Volume 6, Issue 4, April 2016 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Multidimensional Attack Mechanism Based Detection Tool Using Crafted Queries

Mohamed Suhail B[*]

M.Tech Student, Department of CSE, B S Abdur Rahman University,
Tamilnadu, India

Abstract— Cyber security is a key research area in the digital technology arena. New vulnerabilities and exploits are regularly developed to cause severe damage to popular web applications. So performing penetration testing has become a mandatory process in the technology world. Instead of penetration testing with conventional defensive techniques to guard against attacks, the proposed tool is an offensive attack mechanism that will try to penetrate into network and server hosts of any given web application and attempts to test the security of the web application. The tool focuses on port scanning and packet sniffing before performing database attack. Queries are sent to the respective web servers to determine services and ports that are used by the web application, also the availability of SQL Injection vulnerability is tested by sending crafted SQL injection packets. The test results of the program may reveal critical services, web server responses and vulnerabilities in websites that are tested. The multidimensional attack mechanism focuses on collecting network, server host information and application data to test the security perimeter of the given web application. After reporting these security vulnerabilities adequate countermeasures can be taken to secure the web application.

Keywords— Cyber security, database attack, packet sniffing, ports, SQL injection, vulnerability

I. INTRODUCTION

Cyber security is primarily dealt with sophisticated attack mechanism that if once successfully executed prevents the legitimate user accessing the software, web or mobile application. A computer security vulnerability is any weakness in a computer software, website, operating system or mobile platform that can be exploited by malicious attackers for their own personal gains. Web sites and web based applications are unfortunately prone to security vulnerabilities. The background networks that support web applications are prone to serious security attacks. Any system with multiple open ports, multiple services and supporting multiple scripting languages is vulnerable simply because it has several entry and exit points. Web applications face several intelligent attacks that is a combination of both multiple vulnerabilities in networks and web database servers.

Web application must be periodically tested with an intention to find security issues so that security attacks and breaches can be prevented. Penetration testing is an assessment for checking the security of digital system using real time attack programs. It answers the question of whether a computer based information system contains possible weak components and vulnerabilities which are targeted. It also includes the process of compromising a medium to large scale network after getting prior permission therefore with good intentions. This type of ethical hacking is always legal and it is focused on securing and protecting Internet Protocol systems.

Port scanning is a major activity in the penetration testing process which is the act of systematically scanning a computer's ports. Since a port is a pipe where data goes into and out of a computer, port scanning identifies open doors to a computer. Port scanning has legitimate uses in troubleshooting network issues but this technique also can be used for malicious purpose if someone is looking for a vulnerable access point to break the application.

A packet sniffing sometimes referred to as a network monitoring or network analysing that can be used legitimately by a network or system administrator to monitor and troubleshoot network traffic. Using the information captured by the packet sniffer suspicious packets can be determined and the output of sniffing process can be used to pinpoint bottlenecks in the network and help maintaining efficient network data transmission. The process of packet sniffing involves capturing the packets of data that pass through a given network interface such as wireless or Ethernet. A packet sniffer would only capture packets that were intended for the host in question but if the packet sniffer is placed into promiscuous mode it has the capability of capturing all packets traversing the network regardless of destination address.

Both port scanning and packet sniffing can be used to find critical services and vulnerable ports of web applications. The most common type of attack on web application is SQL injection attack, an attack technique that takes advantage of the syntax of SQL to inject malicious commands. This attack contains a mixture of SQL instructions statements and the payload which contains the strings to be inserted that can manipulate the original code. Any web application may construct a query manually or automatically by simple string concatenation. Such an approach is not recommended if the query comes from an untrusted or malicious source because if the user name or description has double quotations, that malicious source can then provide command to exploit the vulnerability instead of value data.

Further complicating the problem, the SQL language provides a comment operator "—", which causes the SQL server to ignore the rest of the line so that the original code can be ignored.

The multidimensional attack mechanism based detection tool is a simple alternative to the existing complex web application firewalls (WAF) that protects websites by handling its input and output and the access to and from the application. WAFs run as server plug-in, cloud-based web service which can inspect every HTML, HTTPS, SOAP and XML-RPC data packets through intelligent and customizable inspection, They are able to prevent attacks such as SQL injection, session hijacking and buffer overflows. Network firewalls devices, unified threat management (UTM) systems, intrusion detection systems are often not capable of doing. A WAF is also able to detect and prevent new unknown attacks by monitoring for unfamiliar patterns in the web traffic data provided by the results.

II. RELATED WORK

It is unreliable to produce complex applications without defects. The open web application security project report listed the top 10 critical web application security risks that has SQL injection at the top and the list continues. SQL injection attacks take the advantage of unchecked in out fields as the web application interface to maliciously modify the SQL query sent to the back end database. By exploiting the vulnerability the attacker is able to inject code. Strong typed programming languages such as python, ruby are having less security problems comparing to their counter parts [1]. A denial of service (DoS) is an attempt by malicious program to completely disable the availability of services and resources to authorized users. DoS attacks are SYN packets flooding, teardrop, smurf, black holes DoS attacks exploit weaknesses in IP protocols, operating systems software, and web applications [2]. The services to the legitimate users may be disturbed completely by the denial of service attacks. In Distributed denial of service (DDoS) attack scenario the hosts are computers who are unaware that they have been used for attacking by the attackers. Port scanning can be used to identify the vulnerable services that cause the denial of service attack to be successful.

An attacker must have gained maximum knowledge about the target host. It is important to get an overall sketch of the network and details about the users and gain information about servers, administrator contacts and IP ranges can be collected [5]. During the information gathering phase different kind of tools are be used for mapping the network and vulnerability scanning tools are the commonly used. They can be of great help later on during the attack phase or to get an overview about the network. At the end of the reconnaissance phase, an attacker will have chunks of information about the web server to be compromised [7]. With all these pieces of information, a promising attack path can be constructed.

Signature based detection is commonly used by virus digital software vendors who will compile a database of different malicious programs signatures that are popularly known as virus definitions. These virus signatures are matched with local files, remote files, web downloads depending on settings chosen by the user. This database is constantly being updated for new signatures of newly exploited vulnerabilities [6]. For detection of exploited vulnerabilities this technique requires a virus signature to be present in the signature database because of this purpose all the security products vendors are frequently updating their signatures.

III. DETECTION TOOL

A. Attack Mechanism

The multidimensional attack mechanism comprises of various techniques that are used in the detection tool which tests the web application programs in order to find vulnerabilities in a much simpler way. It probes the given web application using the homer page uniform resource locator (URL) for basic server details, location, firewall details, server host information, open ports and services and database vulnerabilities. The detection tool's architecture is shown in the Fig. 1.

B. Port Scanner

This module is used in the mechanism for network exploration and security auditing in the multidimensional attack framework. It used to rapidly to scan large networks, although it works fine against limited number of hosts. The port scanner uses raw network packets to determine what hosts are available on the network, details of the applications such as name and version, that utilizes the service, what operating systems and operating system version they are running, types of packet filters and firewalls that are in use, and dozens of other characteristics. While the scanner is commonly used for security audits now. It can be used for routine tasks such as network inventory, managing service, and monitoring host active time. The output from port scanner module is a list of scanned targets, with additional information on each depending on the options used by the scanner.

Port scanner reports the state combinations open filtered and closed filtered when it could not figure out which of the states explains a port. The port table may also include software version information when detection has been requested. When an IP protocol scan is requested, the scanner provides information on standard network protocols rather than listening ports.

C. Sniffer

Sniffer is the default network protocol analyser that is used in the multidimensional attack mechanism based detection tool which allows to analyse the network at a microscopic level for web application vulnerability vulnerabilities by collecting the HTTP packets that are sent and also it collects the web server responses. Sniffer module is used in this mechanism to perform deep inspection of hundreds of protocols. Live capture is done followed by performing analysis with a standard three-pane packet browser option provided by the tool can be used to dissect the structure of potentially vulnerable HTTP GET POST messages.

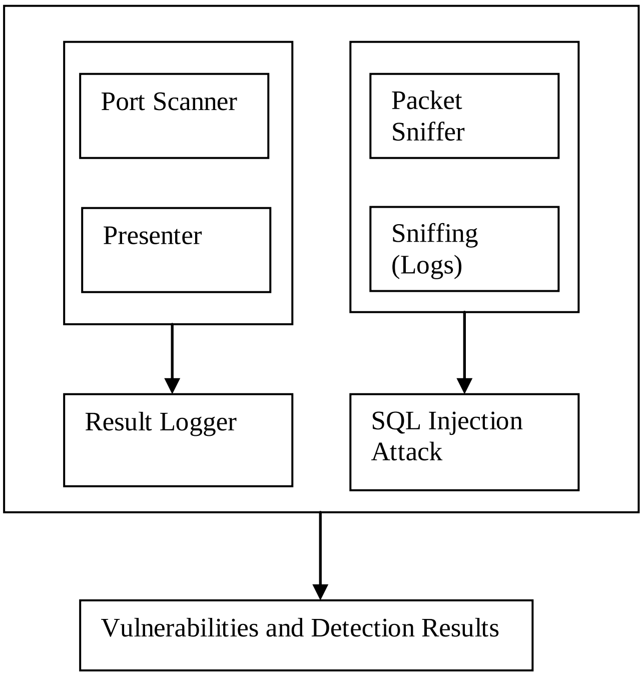


Fig. 1 Architecture of Multi-Dimensional attack mechanism implemented in Detection Tool

D. Database Module

The given URL is sent an HTTP request that contains the SQL injection code encoded as "%027" which attempts to comment the rest of the SQL query that the web application internally executes to fetch the data from the database. The HTTP request will be successfully processed by the web application then the query is parsed. If the web program does not have adequate defence mechanism against SQL injection attack then the query is executed by the database with the modification made by the attack mechanism. This will yield result of the query to the HTTP response object that the attack program reads. The result will reveal the SQL database details thus proving the SQL injection vulnerability. In the case of Microsoft Access used the backend database of the web application then the program will send crafted SQL injection queries that will display the names of the drives that the database server is currently using indicating a serious information disclosure vulnerability.

E. Logger

All the results of the port scanner module is written into the logs using sub process logger functions, similarly all the packets that are captured by the sniffer module is written in the logs with detailed information for further analysis of the behaviour of the web application that is being tested by the detection tool. The list of open ports and services are recorded with respective IP addresses and host information of the web application. Finally the Logger records the database module activity, the crafted queries that the detection tool sends is a small sized HTTP request requiring no time more than the average HTTP browser GET and the response to the crafted packet which may display the vulnerability status.

IV. IMPLEMENTATION AND PENETRATION TEST

A. Three Port Scanner based Scanning

Three port scanner threads are implemented in the detection tool that targets the operating system, all services and ports respectively and one more thread for verbose scans. The three port scanner scans 1,000 TCP ports on the web application. Port scanner will determine all ports either into the open or closed states category. It divides ports into six states: open, closed, filtered, unfiltered, open filtered, or closed filtered. A scan from the same web application as the target may show port 135/TCP as open, while a scan search at the same scenario with the same options from across the Internet might show that port as filtered. Statuses of ports given by scanner are mentioned below.

- Close - A closed port is accessible that it receives and responds to scanner's probe packets, but there is no process listening on it. It can be helpful in showing that a system is up on the network, host discovery, or ping scanning, and as part of OS detection.
- Filtered- The scanner cannot find out whether the port is open because packet filtering is implemented which prevents the probes from reaching the port. This filtering may be from a Intrusion prevention system, router rules, or host-based firewall software
- Unfiltered state – This status indicates that a port is accessible, but the scanner is cannot determine whether the consent port is open or not. Only the acknowledgement scan can be run, which maps firewall rule sets and classifies ports into this state.
- Open Filtered - Ports that fall in this category are either open or filtered. This occurs for scan types in which open ports doesn't give any response. The lack of response could also imply that a packet filter dropped the scan.

B. Packet Sniffing

Instead of integrating third party sniffers, the detection tools has its indigenous sniffer thus reducing the time for function calls, arguments space and the time and space for parsing the third party sniffing tools' packet results. First the interface for sniffing is configured to capture packets either on Ethernet interface or wireless LAN interface. Then advanced options can be implemented like capturing a file, resolving MAC addresses, resolving DNS names, or limiting the time and size of the capture are enabled. Many of these options when implemented can improve the performance of sniffing. For example, certain settings can be adjusted to avoid name-resolution issues, as they will otherwise slow down capture time and generate large numbers of name queries. Time and size limits can also limit the sniffing process. Every packet that is captured on the network contains the time of the packet, the source and destination network addresses, protocol used and some information about packets. There are varying levels of details about each layer of information contained within the packet.

C. Database Testing

The database testing phase comprises of sending simple crafted SQL query embedded into HTTP request that is send using the web methods libraries available. If there is an SQL injection vulnerability the database throws an error. Based on this error message the presence of the critical vulnerability is determined. The message is sent back to the detection tool using HTTP response and the result is parsed. After performing automatic analysis on the result retrieved by sending crafted queries the database server's vulnerabilities are determined. Based on the parsed result the detection tool will display the database server being used which can further make the penetration testing process much easier.

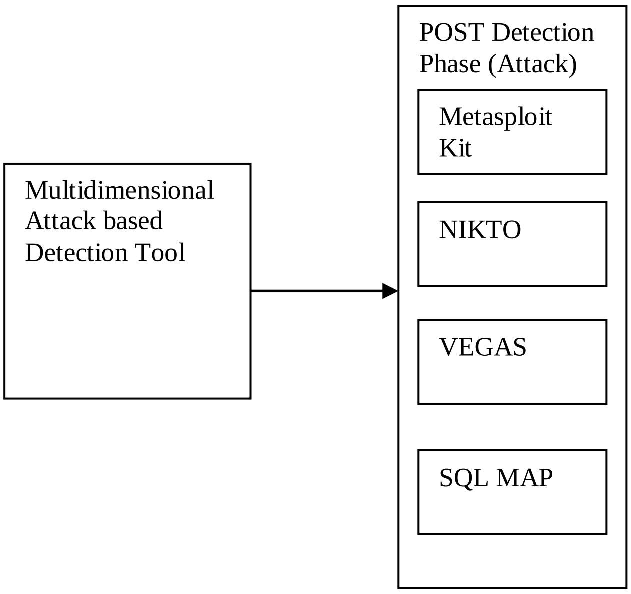

Multidimensional Attack based Detection Tool
POST Detection Phase (Attack)
Metasploit Kit
NIKTO
VEGAS
SQL MAP

Fig. 2 After the detection Tool's successful test penetration toolkits are integrated used to attack

D. Post detection phase and Comparison

In the post detection phase the detection tools is integrated with SQLMAP which is used to exploit the SQL injection flaws and attack the database servers as shown in Fig. 2. The powerful detection engine is used in the framework for detecting the firewall presence behind the database server. SQLMAP has unique features for the penetration testing and a broad range of options from database reconnaissance, over information leakage from the database, to accessing the underlying file system of the server and executing arbitrary commands on the operating system through out-of-band connections are required to successfully attack a web application based database. It is used by the attack mechanism to support for six SQL injection techniques such as Boolean, time based, error based, UNION query, stacked queries and out-of-band.

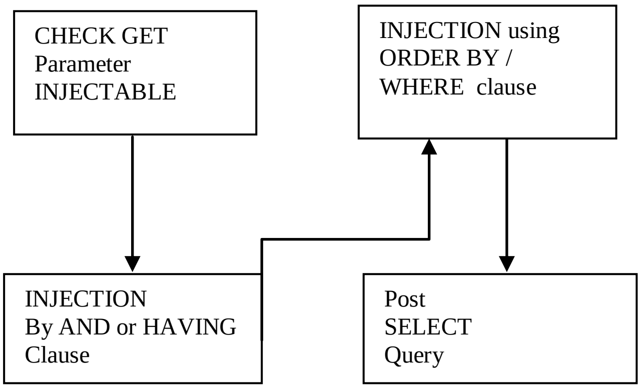

CHECK GET Parameter INJECTABLE
INJECTION using ORDER BY / WHERE clause
INJECTION By AND or HAVING Clause
Post SELECT Query

Fig. 3 SQLMAP working mechanism

The queries can be directly connected to the database and executed without passing via a SQL injection, by providing database details, port and database name that are self-identified by the SQLMAP program as shown in Fig. 3. Few existing tools possess additional functionalities other than the proposed detection tool as shown in Table I, such as automatic recognition of password hash formats and support for cracking them using a dictionary-based attack is also a vital feature. A text file can be piped to dump the database tables entirely from a range of entries or specific columns as per the web application database nature in SQL MAP. While doesn't have any detection mechanism inbuilt in it. Web application firewalls are costly compared to the multidimensional attack mechanism based detection tool (MAMDT).

Table I. Comparison of Multidimensional attack mechanism based detection tool (MAMDT) with SQLMAP and Web application firewall

Parameter	**MAMDT**	**SQLMAP**	**Imeperva/ Barracuda Web Application Firewall**
Tool Type	Detection Tool	Attack tool, it cannot detect the vulnerability before attacking.	Web application Firewalls are prevention tools.
Request Mechanism	HTTP Request	HTTP Request	HTTP Request
Metasploit Support	Yes	Advanced Metasploit Integration	-
Parameter	**MAMDT**	**SQLMAP**	**Imeperva/ Barracuda Web Application Firewall**
Support to Execute Arbitary commands	Yes	Yes	Inbuilt features available.
Password Cracking	No	Support to enumerate password hash formats.	-
File Downloading	No	Yes	Not Applicable since WAFs being prevention tools.
Cloud Support	No	No	Yes. Extensive Cloud Support
Injection Technique	OR clause	Boolean-based blind, time-based blind, error-based, UNION query-based, stacked queries and out-of-band	Not Applicable.
Vulnerability Scanner Integration	Inbuilt	No	Yes
Database Support	MySQL, Oracle, MS SQL server, MS Access	All databases.	Not Applicable.
Automatic Tool Updates	No	Yes	Yes

V. EXPERIMENTAL RESULTS

A. Scan Results

The scan results reveal the IP address and the relevant ports, services that are currently running on the web server. The basic thread displays the miscellanous packet information such as the reset flag, syn-ack flag statuses and the vital application and service details are revealed in the second thread of the port scanner. Once the port scans are analyszed then kali linux based penetration testing tool Metasploit can be used to test the web application for exploiting the vulnerability. Table II shows result which reveals the exploit and payload for a particular version of IIS web server that was gathered from the port scanner modules earlier. As the impact of vulnerabilities increases it gets easier to launch exploits against such applications with the penetration testing tools available such as KALI Linux based programs.

B. SQL Injection Detection and Attack

The database module that send the crafted queries using HTTP methods which yields HTTP responses that contain the error message shown by the respective database server that is connected to the web application being tested. After pasrsing the error message, the presence of the SQL injection vulnerability is determined. A Sample web application that had SQL injection vulnerability being successfully determined by the multidimenional attack mechanism based detection tool using crafted queries as shown in Table III. The database attack reveals the critical database contents which includes table data and schema as shown in Fig. 5.

Table II. Port Scanner module results for website from thread one

IP Address and Ports	**Miscellanous Packet Information**	**Appication Information**
6x.2x9.1x7.132;tcp;25; 6x.2x9.1x7.132;tcp;53; 6x.2x9.1x7.132;tcp;80 6x.2x9.1x7.132;tcp;110 6x.2x9.1x7.132;tcp;113	smtp;open;MailEnable smptd;;syn-ack;1.986--;10;cpe:/o: domain;open;ISC ;http;open; ;pop3;open;MailEnable POP3	microsoft:windows BIND;;syn-ack;9.4.1;10;cpe:/a:isc:bind:9.4.1 /o:microsoft:windows Microsoft IIS httpd;;syn-

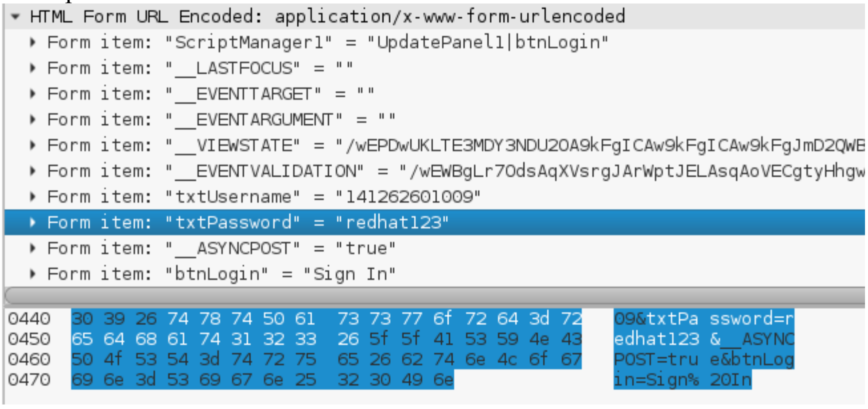
6x.2x9.1x7.132;tcp;143
6x.2x9.1x7.132;tcp;443
	Server;;syn-ack;;10;cpe:;ident;closed;;;reset;;3;
	;imap;closed;;;reset;;3;
	https;open;;;syn-ack;;3;
		ack;6.0;10;cpe:/o:microsoft:windows

Table III. Error Message in HTML

Response Message
<font face="Arial" size=2> <p>Microsoft JET Database Engine</font> <font face="Arial" size=2>error '80040e14'</font> <p> <font face="Arial" size=2>Syntax error in string in query expression 'item_id=10".</font> <p> <font face="Arial" size=2>/project.asp</font><font face="Arial" size=2>, line 6</font>

C. Sniffing Results

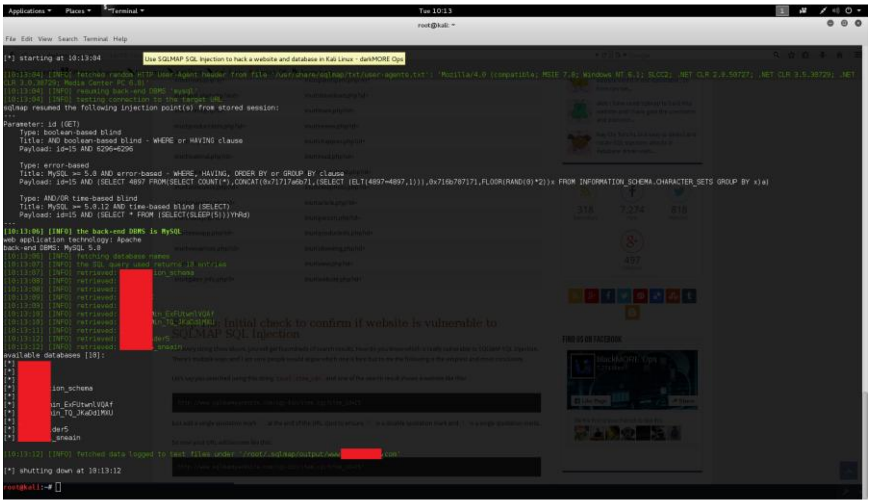
The port sniffer module has the packet threshold value set to 5 counts by the detection tool mechanism. After the port scanner module completed its function and when the logger has updated the scan result, the sniffer starts capturing the packets. The results contain IP addrersses of source client and destination server and plain HTTP requests and HTTP respsone that can be analysed manually. This multidimensional attack mechanism based detection tool has the abiity to automate the process of finding the clear text password vulnerabilities. The packet sniffer module in this test phase revealed clear text password vulnerabilies in 2 web application's login portals as shown in Fig. 4 where a query string "txtPassword" reveals the password "xxxxxxx".



```
▾ HTML Form URL Encoded: application/x-www-form-urlencoded
  ▸ Form item: "ScriptManager1" = "UpdatePanel1|btnLogin"
  ▸ Form item: "__LASTFOCUS" = ""
  ▸ Form item: "__EVENTTARGET" = ""
  ▸ Form item: "__EVENTARGUMENT" = ""
  ▸ Form item: "__VIEWSTATE" = "/wEPDwUKLTE3MDY3NDU2OA9kFgICAw9kFgICAw9kFgJmD2QWB
  ▸ Form item: "__EVENTVALIDATION" = "/wEWBgLr70dsAqXVsrgJArWptJELAsqAoVECgtyHhgw
  ▸ Form item: "txtUsername" = "141262601009"
  ▸ Form item: "txtPassword" = "redhat123"
  ▸ Form item: "__ASYNCPOST" = "true"
  ▸ Form item: "btnLogin" = "Sign In"

0440  30 39 26 74 78 74 50 61  73 73 77 6f 72 64 3d 72   09&txtPa ssword=r
0450  65 64 68 61 74 31 32 33  26 5f 5f 41 53 59 4e 43   edhat123 &__ASYNC
0460  50 4f 53 54 3d 74 72 75  65 26 62 74 6e 4c 6f 67   POST=tru e&btnLog
0470  69 6e 3d 53 69 67 6e 25  32 30 49 6e               in=Sign% 20In
```

Fig. 4 Finding of the Clear test password vulnerability



Applications
Places
Terminal
Tue 10:13
root@kali: ~
File Edit View Search Terminal Help
Use SQLMAP SQL Injection to hack a website and database in Kali Linux - darkMORE Ops

```
[*] starting at 10:13:04

[10:13:04] [INFO] fetched random HTTP User-Agent header from file '/usr/share/sqlmap/txt/user-agents.txt': 'Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET
CLR 3.0.30729; Media Center PC 6.0)'
[10:13:04] [INFO] resuming back-end DBMS 'mysql'
[10:13:04] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: id=15 AND 6296=6296

    Type: error-based
    Title: MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause
    Payload: id=15 AND (SELECT 4897 FROM(SELECT COUNT(*),CONCAT(0x71717a6b71,(SELECT (ELT(4897=4897,1))),0x716b787171,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.CHARACTER_SETS GROUP BY x)a)

    Type: AND/OR time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (SELECT)
    Payload: id=15 AND (SELECT * FROM (SELECT(SLEEP(5)))YHRd)
---
[10:13:06] [INFO] the back-end DBMS is MySQL
web application technology: Apache
back-end DBMS: MySQL 5.0
[10:13:06] [INFO] fetching database names
[10:13:07] [INFO] the SQL query used returns 10 entries
[10:13:07] [INFO] retrieved: [redacted]ion_schema
[10:13:08] [INFO] retrieved:
[10:13:08] [INFO] retrieved:
[10:13:09] [INFO] retrieved:
[10:13:09] [INFO] retrieved:
[10:13:10] [INFO] retrieved: in_ExFUtwnlVQAf
[10:13:10] [INFO] retrieved: in_TQ_JKaDdlMXU
[10:13:11] [INFO] retrieved:
[10:13:12] [INFO] retrieved: der5
[10:13:12] [INFO] retrieved: _sneain
available databases [10]:
[*] [redacted]
[*]
[*]
[*] ion_schema
[*]
[*] in_ExFUtwnlVQAf
[*] in_TQ_JKaDdlMXU
[*]
[*] der5
[*] _sneain

[10:13:12] [INFO] fetched data logged to text files under '/root/.sqlmap/output/www[redacted].com'

[*] shutting down at 10:13:12

root@kali:~# 
```

Initial check to confirm if website is vulnerable to SQLMAP SQL Injection
318
7,274
818
497
FIND US ON FACEBOOK

Fig. 5 Successful attack result revealing database contents

VI. CONCLUSIONS

Multidimensional attack mechanism is implemented in a detection tool (MAMDT) that can determine the web server version being used at the target and also which can be used in finding the related vulnerabilities in the common vulnerability database that can be compromised later. The tool can sniff the data packets sent between the target and the testing machine for HTTP packet information and this result can be used for automated analysis of clear text password vulnerabilities. The tool is built with the SQL Injection testing mechanism that can test the web application for SQL injection vulnerability by sending a crafted attack HTTP GET request to the target Web server. Then based on the vulnerability and its impact on the back-end database of the web application SQLMAP tool bundled within the Kali Linux Penetration Testing Framework can be used to compromise the database server. This detection tool was used to find clear text password vulnerabilities in 2 web applications portals and SQL injection vulnerabilities in 7 web applications. Further many new areas or dimensions can be added into this detection tool to find other popular security vulnerabilities that are not covered.

REFERENCES

[1] Jose Fonseca, Nuno Seixas, Marco Vieira, and Henrique Madeira, "Analysis of Field Data on Web Security Vulnerabilities", *IEEE Transactions on Dependable and Secure Computing,* Vol. 11, No. 2, pp. 80-100, April 2014.

[2] Sreeja Rajesh, "Protection from Application Layer DDoS Attacks for Popular Websites*", International Journal of Computer and Electrical Engineering*, Vol. 5, No. 6, pp 554-558, December 2013.

[3] Harshpal and Anant, "A Review on Zero Day Attack Safety Using Different Scenarios", *European Journal of Advances in Engineering and Technology,* Vol. 2, No. 1, pp. 30-34, September 2015.

[4] Gurpreet Juneja, "Ethical hacking: A Technique to enhance Information Security", *International Journal of Innovative Research in Science, Engineering and Technology,* Vol. 2, No. 12, pp. 7575-7580, December 2013.

[5] Aniruddha P Tekade, Pravin Gurjar, Pankaj R. Ingle, Dr.B.B.Meshram, "Ethical Hacking in Linux Environment", *International Journal of Engineering Research and Applications,* ISSN: 2248-9622,Vol. 3, No. 1, pp.1854-1860, February 2013.

[6] Bechtsoudis, "Aiming at Higher Network Security Through Extensive Penetration Tests", *IEEE Latin America Transactions,* Vol. 10, No. 3, pp. 1752-1756, April 2012.

[7] Nmap documentaion - www.insercure.org.

[8] Python documentaion – www.docs.python.org.