Detection of Cross Site Request Forgery Attacks on the Web Using Machine Learning Based Vulnerability Technique

Pinninti Siva teja[1] , Mohammed Jabeen Maleka[2] , Panjala Shreya[3] , PVS Srinivas[4]

smilysivateja01@gmail.com, mdjabeenmaleka5786@gmail.com, shreyapanjala19@gmail.com, pvs.srinivas@vbithyd.ac.in

Department of Computer Science and Engineering, Vignana Bharathi Institute of Technology
Telangana, India, 501301

Abstract

In this paper, we propose a machine learning based technique for detecting the security flaws on the web. Due to their unique nature and widespread usage of custom development methods, web applications are notoriously difficult to assess. Machine learning is very useful for web application security because it blends automated analytic tools with human understanding of web application semantics. It is possible for a user's browser to be deceived into doing harmful activities on other, legitimate websites after visiting a malicious website. Attacks of this kind are categorised as Cross Site Request Forgery (CSRF). As a result, web development and security organisations ignoring them for the most part, due to which numerous websites on the internet are susceptible to these types of attacks, giving them the moniker of "sleeping giant" of web-based vulnerabilities. As one of four serious CSRF vulnerabilities, we discovered on four important sites, we detail the first recorded assault against a financial institution. These vulnerabilities allow an attacker to breach a user's account and steal their personal information, including their bank details and email. We have implemented changes to the server that eliminate CSRF attacks totally, and we recommend that other websites do the same. Using this method, we created Mitch, the first machine learning (ML) fix for CSRF vulnerabilities. Mitch helped uncover 35 more CSRFs across 20 critical websites and 3 other CSRFs in usable software.

Keywords: Web application security, Machine Learning (ML), Cross Site Request Forgery(CSRF)

1. Introduction

Web applications have become extremely important due to their ability to make life easier and cause fewer overall issues. Its importance in daily life has some disadvantages. Outsiders and hackers, for example, are drawn to it because it handles all types of communication, including harassing and financial security operations. Attackers are unable to quickly take over another person's web applications because they need a system or an open pore flaw to do so. This blunder is classified as an accident. A flaw that allows hackers to gain entry and perform actions such as changing, deleting, or manipulating data. [2]

When creating a system, vulnerabilities are frequently created unintentionally. Vulnerabilities emerge as a consequence of bad design decisions made at some point during a system's lifespan. The system may contain flaws discovered and fixed during development and testing; however, these bugs are not considered weaknesses. For the manufacturing to be malicious and intentional, the identity and output must match. You can go back in time after identifying a vulnerability to see when it initially appeared.[3]

2.Literature Review

Mauro Conti et al., (2020) Because of their variety and heavy use of private

It is susceptible to a wide range of assaults, including XSS, SQL injection, adware, spam, and defacement. Only adware, defacement, and scam assaults are taken into account.

Various conventional techniques can be used to identify these weaknesses in online apps, but they are only effective in certain situations. We employ machine learning strategies to improve precision and decrease inaccuracy. [4]

We use the random forest algorithm to detect various types of attacks because there are numerous algorithms that can aid in the discovery of weaknesses, but we want the best one with the highest precision. We need a large number of examples before we can train a machine learning programme. It is a major problem because certain types of assaults are difficult to plan due to a dearth of a large enough database.[5]

Efforts are being made to spot adware, phishing, and defacement with 96.6% accuracy. Whenever a user clicks on an unsafe link, the software aids individuals and groups in spotting potential attacks.[6]

programming methods, web apps are challenging to evaluate. ML is especially helpful in the online world because it uses specifically data that has been marked for automatic analysis tools how users understand the meaning of a web application. They tried and evaluated the efficacy of MITCH, the first machine learning (ML) solution for black box detection of CSRF vulnerabilities, in order to substantiate this assertion. They postulate that other scholars will be able to discover various errors in online applications using their methodology [7]. **Noman, M.; Iqbal, M.; Manzoor, A. (2020)** performed a poll on online vulnerability detection and avoidance. They went into great depth about web weaknesses and the various defences. They did not examine any cutting-edge study on machine learning-based online attack detection, though[8].

Tuong Ngoc Nguyen et al., (2019) In order to identify website tampering, the study suggests a hybrid model based on attack signs and machine learning methods. The machine learning component can precisely recognise corrupted webpages and build a detection profile that makes use of both healthy and damaged pages. The processing of typical forgery assaults is sped up with the aid of the signature-based component. The damage detection model evaluated in experiments worked well on both static and dynamic webpages, with a total false positive rate of less than 0.62% and detection accuracy of more than 99.26%. Websites that are not in the language of the internet pages can be tracked using model training data. [9].

Alvise Rabitti, Alessio Ragazzo, and Michele Bugliesi (2019) Black box scanners, also referred to as online automated programmes called application vulnerability scanners scan online applications for security issues. Research was done on the vulnerability categories that these scanners look for, how well they work against targets of vulnerabilities, and how relevant target vulnerabilities are to actual vulnerabilities. It uses eight well- known methods to evaluate the state of the art. This study used a specifically created online application that is susceptible to known and suspected bugs in addition to earlier versions of well-known web apps with known vulnerabilities. These results draw attention to both some automated technology flaws as well as their general potential and benefits. Many tools currently do not identify SQL Injection "stored" versions or Cross Site Scripting (XSS). (SQLI) [10].

3. Methodology

3.1 System Architecture

The system's design offers a high-level perspective of its operation. The mechanism operates as follows:

The collecting of information, including possibly malicious or reliable URLs and webpages, is known as record collection. Using the feature extraction technique, we capture and isolate assaults, and then we further examine them to determine whether or not they are real.

The trained dataset is then completed by deciding whether the found harmful urls are truly risky before the input urls are added. In this case, it might be defacement,

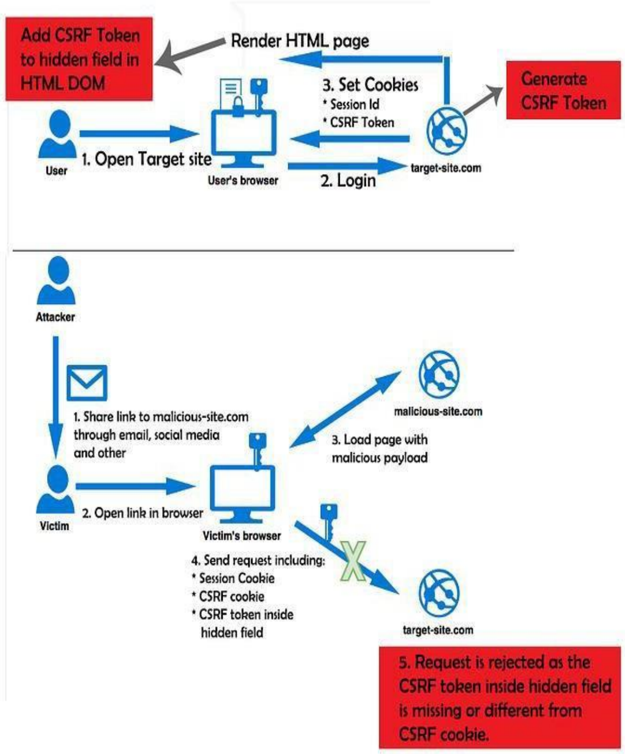

Add CSRF Token to hidden field in HTML DOM
Render HTML page
3. Set Cookies
* Session Id
* CSRF Token
Generate CSRF Token
User
1. Open Target site
User's browser
2. Login
target-site.com
Attacker
1. Share link to malicious-site.com through email, social media and other
malicious-site.com
3. Load page with malicious payload
Victim
2. Open link in browser
Victim's browser
4. Send request including:
* Session Cookie
* CSRF cookie
* CSRF token inside hidden field
target-site.com
5. Request is rejected as the CSRF token inside hidden field is missing or different from CSRF cookie.

Fig2. CSRF Process

deception, or malware. If the website is not malicious, we display the warning window with the type of attack present and let the page load normally after detecting them with an algorithm.

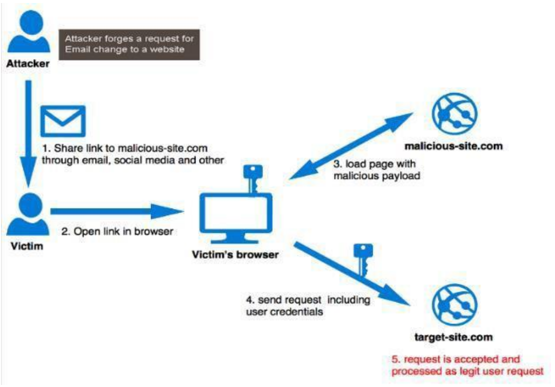

Attacker
Attacker forges a request for Email change to a website
1. Share link to malicious-site.com through email, social media and other
Victim
2. Open link in browser
Victim's browser
3. load page with malicious payload
malicious-site.com
4. send request including user credentials
target-site.com
5. request is accepted and processed as legit user request

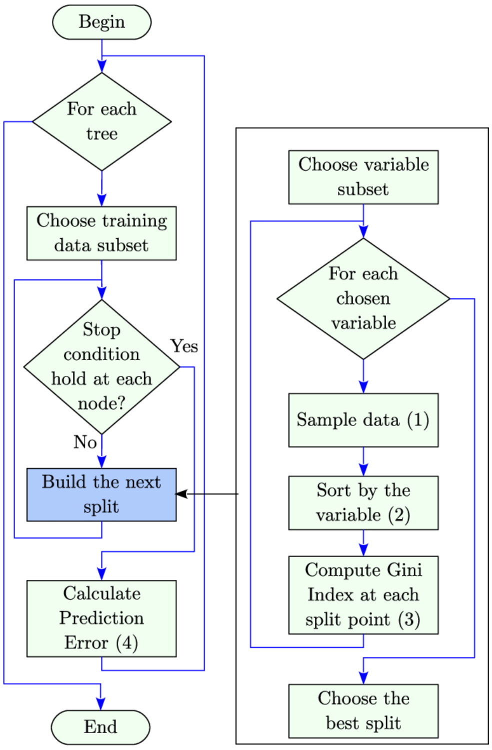
Fig1. CSRF attack

3.2 Algorithm

Random Forest

Machine learning methods like random forest are used to address categorization and error issues. It employs ensemble learning, a method that blends various algorithms to offer answers to challenging issues.

Random forest is a method that employs multiple decision trees. The random forest method's "forest" is discovered through tagging or bootstrap aggregation. The group meta-algorithm of bagging increases the precision of machine learning methods.



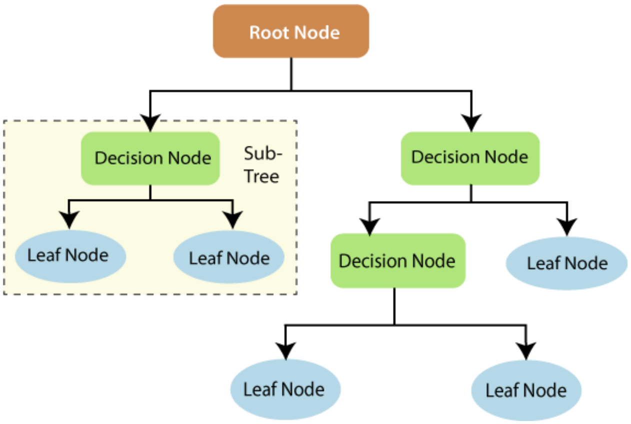


Fig3.Random Forest

Results

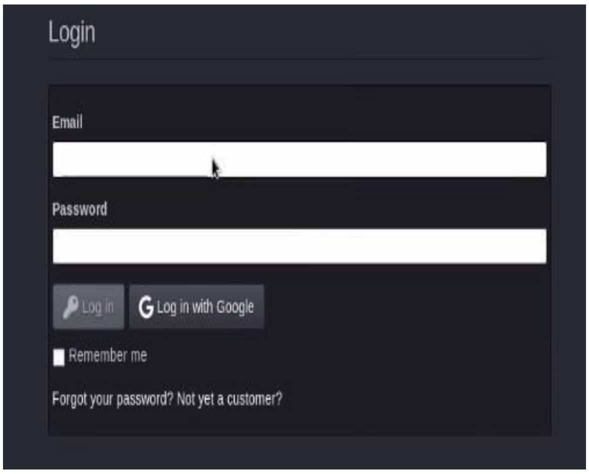


Fig4.Login

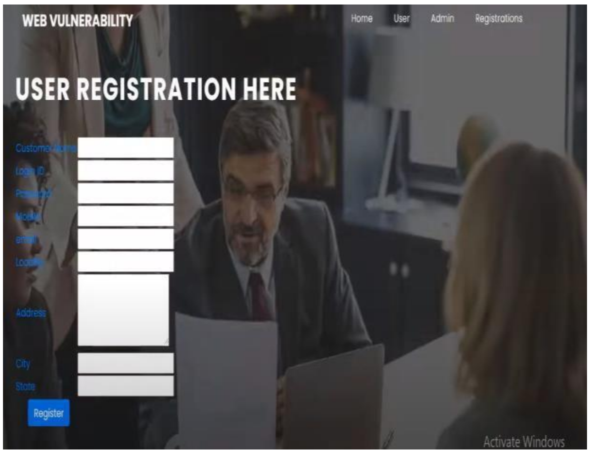
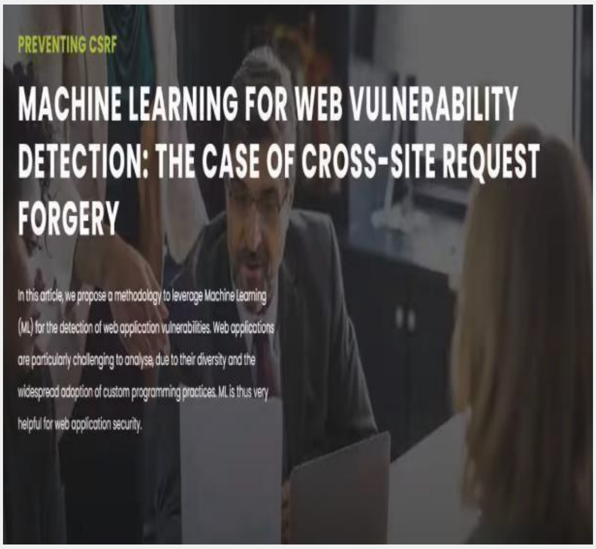


Fig6.User registration


PREVENTING CSRF
MACHINE LEARNING FOR WEB VULNERABILITY DETECTION: THE CASE OF CROSS-SITE REQUEST FORGERY

In this article, we propose a methodology to leverage Machine Learning (ML) for the detection of web application vulnerabilities. Web applications are particularly challenging to analyse, due to their diversity and the widespread adoption of custom programming practices. ML is thus very helpful for web application security.

Fig5. Home page

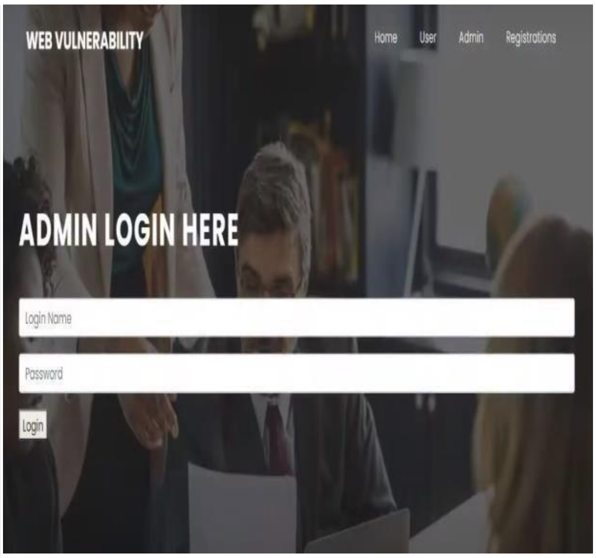


Fig7.Admin login

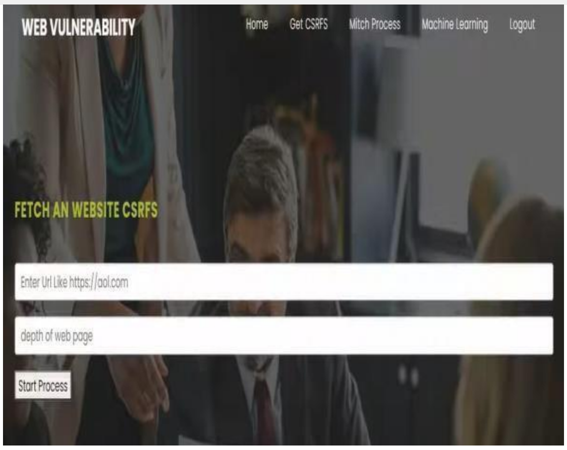
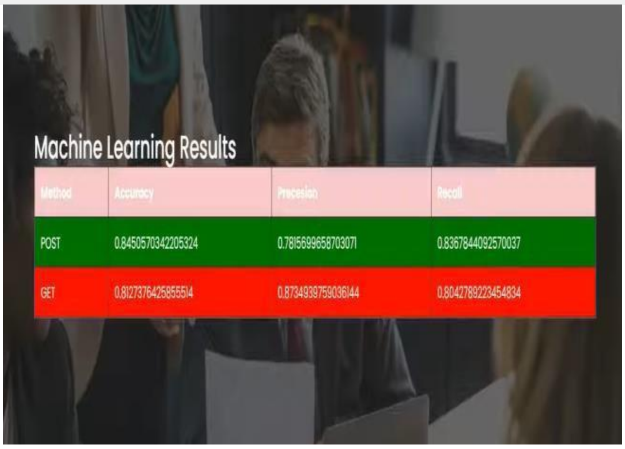


Fig8.Fetching an website


Machine Learning Results

Method	Accuracy	Precesion	Recall
POST	0.8450570342205324	0.7815699658703071	0.8367844092570037
GET	0.8127376425855514	0.8734939759036144	0.8042789223454834

Fig9.ML results

Conclusion

Web apps are difficult to evaluate due to their diversity and substantial usage of adaptive programming techniques. Because it uses carefully labelled data to show automatic analysis tools that help people comprehend the significance of web applications, machine learning is very helpful in the web world. By creating Mitch, the first machine learning (ML) solution for arbitrarily finding CSRF mistakes in forests, and testing and analysing its efficacy, we confirmed this claim. We think that by using our approach, other specialists will be able to recognise different web programme vulnerabilities.

Future work

This study presents the most recent and thorough analysis of the causes and countermeasures for CSRF attacks. However, as the groundwork laid in this effort continues to bear fruit, more developments in this area are inevitable. Here are some identified future areas for research: Since a probability ratio of 1% or less of a suspicious CSRF page to a safe page might be either random or unanticipated variance, it may be conceivable to execute the Bayesian estimate using this ratio in a future development of our work. Commercial anti-malware that includes regular CSRF scanning; anti-CSRF solutions that are browser-specific and platform-agnostic.

References

[1] Hoang, X. D. (2018, December). A website defacement detection method based on machine learning techniques. In Proceedings of the Ninth International Symposium on Information and Communication Technology (pp. 443-448).

[2] Hoang, X. D., & Nguyen, N. T. (2019). Detecting website defacements based on machine learning techniques and attack signatures. Computers, 8(2), 35.

[3] Sahingoz, O. K., Buber, E., Demir, O., & Diri, B. (2019). Machine learning based phishing detection from URLs. Expert Systems with Applications, 117, 345-357.

[4] Jain, A. K., & Gupta, B. B. (2018).
Towards detection of phishing websites on client-side using machine learning based approach. Telecommunication Systems, 68(4), 687-700.

[5] Althubiti, S., Yuan, X., & Esterline, A. (2017). Analyzing HTTP requests for web intrusion detection.

[8] Mereani, F. A., & Howe, J. M. (2018, February). Detecting cross-site scripting attacks using machine learning. In International conference on advanced machine learning technologies and Noman, M.; Iqbal, M.; Manzoor, A. A Survey on Detection and Prevention of Web Vulnerabilities. Int. J. Adv. Comput. Sci.
Appl. 2020, 11, 521–540

[9] Nguyễn Trọng Hưng , Dau Hoang
.”Detecting Website Defacement Attacks using Web-page Text and Image Features”. August 2021International Journal of
applications (pp. 200-210). Springer, Cham. Advanced Computer Science and Application

[6] Stefano Calzavara∗, Mauro Conti, Riccardo Focardi∗, Alvise Rabitti∗, and Gabriele Tolomei Machine Learning for Web Vulnerability Detection: The Case of Cross-Site Request Forgery, Article in IEEE Security and Privacy Magazine · January 2020

[10] Alvise Rabitti, Alessio Ragazzo, and Michele Bugliesi are Stefano Calzavara, Alvise Rabitti, Alessio Ragazzo, and Michele Bugliesi. Web sessions are being tested for integrity problems. ESORICS 2019, Luxembourg, Luxembourg,
September 23–27, 2019, pages 606–624, in Computer Security - 24th European Symposium on Research in Computer Security