European Journal of Computer Science and Information Technology
Vol.4, No.5, pp.65-69, September 2016
Published By European Centre For Research Training And Development UK (www.eajournals.org)

LOCATE A PIN IN A HAYSTACK BEFORE THE CUSTOMER FINDS

Dr. Rajiv Chopra
Ph. D. (Computer Science), Assistant Professor
Department of Computer Science Engineering/IT
Guru Teg Bahadur Institute of Technology, GGSIPU, Delhi.

ABSTRACT: *Software testing is becoming as much vital today as is taking meals everyday. Researchers say that testing a day, keeps errors at the bay. Literature suggests that software testing should be done during requirements analysis phase, software design phase, software coding phase and also during software maintenance phase. Different types, different models and different layers of testing have been proposed in literature. The need is to test the software at every stage by any means. Testing may vary from documents testing (static testing) to code testing (dynamic testing). And finally, during maintenance phase also regression testing must be done. But locating errors in the source code is as much difficult as much is locating a pin in a haystack.*

KEYWORDS: Software testing, Testing everyday, Static testing, Dynamic testing, Regression testing.

INTRODUCTION

Software testing is both an art and science. The security aspect of testing can be introduced right from the start of software development life cycle (SDLC) – right from requirements analysis phase far architecture and design, coding, software testing, quality production, to the operation and maintenance [1]. Weaving security into the web application delivery lifecycle does require a synergistic approach that incorporates people, process and technology (PPT). There are various security testing standards and methodologies that represent a general consensus on the major areas of threats.

The Open Web Application Security Project[2] (OWASP) experts list standard top-ten threats to web security. This provides a common reference point for developers, consumers and vendors. It helps to build and test secures web applications. Web Application Security Consortium (WASC) is another standard that classifies the vulnerabilities to a web application. There is some obvious overlap with OWASP model. Both OWASP and WASC items can be interwined together to provide the best benefits to the client base.

Another standard, OSSTM, is for unprivileged security testing of an application. It provides a security scattershot of the environment and system. It includes many other items like Information, Internet and Communication.

ISSN 2054-0957 (Print), ISSN 2054-0965 (Online)

Also there exists a common criterion that defines evaluation of security properties of IT products and systems. Here, applications are defined at 7-levels ranging from EAL 1-7. Common criteria 2.1 is now ISO 15408.All these standards do vulnerability analysis, we need to couple our findings with that of a finite set of criteria given by above standards, gathered overtime [4]. Without a trained eye and a deep understanding of web technology, this analysis may lead to open gaping holes that will defeat the purpose of pen testing web applications.

REVIEW LITERATURE

Web applications testing is a specialization of software testing. testing a web application is challenging, not only for mission critical tasks it carries out but also for factors like security, the scalability and the high dependency of the outputs on the web browser and on the way a user interacts with the web browser. A few of recent studies are stated below.

1. In paper given by Diana Kelley [1] software testing is suggested to be performed in every phase of software development life cycle. It also discusses about the challenges, issues and limitations related to it.
2. In [2], Open Web Application Security Project lists top ten vulnerabilities related to website's security.
3. In [3], another organization, Web Application Security Consortium also lists some of the vulnerabilities related to software security.
4. Andres Andreu, explores pen testing of websites and web applications.
5. Arabi Keshk, Amal Ibrahim discusses about ensuring of the Quality Testing of Web Using a New Methodology.
6. Alessandra Bagnato, Fabio Raiteri, Wissam Mallouli, Bacher Wehbi [6], discusses about the practical experience gained from passive testing of Web based systems.
7. In [7], Wenhua Wang, Sreedevi Sampath, Yu Lei, Raghu Kacker, emphasize on the Interaction based Test Sequence Generation Approach for Testing Web Applications.
8. Andy Ju An Wang [8], discusses about the Security Testing in Software Engineering Courses. They focus on that security must be taught as a part of professional courses.

PROPOSED WORK

Software testing is the process of executing the program with the intent of finding the faults. A web application pen test uses unique techniques to expose potential flaws in client/server applications. Actually, a web application is a client/server model with the browser as the client. Both black box (functional) and white box (structural) testing can be used in the upcoming phases. In this paper, it is proposed to do functional or black box testing of websites due to the following reasons:-

ISSN 2054-0957 (Print), ISSN 2054-0965 (Online)

1. Initially, when a project starts, source code is usually unavailable.
2. It is easier to get black box testers as strong programming skills are not required.
3. Doing static analysis, vital information can be gathered for web masters, developers and testers.
4. Since the errors found during testing have their root cause to coding, which depends on design and requirements analysis? It is therefore, very right to go back to design phase and find out the root cause of software failure.

A website is a collection of related web pages. In this paper, static analysis is therefore, emphasized. It includes:-

a) Size of web pages.
b) Overall structure or site map of the website.

Static analysis of websites is a specialized area covered by dedicated tools like hyperlink tools. These tools are also known as web-spiders. These tools must interact with website and Internet (both) in order to find defects. The defects found by these tools include:-

a) Incorrect route to a wrong page.
b) Undesired redirection.
c) Hyperlinks do not link to a website at all.
d) Downloading and uploading takes huge amount of time.

The point is that if at the design level only, a website security is thoroughly tested then the errors need not propagate to further phases of SDLC.
Web applications are developed using spiral model. They have shorter delivery schedules than normal conventional IT projects. Web projects are developed often in the range of 1 to 3 months.

During design phase, since a website design is available, so test websites for its usability i.e. its navigation and data flow between components, tiers or systems.
In order to satisfy users requirements and make web applications easy to use, usability testing is carried out [5]. Usability is embodied in such aspects as navigation, graphics and appearance. The navigation should be concise, clear and uniform. The graphics should have definite intentions. The correctness, precision and relativity of the graphical information also should be guaranteed. The main objective is to check whether the pages are compact, consistent and contrastive.

All entities of a web application must be tested thoroughly. But web testing raises new issues also. One of the key issues is the unexpected change in state when the browser's back button is clicked or direct entry of URL in the browser. Thus, there is a need-

a) To test the navigation between pages those are of interest.
b) To propose a new web testing model for web application testing.

ISSN 2054-0957 (Print), ISSN 2054-0965 (Online)

Since a website is defined in terms of objects (i.e. a link, a command button, a list box on a web page, a message, an image, a downloadable file, audio or another website itself). So the objective is to test these objects [5]. Practically speaking, an application with broken links, missed images and slow download will not attract customers.

Both passive and active testing can be performed [6]. Passive means that tests do not disturb the natural operation of protocol. The records of the observed events are called as a trace. This trace will be compared to security properties derived from the standards. Passive testing can be applied on a system in its real context (with real users) whereas active testing is performed in a simulated environment with simulated/ emulated system users.

Interactions between dynamic pages can be carefully tested. However, it is impossible to test all interactions [7]. A trade-off exists between test coverage and test effort. So, at least pairwise interactions i.e. between two pages are the need. If a page, P1, could reach another page P2 then there must exist one test sequence in which both P1 and P2 are visited in the given order. So, we need to capture the navigation structure of the application under test (AUT). Hence, an abstract model of AUT is desired. Test sequences are generated from this model.

Static analysis means testing system properties by code inspections without executing them. Static techniques for source code analysis allow us to proactively eliminate or neutralize security bugs before they are developed or exploited [8]. Also if a program and its security properties are modeled then model checking techniques will help to identify whether there is any state that violates the desired security goal.

SQL injection occurs when a user enters inputs through forms to the background (server side) database in order to generate SQL query statements. To find all injection points i.e., the attack points of the application, a complete scan of an application is to be done. To scan an application, the basic structure of the application must be extracted. It can be expressed as a tree whose root node is the front page of a website. The child nodes of a node represent its link points. A tree scan is performed with breadth-first search. Then test cases are generated and run. Test results are collected and test reports are generated.

CONCLUSIONS

Locating a pin in a haystack means locating defects in both workable and non-workable code. This activity is so much difficult that in spite of the fact software tester's claim that the testing is complete yet the errors escape out. And if you will not locate and debug these errors then the customer will do it. Testing involves some amount of cost, effort and time. The objective is to do testing using optimal resources.

ISSN 2054-0957 (Print), ISSN 2054-0965 (Online)

REFERENCES

[1] Diana Kelley, Practical Approaches for Securing Web Applications across Software Delivery Life Cycle, 2009, http://www.securiycurve.com.
[2]http://www.owasp.org/documentation/topten.html.
[3]http://www.webappsec.org/projects/threat.
[4] Andres Andreu, Professional Pen Testing for Web Applications, Wiley India Pvt. Ltd., 2006.
[5] Arabi Keshk, Amal Ibrahim, Ensuring the Quality Testing of Web Using A New Methodology, IEEE International Symposium on Signal Processing and Information Technology, 2007, pp 1071-1076.
[6] Alessandra Bagnato, Fabio Raiteri, Wissam Mallouli, Bacher Wehbi, Practical experience gained from Passive Testing of Web based systems, IEEE, Third International Conference on Software Testing, Verification and Validation Workshops, 010, pp 394-402.
[7] Wenhua Wang, Sreedevi Sampath, Yu Lei, Raghu Kacker, An Interaction based Test Sequence Generation Approach for Testing Web Applications, IEEE, High Assurance Systems Engineering Symposium, 2008, pp 209-218.
[8] Andy Ju An Wang, Security Testing in Software Engineering Courses, 34th ASEE/IEEE Frontiers in Education Conference, 2004, pp F1C-13 – F1C18.

ISSN 2054-0957 (Print), ISSN 2054-0965 (Online)