International Journal of Electrical and Computer Engineering (IJECE)
Vol. 9, No. 2, April 2019, pp. 1122~1130
ISSN: 2088-8708, DOI: 10.11591/ijece.v9i2.pp1122-1130

Proposed T-Model to cover 4S quality metrics based on empirical study of root cause of software failures

Dheeraj Chhillar, Kalpana Sharma
Department of Computer Science and Engineering, Bhagwant University, Ajmer, Rajasthan, India

Article Info

Article history:

Received Jan 7, 2018
Revised Sep 20, 2018
Accepted Oct 14, 2018

Keywords:

Performance testing
Recent software failures
Safety, scalability, stability and serviceability (4S)
Security testing
Survey
T-model

ABSTRACT

There are various root causes of software failures. Few years ago, software used to fail mainly due to functionality related bugs. That used to happen due to requirement misunderstanding, code issues and lack of functional testing. A lot of work has been done in past on this and software engineering has matured over time, due to which software's hardly fail due to functionality related bugs. To understand the most recent failures, we had to understand the recent software development methodologies and technologies. In this paper we have discussed background of technologies and testing progression over time. A survey of more than 50 senior IT professionals was done to understand root cause of their software project failures. It was found that most of the softwares fail due to lack of testing of non-functional parameters these days. A lot of research was also done to find most recent and most severe software failures. Our study reveals that main reason of software failures these days is lack of testing of non-functional requirements. Security and Performance parameters mainly constitute non-functional requirements of software. It has become more challenging these days due to lots of development in the field of new technologies like Internet of things (IoT), Cloud of things (CoT), Artificial Intelligence, Machine learning, robotics and excessive use of mobile and technology in everything by masses. Finally, we proposed a software development model called as T-model to ensure breadth and depth of software is considered while designing and testing of software.

Corresponding Author:

Dheeraj Chhillar,
Department of Computer Science and Engineering,
Bhgwant University,
Sikar Road, Ajmer, Rajasthan, 305004-India.
Email: Dheeraj.chhillar@gmail.com

1. INTRODUCTION

This paper is to review the critical software failures happened recently along with root cause of their failures. Software applications are being used excessively by people in all businesses as well as in their day to day activities. Automation is a necessity if consistency in services is required or business needs to be scaled up. New challenges are faced by users as internet and digitization has penetrated at all levels of society and is being used by masses. These days handling scale of usage of application and safety of personal data is of utmost importance unlike in past where having required functionality of the software was a challenge. The research is conducted to understand evolution of technology in context of software testing, so that appropriate research could be done to mitigate risks of failures and improve quality of the software applications. Nowadays development methodologies have switched from traditional SDLC models to agile methodologies where customers feedback is taken at every point to minimize risk of failure. As software designing, development and functional testing matured over time, these days failure mainly happens due to performance issues and security reasons. Moreover, most of the recent software development uses mobile and cloud systems, which makes them more vulnerable to failure due to security and performance issues.

Performance and Security parameters mainly constitute the non-functional requirements. A software system's utility is determined by both its functionality and non-functional characteristics, such as usability, flexibility, performance, interoperability and security [1]. Customers can just ensure conformance to functional requirements by verifying functionality and checking look and feel of software. Testing non-functional requirements is a difficult task because it require use of tools and more technical knowledge. Based on our research we have proposed a T model to design and test a software to ensure full test coverage of software application. T-model ensures that breadth and depth of software application is being considered while designing the software and same is being tested. Breadth and Depth parameters are explained in detail in this paper.

2. SHIFTING OF TECHNOLOGY

Decades ago software were mainly window based and people used to work on desktops in isolation. Then in last decade most of the applications switched to web based due to substantial increase in use of internet. Shifting of technology is explained in Figure 1. In last few years due to increase in use of smart cell phones people like to access everything from their cellphone or palmtop. Each and every web-based business or application has started switching to mobile applications. People access everything from mobile phone be it e-commerce, banking, hospitals, shopping, education, search services etc. These days' software architectures are more flexible and interact a lot with outside applications. Architectures involve use of cloud to add more scalability and flexibility. All code resides in cloud and anyone can access and code from anywhere in the world. Applications involve micro services, rest services, web services to interact with other applications. Latest research involves internet of things, smart devices, robotics, artificial intelligence, big data and machine learning. A new scientific paradigm is born data intensive scientific discovery (DISD), also known as big data problems. A large number of fields and sectors, ranging from economic and business activities to public administration, from national security to scientific researches in many areas, involve with big data problems [2]. Now internet, software and applications are tied to each and every individual directly or indirectly. Every individual's personal, family, banking, private data is exposed on web or internet due to excessive use of technology in everything they do. Now software has become integral part of each and every individual. Data privacy has become the biggest concern in today's world. Cost of data has become way more than hardware and software. Cybercrime & data theft is increasing exponentially. Most of the focus these days is on software development and technology enhancement whereas security measures are majorly compromised.

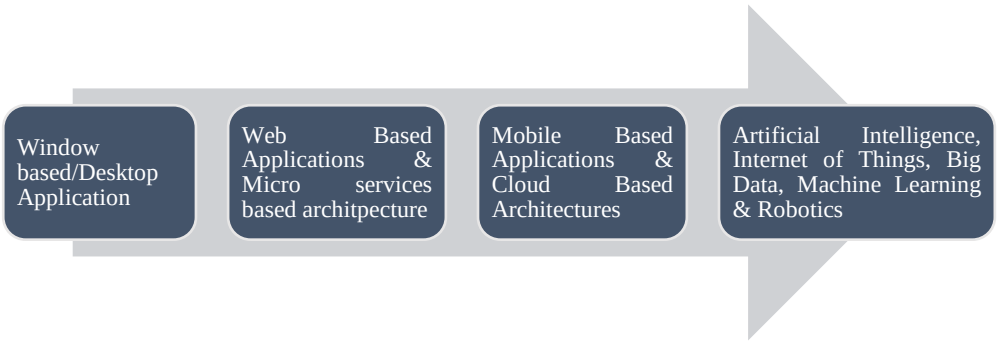

Figure 1. Shifting of technology

The trend of IoT in our modern society is explained in a recent report from Gartner, which estimated that 63 million IoT devices will be attempting to connect to the network each second by 2020 [3]. Internet of things systems data is stored in cloud which is accessed by users. This is called as cloud of things (CoT). In Cloud of things data of all internets of things is uploaded to cloud and from their data is accessed by users using mobile and desktops. This is shown in Figure 2 in the paper.

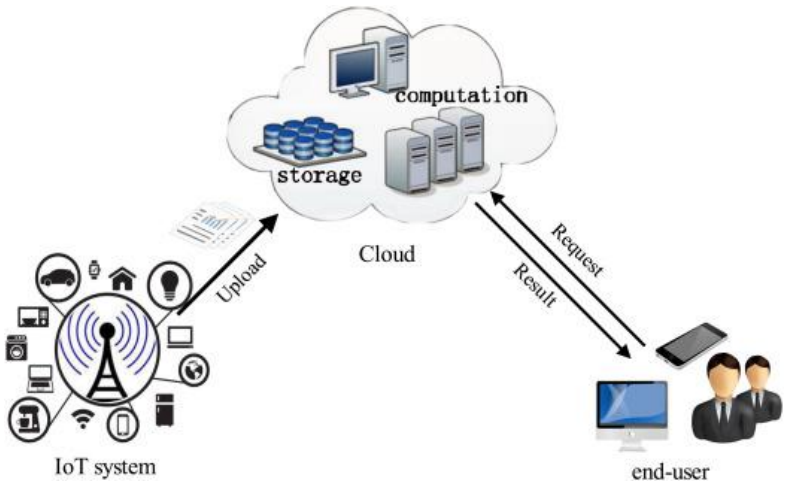

Figure 2. A typical cloud of things (CoT) system [4]

At the time when internet was not much used, mainly standalone applications or window-based applications were developed. At that time unit testing, component testing, integration testing, system testing, black box and white box testing were mainly performed. Then when internet became prominent and web-based applications were developed, main focus was on functional testing and some level of performance testing also started. In past around 2 decades functional testing became mature over time and nowadays there is hardly any software that fails due to lack of functional testing. A lot work was done in functional automation testing as well. In current scenario where machine learning, internet of things, big data, artificial learning & robotics is being used main concern has shifted to security testing. With the development of cloud computing more and more enterprises host and deploy their servers and systems in the cloud and deploy/migrate to the public cloud environment. In order to protect the security of the cloud computing information system, the information security products need to be deployed in cloud computing environment. Due to the characteristics of the cloud computing environment, the traditional information security products cannot meet the security needs of cloud computing environment. In recent years, information security product manufacturers and cloud computing service providers are committed to research and development of the new cloud computing information security products [5]. Focus of testing has shifted from black box to white box testing and to performance and security testing and data integrity. This is shows in Figure 3.

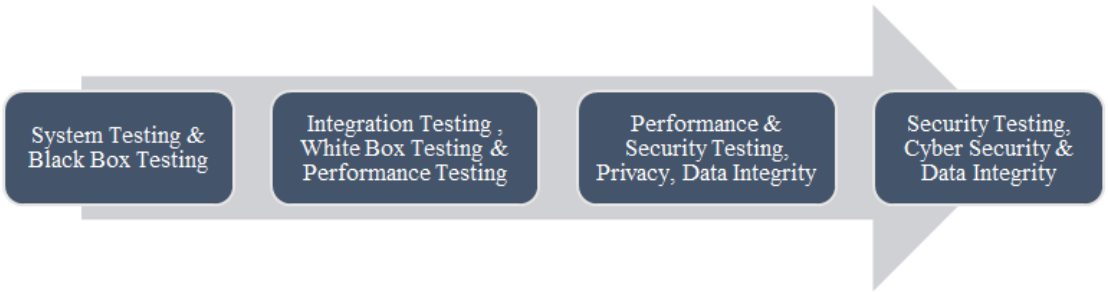

Figure 3. Shifting focus of testing

3. IT INDUSTRY SURVEY

A survey of 50 senior IT professionals was done to know root cause of software failures and to figure out which phase is most time consuming. We used survey monkey tool to conduct the survey and also used emails and manual form entry for the same. Around 10 questions were prepared to know the details of current challenges, tools, root cause of failures & techniques used in software industry. Close to 70% result showed that root cause of software failure these days is mainly security testing and performance testing. Security testing and Performance testing is done by tools and most of the times it is compromised. There are

no set requirements specified to measure security and performance parameters. Industry is not mature enough to gauge the after effects of lack of performance and security testing. Result of survey shown in Figure 4 and Figure 5.

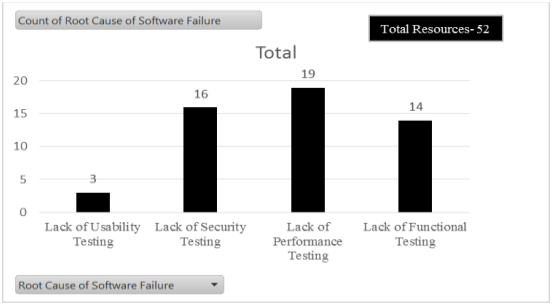

Figure 4. Survey result showing root cause of failure

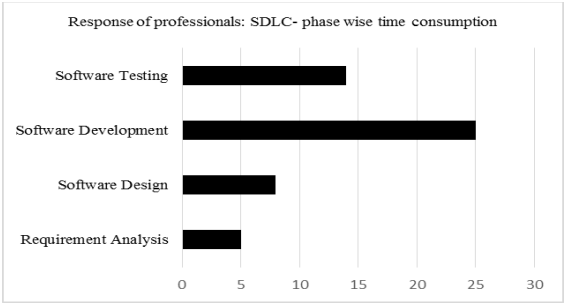

Figure 5. Survey result showing phase wise time consumption

4. SOFTWARE FAILURES

Software defects could be classified mainly in 4 categories i.e. Functional Defects, Performance Defects, Security Defects and Usability Defects. Functional testing is important in all types of technologies, however due to the maturity in functional testing software hardly fails these days due to lack of functional testing. A lot of mobile apps and web applications failed in past due to lack of performance testing as performance requirements were not identified & elicited beforehand. These days' performance parameters are also considered and a lot of research is being done in the field. There is still more research required to simplify performance testing as tools used these days are really difficult to use. Next comes the failures due to lack of security testing. Each and every organization and application are impacted due to this at some level. Due to excessive use of technology in banking and data being shared online it has become most important to ensure safety from cyber-attack and data breach. Majority of the software failures these days are due to lack of performance & security testing. Some of the recent software failures are discussed in below section outlining root cause of failures. The mosaic of technologies used in current Web applications (e.g., HTML5 and JavaScript frameworks) increases the risk of security breaches. This situation has led to significant growth in application-level vulnerabilities, with thousands of vulnerabilities detected and disclosed annually in public databases such as the MITRE CVE-Common Vulnerabilities and Exposures. The most common vulnerabilities found on these databases especially emphasize the lack of resistance to code injection of the kind SQL Injection (SQLI) or Cross-Site Scripting (XSS), which have many variants. This kind of vulnerabilities indeed appears in the top list of current Web applications attacks [6].

4.1. Ransomware

One of the most recent and major software failures that happened in June 2017 is ransomware [7]. Ransomware is considered as a type of malicious software which stops user from using his PC [8]. It is called as ransomware, because it asks for payment to give you back your files and system. It holds your PC or file for ransom. The malware shutting down computers worldwide is known as WannaCry. WannaCry was sent to systems as a virus, which spreads among computers by its own via internet. It finds vulnerable machines and infects them. 99 countrie's hospitals, automakers, IT Companies and all types of industry got impacted with loss of millions of dollars. Cyberattackers hit 99 countries across the world and is known as one of the biggest cyber-attack in history. It impacted major countries like USA, Russia, China, UK and India. In the world of digitization, where every information is stored digitally, information can be accessed 24X7, can be accessed via internet and easily retrieved at cheaper rate. Everything is done smoothly on one click, effortlessly and efficiently maintained. Digitization has improved the life style of the computer users. But as it is said "Every pillar has two sides". Digitization has helped in decreasing crime if applied on whole, getting things done easily and has decrease documentation work. But still it creates a problem of security for personal and confidential information of an individual. Many thefts or cyber-attacks like spyware, malware, Trojan, phishing, intruders, spam, virus occurs. Ransomware is also a theft. It is a kind of infection that if transmitted, it's difficult to get out. It infects all essential data and file in user's computer system. If ransomware get activated in user's system, it encrypts file like .doc, .xls,.mp3, etc. by the public key-private key combination. A ransom is demanded pay ransom for your data and then only you will get those files. It becomes difficult to detect that the data or files has been hijacked. WannaCry was sent to systems as

a virus, which spreads among computers by its own via internet. It finds vulnerable machines and infects them.

4.1.1. Root cause of failure

It happened due to loophole in Microsoft Windows Security. Microsoft fixed this in windows prior to this attack, however organizations did not upgrade it due to time taking process. As it required thorough testing of complex systems, which takes lot of time. Risk based software testing was not done in mostly organizations. Software should be tested after development and secure SDLC guidelines should be followed in life cycle. Secure code analyzers to be included at the time of development.

4.2. Zomato failure

Zomato initially named as foodiebay was startup in 2008 by Mr. Deepinder Goyal. It is a restaurant searching platform. Foodiebay, the initial name was named to Zomato in November 2010 to increase their reach among people. Zomato covers a list of over 1, 20,000 restaurant and more than 100 million customers worldwide. Currently it has more than 90 million monthly visits and has operations across the globe. It is headquartered in Delhi. It was named among top 25 most promising internet companies in India by SmartTechie Magazine [9]. Zomato encountered a security breach and was hacked. As per hackread blog and timesofIndia news, data of 17 million users was stolen in May 2017. The user's information was put on dark web for sale for just $1000. One of the cyber security company infySEC, from Chennai, India, was able to pull user information from Zomato database as well. There is no denying the fact that multi-million-dollar company having operations across the globe risked user information due to lack of secure systems. On 18th May 2017, it encountered a security breach and was hacked. As per hackread blog and timesofIndia news, data of 17 million users was stolen. The user's information was put on dark web for sale for just $1000. Later company asked its users to change their passwords. Zomato CTO also confirmed that all affected users have been logged out of the website and app. One of the cyber security company infySEC, from Chennai, India, was able to pull user information from Zomato database as well.

4.2.1. Root cause of failure

There is no denying the fact those multi-million-dollar companies having operations across the globe risked user information due to lack of secure systems.

4.3. Other recent software failures

Over the last few years technology has seen great advances but has also witnessed some really major failures. There have been lots of ransomware attacks, data breaches and IT failures. Based on our research we found that software failures mainly happen due to lack of security testing or performance testing. Table 1 lists software failures of 2016-17 with their description and root causes of failures. We researched a lot of failures and every second failure was due to either lack of performance testing OR security testing. We just outlined a few of the most famous failures in Table 1 to reach at our conclusion.

Table 1. Recent Software Failure

S.No.	Name	Year	Impact and Description	Root Cause
1	Equifax [10]	Jul-2017	Equifax, one of the largest credit bureaus in the U.S. Personal information (including Social Security Numbers, birth dates, addresses, and in some cases drivers' license numbers) of 143 million consumers; 209,000 consumers also had their credit card data exposed.	Lack of Security Testing
2	Yahoo Data Breach	Sep-16	As per https://www.hackread.com/- Forensic experts of US Securities and Exchange Commission (SEC) confirmed that more than 32 million accounts were breached in a cookie forging attack. Forged cookies allowed intruder to access user's accounts without a password. Information like email address, user names, passwords, date of births, security questions and phone numbers were reportedly leaked.	Cookies were not secure.
3	Bangladesh Bank Hack	May-16	As per www.reuters.com/article/- $81 Million were stolen from Bangladesh bank in few hours and deposited them into 4 accounts in Rizal Commercial Banking Corporation in Phillipines. Unknown hackers used SWIFT (Society of Worldwide Interbank Financial Telecommunication- to communicate between member banks around the world) credentials of Bangladesh Central Bank employees for fraudulent money transfer requests.	Lack of secure banking systems.

Table 1. Recent Software Failure *(continue)*

S.No.	Name	Year	Impact and Description	Root Cause
4	Etherum, Initital Coin Offering (ICO) data breach	Jul-17	As per https://thehackernews.com/2017/10/etherparty-ethereum-ico.html- Etherum is one of the most popular cryptocurrencies after Bitcoin. There had been lots of cryptocurrency & ICO hacking done in past. Most recently there was an etherum related breach, in which US$8.4 million were stolen from Veritaseum ICO. It's been third time in a week that hackers have stolen massive amount of etherum.	It is being done possibly by exploiting one of the vulnerabilities in company's software's version 1.5 and above.
5	Canadian Immigration Site Crash	Nov-16	As per https://www.usatoday.com/- Americans crashed the canadian immigration site on election day. There were around 200, 000 users on canada's immigration site when Donald trump was declared as President of USA. Site was not able to take the load and was reportedly crashed due to huge traffic.	Lack of Performance testing.
6	IIT Delhi, DU and other Indian Universities hacked	Apr-17	As per http://www.hindustantimes.com/ Websites of Delhi University, IIT Delhi and Aligarh Muslim University were hacked in April 2017. Interface of major university websites were changed with messages for government and citizens of India. It was really an embarassment for Universities. The Hackers identified themselves as Pakistan Haxors Crew (PHC) in all 3 websites.	Lack of website securities.
7	Website and app failures due to huge traffic	Jun-16	As per http://metro.co.uk In June 2016, A popular british clothing site ASOS website and app crashed after Brexit.	Lack of performance testing for expected and unexpected heavy loads. Lack of tracking of end user performance
8	U.S. Largest Department Store App Crash	Nov-16	As per https://dzone.com/articles/6-biggest-web-failures-of-2016 On Black Friday, Macy's, the largest U.S. department store website and app crashed due to heavy loads.	Lack of Performance testing
9	Movie Tickets Website Crash	Nov-16	As per https://movieweb.com, In November 2016, Fandango movie tickets website crashed due to high demands for movie tickets.	Lack of Performance testing
10	Paytm app crash	Dec-16	As per http://indiatoday.intoday.in, after announcing demonitisation in India in Nov-16, people started using paytm heavily. Paytm app was not able to handle load and crashed several times and impacted lots of transaction.	Lack of Performance testing

5. PROPOSED MODEL

A software development life cycle (SDLC) model is a set of activities, if performed in a manner will produce desired product. A SDLC model specifies about the ways these activities are framed to produce quality software [11].

The various life cycle models are-

a. The Waterfall Model
b. Prototype Model
c. Iterative Model
d. Spiral Model
e. Fish Model
f. V- Model
g. Object Oriented Modeling

All these models talk about set of activities like requirement analysis, designing, development, testing and implementation in different manners. They simply focus on the software development and sequence of steps to be performed while developing the software. None of them specified critical elements to be considered while designing and testing of software. Survival of the software product these days require fulfillment of all 4S Quality metrics at all levels of software development.

5.1. T-Model

In T-Model as shown in Figure 6 we considered breadth and depth as 2 arms of software product. It looks like 'T' alphabet so it is named as T-model. Horizontal component is considered as breadth arm of the software and Verical component is taken as depth arm of the software. Both horizontal and vertical sections should include security and performance design and testing considerations to ensure better quality of the software application. Both breadth and depth arms are explained in brief in next section of the paper. T-Model is proposed to architect the overall software product in such a way that 4S- safety, scalability, stability and serviceability are considered at all levels of breadth and depth along with uncompromising on

functional requirement implementation. Functional requirements are base of the product and should be outlined clearly and there are various SDLC models which ensure that requirements are specified clearly. Customers are involved to approve look and feel of software and to test functionality of the application. All interfaces should be designed appropriately to have better usability of the software application.

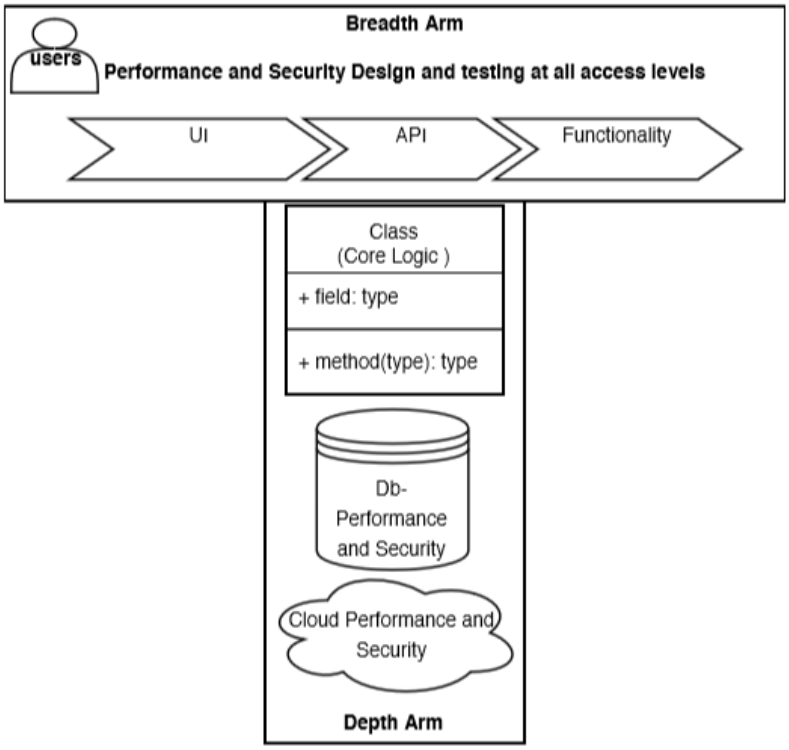

Figure 6. T-Model representation

5.1.1. Roles and responsibilites in T-Model

a. T-Model team should include a Product Manager, Engineering Manager, Infrastructure Manager, QA Manager and Project Manager. All roles should spread across breadth and depth of application along with their teams.
b. Product Manager should be domain expert and should have clear understanding of business requirements, performance and security requirements.
c. Engineering Manager should decide on architecture of product, technology and database
d. Selection along with build deployments tools.
e. Infrastructure Manager should decide on all servers, patching so that performance and security requirements will not be compromised. All servers should support scalability and security. All QA, Dev and Production servers should be identified and hosted appropriately.
f. QA Manager should identify tools for functional automation, defect management tool, performance and security testing tools.
g. Project Manager should work with all other managers and ensure that all teams work in conjunction towards success of the project. All plans related to scope, cost, schedule, effort, resources, risk are prepared, implemented and managed.

5.1.2. Breadth arm of T-Model

Breadth arm covers the Performance and Security parameters at User Interface (UI), Application Programming Interface (API) and functionality level accessible by the users of the application. This arm handles the customers who are less technical and it takes approval from customers on look and feel of software user interface designs and functionality of the application. Most of the software programmers work on this arm. Integration of horizontal and vertical arm is done in most compatible manner to ensure that the outcome application will work effectively and efficiently.

5.1.3. Depth arm of T-Model

Depth arm covers the Performance and Security parameters at Architecture, database, cloud, firewall, server level and all backend systems. In security landscape desktop, transport and network layers and web applications should be protected using antivirus, encryption and firewalls. All web servers, applications servers, database servers and backend servers should be security scanned and all patches should be updated timely to ensure all vulnerabilities are handled.

5.1.4. Performance and security parameters in T-Model

The main focus of the T-model is on performance and security requirement specification, design, development and testing at both breadth and depth arms without impacted functional requirement specification, design, development, testing and implementation. Performance requirements at a minimum should include access time, load time, number of concurrent users accessing application and latency time. Stress test, load test, endurance test, peak test and various other tests are performed to ensure performance of the application is intact. Usually tools are used to perform performance testing of applications, database and APIs. Main attributes of performance testing are speed, scalability, stability and reliability. Attributes of security testing includes Authentication, Authorization, Access Control and Audit control. There are various networks defence mechanism available for web applications like firewall, intrusion detection system, intrusion prevention system and application firewalls. If web applications would not be secure then chances of sensitive data leakage, identity theft, defacement and application shutdown will increase significantly. That will impact business, hurt brand image and goodwill of the company. Major threats for web application include authentication, authorization attack, client-side attacks, command execution on the web site, information disclosure and logical attacks to exploit a web application's logic flow. This is one of major difference between T-model and other existing software development models.

5.1.5. 4S quality metrics

T-model mainly focus on coverage of all 4S quality metrics at breadth and depth arms. Based on research we found 4S Quality metrics is most important to ensure successful release of the software product. It touched all the major software quality attributes. 4S denotes Safety, Scalability, Stability and Serviceability to test Security, Performance, Functionality and Usability respectively of the software product. There are a lot of other quality attributes at each level of software testing, however these are considered as most critical and could categorize all other attributes as well.

6. CONCLUSION

Research of most recent software failures and survey results of senior IT professionals showed that main root cause of software failures these days is lack of security testing and performance testing. Security testing and Performance testing is usually done by tools and most of the times it is compromised these days due to lack of non-functional requirement specifications. Security testing is mainly done at the end to satisfy auditors and not to find security issues from the time of architecture. Recent technologies and architectures focusing on IOT, Cloud of things, big data, machine learning and Artificial intelligence lacks security testing tool and techniques. There is lack of availability of penetration testing and testing mechanism for such technologies. There are lots of tools available for security scanning and performance testing, however each tool has its own pros and cons, be it a licensed vendor tool OR an open source tool. Today there is need of "inhouse developed tools" to satisfy needs of testing non-functional requirements. The rapid changes in software development technology stack requires more improvement in software testing as well to speed up overall software product release. There is need of setting up of requirements specifications highlighting non-functional requirements specific to security and performance parameters. Standardized automation interface should be used supporting programming language familiar to team. It should be able to use and extend open source interface drivers and support reuse of test harnesses and languages. Team should find or create drivers for the interface your product has. In addition to vulnerability security scanning, auditing and reviewing, more focus needs to be done on security and performance testing at architecture level, cloud, data and application layers to ensure safety, scalability, stability and serviceability of the product. T-model is proposed to consider security and performance parameters at requirement specification, design, development, testing and deployment levels. Further research needs to be done to ensure that these parameters are considered during continuous testing to ramp-up overall software testing process.

REFERENCES

[1] L. Chung, B. Nixon, E. Yu, and M. J. "Non-Functional Requirements in Software Engineering," volume 5 of *International Series in Software Engineering*, chapter The NFR Framework in Action. Springer, Heidelberg, 2000.

[2] C.L. Philip Chen, Chun-Yang Zhang, "Data-intensive Applications, Challenges, Techniques and Technlogies: A Survey on Big Data". *Elseveir, Information Sciences*, Volume 275, 10 August, Pages 314-347, 2014.

[3] Haohao Song, "The Development and Application of Information Security Products Based on Cloud Computing", *ICMIR*, pp 223-228, 2017.

[4] Libing Wu, Biwen Chen et al "Efficient and Secure Searchable Encryption Protocol for Cloud-Based Internet of Things", *Elseveir, Journal of Parallel and Distributed Computing*, Volume 111, Pages 152-161, January 2018.

[5] L.-O. Wallin, T. Zimmerman, "2017 Strategic Roadmap for IoT Network Technology". *Technical Report; Gartner*, 2017.

[6] T. Margaria and B. Steffen (Eds.): ISoLA 2014, Part II, LNCS 8803, pp. 337–352, 2014.c©Springer-Verlag Berlin Heidelberg 2014

[7] Savita Mohurle et al, "A Brief Study of Wannacry Threat: Ransomware Attack 2017", *International Journal of Advanced Research in Computer Science*, Volume 8, No-5, May-June-2017.

[8] Azad Ali, "Ransomware: A Research and a Personal Case Study of Dealing with this nasty malware" Issues in *Information Science + Information Technology*, Volume-14, 2017.

[9] Anirudh Deshpande, "Zomato-Market and Consumer Analysis", *International Journal of Advance Scientific Research and Engineering Trends*, Volume-1, Issue- September-6, 2016, ISSN (Online) 2456-0774

[10] Daniel Hedley, Matthew Jacobs, "The Shape of Things to Come: The Equifax Breach, The GDPR and Open-Source Security" *Elsevier, Computer Fraud and Security*, Volume-2017, Issue-11, Page 5-7, Nov-2017.

[11] Vinish Kumar, Sachin Gupta, "Proposed Software Development Model for Small Organization and Its Explanation" *International Conference on Computer Science and Information Technology- CCSIT*, pages- 167-175, 2012.

BIOGRAPHIES OF AUTHORS

Dheeraj Chhillar
Research Scholar, Department of Computer Science and Engineering, Bhagwant University, Ajmer, Rajasthan, India- 305004
M.Tech- Maharishi Dayanand University, Rohtak, India
B.Tech- Maharishi Dayanand University, Rohtak, India

Dr. Kalpana Sharma
Head of Department, Department of Computer Science and Engineering, Bhagwant University, Ajmer, Rajasthan, India- 305004
Ph.D., Masters of Computer Applications, M.Sc in Computer Science, B.A.