when JMeter starts, since there is 0 seconds waiting.

- Every 30 seconds 100 users will be added, until we reach 1,000 users. The first step is 1-100, the second 101-200, etc., because we defined 0 threads to run at the beginning.

- It will take each of the steps (with 100 users each) 10 seconds to complete. After that JMeter waits 30 seconds before starting the next step.

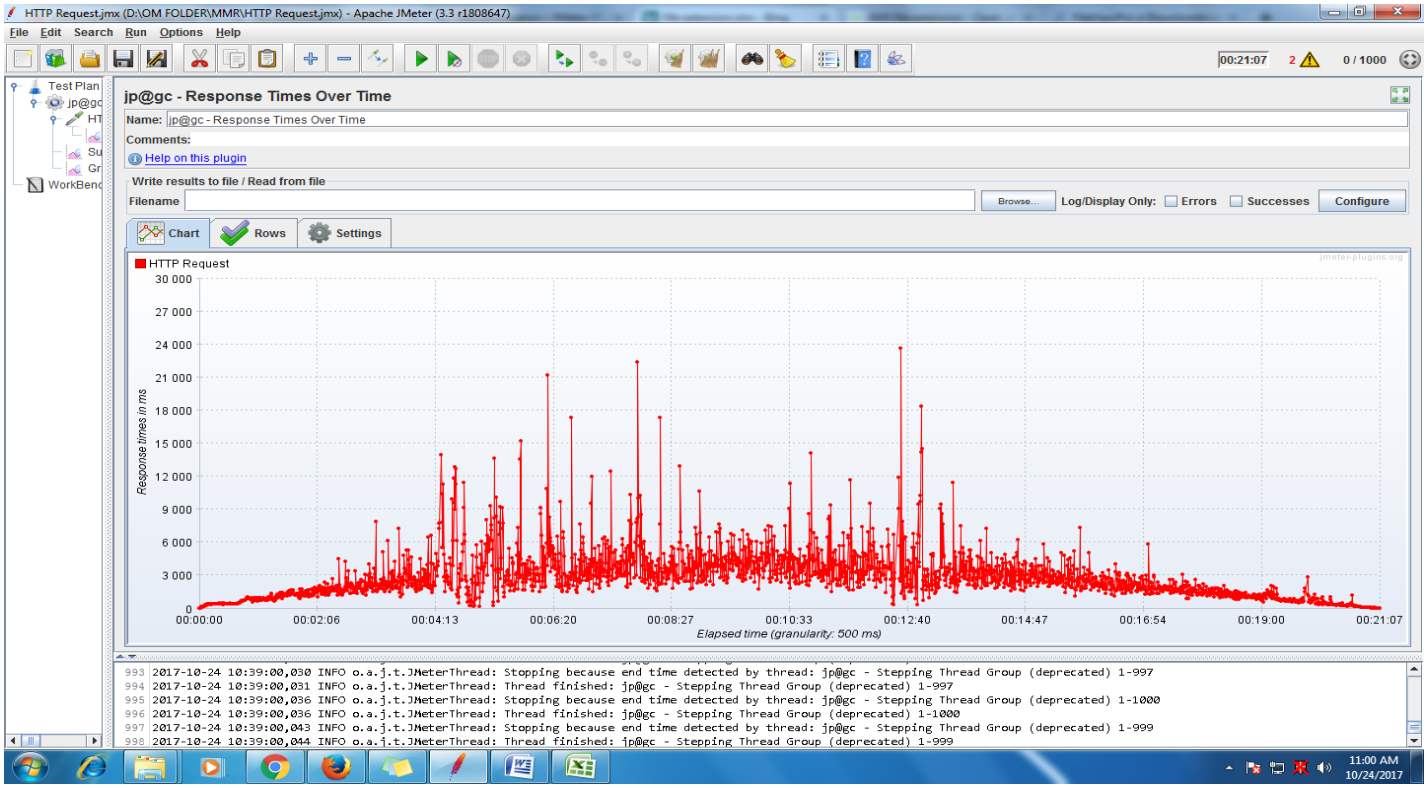
- After reaching 1,000 threads all of them will continue running and hitting the server together for 5 minutes.

- At the end, 5 threads will stop every 3 seconds.

The Stepping Thread Group shows us the test in a real-time preview graph.

10.2 Home Page/Login screen Result:

Response Times over Time:

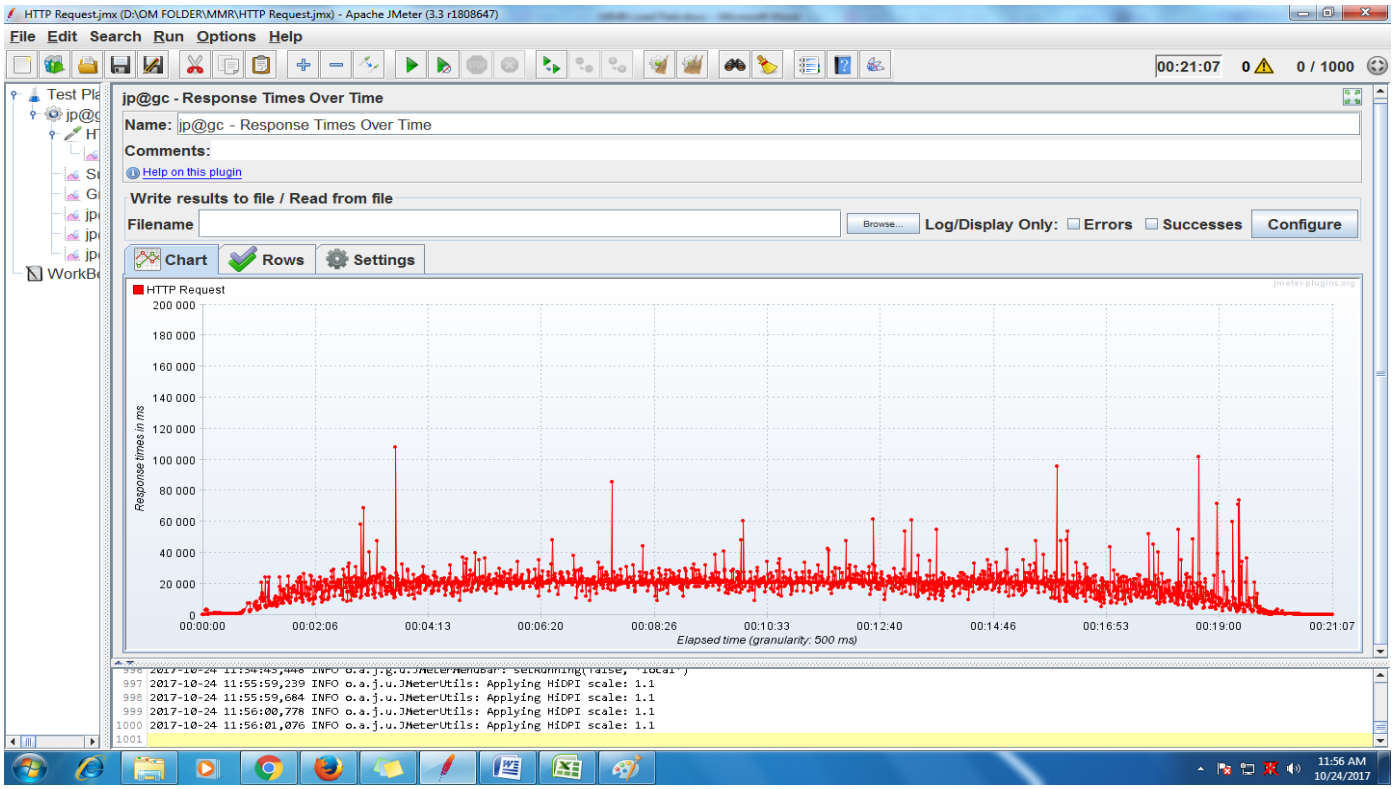


3: Summery Report:

Home Page. Login Scren										
Lable	Sample	Average	Min	Max	Std.Dev	Error%	Throughout	Received KB/sec	Sent KB/sec	Avg. Bytes.
HTTP Request	271299	2891	19	96863	5173.57	4.04%	214.1/sec	4230.02	37.93	20228.1
Total	271299	2891	19	96863	5173.57	4.04%	214.1/sec	4230.02	37.93	20228.1

4 Create contact page Result:

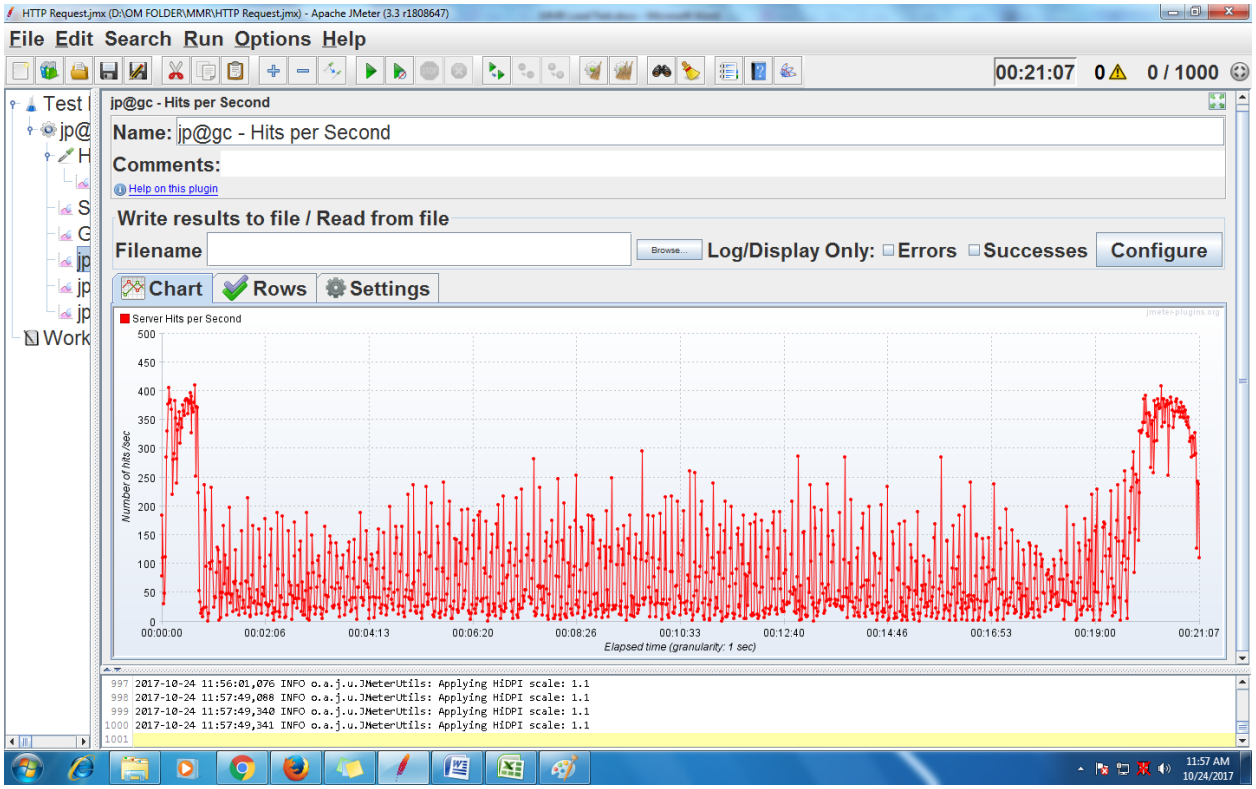
1: Create Contact Page: makemyreport.com/Users/manualRegister



5: Summery report.

Create contact page										
Lable	Sample	Average	Min	Max	Std.Dev	Error%	Throughout	Received KB/sec	Sent KB/sec	Avg. Bytes.
HTTP Request	53791	14765	31	127014	12230.42	49.45%	42.5/sec	353.19	11.11	8518.9
Total	53791	14765	31	127014	12230.42	49.45%	42.5/sec	353.19	11.11	8518.9

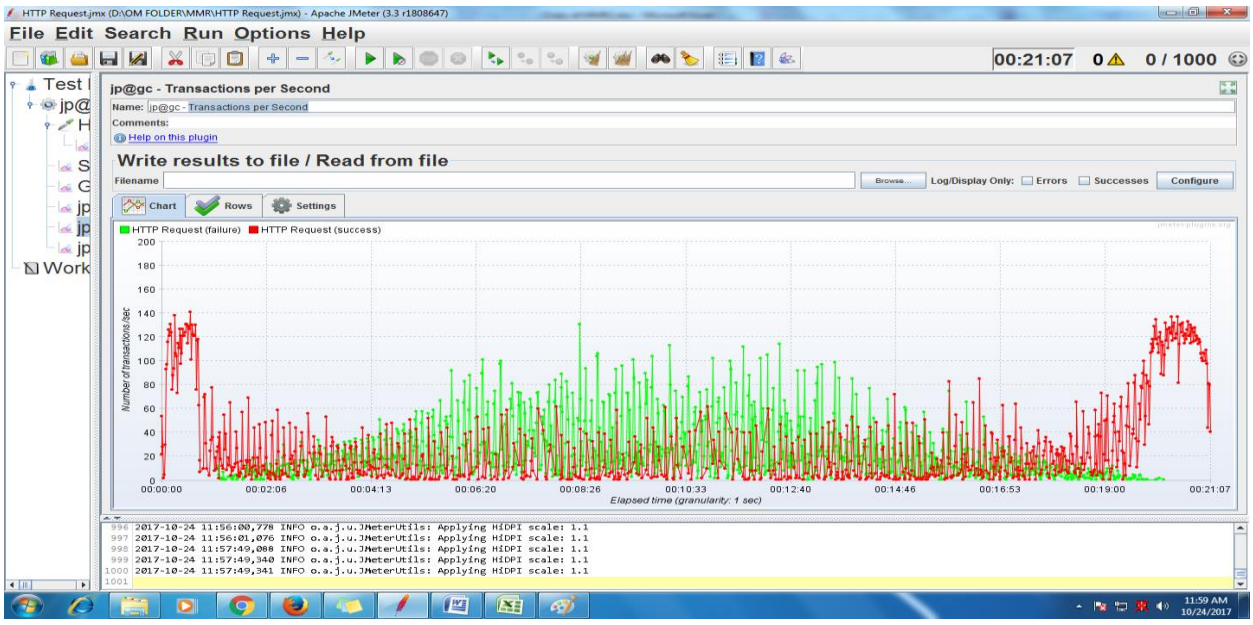
6: Hits per second:



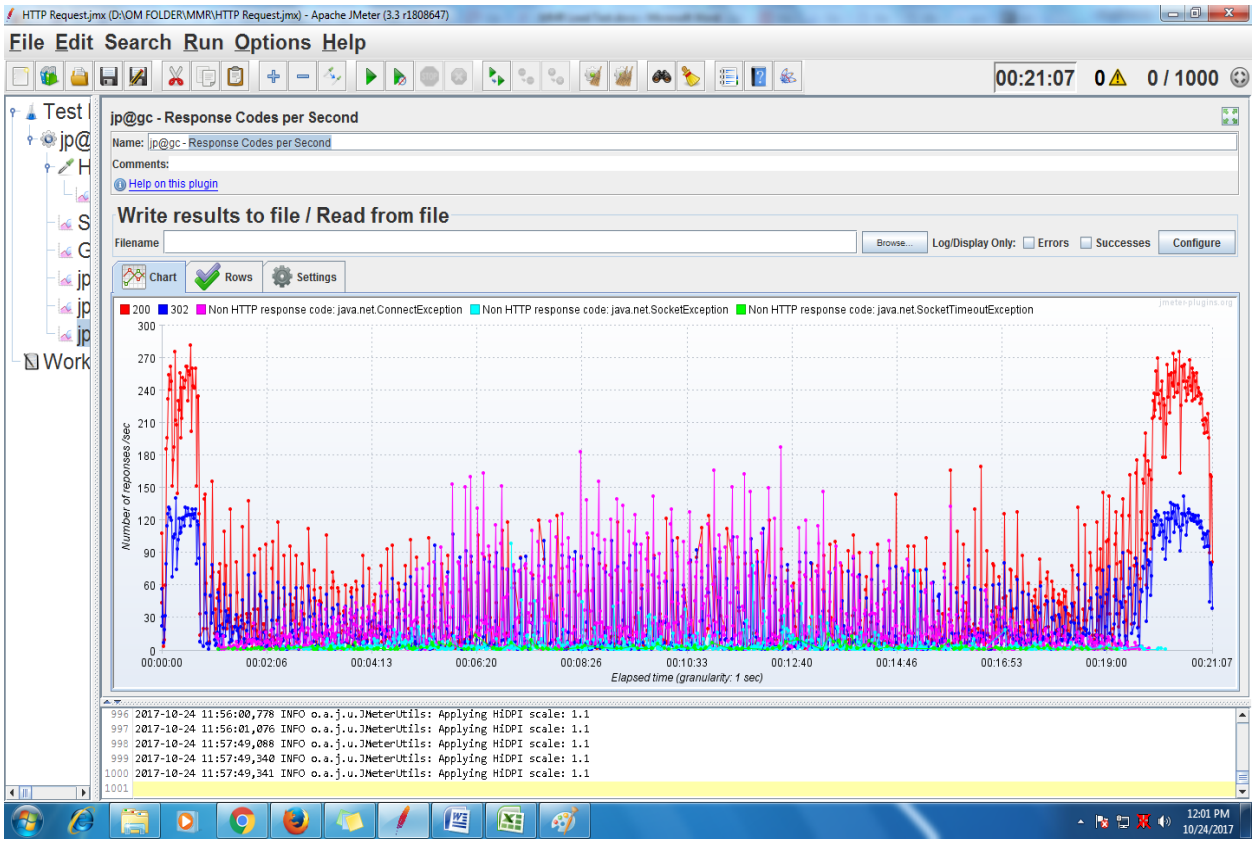
7: Transactions per Second

Green color – HTTP Request Failure

Red Color – HTTP Request Success.



8: Response Codes per Second:

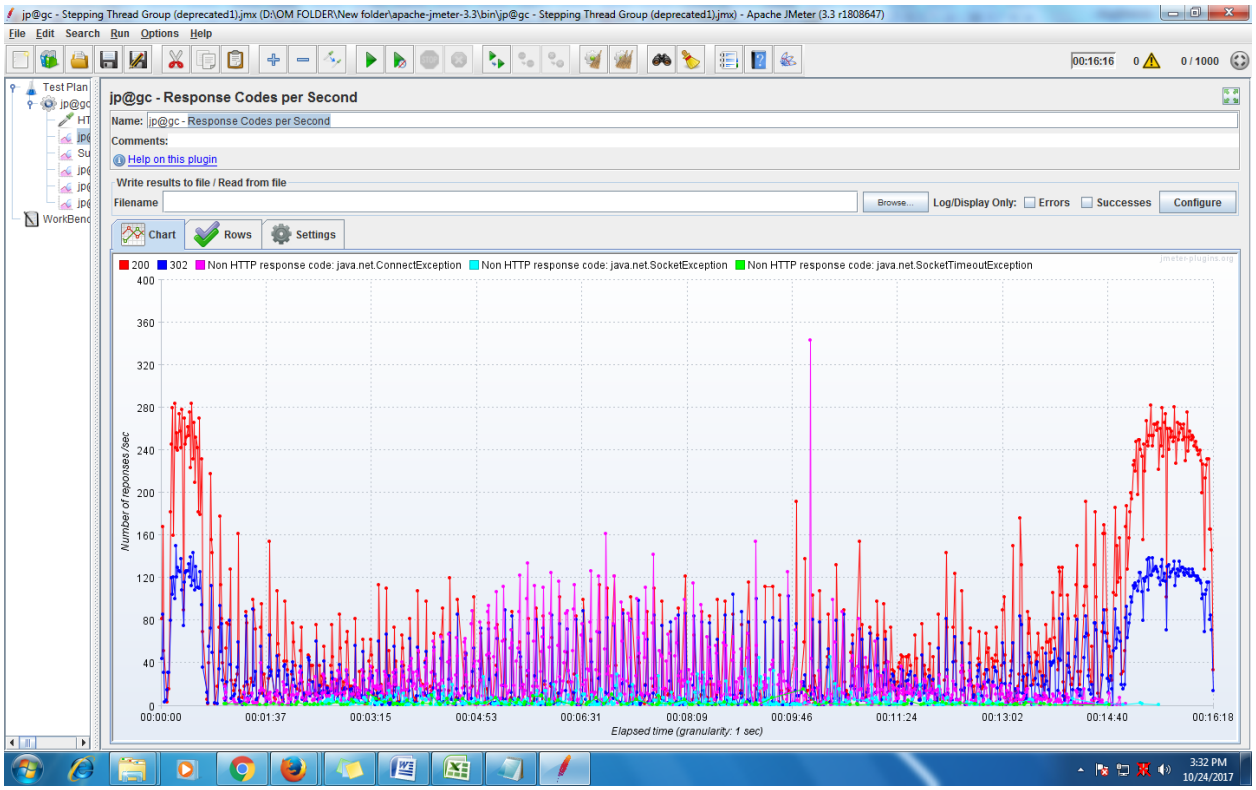


9 Contact list page (makemyreport.com/Users/manualRegisterList): Result

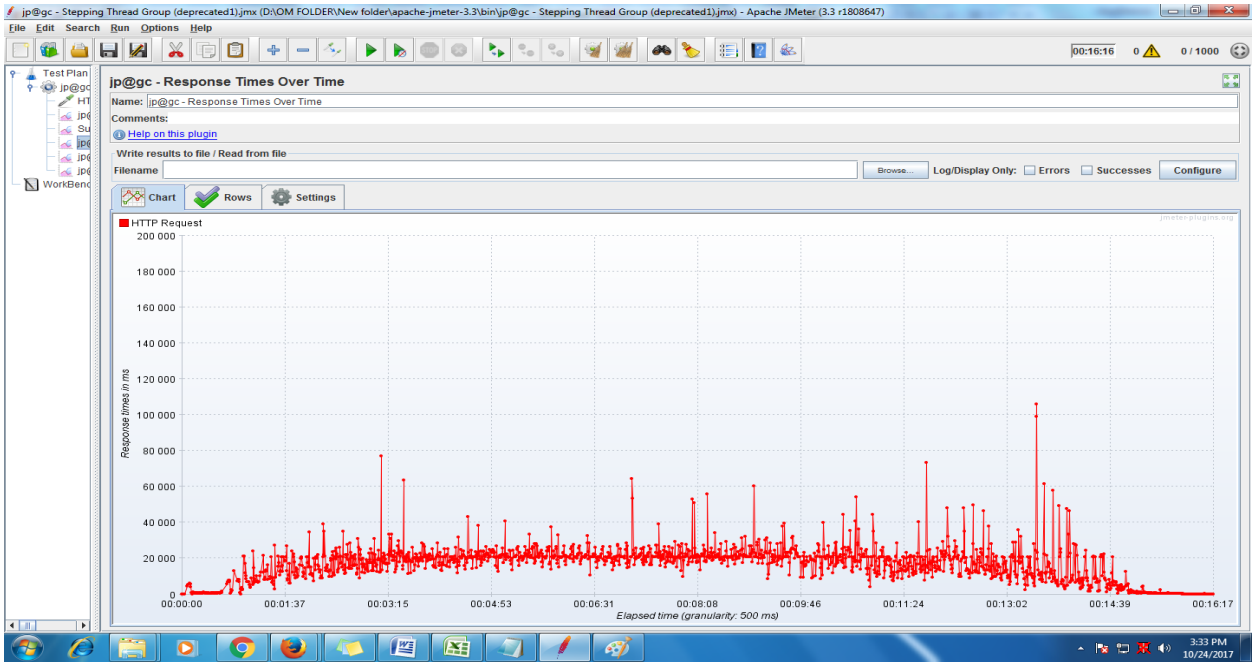
Summary Report:

Create contact page										
Lable	Sample	Average	Min	Max	Std.Dev	Error%	Throughout	Received KB/sec	Sent KB/sec	Avg. Bytes.
HTTP Request	41817	12657	32	127682	12501.19	39.87%	42.80411	404.687	12.84	9681.3
TOTAL	41817	12657	32	127682	12501.19	39.87%	42.80411	404.687	12.84	9681.3

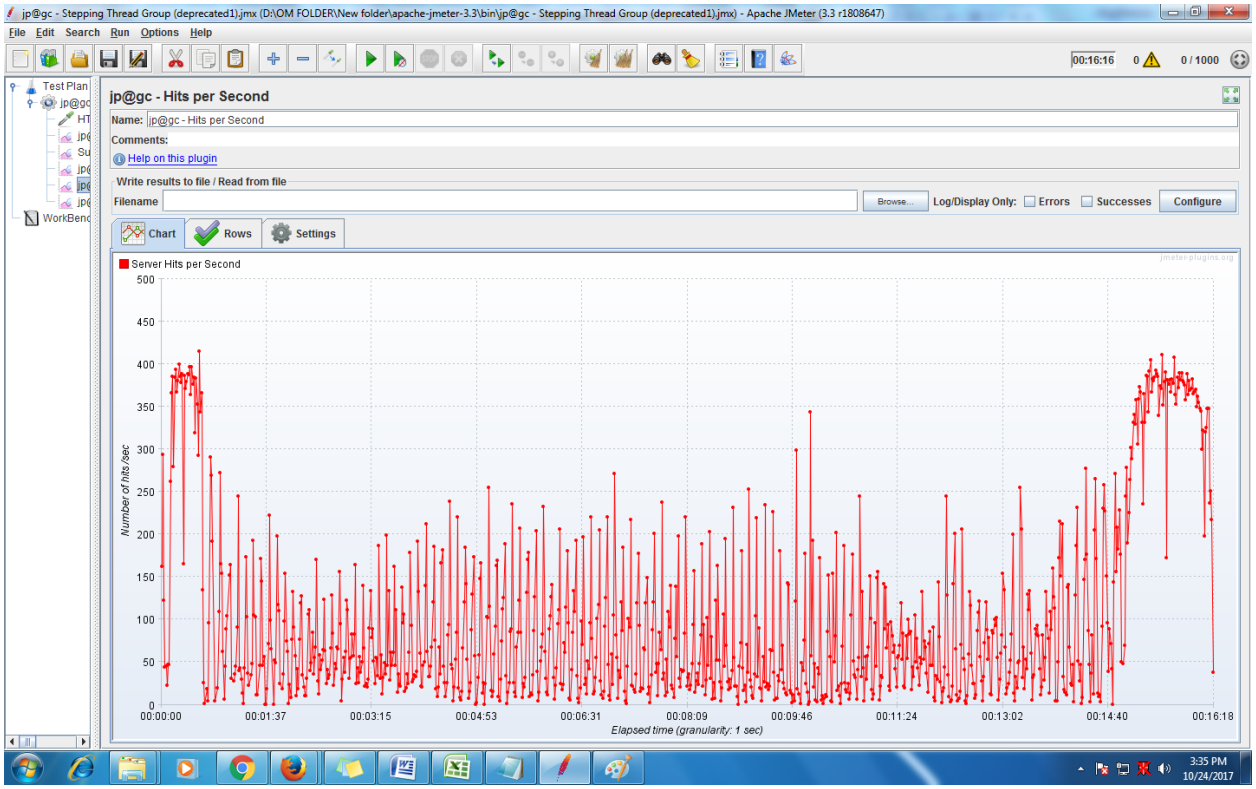
Response Codes per Second:



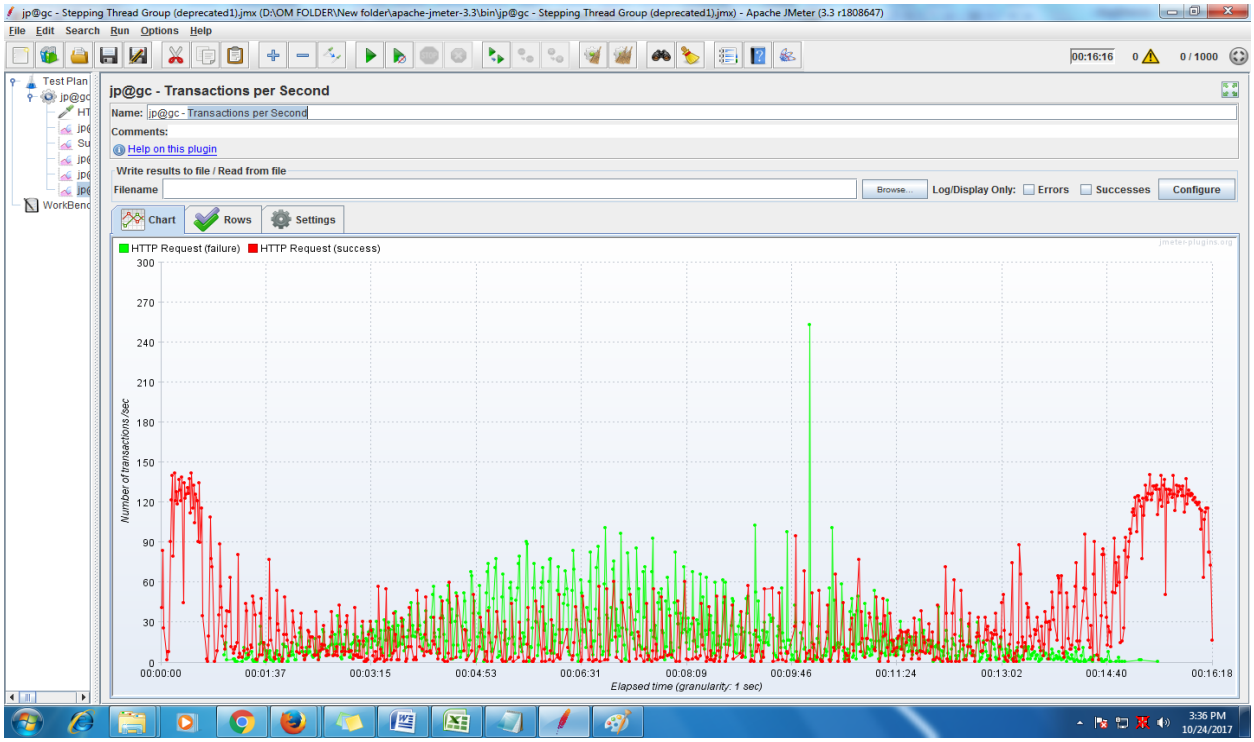
Response Times over Time



Hits per second:



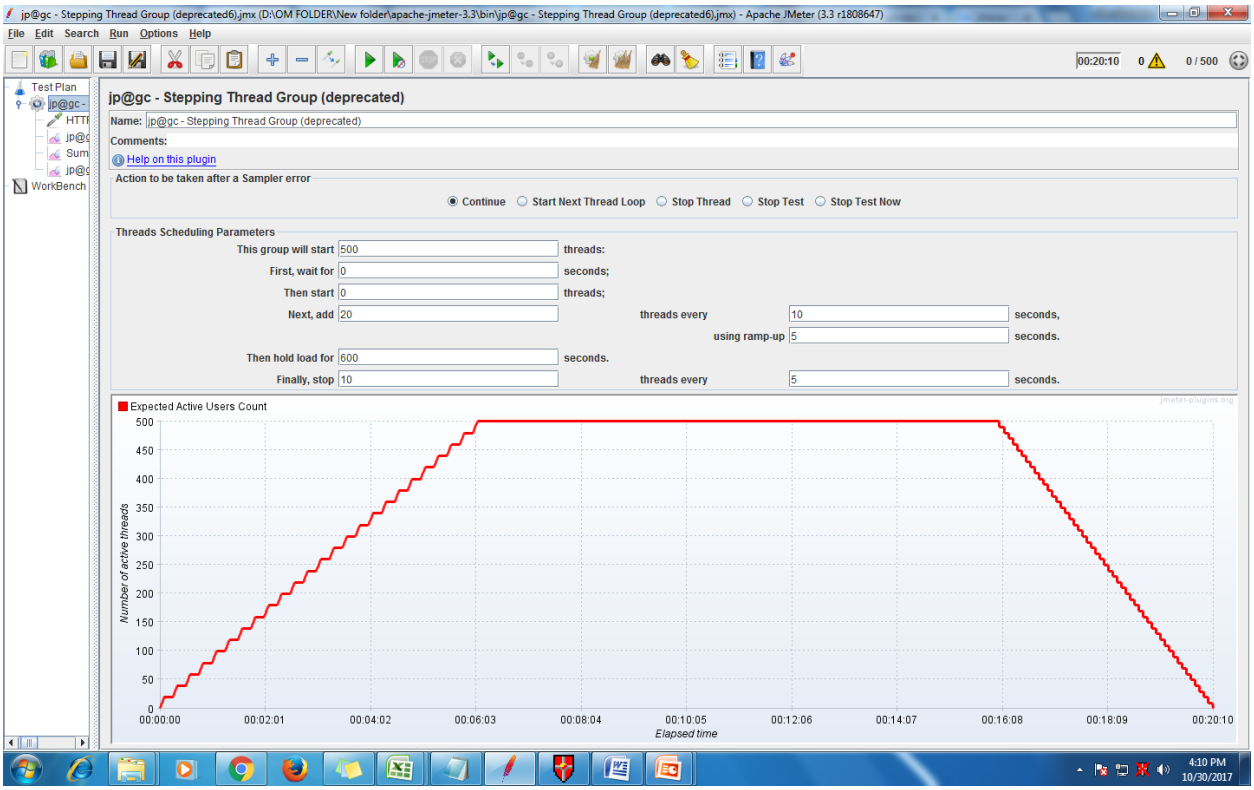
Transactions per Second:


jp@gc - Stepping Thread Group (deprecated1).jmx (D:\OM FOLDER\New folder\apache-jmeter-3.3\bin\jp@gc - Stepping Thread Group (deprecated1).jmx) - Apache JMeter (3.3 r1808647)
File Edit Search Run Options Help
00:16:16 0 0 / 1000
Test Plan
jp@gc
WorkBench
jp@gc - Transactions per Second
Name: jp@gc - Transactions per Second
Comments:
Help on this plugin
Write results to file / Read from file
Filename
Browse...
Log/Display Only: Errors Successes
Configure
Chart
Rows
Settings
HTTP Request (failure)
HTTP Request (success)
jmeter-plugins.org
Number of transactions /sec
300
270
240
210
180
150
120
90
60
30
0
00:00:00
00:01:37
00:03:15
00:04:53
00:06:31
00:08:09
00:09:46
00:11:24
00:13:02
00:14:40
00:16:18
Elapsed time (granularity: 1 sec)
3:36 PM
10/24/2017

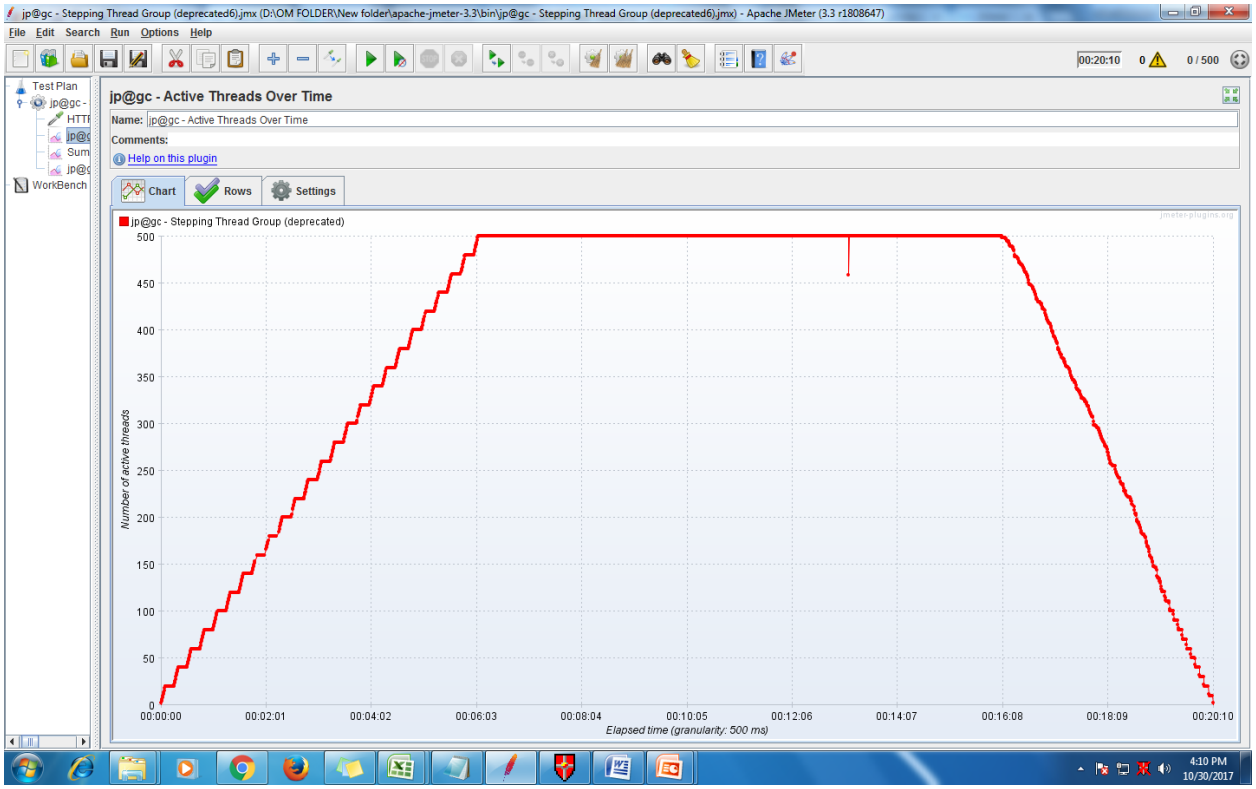
11. Let's implement another scenario: With 500 User.

11.1 Submit a report:

- **500 threads as target load**
- **0 seconds waiting after the test starts**
- **0 threads run at the immediate beginning of the test**
- **20 threads are added every 10 seconds with a ramp-up (or step transition time) of 5 seconds**
- **The target load is held for 300 seconds (5 minute)**
- **Finally, 10 threads are stopped every 5 seconds.**



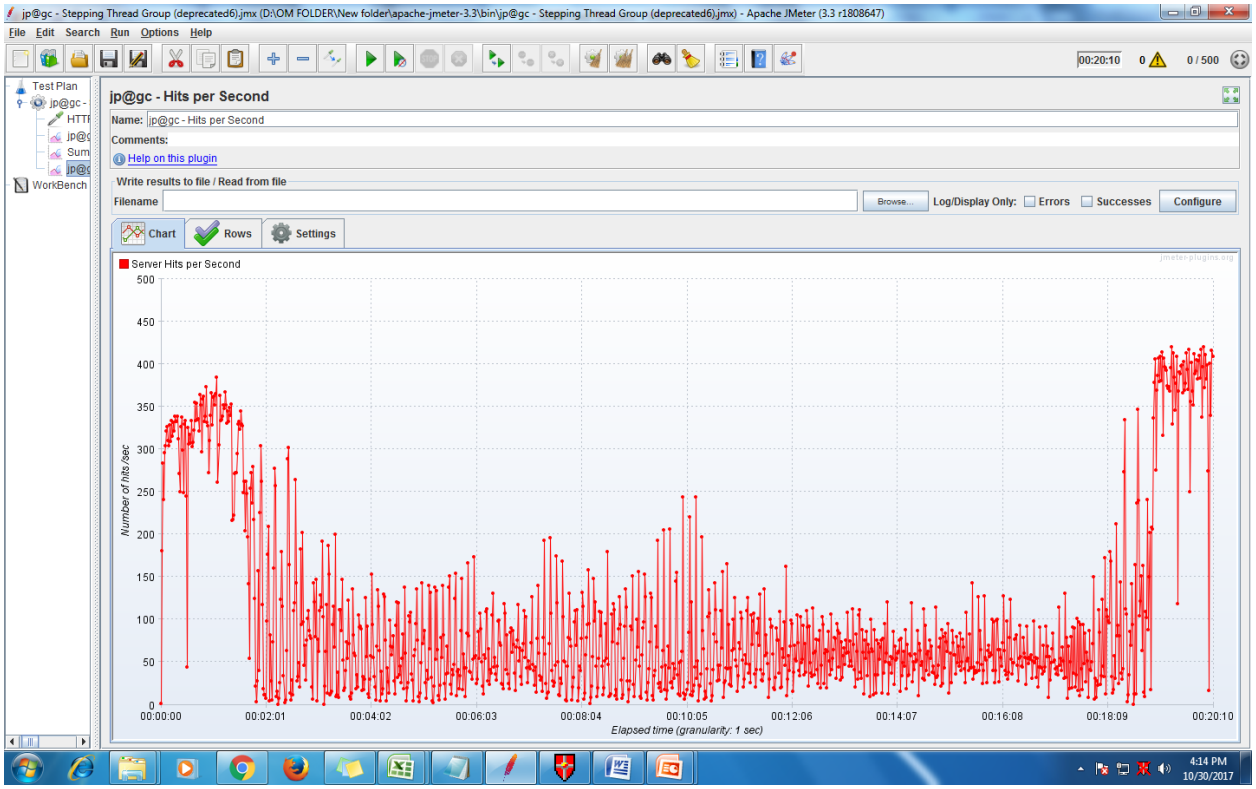
Active threads over time:



Summary Report:

Add New Contact										
Lable	Sample	Average	Min	Max	Std.Dev	Error%	Throughout	Received KB/sec	Sent KB/sec	Avg. Bytes.
HTTP Request	47994	9513	31	137290	12399.08	24.31	39.66279	448.341	14.2	11575.1
Total	47994	9513	31	137290	12399.08	24.31	39.66279	448.341	14.2	11575.1

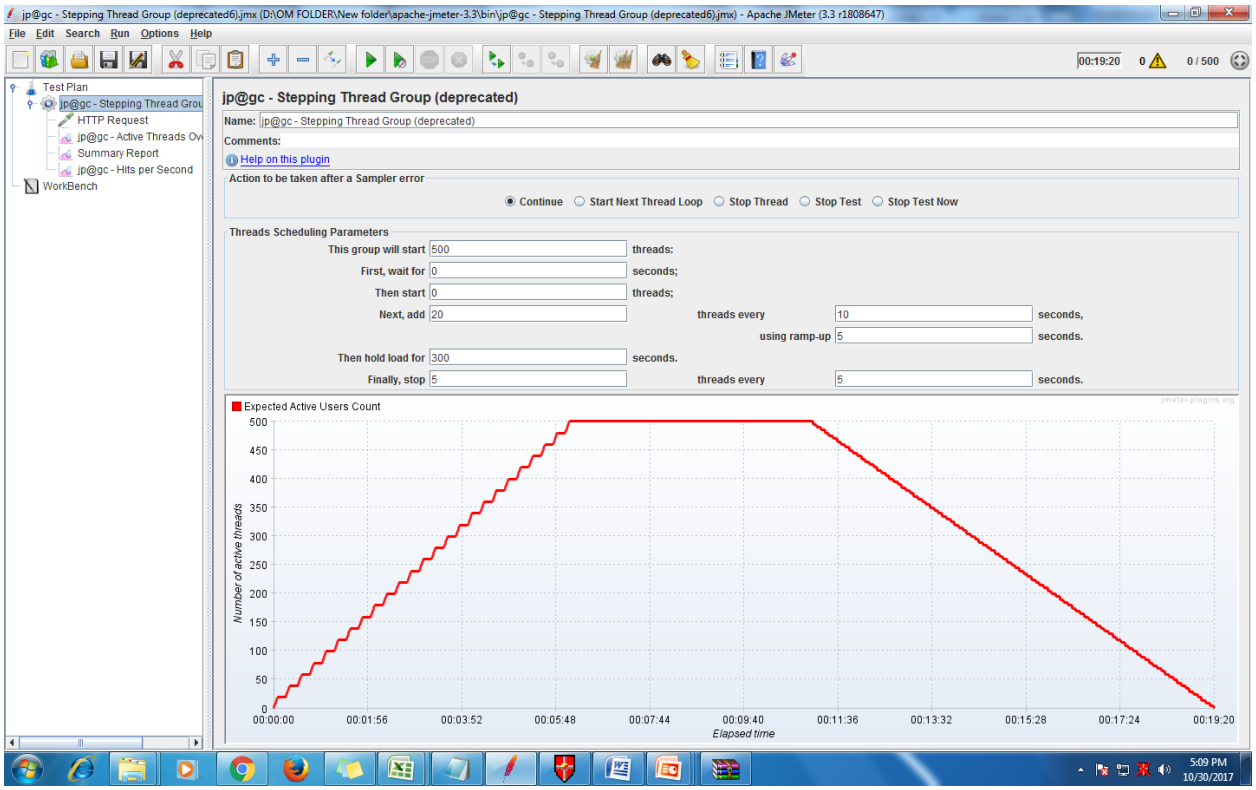
Hits per second:



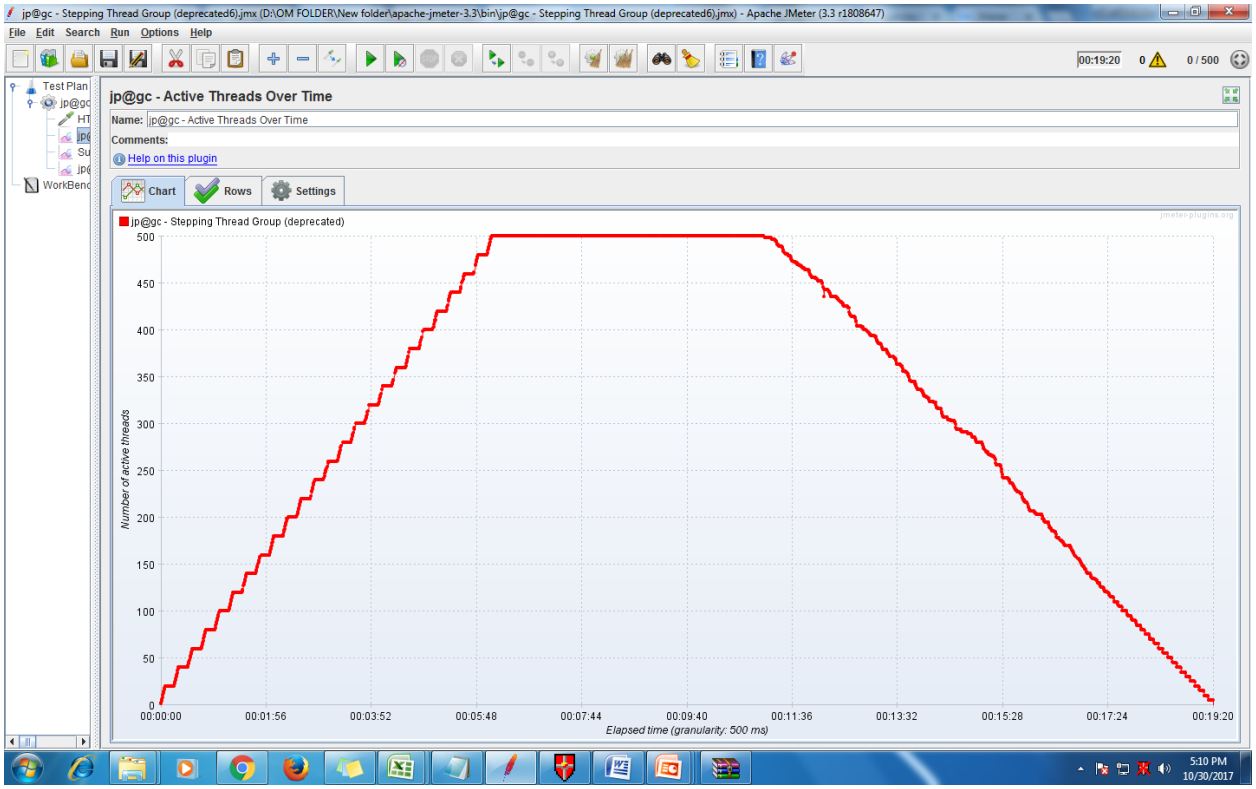
12. Let's implement another scenario: With 500 User.

12.1 My contacts:

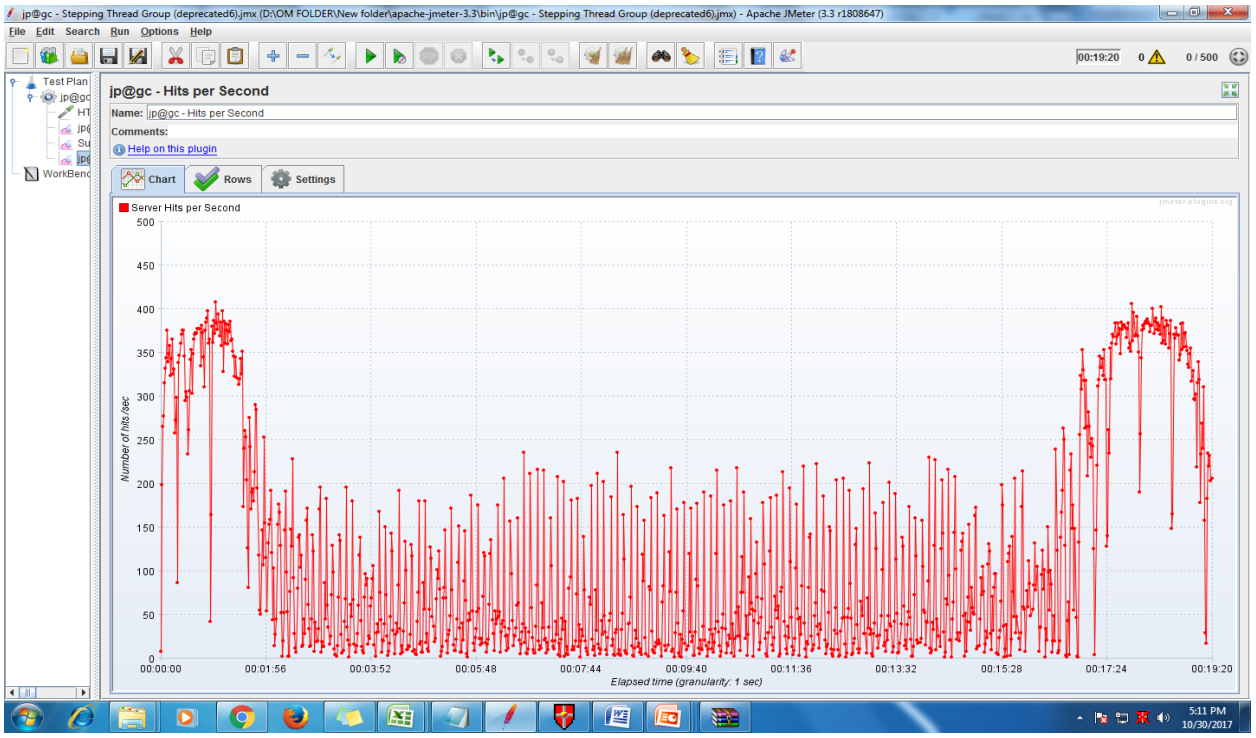
- **500 threads as target load**
- **0 seconds waiting after the test starts**
- **0 threads run at the immediate beginning of the test**
- **20 threads are added every 10 seconds with a ramp-up (or step transition time) of 5 seconds**
- **The target load is held for 300 seconds (5 minute)**
- **Finally, 5 threads are stopped every 5 seconds.**



Active threads Over time:



Hits Per second:

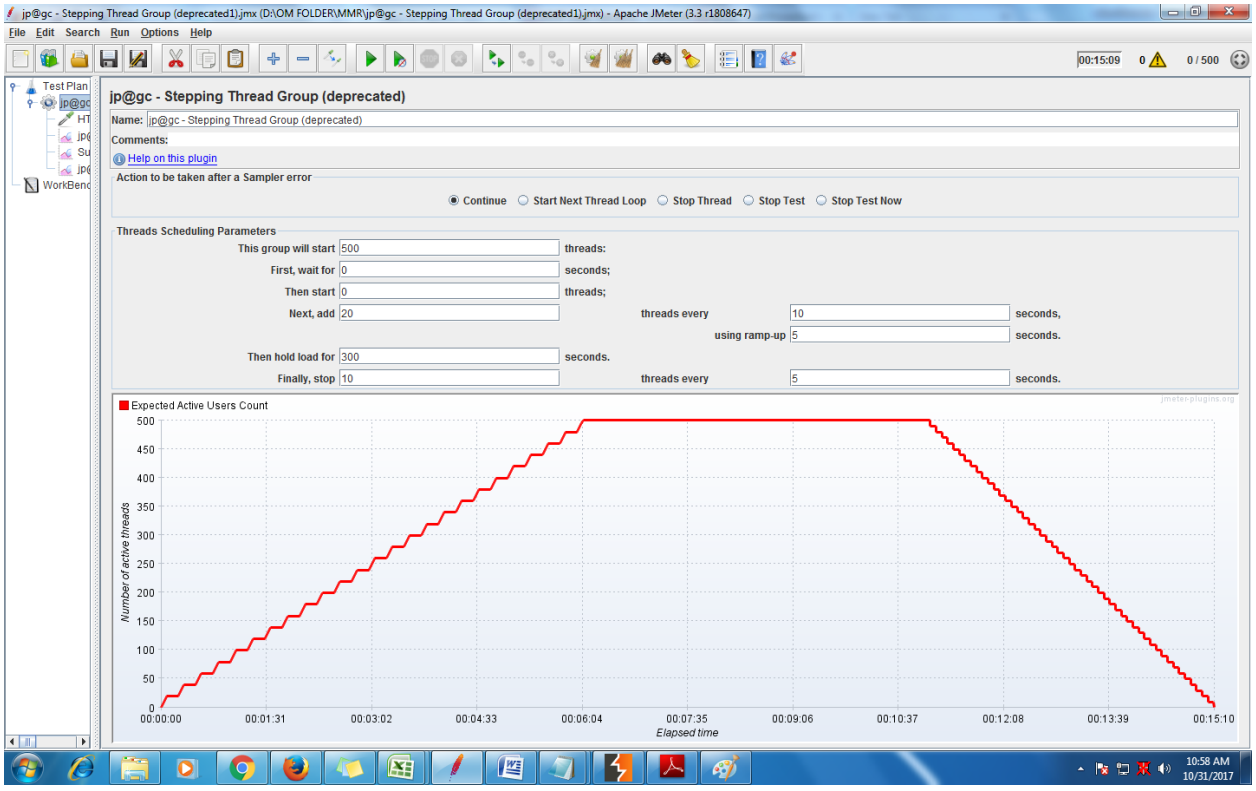


Summary Report:

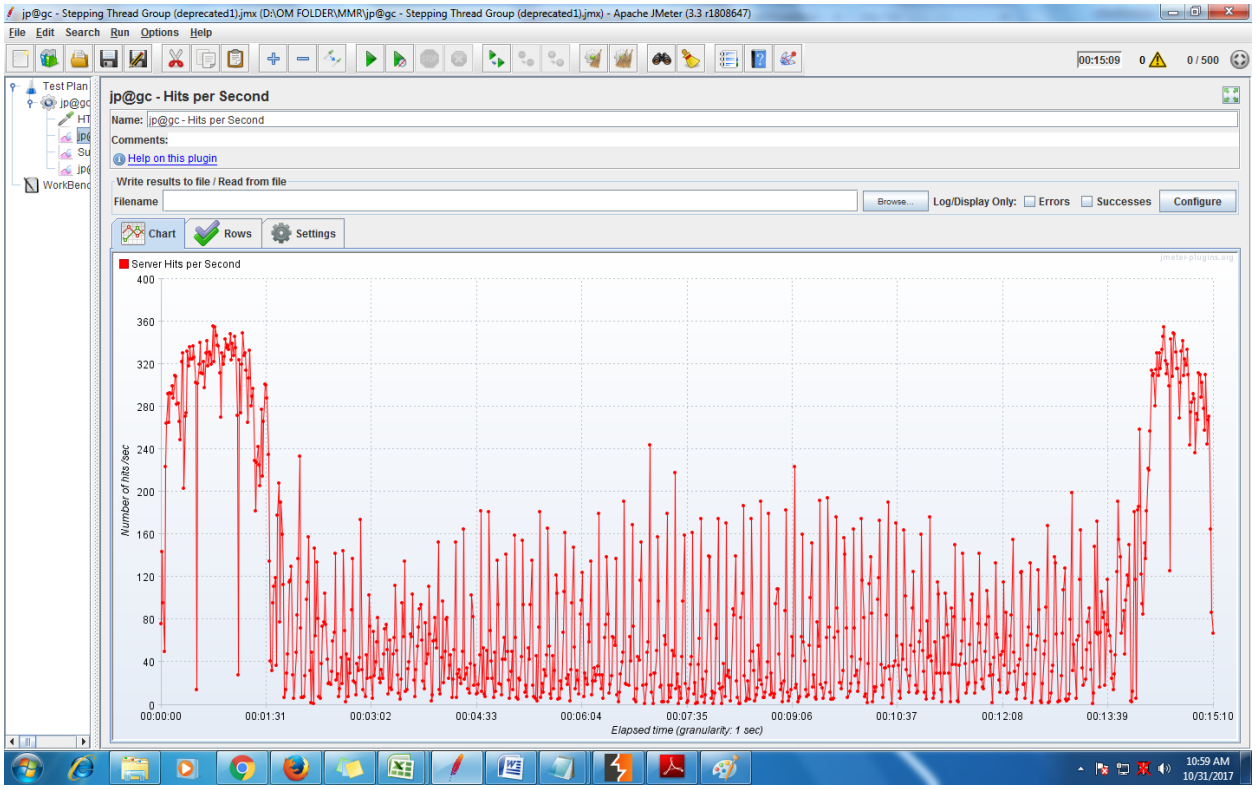
Contact list/My contacts										
Lable	Sample	Average	Min	Max	Std.Dev	Error%	Throughout	Received KB/sec	Sent KB/sec	Avg. Bytes.
HTTP Request	49104	7523	32	120484	11427.28	17.26%	42.33111	514.065	16.55	12435.4
Total	49104	7523	32	120484	11427.28	17.26%	42.33111	514.065	16.55	12435.4

12.2 Response report: With 500 User

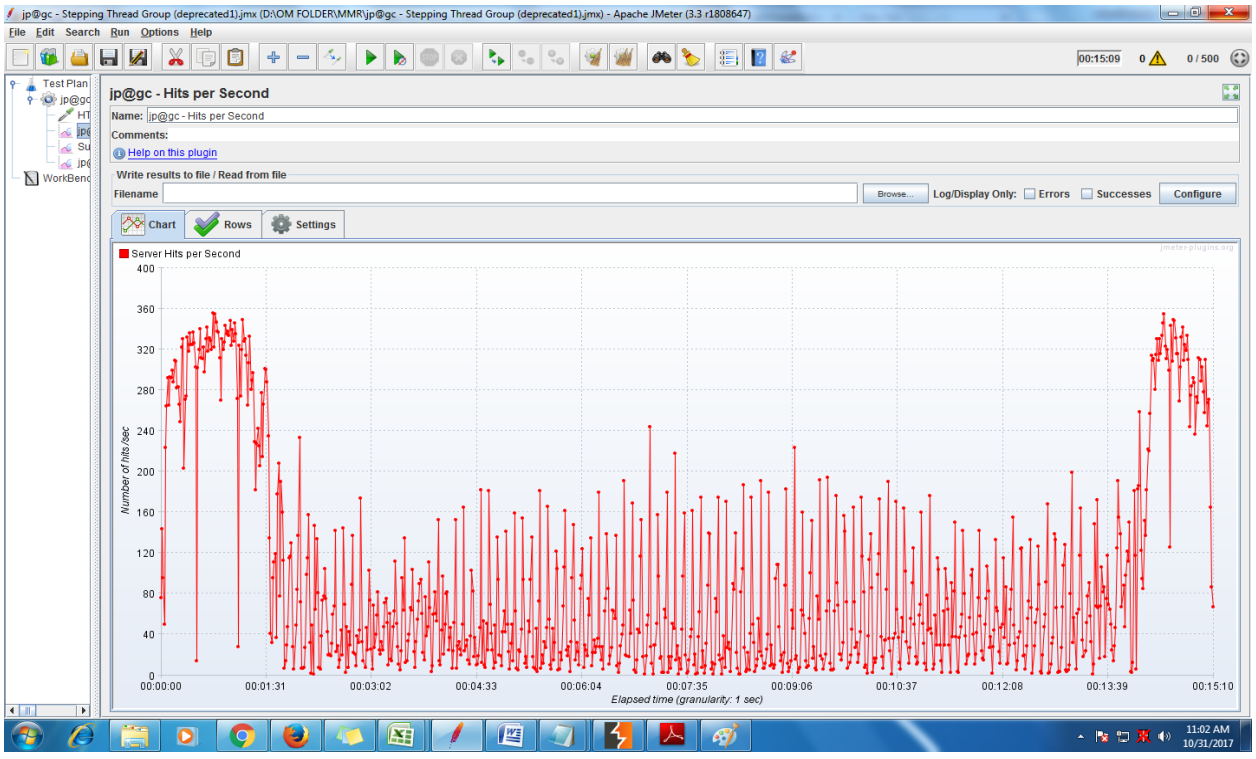
- **500 threads as target load**
- **0 seconds waiting after the test starts**
- **0 threads run at the immediate beginning of the test**
- **20 threads are added every 10 seconds with a ramp-up (or step transition time) of 5 seconds**
- **The target load is held for 300 seconds (5 minute)**
- **Finally, 5 threads are stopped every 5 seconds.**



Active Threads over time:



Hits per second:

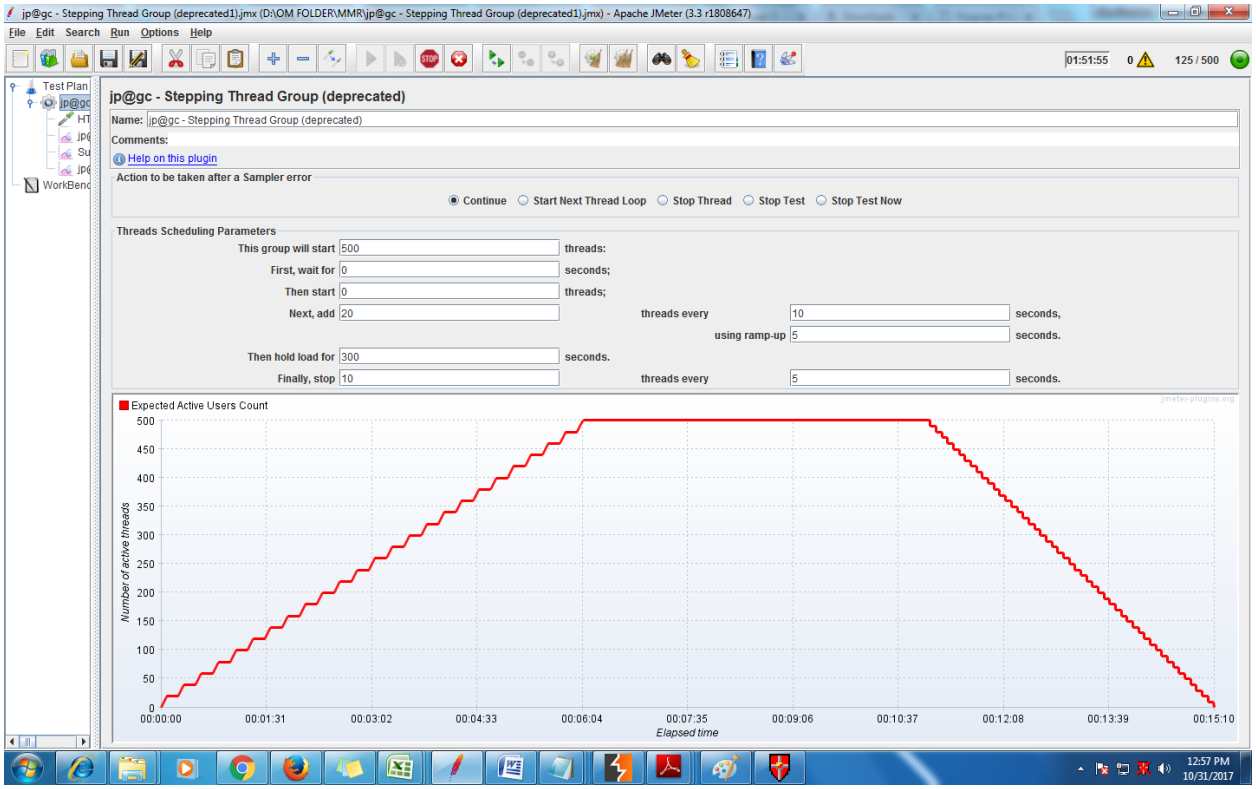


Summary Report:

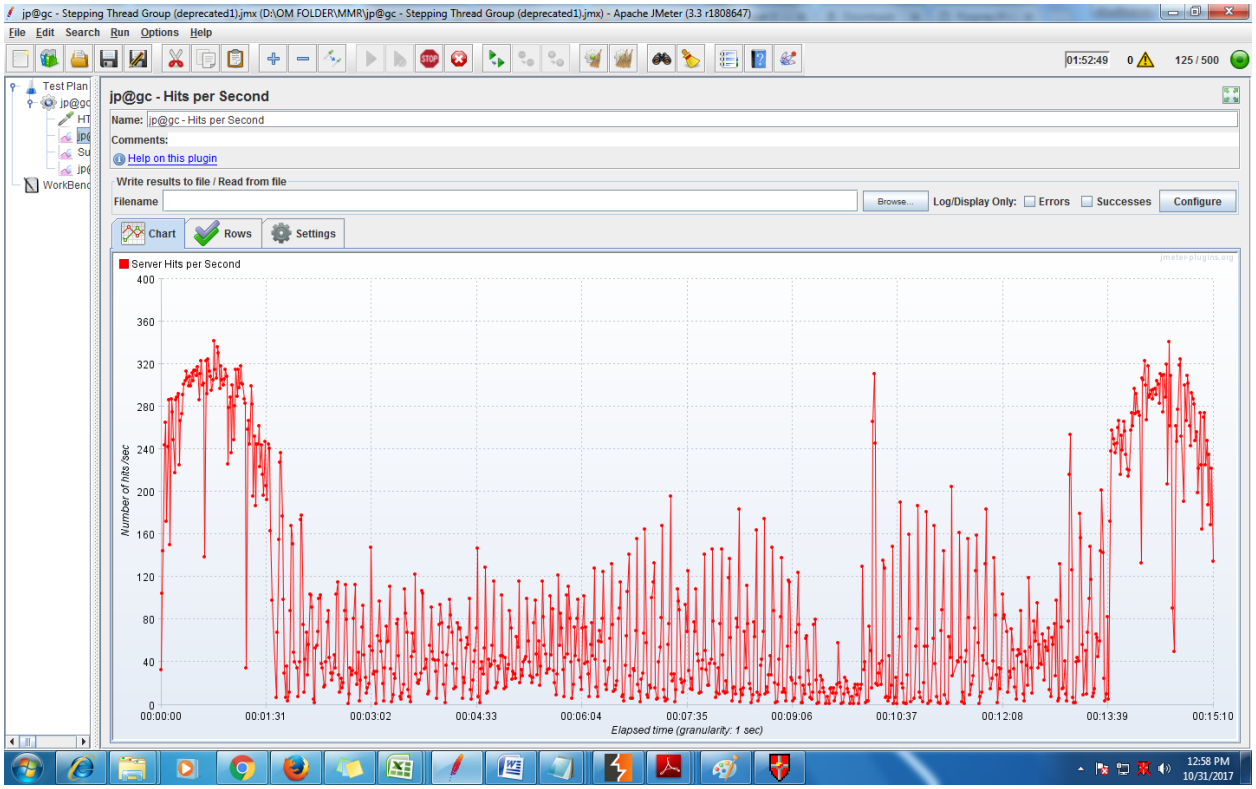
Response report										
Lable	Sample	Average	Min	Max	Std.Dev	Error%	Throughout	Received KB/sec	Sent KB/sec	Avg. Bytes.
HTTP Request	32257	9487	32	126072	12100.83	23.35%	35.44554	404.898	12.85	11697.2
Total	32257	9487	32	126072	12100.83	23.35%	35.44554	404.898	12.85	11697.2

12.3 Archive report: With 500 User

- **500 threads as target load**
- **0 seconds waiting after the test starts**
- **0 threads run at the immediate beginning of the test**
- **20 threads are added every 10 seconds with a ramp-up (or step transition time) of 5 seconds**
- **The target load is held for 300 seconds (5 minute)**
- **Finally, 10 threads are stopped every 5 seconds.**



Hits per second:

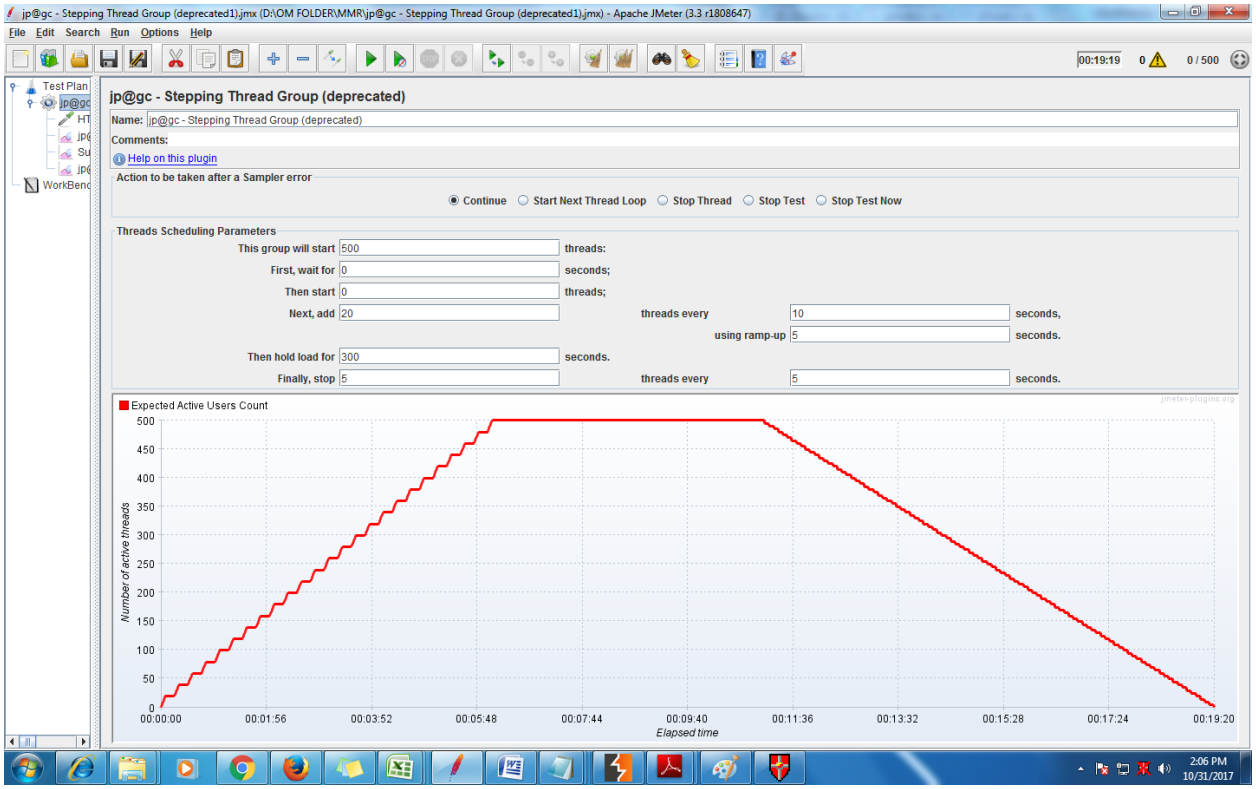


Summary Report:

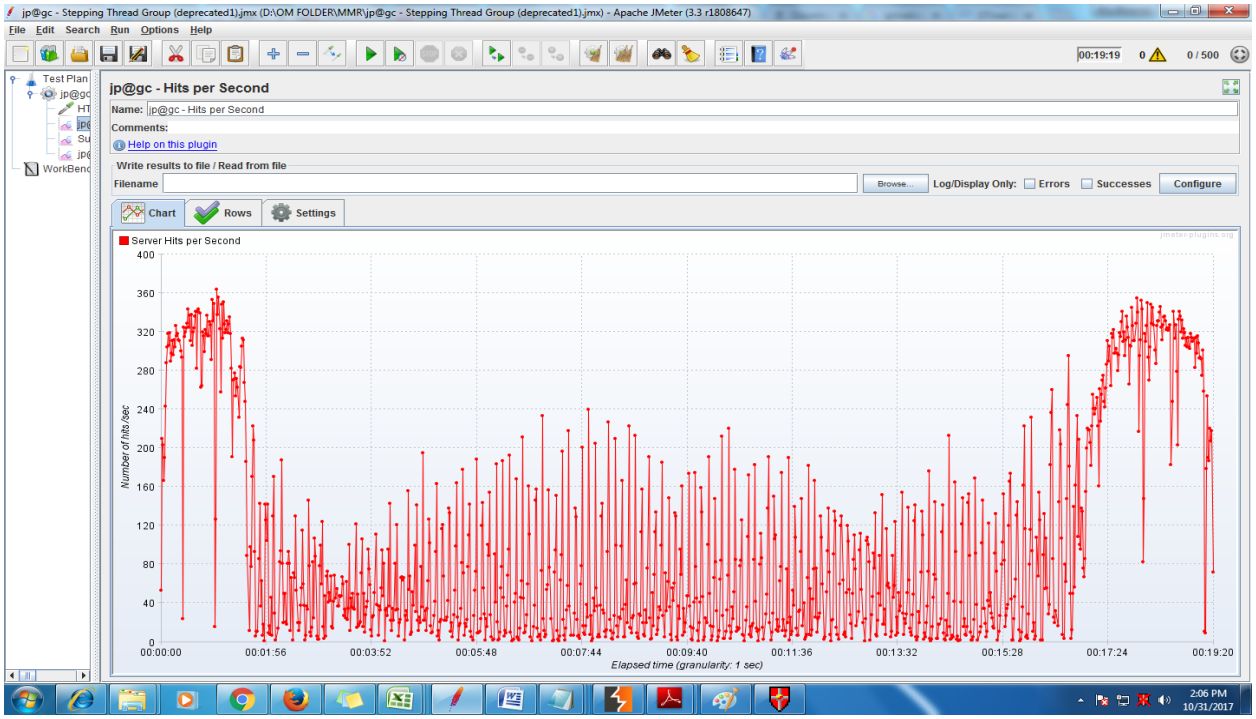
Archive report										
Lable	Sample	Average	Min	Max	Std.Dev	Error%	Throughout	Received KB/sec	Sent KB/sec	Avg. Bytes.
HTTP Request	31227	8872	39	165304	12316.68	20.84%	34.31339	402.291	12.72	12005.4
Total	31227	8872	39	165304	12316.68	20.84%	34.31339	402.291	12.72	12005.4

12.4 Track Response report: With 500 User

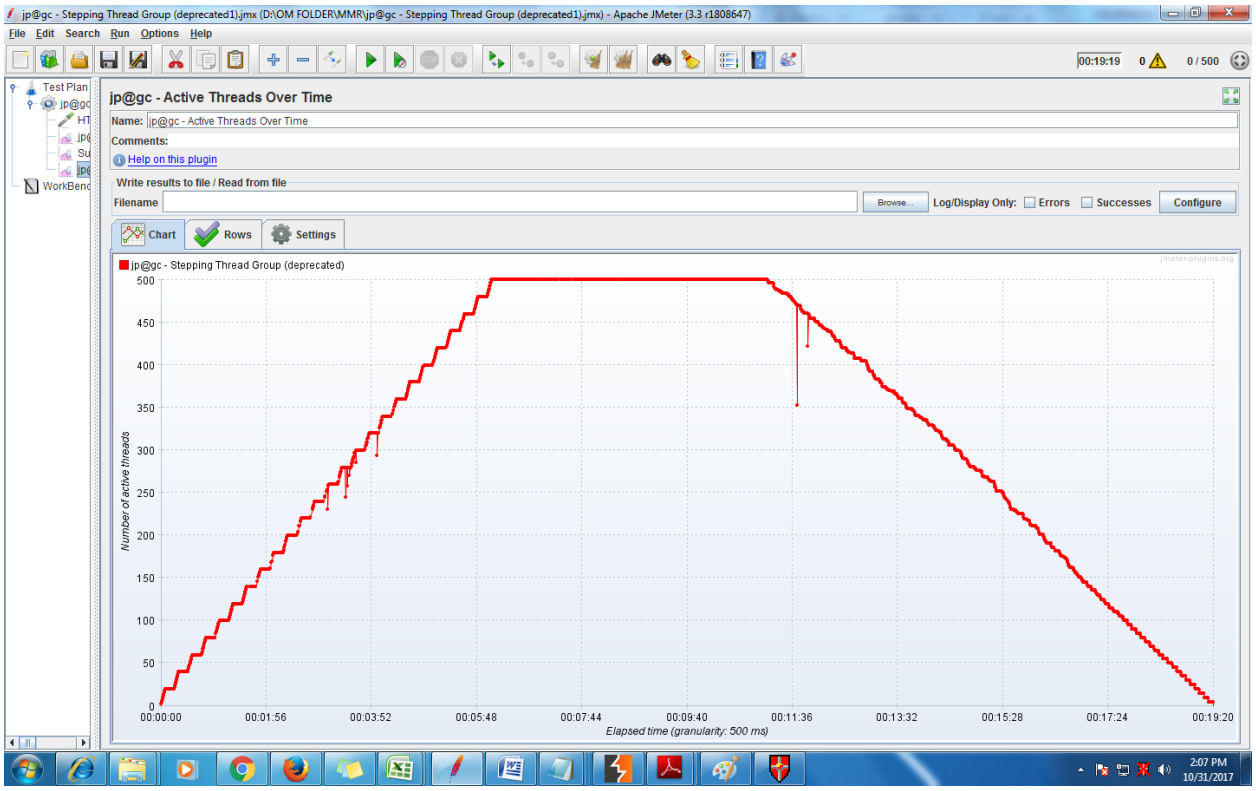
- **500 threads as target load**
- **0 seconds waiting after the test starts**
- **0 threads run at the immediate beginning of the test**
- **20 threads are added every 10 seconds with a ramp-up (or step transition time) of 5 seconds**
- **The target load is held for 300 seconds (5 minute)**
- **Finally, 5 threads are stopped every 5 seconds.**



Hits Per second:



Active threads over time:

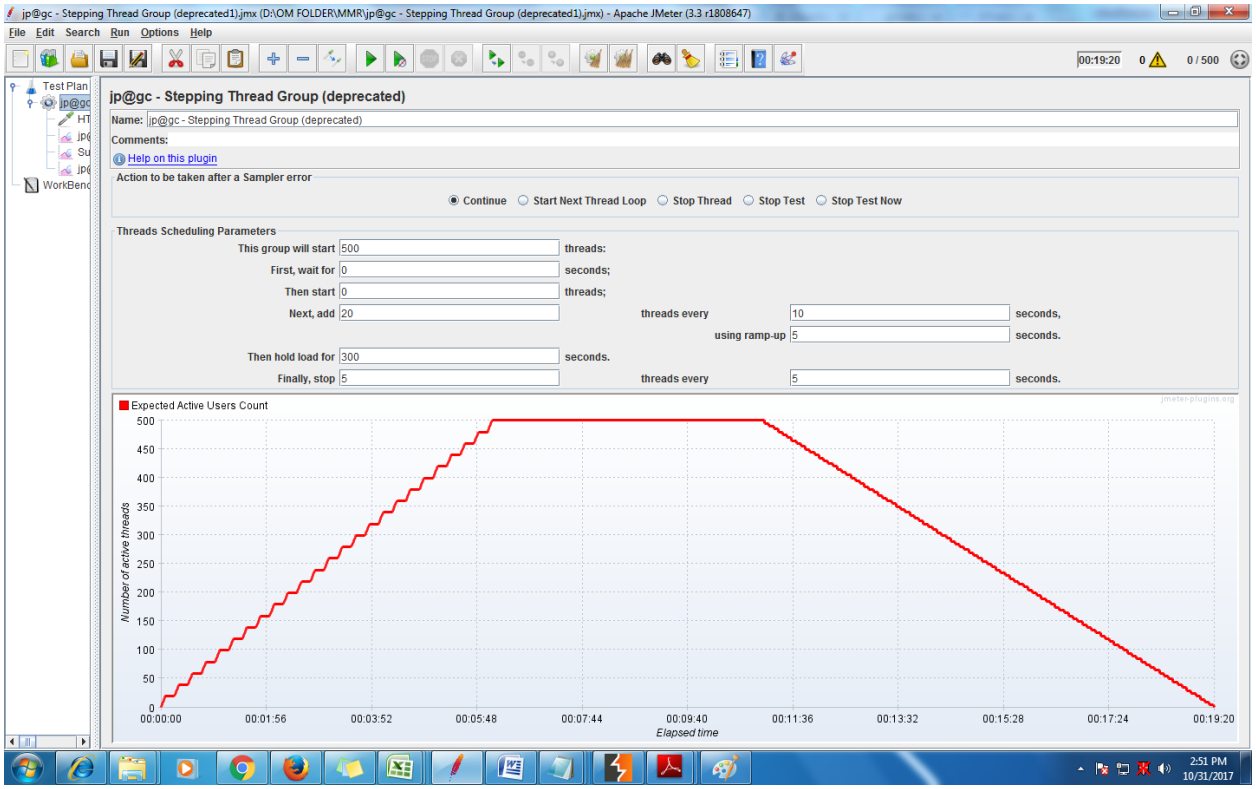


Summary Report:

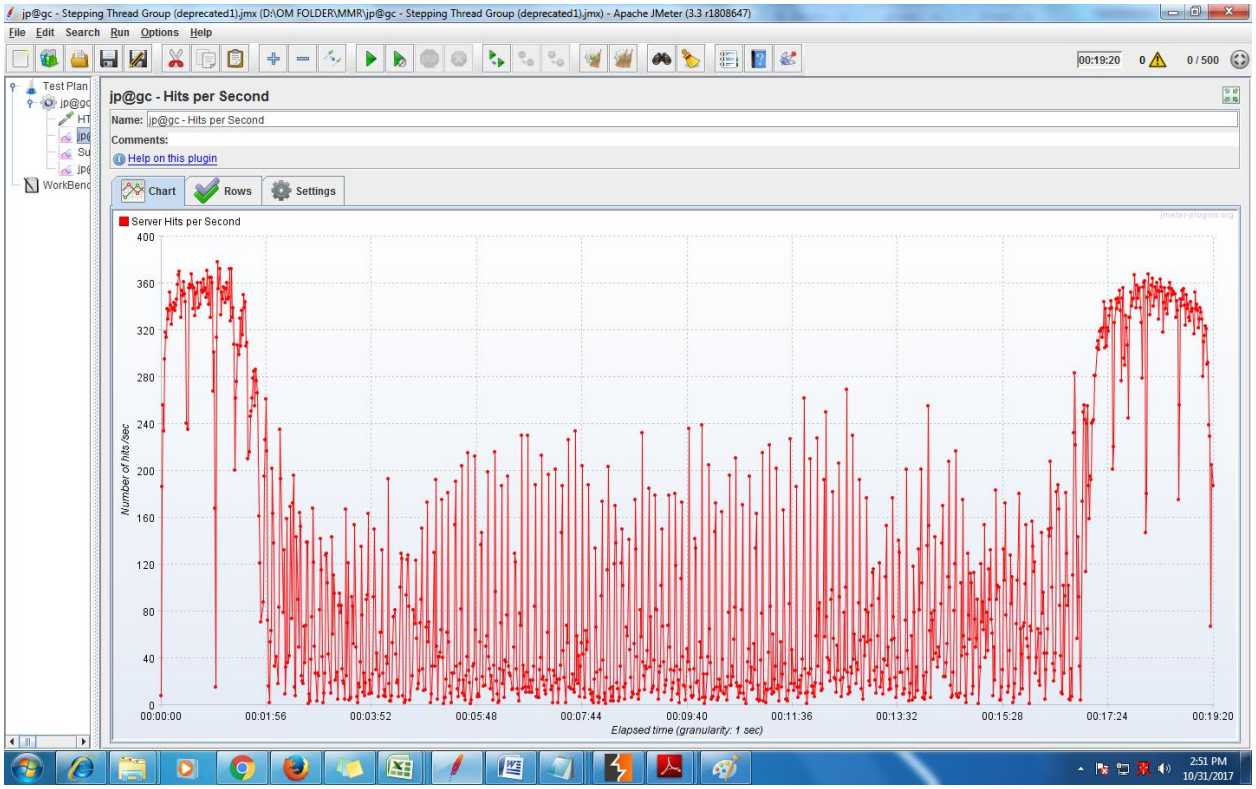
Track Response										
Lable	Sample	Average	Min	Max	Std.Dev	Error%	Throughout	Received KB/sec	Sent KB/sec	Avg. Bytes.
HTTP Request	43123	8562	33	139427	11934.93	19.46%	37.1783	442.073	14.1	12176
Total	31227	8872	39	165304	12316.68	19.46%	34.31339	402.291	12.72	12005.4

View reply from contacts: with 500 User

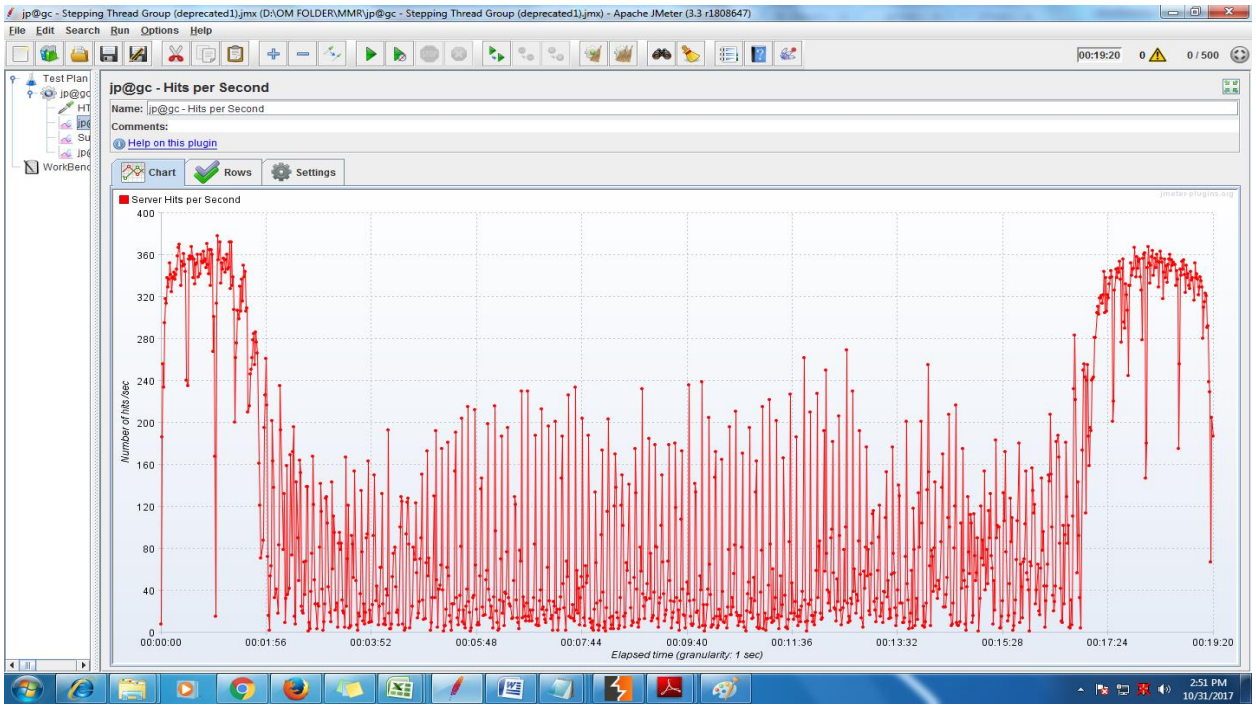
- **500 threads as target load**
- **0 seconds waiting after the test starts**
- **0 threads run at the immediate beginning of the test**
- **20 threads are added every 10 seconds with a ramp-up (or step transition time) of 5 seconds**
- **The target load is held for 300 seconds (5 minute)**
- **Finally, 5 threads are stopped every 5 seconds.**



Hits per second:



Active threads over time:

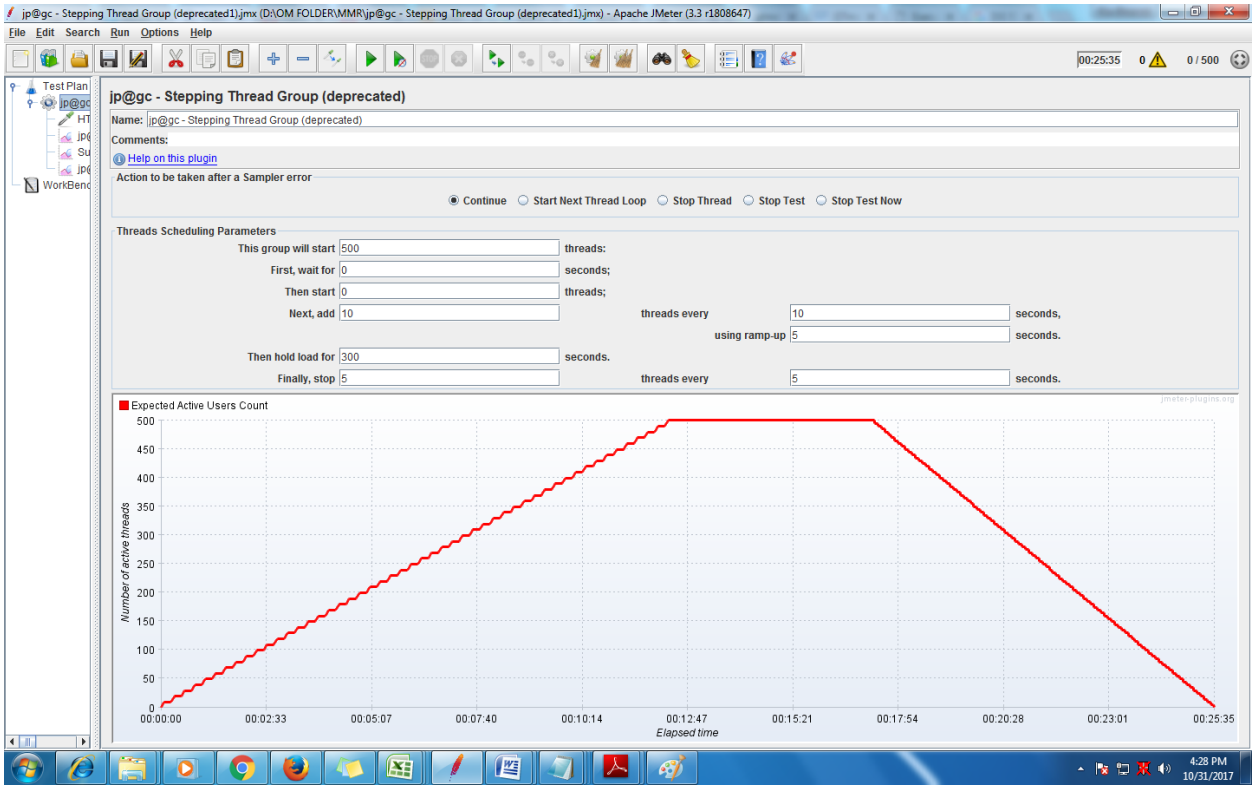


Summary Report:

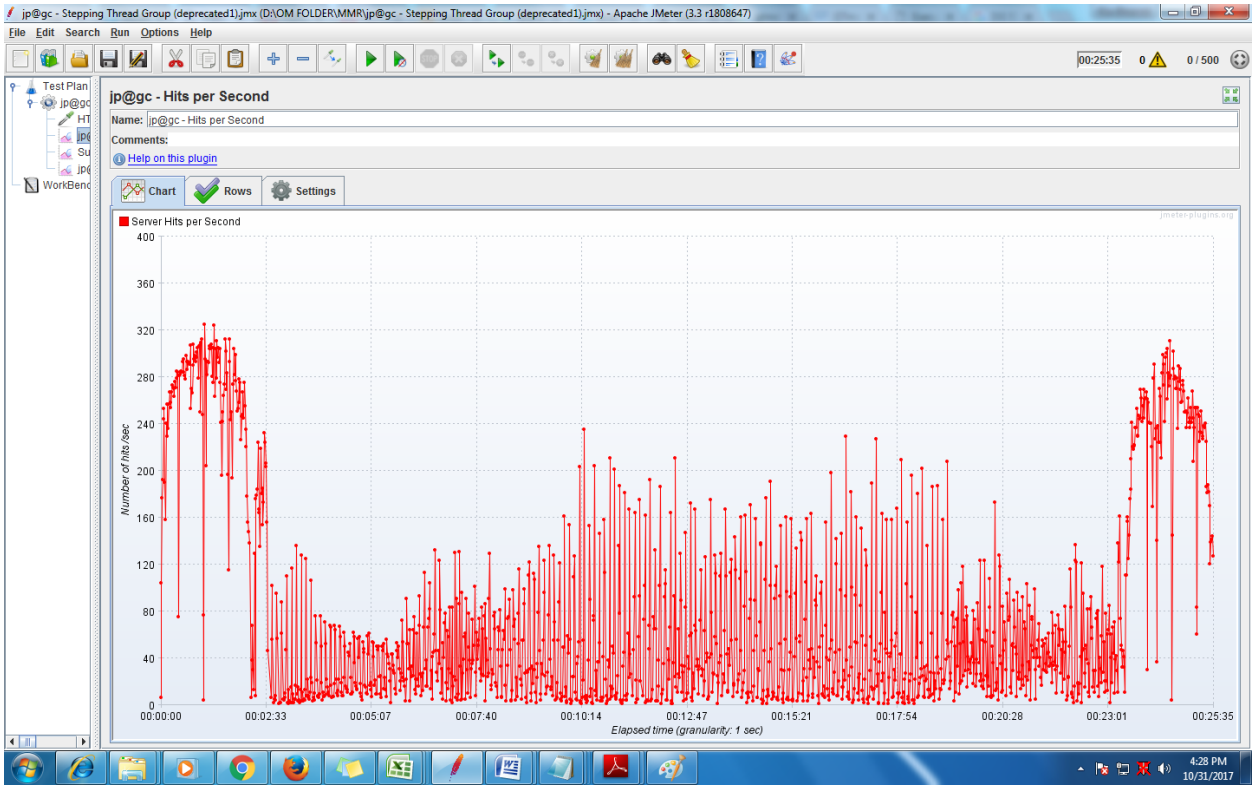
View reply from contacts										
Lable	Sample	Average	Min	Max	Std.Dev	Error%	Throughout	Received KB/sec	Sent KB/sec	Avg. Bytes.
HTTP Request	48545	7607	35	119724	11341.9	17.49%	41.84871	507.079	16.14	12407.8
Total	48545	7607	35	119724	11341.9	17.49%	41.84871	507.079	16.14	12407.8

Design your report and assign to contacts: with 500 User

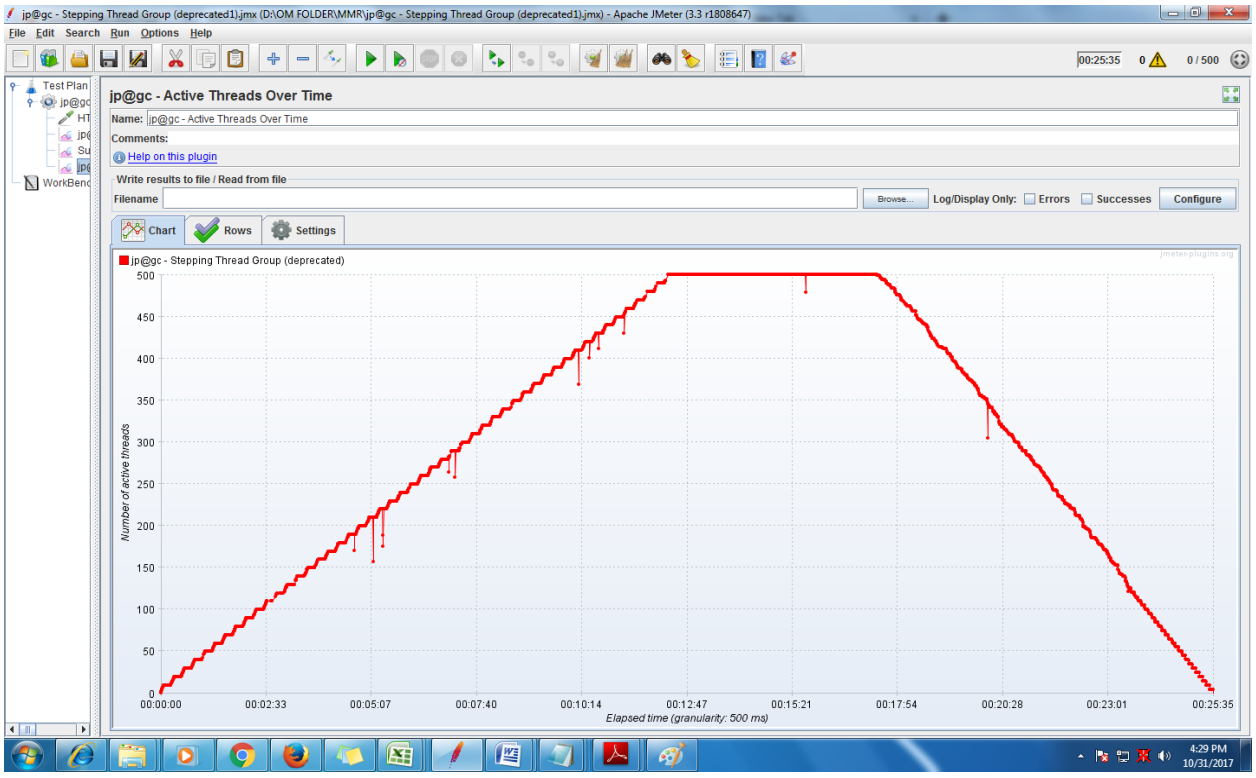
- **500 threads as target load**
- **0 seconds waiting after the test starts**
- **0 threads run at the immediate beginning of the test**
- **20 threads are added every 10 seconds with a ramp-up (or step transition time) of 5 seconds**
- **The target load is held for 300 seconds (5 minute)**
- **Finally, 5 threads are stopped every 5 seconds.**



Hits per second:



Active threads over time:



Summary Report:

Design your report and assign to contacts										
Lable	Sample	Average	Min	Max	Std.Dev	Error%	Throughout	Received KB/sec	Sent KB/sec	Avg. Bytes.
HTTP Request	45258	10248	35	154279	13436.16	22.36%	29.48246	340.417	10.78	11823.5
Total	45258	10248	35	154279	13436.16	22.36%	29.48246	340.417	10.78	11823.5