Volume 5, No. 3, March 2014
Journal of Global Research in Computer Science
ISSN-2229-371X
RESEARCH PAPER
Available Online at www.jgrcs.info

SOFTWARE SECURITY TESTING A PERTINENT FRAMEWORK

Rajeev Kumar[*1], Suhel Ahmad Khan[2], Raees Ahmad Khan[3]
[*1]Department of Information Technology, Babasaheb Bhimrao Ambedkar Central University, Lucknow, India
rs0414@gmail.com[1]
[2]Department of Information Technology, Babasaheb Bhimrao Ambedkar Central University, Lucknow, India
ahmedsuhel128@gmail.com[2]
[3]Department of Information Technology, Babasaheb Bhimrao Ambedkar Central University, Lucknow, India
khanraees@yahoo.com[3]

Abstract: Different surveys point toward that over the past several years' software security has raised its precedence for many software organizations. Software security testing is upgraded with the help of security attributes, security models, testing tools and most importantly test case used in testing. Software security testing was once considered a technical obligation performed by software developers. At that time, software security was much important during the test phase of software development life cycle. Present time an emergent number of security incidents and a growing awareness among business owners about invalidated software due to security issues have moved security testing into software world of testers. It is often impossible to find all the errors in a program. This basics problem in testing thus arise an open question, as to what would be the policy that we would adopt for testing. Thus now security testing is important in each phase of development life cycle. Here is a proposed framework is finalized for that security testing.

Keywords: Software security, Security testing, Test cases, Software development life cycle (SDLC)

INTRODUCTION

As per demands and an extensive use of computer and internet, software becomes more complex and large –scale, which also results in software security troubles gradually more. The technology and data attacks are maximizing the need of software security. The systematic development of software that considers security risks and threats clearly is increasingly predictable as critical to improving the overall software security. As the need for security is increasing, need of security testing is also ever-increasing. Standard software testing is all about with what happens when software fails in spite of target. The difference between software safety and software security is very minor that is the presence of an intelligent rival bent on breaking the system [5].

Security testing is the process of identifying how its attributes are behaving in presence of an attack which may harm the whole software. In every security testing it is checked that all the security factors are working properly or not, if factors are working properly then software is secure. It is a set of activities which include preparing test plan and activities related to it. Security testing is part of the analysis of security properties and its factors; these are verified relative to functional specification document, and high level design of the development system [2].

Security testing with a structured approach throughout the entire life cycle gives a good understanding of the software quality and also protects from known threats and risks [6]. With the use of a proper life cycle and design of security testing vulnerabilities and threats can be identified in the beginning of testing. A lot of work has been done in order to improve the security development life cycle but a pertinent framework of security life cycle is missing in this field. This paper will define software security and also provides a framework for security testing and its life cycle. In the absence of any standard framework able to prepare test cases for security testing, it appears worthwhile proposing a framework to identify a pertinent framework for security testing at the stage of development life cycle.

SOFTWARE SECURITY TESTING

Security is one of the important factors for quality of software. Software is fully secure when it behaves in a certified manner in presence of malicious attacks. For ensuring the security of software security testing process is applied. After going for further discussion we will discuss about security testing.

Software security testing is series of process which is designed to make sure that the computer code does what it is designed to do. The main purpose of software testing is to assure quality, estimate the reliability of software or verification and validation [7] [8][9].The security testing is performed to check whether there is any information leakage in the sense by encrypting the application or using wide range of software's and hardware's and firewall etc.

If security testing is not done during development time, software will have to suffer for that and after deployment it will make problems to the end users. The appropriate framework of security development during design time has a

strong interconnection with quality as well as with the security of the software.

SECURITY TESTING LIFE CYCLE

To maintain the quality of the software there is a need to test the security of software in a manageable way. The eventual objectives of security testing are to validate the robustness and to prevent security vulnerabilities from ever entering the software [10]. A test process is needed to ensure that the finalized system can protect itself from various malicious attacks and vulnerabilities caused by the environment. To gain this here is a proposed framework which finalizes the framework of security testing. The following are steps involved in this life cycle of security testing.

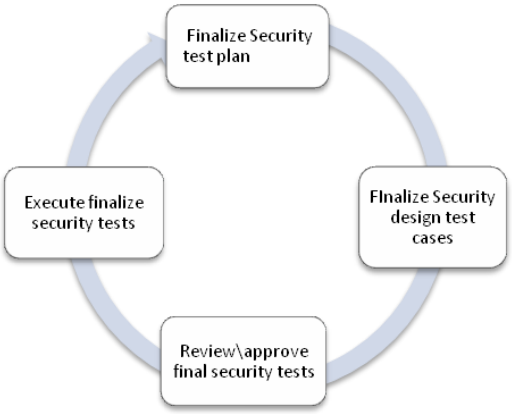

Fig 1- Security testing finalized life cycle

A. Finalize the Security Test Plan

A proper test plan should be prepared for a better implementation of security testing which includes these steps:-

- *Finalize security test types*

  Security testing consists of one or more tests that are based on the original objectives of the software security, which were defined during the project interview. The purpose of this task is to select the software security tests to be performed, not to implement the tests. Finally the software security tests that can be automated with a security testing tools need to be finalized.

- *Finalize security test schedule*

  In this task, the software security test schedule should be finalized, includes the testing steps, target start, target end dates and many responsibilities. It is also described how it will be reviewed, tracked and validation processed.

- *Organize security test team*

  With all security testing types, the software test needs to be organized. The software test team is responsible for designing and executing the tests, evaluating the results. When development corrects defects, Software test team retests the defects to verify the correction.

- *Establish software security test environment*

  In this obligation the software security test surroundings are finalized. The purpose of the security test environment is to provide a physical framework for security testing action. The main components of security environments include the physical test capacity, tools and technologies. The software technologies need to be setup. This includes the installation of test software and management with vendors.

- *Install security test tool*

  Tools and techniques of security must provide guidance to their users throughout the testing task. The fact that security testing is about preventing intelligent adversaries from reaching their objective, useful tools will likely support human testers rather than trying replacing them. A trial run of testing tools should be performed for checking whether they are ready for the test or not.

B. Software security design test case

Security strategy for designing software security test cases is to focus on the following four security components: the assets, threats, exposures and controls.
Assets are the intangible and tangible resources of an entity. The evaluation approach to list what should be protected, such as amount of software are value, use and characteristics. Threats are the event with the potential of cause loser harm of software. Exposures are forms of possible loss or harm of software security. Security controls are measures that protect against loss or harm.
It is important to assess the performance of the software security mechanisms as well as the functions themselves. Some questions and issues regarding software security performance are given below:-

- *Availability*

  What portion of time is the software or control available to perform critical security functions? Software security controls usually require higher availability then other portions of the software.

- *Survivability*

  How will software understand the major failures and attacks or natural disasters? This includes the support of emergency operations during failure, backup operations afterwards to return to the regular position.

- *Accuracy*

  How accurate is the software security control? It measures. Accuracy encompasses the number, frequency, and significance of errors.

- *Response time*

  Response time acceptable or not? Slow response time can tempt users to bypass software security

controls. Response time can also be critical for control management.

- *Through-put*

  Software security control supports required use capacities. Capacity includes the pack and average loading of software security users and service requests.

C. Review\approve software security tests

In this phase of cycle software test cases and tools are prepared and security cases are approved for being executed in the next phase. It includes the following steps:-

- *Schedule\conduct security review*

  The software security test plan should be scheduled and reviewed in advance. Testers should get the latest copy of review. The actual goal of this task is for development and Development Company or sponsor to agree and accept the reviewed plan.
  As with any review or revision, assured elements should be present. The first is defining what will be discussed of security testing [10]. The second is discussing the important details related with it. Third is summarization of security testing and its tools. The final element is correctness.

- *Obtain validation*

  Validation is critical in a testing effort because it improves testing, helps in development. The best approach is with a formal sign-off process of a software security test plan. In this case use the management approvals sign-off forms. In attach document the latest test plan of software security and point out that all their feedback comments have been included. Software security test plan will evolve with each iteration but that will be included with the alteration.

D. Execute software security tests

This phase executes all the prepared and approved test cases using prescribed tools and techniques in the last phase. This phase includes the following steps:-

- *Regression test the software security fixes*

  The intention of this assignment is to retest the security tests that discovered defects in the previous security test cycle for this phase. The regression testing technique is used for this task. Regression testing is a technique of software security that detects those errors which are the reason for origin of other errors. The different test cases are prepared for this task than a checklist is prepared for detecting the errors. A retest matrix relates test cases to functions.

- *Execute new software security test*

  The purpose of this task is to execute the new software security tests which were prepared in previous security test life cycle. In the previous phase the testing team updated the functions, software fragment, and the acceptance tests in preparation for the current phase.

- *Document software security defects of overall result*

  During execution of software security test, the results must be reported and noted down in the defect-tracking database. These software security defects are classically related to the individual tests. A document is prepared for these defects log. The objective of this task is to document these security defects properly with their existence in place and preparation of complete records of defects.

DIFFERENT WAYS TO TEST SECURITY

We can secure software in different ways. Every software is build in many modules, which have several security attributes. So, for getting the whole different ways a formula is to be modified, where security attributes are on focus. From that formula we can find out before software testing that how much testing should be done. Security testing can be completed in a number of different ways and security testing as term has a number of different meaning or method [10, 11, 14]. From that formula we can find out time and cost of the security testing that reduces repetition of testing.

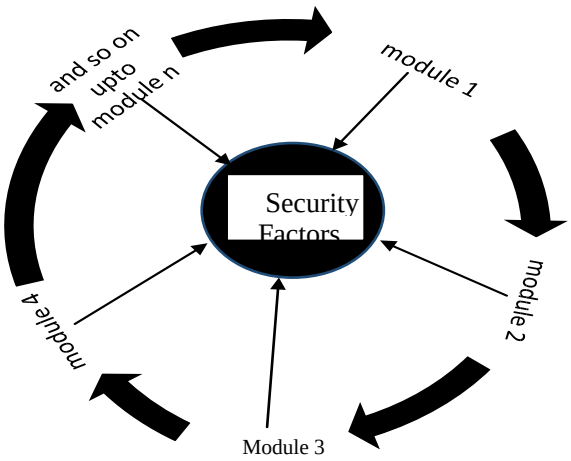

Fig 4: Security testing under modules where factors on focus

It would be a research project on its own to gather and specify detailed specification of tools and cases for testing of security. As security testing is done in modules than the number of ways in a module is (K*P) for module 1,(K-1)*P for module 2, (K-2)*P for module 3 and so on up to n. hence, number of different ways to test security in all modules will be A that is

A = P*(K-0) + (P* (K-1) +P* (K-2) +P* (K-3) + …….. +P* (K-n+1) +P* (K-n)

A = P* [(K-0) + (K-1) + (K-2) + (K-3) + …… + (K-n)]

$$A = P * [\sum_{f=0}^{n} (K - f)]$$

Where,
P = numbers of attributes of software security
K = number of total modules
f= (whole numbers) number of modules, which is to be tested.
i.e. (f= 0, 1, 2,3,4,5, - - - - - , n)

For example: - K=6, P=5, f= 0, 1, 2,3,4,5.
Then, total number of different way to test of software security A = 5* [(6-0) + (6-1) + (6-2) + (6-3) + (6-4) + (6-5)]
A= 5*[6+5+4+3+2+1] =5*[21] = 105

Number of software models	Number of modules	Number of attributes	Total ways to secure a software
1	**6**	**5**	**105**
2	**6**	**6**	**126**
3	**7**	**4**	**112**
4	**6**	**4**	**84**

Table-2 shows ways to secure software

MOTIVATION

The study in this paper will help the testers and beginners to plan a security development life cycle. This work is done to improve the reliability and security of software. The proposed mathematical formula is a step ahead to prepare the test cases during any security development life cycle. If applied this work will enhance the features of security and also preserve the time and cost involved in the development of secured software.
Future goal of this work will be to identify factors and develop a framework of security using security features to enhance the security and making software more reliable.

CONCLUSION

It has become essential to integrate and finalize a life cycle for software security testing. This paper analyses the definition, classification, major methods, tools and techniques of software security and also presents a life cycle for software security. The major techniques used in security testing are described briefly. It also suggests a mathematical formula to evaluate a ways to test security which is to be used in security testing, from that formula we can reduce the cost and time of security testing in SDLC. The finalized life cycle of security testing helps a developer or tester to manually prepare or finalize a security test case as a report says that 40% of security testing is done on paper [12].

REFERENCES

[1] K. He, Z. Feng, X. Li, "An Attack Scenario Based Approach for Software Security Testing at Design Stage" In: 2008 International Symposium on Computer Science and Coputational Technology, pp.782-787. IEEE Computer Society (2008)

[2] M.D. Abrams "FAA System Security Testing and Evaluation" In: MITRE Technical Report (May 2003)

[3]Gu.T.Y, Shi, Y.S., Fang, Y.U. "Research on Software Security Testing" In: World Academy of Science, Engineering and Technology, 647-651(2010)

[4] A. Agrawal, R. A. Khan and S. Chandra, "Software Security Process – Development Life Cycle Perspective" In: CSI communications, August 2008, pp. 39-42

[5] G.McGraw, B.Potter,"Software Security Testing" In: IEEE Security and Privacy 1540-7993/04 copyrirgt 2004.

[6] S. Turpe. "Security Testing: Turning Practice into Theory" In: IEEE international conference on software testing, verification and validation workshop (ICSTW 2008), IEEE computer society (2008)

[7] Introduction to Software Testing Available at http://www.onestoptesting.com/introduction/

[8] Software Testing Techniques Available at http://pesona.mmu.edu.my~wruslan/se3/readings/gb1/pdf/ch14-gb1

[9] Paper by L. Luo Available at http://www.cs.cmu.edu/-luluo/courses/17939 report pdf.

[10] S.A.Khan and R.A.Khan,"software security testing process: phase approach" In: A.Agarwal etal(Eds):IITM 2013. CCIS 276. pp. 2011-2017, 2013 c Springar-verlag Berlin Heidelbarg 2013.

[11]http://www.Computechdoc.org/independent/security/recommendations/secsoftwarev.html.

[12] M. E. Khan,"Different forms of Software testing techniques for finding errors", In: IJCSI, Vol.7, Issue.3, No1 May 2010.

[13] C. Kaner, "What is a Good Test Case?" In: STAR East May 2003.

[14] J. Vemulpati, N. Mehrotra, N. Dangwal "SaaS Security Testing: Guidelines and Evaluation Framework" In: 11th Annual International Software Testing Conference 2011.

[15] D.P. Gilliam, J.D. Powell, M. Bishop. "Application of Lightweight Formal Methods to Software Security"[C]. In proc. 14th IEEE International Workshops on Enabling Technologies (WETICE 2005), 13-15 June 2005, Linkoping, Sweden.pp. 160-165

Dr. R. A. Khan is currently working as a Associate Professor and HOD of Department of Information Technology, Babasaheb Bhimrao Ambedkar University (A Central University), Lucknow, UP. His area of interest is Software Security, Software Quality and Software Testing. He has authored two books on software quality and software testing.

S. A. Khan is pursing PhD in Information Technology form Babasaheb Bhimrao Ambedkar University (A Central University), Vidya Vihar,Raibareli Road, Lucknow This Young, energetic Research Fellow, who has completed a Full Time Major Research Project funded by University. His interest area of software security and testing.

Rajeev Kumar is pursuing Msc in information technology from Babasaheb Bhimrao Ambedkar University (A Central University), Vidya Vihar,Raibareli Road, Lucknow. His interest of area in software security, software testing and software risk.