ISSN No. 0976-5697

Volume 8, No. 5, May-June 2017

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Reverse Engineering and Vulnerability Analysis in Cyber Security

Manu Kumar
Department of Information Technology
Babasaheb Bhimrao Ambedkar University
Lucknow, India

Alka
Department of Information Technology
Babasaheb Bhimrao Ambedkar University
Lucknow, India

Abstract: Most of the incidents response environment and computer forensics investigations cannot be successful accurately or thoroughly without understanding the runtime nature of a binary. Hackers increasingly use customized Trojans that are not detected by any antivirus which can only be analyzed and traced back to the original attacker via reverse engineering. Sometime, many executable programs contain vulnerabilities, such as the use of very weak cryptographic algorithms and the buffer overflows. The most useful way to discover these extreme critical vulnerabilities for binary closed-source programs is to reverse engineer them. Reverse engineering is required in order to understand and define complex binary obfuscation schemes used by copy protection vendors, as well as obfuscation put in place by commercial software vendors. In some different cases Vulnerability Analysis may be the ambition of the test to validate mitigation is in place and the vulnerability is not accessible; while in another mode the ambition maybe to test every applicable variable with authenticated access in an effort to discover all applicable vulnerabilities. Testing may be to find all the vulnerabilities in a hosting system; while in other factors we may need to find all the vulnerabilities on hosts within a given boundary or inventory. Vulnerability analysis performs according to the threat level of a system detected by the penetration testing.

Keywords: Reverse Engineering, Trojans, Vulnerabilities, Cryptographic, Antivirus, Penetration Testing.

I. INTRODUCTION

To improve the Cyber Security Reverse Engineering and vulnerability analysis performs with the help of penetration testing. Penetration testing is required to analyses the system security to prevent from the attackers. Penetration Testing helps in vulnerability analysis and assessment to establish the secure system. Reverse Engineering and vulnerability analysis perform through an automated system or manual system. The automated system provides the list of penetration testing tools that can detect the security breaches and vulnerability through several techniques. Reverse engineering tool helps to find the footprint of the attacker. So we can say that reverse engineering is also useful in the development of a secure system and mitigation of vulnerability[1]. In this article, Reverse engineering and vulnerability analysis both terms are essentially useful to create, develop and update the cyber security [2]. Vulnerability analysis is beneficial for the system –

- Improvement in management of security system.
- Provide weakness points to improve the security system to protect from financial damage.
- It provides analysis to improve the system to protect customer's data over the network.

II. PROCEDURES OF REVERSE ENGINEERING AND VULNERABILITY ANALYSIS

A. Reverse Engineering of Source Code

For Reverse Engineering Source Code, we try to analyze it in order to given it at a higher level of abstraction. The Source code of the design and Reverse Engineering Tools for object oriented languages (Sniff, TogetherJava, and RationalRose) can help us to actually see this design. RationalRose and TogetherJava Reverse Engineering Tools can generate UML diagram from the source code.

This can be done in several ways:

-First, One possibility is to attempt to extract certain kinds of UML diagrams from source code, such as class diagrams, package diagrams, or sequence diagrams.

--Second, An alternative is to view Reverse Engineering Tools as documentation generators, i.e., tools that let we browse the source code at different levels of abstraction, including cross-reference information, graphical representations of methods, class, and package interactions, comments extracted from the source, and so on.

B. Reverse Engineering without the Source Code

You may be allowed to recover a new source text from an executable file without the source. For example, the *source recovery company* does some very clever tricks to recover Cobol Language sources from mainframe loads -- using all possible knowledge about compilers used and all their weird flags. A COBOL program is partitioned into divisions, sections, paragraphs and sentences. COBOL supposes hierarchical data records. This stable fittest most of data of the time: Tabulators, Unit Record Systems, Punch Cards, Magnetic Tapes, Line Printers, etc.

C. Identifying Vulnerabilities

Vulnerabilities can be identifying through penetration testing. Penetration testing gathers a number of threats level in a particular project. When the huge number of vulnerabilities found in the system, the system may be harmed by the attackers. Vulnerabilities analysis helps to reduce the number of negative facts of a system through the mitigation of vulnerability. Vulnerability assessment helps to reduce the no of vulnerability in the system.

D. Vulnerability Analysis

- Human interpretation is required to make results meaningful
- That interpretation includes
- Assessing risk presented by vulnerabilities
- Comparing the results to security policy
- Verifying vulnerabilities
- Prioritizing vulnerabilities
- Assessing risk and prioritizing vulnerabilities
- A subjective process but you can be objective by using CVSS
- Common Vulnerability Scoring System (CVSS)
- Researching vulnerabilities
- The Common Vulnerabilities and Exposures (CVE) numbers
- Causes of errors during vulnerability analysis
- Environmental Issues
- Timing Issues
- Privilege Issues
- Tool Issues
- People/knowledge Issue

E. Process of Reverse Engineering

Reverse Engineering process is used to find the footprint of attackers and it also helps in a finding of weakness points of System that is harmed by the attacker. Reverse Engineering process performs step by step procedures that are given below.

i. System To be Structured
ii. Manual or automated analysis
iii. System Information Store
iv. Document Generation
v. Traceability matrices, Data Structure, and Program Structure Use.

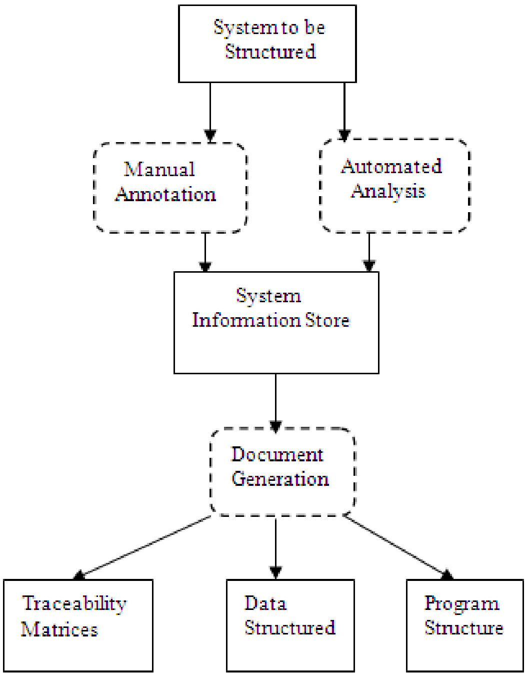


Fig 1.0 Process of Reverse Engineering

A system to be structured: The system which is harmed by the attacker need to be structured in a specific manner. The system is examined in a different environment.

Manual or automated analysis: Structured System analyzes from bottom to top approach or backward approach. This analysis performs by the manual or automated tools.

System Information Store: The Information regarding illegal activities or unwanted operations store for investigates the attack.

Document Generation: The Document Generation step generates document regarding cyber-attacks or illegal activity.

Traceability Matrices, Data Structured and Program Structure are used: These steps are used to trace the activity or footprints of the attacker and find the source of attack [3][4].

III. TECHNICAL RISK OF REVERSE ENGINEERING

There is various Risk of reverse Engineering that are given below and each risk is associated with the others.[We are focusing only technical risk of reverse Engineering.

- Confidentiality Risk
- Integrity Risk

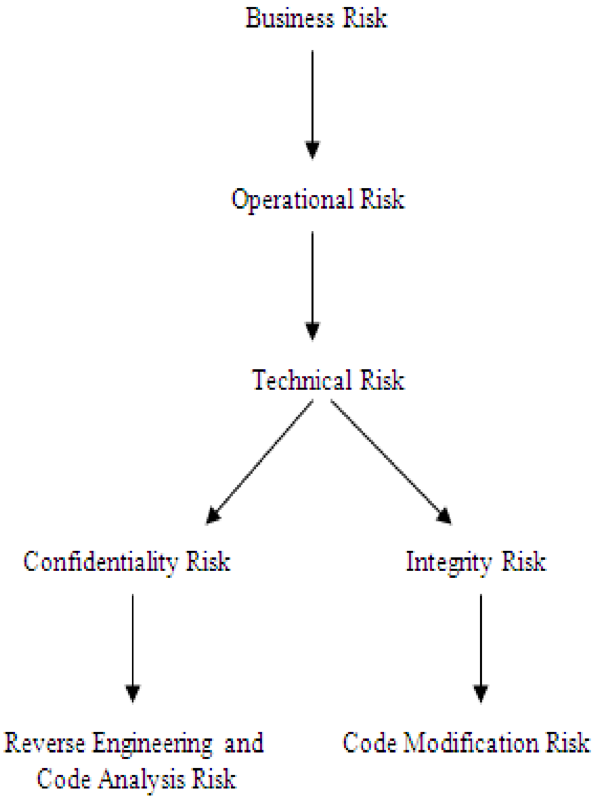
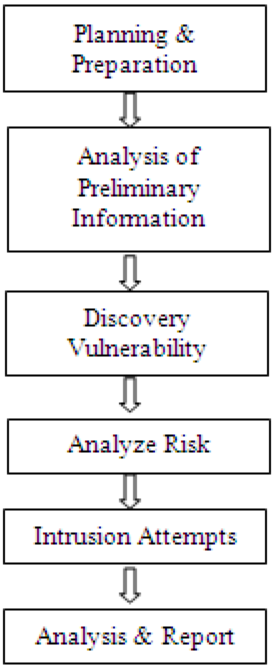


Fig. 2.0 Risk Overview

Confidentiality Risk: when we are applying reverse engineering to find the footprints of the attacker, there is a confidentiality risk for securing the confidential information.

Integrity Risk: In Reverse Engineering Process, there is an integrity risk. When we apply the reverse engineering process there is a risk of code modification [5][6].

IV. METHOD OF VULNERABILITY ANALYSIS



Planning & Preparation: The First method of Vulnerability Analysis is Planning & Preparation. It defines the objective and goal of the Vulnerability Analysis.

Analysis of Preliminary Information: It analyzing the preliminary information of Vulnerability and then proceed to the next method of Vulnerability Analysis.

Discovery of Vulnerability: Discover the vulnerability by some automated tools or manual tools.

Analyze the risk: After Discover the Vulnerability, we Analyze the risk during analysis of Vulnerability.

Intrusion Attempts: we attempts few intrusions on a system and then analyze the Vulnerability. When a verification of potential vulnerability is needed, this step must be performed.

Analysis & Report: After Analysis we create a report of an overall summary of vulnerability detection [7].

V. USE OF VULNERABILITY AND REVERSE ENGINEERING TOOLS

We are Using Different Type tools to find the vulnerability in a web application and perform penetration Testing.

- 3d trace out
- Acunetix web Vulnerability Scanner
- Angry IP Scanner
- Brutus
- Global Network Inventory
- Httprint
- Megaping
- Netscan
- Port scanner
- Path Analyzer Pro
- Nmap
- Spy-net
- Superscan

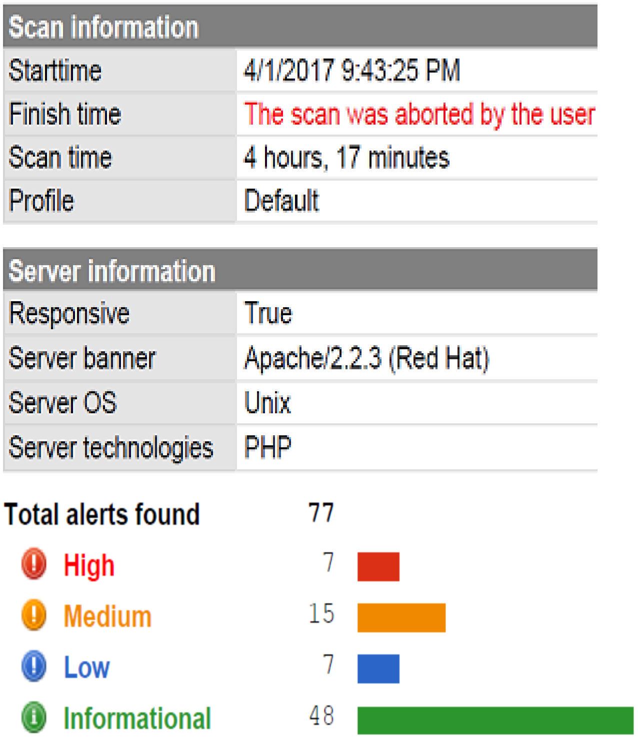
Screenshots of Vulnerability Analysis on ebc.com: ebc.com is a website of Book Company and this website built for sell the books online like another E-commerce website. The Vulnerability Analysis Perform on this website for only study purpose.

acunetix threat level

Level 3: High

Fig.3.0 Threat level

The Threat Level decided by the scanning tool and the scanning detail is given below-



Scan information	
Starttime	4/1/2017 9:43:25 PM
Finish time	The scan was aborted by the user
Scan time	4 hours, 17 minutes
Profile	Default

Server information	
Responsive	True
Server banner	Apache/2.2.3 (Red Hat)
Server OS	Unix
Server technologies	PHP

Total alerts found	77
High	7
Medium	15
Low	7
Informational	48

Fig 4.0 Scan Details

The Vulnerability details of this website are given below:

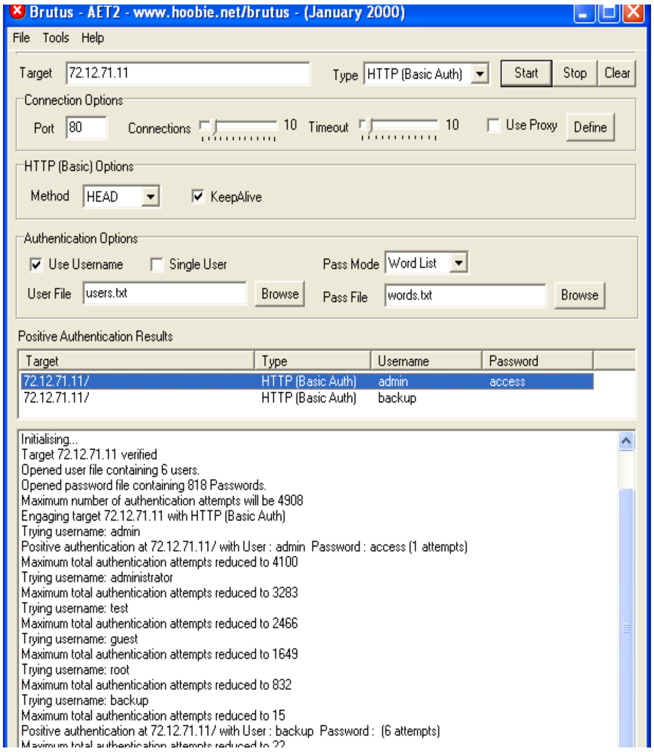
- Cross Site Scripting
- Apache 2.x version older than 2.2.6

- Apache HTTP Remote Denial of Service
- Apache httpOnly Cookie Disclosure
- HTML form without CSRF protection

The Vulnerability Analysis for this portal performs in 4Hours, 17 minutes and it analyses only 5.20% percent. After 5.20 % Abort the Vulnerability Analysis Process because we are getting the most of the vulnerability in this small duration scanning. It is just the preview of gathering the vulnerability in the system from the applying of penetration tools.

Recommendations: - the recommendation to mitigate the vulnerability from the system is given below-

- The script should filter metacharacters from user input.
- Upgrade Server
- Use Remote denial of service
- Implement csrf (Cross Site Request Forgery) countermeasure if csrf protection requires.
- Sensitive information should transfer to the server over encrypted connection(HTTPS)



Brutus - AET2 - www.hoobie.net/brutus - (January 2000)

Target 72.12.71.11 Type HTTP (Basic Auth)

Connection Options: Port 80 Connections 10 Timeout 10 □ Use Proxy

HTTP (Basic) Options: Method HEAD ☑ KeepAlive

Authentication Options: ☑ Use Username □ Single User Pass Mode Word List

User File users.txt Pass File words.txt

Positive Authentication Results

Target	Type	Username	Password
72.12.71.11/	HTTP (Basic Auth)	admin	access
72.12.71.11/	HTTP (Basic Auth)	backup	

```
Initialising...
Target 72.12.71.11 verified
Opened user file containing 6 users.
Opened password file containing 818 Passwords.
Maximum number of authentication attempts will be 4908
Engaging target 72.12.71.11 with HTTP (Basic Auth)
Trying username: admin
Positive authentication at 72.12.71.11/ with User : admin  Password : access (1 attempts)
Maximum total authentication attempts reduced to 4100
Trying username: administrator
Maximum total authentication attempts reduced to 3283
Trying username: test
Maximum total authentication attempts reduced to 2466
Trying username: guest
Maximum total authentication attempts reduced to 1649
Trying username: root
Maximum total authentication attempts reduced to 832
Trying username: backup
Maximum total authentication attempts reduced to 15
Positive authentication at 72.12.71.11/ with User : backup  Password :  (6 attempts)
Maximum total authentication attempts reduced to 22
```

Fig. 5.0 Port Scanning

Port Scanning: Port Scanner scan the port where the Attacker can perform the attack and find the vulnerability in a particular system. Above Figure Shows that The Port Scanner scans the port like FTP, HTTP and find the details about this usable port.

VI. CONCLUSION

Hackers, competitors, and software pirates are working overtime to find and exploit vulnerabilities in our Software Code, our infrastructure, and our business operations. We Find and detect Vulnerability via powerful reverse engineering tools, white box testing tools, etc. Reverse Engineering and Vulnerability Analysis in Security of Web application over the server is mandatory to detection of hackers or intruders. Sometimes investigators need to investigate many cyber-crimes cases through reverse engineering and vulnerability analysis in cyber security.so we can say that Reverse engineering and Vulnerability Analysis is also a part of the Cyber security.

REFERENCES

[1] Aileen G. Bacudio, Xiahong Yaun, Bei-Tseng Bill Chu, Monique Jones, “An Overview Of Penetration Testing,” Dept. of Computer Science, North Carolina, USA, Published in Nov 2011.

[2] Philip S. Anton, Robert H. Anderson, Richard Mesic and Michael Scheiern, “Vulnerability Assessment & Mitigation Methodology,” National Defence Research Institute, Published 2003 by RAND.

[3] Darren Mutz Christopher Kruegel, William Robertson, Giovanni Vigna, Richard A. Kemmerer on “Reverse Engineering of Network Signatures,” University of California, Santa Barbara

[4] Chikofsky, E. J. & Cross, J. H., II (1990). "Reverse Engineering and Design Recovery: Taxonomy". IEEE Software 7 (1): 13–17. doi:10.1109/52.43044.

[5] A Survey of Reverse Engineering and Program Comprehension. Michael L. Nelson, April 19, 1996, ODU CS 551 – Software Engineering Survey.arXiv: cs/0503068v1

[6] Reverse Engineering edited by Linda M. Wills, Philip Newcomb (Springer Science & Business Media. Copyright)..

[7] “Vulnerability Analysis and Defense for the Internet” edited by Abhishek Singh (Springer Science & Business Media. Copyright.)

[8] David Brumley, “Analysis and Defense of Vulnerabilities in Binary Code,” School of Computer Science, Camegle Mellon University, CMU-CS-08-159, Sep 29,2008

[9] Reverse Engineering By C. Werner Dankwort (Springer Science & Business Media. Copyright.)

[10] Guide to risk and vulnerability analyses, Editors: Jonas Eriksson, Anna-Karin Juhl, Publ.nr MSB366 - March 2012, ISBN 978-91-7383-208-3

[11] Assets, Threats and Vulnerabilities: Discovery and Analysis, A comprehensive approach to Enterprise Risk Management By Symantec Corporation.