Annals of DAAAM for 2011 & Proceedings of the 22nd International DAAAM Symposium, Volume 22, No. 1, ISSN 1726-9679
ISBN 978-3-901509-83-4, Editor B. Katalinic, Published by DAAAM International, Vienna, Austria, EU, 2011
Make Harmony between Technology and Nature, and Your Mind will Fly Free as a Bird
Annals & Proceedings of DAAAM International 2011

SECURITY TESTING OF WEB APPLICATIONS

VALA, R[adek] & JASEK, R[oman]

Abstract: This article provides brief methodology for security testing of web applications. Web applications are very popular in recent years and in many areas are beginning to replace desktop applications. However developers of web applications are facing the fact that there is not a standardized way for creating secure code and complete security testing of their applications. This article brings together the methodology and best practices for web applications security testing from different resources (recommendation of professionals, standards (ISO/IEC 27000-series [1]), open security project OWASP [2]).

Key words: security testing, web application, OWASP, software testing

1. INTRODUCTION

Information society of these days fully uses information and communication technologies for managing and sharing information. The risk of information abuse increases in proportion to its value. Internal information systems of many organizations are in recent years transformed to open Internet information systems (web applications) which provide benefits of data sharing. However, developers of web applications (WApps) are facing the fact, that there is not a standardized way for creating and testing some WApp like in case of standard desktop application.

According to WhiteHat Security research (WhiteHat Security, 2010) 64% of web pages contain at least one serious vulnerability, which can be abuse by the hacker attack. This number is alarming and shows the high importance of using appropriate security policy for web applications developing. One important part of this security policy is security testing of WApps, which are much more exposed to misuse and hacking, than the desktop application in general. This paper summarizes the important steps of testing security of WApps and brings the important information from different resources like OWASP (The Open Web Application Security Project), ISO standards or recommendation of professionals.

Each process of security testing should consist of 5 steps (Risk assessment, Establishing testing objectives, Creating test plan, Testing and Analyzing and Reporting test results).

2. RISK ASSESSMENT

Risk assessment (RA) is overall process of risk analysis and risk evaluation (ISO/IEC 27000:2009(E), 2009) and it is used to identifying high risk areas of application. Each application has areas which are more or less crucial for its function. One method of creating RA is to identify and weight these areas with risk attributes. There are basically three common factors to consider by creating risk dimensions.

2.1 Project structure

With respect to project structure, the more structured project means the less the risks.

2.2 Project size

Project size is directly proportional to risk. The larger the project in terms of cost, staff, time, number of functional areas involved is, the greater the risk.

2.3 Experience with technology

Experience with technology is inversely proportional to project risk. It means, that better experience with the hardware, the operation system, the database or the development language leads to the less risks.

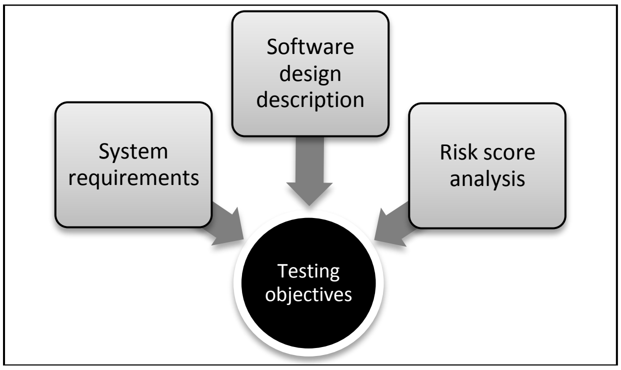

Fig. 1. Inputs by establishing of testing objectives

3. ESTABLISHING TESTING OBJECTIVES

Establishing testing objectives is very often a creative process (brainstorming), by which testers and developers are going through all system requirements and try to find and describe all use cases.

More accurate results are received if establishing testing objectives is based on business transactions.

Necessary input documents by establishing testing objectives are System requirements, Software design description and Risk score analysis.

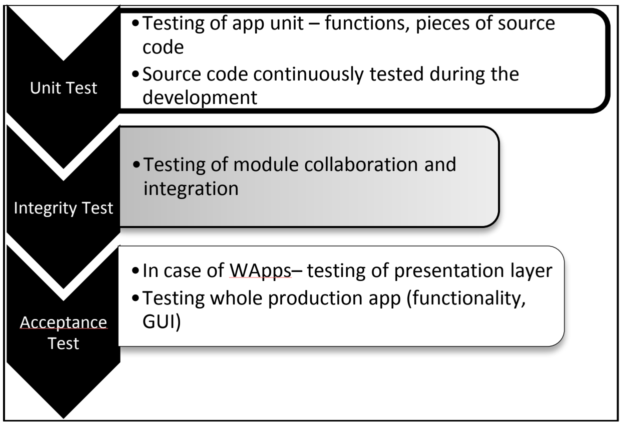
Unit Test
• Testing of app unit – functions, pieces of source code
• Source code continuously tested during the development
Integrity Test
• Testing of module collaboration and integration
Acceptance Test
• In case of WApps– testing of presentation layer
• Testing whole production app (functionality, GUI)

Fig. 2. Testing types

4. CREATING TEST PLAN

The Test plan is based on 4 input documents – System requirements, Software design description, Risk score analysis and Testing objectives. It is an operation document and is basis for testing.

4.1 Types of tests

The process of testing is generally divided into three main types (Fig. 2.).

4.2 Test case design strategy

By creating test plan, developers should also consider the test case design strategy. There are three testing methods: Black Box, White Box and Gray Box method. The choice of the right testing method should depend on few factors: Project testing money and time budget, number of inexperienced testers and number of experienced testers.

The Black Box method is the cheapest method, because there is no need for experienced testers. The tester does not know inner structure and functionality of the tested script. Testing is based only on defined input and output data. If the output data are expected, the test passed.

More time and money consuming is White Box method. Experienced tester is going through source code of the tested unit. However, this method is able to expose more vulnerabilities or bugs than Black Box method.

Compromise between these two methods is Gray Box method. In context of security tests white box testing is meant as source code analysis, and black box testing is meant as penetration testing. Gray box testing is similar to Black box testing. In a gray box testing we can assume we have some partial knowledge about the session management of our application, and that should help us in understanding whether the logout and timeout functions are properly secured (OWASP Testing Guide V3, 2008)

4.3 Goals of testing

A good practice for developers is building security test cases as a generic security test suite that is part of the existing unit testing framework. A generic security test suite might include security test cases to validate both positive and negative requirements for security controls such as: Authentication & Access Control, Input Validation & Encoding, Encryption, User and Session Management, Error and Exception Handling, Auditing and Logging (OWASP Testing Guide V3, 2008).

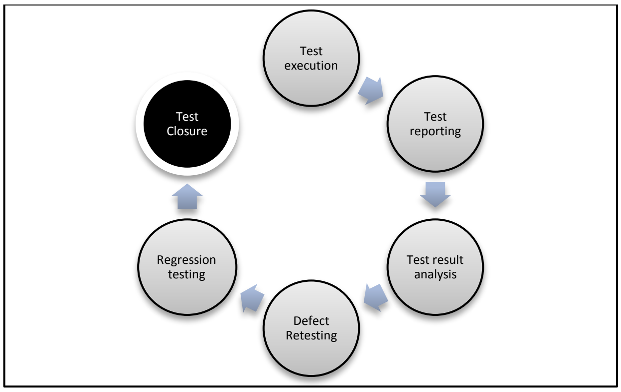

Fig. 3. Testing process

5. TESTING AND ANALYZING

If the test cases and test plan are done the testing part can begin (Fig. 3.). The first step is test execution. All results of this part should be covered by the testing report. This testing report is the basis for test result analysis. The purpose of test result analysis is to show any problem areas which have to be repaired and then tested again. This testing is called as defect retesting and should be done after every code repair. By fixing one part of functionality is possible that another part is damaged. Therefore is necessary to make regression testing also. This regression testing should retest if no bug were created by code fixing. If regression tests passed whole testing process is finished.

6. REPORTING TEST RESULTS

One of the most important parts of testing is analyzing and reporting test results. The result of this part is a report which describes the level of meeting the test objectives. There should be also recommendations based on the test results and this report should cover all of the completed testing activities.

7. CONCLUSION

In these days web applications are largely replacing desktop applications. It can be seen that even in critical areas, such as banking, energy and government are deployed and operated web applications with remote control. In the process of developing web applications takes one of the most important parts security testing. Testing should be systematic process which is divided into several steps and which verifies the product application. For testing purposes there should be available these documents as an input: System requirements and Software design description. Five steps are then performed: Risk assessment, Establishing testing objectives, Creating test plan, Testing and Analyzing and Reporting test results. Whole process should be well documented and reported.

This article is focused on security testing of web application and is limited by the fact that currently there are no standardized approaches in this area of security testing. The process is based on general software testing approaches and standards and the best practices of security testing groups like OWASP are considered.

8. ACKNOWLEDGEMENTS

This survey was financed by Internal Grant Agency of Tomas Bata University in Zlin, Faculty of Applied Informatics IGA/49/FAI/10/D.

9. REFERENCES

Everett, G. & McLeod, R. (2007). *Software testing: testing across the entire software development life cycle*, Wiley-IEEE, 047179371X, New Jersey

Hope, P. & Walther, B. (2009). *Web Security Testing Cookbook: Systematic Techniques to Find Problems Fast,* O'Reilly Media, Inc., 0596514832, Sebastopol

*** (2009) ISO/IEC 27000:2009(E): Information technology - Security techniques - Information security management systems - Overview and vocabulary, *Available from:* http://www.iso.org, *Accessed on*: 2011-08-10

*** (2011) Microsoft TechNet: Planning, Testing, and Piloting Deployment Projects, *Available from:* http://technet.microsoft.com/cs-cz/library/cc786374(WS.10).aspx, *Accessed on*: 2011-07-28

*** (2008) OWASP Testing Guide V3, *Available from:* http://www.owasp.org/images/5/56/OWASP_Testing_Guide_v3.pdf, *Accessed on*: 2011-08-20

*** (2010) WhiteHat Security, WhiteHat Website Security Statistics Report, *Available from:* https://www.whitehatsec.com/home/resource/stats.html *Accessed on*: 2011-08-13