LAYERED ARCHITECTURE FOR ASSESSMENT OF SECURITY VULNERABILITIES
A Cornerstone of Effective Security Planning

[1]Chanchala Joshi, [2]Umesh Kumar Singh
[1,2]Institute of Computer Science,
Vikram University, Ujjain, Madhya Pradesh, India

Abstract: With the increasing growth of Internet it is extremely difficult to prevent unauthorized users from compromising the confidentiality, the integrity or the availability (CIA) of sensitive information. The development of comprehensive safety and security plans commonly overlooked the critical foundation step of vulnerabilities assessment. Knowing what vulnerabilities exist and could therefore be exploited allows organizations and businesses to pool that information with their knowledge of potential risks and threats to their operations and build their plans accordingly. Organizations need to have a clear plan in place to help better mitigate the vulnerabilities lies in the network or information system.

This paper presents a layered architecture for identification and assessment of security vulnerabilities. The developed architecture evaluates the organization's current policies and common practices and helps in identification and assessment of vulnerabilities by enlisting the aid of trained security professionals. Making the investment in a methodical assessment process will ensure the next steps in developing a safety and security plan are most effective and no more costly than necessary.

Index Terms **- network security; vulnerability analysis; vulnerability scanner; security threats**

I. INTRODUCTION

Use of computers is increasing day by day, which leads to increasing System's complexity. Most of the systems now are connected to the Internet. New and sophisticated software are coming in the market. All these activities are tremendously increasing vulnerabilities in systems. The vulnerability is a weakness or flaws in software applications or computer networks, which can be implementation bugs or design or implementation flaws that allow an attacker to cause harm to the user of the application and get extra privilege [1]. Vulnerabilities are the potential risk for the system. The attacker uses these vulnerabilities to exploit the system and get unauthorized access and information.

Vulnerabilities are a significant flaw in system security and Information assurance. Attackers use these threats and vulnerabilities to exploit the network system or victim's machine. It is better for security person or network administrator to identify the vulnerabilities present in the system in advance by before an attacker does. In organizations', the need for vulnerabilities detection and assessment is usually underestimated till now. It is just considered as a formality activity and use by decidedly fewer people. By performing regular and consistence vulnerability assessment, we can reduce a substantial amount of risk to be attacked and have more secure systems. This chapter illustrates the performance of vulnerability assessments to identify weaknesses in systems and applications by the automated tool, called vulnerability scanners. Vulnerability scanners help to automate the process of identifying such security concerns in an organization.

Vulnerability discovery and assessment is a step by step process. Vulnerability discovery is the process of scanning the system or software or a network to find out the weakness and loophole in that. These loopholes can provide a backdoor to the attacker to attack the victim. A system may have access control vulnerability, Boundary condition vulnerability, Input validation vulnerability, Authentication Vulnerabilities, Configuration Weakness Vulnerabilities and Exception Handling Vulnerabilities, etc.

Vulnerability assessment is the next step after vulnerability discovery. Vulnerability assessment primarily performs penetration testing to exploit the identified vulnerabilities in an authorized manner to assess True Positive and False Positive vulnerabilities. In penetration testing, the tester (also called whitehat hacker) has authority to do penetration testing, and intently exploits the system and finds possible exploits. For proper security assessment, it is essential to determine system's capability to resist attacks. Vulnerability assessment involves probing the system to detect the presence of well-known vulnerabilities because most of the attacks typically exploit well-known vulnerabilities that have not been patched. For the secure system, it is essential to find and assess these vulnerabilities.

II. FINDING VULNERABILITIES IN INFORMATIONSYSTEMS

Vulnerability scanning is an examination of the possible weak points of exploit on a network or computer system to recognize security breaches. A vulnerability scan process discovers and classifies system weaknesses in networks, computers, and communications equipment and predicts the efficiency and effectiveness of countermeasures. A vulnerability scanner runs and operates from the end point of the security persons investigating the attack surface. The software matches details about the attack surface of a target to a database contains the information about known security holes in application services and ports, anomalies or maliciousness in packet construction and potential paths to exploitable programs or scripts. The scanner software attempts to exploit each of the discovered vulnerability.

A vulnerability scanner assesses security vulnerabilities in networks or host systems and produces a set of scan results. However, both administrator and attackers mostly use the same tool for fixing or exploiting a system. Therefore, an administrator needs to conduct vulnerability scanning and fix problems by applying patches, before an attacker can perform the same scan and to exploit the vulnerability. One of the most crucial advantages of a vulnerability assessment is that it will always keep organization's security persons one step ahead of the attackers. Vulnerability scanning is the most robust proactive security mechanism of securing an organization' networks. Since Vulnerability assessment process already recognizes all the security holes an attacker can exploit, a network administrator just needed to patch the identified vulnerabilities. All he needs to do is to keep running vulnerability assessment process from time to time on a regular basis, just to keep track of new vulnerabilities.

Vulnerability assessment can be divided into two major categories [2]:

- Network-based scan
- Host-based scan

The network-based vulnerability assessment tools empower the network administrator to discover, identify and eliminate network-based security vulnerabilities in organization's computing environment. On the other hand, host-based scanning tools assist the network administrator to secure his organization's internal computing environment by providing a supportive security layer. Giving limited access to the hosts, it prevents him from accessing confidential or sensitive data and information of the organization. In summary, the network-based analysis is to examine and keep an eye on the whole network, while the host-based investigation is to keep an eye on particular host or group of hosts [3].

2.1 Network-based scan

The network-based scan can distinguish vulnerable hosts on a network. A network administrator or security persons while conducting the vulnerability assessment should initially perform the network-scan process. A network-based test produces the instant outcomes of highly severe vulnerabilities that needed an immediate fix. A misconfigured firewall or vulnerable web server, which is considered very critical vulnerabilities, can be detected easily by running a network vulnerability test.

Some essential tools for network scan are Nessus, Nexpose, Nmap, and OpenVAS.

2.2 Host-based scan

The host-based vulnerability assessment works on a client-server model where client performs the scan and sends the report back to the server. In this scan, client files, that administrator want to check should installed on every machine. The main benefit of host-based vulnerability assessment is to keep an eye on a suspect [4].

Suppose, if we want to monitor the activities of an employee in the workplace, because he might inject vulnerabilities and malware in the organization' network; hence we will perform host-based vulnerability scanning. The host vulnerability assessment allows a network administrator to eliminate the inside security risks from the organization' computing environment. In the host-based scan, vulnerability assessment tools run the scan tests from the viewpoint of a user who is enrolled, a local account on his computer system. Once a user connects to the local network of the organization, even from guest account, he can exploit the vulnerability or security holes in the local servers and could end up taking control of organization's local systems.

The host-based vulnerability assessment permits a network administrator to evaluate security risks caused by inexperienced users (who do not follow the security protocols), malicious users, and also users in between them.

III. VULNERABILITY ASSESSMENT PROCESS

The vulnerability assessment process should be well-organized. It is also essential that vulnerability assessment must follow a systematic approach using a standard process to ensure efficiency and effectiveness. Fig. 1 represents the steps involved in vulnerability assessment process.

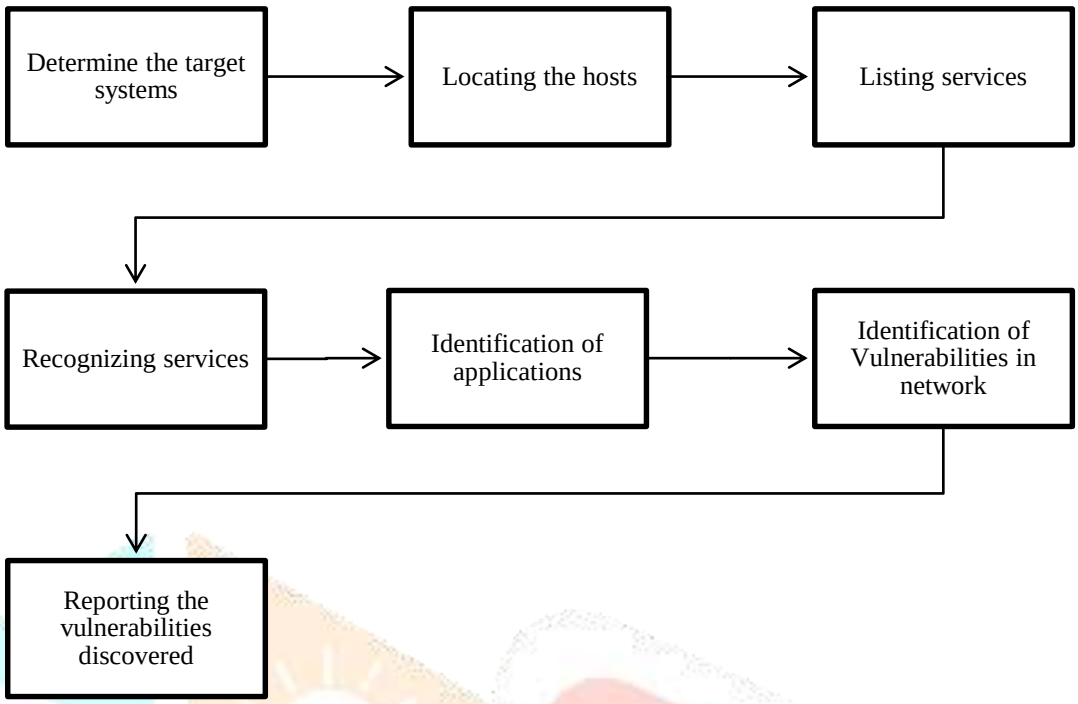

Fig. 1 Vulnerability assessment process

3.1 Discovery of the Target Systems

The first and initial step of vulnerability assessment process is to discover the IP (internet protocol) that a network administrator wants to target in the network. To determine the intended IP administrator examines the span of IP addresses that are mapped to his online system. Further, enquirers are sent all IP addresses specified by the computer or user to extract the response. After receiving the response in return of inquiries, the system will designate that IP address as a valid host.

3.2 Locating the hosts

After executing the initial state of vulnerability assessment process, the network administrator will perform numerous host scanning routines such as connect scan, TCP scan, synchronize scan, etc. to detect the live hosts and their information regarding the services operating on them. It results in the information about the applications running on the host, the knowledge of open ports and the data about the operating system running on the host. These fingerprinting techniques range from SNMP (Simple Network Management Protocol) to complex TCP connect based operating system description.

3.3 Listing services

After successful completion of the first and the second steps of vulnerability assessment, the port scanning is done by the network administrator. Port scanning discovers the vulnerable ports on the host machine and the vulnerable services associated with that particular port number. Therefore, it is the most crucial step of vulnerability assessment process.

Nowadays, several techniques are available for examining the ports and protocols on which a target machine is listening. They all have different advantages and difficulties. Some of the prominent methods are:

TCP connect scanning (-sT): connect scan is the most basic type of TCP scan, that establishes a connection to every open port on the machine. TCP connect doesn't require any special privileges, and moreover, this is the fastest scanning method. The primary disadvantage concerning TCP connect() scan is readily detectable and filterable.

TCP SYN scanning (-sS): TCP synchronize scan is the default and most commonly used scanning method. It results in speedy performance. In a fast network which is not restricted by firewalls, TCP SYN scans thousands of ports per second. Because the SYN scan never completes TCP connections or never establishes full TCP connection, hence it is stealthy and modest.

In TCP SYN attacker does not open a full TCP connection. Attacker sends a synchronized SYN packet, to establish a connection and wait for a response. An acknowledgment SYN|ACK shows that the port is listing, while a reset RST indicates a non-listing port. The main advantage of TCP SYN is that very few sites keep the log of it. However, an attacker requires necessary privileges to develop SYN packets. Fig. 2 depicts the Wireshark trace of TCP SYN executed by Nmap:

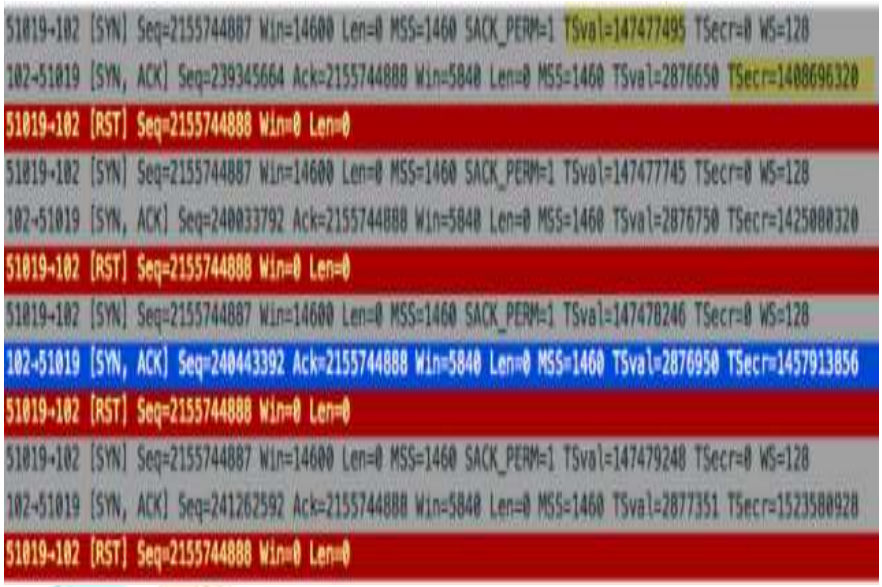

Fig. 2 Wireshark trace of TCP SYN executed by Nmap

UDP scanning (-sU): On Internet, numerous of common services mostly run over TCP; however, UDP services are extensively deployed. DNS (port 53), DHCP (port 67/68), and SNMP (port 161/162) are three most common UDP services. Most of the security persons and WhiteHat hackers avoid these ports because UDP scanning is more difficult than TCP and performance is also slow. However, these UDP services are quite common and even have a high probability of exploit. Furthermore, attackers unquestionably do not ignore the UDP protocol.

TCP ACK scanning (-sA): TCP acknowledgment scan, mostly performed by an authorized WhiteHat attacker, who has an authenticate access to the network. This scan is distinctive than the others as it never determines open (or even open|filtered) ports. It usually maps out the firewall rulesets for deciding whether they are stateful or not and which ports are filtered.

3.4 Recognizing the services on open ports

The next step of the vulnerability assessment is identifying the services on every open port, (which is detected in the previous step) of a network. In the test initially, we send the similar requests frequently, and assessment tool will evaluate the responses against the set of signatures. Once a match is made between signatures of known application, the data will be saved for later use, and the tool will start running tests on other services.

3.5 Identification of vulnerable applications

One of the most critical and necessary steps in all vulnerability assessment process is the identification of vulnerable application. Once a network administrator moves from identifiable services stage he needs to pinpoint vendor and every type of service which was discovered in the previous fourth step. This is the most critical phase, as we have to take care of vulnerability test on an application should not crash or affect other applications. If any application cause crashing of another application, then the resultset of performed test cannot be rated as efficient.

A network administrator needs to distinguish false positive or false negative vulnerabilities. The common cause of a false positive is flawed application description on which the test was conducted.

3.6 Identification of vulnerabilities present in network

After the system is equipped to perform the test, as all of the open ports of hosts on a network are mapped, all services running on these ports are also mapped to a particular application; the vulnerability assessment test is going to start. This step of the vulnerability discovery process, active configuration probes are conducted, and in the end, a set of custom attacks on the network define either a stated vulnerability exists in a system or not.

3.7 Reporting the vulnerabilities discovered

Reporting is the final stage of vulnerability assessment process, where the discovered vulnerabilities in the system or network are documented. The report will indicate which vulnerabilities are highly severe and which are not. The report will also provide solutions to a network administrator that how the detected vulnerabilities will be fixed.

One thing which most of the assessment tools have in common is the ability to show tendency report of how a tested network progressed over time. The network administrator can also choose the report summary to present it to his organization's management.

IV. VULNERABILITY SCANNING TOOLS

Vulnerability scanners enable early detection and handling of known security threats and vulnerabilities. By applying continuous security assessment process using vulnerability scanners, it is easy and secure to recognize security vulnerabilities, which may exist in the network, from both the internal and external aspect.

Vulnerability scanners are mainly having four components: scan engine, scan database, report module and user interface (shown in Fig.3).

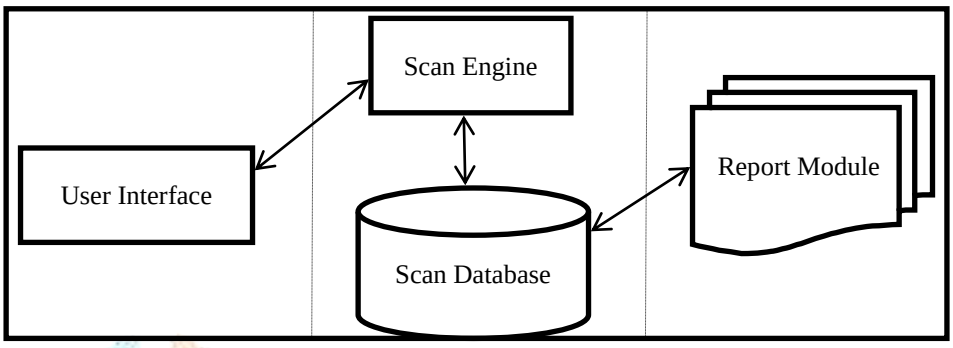

Fig. 3 Components of vulnerability scanning tool

i. **The Scan Engine** runs security constraints according to its installed plug-ins for knowing computer system or network information and vulnerabilities. It scans several hosts at a time and correlates the results against known vulnerabilities.

ii. **The Scan Database** keeps vulnerability data, scan results information, and the data used by a scanner. Depending on the corresponding vendor, the number of possible plug-ins and the updating recurrence of plug-ins may vary.

Each plug-in might include not only the test cases but also a vulnerability description, a Common Vulnerabilities Exposures (CVE) [5] identifier. Even it also contains the fixing instructions for a detected vulnerability. Scanners set with an "auto-update" feature are able to automatically download and install the latest set of plug-ins to the scan database.

iii. **The Report Module** accommodates several types of reports against the scan results, such as summary reports for security managers, detailed technical reports with suggested remedies for system administrators, and high-level graph and trend reports for executives.

iv. **The User Interface** enables the administrator to operate the scanner. It may be either a command line interface like in Nmap or a Graphical User Interface (GUI) like Nexpose or OpenVAS.

Some of the prominent scanning tools used in our work are:

4.1 NMAP

Nmap (Network Mapper) initially written by Gordon Lyon is an openly available security scanner, port scanner and network exploration tool [6].

Nmap is used to discover hosts in a computer network and subsequent services running on the hosts, consequently building a “map” of the network. To accomplish its goal, Nmap sends specially crafted packets to the target hosts in the network and then analyzes the responses. Nmap initially started as a Linux-only utility, but porting to Windows, Solaris, OS-X, HP-UX, AmigaOS, and IRIX have followed. Linux is the most popular platform, followed closely by Windows.

4.2 Nessus

Nessus vulnerability scanner is designed and developed by Tenable Network Security. According to surveys done in 2015 by sectools.org [7], Nessus is the world's most popular vulnerability scanner, taking first place in the security tools survey. Nessus is an ideal assessment tool for the large networks. Its extensibility allows its users to leverage their expertise in generating vulnerability checks.

4.3 Acunetix

Acunetix Web Vulnerability Scanner [8] is an automated web application security testing tool that audits web applications by checking for vulnerabilities like SQL Injections, Cross-Site Scripting, and other exploitable hacking vulnerabilities. Acunetix WVS can perform scanning on any website or web application that is and uses the HTTP/HTTPS protocol and accessible via a web browser.

4.4 Netsparker

Netsparker does not require an expectation to use the tool, it provides very easy and convenient user interface, and it does an excellent job detecting the most critical vulnerabilities [9]. It has excellent and useful reporting features that are easy to read and intuitively

designed. Moreover, it can confirm detected vulnerabilities. Security persons do not need to validate the vulnerabilities that have been confirmed by Netsparker, so it is an excellent time saver tool.

4.5 OpenVAS

OpenVAS (Open Vulnerability Assessment System) is a tool used for vulnerability scanning and vulnerability management [10]. OpenVAS is a "remote scanner" because it does not have to be introduced on a target for it to test. Instead, it could be installed and configured on only device and test many hosts. OpenVAS is client-server architecture over SSL

V. COMPARISON OF SCANNING TOOLS

We have evaluated the performance of four prominent network vulnerability scanners (OpenVAS, Nessus, Nexpose, and NMAP) and three widely used web vulnerability scanners (Acunetix, Netsparker and Burp Suite) [11].

Network vulnerability scanners OpenVAS, Nexpose, and NMAP were inspected on exploitable and misconfigured services on the Metasploitable framework. The scanning performed in the following manner:

i. OpenVAS was tested with the full and fast scan profile.

ii. Nessus was propelled utilizing the external network scan profile.

iii. The Nexpose scanner [12] was executed with the full review profile.

iv. Nmap was examined by TCP connect scan with a port scan.

These are the amounts of vulnerabilities uncovered efficiently and appraised by every vulnerability scanner, from the example set by exploitable services.

Nmap and Nessus discovered 11 vulnerability holes out of 15 while OpenVAS identified 7 and Nexpose detected only six vulnerability holes.

Web vulnerability scanners, Acunetix, Netsparker and Burp Suite [13] were tested with the vulnerable web application of Acunetix test.php. All web scanners follow the common strategy: firstly they crawl the victim website, then they create and insert payloads, and finally, they analyze the response [14].

The performance evaluation of these three web scanner is summarized in Table 1.

Table 1 Web vulnerability scanners results

SNo	Vulnerability Category	Vulnerability Type	Acunetix	Netsparker	Burp Suite
1	SQL Injection		15	4	7
2	Broken Authentication and Session Management	Password Guessing	5	0	2
		Brute Force	1	1	0
3	Cross Site Scripting	Non-Persistent XSS	9	9	2
		Persistent XSS	1	3	1
		DOM XSS	3	1	0
4	Security Misconfiguration	Password sent via GET Method	5	5	5
		Web Server DDoS	2	0	2
		Sensitive Data display	0	4	2
Total			40	27	18

Table 1 reports the vulnerabilities that were detected by web application scanners. As seen from the Table 1 all the tool tools missed some weaknesses.

The analysis of why the scanners missed specific vulnerabilities is as follows:

1. **SQL Injection:** Acunetix Scanner can discover all SQL Injection vulnerabilities. However, Netsparker and Burp Suite scanners are failed to find some SQL Injection vulnerabilities, which are not executed immediately.
2. **Broken Authentication and Session Management:** Both Netsparker and Burp Suite scanners were not able to find the vulnerability.

3. **Cross-Site Scripting:** Acunetix and Netsparker Scanners discovered all NonPersistent XSS vulnerabilities. Burp Suite scanner result is inferior. All scanners missed most of the Persistent XSS and DOM XSS vulnerabilities.
4. **Security Misconfiguration:** All the scanners can find the vulnerability Password get via GET Method. Acunetix Scanner missed Sensitive Data Display vulnerability.

The comparison of the three chosen scanners shown by the following graph:

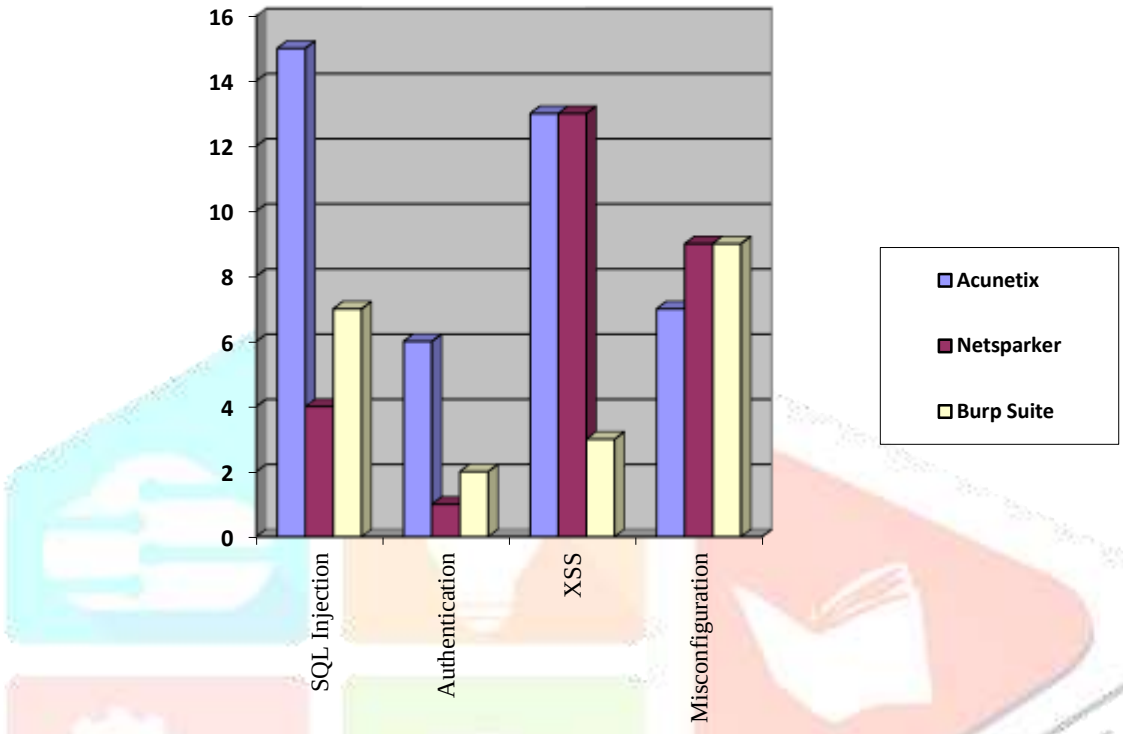

Fig.4 Comparison of Acunetix, Netsparker and Burp Suite

The evaluation of result shows that both Acunetix and Netsparker scanners able to discover cross-site scripting XSS but Burp Suit results were very poor. For SQL Injection Acunetix detect all the vulnerabilities. Scan results of Acunetix WAVS for Broken Authentication and Session Management vulnerabilities are better than other two scanners. However, Security Misconfiguration vulnerabilities are not discovered adequately by Acunetix. In such case, the result of Netsparker and Burp Suit Scanners are better.
The results show that the crawling has been significantly improved, although there are still limitations that affect the detection rate of such vulnerabilities as SQLI and XSS.

VI. PENETRATION TESTING

Penetration Testing is a proactive effort to regulate the security of an IT system by safely exploiting its vulnerabilities either in an authorized manner by WhiteHat hacker or in an illegal way by RedHat or BlackHat hackers. Mostly penetration testing evaluates application flaws, improper configurations, and risky end-user behavior [15]. Penetration testing is the additional step where we perform the attacks as an attacker does. The penetration testing process has the vulnerability assessment in it, where the specialist first performs the vulnerability assessment to find the issues, and then he/she act as an attacker to exploit the found vulnerabilities.

Metasploit is one of the most commonly used penetration testing tool [16]. The objective of penetration testing is to provide the proof-of-concept and to illustrate the attacking vectors, or merely the aim is to show how risky and vulnerable the network is.

After penetration testing, the scan results obtained during vulnerability scanning process can be categorized as:

6.1 True Positive (TP)

True Positive (TP) means the number of correctly identified malicious codes during vulnerability scan.

6.2 True Negative (TN)

True Negative (TN) refers to the number of correctly identified benign codes, means the non-malicious code that is classified as a genuine code.

6.3 False Positive (FP)

False Positive (FP) shows that the alarm is generated when there is no actual attack. FP is the number of incorrectly identified trusted code as malicious code.

6.4 False Negative (FN)

False Negative (FN) is when the system fails to detect the malware activity due to it being similar to normal activity, or no signature is available in the database.

It is noteworthy that FN and FP refer to misjudgment. Table 2 summarizes the possible cases of the classification scheme.

Table 2: Judgment cases of vulnerability assessment

Classification / **In reality**	**Malicious Code**	**Non-Malicious Code**
Malicious Code	True Positive (TP)	False Positive (FP)
Non-Malicious Code	True Negative (TN)	False Negative (FN)

The rates of TP, FP, FN and TN will be computed by using four standard metrics to evaluate the performance of our technique:

True Positive Rate (TPR): This is the percentage of correctly identified malicious codes. TPR is measured as the ratio between the number of events that have accurately classified as positive and the total number of events that can be classified as positive, which is given by:

$$TPR = \frac{|TP|}{|TP| + |FN|}$$

False Positive Rate (FPR): This is the percentage of wrongly identified benign codes, measured as the ratio between the numbers of events that were considered positive on the number of events that should have been negative, which is given by:

$$FPR = \frac{|FP|}{|FP| + |TN|}$$

False Negative Rate (FNR): This is the rate of incorrectly rejected malicious code.

$$FNR = \frac{|FN|}{|TP| + |FN|}$$

True Negative Rate (TNR): This is the percentage of correctly identified benign codes.

$$TNR = \frac{|TN|}{|FP| + |TN|}$$

6.5 Accuracy and Precision rate

Accuracy, Precision, and Recall will be used to evaluate the filtering accuracy of scanners.

- Accuracy is used to evaluate the accuracy of the classification results, namely, the proportion of the malicious codes that are accurately classified into their categories. It is calculated through the following formula:

$$Accuracy = \frac{|TP| + |TN|}{|TP| + |FP| + |FN| + |TN|}$$

- Precision is the positive detection value which measures the effectiveness of vulnerability scanning tool. Precision is used to evaluate the proportion of malicious codes among all the network activities that are judged to be malicious. The following formula calculates the value of precision:

$$Precision = \frac{|TP|}{|TP| + |FP|}$$

- Recall is used to evaluate the proportion of the malicious code that is accurately classified as malicious.

$$Recall = \frac{|TP|}{|TP| + |FN|}$$

F-measure is the harmonic mean of precision and recall. It is adopted as one of the measuring indexes of the filtering mechanism, and calculated as:

$$Fmeasure = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Consequently, F-measure is a means of evaluation that combines precision and recall effectively.

The principal goal of our work is to identify internal and external threats by vulnerability assessment using network vulnerability scanner. The research at hand recognizes the importance of network security and specifically that of current vulnerability scanners. It is aimed principally at contributing to enhancing vulnerability assessment process by combining tools and reducing administrative burden.

VII. CONCLUSION

This paper presents a layered architecture for identification and assessment of security vulnerabilities. Various types of scanning (such as network scan, port scan, service scan, web scan) and diverse kind of scanning tools used for network vulnerability scanning are discussed. In today's market, a large number of vulnerabilities scanners are available; this paper evaluates the effectiveness of some of the prominent tools against a common type of vulnerabilities categories.

VIII. ACKNOWLEDGMENT

Authors are thankful to Vikram University, Ujjain for providing support and financial grant for the research work.

REFERENCES

[1] B. Wu, A. J. A. Wang, "EVMAT: an OVAL and NVD based enterprise vulnerability modeling and assessment tool", Proceedings of the 49th Annual Southeast Regional Conference, pp. 115-120, 2011.

[2] W. Koch, and A. Bestavros, "PROVIDE: Hiding from Automated Network Scans with Proofs of Identity", 2016 Fourth IEEE Workshop on Hot Topics in Web Systems and Technologies (HotWeb), 2016, pp. 66-71.

[3] S. Sarasan, "Detection and Prevention of Web Application Security Attacks", International Journal of Advanced Electrical and Electronics Engineering, (IJAEEE), ISSN (Print) : 2278-8948, Volume-2, Issue-3, 2013, pp. 29- 34.

[4] A. A. Neto, and J. Duraes, "Assessing and Comparing Security of Web Servers", 2008. 14th IEEE Pacific International Symposium on Dependable Computing. IEEE Computer Society.

[5] CVE - Common Vulnerabilities and Exposures (CVE), Available: https://cve.mitre.org/

[6] R. Bowes, "Scanning Windows Deeper With the Nmap Scanning Engine", SANS Institute InfoSec Reading Room, June 2009.

[7] H. Mu, Y. Wen, S. Guo, Y. Zhang, and H. Wang, "Reliability design of a retracting actuator based on NESSUS", 10th International Conference on Reliability, Maintainability and Safety (ICRMS), 2014, pp. 698-701.

[8] Acunetix Web Vulnerability Scanner, 2008, http://www.acunetix.com/vulnerability-scanner/

[9] Netsparker Web Vulnerability Scanner, 2012, https://www.netsparker.com/web-vulnerability-scanner/

[10] Y. Wang and J. Yang, "Ethical Hacking and Network Defense: Choose Your Best Network Vulnerability Scanning Tool", 31st International Conference on Advanced Information Networking and Applications Workshops (WAINA), 2017, pp. 110-113.

[11] C. Joshi and U. K. Singh, "Security Testing and Assessment of Vulnerability Scanners in Quest of Current Information Security Landscape", International Journal of Computer Applications (IJCA 0975 – 8887), Volume 146(2), pp. 1-7, July 2016.

[12] W. Rosenberry, "Nexpose: Vulnerability Management and Penetration Testing System v.5.1 Security Target", Technical Report 545 Boylston Street, Suite 400 Boston, MA 02116, May 2012.

[13] Burp Suit Web Vulnerability Scanner, https://portswigger.net/burp/

[14] C. Joshi, U. K. Singh, "Performance Evaluation of Web Application Security Scanners for More Effective Defence", International Journal of Scientific and Research Publications (IJSRP), Volume 6, Issue 6, pp 660-667, June 2016.

[15] N. Antunes and M. Vieira, "Enhancing Penetration Testing with Attack Signatures and Interface Monitoring for the Detection of Injection Vulnerabilities in Web Services," Proc. IEEE Int'l Conf. Services Computing (SCC 11), IEEE CS, 2011, pp. 104-111.

[16] D. Kennedy, J. O'Gorman, D. Kearns, and M. Aharoni, "Metasploit: The Penetration Tester's Guide", San Francisco: William Pollock, 2011.