Seven Phrase Penetration Testing Model

Parvin Ami
Assist. Professor
B.K. Mehta IT Center, Palanpur
Banaskantha, Gujarat -385001, India

Ashikali Hasan
Chief Technical Officer
Xeniar Technology Pvt Ltd
Ahmadabad, Gujarat-380015, India

ABSTRACT
Generally in penetration testing process a simple way to test a system is held by primary analysis and formal methods, Based on the observation that most security flaws are triggered due to a flawed interaction with the environment[12]. Herein have describe an approach for testing web application and database integration system for possible security flaws. This approach is to be included in a dynamic model which have the aim to bind the complex and lengthy procedure of penetration testing process. Proposed approach is prepared a model using seven different Phases called as Seven Phase Penetration Testing Model (SPPT-Model). It simplifies the complex penetration testing procedure and allow penetration tester to evaluate accurately what faults to be exist in the target system. The dynamic model of penetration testing can be implementing freely and efficiently on almost all type of applications. This scheme can be used to classify informatics, analytical, complex, logical, well-known and common security flaws of web application.

General Terms
Web application threats, Web application security, exploitation process and reporting.

Keywords
Penetration Testing, System Security, Data Analysis, Vulnerability and countermeasures, System Analysis and Testing, advisory on language flexibility, Seven Phrase Penetration Testing Model.

1. INTRODUCTION
Security vulnerabilities in web applications may result in stealing of confidential data, breaking of data integrity or affect web application availability. Thus With the rapid growth of IT development the precaution are also big concerns for the research community against various threats and vulnerabilities. According to sophisticated vulnerability assessment tools 60% vulnerabilities can be found in most of web applications [1]. Even due to automation in form of software many patches and security software are exist in the global world of IT for evade this type of threats such as antivirus, Intrusion detection system, Honey port, Firewall, application filtration software, source code reviewer etc. However the most common way of securing web applications are searching and eliminating vulnerabilities [12]. Another ways of securing web application includes safe development while on other hand efficient way of finding security vulnerabilities from web applications is manual code review. In every approach all the techniques are either more time-consuming or require expert skills, and is prone to overlooked errors. Therefore, security society actively develops automated approaches to finding security vulnerabilities. According to predefined and general approach of testing application, is divided into two wide categories such as black-box testing and white-box testing:

1.1 Black Box Testing
This approach is based on web application analysis from the user side, assuming that source code of an application is not available. Specific knowledge of the application's code/internal structure and programming knowledge in general is not required. The tester is only aware of what the software is supposed to do, but not how i.e. when a certain input is entered, a certain output is returned; without being aware of how the output was produced in the first place. This testing can be expands in two forms as Remote penetration Testing and Internal Penetration Testing [2].

1.1.1 Remote Penetration Testing
The remote penetration testing is used when the developed resource is not physically reachable to the tester however tester may performs testing process through any connectivity option. This testing generally executes to test the accessibility of the general users while the application or software is broadcast globally. Real Life example of this testing is, assume that a tester receive challenge to test the web application and received only the information of the target address. Since tester must have to start testing process by establishing the connection through any connectivity option. At this point tester will not have any other information of the target system except address only [15].

1.1.2 Internal Penetration Testing
This testing refers to test the application or network with the active presence for analyzing internal structure and security policy of system. The purpose of this test is to evaluate internal security of network or application access, for example tester can determinate that any Authorize person has the necessary rights which may result in manipulate the data or the provided rights may be harm full for the network or applications in future disputes. Real Life Example: in previous remote penetration testing, tester test the application with no access while in this testing tester will have the opportunity to enter in the application having some limited access i.e. Tester is now member of bank website and now he/she will be able to test the internal function too due to his/her user level access [16][17][18].

1.2 White Box Testing
White-box testing is also known as clear box testing, glass box testing, and transparent box testing and structural testing. It is a method of testing software that tests internal structures or workings of an application [3]. In white-box testing an internal perspective of the system, as well as programming skills, are used to design test cases. The tester chooses inputs to exercise paths through the code and determine the appropriate outputs. While white-box testing can be applied at the unit, integration and system levels of the software

testing process, it is usually done at the unit level. It can test paths within a unit, paths between units during integration, and between subsystems during a system–level test. Though this method of test design can uncover many errors or problems, it might not detect unimplemented parts of the specification or missing requirements. Real Life scenario of this testing is that assume that penetration tester will test the application with no access, with half access and with full access i.e. Tester will become now member of bank or he/she will become administrator of bank website and now he/she will test the internal function with different access[4][13][14].

2. LIMITATION AND PROBLEMS IN EXISTING PENETRATION TESTING

A penetration test can have tremendous value for an organization. It has the ability to identify vulnerabilities before an attacker does, emphasize security shortcomings to management, and prioritize defense spending. However, penetration testing also has limits. Such as defined below

2.1 Tester Qualification

Penetration testing process is advance approach which executes process to find the vulnerability from system therefore it is essential that penetration tester must have to be completely expert of the system and its usage. A penetration test will never identify all of the vulnerabilities within an environment. Tool limitations, bounded scopes, and time boxed testing all play into this. Moreover, penetration testers, like software developers, are not created equally. While a methodology can be followed, penetration testing is not an exact science. For example, a junior tester may examine multiple low risk vulnerabilities and when reviewed individually may conclude no major risk exists. On the other hand, a senior tester, through past experience, may see that when the individual low risk vulnerabilities are taken as a whole, they lead to a significant compromise of the environment. Similarly, penetration testing requires a degree of problem solving and creativity skills. While such skillets can be learned and improved, they are not distributed equally between people [6].

2.2 Testing Sequence

It is also very essential that the penetration process must be perform in the proper sequence as it is required because improper sequence may result data loss or receive unexpected result. Many of the techniques used to penetrate a security perimeter can adversely affect the applications or communications systems that are core to your network. It's not just testing of firewalls, but also exposed/public applications that should be hardened but which still may be vulnerable. Process may take down services, they could corrupt live data.

2.3 Other Factors

Even testing is not the final solution or penetration tester cannot claim that after penetration testing of system it will be secure totally.

2.4 Budget and Costing

Penetration testing process is more time and resource consuming and therefore creates a large expense for an organization to execute. There is the possibility of equipment failure from penetration testing since tools and techniques are used to exploit known vulnerabilities in network devices. So in such cases Small organizations may or may not afford this type of process [12].

2.5 Update Problems

Even due to the penetration testing process until it is not completed system cannot be update with any other additionally function else penetration process will be considered as either incomplete or expired.

3. THE PROPOSED MODEL

The following diagram shows the complete model which includes seven different phases to fulfill penetration testing process.

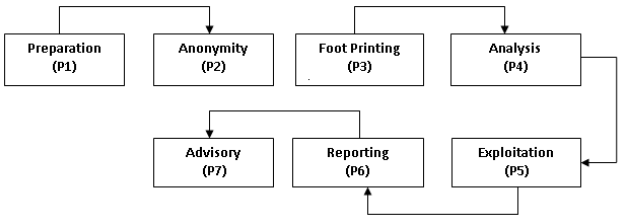

Fig. 1 Seven Phase Penetration Testing Process Model

3.1 Preparation

The system Preparation is the first phase of seven phase penetration testing model, inspect the hardware, inspection of installed software, confidentially and privacy policy, Physical Inspection, attack diagram and mapping process. An inspection of hardware and software is a formal examination or evaluation exercise of computer hardware and software. It normally involves efficiency, measurements, and tests to determine whether the characteristics of the relevant hardware/software or activity being inspected confirm to pre-decided standards. Additional purpose of hardware and software inspection is to ensure that during the penetration testing tester will not receive type of interruption. Privacy policy is the process to review the local laws of the jurisdiction and the review and understanding the company privacy policy. It is also determinate that the result received from penetration test must be store securely therefore it is essential to inspect the physical resources. When any system is going to be penetrate at the time it is essential to make clear map or process. For example penetration tester may create the diagram or the scratch that may be helpful to represent preparation or testing process [19][5].

3.2 Anonymity

Anonymity phase is the primary process which initiates the actual and first phase of penetration process. This phase includes making the attacking (Penetration) system more invisible. There are several ways which can use to fulfill this requirement such as using proxy, VPN, Socks etc... There is more than one option when choosing which proxy to use, which is why it's important to know the different types of proxies offered. It is also essential to decide what type of proxy server should be use to initiate the attack. The anonymity phase also giving recommendation to create few more anonymity detail such as fake profile, fake email address, fake identification etc… However, with the increased online usage comes a higher possibility of becoming the victim of phishing or email scams, which can lead to identity theft. So penetration tester should operate this type of attacks also [19].

3.3 Foot Printing

The Purpose of foot printing is to learn as much as tester can about a system, its remote access capabilities, its ports and services, and the aspects of its security, this process is also known as information gathering. Traditional foot printing approach includes getting information by Network Enumeration, Organizational Query, and Domain Query etc. This proposed model specified foot printing in three different modules such as (I) Active Foot printing, (II) Passive Foot Printing and (III) Search Engine Foot printing [6][7][8]. The second phase initiates the process of gathering all possible information of the target. The purpose of executing Active footprint is to gather technical information of the target. This function give advantage to gather all possible information of system remotely such as IP Address, Mail Service, Request Rough of Target, Server Versioning and Information, Public Resource, Web Application Fingerprint, Directory information, Error Code and Response information, DNS Query information, Bandwidth information, SSL information, Database information. Using passive footprint penetration tester is able to collect some more data of the company via providing testers' physical interaction. This function can provide information such as Job information, Employee detail, Financial Services information and other information of company. By Passive footprint penetration tester is able to prepare by several preliminary actions to obtain the most comprehensive inventory of resources hardware, software and even human target network. It is to recover the maximum information on the network architecture, operating systems, applications and users. Google aids security professionals and penetration testers with tons and hundreds of background intelligence and information about the target company before actually launching an attack. An attacker can use a variety of strings to pull up all the information of what he needs, from finding sensitive data to what operating systems and services running on the target network. Therefore this model is suggesting receiving information by various search engines [8].

3.4 Analysis

The third phase is the core procedure which represents vulnerability assessment of the target. Through this phase, penetration tester will receive the challenge to evaluate and find the necessary security defeats from the target. This task requires complete attention in process of the penetration testing. It is very important to ensure that each task, functions, and processes must be followed in specific and proper way step by step as per proposed model of penetration testing this phase expands in two main procedures such as Code Analysis and Vulnerability Analysis. Code Analysis is used to analyze source code and/or compiled version of code in order to help find security flaws. Ideally, such analysis would automatically find security flaws with a high degree of confidence that what is found is indeed a flaw. However, this is beyond the state of the art for many types of application security flaws. Thus, such code analysis frequently serve as aids for an analyst to help them zero in on security relevant portions of code so they can find flaws more efficiently, rather than a tool that simply finds flaws automatically. We have used it. The second function represent to analyzing the vulnerability. To reduce the security risks posed by software vulnerabilities, we strive to address both the number of vulnerabilities in software that is being developed and the number of vulnerabilities in software that is already deployed. This vulnerability analysis work is divided into two areas. Identifying and reducing the number of new vulnerabilities before the software is deployed is the focus of vulnerability discovery effort, while proposed model's vulnerability remediation work deals with existing vulnerabilities in deployed software. With vulnerability discovery, it strives to help engineers understand how vulnerabilities are created and found. Main goal is that, with this education, engineers will learn how to detect and eliminate and eventually avoid vulnerabilities in software products before the products are shipped. The unfortunate reality is that many software products are being shipped with vulnerabilities that attackers may be able to exploit [6][7][8][9][10]. This vulnerability remediation process involves a comprehensive approach to protecting systems using below equation

Total Vulnerability = (CA + VA)

Where CA =Code Analysis and VA=Vulnerability Analysis.

3.5 Exploiting

Exploitation is the process to gain access by taking advantage of vulnerabilities which received previously through analysis phase Generally this phase performs if client is agreeing to evaluate impact of risks due to existing vulnerability because this phase contains high risk and may damage the targeted system. However using this task penetration tester can evaluate the perfect solutions and impacts of existing vulnerability. An exploit is a piece of software, a chunk of data, or sequence of commands that takes advantage of a bug, glitch or vulnerability in order to cause unintended or unanticipated behavior to occur on computer software, hardware, or something electronic. Such behavior frequently includes such things as gaining control of a computer system or allowing privilege escalation or a denial-of-service attack. The model optimizes this process through the six different activities. Below diagram represents entire process [6][7][8][9][10][11].

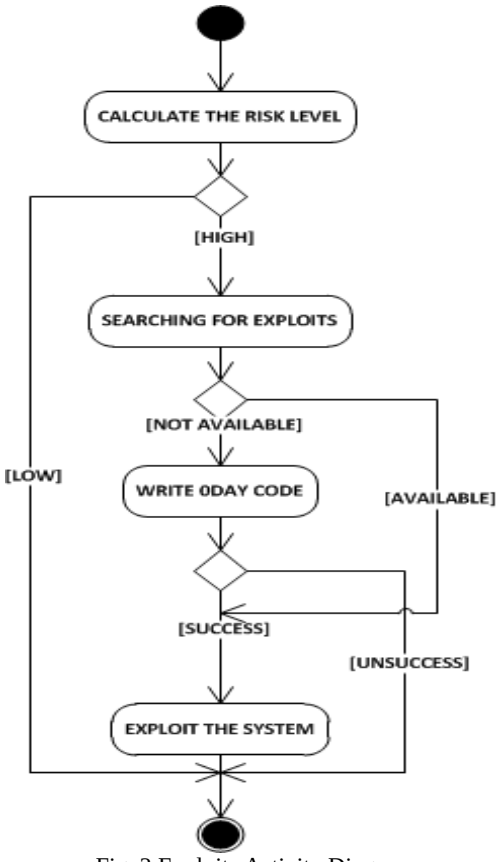

Fig. 2 Exploits Activity Diagram

1. Calculate the risk level of vulnerability.
2. If risk level is high then continue to next level otherwise end the process and note in the report.
3. Check whether the exploit is available to exploit the vulnerability.
4. If exploit is available go ahead with the exploit. Otherwise continue to next process.
5. Write 0day code or custom code to exploit.
6. Pass the results to the reporting phase.

And thus the exploit process is been optimized and fulfilled.

3.6 Reporting

When penetration testing process has been compiled at same time the next process is begin to provide advisory and various reports to senior management through reporting process, IT management and IT technical staff will all likely see the report, or at least part of it. Performing the technical side of the assessment is only half of the overall assessment process; the final product is the production of a well-written, and informative, report. A report should be an easy to understand and highlight all the risks found during the assessment phase and appeal to both management and technical staff. The report needs to have three major sections and be created in a manner that allows each section to be split off and printed and given to the appropriate teams, such as the developers or system managers. The sections generally recommended are: core Summary, Technical detail, assessment Findings, Risk Level indication overview, Patch information advisory, Budget information and Time Estimation etc... Using this report penetration tester can represent the entire process to the IT management so that the final solution can be obtain and implement. As per proposed model and study discovering vulnerabilities is important, but just as important is being able to estimate the associated risk to the business. Therefore it inspires to find the solution to calculate risk using following equation.

Risk = Possibility * collision

By this approach tester will able to estimate the level of all of these risks to the business. This system will help to ensure that you don't get distracted by minor risk while ignoring more serious risks that are less well understood [11][20][21][22].

3.7 Advisory

The final phase of penetration model includes security solution and patched information against all found risks such as Preparation of Countermeasures, Budget Estimation, Time Estimation, Creating Advisory Map, Discussion with the Client, Recheck the implemented Solution etc… this is the task where penetration tester has to give definitive and conclusive advisory report for various solutions and the cost. In many instance when the penetration testing is completed at the time it is essential for the client to install the suitable patches or to follow the advisory report given by penetration tester. In such cases, security solution should be provide in both open source and paid solutions. The advisory phase is dependent on reporting phase because advisory must be prepared after complete review of all different reports. Advisory mainly direct to include three major components to install such methods is mentioned below

1. Advice to install the patch is available from the vendor
2. Advise to install the open source patch if available
3. Advise to install paid patches and software

This process must be including comprehensive report which should cover differentiation and reliance of each solution and advisory [21].

4. SIGNIFICANCE

The proposed model of penetration testing is the combination of different phases where penetration tester is receiving the opportunity to perform sequential and well organized procedure to test application. Model is helpful for testing assesses every security detail about a website for complete trust and confidence. Each phase includes checks for various vulnerabilities, including buffer overflow, input validation, cross site scripting, URL manipulation, SQL injection, Cookie modification, bypassing authentication, and code execution. The testing is comprehensive and regular.

5. CONCLUSIONS

Penetration testing is like the annual physical at your doctor's office. There are many diagnostic tools are available to test the system, much like a blood test or an X-ray. A blood test will check for many things, but it still takes a doctor to review the data, make inferences, perform additional tests and then reach a diagnostic conclusion. Tools will test for many things, but it will always take a human to review the results and make inferences based on knowledge and experience that will never be able to put in a tool [22]. The proposed penetration testing model is providing the accurate way, well arranged and planed process of the penetration testing procedure. As a result from seven phrase penetration testing model, the tester can do better work in less time meaning they can secure more systems without sacrificing the overall quality of their testing. The model is proposed to help, small IT Company so that they can get benefits of penetration testing of their developed product or their network with less cost and automatic and perfectly well directions. The proposed model will helps automate a great deal of the penetration test and provides services and tools to the new penetration testers as well as the seasoned veteran, allowing each to focus on the part of the test they excel at. This creates a business process that allows for the performance of penetration tests in a more efficient and standard way [21][22].

6. REFERENCES

[1] Andrey Petukhov, Dmitry Kozlov, Detecting Security Vulnerabilities in Web Applications Using Dynamic Analysis with Penetration Testing, https://www.owasp.org/images/3/3e/OWASP-AppSecEU08-Petukhov.pdf accessed on 22/11/2012

[2] Boris Beizer, Black-Box Testing: Techniques for Functional Testing of Software and Systems, Edition illustrated, Wiley, 1995, ISBN 978-0471120940

[3] Scott Loveland, Michael Shannon, Geoffrey Miller, Richard Prewitt, Jr., and Software Testing Techniques: Finding the Defects That Matter, Programming Series, Editor, and Scott Loveland Editionillustrated, Cengage Learning, 2004, ISBN 978-1584503460.

[4] Software Testing: Principles and Practice, Srinivasan Desikan, Gopalaswamy Ramesh, Pearson Education India, 2006 ISBN 978-8177581218.

[5] Professional Penetration Testing: Creating and Operating a Formal Hacking Lab, Thomas Wilhelm, Syngress, 2009, ISBN 978-1597494250.

[6] Patrick Engebretson, The Basics of Hacking and Penetration Testing: Ethical Hacking and Penetration

Testing Made Easy, Syngress Basics Series Syngress Media The Basics,illustrated,Elsevier, 2011 ISBN 978-1597496551.

[7] Hacking Exposed Web App, Authors Joel Scambray, Mike Shema, Joel Scambray, Tata McGraw-Hill Education, 2006, ISBN 9780070619807.

[8] Harris, Gray Hat Hacking 2E, Tata McGraw-Hill Education, 2008 ISBN 9780070248649

[9] Web Applications (Hacking Exposed) by Joel Scambray and Mike Shema, published by McGraw-Hill Osborne Media, ISBN 007222438X

[10] Hacking Exposed 6, McClure, Tata McGraw-Hill Education, 2009, ISBN 0070147183, 9780070147188

[11] The Unified Modeling Language – A User Guide

[12] Andrey Petukhov, Dmitry Kozlov, Detecting Security Vulnerabilities in Web Applications, Using Dynamic Analysis with Penetration Testing https://www.owasp.org/images/3/3e/OWASP-AppSecEU08-Petukhov.pdf

[13] Williams, Laurie. White-Box Testing. pp. 60-61, 69. Retrieved 13 February 2013.

[14] Ehmer Khan, Mohd (july 2011). "Different Approaches to White Box Testing Technique for Finding Errors". International Journal of Software Engineering and Its Applications 5: 1-6. Retrieved 12 February 2013.

[15] Lloyd Greenwald and Robert Shanley, Automated Planning for Remote Penetration Testing, http://www.lgsinnovations.com/sites/default/files/sites/lgsinnovations.com/files/Automated%20Planning%20for%20Remote%20Penetration%20Testing.pdf

[16] Internal Penetration Testing, http://www.security-assessment.com/page/internal-penetration-testing.htm

[17] http://www.redspin.com/blog/2011/12/22/how-an-internal-penetration-test-can-help-your-organization/

[18] Case Study: Internal Penetration Test, www.trustwave.com

[19] Thomas Wilhelm, Jason Andress, Ninja Hacking: Unconventional Penetration Testing Tactics and Techniques, Elsevier, 2010, ISBN 1597495891, 9781597495899

[20] Rafiq AHMAD,Stephane TICHADOU,Jean-Yves HASCOET, Integration Of Vision Based Image Processing For Multi-Axis Cnc Machine Tool Safe And Efficient Trajectory Generation And Collision Avoidance, Journal of Machine Engineering, Vol. 10, No. 4, 2010,

[21] Marius Corîci on May 5, 2012, HowTo: Penetration test report example [Metasploitable], http://blog.hackaserver.com/howto-complete-a-penetration-test-report/

[22] The Art of Writing Penetration Test Reports, http://resources.infosecinstitute.com/writing-penetration-testing-reports/ accesessd on 1st March 2013

[23] Penetration Testing, Stephen Northcutt, Jerry Shenk, Dave Shackleford,Tim Rosenberg,Raul Siles, and Steve Mancini, http://www.sans.org/reading_room/analysts_program/PenetrationTesting_June06.pdf accessed on 1st march 2013