Volume 6, Issue 4, April 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Cloud Testing Techniques and Its Challenges

Manika Bisht*, Tanya Rastogi, Prof. Manjula R
SCOPE, VIT University, Vellore,
Tamilnadu, India

Abstract*—The next stage of Internet evolution is Cloud Computing. It provides an opportunity in offering testing as a service (TaaS) for SaaS and clouds. It is a type of computing that depends on sharing computing resources to handle applications. Software Testing is a process which is used to find out software bugs while executing a program or application. Before coming into service the cloud must be tested to provide effective cloud services and sharing of resources. Cloud testing is a type of testing in which web applications utilizes cloud computing environment and infrastructure to simulate real world user traffic by applying cloud technologies and solutions. In this paper the authors have described various testing techniques in cloud based environments. The testing techniques are divided into three major categories that are functional, non-functional and ability cloud testing. These techniques can be further subcategorized as their testing areas. The paper also brings forth the various challenges faced while performing the testing techniques.

***Keywords*— Cloud Computing, Cloud Testing, Testing Techniques, Cloud Computing Challenges, Cloud Based Environment**

I. INTRODUCTION

Cloud computing is a kind of Internet-based computing which is also referred as 'on-demand computing' where shared resources, information and data are delivered to computers and other devices on-demand. Cloud computing has earned major attention in recent years as it changes the method of computation and delivers the services to the customers whenever and wherever needed.Cloud Testing refers to the validation and verification of infrastructure, environments and application which are available on demand by conforming them to the cloud computing business model expectations. Testing a Cloud includes performance, security, availability, multi-tenancy testing, disaster recovery and interoperability. Cloud testing is challenged by several problems such as meeting deadlines, limited test budget, large number of test cases, high costs per test, geographical distribution of users and little reuse of tests. Cloud testing aims at ensuring delivery of high quality service and avoiding data outages which requires testing outside the datacentre or inside datacentre or in both place. [10]

Services provided by cloud testing: [14]

- Test Environment Management
- TaaS Management
- On-Demand Testing
- Test Management
- Performance & Scalability Testing

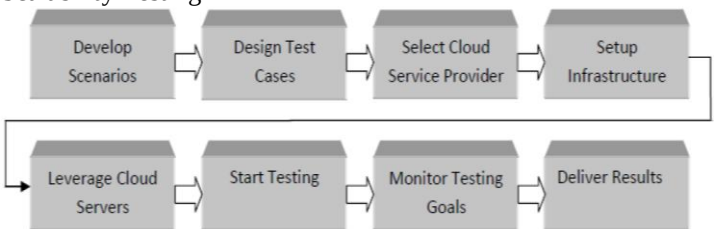

Fig1. Cloud Testing Life Cycle [8]

II. TESTING TYPES

A. Functional Testing

1) System Testing: In order to evaluate the system's compliance with its specified requirements, System testing of hardware or software is conducted on an integrated and complete system. It falls under the scope of black box testing, and requires no knowledge of the inner code or logic design. The input of system testing are all of the "integrated" software components which have passed integration testing and also the software system itself embodied with any applicable hardware system(s). It is performed on the entire system in the context of a System Requirement Specification(s) (SRS) and/or a Functional Requirement Specification (FRS). It not only tests the design, but also the customer's behaviour and believed expectations. System testing is also bound to test up to and beyond the boundaries that are defined in the software and hardware requirements specification(s). It's main focus is to evaluate Functional, Business and End-user requirements.

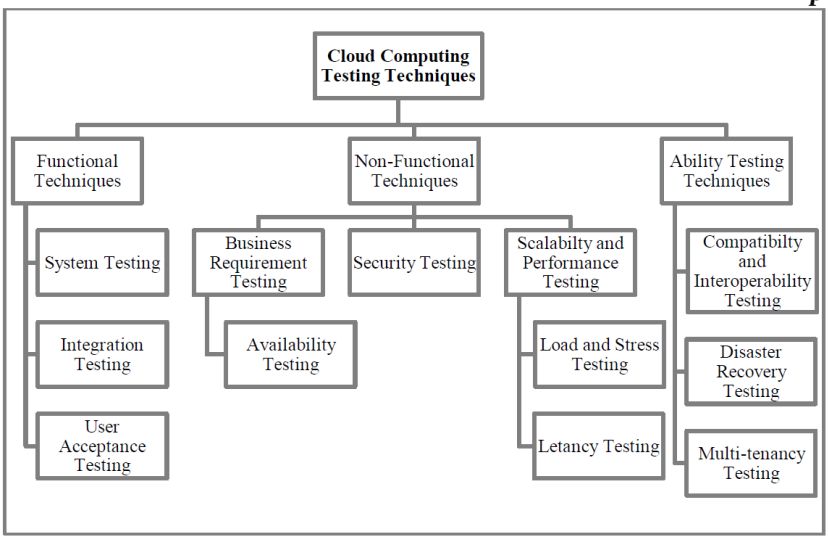

Fig2. Cloud Testing Techniques [10]

2) Integration Testing: Integration testing is the technique in which each software module is tested as a group. When it comes to overall business strategy, this testing technique fits the cloud computing system. [11] It helps the business to verify if the cloud solution will work within the current infrastructure and environments which will ultimately prove that the cloud solution implementation does not have any harmful impacts on any existing systems. Finally, the business requirements need to be validated and verified to justify that the end result of the cloud solution meets the required needs of the business [8].

Approaches of Integration Testing are:

Big Bang is an approach to Integration Testing in which most of the units are united together and tested all at once. When the testing team receives the complete software in a bundle, this approach is applied. The Big Bang testing tests only the interactions among the units while the system testing tests the complete system.

Top Down is an approach to Integration Testing. As the name suggests it is a step by step approach in which firstly, the top level units are tested and then the lower level units. When top down development approach is followed, then apply top down approach.

Bottom Up is an approach to Integration Testing. As the name suggests it is also a step by step approach in which bottom level units are tested first and then upper level units after that. When bottom up development approach is followed, then apply this approach.

Sandwich/Hybrid is an approach to Integration Testing which is a combination of Bottom Up and Top Down approaches.

3) User Acceptance Testing: User Acceptance Testing will be done to prove that delivered cloud solution meets business requirements so that the user accepts the developed cloud solution. User acceptance testing is done on both on premise and off-premise. However, the onsite testing allows immediate monitoring and control of test progress [9].

Types of User Acceptance Testing are:

Alpha Testing is used in the development environment by internal staff much before the product is released to external testers or customers. It can also be done by some capable user groups. To rectify certain issues or bugs and improve the reliability of the product the feedback collected from alpha testers is used.

Beta Testing is done in the customer's environment by a set of customers who use the system in their environment. It is also known as "field testing". The feedback provided by the beta testers helps in improving the product.

In Contract Acceptance Testing the software that developed is tested against certain specifications and criteria which are predefined and agreed upon in a contract.

Regulation Acceptance Testing checks whether the software complies with the legal and governmental regulations. It is also known as Compliance Acceptance Testing.

Operational acceptance testing makes sure that all the workflows is in place known to allow software to be used. It is also known as Operational Readiness Testing.

Black Box Testing analyses certain functions and keeps the internal code invisible to the tester. It is the most commonly used user acceptance technique.

B. Non Functional Testing

1) Business Requirement Testing: Requirements-based testing is an approach where data, conditions and test cases are derived from the requirements. This includes functional tests and also non-functional attributes like usability, performance and reliability.

The testing process includes:

- It should be carried out in a timely manner.
- It should add value to the software life cycle, and thus it must be effective.
- Exhaustive testing of the system is impossible, thus the testing process should be efficient as well.
- Testing should provide the overall project status, hence it needs to be manageable.

Business requirements should be delivered to provide value. Systems, products, processes, and software are the ways of how to satisfy, meet, or deliver the business requirements. Many times the business requirements topic comes up in the context of developing and procuring cloud environment, but business requirements occur much more broadly. Business can be at personal or work, for non-profit or profit.

2) Security Testing:Security testing is an indispensable part of testing applications because of increase in security breaches in business. This can provide assurance that business critical data is stored and transported safely. [10] In a cloud environment, network security is most important. Many security appliances are in rampant use that protect data centres and enterprises. These devices encourage the roles of anti-virus, data-loss prevention, firewalls, intrusion prevention systems and anti-spam.[11] It also targets at verifying six basic principles Authorization, Availability, Confidentiality,Authentication, Integrity and Non-repudiation.

3) Scalability and Performance Testing: One of the simplest forms of performance testing is Load Testing. A load test is usually conducted to understand the system behaviour under a specific expected load. This load can be the expected, concurrent number of application users who are performing a specific number of transactions within the set duration. This test provides the response times of all the important critical business transactions. Load testing is verifying that the cloud system can operate at the required response times when it is subjected to its expected load. While the system is under load, failover test which is a type of non- functional testing should be undergone. These failover tests are done under peak anticipated load to determine how the cloud service reacts to such a condition.

Stress Testing means putting the application under extreme load for a long period of time. We want to place the application under higher load than what it is expected to encounter in the real world to see what breaks it down when pushed to the edge. This provides clues on where the environment may not be as robust as it should be, and what users may see under some unusual conditions. Running a heavy load for many days makes it possible to more quickly identify problems such as connection, memory, file handle, or any other kind of resource leaks, which can cause premature application failures. The goals of such tests may be to ensure the software does not crash in conditions of insufficient computational resources (such as disk space or memory).

Latency is defined as the delay from input into a system to the desired outcome. The term holds different meaning in different contexts and latency issues may also vary from one system to another. Cloud testing is used to measure the latency between the action and its corresponding response for any application after it has been deployed on the cloud.

C. Ability Testing

1) Compatibility and Interoperability Testing: Compatibility testing validates how a system under test works within a particular environment. For example, a large amount of research has been done for understanding whether devices used in heart transplants are compatible with the user's body. In software development, compatibility testing tests how applications work within different browsers or operating system. Interoperability testing validates how a system under test works when interacting with some other thing. For example, major surveys have been conducted to conclude how devices used for heart transplants interact with numerous microwave technologies. In software development, interoperability testing tests how applications works with other services or applications.

*2) Disaster Recovery Testing:*For any organization, disasters are an inevitable certainty but while inevitable, they are also unpredictable. Cloud service provider wishes that its cloud services should be available to users all throughout. Disaster recovery time must be low after some failure occurs. [5] This is a process of testing the success of the restoration procedures that are executed after some major failure or disruption occurs. Sometimes the combination of this type of testing with load testing is carried out. Disaster Recovery Testing is much more genuine if the tests are carried out while the application is busy servicing a user workload.

3) Multi-tenancy Testing: Multi-tenancy is the key attribute in both public and private clouds and it applies to all three layers of cloud: Infrastructure-as-a-service(IaaS), Software-as-a-Service(SaaS) and Platform-as-a-Service(PaaS). Multi-tenancy testing refers to a software architecture in which a software's single instance runs on a server and serves multiple tenants. A tenant is a group of users sharing a common access with some specific privileges to the software instances. With this testing, cloud environment aims at providing a dedicated share of the instance to every tenant including tenant individual functionality, data, user management, configuration and non-functional properties.

III. CHALLENGES

A. Service Challenge:

Cloud service challenges is the fundamental challenge in cloud computing testing environment. The first challenge in this category is service availability without experiencing unwarranted delays as user organization is willing to adopt cloud services rather than maintain local installations. [5],[6]. Cloud Services must feel and look as local services and not as remote services. The second challenge is the cloud service assurance. The final challenge is service efficiency. [5],[6].

B. Security Challenge:

Security in the public cloud is a major issue. When multiple organizations share resources there is a risk of data misuse. In order to avoid risk, it is mandatory to secure data repositories and also the data that involves storage, transit or process. In cloud computing, one of the most important challenges is protection of data. To enhance the security in cloud computing, it is important to provide access control, authorization and authentication for the data stored in cloud.

C. Layered Testing Challenge:

The third major challenge category is the testing at multiple layers of the cloud environment. Layers testing Protocol is the first challenge in this. Here, the testing network connection, database, server performance and software application adds multiple layers to cloud testing. [4],[7].Second challenge in this is maintaining the communication between layers. Testers test the communication between the layers, test connection between the elements and also plan for the risks such as server down, connection breaks midway and software crashes [1],[4].

D. Lack of Universal Standard and Infrastructure:

Lack of universal standards is the first challenge in this. Currently, there are no proper universal standard solutions to integrate user company's internal datacentre resources with public cloud resources. Public cloud providers have their own pricing mechanisms, operating models and architecture that offer very little interoperability. As a result, when companies need to switch their vendors, it leads to a big challenge for them. The second challenge in this is the limited infrastructure. Some cloud providers offer only limited types of servers, technology, configurations, bandwidth, networking and storage which makes creating real-time test environments difficult. [1],[3]

E. Guidance, Knowledge and Staff Expertise:

The fifth challenge category is obtaining knowledge, guidance and staff expertise. Obtaining guidance for testing cloud service is the first challenge encountered in this category. The federal guidance that currently exists to use cloud services is incomplete or insufficient. The second challenge is taking expertise teaching staff. Service provider may not be able to provide necessary tools or resources such as expertise staff to implement cloud solutions. Teaching an entirely new set of tools and processes such as monitoring performance in a cloud environment to the staff has been a challenge. Another challenge is acquiring direct knowledge. The delivery of cloud services without any direct knowledge of the tools and technologies has been a challenge [3],[7].

F. Procuring Cloud Service on-Demand Basis:

The first challenge in this is the delivering on-demand services with specific quantity and costs. The on-demand and scalable nature of cloud services can be difficult to define specific quantities and costs. These uncertainties make contracting and budgeting challenge because of the fluctuating costs associated with scalable and incremental cloud service procurements. The second challenge is the dependency on remote installed applications. Since applications are not installed locally in controlled environments, this makes it harder for testers to replicate the user environment. The third challenge is the increasing expenditure on encrypted data. Improper usage of cloud based test environments can increase costs [1],[3],[4].

IV. CONCLUSIONS

Cloud testing is the most recent research topic among new researchers nowadays. With the advancement of testing as services and cloud technology, more research work must be done to address the open challenges and issues in cloud testing. This paper gives an overview of the various testing techniques and the challenges encountered in the cloud environment. Functional testing requires high usage of software and hardware to simulate user activity.While non-functional testing allows the association and measurement of the testing of non-functional attributes of software systems. Very few of the testing challenges in cloud environment have been observed and with the growing developments in each of these techniques, we certainly cannot state one to be better than the other, as each testing technique has its own set of benefits and limitations.

REFERENCES

[1] Laurin H. Mills (2009), "Legal Issues Associated with Cloud Computing".

[2] R. Velumadhava Rao, K. Selvamani, Data Security Challenges and Its Solutions in Cloud Computing, International Conference on Intelligent Computing, Communication & Convergence, ICCC-2105, Procedia Computer Science 48 (2015) 204 – 209 .

[3] Pat Hyek (2011)," Cloud Computing Issues and Impacts", Global Technology Industry Discussion Series.

[4] Priyanka, Inderveer Chana & Ajay Rana (2012), "Empirical Evaluation of Cloud-based Testing Techniques: A Systematic Review", ACM SIGSOFT Software Engineering Notes, Vol. 37, No. 3, Pp. 1–9.

[5] IXIA (2011), "Testing the Cloud: Definitions, Requirements and Solutions", August 2011, CA, URL: http://www.ixiacom.com/.

[6] S. Ravichandran (2012), "Innovative Method of Software Testing Environment using Cloud Computing Technology", International Journal of Communication and Networking System, Vol. 01, No. 01.

[7] Swapnil H. Chandane& Prof. Mahip M. Bartere (2013), "New Computing Paradigm: Software Testing in Cloud, Issues, Challenges and Need of Cloud Testing in Today's World", International Journal of Emerging Research in Management & Technology.

[8] A. Vanitha Katherine &Dr.K. Alagarsamy (2012), "Conventional Software Testing Vs. Cloud Testing", International Journal of Scientific & Engineering Research, Vol. 3, No. 9.

[9] Keerti Kulkarni, "Testing Services through Cloud", A White Paper, MicroFocus.

[10] Dr. Rahul Malhotra and Prince Jain, Testing Techniques and its Challenges in a Cloud Computing Environment, The SIJ Transactions on Computer Science Engineering & its Applications (CSEA), Vol. 1, No. 3, July-August 2013.

[11] Akash Shrivastva, Shubham Gupta and Rinki Tiwari, Cloud Based Testing Techniques(CTT), International Journal of Computer Applications (0975 – 8887) Volume 104 – No.5, October 2014.

[12] Spirent Communications (2010), "Cloud Computing Testing", Spirent Communications, California.

[13] EljonaProko& Ilia Ninka (2012), "Analysis and Strategy for the Performance Testing in Cloud Computing", Global Journal of Computer Science and Technology Cloud & Distributed, Vol. 12, No. 10, Version 1.0, USA.

[14] A.K. Srivastava, D. K. Yadav & S. K. Pandey, "TaaS: An Evolution of Testing Services using Cloud Computing," International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), vol. 1, no. 10, pp. 42-49, 2012.

