Website Security Tool

Vaibhavi Rane[1], Chaitrali Rane[2], Mayur Shelar[3], Vijaya Pinjarkar[4]

Information Technology department,
K. J. Somaiya Institute of Engineering & Information Technology,
Mumbai University, Mumbai, India
Vaibhavi.r@somaiya.edu
Information Technology department,
K. J. Somaiya Institute of Engineering & Information Technology,
Mumbai University, Mumbai, India.
Mayur.shelar@somaiya.edu
Information Technology department,
K. J. Somaiya Institute of Engineering & Information Technology,
Mumbai University, Mumbai, India.
Chaitrali.r@somaiya.edu
Professor,Dept. of Information Technology ,
K. J. Somaiya Institute of Engineering & Information Technology,
Mumbai University, Mumbai, India.
ykhirodkar@somaiya.edu

Abstract - *The project aims to protect the website of the user from the various vulnerabilities so the data can be protected from unauthorized access and also it will help the user to acquire the knowledge of the various attack to which there websites can be prone to for example through SQL injection the attacker can acquire the enter database of an organization .In such types of attacks the tool will help the organization to know the attack that there websites are vulnerable and implement measures to protect them .The websites cannot work properly if it has been hacked. This means that millions of people around the world who are trying to get access the website will be deprived of the ability to communicate with organization, when they need it the most .The paper proposes idea to allow the user to detect the various vulnerabilities also get the knowledge regarding the various attacks in one single tool. . The tool is designed to be used by people to get security experience and as such is ideal for developers and functional testers who are new to security testing. The Website Security Tool is a process intended to reveal flaws in the security mechanism of an organization so to protect data and maintain the functionality as intended .The user can detect SQL Injection, Logic Bomb, Nmap attacks and get information of the attacks like Time bomb, Cross-site scripting, Salami attack. This application is developed specifically for people who are new to security.*

***Key Words*: Web Application Security, Website Security Tool for Detection, Cross Site Scripting (XSS), SQL injection, Logic Bomb, Network Mapper (NMAP).**

1. INTRODUCTION

Web Applications are being extensively used by people at all levels, businessman, doctors, engineers, service man and common man and woman in their day-to-day activities. Today communication is next to impossible without web application. Security vulnerabilities results in stealing of confidential data of the web applications, data integrity is endangered or results in the unavailability of the data to the users. The task to secure web applications has now become a need: Acunetix survey 60% of found vulnerabilities affects web applications.

The most common way of securing web applications is to search and eliminate the vulnerabilities therein. Another way of securing web application includes proper development, having an intrusion detection system and/or protection systems and web application firewalls. The most efficient way is to use the manual code review to find security vulnerabilities in web applications. The manual code technique is difficult to implement as lot of time is consumed in finding errors. The technique requires expert's supervision and may also result in some overlooked errors. Therefore, security society uses two new automated approaches that are black-box and white-box testing for security vulnerabilities. The first approach the source code is not available so it does the web application analysis from the

user side. The idea is to submit various malicious patterns into web application forms and to analyses its output thereafter. Depending upon the errors that are detected the assumption about a particular vulnerability is made. The black-box testing is not accurate and all the errors are not detected hence is incomplete. The second approach the source code is available so it does the web application analysis from the server side. In this case dynamic or static analysis techniques can be applied.

1.1 SQL Injection

SQL Injection is a form of attack in which the malicious SQL queries are used by the attacker to get access to the user database. It is done by executing a SQL Query/Statement or syntax by injecting it in a user input field on the web application.

1.2 Logic Bomb

In logic bomb a condition is checked, if the condition is satisfied then the virus spreads and harmful effects take place.

1.3 NMAP

NMAP is an open source utility which can quickly scan broad range of devices. The valuable information about the devices on your network, ports information is provided.

1.4 Cross Site Scripting

Cross Site Scripting ('XSS') is an attack in which attackers inject script into web pages at the client-side that are viewed by the user to get the sensitive input data of the user. Cross Site Scripting occurs mainly in dynamic web pages. The code for cross site scripting is generally present within the (<script> tag) which is embedded in the HTML documents. The code can be written in JavaScript, VBScript, HTML, or Flash.

1.5 Advantage of Proposed System:-

- The tool can detect various attacks.
- A guide for various attacks.
- The Tool is platform-independent.
- Provide unlimited scanning of IP address.
- No history storage is needed so it requires less amount of memory.
- In this tool there is no programming language barrier.
- Detection of various attacks will be on a click.
- Satisfy the security requirement of the organization.

The rest of the paper is organized as follows: Section II gives a brief idea of the problem that exists and idea how it will be solved by our proposed system. Section III provides comparisons of various existing system. System IV describes the implementation plan along with the algorithm and details of the android manifest file. Finally, the conclusion and the major contribution to this paper are discussed in the remaining sections.

2. PROBLEM DEFINITION

Web applications are in demand due to the availability of web browsers anywhere, anytime and the convenience of using a web browser. The client can update and maintain web applications without installing any software on his mobile phones, Laptops, Computers. This potentially thousands of client computers is a key reason for their popularity, and due to their inherent support for cross-platform compatibility. The information sharing, file sharing, business ideas, and also the small business are grown into the big ones with the development of the Web. The Web applications are used as a means for doing business and delivering services to the end users. This websites are often attacked directly. Hackers either target the corporate network or the end-users accessing the website. They are targeted by subjecting them to drive-by downloading.

In today's life, every user have come across the various attacks that hack there website. The organizations have suffered from a heavy loss of confidential information. Till now there are various security tool that have been developed to minimize this attacks. Considering the Example of Scapy which is used by the user to detect the network manipulation and packet sniffing attack. The Scapy Tool uses the python language to detect the vulnerabilities. So the user must know this language in order to use Scapy .In this system i.e. Website Security Tool a tool will be created for the Web Application. With the help of this tool the Web Application user can test the vulnerabilities of their web Application. The Tool can run on any browser. And the detection of the attacks will be done on a click of a button.

3. LITERATURE REVIEW

The proposed system is compared with the following existing systems as shown in table 1: System Comparison:

Security Tool	Purpose	Programming Language Used	Report Reading	Storage
Openvas	Used for testing various attacks	NASL Scripting Language	Hard to read reports.	It stores the history of past scans.
Metasploit	Developing and testing the exploit code	Java Programming Language	Easy to read reports.	It stores the history of past scans.
Scapy	Network Manipulation Network Scanning Packet Sniffing	Python Programming Language	Hard to read reports.	It does not store the history of past scans.
Website Security Tool	SQL Injection Logic Bomb NMAP	HTML, Java, C Programming Language	Easy to read reports.	It does not store the history of past scans

Table 1: System Comparison

4. METHODOLOGY

In the proposed system the website security tool is created to detect the vulnerabilities of the website.

The tool works by taking the data from the user and using it to detect the attacks the user wished to. The programs that are run at the back end are the major components that are used to detect the attacks to those that there website are vulnerable. Its works in the layer interface that provides which helps the user to send and receive data.

A HTML code will be written. It is used for creating the GUI of the tool. The NMAP detection is done by running the various commands in the command prompt. All this commands will be automated the user just have to enter its IP address. A C program will be written for the detection of the Logic Bomb which will check the presence of the certain logic in the file. The database on which the SQL injection is to be performed is store in XAMP server. The various SQL queries will be run to get the access to the database. The websites links would also be checked to detect whether they are prone to SQL Injection. Thus for communication to be established the layer interface interacts with the various program codes which is essential for sending and receiving data.

The tool will also have the various programs codes which will help the user to know how the various other attacks work. This detection of the attacks can be done using specific algorithms.

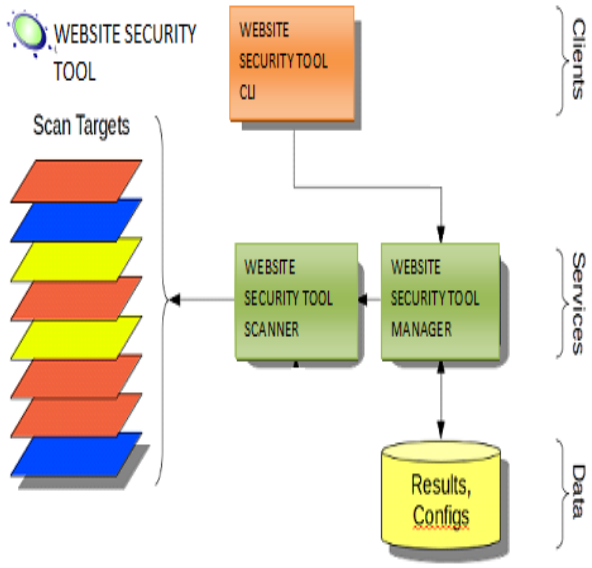

Figure 1: System Architecture

The project aims to detect vulnerabilities of website. The main idea of the Website Security Tool use the user input to detect the attack and give the output to user.

The proposed system architecture in figure 1: System Architecture describes the following stages:

1. Choose the attack: It is the first step in the proposed system. The website security tool is to detect the various attacks like SQL injection, NMAP ,Logic Bomb .The user have the choice to select any of the attack out of this attacks which he wishes to detect.
2. Taking the data from the user: Depending on what type of the attack is selected by the user the tool have the GUI for collecting the data from the user. The users have to enter the data for each of the attacks to be detected. The data can be website link, IP address and File.
3. Confirmation of the data: The data entered by the user will be used throughout the processing of the tool. So it is necessary to confirm the correctness of the data for this a confirmation is being asked to the users who have entered the data.
4. Detection of the attack: Once the user has entered the data properly for the attacks the next step is to run the program for detecting these attacks. The program is run in the back end. The user is unaware of the program that is being running at the back end.
5. Displaying the result: As the attacks are detected the output is shown to the user .The output will be like whether the user website is prone to the attack or not. If prone then what type of the attack it is.
6. Closing the tool: When the attacks are detected and the output is displayed the user can close the tool and use it whenever necessary again.

4.1 Algorithm:

Step 1 - The user when open the tool have the choice to select between the various attacks i.e. SQL injection, Logic Bomb, NMAP.
Step 2 - If the user selects the SQL Injection then the user will enter the website link.
Step 3 - When the link is entered the admin login page is opened.
Step 4 - The Tool will then encounter a SQL query in order to gain access.
Step 5 - If the website is not secured then the query will be accepted and the login access would be granted.
Step 6 - Completion of SQL Injection Technique and the output will be shown to user.
Step 7- If the user selects the Logic Bomb then the user will have to insert a file.
Step 8 - Then the Logic Bomb program is run in the back end on the file that is inserted by the user.
Step 9 - Completion of Logic Bomb and the output will be shown to user.
Step 10 - If the user selects NMAP then the user will have to enter the IP address of the website.
Step 11 - After entering the IP address, the NMAP commands will run in the backend of the IP address entered by the user.
Step 12 - Completion of NMAP and the output will be shown to the user.

The Website Security Tool is a tool implemented for all the web application to secure them from the various attacks and the hackers. The tool will be efficient in detecting the various attacks which the web applications may be vulnerable to in a less amount of time and at just a click. The implementation of the Website Security Tool for the detection will be based on a three-layer model. The layers are application layer, Verification, Detection [Figure 2]. The lowest layer (layer 1) will handle the communication between the user and the tool. This layer is responsible for transmitting and receiving of the data. Layer 1 will maintain all the user data. Layer 1 will communicate only with the next layer above it, which is layer 2. Layer 2 is responsible for verification of the data. It contains code for verifying the data entered by the user. The data is received at layer 1 and passed to layer 2 for verification. Layer 3 is the next layer above layer 2. Layer 3 is the detection layer, and only communicates with layer 2. The job of layer 3 is to detect the various attacks to which the web application is vulnerable. Layer 3 communicates with layer 2 to take all the verified data and run the program on the data. A layered approach allows us to modify specific parts of the implementation in a modular fashion. From a high-level overview, these are the following steps that will occur when the software is working properly.

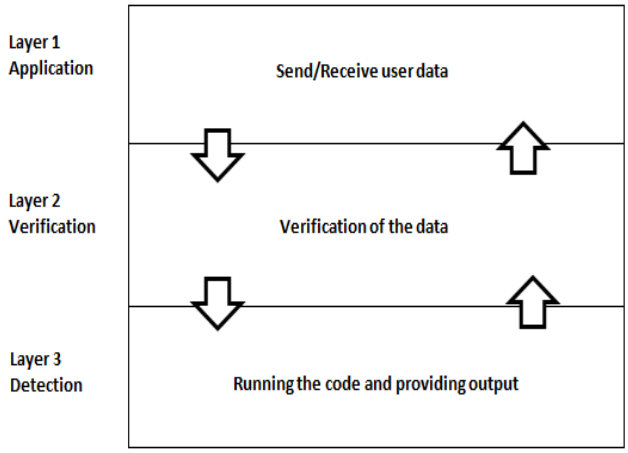

Figure 2: Establishing Basic Detection Process

The aim is to detect the attacks which may render the security of the web applications. If such attacks exist the confidential data will be available to the hackers and the malicious use of the data may take place. As mentioned previously at the layer 1 the data is taken from the user. The GUI [Figure 3] of the layer is created using a HTML Language. The user can choose the attack they want to perform. The user can also having a learning guide to the various attacks.

Figure 3: Website Security Tool GUI

If the user choose to perform the NMAP attack. In the Figure 4 the user has selected the NMAP. The user have the choose to perform the regular scan, ping scan or the intense scan.

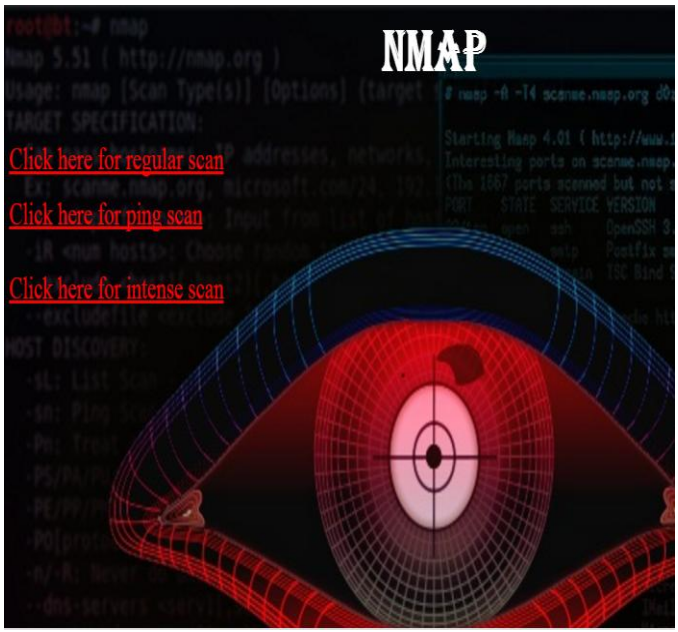

Figure 4: NMAP

In Figure 5 the regular scan is performed. The user just enters his IP address and the NMAP commands are run automatically and the output regarding all the ports will be provided to the user. The user gets all the information about the various ports which are open or close. These ports are used by the hacker malicious to get sensitive data.

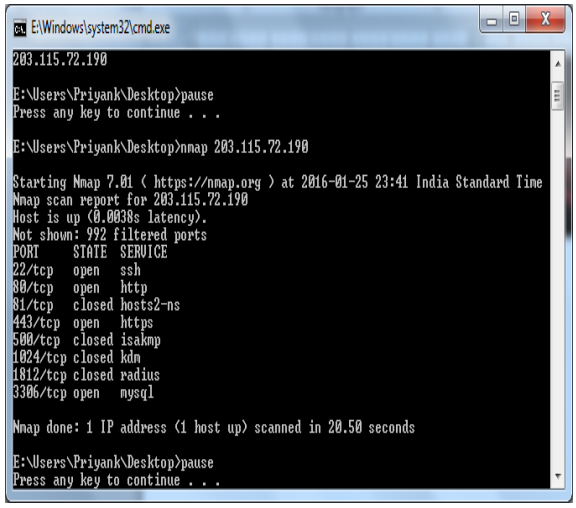

Figure 5: NMAP Commands

In the Figure 6 The output will be directly copied from the command prompt to a text file which the user can refer anytime. The connection between the HTML and command prompt is done using batch file which help the commands to run automatically in the command prompt.

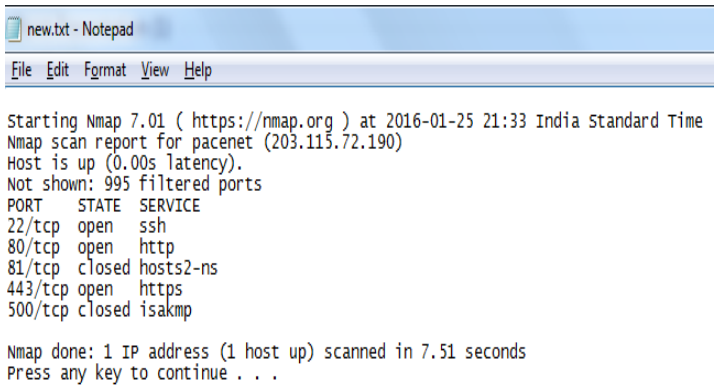

Figure 6: Successful NMAP Output

5. CONCLUSION

This paper describes the architecture for the implementation of Website Security Tool. We Presented can run over Any Web application Independent of the Operating System (Windows 7, Windows 8, Windows XP, Linux,) which make it more portable and ease of use .We demonstrated how the working of the application will take place on the various web applications and the communication between the user and the tool, and we

also demonstrated the NMAP detection using the Website Security Tool and also represented the stages taking place while the user have entered the data and the running of the various program codes to detect the attack.

It seemed to us that there is compelling need for such a security tool as today various new technologies have emerged and are used by the hackers so a tool that will help the user to detect the attacks and maintain the security is much needed. The Website Security Tool operates without the support of any infrastructure, physical or organizational, and thus allows communications to continue. Moreover, it is necessary that predefined condition exist to create a system that has the potential to achieve this goal.

6. REFERENCES

[1] Banja Luka, Bosnia and Herzegovina " A Black-box Testing Tool for Detecting SQL Injection Vulnerabilities"," Informatics and Applications(ICIA)",pg 216 - 221,September 2013.

[2] Yousra Faisal Gad Mahgoup Elhakeem, Bazara I. A. Barry,"Developing a Security Model to Protect Websites from Cross-Site Scripting Attacks using Zend Framework Application"," Computing, Electrical and Electronics Engineering(ICCEEE)", Pg. 624 - 629,August 2013.

[3] Bharti Nagpal, Naresh Chauhan, Nanhay Singh, Angel Panesar," Tool Based Implementation of SQL Injection for Penetration Testing","Computing, Communication and Automation (ICCCA)", Pg. 746 - 749, May 2015.

[4] Huyam AL-Amro and Eyas El-Qawasmeh" Discovering Security Vulnerabilities And Leaks in ASP.net Websites","Cyber Security,Cyber Welfare,Digital Forensic(Cybersec)",pg. 329 - 333,June 2012.

[5] Josip Bozic,Franz Wotawa," PURITY: a Planning-based security testing tool"," Software Quality, Reliability and Security - Companion (QRS-C)",pg 46 - 55,August 2015.

[6] Xiang Fu Xin Lu Boris Peltsverger Shijun Chen," A Static Analysis Framework For Detecting SQL Injection Vulnerabilities"," Computer Software and Applications Conference ",Volume 1,pg 87-96,July 2007.

[7] Mukesh Kumar Gupta , Mahesh Chandra Govil , Girdhari Singh," Predicting Cross-Site Scripting (XSS) Security Vulnerabilities in Web Applications "," Computer Science and Software Engineering (JCSSE) ",pg 162 - 167,July 2015

[8] Yu Sun, Dake He," Model Checking for the Defense against Cross-site Scripting Attacks "," Computer Science & Service System (CSSS) ",pg 2161 - 2164,August 2012

7.BIOGRAPHIES

Mrs.Vijaya Umesh Pinjarkar(Khirodkar) is working as an Assistant Professor in the Department of Information Technology since March 2007. She received the B.E. Degree in Information Technology from Anuradha Engg College Chikhali, Amravati University & is pursuing M.E. in Information Technology from VES Institute of Technology, chembur Mumbai University.

Vaibhavi Rane born on 27st September 1994, Malad, Mumbai, India. She is pursuing the Bachelor of Engineering Degree in Information Technology from K. J. Somaiya Institute of Engineering & Information Technology, India.

Mayur Shelar born on 21st May 1994, Parel, Mumbai, India. He is pursuing the Bachelor of Engineering Degree in Information Technology from K. J. Somaiya Institute of Engineering & Information Technology, India.

Chaitrali Rane born on 9th May 1995, Kawade, Alibaug, India. She is pursuing the Bachelor of Engineering Degree in Information Technology from K. J. Somaiya Institute of Engineering & Information Technology, India.