

# Time and Cost Effective Web Testing on AWS

Rajeshri Sonwane<sup>1</sup>, Prof. Manisha M. Mali<sup>2</sup>

<sup>1</sup>Product Analyst, Cognizant Technology Solutions, Hinjewadi, Pune-411057, Maharashtra, India

[rajashreesonawane26@gmail.com](mailto:rajashreesonawane26@gmail.com)

<sup>2</sup> PhD (Computer Science), Vishwakarma Institute of Information Technology, Kondhwa Budruk, Pune-411048, Maharashtra, India

[manissapmali@rediffmail.com](mailto:manissapmali@rediffmail.com)

**Abstract:** *In general, testing is one of the most important part of software development lifecycle (SDLC). Now a days, usage of internet is increasing rapidly, so it becomes necessity to test the websites. The web testing is more complicated as it involves different testing strategies and tools, because the tool has to support different characteristic properties and also different frameworks. So there is need to test websites as early as possible so that you will get require result in less time.*

*The main objective of our project is to develop a time and cost effective web testing environment using selenium grid and spot instances of AWS (Amazon Web Service) on cloud to reduce testing Time using selenium grid on multiple machines and to reduce testing Cost by use of lower cost spot instances of Amazon Web Services (AWS).*

**Keywords:** Amazon Web Services (AWS), Amazon Elastic Compute Cloud (Amazon EC2), Spot instance, Selenium grid

## 1. Introduction

The main purpose of the project is to do time and cost effective web testing on spot instances with using selenium grid Tool. We are creating an environment for acceptance and load testing of websites. Selenium grid transparently distribute our tests on multiple machines so that we can run our tests in parallel, cutting down the time required for running in-browser test suites. This will dramatically speeds up in-browser web testing, gives you quick and accurate feedback we can rely on to improve our web application and as well as lower cost of use of spot instances by reducing the time on Cloud Computing.

## 2. Research

### 2.1 Cloud Computing

Cloud Computing is a technology which you can access and use for your work. You can run all the applications that you want to use without installing software, updating files with using internet. It also

uses central remote servers to maintain data and applications. This technology allows customer and business to run their applications efficiently.

A cloud has two types Private and Public. A Public cloud is used to give services to the all users which uses internet. A Private cloud is used to give the services to limited number of people. The main purpose of cloud is to give easy access of technology resources and services.

These are the types of Public Cloud:

- Infrastructure as a service
- Platform as a service
- Software as a service
- Storage as a service
- Security as a service
- Data as a service
- Test environment as a service
- Desktop as a service
- API as a service

Cloud provides three services as follows:

- Amazon Web Services
- Windows Azure
- Sales force

## 2.2 Amazon Web Services (AWS)

AWS is a service of cloud which provides several Web services. AWS gives various infrastructure services with minimum cost as customer can use it for his benefits. AWS focuses only on basic building block services which will fulfill some core needs of most systems i.e. storage, computing, messaging and datasets. AWS provides facility to host your application on cloud.

### **Benefits of AWS:**

- *Easy to use:* AWS allows customers quickly and securely host their applications. So it is easy to use.
- *Open & Flexible:* AWS is a flexible because it will provide facility that to select the operating system, programming language, web application platform, database, and other services which customer need. Also it will give the virtual environment for customer application which will avoid migration process
- *Cost-Effective:* AWS provides facility to customer to use it with minimum cost and pay for resources use. It will provide multiple pricing models to optimize low cost.
- *Reliable:* AWS provide the reliable source of information.
- *Scalable and high-performance:* AWS provides scalable and high performance tool for customers.
- *Secure:* AWS provides the secure approach to customer which is very important factor.

Amazon Web Service provides two main components:

- Amazon Elastic Compute Cloud (Amazon EC2)
- Amazon Simple Storage Service (Amazon S3)

## 2.3 Amazon ELASTIC COMPUTE CLOUD (AMAZON EC2)

Amazon Elastic Compute Cloud (Amazon EC2) is a component of AWS which will provide web service. Amazon EC2 provides customer various types of instances to purchase which will give easy access at low cost. It will provide complete control of computing resources and run on amazons computing environment. Also it will help to provide instance as soon as possible. Amazon EC2 also provides high security mechanism. Some of them are Virtual Private Cloud, API, Subnets Route, Tables and Routes, VPN Gateway, Internet Gateway, Tenancy, Firewall (Security Groups) etc.... It can serve as set of virtual machines.

AWS has there three Instance types:

### 2.3.1 Reserved Instances –

Reserved Instances are high priority instances in Amazon EC2. This type provides us the instances at as low as possible cost and also gives facility of one-time payment.

There are three types of Reserved Instances:

- a) Light
- b) Medium
- c) Heavy

It also gives us hourly discount for instances for various prices. We can reserve these instances up to 3 years. It will save your money up to 71%. Also these are available on all AWS regions. Reserved Instances are easy to use and reliable.

### 2.3.2 On - Demand Instances –

On-Demand Instances are second priority instances in Amazon EC2. This type provides the facility that pay as per use. These instances are purchased for hours. This will frees you from complexities of planning, purchasing and maintaining hardware. It will give smaller cost. It will also give the facility for handling periodic traffic spikes.

### 2.3.3 Spot Instances –

Spot Instances are the low priority instances in Amazon EC2. This allows using Amazon in very low cost. These are very cheap as compare

to Reserved Instances and On-Demand Instances. But one disadvantage of these instances is that it can terminate when hourly price goes high as compare to that of customers price. So that in some cases customer can lose data on which he/she is working. But also it provides facilities for storing the data as instances gets terminate.

## 2.4 Spot INSTANCES

Spot Instance is one of the types of instances provided by the AWS. These instances have very low price as compared to On-demand Instances and Reserved Instances. If customer wants to use the unused spot instances then he/she has to bid for those instances. AWS allows customers to bid on unused Amazon EC2 capacity and run those instances until their bid exceeds the current Spot Price. The Spot Price changes periodically based on supply and demand and customers whose bids exceed the Spot price gain access to the available Spot Instances. Spot Instances are different from On-Demand Instances and Reserved Instances. It provides another option for obtaining compute capacity. In this way the customer can bid the spot instance and get the access of that Spot instances.

### Working of Spot Instances

Customer can bid unused Amazon EC2 capacity Spot Instances. Spot Price is set by Amazon EC2 and it fluctuates periodically depending on the supply of and demands for Spot Instance capacity. To use Spot Instances, customer has to place a Spot Instance request, specifying the information about the instance type, the Availability Zone desired, the number of Spot Instances you want to run, and the maximum price you are willing to pay per instance hour.

Amazon provides a feature called History, by which anyone can get the information about any spot instances via the Amazon EC2 API and the AWS Management Console. The information includes the past 90 days History of those spot prices. With the help of these information customer can determine how that maximum price compares to past Spot Prices. If the current Spot Price exceeds your maximum bid price, your request is fulfilled and you

will get access to that instances. You can run these instances until either Spot Price increases above your maximum price or you choose to terminate them (whichever is sooner). In such a manner, the Spot Instances works.

### Deciding on Your Spot Bidding Strategy

- As the Spot Price changes periodically depending on demand and supply, you will often pay less per hour than your maximum bid price. Everyone pays that same Spot Price for that period regardless of whether their maximum bid price was higher. You will never pay more than your maximum bid price per hour. If your maximum bid price is greater than the current spot price then you have to pay that same Spot Price for that period.
- If your maximum price no longer exceeds the current Spot Price, your instances will terminate automatically. This is the main disadvantages of Spot Instances. In this case, if it's important for you to run Spot Instances uninterrupted for a period of time, it's advisable to submit a higher maximum bid price. As Spot Instances terminated automatically, you have to store your information.
- The pricing of Spot Instances is similar to that of On-demand Instances and Reserved Instances. You can terminate the Spot Instances when you no longer need them. But if your Spot Instance gets terminated automatically by Amazon EC2, you will not be charged for any partial hour of usage. If you choose to terminate your instance, you have to pay for that partial hour as you do for On-Demand or Reserved Instances.

## 2.5 Selenium GRID

### *Why selenium?*

- In-Browser Tests Are Slow
- Maintenance Costs Are the Real Challenge
- Prompt and Reliable Feedback is Crucial
- Hardware is Cheap

Selenium Grid is a web testing tool that transparently distributes your tests on multiple machines so that you can run your tests in parallel. As test cases runs parallel, the time required for running test suites in-browser will be reduce. This will speed up in-browser web testing and will give you quick and accurate feedback. Selenium Grid cuts down on the time required to run a Selenium test suite to a fraction of the time that a single instance of Selenium instance would take to run. In this way, you can improve your web application.

### Working of Selenium Grid

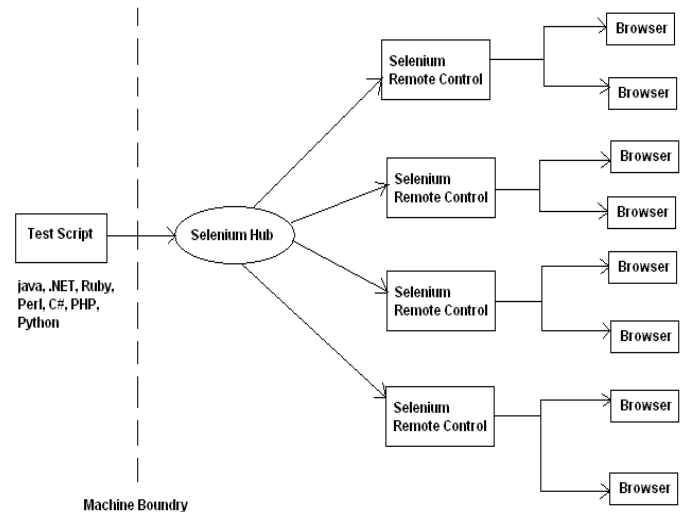
Selenium Grid is a Web testing tool that dramatically speeds up functional testing of web-applications. It allows you to run multiple tests in parallel, on multiple machines. The working of the Selenium is as shown in figure.

- **Selenium Hub** – The Hub is an external interface to the Test Scripts. When a new test starts, the Hub allocates Selenium Remote Controls to each test. If there is no available Remote Control in the grid providing the appropriate capabilities, the Hub puts its first request on hold. The Hub is responsible for routing the Selenese requests from the tests to the appropriate Remote Control as well as keeping track of testing sessions. As soon as a suitable Remote Control becomes available, the Hub will serve the request

The Selenium Hub takes on these responsibilities:

- Allocating a Selenium Remote Control to a specific test (transparently).
  - Limiting the number of concurrent test runs on each Remote Control.
  - Shielding the tests from the actual grid infrastructure.
- **Selenium Remote Control**- Selenium Remote Control runs the tests in browsers and the communication is done through the HTTP protocols.

- **Test Script**- First of all, the user should write the test script. Test Script can be written by any of these languages: Ruby, Java, Python, C#, PHP. Then the test script is passed to the Selenium Remote Control. Then the tests do not have to be aware of what is happening within the grid; it is just waiting for an HTTP response to come back.



**Figure 1:** Working of Selenium Grid

Selenium Grid allows you to run multiple instances of Selenium Remote Control in parallel. Selenium Grid is easy to use.

### 3. STUDIES AND FINDINGS

Based on research and studies, we found that, Selenium Grid/Tool transparently distributes test cases among the multiple machines. So the time required for web testing is reduced. The second important part of our project is AWS Cloud. Amazon cloud provides several Web services. EC2 is a component of AWS which will provide web service. Amazon EC2 provides customer various types of instances to purchase which will give easy access at low cost. Hence, we combine these two concepts to produce a tool which will test the web sites in minimum time and cost. Thus web testing can be carried out in low cost and minimum time.

We found that Web testing is done with Selenium Grid on Reserved Instances of EC2. But we are using the Spot Instances in our project just to reduce cost of testing as Spot Instances has low cost as compared to On-demand and Reserved Instances.



Though we are using Spot Instances, There are disadvantages of Spot Instances like it terminates automatically when bidding price exceeds your maximum price. Then in this case, we are going to use various techniques like splitting your work into small increments (via Grid, Hadoop-based, or queue-based architectures) or adding checkpoints to you application and saving your work frequently or by backing up data to Amazon Relational Database Service (Amazon RDS) every ten seconds while using Spot Instances. Using Amazon EBS volumes to store your data is one easy way to protect your data.

In future, there will be demand to lower prize Spot Instances which will surely give benefits in our work.

#### 4. CONCLUSION

Thus by implementing Selenium Grid on Spot Instances, we can reduce the time and cost of our testing on AWS. This is efficient method as it requires low cost and time for Web testing. So it will be very useful in future because of low cost and time.

#### 5. ACKNOWLEDGMENT

We would like to express my deep gratitude to **Prof. M. M. Mali** who has always been a guiding force behind this project work. Her highly influential personality has provided me constant encouragement to tackle any difficult task assigned. I am indebted to her for his invaluable advice, unstinting support and helping me prepare the project paper.

I would also like to thank all my friends and especially my classmates for all the thought provoking discussions we had, which inspired me to think beyond the obvious.

#### References

- [1] "AWS\_Cloud\_Best\_Practices.pdf"  
<http://d36cz9buwru1tt.cloudfront.net/>
- [2] Amazon Elastic Compute Cloud (EC2),  
<http://www.amazon.com/ec2/>
- [3] "AWS\_Storage\_Options.pdf"  
<http://media.amazonwebservices.com/>
- [4] Prasanth Yalla, Dr. L S S Reddy, M.Srinivas, T.Subha MastanRao, "Framework for Testing

Web Applications using Selenium Testing tool with respect to Integration Testing", IJCST Vol. 2, Issue 3, September 2011

- [5] Zhen Li, Yong Hu Sun, "Use Selenium Grid to enhance testing of web applications", 07 Jun 2011  
<http://www.ibm.com/developerworks/web/library/wa-seleniumgrid/index.html>
- [6] Mladen A. Vouk, "Cloud Computing – Issues, Research and Implementations " Received: June, 2008. Accepted: September, 2008
- [7] "computing-whitepaper.pdf"  
<http://www.thinkgrid.com/docs/>
- [8] Selenium-Grid,  
<http://selenium-grid.seleniumhq.org/>
- [9] Geoffrey Wiseman, "Selenium Grid: Web Testing in Parallel" Sep 26, 2007  
<http://www.infoq.com/news>
- [10] David Burford, "CLOUD COMPUTING: A BRIEF INTRODUCTION" Lad Enterprises, Inc. February 20, 2010
- [11] "Cloud Platforms--Chappell.pdf"  
[www.davidchappell.com/](http://www.davidchappell.com/)

#### Author Profile



**Rajeshri Sonwane** received the B.E. degree in Computer Engineering from Vishwakarma Institute of Information Technology (affiliated by Pune University) in 2013.

She is now working as a Product Analyst in Cognizant Technology Solutions, Pune.