Manual and Automation Analysis of Load Testing

Ritika

(M.Tech, Software Engineering (CSE)) BBD University, Lucknow E-mail: ritikasin25@gmail.com

Abstract—In today's world, organizations are heading towards the technological era, where most of the work is done technically rather than manually. Softwares are the core part of the technology, due to which there processing needs to be correct as required. We need what is the expected output from these softwares should be the actual output under all the circumstances. Therefore, software testing is the integral part of the technology. Organizations have many arrangements to check the softwares whether they are able to work under the heavy load but manually the load could not be provided accurately because it is not necessary that all the users log into the application at same time, so here in this paper we are providing a manual as well as the automation testing of the live application using tools. In this research paper, we will be providing virtual users to the live applications and then examine its performance and logs. We will be using some tools that will help us examine the exact processing of the application and then we can compare it with the manual testing.

1. INTRODUCTION

Effective functioning of the modern system depends on the ability to produce cost effective software products. Many systems are available all over the world-wide whose infrastructures must support the random and the concurrent access of the multiple users. Sometimes, when applications is working live and hundreds, thousands or millions of users access that particular application then the error occurs as it unable to support bulk users. It happens due to, when the software/application providers do not properly load test or create enough resources to carefully deal with the heavy traffic. Unlike testing the functionality of the system i.e., unit testing, integration testing, system testing which does not depend upon the number of users, load testing deals with the large number of users hitting at the same time. Therefore, it can last to no. of hours to no. of days.

The load testing process that is carried out is divided into the following steps:

- SYSTEM ANALYSIS
- USER SCRIPTS
- SETTINGS
- PERFORMANCE MONITORING
- ANALYZING RESULTS

Hence, we can say load testing is the measure of the entire web application / software's ability to sustain a large number of users and transactions.

Basically, there are two methods to carry load testing.

- MANUAL TESTING
- AUTOMATION TESTING

Manual Testing, as the name suggests is the testing that is done manually by the testers or developers or the clients. Whereas, automation testing deals with the related tools that help in testing the application after the settings are done. Due to the iterative nature of the load testing manual testing is not very beneficial or practical in nature that is why now in the present world most of the organisations are heading towards the automation testing of an application.

2. RELATED WORK

In the research paper by Zhen Ming Jiang, he proposes an automated analysis of load testing. In his work he starts by recording the execution logs and performance metrics. He performs the functional analysis and performance analysis under the load test. His approach is based into 3 steps: Firstly, abstracting the log line execution and performance metrics into performance metrics. Then, functional models are generated to uncover functional problems.Lastly, the automated evaluation of performance metrics is done. Therefore, in this paper the author has done whole working related to functional and performance testing through automation tools.

Kartheek Muthyala, Rajshekhar Naidu presents a novel approach where both testing and data mining domains are combined. They proposes the technique in which all the test cases are generated and load testing is done manually. After creating test cases they are parsed using data mining techniques.

Jorge Vizcaíno, in his paper has created a model through which the traffic system could be tested for machine to machine communication. It was also made possible to increase

273

the number of servers thus increasing the number of instances in this script.

Niclas Snellman, in his research paper shows performance and scalability analysis in rich internet applications (RIA) through automation tools. In this proposed approach there is no requirement server-side monitoring or any server modifications. First the ASTORIA framework is build of RIA and then performance testing is done.

Diplom-Ingenieur, George Din, in their paper discusses about the telecommunication applications. In such applications it is hard to analyse the performance of the application. So they proposed a methodology that copes up with the mentioned characteristics.

3. PROBLEM STATEMENT

As we are aware with the fact that testing is most renowned and important technique that is used in every organisation be it related to e-commerce, telecommunication, etc. Therefore, testing is the process of executing a program in order to find the relevant errors that could be solved as soon as possible. But when it comes to the load testing where we need to check whether an application is supporting a bulk of users than the problem occurs to how to get it tested. Because conducting such tests manually is possible but not accurate neither could be performed at a definite time.

Thus, in this system we are building a model where a live application testing will be carried forward. The problem occurs while the manual testing is performed because at the same number of users cannot hit the same application there will occur a minute difference in the login timings.

Here we will perform both the manual as well as the automation testing. Manual testing will be performed by the users, whereas, the automation testing will be performed by the virtual users via. load generators. And after both the testing will be performed we can compare in the results of the accuracy. Because when it comes to the load testing the performance and functionality could only be accessed correctly using automated tools.

4. OUR APPROACH

Our approach consists of the following steps:

- Prepare the requirements of the application that is needed to be tested.
- Create the test plans required for the test case generation.
- Test cases are generated and their execution is done accordingly
- Results are tabulated according to the expected and the actual output.

The above steps are to be followed first through manual testing i.e. users/ testers and then through the automated tools.

4.1 MANUAL ANALYSIS

The testing techniques that are followed during the test cycle during testing manually are,

- White box testing
- Black box testing
- Unit testing
- Integration testing
- System testing
- Acceptance testing

In this technique the tester takes over the role of end users and makes the application/ software defect-free.

First and foremost, the test plans are generated for the testers to ensure the completeness of the testing. The test cases are generated manually without any involvement of the automated tools.

Then accordingly tests are PASS OR FAIL based on the user analysis. The failure or defect is categorised under the medium, average, high or very high category.

The failure where occurs a bug report is attached to it so as to let other team members or team lead to go through the bug that was encountered by the tester.

4.2 AUTOMATED ANALYSIS

It is the process in which the pre-scripted tests are executed using the software tools based on software application. Automation analysis helps in minimising the sets of the scripts so as to simplify the tests. These tools are capable of executing tests, reporting results and comparing results with expected and output. These processes can be done repeatedly any day any time.



The requirements are added in this tool, then the test plan is created where test scripts are generated and lastly in the test lab the test scripts run. Based on which the results are tabulated.

tp://got/15/gobe/start_a.htm RY									183 - 855 -
Represents bit New Fact	itri Atalytit					_	Dorwin	I TOURSTRAVELS, P	ujedt Marcury Jou
44×57.08									
Nere Versaulars	Dead Give Status No. But	- RegD	Adror	Revenued Act Revenued	Creation Time	Greation Date 1,090014	Printy 4.Very High	Type Software	Product All Projects
+ Rore	Patent	(Piccocc)	rtin	Net Reviewent	1.42.35 PM	1/29/2014	4-Very High	Schware	All Projects
 Flights Roteto 	V Pacand	(P00003) (P00004)	rtin rtin	Not Reviewed Not Reviewed	1.45.14 PM	1/29/2014	3-Hgh 3-Hgh	Software Software	All Projects All Projects
- Car Rentals	V Escant	(90005)	rtka	Not Reviewed	1.45/44 PM	1290014	3-High	Software	All Projects
 Gruises Declinations 	 Passed No.Euro 	(Piccess) (Picces)	ritin ritin	Not Reviewed	1.46:53 PM 1.47:35 PM	1090014	2 Medium 3 High	Software Software	Al Projecto Al Projecto
 Vecidions 	✓ Pacand	(Process)	of the	Not Reviewed	1:47:58 PM	1/29/2014	3.69	Software	All Projects
 sign-on Register 	✓ Excent	(P00010) (P00012)	rtite rtite	Not Reviewed Not Reviewed	1.4805 PM 1.50:37 PM	1/29/0014	4-Very High 4-Very High	Software Software	Al Projects Al Projects
- Support	✓ Encond	(P00013)	rtin	Nd Reviewed	1.51:02.PM	1/29/0014	3 High	Software	Al Project
· Contect_us	✓ Essand In Ban	(PODD14) (PODD15)	rtin rtin	Not Reviewed Not Reviewed	1.51.23 PM 1.22.45 PM	1090014	3-High	Software Software	Al Project Al Project
+ bgin	> 66.8x0	(9)00016)	stin	Nd Reviewed	1.21.11 PM	1.00.0014	4-Very High 4-Very High	Software	AB Projecto
- boding	ie tacha	(POD017)	dia	RE FEVERARE	1.21.489%	1000314	4-Very Hgt	Software	Al Project
Description Attachments Hit	ittry								
			Serv	er Time: 1:46/PH 1/00/001	•				-
Discary Quilty Cert	🗿 Percary Quality Cent	Atione New	nte Bio	ad Test Profession	Dome (c.)	i tenta	1-Dati	VI	Giscie 995 99
		_							_
Ity Center 9.0 - Nicrosoft Int Favortes Tools Help	enti tojunir								
) 🗟 🗟 🖉 🔎 🚧	nti 🐈 Felottes 🙆 😋	- 8 -	3						
Control advantation of the			-					CEACK NEXT >	
and the second sec							Dates	KEACK REET >	TEEL . HEP
				_					
		Design Steps +	Test Script + At	Sacherents Req Coverag	e + Liskes Derect				
i lt X ∮ V · lž ► Saled	El • Q Letats			tacheerts Reg Coverag					
t it X ∮ 7 · lž ► Skjed	E + Q Desab	ie X ia Restane	🐮 🥒 🗽 🕴	- @ # # I	E M				
i lt X ∮ V · lž ► Saled	E + Q Desab	ie X ia Restane	🐮 🥒 🗽 🕴	- @ # # I	1E M	- ('			
t it X ∮ 7 · lž ► Skjed	E + Q Desab	ie X ia Restane	🐮 🥒 🗽 🕴	- @ # # I	E M				
i lt X ∮ V · lž ► Saled	E + Q Desab	ie X ia Restane	🐮 🥒 🗽 🕴	- @ # # I	E M	5 [
L B. X 5 7 · L2 F Salect Duratacled Dog Scient Dog	E + Q Desab	ie X ia Restane	🐮 🥒 🗽 🕴	- @ # # I	E M	5 			
L B. X 5 7 · L2 F Salect Duratacled Dog Scient Dog	E + Q Desab	ie X ia Restane	🐮 🥒 🗽 🕴	- @ # # I	E M				
to Bit Verm Provide Av to Bit Verm Provide Av Dispect	E + Q Desab	ie X ia Restane	🐮 🥒 🗽 🕴	- @ # # I	E M	•			
L B. X 5 T . L . Sulect Distances Dop Neury Just Contention	E + Q Desab	ie X ia Restane	🐮 🥒 🗽 🕴	- @ # # I	E M	•			
li li X ∮ 7 · lž ►	E + Q Desab	ie X ia Restane	🐮 🥒 🗽 🕴	- @ # # I	E M	•			
L B. X 5 T . L . Sulect Distances Dop Neury Just Contention	E + Q Desab	ie X ia Restane	🐮 🥒 🗽 🕴	- @ # # I	E M				
L B. X 5 T . L . Sulect Distances Dop Neury Just Contention	E + Q Desab	ie X ia Restane	🐮 🥒 🗽 🕴	- @ # # I	E M				
L B. X 5 T . L . Sulect Distances Dop Neury Just Contention	E + Q Desab	ie X ia Restane	🐮 🥒 🗽 🕴	- @ # # I	E M				
L B. X 5 7 · L2 F Salect Duratacled Dog Scient Dog	E + Q Desab	ie X ia Restane	🐮 🥒 🗽 🕴	- @ # # I	E M				
L B. X 5 7 · L2 F Salect Duratacled Dog Scient Dog	E + Q Desab	i ji X. ilg Net New Zee 1	E P K C	L · C · A · A · C · A	LE bi				
a fa X S Y A A	E + Q Desab	ie X ia Restane	E P K C	- @ # # I	LE bi				Qued
a a X S S T L S N L S S S S S S S S S S S S S S S S	E + Q Desab	12 X In No for No 1	E P K C	L • @ (P) (A) (E) (A) (A) (A) (A) (A) (A) (A) (A) (A) (A	LE bi		ees Jak		Store
a a X S S T L S N L S S S S S S S S S S S S S S S S		12 X In No for No 1	Conception effect of the serve losi the doubt of	L • @ (P) (A) (E) (A) (A) (A) (A) (A) (A) (A) (A) (A) (A	LE bi	9 mar			N laces
Intel Intel Intel		12 X In No for No 1	Conception effect of the serve losi the doubt of	L • @ (P) (A) (E) (A) (A) (A) (A) (A) (A) (A) (A) (A) (A	LE bi		vero 241	~	Yunin Q 1 1 4 3
Intel Intel Intel		12 X In No for No 1	Conception effect of the serve losi the doubt of	L • @ (P) (A) (E) (A) (A) (A) (A) (A) (A) (A) (A) (A) (A	LE bi		news_tark.	×	Qual No.
In bit 1 & 1 S - 1 2 State In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state <td></td> <td>12 X Iga Date Name Tree 1</td> <td>Server</td> <td>L • @ (P) (A) (E) (A) (A) (A) (A) (A) (A) (A) (A) (A) (A</td> <td>LE bi</td> <td>S man</td> <td>vert):74t</td> <td></td> <td>Yunin Qoʻto o</td>		12 X Iga Date Name Tree 1	Server	L • @ (P) (A) (E) (A) (A) (A) (A) (A) (A) (A) (A) (A) (A	LE bi	S man	vert):74t		Yunin Qoʻto o
A = A ≤ (3 + (2 + A)) A = A ≤ (3 + (2 + A))		12 X Iga Date Name Tree 1	Server	L • @ (P) (A) (E) (A) (A) (A) (A) (A) (A) (A) (A) (A) (A	LE bi		nems_fart.		Yunna Qool V a V
In bit 1 & 1 S - 1 2 State In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state In a state <td></td> <td>12 X Iga Date Name Tree 1</td> <td>Server</td> <td>L • @ (P) (A) (E) (A) (A) (A) (A) (A) (A) (A) (A) (A) (A</td> <td>LE bi</td> <td>9 mar</td> <td>tions_fait</td> <td><14.02 10.2275</td> <td>W.</td>		12 X Iga Date Name Tree 1	Server	L • @ (P) (A) (E) (A) (A) (A) (A) (A) (A) (A) (A) (A) (A	LE bi	9 mar	tions_fait	<14.02 10.2275	W.
A = A + G → A + S → A + A + S → A + S → A + A + S → A + A + A + A + A + A + A + A + A + A		12 X Iga Date Name Tree 1	Server	L • @ (P) (A) (E) (A) (A) (A) (A) (A) (A) (A) (A) (A) (A	LE bi	9 100	vent, fet		
A Second Se		i k k ia bestee http://www. http://www. a whose here:	E S L Constantino de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma del norma del norma de la del norma de la del norma de la del norma del norma del norma de la del norma del norma del norma del no	L • @ (P) (A) (E) (A) (A) (A) (A) (A) (A) (A) (A) (A) (A	12 M excellence excellence ===================================	19-04	vers; ?er	<14.02 10.2275	W.
Note: Second Sec		i k k ia bestee http://www. http://www. a whose here:	E S L Constantino de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma del norma del norma de la del norma de la del norma de la del norma del norma del norma de la del norma del norma del norma del no	 C (F) (F) (F) (F) (F) (F) (F) (F) (F) (F)	12 M excellence excellence ===================================	19-04	tore	<14.02 10.2275	-
IN ALL SUPERIOR STATES		i k k ia bestee http://www. http://www. a whose here:	E S L Constantino de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma del norma del norma de la del norma de la del norma de la del norma del norma del norma de la del norma del norma del norma del no	 C (F) (F) (F) (F) (F) (F) (F) (F) (F) (F)	(c) Life being the second s	19-04	neer feet Loose	<14.02 10.2275	W.
In a set of the set of		i k k ia bestee http://www. http://www. a whose here:	E S L Constantino de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma de la del norma del norma del norma de la del norma de la del norma de la del norma del norma del norma de la del norma del norma del norma del no		(c) Life being the second s	19-04	annes, det	<14.02 10.2275	-
IN ALL SUPERIOR STATES		i k k ia bestee http://www. http://www. a whose here:	E P Is A Compton The American American The American The American The American The American American The American American The American American The Americ		(c) Life being the second s	19-04	trace	<14.02 10.2275	-
IN ALL SUPERIOR STATES		IP N Spectrace Provide Deeps Provide Deeps Provide Deeps Provide Deeps Provide Deeps Provide Deeps Provide Deeps Provide Deeps Provide Deeps	E P Is A Compton The American American The American The American The American The American American The American American The American American The Americ		(c) Life being the second s	19-04	ren (st.	<14.02 10.2275	-
IN ALL SUPERIOR STATES			E P Is A Compton The American American The American The American The American The American American The American American The American American The Americ		(c) Life being the second s	19-04	ned; det	<14.02 10.2275	-
IN ALL SUPERIOR STATES		IP N Spectrace Provide Deeps Provide Deeps Provide Deeps Provide Deeps Provide Deeps Provide Deeps Provide Deeps Provide Deeps Provide Deeps	E P Is A Compton The American American The American The American The American The American American The American American The American American The Americ		s s s s s s s s s s s s s s	19-04	ener, fagr	<14.02 10.2275	W.
In a set of the set of		In the second se	E P Is A Compton The American American American The American American American The American The American American The American American The American The American American The American The American The American The American American The American		s s s s s s s s s s s s s s	19-04	Source Services	<14.02 10.2275	W.
		Note Department The Second Se	1 Server In Part of the Server In Part of the Server In Part of the Server In Server		n n n n n n n n n n n n n n		low	<14.02 10.2275	W.
In a k i general second		That See	E P Is A Compton The American American American The American American American The American The American American The American American The American The American American The American The American The American The American American The American		n n n n n n n n n n n n n n	Second Seco	torer dt	<14.02 10.2275	W.
In a k i general second		Third Steer Third Steer Third Steer Composition 1 Composition	1 Server In Part of the Server In Part of the Server In Part of the Server In Server		n n n n n n n n n n n n n n	- Senter Section 2014	torer dt	<14.02 10.2275	
In a k i general second		Third Steer Third Steer Third Steer Composition 1 Composition	1 Server In Part of the Server In Part of the Server In Part of the Server In Server		n n n n n n n n n n n n n n	Second Seco	torer alt Artical	<14.02 10.2275	

Librer Dare

ERCU lity Cer												🛩 🛃 sa
											ALIT > TILL	t Mercury Jours, Use
×	TextSets Edit View Texts	Analysis										
*	11 11 X 57.	🗿 Select Texts 🕨 Run	• IS Run Test Sal	× 57.	II # /	•					Test Ran Tree	Requirements Tree
20	Rost • 10 Undisched • 10 Indisched • 10 Ind	Execution Onld Execution	in Flow Test Set	Popertes Line	ed Defects						\$ 57	·
		t 1 Part Test Name	Plat Type	Setus	terstore	Parred Host Name	Responsible Tester	Ever Date	Time	Pier	Subject	ed.
5		75 (1)ogn	QUICKTEST_TEST	Pacced				1/05/2914	1.4t.3t PM	de la	E book to	kets
1 Plan											Nercury	
L	gi Cruite										Heldy	
tLeb	gi Flights gi Hone											
2	gi Hotes gi Register											
2	B Sign_an											
fects	B Vacatana											
fects												
fects										1		
fects										l		
fects												
fecti		¢								Þ		

The above figures show the functioning of the Quality Centre that runs in co-ordination with Quick Test Professional.

🔢 Run-Time Data Table	7.10.10.0	
Test Renation 1 (Row 1)	Test Results Summary	
	Test Test	
	Results name: Res!	
	Time Zone: India Standard Time	
	Run started: 1/30/2014-13:51:22	
	Run ended: 1/30/2014-135125	
	Iteration /	Results
	1	Done
	Status	Times
	Passed	0
	Failed	0
	Warnings	Q

The following result is generated.

Now the virtual users are provided using LoadRunner.

原原物的	stine 💌 🗚 Vuzers 💌					
0	Status	Scipt	Load Generator	Elapsed Time	^	
1.	Stopped	thine	locahost		10	\$ Bn
2	TError. Abnormal terror and on caused by indiv process terror ation	the	locahost			Sector Sec
3	Stopped	the	locahost			
41	Stopped	thine	locahort		11	\$ 90
5	L. Down	dee	locahost			44 Rest
6	La Down	2ee	locahost			44 ulber
7	& Down	the	locahost.		1	Briesh
0	& Down	shine	locahost			
9	& Down	thine	locahort			Help
10	â Down	shine	locahost		1	Quee
11	& Down	the	locehost			Non
12	L Down	shine	locahost			
13	In Down	shine	locaheut			
74	La Down	thre	locahost			
15	& Down	stine	locahost			
16	& Down	the	locahost			
17	& Down	shine	localhost			
18	in Down	thre	locahost			
19	A Down	shine	locahost			
20	A Down	shine	locahost			
21	1 Down	the	locahost			
22	& Down	shine	locahost			
23	L Down	shine	todeool			

Hence through this we can check out that how much users at a time can hit the application and how many couldn't. Therefore, this technique is much more precise and accurate for load testing than manual testing.

5. CONCLUSION

As in this research paper we have used both the techniques for the load testing i.e. through testers manually and through automated tools also where the virtual users are provided. Hence, we can see the difference between the results of both the techniques. As automated testing should be used in all the fields so as to check the applications. Because now most of the work is conducted through softwares or application be it in any field. So it should be checked that whether it is supporting the traffic that will be available in the future use.

REFERENCES

- [1] Zhen Ming Jiang, Automated Analysis of Load Testing Results, SAIL, School of Computing, Queen's University, Canada.
- [2] Z. M. Jiang, A. E. Hassan, G. Hamann, and P. Flora. Automatic identification of load testing problems. In Proceedings of the 24th IEEE International Conference on Software Maintenance, 2008.
- [3] Diplom-Ingenieur, George Din, A Performance Test Design Method and its Implementation Patterns for Multi-Services Systems, *Berlin 2009*.
- [4] Kartheek Muthyala, Rajshekhar Naidu P, A Novel Approach To Test Suite Reduction Using Data Mining, *Indian Journal of Computer Science and Engineering (IJCSE)*
- [5] Antonia Bertolino."Software testing Research: Achievements, challenges and dreams "Future of Software Engineering,2007.
- [6] Niclas Snellman, Automated Performance and Scalability Testing of Rich Internet Applications, 2012.
- [7] Timo-Pekka Heikkinen, Testing the Performance of a Commercial Active Network Measurement Platform, *Dissertation*, 2011.
- [8] Z. M. Jiang, A. E. Hassan, G. Hamann, and P. Flora. Automatic identification of load testing problems. *In Proceedings of the* 24th IEEE International Conference on Software Maintenance, 2008.
- [9] Z. M. Jiang, A. E. Hassan, G. Hamann, and P. Flora. Automated performance analysis of load tests. In Proceedings of the 25th IEEE International Conference on Software Maintenance (ICSM), 2009.
- [10] E. J. Weyuker and F. I. Vokolos. Experience with performance testing of software systems: Issues, an approach, and case study. *IEEE Trans. Software Engineering*, 26(12), 2000.