Web Vulnerability Analysis

ASHWINI CHAMBHARE [1], JAYDEEP BHUIBHAR [2], HRUSHIKESH PANCHBUDHE [3], PALLAV KULKARNI [4], RAJATKUMAR KARMARKAR[5], DEVIKA DESHMUKH[6]
[1,2,3,4,5,6] *Dept. of Computer Technologies, Rajiv Gandhi College of Engineering & Research, R.T.M.N.U, India*

Abstract -- As the popularity of the web increases and web applications become tools of everyday use, the role of web security has been gaining importance as well. The last years have shown a significant increase in the number of web-based attacks. Too many nouns web application security vulnerabilities result from generic input validation problems. Examples of such vulnerabilities are SQL injection and Cross-Site Scripting (XSS). Although the majority of web vulnerabilities are easy to understand and to avoid, many web developers are, unfortunately, not security-aware. As a result, there exist many web sites on the Internet that are vulnerable. This project implemented an automated vulnerability analysis that for the injection attacks. To this end, we implemented a system that automated scanned the injection attack vulnerabilities. Our system automatically analyse web sites with the aim of finding exploitable SQL injection and XSS vulnerabilities. It is able to find many potentially vulnerable web sites.

I. INTRODUCTION

Security is a critical part of your Web applications. Web applications by definition allow users access to a central resource — the Web server — and through it, to others such as database servers. By understanding and implementing proper security measures, you guard your own resources as well as provide a secure environment in which your users are comfortable working with your application. [1]

Web application security is a branch of “Information Security” that deals specifically with security of websites, web applications and web services. Web Security blocks web threats to reduce malware infections, decrease help desk incidents and free up valuable IT resources. It has more than 100 security and filtering categories, hundreds of web application and protocol controls, and 60-plus reports with customization and role-based access. You can easily upgrade to Web Security Gateway when desired to get social media controls, SSL inspection, data loss prevention (DLP) and inline, real-time security from Websense ACE (Advanced Classification Engine). Vulnerability is a hole or a weakness in the application, which can be a design flaw or an implementation bug that allows an attacker to cause harm to the stakeholders of an application. Stakeholders include the application owner, application users, and other entities that rely on the application. The term "vulnerability" is often used very loosely. [2]

II. LITERATURE REVIEW

The rapid and tremendous growth of Information and Communication Technology (ICT) has increased access to web applications. This increased access has paved the way for disadvantageous security and vulnerable threats in the form of attacks in web applications. Various detection and prevention techniques have been proposed by researchers in the field of web applications and technologies development. Through relevant literature and existing research presents a viewpoint of different web application vulnerabilities and security threats and also outlines some open research issues in accordance to the state-of-the-art. The following diagram depicts these security layers as a holistic outlook that looks at security as hardened measures taken to minimize intrusion risks and maximize the protection around the key asset of any organization, its data [2]. Between 2014 and 2017 Context performed penetration tests for companies operating across a broad range of business sectors. For the purposes of trying to discover whether any particular industry used web applications that were more susceptible than others to security issues, we have used the following industry groupings: Media and Advertising, UK Government (the dataset does not include classified government findings), Healthcare, Technology and Telecommunications, Insurance and

Law, Financial Services (Europe); and Other (a group including organisations in the charity, not-for-profit, education, extraction, gambling, recruitment, and services sectors). In 2017, most sectors had an average issue count between 11 and 14 issues per application. The notable exception was the Media and Advertising sector, which had roughly 40% more issues per application than the sector with the next highest count. Additionally, for most sectors, the vulnerability categories which accounted for the highest proportion of the total issues found were the Information Leakage, Authentication, and Server Configuration categories. [3]

III. PROPOSED SYSTEM

Vulnerabilities Analysis basically software designed in java platform that aims to scan the Website's and finding out vulnerabilities that sustains in the websites and providing report to the client. Analysis is going to take the URL of the website as an input, to test for vulnerabilities as per the selected vulnerability. It will check whether the website exists in the domain or not. If not, it will ask the user to enter a valid website. It checks whether the website really exists or not by checking its status code. If status code is 404 the URL does not exist. For a valid URL, it will crawl the Website with the help of our own developed Crawler "basic_crawler" basically made in Java. The Crawler will parse the website. The output of this phase is multiple frameworks, form details and platform used in developing web pages of respective website. After parsing the website, Analysis will test all the forms and links of the website for the vulnerabilities. The Analysis is scanning for two top most vulnerabilities i.e. SQL Injection and Cross Site Scripting. It has different criteria's for both SQL Injection and Cross Site Scripting. [1]

To check for SQL injection the analysis creates two test cases. After preparing the test cases, cases are passed to every links and web forms of inputted website. With these test cases, the test will be performed on the website. Analysis will build the HTTP request to send it to the respective URL and in response URL will give HTTP response. The response will contain all the details and the test will be performed on these details. If any kind of syntax error is found as per a database server in response text or any internal error in a webform is found then that particular webform or link is vulnerable to SQL Injection. To check for the syntax error the analysis has all the patterns of syntax error in the 'pattern.xml' file. If any of the pattern found in response text is matched with the patterns in pattern.xml file then the particular webform is vulnerable to SQL Injection. [2]

To check for cross site scripting (XSS) the analysis creates a test case: Automatic Insertion of "<script>document.write('css attack')</script>" in every textbox available in the web form. After preparing the test cases, cases are passed to every links and web forms of inputted website. With these test cases, the test will be performed on the website. Analysis will build the HTTP request to send it to the respective URL and in response URL will give HTTP response. The response will contain all the details and the test will be performed on these details. If the same inputted content "<script>document.write('css attack')</script>" is found in response text then that particular webform or link is vulnerable to cross site scripting. [3]

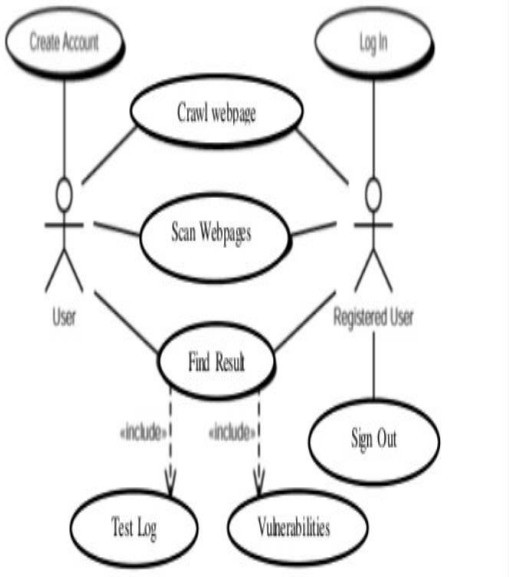

Fig. 1: Use Case Diagram for Web Vulnerability Analysis

Basic steps involved in operating the portal:

1. Register/Login:

The user needs to Register or Login to use the services like Web Scanning, Web Crawling and PDF generation.

2. Scan/Crawl:

Once the user get registered on the portal. He/She able to perform website scan as well as crawling the website and it's services.

3. Report Generation:

Once the User done with the Scanning or Crawling. He/She can able to view and download the full detailed report which is in PDF format which contains the website vulnerabilities and links.

4. Support:

If user or client faces any problem he/she can chat with customer support and solve their problems regarding usage and portals. It will also able to see latest updates to the applications regarding security databases.

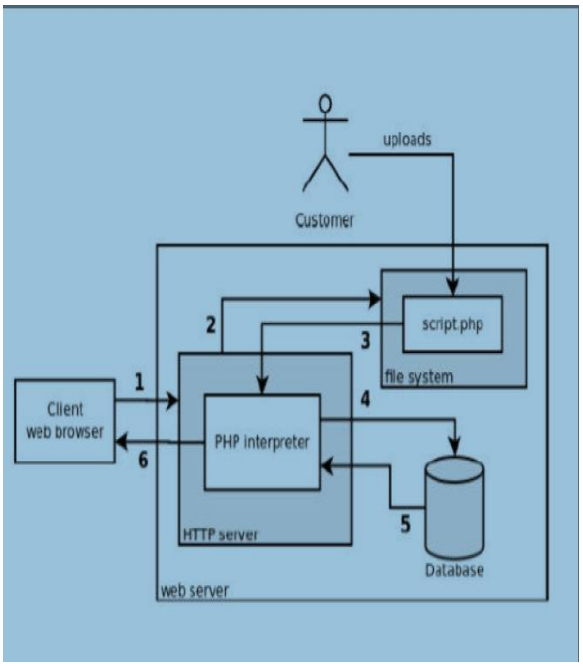

Fig. 2: Block Diagram Diagram for Web Vulnerability Analysis

IV. CONCLUSION

In this report, we presented a Vulnerability analysis that aimed at detecting web application vulnerabilities. We have introduced our own crawler name "basic_crawler" and described how the website is vulnerable to SQL Injection and XSS based on automatically generation of specially crafted request allowing the successful exploitation of detected vulnerabilities.

REFERENCES

[1] www.owasp.org/index.php/Top_10_2013

[2] Kevin J Vella, "The True Nature of Web Application Security: The Role and Function of Black Box Analysiss"21 Feb. 2007 ; http://www.acunetix.com/websitesecurity/blackbox-analysiss/

[3] Vieira, "Using Web Security Analysiss to Detect Vulnerabilities in Web Services"; IEEE/IFIP Intl Conf. on Dependable Systems and Networks, DSN 2009, Lisbon, Portugal, June 2009; http://eden.dei.uc.pt/~mvieira

[4] Jan Tudor, "Web Application Vulnerability Statistics 2013"; June2013; whitepapers@contextis.co.uk

[5] IEEE paper on "Security scanners to detect vulnerabilities in web services" By Marco Vieira.