International Journal of Engineering & Technology, 7 (1.1) (2018) 411-413

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

A systematic review of vulnerability analysis & penetration testing tools

K. Raja Sekhar [1] *, Pavanasurya. M [1], Komal Bharti [1], Dhanya G [1]

[1] *Koneru Lakshmaiah Educational Foundation, Vaddeswaram Department of Computer Science and Engineering*
**Corresponding author E-mail: bhartikomal23@gmail.com*

Abstract

In Computer Security, the term vulnerability refers as a flaw in the system which creates a hole, giving an attacker a chance of taking control over the system. Any Software, Web application or anything related to computer product is vulnerable to attack in different ways like code stealing, sniffing of packets, hijacking the network, making the system compromised etc. In order to avoid such attacks a constant check has to be done and the check has to be done through various Pen testing tools. Penetration tools are one which is used to perform security check on an application to find the presence of exploitable vulnerabilities. In this paper, we look over the penetration tools like CODEPULSE (the code stealer), ETTERCAP (the Sniffer and Hijacker) and made a systematic review of various websites which are vulnerable to SQL Injection and Cross-site Scripting.

Keywords: *Social Engineering; Penetration Testing; Exploit; SQL Injection; Cross Site Scripting; Glass Box Testing.*

1. Introduction

The term Security has become more popular now-a-days due to the advancements of various technologies. It stands as one of the key constraint in the life of a application. In the world of computers, attack is a way to destroy someone's data. In order to gain someone's unauthorized information attacker attacks on user data. Attack is very complicated and it is difficult task for researchers and companies who work in the field of security to prevent it. Attack can be active or passive. When attacker only steals the data of user without destroying user's information, it is called passive attack. When attacker destroys the system of user and harm the information of user then it is called active attack. In order to keep up the security issues, we work on the various tools like Code Pulse and Ettercap.

Code Pulse, shortly as a code organization stealer. It is also a Glass box testing tool – it is a runtime testing tool where the internal structure and working of the test application can be seen. It is an open source Glass box testing tool which aims mostly at finding the code covering of penetration testing aspects. After finding the vulnerable area in the software we apply different type of method- ology to test that area and recover from vulnerability.

1) Code pulse directly integrates with dependency check to notify automatically when a third-party dependency has a known vulnerability.[2]
2) With code pulse we can spread testing in many session, there is no need to do whole testing in one session.
3) It shows detailed coverage information about the software from a high level view all the way to single methods.

Ettercap shortly called as the basic sniffer and hijacker over a particular network. It sniffs the packets that are transferred from the host, basically it is eavesdropper. Ettercap is a tool developed by Alberto Ornaghiand Marco Valleriand it is normally used for MIMT's (man in the middle attacks) on a LAN. Ettercap is also provided with pure Graphical Interface, for those who are not familiar with the usage of CLI (command line Interface).Ettercap with driftnet performs some of the real-time attacks. Both to be run on the same host. Ettercap is also able to perform similar attacks against the ARP protocol by making itself as "man in the middle" and, once it is made as this, it will be able to:

- Change, truncates the data in present in the connection over a net- work.
- For protocols like SSH1, POP, FTP, HTTP etc. the passwords can be easily detected.
- Unauthorized SSL certificates can be provided for the HTTPS registered sites and many more.

International Journal of Engineering & Technology

2. Related work

2.1. Code pulse

In designing this system, we have two important aspects one is how effectively to recognize the coverage data and the other is how effectively to interact with others.

2.2. Coverage identification

For a web application the testers is connected with many web page contents that form the starting of the application. But it is not necessary to translate sitemap and corresponding URL directly into basic building blocks. Considering the web application along with URL which generates the report of the form PDF as input for the URL which is as same as it. This PDF and XML are used to call various codes with same URL entry point.[2] In penetration testing procedure after identifying the vulnerabilities the next step is to marked the is- sues for development team. Understanding of this coverage data at the various source code levels will gives the inventors the best idea about vulnerable part of the application.

Vulnerable part of the application is identified by the best remediation resources.
Code Pulse coverage identification will be done at the source level. From a tester and user's perspective the data would detect both URL map of an application and source code coverage.

2.3. Coverage communication

Providing coverage in the present real-time is a challenging task due to the huge data volume. Tens, hundreds of method calls will result for a single web re- quest. Providing the user with excess coverage of in- formation is unfaith because the abundant data relies in cognitive load and instead of enhancing the testing process, it ruins it. Visualizations are used to summarize huge volume of data meaningfully. Processing the data visually is more effective than any other alternatives. Identifying the data patterns visually is efficient than using the combination of relevant data filters and inter- actions. Regarding communication of coverage data it can be concluded that the data should be presented to the testers as quickly as possible and visualizations should be done to improve data reliability.
Code Pulse aims to convert the black box testing concept of penetration into glass box testing. While conducting the test, the penetration tester's goal is to visually represent the code coverage in real-time. There are two parts of the code pulse architecture defined:

1) Instrumentation, which is responsible for monitoring code coverage during runtime of a application.
2) Visualization and front end user interface who's Functionality to represent the coverage information in an understandable manner.

2.4. Instrumentation

The main requirement of instrumentation is to have minimum impact on the resources of target application. To meet these requirements, the system has to be designed to perform minimum work in the same execution context of target application. Therefore, while using before a client/server model the monitoring parts was set into two distinct parts. The agent will listen to the execution process and will send the total coverage information to the server for the process of processing and storage, this happens while running the target application. Out of the number tests which are conducted on the various performances of the instrumentations, the results varied will also depends on the application's nature.

2.5. Visualization

Tree map visualization was used for the representation of the coverage information in Code Pulse. The two types of nodal points were shown in the tree map. Java package nodal points will be displayed in slim labels and are serving as a point of references as the visualization. While all other nodes represented either using Java classes or JSP's (Java Server Page).Visualization is focused by the node types and are sized by byte code instruction count. Nodes are colored grey in the default state and the node shading changes when coverage activity occurs to a method or a JSP file to show that it had been covered during the testing process. Tree map serves many purposes

1) Real-time activity high lighting: The methods are highlighted in real-time when they are called within the tree map to let the user know which parts of the application have the impact of their testing activity.
2) Served as one of the persistent coverage blinker for tested methods and files.
3) Used to drive the coverage overlap analysis. Labelled markers are used to segment the coverage data in Code Pulse. The tree map changes the coloring of nodes which indicates the nodes that are covered in a single segment and which are overlapped by using the multiple markers.

3. Ettercap

Ettercap is a open source tool which can be run on many Unix operating system like MAC, Solaris, BSD and Microsoft Windows. It can be used to intercept traffic.[7] It can capture password and it also performs active eavesdropping against a number of common protocols. It is a pen testing tool which can perform man in middle attack on local area network (LAN), and this can be used for ARP spoofing and ARP poisoning. What do you think about ARP? ARP (Address Resolution Protocol) is a process by which networking and contact information of a computer can be resolved. When a computer is connected to another computer in local area network, an ARP entry is made. ARP change the IP address into MAC address and tells that computer is connected to legitimate computer when in reality it is connected to malicious computer.
By using man in middle attack data can be analyzed during transfer to the network, and can be changed in middle of transmission. Data can be injected to the server's application to maintain the connection alive.
Ettercap can work with modules like
International Journal of Engineering & Technology
ARP-based: This is used for the process of sniffing packets between the 2 hosts on a switched network.
Public ARP-based: This is very much useful for sniffing or hijacking the packets from a user to all the hosts present.

3.1. ARP – based filtration

The IP based Filtration means the attacker who wants to sniffs the packets can make use of this tool. Initially, the attacker must be in the network in which the victim is present, then only it is possible to sniffs those packets. After the attacker changes the network to the same as victim then he has to scan for the ipv4 address through the tool.[5] This Process is done until the victim's machine was caught, this can be done through the help of other tool like NMAP(Network Mapper). After finding the address, the attacker has to look after the GATEWAY. Both the machines are to be in the same, or else it has to be changed until they are in same. After all these constraints are satisfied, then attacker will start the ARP (Address Resolution Protocol) Spoofing by adding the target machine IP address and the Gateway through which both are connected

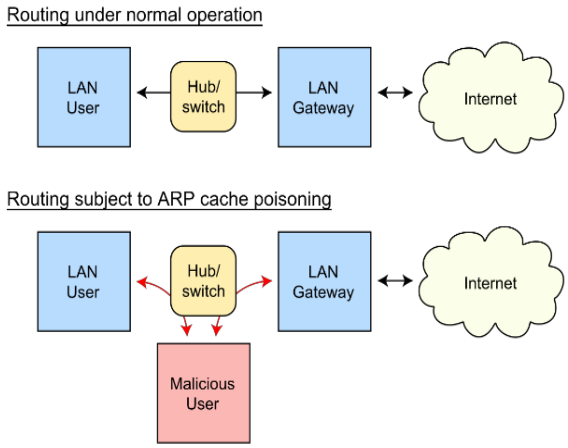

Fig. 1: Routing Under Normal Operation and Routing Subject to ARP Cache Poisoning.

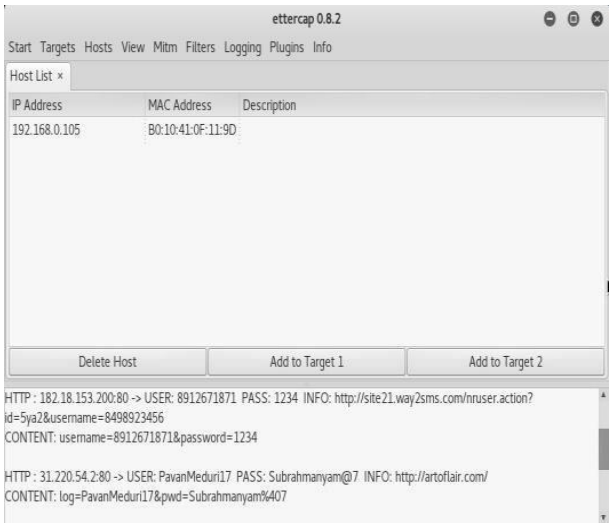

Fig. 2: In This Way, the Packets are Being Sniffed to the Host Machine.

3.2. PublicARP-based DNS spoofing

DNS spoofing means by-passing the traffic to a particular server. This is because of various reasons like stealing of passwords, to increase their traffic etc. [5] this can also be implemented through the ETTERCAP tool.

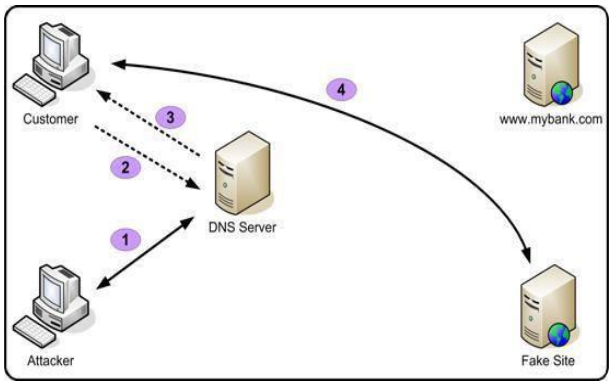

Fig. 3: Attacking Scenario in Network.

Initially, this process is started as the attacker will replicate the most famous site and create its name as the original like the www.facebook.com is replicate it with domain name as "www. Facebook.com". For a normal user it seems to be as a same but both the sites are different. [4] This whole process is called Social Engineering. When the user types the www.facebook.com as the victim's IP address is spoofed, he will re-direct to the site hosted by the attacker and then attacker steals what the data he wants. Ettercap with the help of Drift net is used to capture what the user is doing in his/her system.

4. Proposed methods

Modified Approach using Ettercap Tool:
Both the Tools, Code Pulse and Ettercap are active tools. It is possible to sniff only those packets the are sent by the local system. In order to sniff those packets anywhere, it is impossible with Ettercap itself. With the help of other tools NMAP, Driftnet and Metasploit it is made possible to sniff those packets. With the NMAP we have to search for the victim is connected to which network and the attacker has to move over to that network to start the sniffing process. If the attacker is not in the same network, he will use the Metasploit tool and try to get his/her information by various social engineering techniques. After the attacker is connected to the same network, then the attacker can start the attacks like ARP Poisoning, DNS Spoofing and Hijacking. With the help of drift net with Ettercap we can able to see what the user is trying to open and with the help of Metasploit with Ettercap we will be able to inject the virus and trojan files into the victim's machine and steal the data.
The limitations of using the above mentioned tools is they all work perfectly when they try to sniff packets over the HTTP sites. If any attacker wants to sniffs the packets over HTTPS sites, there are two possibilities, the sniffed packets are ENCRYPTED using SHA – Encryption and can't be decrypted and the other possibility is the browser will stop the sniffing process, by mentioning that some attacker is trying to sniff your personal information. In order to overcome these scenarios the attacker has to send a Torjan or virus with the help of Metasploit using Ettercap
International Journal of Engineering & Technology

5. Conclusion

Any digitalized medium can be compromised now- a- days with the help of various tools present. The main agenda of this work is find the various vulnerabilities present in/on a website and shown how the attack is going to take place and what are the ways that user can make themselves secure over the net- work are mentioned in this paper. For the reason of checking we had done our implementation on various websites with prior permissions taken.

6. Vulnerable Sites

Vulnerable sites for SQL injection: 40, 60,000 (approximately)
Vulnerable sites for Cross site scripting: 67,000(approximately)

References

[1] Improving Accuracy of Applications Fingerprinting on Local Networks using NMAP-AMAP-ETTERCAP by Waheed Ali H. M. Ghanem School of Computer Sciences University Sains Malaysia (USM) and BahariBelaton School of Computer Sciences University Sains Malaysia (USM) Penang, Malaysia.
[2] Code Pulse: Real time code coverage for Penetration Testing Activities by Hassan Radwan, Kenneth Prole Secure Decisions Division Applied Visions, Inc. Northport, NY, USA.
[3] Hassan.Redwan, Ken. Prole "code pulse: Real time Code Coverage for penetration testing Activities" IEEE.2015.
[4] Security Quality Assurance through Penetration Testing by Kamran Shaukat University of the Punjab, Jhelum Campus. Jhelum.