Web Application Vulnerability Exploiter - WAVE

J. Dhiviya Rose
Assistant Professor(SG)
Cybernetics Cluster, SOCS,
University of Petroleum and Energy Studies (UPES),
Bidholi, Dehradun.
INDIA 248007.

Abhinav Tyagi
Student, School of Computer Science, University of Petroleum and Energy Studies (UPES), Bidholi, Dehradun.
INDIA 248007.

Tushar Rathee, Student,
School of Computer Science,
University of Petroleum and Energy Studies (UPES),
Bidholi, Dehradun.
INDIA 248007.

Kushagra Chopra, Student,
School of Computer Science,
University of Petroleum and Energy Studies (UPES),
Bidholi, Dehradun.
INDIA 248007.

***Abstract*–Security of web applications aims in protecting the attacks and safe guarding the app which can be performed using some manual ways and system automated techniques. It covers methodologies which includes testing every type of web technology, penetration testing ,secure development and protect users from risk of a data breach by finding vulnerabilities.The proposed system. Web Application Vulnerability Exploiter (WAVE) is basically a vulnerability scanner which scans for security vulnerabilities in web applications. The vulnerability scanners scans for the various security threats like cross-site scripting (XSS), SQL injection, and cross-site request forgery (CSRF).Our strategy helps to automate the detection of inputs that securely discriminate against vulnerable people on installed servers and pools due to recent exposure. This enables faster updates to the risk scanning tools as the risk of new software is detected, allowing controllers to scan and protect their networks very quickly. With the help of these tools one can find the errors that can improve the functioning.**

Keywords - ***Vulnerability, SQL Injection, Security***

I. INTRODUCTION

Security of web applications aims in protecting the attacks and safe guarding the app which can be performed using some manual ways and system automated techniques. It covers methodologies which includes testing every type of web technology, penetration testing ,secure development and protect users from risk of a data breach by finding vulnerabilities.The proposed system The commonly used methodologies includes testing every type of web technology, penetration testing, secure development, protect users from risk of a data breach by finding vulnerabilities and scanning for vulnerabilities. The proposed system Web Application Vulnerability Exploiter (WAVE) is basically a vulnerability scanner which scans for Security Vulnerabilities in web applications. The self-customized tools that helps in autonomous execution of scanning process in the web app falls under the roof of vulnerability scanners. The tool will check for various security breached like SQL injection, cross-site request forgery (CSRF).cross-site scripting (XSS),and On securing the systems various properties an tool should have includes

- ✓ It helps in secure development of web site.
- ✓ It protect the users from risk of a data breach by finding Vulnerabilities
- ✓ It is used for Scanning for Vulnerabilities, which are being present in the web site.
- ✓ Secure the data of the user in web.
- ✓ Protects from the unauthorized excess of the other person.

Weakness is that it is never a perfect solution, it’s an essential process – and there are ways of maximizing the benefits while minimizing the drawbacks. Thus the opportunity of any application vulnerability exploiter is that it is helps to find the errors that can improve the functioning.

II. BACKGROUND STUDIES

There are various studies involved in the various areas of web security breach. The various limitations includes with respect to your work in future are OS Command Injection, Cross Site Scripting, Configuration. Management issues like session hijacking and IDOR vulnerabilities seems to be increased in recent times[1].SQL injection is one of the common type of vulnerability that most attackers do where the malicious SQL query where made to execute in the database through some scripts.

The hackers use the access of the SQL queries to fetch the data from the main database with the help of SQL queries.
SQL which is used for communication to the database called the structured query language will make it to communicate to the database in a user specific view. Many of the servers that store critical data for websites and services use SQL to manage the data in their databases[2]. This type of attackers are most common in many real life situation where each of the attack is targeted in fetching the data in the server with the help of SQL command. This type of attack will become critical if the attackers fetch some credential data form the attacks[3].

Cross-Site Scripting (XSS) is a type of attack where the attacker will enter a corrupt code into the controls of the forms and its entered, once the visitor of the page opens the form the malicious scripts which is present in the control gets executed automatically. This attack is predominantly used by the hackers to damage a popular website of reputed institutaion or organization that makes to use the customers or clients information[3]. Thus the users more credentials informations like account number, password, username etc. is hijacked initially which makes the user to rethink the use of the same website and makes them to avoid using that popular website from usage.
Cross-Site Request Forgery (CSRF) attack is performed by malicious use of already logged website but some user and

not terminated. This will give access to the hacker to used the unclosed session to perform some money transaction and other kinds of modification like user information change, password change etc.[4]. This type of attacks are normally prevented by adding validation and verification process by organizations for unauthorized user entry by rechecking with OTP and other mail verifications code.This will enable them to identify the user's browser and session to verify their authenticity[5].

System Misconfigurations: Network assets that have disparate security controls or vulnerable settings can result in system misconfigurations. Cybercriminals commonly probe networks for system misconfigurations and gaps that look exploitable. During this COVID crisis most of the company and organization is changing the move from offline platform to online this type of attack seems to be of more target due to the lack of security tools implementation inside the software network. Outdated software, lack of firewall, use of plagiarized software's are the most important loopholes for this type of attacks.To minimize these kinds of risks, it is essential to establish a patch management schedule so that all the latest system patches are implemented as soon as they are released.

Missing or Weak Authorization Credentials- This type of attacks like brute force attacks occur where the user guess and try the possible credentials to hack the system. This is normally prevented by education the employees by various cyber security training and awareness camps in various organizations.

Missing or Poor Data Encryption- In this attack the communication channel is been hacked by the hackers and the encrypted message is been studies and decrypted by observing the pattern of encryption. This is commonly performed by eavesdropping and most common if the website don't have a strong encryption mechanism. This type of attacks can be prevented by having a strong encryption mechanism[6]. This can seriously undermine an organization's efforts towards cyber security compliance and lead to fines from regulatory bodies.

Zero-day Vulnerabilities- In this attack the hackers try to monitor the action that has less severity and gradually get the insight on the website. These are especially dangerous as there is no defense against such vulnerabilities until after the attack has happened[7].

III. PROPOSED SYSTEM DESIGN

In this we are discussing about the vulnerability scanning which are used for exploiting the web application using vulnerability. The proposed system helps in the penetration testing and scanning for vulnerabilities, which are being present in the site. This is used for secure development of the things and protect users from risk of a data breach by finding vulnerabilities, which helps in proper functioning.The users would be the security provider, open-source project owners ,IT teams at various firms etc.The project will only work on the platform or the IDE that supports solidity programming language. The flow process of the proposed model and the use case is shown in Figure 1.

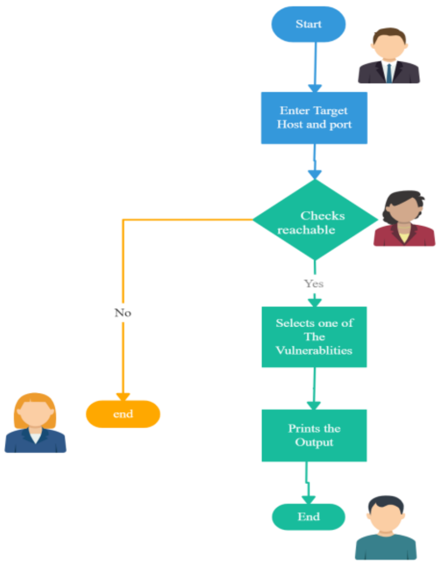

Figure 1. Process Flow of WAVE

Identifying and patching vulnerability-In this the step 1st is to find out all the vulnerability and to make this vulnerability to find out more easily and can patch them. These consist of two things -

Penetration Testing-This stage is performed by finding the possible ways of penetration into the website. This is ethical hacking and helps the system to identify the loopholes and it is documented. It describes the intentional launching of simulated cyber-attacks that seek out exploitable vulnerabilities in computer systems, networks, websites, and applications. This once performed helps to his once performed helps to make organizational policies and mainly performed during the testing phase.

Regular intelligence testing-This type of testing is performed using the various AI and Machine Learning (ML) algorithms which helps in identifying the loops in the website product. This make the system free from error when compared to the manual make. These days many software agents and bots are used in this process which make the system robust.

Security Measures – There are some of the security measures which are to be taken which provide the security the web sites that are –

Data Encryption- Data encryption is a way of translating data from plaintext (unencrypted) to cipher text (encrypted). Users can access encrypted data with an encryption key and decrypted data with a decryption key. Protecting your data. Types of data encryption: asymmetric vs symmetric. Benefits of data encryption.

Strong accesses control- The accesses control of the web sites should be strong so that no person can accesses them easily.

Authentication measures-Authentication measures means the techniques, methods and processes used to verify the identity of the customer and the debit card transaction.

Secure coding practices-Secure coding is a set of practices that applies security considerations to how software will be coded and encrypted to best defend against cyber attack or vulnerabilities.

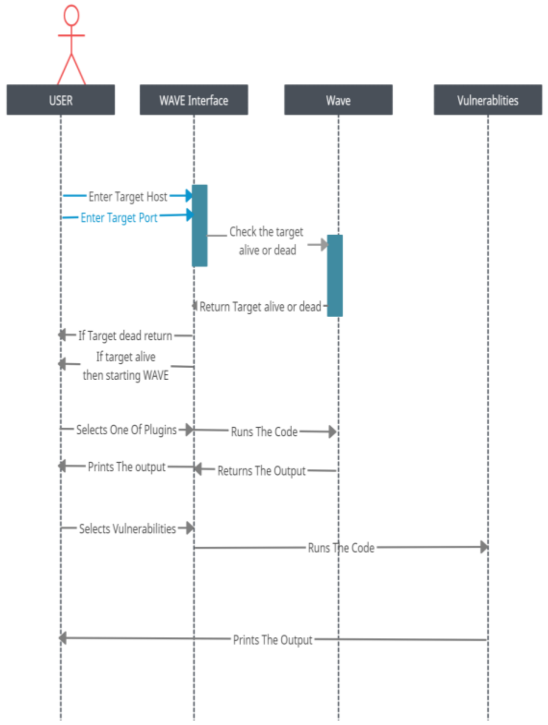

Figure 3. Sequence diagram

IV. SYSTEM IMPLEMENTATION

The number of online services is slowly increasing. Apps are more integrated than ever. Thus, the potential for attack on a particular business is growing. In general, a single vulnerability or vulnerability to security may seriously undermine the security of all systems. Sometimes, several tandem errors can be fatal if tied together by a skilled attacker. This thesis explores the use of non-invasive risk assessment methods. Several risks and methods of detection are discussed, and a psychiatric evidence (PoC) scanner that can detect certain risks, while not interfering, is used. Several methods have been used on the top 1 million Alexa sites in WWW. And a small percentage of the sites appear to be at risk for each scanner scanner. This points to the potential conclusion that even the most powerful actors on the Internet need to take security precautions. The so-called "fruit hanging" still exists; the digital security of many organizations can be greatly improved with simple security measures. There is no need for expensive solutions, such as advanced access systems, where one is better served by focusing on the basics. The web application has undergone a dramatic change over the past few years; many new technologies reshape the pattern of Web applications. As more and more developers upgrade to HTML5 technology, more websites are using HTML5 more slowly. To illustrate the power of this approach, we present a Heart bleed study. Using web-based vulnerabilities scanners is very popular as they promise to detect risk with minimal set-up effort. However, applying them effectively in practice is often difficult. The two main reasons for this are limitations regarding crawling skills and problems with certified scanners. In this paper, we introduce JARVIS, which provides technological solutions that can be applied to a wide variety of risk scanners to overcome these limitations and significantly improve their performance. To test JARVIS, we used it on five free risk scanners and evaluated the performance of the risk detection in the context of seven intentionally unprotected web applications. Preliminary general tests showed that by using JARVIS-containing scanners, the amount of risk detected could be increased by more than 100% on average compared to using non-JARVIS scanners. In the age of information, computer programs have a direct impact on the safety and well-being of people and therefore we must have low risk tolerance. Risk scanning technology can detect network security risks, so network operators can detect in advance when network risk exists and thus ensure the security of the network system. NVT is a network risk plug-in provided by OpenVAS, which is highly priced and provides a daily renewal service. A combination of many NVTs is used to determine risk. Based on an in-depth analysis of current network-based scanning information and the use of NVTs, this paper designs a NVTs-based network vulnerability scanning system, performs a network vulnerability scanning system, and ultimately introduces its use. hope.

V. CONCLUSION

The new technology provides users with a variety of online applications, but introduces new security issues at the same time. Currently, most web application scanners cannot detect security issues with HTML5 features, which makes HTML5 security issues blind spots in security scanning process. The paper focuses on research among existing Web scanners for the first time. We then selected the W3af (Web Application Attack and Audit Framework) as the basis for change, and to customize the scanning and documentation modules, we designed a web-based security scanning service. Visual scan results show that it can not only detect the risk of Click5ing HTML5, but also provide effective web application security scanning and web testing services. We are exploring a new way of secure fingerprints to perform risky scanning of network servers. Our strategy helps to automate the detection of inputs that securely discriminate against vulnerable people on installed servers and pools due to recent exposure. This enables faster updates to the risk scanning tools as the risk of new software is detected, allowing controllers to scan and protect their networks very quickly. To ensure that such scanners are safe and ethical, we need to reject inputs that have harmful side effects. We have created a framework, based on delta usage, that examines the discriminatory properties of that input, as well as your safety. We use fuzzier to find promising candidate input to automatically improve the process.

REFERENCES

[1] A. Hahn, “Cyber security of the smart grid: Attack exposure analysis, detection algorithms, and testbed evaluation,” *Iowa State Univ.*, p. 140, 2013, [Online]. Available: http://lib.dr.iastate.edu/cgi/viewcontent.cgi?article=4105&context=etd.

[2] M. A. Lawal, A. B. M. Sultan, and A. O. Shakiru, “Systematic literature review on SQL injection attack,” *Int. J. Soft Comput.*, vol. 11, no. 1, pp. 26–35, 2016.

[3] K. Elshazly, Y. Fouad, M. Saleh, and A. Sewisy, “A Survey of SQL Injection Attack Detection and Prevention,” *J. Comput. Commun.*, vol. 02, no. 08, pp. 1–9, 2014, doi: 10.4236/jcc.2014.28001.

[4] G. Parimala, M. Sangeetha, and R. Andalpriyadharsini, “Efficient Web Vulnerability Detection Tool for Sleeping Giant-Cross Site Request Forgery,” *J. Phys. Conf. Ser.*, vol. 1000, no. 1, 2018, doi: 10.1088/1742-6596/1000/1/012125.

[5] Sentamilselvan K, “Survey on Cross Site Request Forgery (An Overview of CSRF),” vol. 29, pp. 30–2013, 2013, [Online]. Available: http://127.0.0.1/klog.js.

[6] C. Cerrudo, “Hacking Robots Before Skynet 1,” *Cybersecurity Insight*, pp. 1–17, 2017.

[7] S. Son, K. S. McKinley, and V. Shmatikov, “Fix Me Up: Repairing Access-Control Bugs in Web Applications,” *Netw. Distrib. Syst. Secur. Symp.*, pp. 1–16, 2013, [Online]. Available: http://www.cs.utexas.edu/~shmat/abstracts.html%5Cnpapers3://publication/uuid/8F0084A2-6070-4624-A328-473A804CA03C.