

Penetration Testing and Vulnerability Assessment to Improve the Web Application's Security

Mohammed Alazabe¹ *, Soad Algaib² *, FaraJ Alyaqobi³

Computer Science Department - Higher institute Of Science And Technology / Yafren - Libya

ARTICLE INFO

Email. mohamedalazabe1985m@gmail.com,

Received: 30-006-2025

Accepted: 14-07-2025

Published: 20-09-2025

Keywords.. Cyber attacks, web application vulnerabilities, vulnerability assessment, penetration testing, VAPT type

ABSTRACT

the rapid advancement of machinery, whether mobile or computer systems, has resulted in more advanced and efficient Windows, Web, and mobile applications, but it has also increased system complexity. The number of individuals who use web applications and the quantity of web applications on the market have both increased dramatically in recent years. Complex cyber-attacks can compromise the majority of web applications Web application testing has grown more important to defend web applications from cyber assaults and online hazards. In online application development, vulnerability assessment and penetration testing are critical security tasks. The practice of finding security flaws in online applications is known as vulnerability assessment [3].

Penetrationtesting is a method of determining if an application is secure by exploiting its weaknesses. It's possible to gain unauthorized access or engage in other hostile behavior. Therefore figuring out which flow is the most dangerous application is very important. It aids in the safeguarding of systems and data against illegal access and breaches. This paper reviews different vulnerability assessments and Penetration testing are [4].

i. INTRODUCTION

Electronic device utilization has skyrocketed in recent years. Almost every device is now connected to the Internet or a source of information sharing. New and advanced programs provide more features, but they also increase system vulnerabilities. Many well-known websites have been hacked in the recent past. Hackers are active currently, and they are constantly attempting to break into websites and steal data. Because of this, web application security testing is critical.

A growing number of websites contain flaws that could allow critical corporate data like credit card numbers and client lists to be stolen. The danger of an attacker disrupting or gaining unauthorized access to the system or any data held inside it is referred to as vulnerability. Hackers are monitoring web-based applications such as shopping carts, forms, and login pages, which can be accessed from anywhere in the globe at any time. Web application testing has become vital to secure web applications against cyber assaults and online hazards[3].

Vulnerability assessments and penetration testing work together to provide a comprehensive picture of a system's vulnerabilities. In web application security, vulnerability assessment and penetration testing are critical operations..

Web applications are scanned to see whether any known or unknown vulnerabilities are present.

Penetration testing is a sort of security testing that looks for dangers, flaws, and risks in a website that an attacker could take advantage of. It's also known as pen testing. Web applications are tested during penetration testing to find defense flaws that a cybercriminal could exploit[4]. In the majority of cases, a vulnerability assessment is carried out with the assistance of a penetration tester. A component for identifying susceptible regions in an application through testing. Financial security is ensured via penetration testing. areas such as banking and investment banking and stock exchanges where customers want their data to be secured.

Penetration testing is also necessary if the software system has already been hacked and the company wishes to recover.

To avert future hacks, assess whether any dangers are still present in the system. Fig. 1 shows the web application environment that needs to be tested and secured.

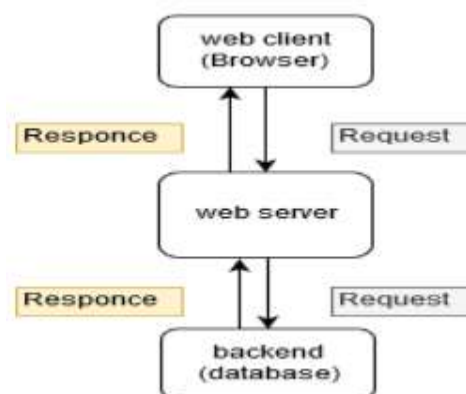


Fig 1. Web Application Environment[2]

ii. LITERATURE REVIEW

Ivan Krsul told that system security experts and professionals do not share and analyze computer and networks vulnerabilities. Computer vulnerabilities provide a wealth of useful information, which may be recognized and visualized, allowing for their prevention. Many of the engineering professions have studied and improved their work by learning from their previous mistakes in software development. Many sorts of study have been conducted in the topic of web apps, particularly in the field of financial online applications. Essentially, many researchers[4] concentrated on the cross-site phenomenon. Scripting, SQL injection, cross-site request forgery, and other types of attacks concentrating on a variety of other financial web-based threats application. A paper was delivered by Tiago Vieira and Carlos Serrao [5]. The focus of this research is on analyzing the outcomes of security audits. which has carried out research on a variety of financial web apps.

Checked the web application with automated tools

Security[5]. As a result, they have improved their work by developing advanced software that is more feature rich, but it has also resulted in that creates more vulnerabilities. This is where the VAPT process comes in by detecting vulnerabilities and fixing them in order to secure the system[5].

TABLE I. RELATED WORK FOR VAPT AND WEB SECURITY[5].

1.	Sugandh Shah,B. M. Mhetre	2014	In this paper focuses on the NetNirikshak tool for analyses security posture, and detects SQLi vulnerability.
2.	Ramos Somya, Danny Manongga.	2018	Focuses on, the Service-oriented Business Intelligence (SoBI) will be implemented to integrate academic and financial data at SWC University into Data Warehouse.
3.	Jai Narayan Goel, BM Mehtre.	2015	In this paper described vulnerability assessment techniques and VAPT tools, and how to use VAPT as a powerful cyber defence technology
4.	Sangeeta Nagpure,Sonal Kurkure	2017	Describes the analysis of web application vulnerability assessment and penetration testing methods
5.	Afsana Begum, Md. Hasan Sharif.	2016	Evaluated 153 vulnerable websites and study on exploitation of RFI and SQLi-based LFI vulnerability and techniques.
6.	Hessa M. Zaheer Al Shebli, Babak D. Beheshti.	2018	This paper focused on importance of penetration testing, factors and components while conducting penetration testing.

iii. VULNERABILITY ASSESSMENT AND PENETRATION TESTING

First, Web applications and websites are recognized to be vulnerable to security assaults, whether code-based or network-based. As a result of these flaws, attackers have the ability to take control of the system and its components.

This is what we refer to as a system that has been compromised. Not just heavily trafficked financial sites, but even a simple static html website requires a thorough vulnerability assessment and penetration test. It's critical to comprehend the gravity of vulnerable websites or web servers. An attacker could steal sensitive information from the server, interrupt website operations, or just deface website pages [1]. Therefore; they recognize using firewalls to safeguard a web application is insufficient. A detailed VAPT on a regular basis to ensure that the system is error-free is required. Vulnerabilities are flows in the framework, such as defects or misconfigurations, that render it defenseless to attacks. The evaluation of these framework vulnerabilities enables us to identify and deploy security patches in order to protect the framework from harm. The VAPT approach is divided into two distinct elements. The primary half is in charge of vulnerability analysis and discovery. The second portion deals with exploiting the distinct set of Vulnerabilities in order to assess their severity and impact on the Target architecture[1].

Vulnerability assessment is a separate approach, whereas penetration testing is a working methodology in which security specialists simulate an attack and test the target site's resilience to attacks.

iv. . TYPES OF VULNERABILITIES

The following are the types of the most common vulnerabilities.

Injection

Injection is the process of inserting data into a web server in order to obtain access to it. This can be done through SQL injections, No SQL injections, OS injections, and LDAP injections, among other methods. A hacker or tester tries various SQL queries and commands to access to a web server or website [4].

Broken Authentication

The application functions for session management and authentication are not implemented properly in Generation. As a result, attackers have easy access to user passwords, keys, session tokens, and other information, that undermine security systems, which leads to the exploitation of other implementation defects, allowing attackers to temporarily or permanently assume the identities of other users[4].

Sensitive Data exposure

Many online applications and APfs that deal with sensitive data, such as banking, health-care, and e-commerce, fail to provide adequate security.

shield. As a result of this omission, attackers are able to steal or modify data that has been improperly protected, resulting in crimes such as credit card fraud, identity theft, and so on. Sensitive data may be put at risk without additional protection, such as encryption, either on stationary data or data in transit. As a result, when sensitive data is sent via browser, certain security precautions are required

XML External Entities (XXE)

Multiple out-of-date and poorly configured XML processors are still in use to estimate external entity references contained in XML documents. Internal port scanning, file URI handlers, internal file shares, remote code execution, and denial of service assaults are all examples of external entities that can leak internal files[4].

Faulty Access Control

Inadequate enforcement of restrictions imposed on authenticated users allows attackers to exploit these flaws and gain unauthorized access to various functionality and data, such as access to other users' accounts, screening of sensitive files, data transmission, access rights modification, and so on[4].

Security Misconfiguration

This is the most commonly encountered issue, which is caused by dubious, insufficient, or ad hoc default setups, open cloud storage, misconfigured HTTP headers, or verbose error messages exposing sensitive information. The need for secure configuration necessitates the patching or upgrading of operating systems, frameworks, libraries, and applications on a regular basis[2].

XSS (Cross-Site Scripting)

When an application embeds untrusted data in a new document with no validations, this issue is known as [4]. Another way that XSS might be problematic is by utilizing a browser API that can write HTML or JavaScript to update an existing page with user-supplied data. It also allows attackers to launch scripts on the victim's browser, allowing them to hijack user sessions and deface websites.

fraudulent websites, or redirecting the user to malicious websites.[4]

Insecure De-serialization

This frequently leads to the execution of remote code. Even if dc-serialization, issues do not lead to remote code execution, they can be exploited to carry out attacks such as replay assaults, injection attacks, and privilege escalation.[4]

Using Components with Vulnerabilities Identified

Libraries, frameworks, and other software modules have the same rights as the application. Such an attack can result in catastrophic data loss or server takeover if a susceptible component is exploited. Applications and APIs that use components with known vulnerabilities might weaken application defenses and open the door to a variety of assaults and consequences [4].

Inadequate Logging & Monitoring

Lack of understanding of how to strengthen a system by keeping logs and monitoring activities, as well as omitted or pointless integration with other systems. Attackers can use incident response to harass systems, maintain persistence, connect to more systems, and corrupt, dig up, or wipe out data. A huge number of breach studies provide us with an estimate of how long it will take to recover[4]. If a breach has been found for more than 200 days, it is likely due to external parties rather than internal processes or monitoring. If a breach has been found for more than 200 days, it is likely due to external parties rather than internal processes or monitoring.

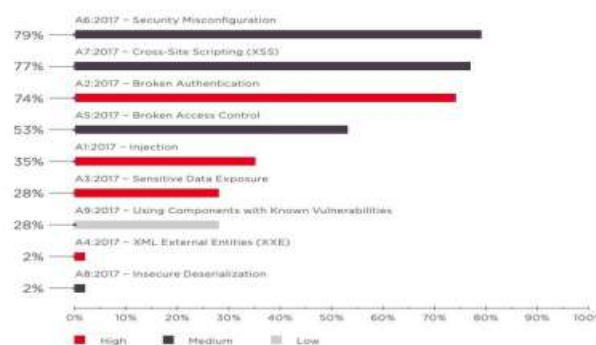


Fig 2. percentage of web application effected based on OWASP TOP 10 Vulnerabilities[2]

v. PENETRATION TESTING METHOD

Penetration testing method has the following procedures as shown in Fig. (3).

Planning

This phase identifies the rules and establishes the testing objectives. During the planning phase, a blueprint for successful penetration testing is developed before acquiring the information in this step. Goals and objectives, as well as scope, are part of the assessment process. The goals and objectives phase determines and examines the target web application [5].

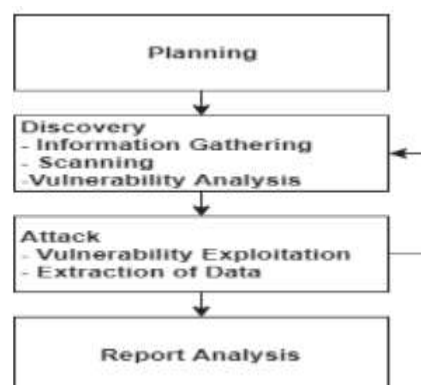


Fig. 3. Penetration Testing Method[5]

.Discovery

The first step is to acquire information, the second step is to scan, and the third step is to perform vulnerability analysis[5].

Vulnerability exploitation

Basically, attack actions are carried out during this phase. Part of the attack is the attempt to execute vulnerability exploitation and data extraction; during the vulnerability exploitation phase, a malicious query is injected in the web application on the input part. Attack tactics differ depending on the information collected in error messages during the vulnerability detection phase[5].

Data Extraction

When an attack is successful during the extraction step, SQL is used. The pen tester can deliver malware through SQL injection attack and compromise the entire website network by extracting table data, column names, and all other sensitive information that is stored in the database such as passwords, access system files, and user information[5].

vi. OVERALL PERSPECTIVE ON VA-PT

What are the differences between vulnerability assessment and penetration testing[4]?

The process of locating and identifying weaknesses in the system is known as vulnerability assessment. These flows give attackers a way into the system. There are numerous sorts of vulnerabilities, however the following are the most common[4].

- 1- Vulnerability of boundary condition
- 2- Vulnerabilities of authentication
- 3- Vulnerabilities of weakness in configuration
- 4- Vulnerabilities of Exception handling

Penetration testing is followed by vulnerability assessment, which exploits the system in a methodical and permitted manner[4]

Vulnerabilities Emerge: Vulnerabilities emerge for a variety of causes.

1. Software and hardware that are poorly designed
2. System misconfiguration
3. System connected to a potentially dangerous network
4. Poor password combination

Methods of Vulnerability Assessment Testing:-

- I. Active Testing - In this phase, new test data is introduced.
- II. Passive Testing - This method comprises data collection without the use of test data
- III. Network Testing - This procedure entails determining the network's present conditions[4].

vii. VAPT AS A CYBER DEFENSIVE TECHNOLOGY

An attacker normally performs is just reconnaissance the victim's network and gathers knowledge about the victim's network, as he describes how VA might be considered in a secure way to carry on. Following the collection of information, the attacker conducts a vulnerability assessment on the victim's system in order to compile a vulnerability list[4]. The attacker intends to create an attack after acquiring the vulnerability list from the victim's system. With the available list, the attacker abuses the victim's system, jeopardizing its security and information. If the victim eliminates all vulnerabilities from his system, the attacker will not be able to exploit the victim's system again[4].

viii. TEST METHODOLOGY

To conduct a penetration test, you must first create a testing strategy. Figure 4 shows a plan for conducting the test. There are numerous existing approaches and tools; one that stands out among them is

The OWASP Testing Guide v4 is the most well-known[1].

The areas of this arrangement will be depicted in this section, along with instructions on how to use them.

The most well-known security flaws are put to the test. To

perform an effective security test that captures the majority, if not all, of the data Several security tools have been developed to address system vulnerabilities.

A web application has been used to test it. All of these resources are available .particularly created with the OWASP security risks in mind. Priority is given to completing specific duties and, as a result, a variety of options are considered. Putting these disparate instruments together provides us a broad picture. [1] or discover false positives, which must be double-checked using capturing and penetration testing tools for monitoring.



Fig.4. Test Methodology [1]

ix. LIFE CYCLE OF VAPT

Life Cycle of VAPT is shown in Fig.5. It has the following procedures.

Scope

To perform a vulnerability assessment, we must first read the defined scope, defined parameters, network range, number of devices, given databases, and applications. It is determined by the VAPT's business goal. This stage also determines the composition of a team, their work, and the discussion of time constraints between the security tester/researcher and the company that wants to test their security, a law agreement is prepared. Later assessment tools, timeframes, team members, and testing methodologies will all be considered. [4].

Reconnaissance

A system tester or security researcher collects all relevant data about a device or system. Information includes operating system versions, logs such as system logs and login logs, recent user activity, and many other traces left by the user. This information aids the security tester in devising attack methods against the system. It might be active, such as directly touching the target machine, or passive, such activities being carried out by others [4].

Vulnerability Detection

In this step, a tester or security researcher looks for flaws in the system or device. He or she works on the victim's network and operating system, gathering records such as system logs and recent activity [4].

Information Analysis and Planning

The tester or security researcher analyzes the data acquired in the previous stage. He or she analyzes the danger discovered throughout the scanning procedure in order to determine the root cause and effect of the risk that will occur once the victim has been exploited. Appropriate attack strategies have been devised and are ready to be implemented.[4].

Penetration (Exploiting)

A Penetration Test is an in-depth expert-driven activity aimed at finding various probable paths through which an attacker could break into the network. Penetration Testing focuses on external real-world risk, whereas vulnerability assessment focuses on internal organizational security.[4]

Privilege Escalation

This approach is used after a successful penetration of the system to identify and get access to higher privileges, such as root access and administrative access [4].

Result Analysis

At this level, the root cause is correctly identified, the road to success is identified, and recommendations are made to correct the risk or issue. It's the life cycle's second to last stage, and it ensures a thorough description of the fault or danger, as well as techniques to correct the bug and make the system/device more secure.[4].

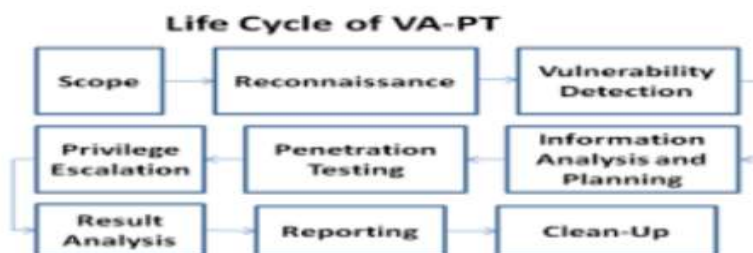


Figure5:Life-Cycle of VAPT[4]

x. VULNERABILITY ASSESSMENT AND PENETRATION TESTING OPEN SOURCE TOOLS

1. Zed Attack Proxy (ZAP)

OWASP's (Open Online Application Security Project) Zed Attack Proxy is a multi-platform, open source web application security testing tool. During the development and testing phases of a web app, ZAP is used to uncover a variety of security flaws. For advanced users, command-line access is available. ZAP is a Java application. Apart from being a scanner, ZAP may also be used to intercept a proxy and test a webpage manually. Application error disclosure, Cookie not Http Only flag, Missing anti CSRF tokens and security headers, Private IP disclosure, Session ID in URL rewrite, SQL injection, and XSS injection are among the faults discovered by ZAP[3].

2. Vega

Vega is a free open source tool for analyzing the security of online applications. You can use this tool to test the security of a web application. This utility is written in Java and has a graphical user interface. It's compatible with OS X, Linux, and Windows.

SQL injection, blind SQL injection, header injection, directory listing, shell injections, reflected cross-site scripting, stored cross-site scripting, file inclusion, and other web application vulnerabilities can all be found with this tool[3]. A robust JavaScript API can also be used to extend this tool.

3. Wapiti

Wapiti is a popular tool for testing the security of online applications. Wapiti is an open source project from Source Forge and developed . It is available for free. It conducts black box testing to identify security flaws in web applications. It is vital to be familiar with the various commands used by Wapiti since it is a command-line application. Wapiti injects payloads into scripts to see if they are vulnerable. Both the GET and POSTHTTP attack techniques are supported by Wapiti. Command execution detection, CRLF injection, Database injection, File inclusion, Shellshock or Bash bug, Server Side Request Forgery (SSRF), Cross Site Scripting, SEL injection and Xpath Injection, Backup files disclosure, weak, htaccess configuration, and many others are among the vulnerabilities discovered by Wapiti[3].

4. W3af

W3af Its purpose is to develop a framework that will assist you in securing your online applications by identifying and exploiting all web application flaws. As a result, it provides a more effective platform for web application penetration testing. It's written in Python to make it simple to use and extend. Over 200 types of online application vulnerabilities can be found using the tool, including blind SQL injection, buffer overflow, cross-site scripting, CSRF, insecure DAV setups, and many more. Fast HTTP queries, integration of web and proxy servers into the code, injecting payloads into various sorts of HTTP requests, and so on are some of its features. It comes with a command-line interface[3].

Linux, Apple Mac OS X, and Microsoft Windows are all supported. All versions are available for free download. It has a graphical as well as a console interface. It has an intuitive UI that makes it simple to use.

5. SQLMap

SQLMap is an open source penetration testing tool for detecting and exploiting SQL injection vulnerabilities in a website's database. It comes with a sophisticated detecting engine as well as a number of useful features. As a result, a penetration tester may easily check a website for SQL injection. MySQL, Oracle, Postgre SQL, Microsoft SQL Server, Microsoft Access, IBM DB2, SQLite, Firebird, Sybase, and SAP Max DB are among the database servers it supports. It's a robust SQL injection testing engine that can handle six different SQL injection techniques: time-based blind, Boolean -based blind, error-based, UNION query, stacked queries, and out-of-band[3].

6. Wfuzz

Wfuzz is a python-based open source tool for web application penetration testing. It does not have a graphical user interface and can only be run from the command line. It can be used to brute force GET and POST arguments for testing against SQL, XSS, LDAP, and other types of injections. Cookie fuzzing, multithreading, SOCK, Proxies, Authentication, parameter brute force, multiple proxy, and a variety of additional features are all supported[3].

7. Ratproxy

Rat proxy is an open source online application security audit tool that may be used to discover web application security flaws. It is compatible with Linux, FreeBSD, MacOS X, and Windows (Cygwin). This program was created to address the issues that users commonly encounter while using other proxy tools for security audits. Ratproxy is a little application that sits between your browser and the application you're testing. It keeps track of the application's outbound requests and responses, and it can build its own changed transactions to see how it reacts to typical threats. It can tell the difference between CSS style sheets and JavaScript code. It also enables the SSL man in the middle attack, allowing you to view data passing through SSL [3].

8. Nmap

Is an open-source mapping program. Network Mapper is abbreviated as Nmap. It's a network discovery and security auditing tool that's free and open source. This utility determines which hosts

are available on the network, as well as the hosts and services (application names and versions) that the devices provide[1]. It is mostly used to locate open ports and detect security threats in connected devices. Nmap can scan websites, servers, networks, and other systems for vulnerabilities. Nmap is a port scanning and discovery tool in general. It operates by delivering raw network packets to system ports, listening for responses, and determining if those ports are open, filtered, or closed. TCP, UDP, and ICMP are all supported by Nmap. It employs a separate packet format to verify for each of these protocols. Nmap is a tool that may be used on Windows, Mac OS X, and Linux.

On Linux, Nmap is preferred because it is faster and more efficient. Nmap provides a command-line executable as well as an advanced version[1]. For your convenience, there is a graphical user interface and a results viewer. If you're not sure where to begin with a security, Nmap can help.

9. Wireshark

Wireshark is an open source network analyzing program that captures network traffic as well as categorizes and analyzes packets. Network sniffers are another name for these gadgets. Ethereal was the previous name for Wireshark[1]. It's used to look at the specifics of network traffic at every level of the OSI model. It can track packets all the way down to the bit level. Wireshark captures packets using pcap; a pcap file can contain information such as the packet transmit time, the source and destination IP addresses, protocol used, and the protocol header data. These facts are incredibly helpful in determining the severity of a security breach and troubleshooting network security issues[1]. Wireshark allows us to see all of the activity on the interface, even packets that did not make it to their intended destination Tcp dump is comparable to Wireshark. Wireshark is better since it includes a user-friendly interface and built-in packet sorting and filtering capabilities. Wireshark can analyze data from the wire over a network connection or from previously captured data packets, it supports live data reading and analysis, it uses display filters to organize the data displayed, it can examine new or additional protocols by creating new plugins, and it can also capture raw USB traffic[1].

10. Nessus

Tenable Network Security is a company that specializes in network security. Nessus is a well-known and very effective vulnerability scanner. Nessus is one of those tools that is easy to use and has a nice interface. Despite the fact that there are other similar tools on the market, Nessus stands out because of how simple it is to use and how quickly it works. It also provides a full report of the vulnerabilities, which is highly important for developers. Nessus is made up of two parts: the Nessus Daemon and the Nessus Client. The actual vulnerability scanning is done by the Nessus Daemon (Nessus), and the vulnerability tests are written in Nessus Attack Scripting Language (NASL), which is optimized for specialized network interaction, and the Nessus client is in charge of controlling the scans done by Nessus[1].

It generates the outcomes of the system's vulnerability. Nessus usually runs a port scan on the target to find any open ports After that, it exploits those ports and scans them for vulnerabilities. It scans with the OWASP top 10 risks in mind, and pays special attention to these and other vulnerabilities. Plugins are what Nessus uses to scan for vulnerabilities. Each plugin looks for a

certain type of vulnerability. For example, there are plugging's for XSS vulnerabilities, SQL Injection, DHCP server difficulties, and so on. Nessus contains roughly 70,000 of these plugins, which are updated weekly. Along with vulnerability scanning, Nessus performs other tasks. basic network scans, security audits, host discovery, compliance checks, and a few additional features are available[1]

11. Air crack

This is a wireless security scanner that hackers employ to take advantage of wireless networks. Everything in today's world works on wireless, whether at home or in the outside world. Although WPA-2 security for 802. 11 WLAN standards is thought to be solved, misconfiguration and the usage of basic passwords leave such networks vulnerable to assaults. Air crack is a software package that includes sniffer, packet maker, and packet decoder functions. A targeted wireless network is caught in this scenario to capture all traffic and more information about encryption[4].

12. Metasploit

Metasploit is a pivotal open source framework that uses anti-forensics to execute comprehensive scans against a group of IP addresses, just like other frameworks. Programmers can create many exploits to make the most of a specific vulnerability, then analyze them with Metasploit to see if they are identified. When a virus exploits a previously unknown weakness, the procedure can be overturned. It can be used to put the patch through its paces[4]

xi. ADVANTAGES AND DISADVANTAGES OF VAPT

A- ADVANTAGES:

Benefits: When it comes to security, VAPT offers numerous benefits, including the following[4]:-

- 1- Describes the hazards that an application poses
- 2- Finding a programming issue in the system
- 3- A commercial company's or any organization's reputation and financial losses are greatly decreased.
- 4- There are less internal and external attacks.
- 5- Protects the system from harmful attacks[4]
- 6- Discover and discover systemic weaknesses that cause harm.
- 7- Identifying system flaws and gaps
- 8- Ensuring that sensitive information about users is kept private.[4]

B- DISADVANTAGES:

Despite its many advantages, not all aspects of penetration testing are beneficial. The following are the key causes for this [4]:

1. The issue of trust - The people who are generally paid to carry out these attacks are still hackers [4].

2- Cost and Management - The cost of recruiting these people Specialists need a substantial sum of money, and administration is not simple [4].

3-Time constraint - Testing is a lengthy procedure that takes a significant amount of time to complete.

4-Scope limitation - Due to resource and money constraints, organizations are unable to test everything. As a result, a lot of things are still unknown.

5-Method limitation- While there are numerous advanced methods for testing a system for finding vulnerabilities, it is also dependent on the system itself system, whether it will treat it in a specific way or not [4]

6-Skill Set limitation – Many experts still have limited knowledge of other types of technology, and they are solely focused on certain methods to fix it, which can be time consuming [4]

xii. CONCLUSION

Security testing methods are reviewed to ensure that data within an information system is secure and is not accessed by unauthorized people. Successful security testing protects web apps from serious malware and other dangerous threats that could cause them to crash or behave strangely.

In the early stages of development, security testing aids in identifying numerous flaws and vulnerabilities in a web application[3]. Organizations all across the world utilize security testing to guarantee that their web apps and information systems are safe[4]. The practice of safeguarding online applications from attackers by evaluating them for gaps and security vulnerabilities is known as vulnerability assessment and penetration testing. VAPT security testing tools are quite helpful in ensuring that online applications are secure. Breach of security This paper introduces a number of VAPT open source security testing tools, each with its own set of features and number of vulnerabilities discovered[3]

References:

- [1] Gitanjali Simran T and Sasikala D," , "Vulnerability Assessment of Web Applications using Penetration Testing," proceedings of IJRTE ,Volume-8 Issue-4, November 2019.
- [2] Matthi Naveen and Pragnyaban Mishra, "Addressing Web Application Security Issues and Vulnerabilities Assessment Pen Testing," proceedings of IJRTE ,Volume-8 Issue-6, March 2020.
- [3] Prof. Vaishali Malekar and Prof. Shilpa Ghode,"A Review on Vulnerability Assessment and Penetration Testing Open Source Tools for Web Application Security," proceedings of IJRTE ,Volume-2 Issue-3, June 2020.

[4]YuganshKhera,DeepanshKumar,Sujay and NidhiGarg,
"AnalysisandImpactofVulnerabilityAssessmentandPenetrationTesting," proceedings of International
Conference on Machine Learning, Big Data ,Cloud and ParallelComputing.2019.

[5] Arvi nd Goutam and Vijay Tiwari, "Vulnerability Assessment and Penetration Testing to Enhance
the Security of Web Application," proceedings of ISCON ,NOV 2019.