iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 Issue: I Month of publication: January 2023

DOI: https://doi.org/10.22214/ijraset.2023.48469

Web Application Vulnerability Detection Using Hybrid Algorithm

Ujjwal Singh[1], Sukanya Sarkar[2], Shubham Kumar[3], Vivek Kumar[4], Mrs.B. Saranya[5]
[1, 2, 3, 4, 5]Department of Information Science and Engineering New Horizon College of Engineering Bangalore, India

Abstract: One of the greatest hazards to websites and web portals of private and public entities has been website attacks. In today's digital world, web applications play a significant role in daily life, making their security a difficult challenge. Through the URL links that are delivered to the victims, the attackers hope to obtain private information about the users. We are attempting to plug the gaps left by conventional means to combat the assaults, but these conventional measures are ineffective since attackers are getting better at targeting online apps. Currently, people are looking for software that can consistently and reliably identify web application attacks. Using machine learning, this approach seeks to protect online applications from flaws and many sorts of attacks.

Keywords: URL, Attacks, Extract, Web portals, Vulnerabilities, Random Forest Model.

I. INTRODUCTION

The most crucial component is now web applications since they make life simpler and less complicated. Due to its significant impact on daily life, it is inevitable that it will experience certain negative effects. It also follows that, given the volume of traffic it processes, including banking defence transactions and cyberbullying, it will draw a lot of attention from outside parties and hackers.[1]

Attackers can't just take over other people's web applications; they need an open port or a weakness in the system that will allow them to do so. This weakness is known as a vulnerability, which is nothing more than a weakness that allows hackers to gain access and carry out tasks like manipulating, erasing, or changing data. [2]

Inadvertent vulnerabilities are frequently produced when developing a system. Inadequate design choices made during one of the phases of the system life cycle lead to vulnerabilities. Only problems that are embedded in the way the system works are considered vulnerabilities; bugs that are discovered and addressed during the development and testing phases are not. The discovery and creation are same if the creation is malignant and therefore deliberate. The moment at which a vulnerability was established can be determined retroactively after it has been found. [3]

This leaves the door open for several assaults, including phishing, defacement, malware, SQL injection, XSS attacks, and many more. Here, we will focus on only three of these attacks: phishing, defacement, and malware. There are many established methods for finding this kind of vulnerability in web applications, but they are only effective for a limited number of tasks. As a result, there was a need for more effective methods to find vulnerabilities, and by using machine learning techniques, we can improve accuracy while reducing false positives. [4]

There are other methods that can be used to find vulnerabilities, but we need the one that is most accurate, so we are utilizing the Random Forest technique to find various attack types. We require enough datasets to train the model and evaluate it because we are utilizing machine learning. As a result, it becomes a significant issue because there aren't enough datasets available to train certain types of attacks.[5]

The project is meant to identify threats such as phishing, malware, and defacement. And is used by 96.6% accuracy. This initiative will assist both individuals and companies in identifying assaults that may occur when a user clicks on a malicious link. [6]

The remainder of the essay is structured as a thorough literature review in Section II. In Section III, the choice of system tools and problem identifications are covered. In Section IV, the system architecture and specific system design steps are covered. Future improvement is discussed as the paper's conclusion.

II. LITERATURE SURVEY

1) Dau Hoang et al., (2018) This research suggests a machine learning-based solution for detecting website defacement. His method involves automatically learning a detection profile from a training dataset of both healthy and damaged web pages. Experimental findings demonstrate the high detection accuracy and low false positive rate of our approach.

His approach is also suitable for building an online monitoring and detection system for website defacement because it doesn't call for a lot of computational power

2) Sara Althubiti et al., (2017) For the objective of intrusion detection, multiple machine learning algorithms have been used in this article using the CSIC 2010 HTTP dataset. Attacks including SQL injection, buffer overflow, data mining, file disclosure, and others were included in the dataset. The results of the experiments indicate that every technique, with the exception of Nave Bayes, has poorer precision, recall, and F1 rates and higher FPR when compared to the other techniques. In order to achieve better outcomes, high accuracy, and shorter training times, (Nguyen et al., 2011) isolated nine features deemed crucial for the identification process and employed the top five as determined by Weka.
3) Mauro Conti et al., (2020) Due to their diversity and the extensive usage of proprietary programming techniques, web apps provide unique analysis challenges. As a result, ML is highly helpful in the web environment since it may employ manually labelled data to expose automatic analytic tools to human comprehension of the semantics of a web application. By creating Mitch, the first machine learning (ML) solution for Blackbox detection of CSRF vulnerabilities, and empirically assessing its efficacy, they verified this assertion. They hope that future researchers will be able to find different types of web application vulnerabilities using their methods
4) Banu Diri et al., (2019) The author constructed a phishing detection system in this research employing seven distinct machine learning algorithms, including Decision Tree, Adaboost, K-star, KNN, Random Forest, SMO, and Naive Bayes, as well as numerous types of features, including NLP, word vectors, and hybrid elements. The main aim is to create a useful feature list in order to improve the detecting system's accuracy. He divided his list of features into two categories, word vectors that concentrate on using words in URLs without doing any additional processes, and NLP-based features, which are primarily human-defined characteristics.
5) Tuong Ngoc Nguyen et al., (2019) A hybrid website defacement detection methodology based on machine learning methods and attack signatures was suggested in the article. With a high degree of accuracy, the machine-learning component can identify damaged web pages, and the detection profile can be trained using a dataset of both healthy and damaged sites. The signature-based component aided in accelerating the analysis of frequent spoofing attack types. According to the experimental findings, the damage detection model can function well on both static and dynamic websites, and it has a detection accuracy of more than 99.26% overall and a false positive rate of less than 0.62%. The model can also track websites in languages other than the one of the websites used for the training data.
6) Truong Son Pham et al., (2016) In this study, we compared various machine learning methods for detecting web intrusions. The CISC 2010 HTTP dataset, which contains attacks including SQL injection, buffer overflow, information gathering, file sharing, CRLF injection, XSS, server side inclusion, parameter forgery, and more, was utilised in the trials. Logistic regression is the most effective learning strategy for this issue, according to experiments. The best performance is shown by logistic regression, which has the highest recall and precision. Using various feature extraction, feature selection, and parameter rotation strategies, we have also attempted to enhance its performance. After that, the outcomes appeared improved.
7) Jacob Howe et al., (2018) When applied to a greater number of data files, SVM, k-NN, and Random Forest can be utilised to develop classifiers for XSS implemented in JavaScript that yield high accuracy (up to 99.75%) and accuracy (up to 99.88%). file of data. This demonstrates that these classifiers can be included as an extra security layer in the server or (as intended) in the browser. The training data was created to offer a balanced representation of scripts, including both hidden and unobscured scripts as well as scripts of various lengths. Data is classified as harmful or benign rather than employing obfuscation as a stand-in for maliciousness. Other classification techniques were anticipated to perform well in addition to the SVM, k-NN, and Random Forest that were utilised in the studies. It is obvious that the Random Forest architecture is popular when looking at the several frameworks already in use.

III. SYSTEM DESIGN

A. *Problem Analysis*

1) The traditional method of identifying dangerous websites is known as the "Blacklist approach," and it involves finding and adding any questionable or harmful URLs, or IPs (Internet Protocols), to the database of websites.
2) Attackers trick consumers in a variety of methods to avoid being blacklisted, including by changing URLs to make them seem true and authentic, obfuscation, and a range of other fundamental techniques like the fast- flow. In this approach, proxies are generated automatically to host the webpages; another approach uses an algorithm to generate URLs; and so on.

3) A heuristic-based detection, it identifies attacks based on the traits and features found in phishing attacks; it can also be used to identify zero hour attacks that the Blacklist method misses; however, it cannot be guaranteed that these traits are always present in the attack, and the positive identification rate for bogus and absurd claims is high.
4) To overcome the drawbacks Artificial intelligence techniques are currently the focus of security experts. AI calculations require historical data to make decisions or build expectations for future data.

B. Dataset

We downloaded the dataset, which consists of 651191 URLs, from Kaggle. The dataset we used has a total of 651191 websites, including all legitimate, phishing, malware, and defacement websites. Of the 428103 safe URLs, 96457 defacement URLs, 94111 phishing URLs, 32520 malware URLs, and over all the websites, 428103 are safe URLs.

IV. METHODOLOGY

A. System Architecture

An overview of the system's operation is provided by the system architecture. Here is how the system functions: Dataset collecting is the gathering of information, including URLs and websites that may be harmful or trustworthy. We separate and extract the attacks through the feature extraction method, then we further process them to determine whether or not they are valid.

First, we put the trained dataset with the input URLs, which is followed by determining whether the given URL is malicious or not. If it is, we intercept it with an algorithm and a hybrid approach and display a warning dialogue with the type of attack that is present. If it is not a malicious website, we then display the dialogue and load the page normally.

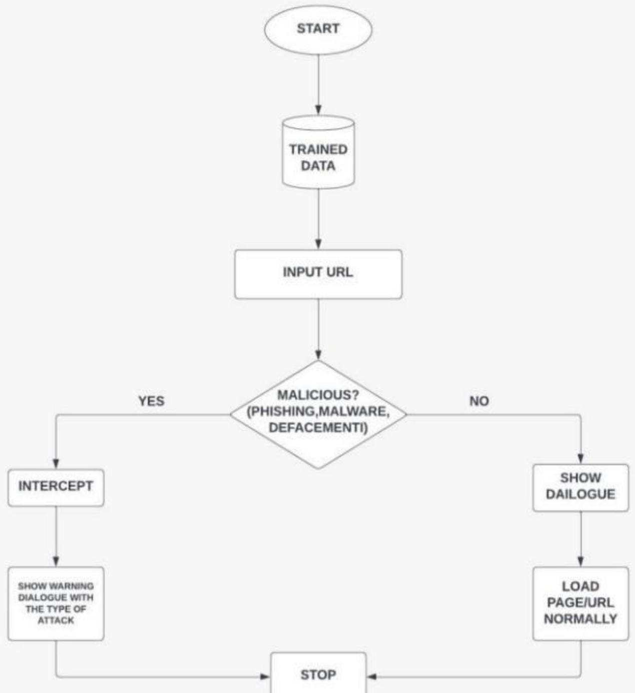

START
TRAINED DATA
INPUT URL
YES
MALICIOUS? (PHISHING,MALWARE, DEFACEMENTI)
NO
INTERCEPT
SHOW DAILOGUE
SHOW WARNING DIALOGUE WITH THE TYPE OF ATTACK
LOAD PAGE/URL NORMALLY
STOP

Fig. 1. Shows the system architecture of proposed model

B. Supervised Machine Learning

The definition of supervised learning in its most basic form is discovering how an algorithm converts an input into a specific output. If the mapping is accurate, the algorithm has likely been trained properly. If not, modify the algorithm as needed so that it can be correctly taught. Algorithms for supervised learning can forecast future reception of invisible data.

Several business applications are developed and improved using the supervised learning methodology, including:

1) *Predictive Analytics:* Predictive analytics is currently seeing exponential growth due to the expansion of use cases in the bitcoin and equity trading industries. This enables firms to validate particular outcomes depending on particular output characteristics, assisting executives in defending choices that are advantageous to the organization.

2) *Analyzing Customer Sentiments:* Businesses may learn crucial information such as context, emotion, and intent using supervised machine learning algorithms. You can optimize the growth of your brand and your customer interactions thanks to this data.
3) *Detecting Spam:* Businesses may learn crucial information such as context, emotion, and intent using supervised machine learning algorithms. You can optimize the growth of your brand and your customer interactions thanks to this data.

C. Random Forest Model

A supervised machine learning approach called random forest is non-parametric (it makes no assumptions about the probability distribution of the data points). This is a development of the machine learning classifier that uses bagging to boost the effectiveness of the decision tree. The tree is based on an independently selected random vector and mixes tree predictors. As it works to lessen variance and creates a "average" decision rule from a collection (the forest) of numerous individual decision trees, it falls under the umbrella of ensemble methods. These trees are designed so that when a new data point is predicted by one portion of the forest, it should match the average rules generated by the other parts of the forest.

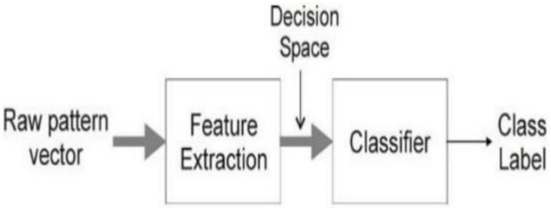
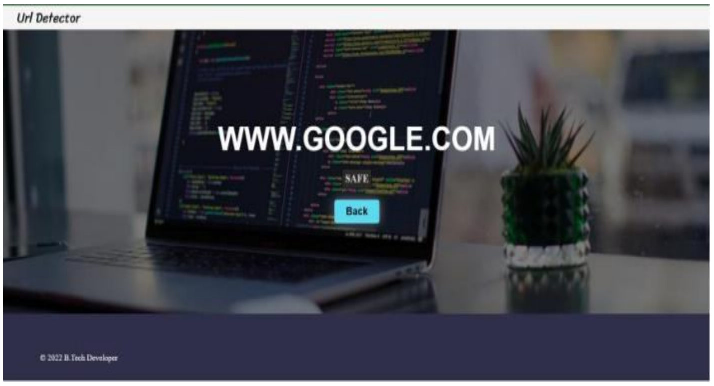


Fig 2. Random Forest architecture

1) The concept is that utilizing complicated parametric models is frequently unnecessary or undesirable if you can achieve good results with much simpler non-parametric models. In an unsupervised approach, the decision border between classes is the most basic classifier rule. This would require us to use a single divider line to categorize all of the data points (hyper plane). The issue is that non-linear issues do not respond well to this straightforward approach. In reality, even numerous hyper planes cannot always be identified accurately and without errors using this type of dataset.
2) The random forest algorithm is useful in this situation. It generates a collection of several decision trees, each of which provides an average categorization rule using a separate set of rules for every data point. Additionally, it can continue to be accurate even when a lot of information is absent. Compared to other algorithms, it takes a specific level of investment and anticipates the output of massive data sets when they are used successfully.

V. RESULTS AND DISCUSSIONS

Attack Detection

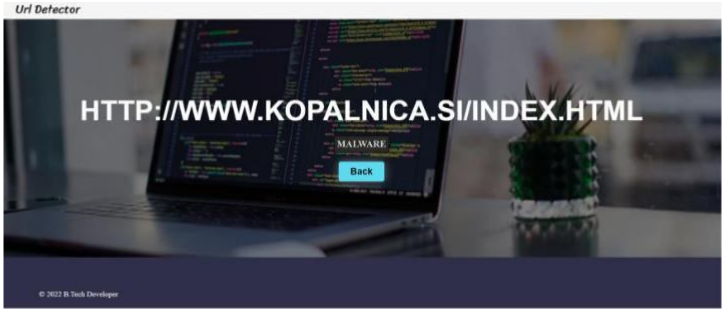

Url Detector

WWW.GOOGLE.COM

SAFE

Fig 3. Safe Website

Fig. 3 shows that This URL does not contain any malicious attacks and is safe to browse.

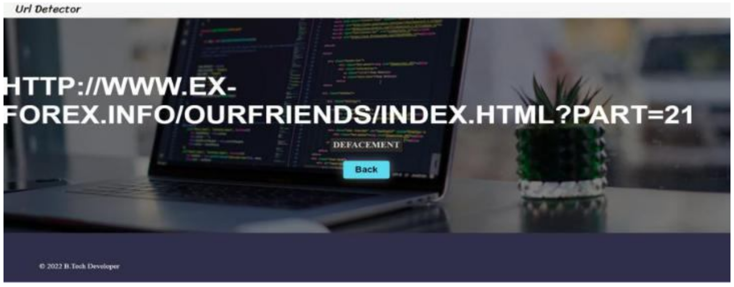


HTTP://WWW.KOPALNICA.SI/INDEX.HTML

MALWARE

Fig 4. Detecting website with malware attack

Fig 4. shows a malicious website which consists of malware attack.

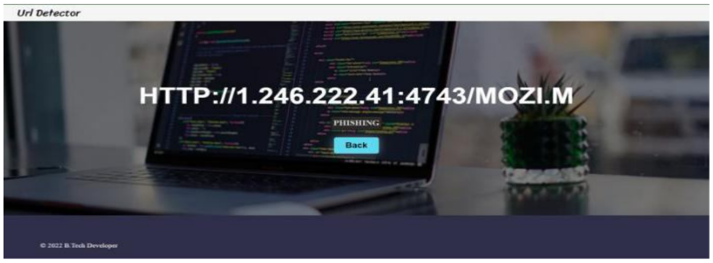


HTTP://WWW.EX-FOREX.INFO/OURFRIENDS/INDEX.HTML?PART=21

DEFACEMENT

Fig. 5. Detecting website with defacement attack

Fig.5. shows a malicious website which consists of defacement attack.



HTTP://1.246.222.41:4743/MOZI.M

PHISHING

Fig 6. Detecting website with phishing attack

Fig 6. shows a Phishing website which consists of Phishing attack.

VI. CHALLENGES

The use of massive datasets continues to be the main problem with the proposed model. We used a dataset including over 65 1191 URLs, both safe and dangerous, to evaluate machine learning algorithms. Each of which has 21 features. Every URL follows a standard. It will be regarded as a malicious URL if the criteria are satisfied. If not, the URL will be regarded as authentic. Therefore, we discovered that the system performance suffers if SVM and logistic regression are used instead of Random Forest.

VII. CONCLUSION

The growth potential of web technology is at its highest in the modern internet age. The security of the website and defence against assaults such phishing malware defacement attacks are to be found and stopped. The model developed by this project contributes to the website's safety and security. This project is optimized to the point that we can distinguish between dangerous and non-malicious websites using datasets, feature extraction, and machine learning with a hybrid algorithm to identify the assault with precision. As we learn about it through machine learning, which is a technology that can have an impact across all domains, this model will provide a safe and secure way to access websites without worrying about the unfamiliarity and unpredictable behavior they possess. In the current digital age, this is one of the best tools for safe and secure internet platforms.

REFERENCES

[1] A Website Defacement Detection Method Based on machine Learning Techniques

[2] Detecting Website Defacement Based on Machine Learning Techniques and Atack Signatures Published by: Telecommunicati ons Institute of Technology, Hanoi, 2019.

[3] Machine Learning based phishing detection from URLs Published by: Marmara University, Turkey, 2018 Published by: Cyber Security Lab, Faculty of Information Technology, Hanoi, 2018

[4] Towards detection of phishing websites on client-side using machine learning based approach Published by: Ankit Kumar Jain1 2018

[5] Analyzing HTTP requests for web intrusion detection Published by: Albert Esterline North Carolina State University, 2017

[6] Detecting Cross-Site Scripting Attacks Using Machine Learning Published by: University of London, 2018

[7] Machine learning techniques for web intrusion detection – a comparison Published by: Le Quy Don technical University Faculty of information technology Hanoi, Vietnam, 2016

[8] Machine Learning for Web Vulnerability Detection: The Case of Cross-Site Request Forgery Published by: Sapienza University of Rome, Italy, 2020

[9] Machine Learning in Vulnerability Assessment Published by: Soana Networks, Canada, 2018

[10] Detection of phishing attacks in financial and e-banking websites using link and visual similarity relation. Published by: International Journal of Information and Computer Security, Inderscience, 2017