RESEARCH ARTICLE OPEN ACCESS

# "Web Application Scanning and Identification of Vulnerabilities for Different Attacks"

Haridas V.Kanure, Jayesh H.Ambawale, Satish S.Chougale
Department of Information Technology, Trinity College of Engineering &Research,Pune,India
Department of Information Technology, Trinity College of Engineering &Research,Pune,India
Department of Information Technology, Trinity College of Engineering &Research,Pune,India

**Abstract:-**
Now days there are lots of websites present in internet. And providing different services to the client is not only important task for website owner but also protecting their website from different attacks. Now days Identification of vulnerabilities in web application is becoming serious issue. Protecting website by identifying vulnerabilities and taking important attack prevention steps is very much important task.Different vulnerabilities identification is based on different test cases which we have and which we are going to build by use of this tool. Those test cases come from combination of some effective code. SQL injection test case can work with effective database query. And Cross site scripting can works with some java script code or other scripting language code etc.

In this tool we are using large collection of different test cases of vulnerabilities for identifying many attacks. Use of large collection of test cases will check all the possibilities of vulnerability and return accurate results. So percentage of accuracy is always better than other tools. We are making this tool more intelligent so it can build some necessary test cases at the time of scanning as per testing condition. Tool implements the technique for the PHP programming language.This paper presents implementation and identification of vulnerabilities in PHP Web applications.The tool detects vulnerabilities in PHP web applications by identifying cross-site scripting, SQL injection and SQL authentication broken, more vulnerabilities.

This tool is automated so it has ability to build query as per testing condition. This automated tool will help a lot to deal with available and latest vulnerability for identification.

**Key Words**- Cross-Site Scripting (Reflected Cross Site Scripting), SQL Injection, HTTP Banner Disclosure, SQL Broken Authentication.

## I. Introduction:-

Research made on topics:
Make tool more intelligent for building test cases as per testing condition need or requirement.
Use of collection of all possible test cases to improve accuracy of tool.
Faster scanning with highly effective code.

Many of the web applications collect sensitive data about their users, and so it is very important to maintain security of that web application. Vulnerabilities in web application cause stealing data, hijacking user sessions, or phishing. Our tool is able to detect top vulnerability in php web application.

Tool can detect following vulnerabilities:

1. SQL Injection
2. Cross site Scripting
   a) Reflected XSS
   b) Stored XSS
3. SQL broken authentication
4. HTTP banner disclosure

This Tool crawl a website looking for vulnerabilities within web applications. The solution analyzes all web pages and files that it finds, and builds a structure of the entire website. The scanner then performs automated checks against security vulnerabilities.

A successful SQL injection exploit can read sensitive data from the database, modify database data (Insert/Update/Delete), execute administration operations on the database (such as shutdown the DBMS), recover the content of a given file present on the DBMS file system and in some cases issue commands to the operating system.

Cross-Site Scripting (XSS) attacks are a type of injection problem, in which malicious scripts are injected into the otherwise benign and trusted web sites. XSS attacks occur when an attacker uses a web application to send malicious code, generally in the form of a browser side script, to a different end user.

HTTP banner Disclosure vulnerability can include information such as the operating system (Linux, Windows, etc.), the operating system version, what kind of web server you are running (IIS, Apache, etc.).

## II. Background Study

**Cross Site Scripting (XSS)**:-

Cross-Site Scripting (XSS) attacks are a type of injection problem, in which malicious scripts are injected into the otherwise benign and trusted web sites. XSS attacks occur when an attacker uses a web application to send malicious code, generally in the form of a browser side script, to a different end user. Flaws that allow these attacks to succeed are quite widespread and occur anywhere a web application uses input from a user in the output it generates without validating or encoding it.

E.g. following some JavaScript code are used to test webpages for cross site scripting vulnerability
<Script>alert (“Cross Site Scripting”) </Script>
If Any site respond to above script then attackers can guess that cross site scripting flaw is present in that webpage.There are lots of test cases are present by which we can test for cross site scripting vulnerabilities.

Our tool can work with two types of cross site scripting vulnerabilities:

1. Reflected cross site scripting
2. Stored cross site scripting.

**1. Reflected cross site scripting:** The end-user is tricked into clicking a malicious link or submitting a manipulated form. The injected code is sent to a vulnerable web server that directs the cross-site attack back to the user’s browser. The browser then executes the malicious code, assuming it comes from a trusted server.

**2. Stored cross site scripting.**

Injected malicious code is stored on a target server such as a bulletin board, a visitor log, or a comment field. When interacting with the target server, an end-user inadvertently retrieves and executes the malicious code from the server.

**SQL Injection**: -

A SQL Injection attack consists of insertion or "injection" of a SQL query via the input data from the client to the application. A successful SQL injection exploit can read sensitive data from the database, modify database data (Insert/Update/Delete), execute administration operations on the database (such as shutdown the DBMS), recover the content of a given file present on the DBMS file system and in some cases issue commands to the operating system. SQL injection attacks are a type of injection attack, in which SQL commands are injected into data-plane input in order to effect the execution of predefined SQL commands.

Our tool is automated tool which is very useful for building Query for testing SQL Injection. This query is built with specific conditions and in proper manner where website can receive those query as input. Server Display result of those queries. And we also maintained list of error for testing SQL injection Against Webpage. Mainly tool Detect SQL Injection on the basis of error matching technique. Where result of every query is matched with list of error which is maintained by our tool. If match is found then user will get result of Found SQL Injection in Webpages.

**HTTP Banner Disclosure:-**

Web servers often broadcast server information by default. This can include information such as the operating system (Linux, Windows, etc.), the operating system version, what kind of web server you are running (IIS, Apache, etc.) and in some cases web server modules installed.

This tool hasList of Operating system and server name.For detecting server information such as operating system, web server info.

We can send request to target URL for testing of HTTP banner Disclosure and after getting response from target server, we check those response result with list of operating system and web server which is maintained by our tool. This is done because when we receive response from target server at that time it comes with server information. We check this information with our list of operating system and web server. If any response resultsmatch then User Will get Found HTTP banner disclosure vulnerability result.

**SQL Broken Authentication:-**

Users are impersonated due to leaks or flaws in the authentication process.Attacks occur when a session ID is visible to others, timeouts are not properly set, SSL/TLS is not used, or any other flaw in the authentication scheme is detected. Flaws used against one account may be replicated against an account with higher privileges.

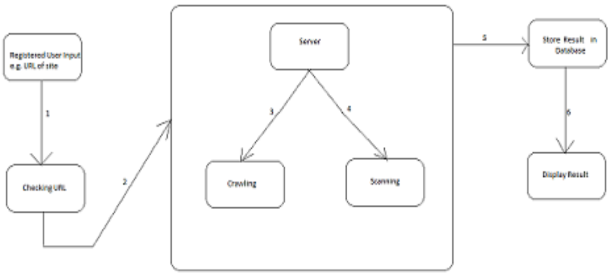
## III. ProblemDefinition:-

Some web services contain known exploits that allow a malicious attacker to use that script as a gateway to send emails, potentially using your organization to launch spam runs; SQL injection exploits might allow an attacker to get hold of usernames and passwords, or inserting his own username, or even to run code remotely. Likewise the use of applications with known vulnerabilities can open your organization to targeted attacks. Malicious hackers might try to send people malicious payloads targeted at these vulnerable applications that, when triggered, would run the code the hacker would have embedded in the payload sent.so Protecting website by identifying vulnerabilities and taking important attack prevention steps is very much important task.

Tool use different test cases to identify vulnerability present in web application. And it is automated so it can able to build query or test cases as per testing condition. Conditions of testing will change when some new vulnerabilities and technique comes in web applications.

Then we have to use some other technique to identify vulnerabilities other than testing with present test cases. Otherwise it might minimize the accuracy of this tool. And at that time this automated tool is used for building test cases.

## IV. System Design:-

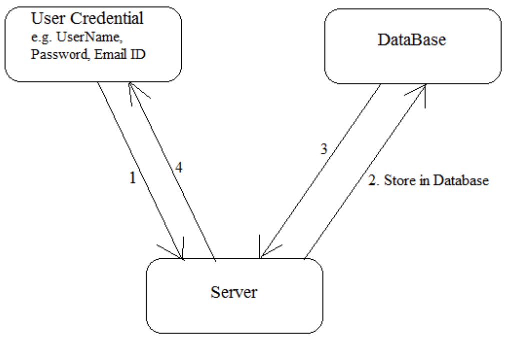


User can receive soft copy of their scan report.This Application check vulnerabilities of web application related to Reflected Cross-Site Scripting, Stored Cross-Site Scripting, Standard SQL Injection and more attacks.

Following important module is covered in our concept.

**[4.1]LOGIN/REGISTRATION MODULE**:-

Every user who wants to scan their website should be a registered user of the application. We will be accepting



Registrtation

Details like user name, email id, mobile No. etc. from user. And provide username and password to their users.

**[4.2] SCANNING:-**

User must have to enter their website URL to perform scan for finding vulnerabilities. Website is scanned for finding any kind of SQL, XSS and other vulnerability.

**[4.3]DISPLAY FOUND VULNERABILITY**:-

Scanning of website is perform through various phases and after that if scanner found any SQL ,XSS and other vulnerability in website then it will inform to that user about those vulnerability.

1] **For Registered User**:-

Step (1):- For a new user, the user will first create a new account. The user details will be stored at the storage database. After logging in the user will get registered account.

Step (2):-With user's login id; user can scan their particular website for vulnerabilities. User first has to give the URL of website which he has to scan. The URL must be correct incorrect URL will not accepted by the system.

Step (3):-After that user has to select what type of vulnerabilities he has to check like SQL Injection, cross site scripting, HTTP Banner disclosure.

Step (4):- Registered user can perform various scanning as SQL Injection Attack, Cross-site scripting, Banner Disclosure, SQL Broken Authentication.

Step (5):-Then user has to click on start scan button. In scanning first step is crawling at which the links on entered URL are selected and scanning is done on each page through queries in our system's database.

Step (6):-After completion of scanning Registered user can get their report of scanning in PDF format. System shows result and prevention techniques of detected vulnerabilities to Registered User's.

## V. Conclusion:-

Tool has collection of test cases of different vulnerabilities of web application. And it is automated tools so it is able build query or test cases as per testing condition. It crawl website and find web pages for testing of vulnerabilities. By use of effective test cases tool perform with better accuracy. And gives good result as compare to others. Tool can able to detect following vulnerabilities: SQL Injection, Cross site scripting, SQL broken Authentication, HTTP Banner Disclosure.

## VI. Acknowledgements:-

Our sincere thanks to our project guide Prof.RahulAkhouri, Prof.Sarita.S.Gaikwad the head of department and Prof. PrakashDabir, the Principle of Trinity College of engineering and research (TCOER), for their valuable guidance and constant encouragement.

## References:-

[1] Julian Dolby,Finding Bugs in Web Applications UsingDynamic Test Generation and Explicit-State Model Checking,2010

[2] M. Johns and C. Beyerlein, “SMask: Preventing Injection Attacksin Web Applications by Approximating Automatic Data/Code Separation,” Proc. ACM Symp. Applied Computing, 2007.