IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

MALICIOUS WEB PAGE DETECTION IN BROWSER BY MACHINE LEARNING TECHNIQUE

[1]Netra Suresh Gudagamnala, [2]AnilKumar B N, [3]Arun K, [4]Kanaka B S, [5]Veena M

[1]Student, [2] Student, [3] Student, [4] Student, [5] Assistant Professor

[1,2,3,4,5] Department of Computer Science and Engineering

[1,2,3,4,5] Alva's Institute of Engineering and Technology, Mijar, India

Abstract: Phishing is the most often employed social engineering and cyber-attack in the current circumstances. As a result of such attacks, attackers attack on naïve online users, duping them into revealing private information in order to utilize it fraudulently. In order to avoid phishing websites, users must be aware of phishing attacks or utilize a technology that detects phishing pages. Our aim is to build a browser extension that acts as a gateway between users and malicious websites. This extension is been developed using a machine learning technique, and we used the random forest, which provides 99.8%. The tool will be trained on both static and dynamic web page characteristics. If a web page includes hidden malicious content or malware, our model will recognize it and display a popup notice to naïve users otherwise load the legitimate web pages.

Index Terms **– Static and Dynamic Features.**

I. INTRODUCTION

The emergence of smart communication technologies has had a significant impact on the growth and promotion of businesses in a variety of applications such as online banking, e-commerce, and social networking. As a result, the World Wide Web's importance has risen significantly and internet has become an essential part of people's lives. However, importance on Internet offers prosperity, it is also causing difficulties like illegal websites, fraudulent medical websites, pornographic, gambling, etc. Despite the use of various detection techniques, the number of malicious websites continues to grow. The large amount of malicious information on the Internet is hazardous to the health of Internet users, particularly children and teenagers [1].

As per our review Malicious web sites, are those that contain content that can be utilized by attackers to exploit end-users. This includes the following malicious contents like phishing URLs, spam URLs, and JavaScript on web pages Malware scripts, spyware, and a variety of other threats. Due to the constant development of new strategies for carrying out such attacks, it is becoming increasingly difficult to detect such vulnerabilities. Researches have investigated a variety of approaches for detecting fraudulent websites, including heuristic methods, machine learning-based methods and etc., here checking the entered URL against a list of websites that have been labelled harmful by a trusted provider. However, the disadvantage of this strategy is that the list is non-exhaustive, indicating that it grows every day.

Furthermore, with such a vast list, the system's latency time will always increase, causing the user to feel frustrated. As a result, in our project, we applied a machine learning approach to determine whether a webpage is malicious or legitimate based on static and dynamic features. URLs and website content can be evaluated with a trained classifier. the trained model embedded with a Google Chrome extension Because it is a very good way of ensuring easy access and user-friendly tool and it is most extensively used web browser on the planet, and it comes with its own set of features. Our goal is to ensure safe browsing of the website which the user wishes to visit. Even if the user decides to visit a phishing website unknowingly, our chrome extension will arise a popup and measures will be taken to protect the user from being harmed [2].

II. LITERATURE SURVEY

The survey deals with a technique to detect fake websites. According to the paper, a number of approaches are available to detect fake web pages, including:

A novel classification mechanism was proposed by Seifert et al [2] to detect malicious web pages. This method is based on analyzing HTTP responses from potentially malicious web servers for malicious features that can be retrieved Y.-T.Hou et al [3]. The method was employed in a hybrid system in which all URLs are identified using a static heuristic and routed to a high interaction client honeypot for verification. For identifying URLs by static heuristics technique, several common features are chosen based on three proposed primary elements of malicious web pages: exploit, exploit delivery mechanism, and obfuscation.

Weka was used to implement the J4.8 decision tree. There was a very good false positive rate for this classifier (5.88 percent), but a very high false negative rate (46.15 %).

J. Ma et al [6] have identified a new approach, called lightweight URL classification, to detect malicious web pages. In this approach, they classify web pages based on lexical and host-based features based on the relationship between URLs. It does not use web page content in the detection process. For classification, Naive Bayes, SVM and Logistic Regression are used. In their study, the authors have done some experiments. The first experiment was to use 1-regularized logistic regression (LR) classifiers to compare feature sets. The results indicated that better classification accuracy was obtained by using more features.

In the paper, Ahmad Abunadi et al [4] demonstrated that a new set of three features can be used to prevent suspicious web content. The first feature is Google Page Rank results. Google PageRank is a website popularity evaluation service provided by Google. Phishing site won't have a high PageRank since most phishing sites are on the web for a restricted time period. A large portion of the phishing sites will have a PageRank in the middle of 0 to 10. The second feature is Google Position and also, it makes use of google search engine. This feature gives a very clear evidence of detecting if a website is phishing because Google search engine will not archive a phishing website in search index. The third component is the Alexa rank. The positioning calculation of Alexa depends on the number of individuals frequenting the site. Certainly, none of the phishing sites will be visited by many individuals every now and again. Therefore, it may very well be detected malignant site if any site was in the middle of 0 or 10.

Based on the literature survey we have been collected 22 set of features, categorized as static and dynamic features and used random forest algorithm for the better accuracy.

III. METHODOLOGY

The operation of two-stage classification model can briefly be explained as follows in Fig. 1:

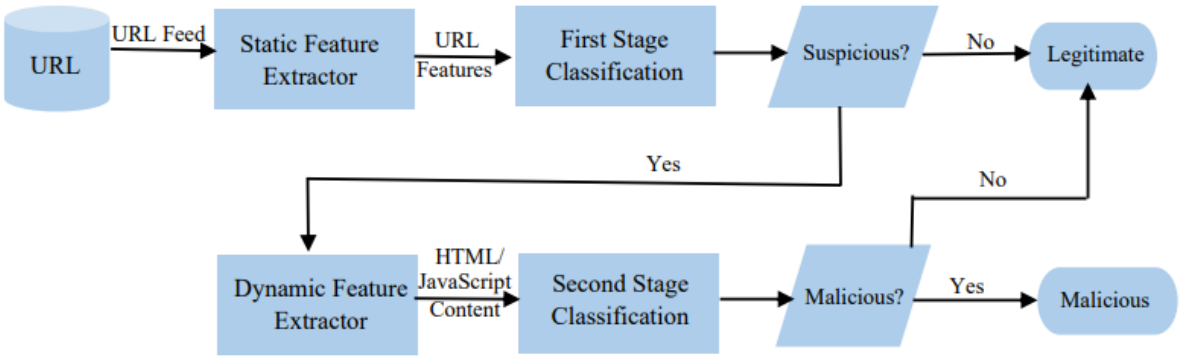

Fig. 1. Operational work flow of two stage Classification model

In the model's operational flow, we must first enter the URL in the address bar, which is then inspected and delivered to the static feature extractor. The static feature extractor extracts some of the potential static features from the input URL, such as the length of the URL, the age of the domain, the count of special characters in the URL, and so on. The retrieved static features are used in the first stage classification to determine whether a given input URL is suspect or legitimate; if it is suspicious, the web page is subjected to a second stage of feature extraction. In Dynamic feature extraction which extract the features such as hidden content in iframe tag, redirection URL in anchor tag or some malicious scripts. The second classification uses the extracted dynamic features and estimates webpage as malicious or legitimate [7].

IV. IMPLEMENTATION AND RESULT

The proposed approach integrates both static and dynamic features, and it is trained Choosing the optimal random forest technique among a variety of algorithms including SVM, logistic regression, and KNN.

i. Obtaining Datasets

A standard UCI-Machine Learning Repository and the phishtank repository were used to retrieve the dataset. The dataset contains both malicious and benign URLs. This dataset contains 450176 entities, with 30 features based on static and dynamic features for each URL.

ii. Feature Selection

Based on the URL, we extracted 20 features out of 30 from the dataset. Some of the followings are:

- Static Feature Selection – Which includes address bar-based features such as Length of the URL, Count of Redirection "//", URL having IP address, http/https in the domain name, DNS record, Age of website, Website traffic and etc.,
- Dynamic Feature selection – Which includes HTML and JavaScript based features like iframe redirection, right click disabling, link in anchor tag and etc.,

iii. Model Training

In our project, we used the best random forest algorithm among other algorithms such as SVM, logistic regression, and KNN to train our model. The accuracy of Random Forest is 99.8 percent, which is superior to the other algorithms.

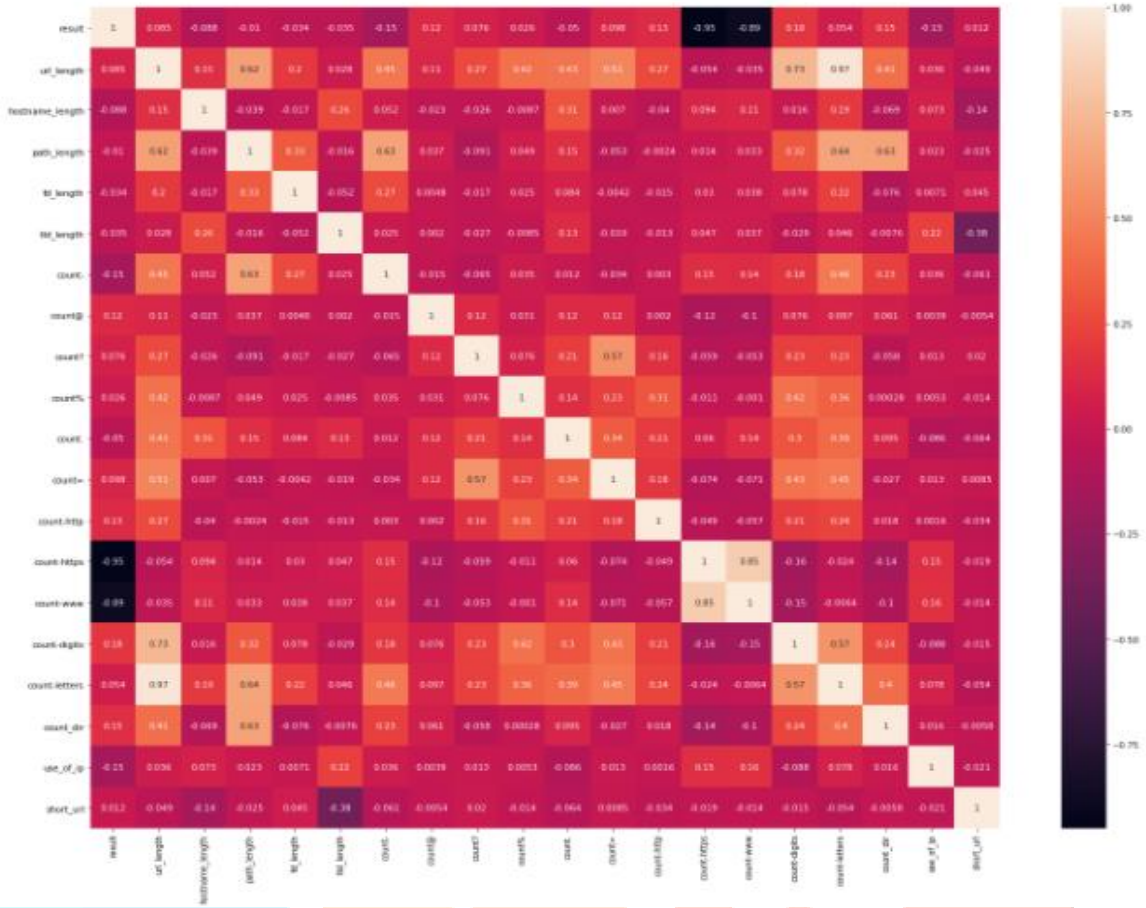

Fig. 2. Feature distribution graphical representation

Fig. 2. shows a graphical representation of how each feature was allocated during model training. We have chosen classification algorithm as SVM, KNN, logistic regression and random forest among which random forest gives better accuracy.

	precision	recall	f1-score	support
benign	0.998	0.999	0.999	241952
malicious	0.998	0.994	0.996	73172
accuracy			0.998	315124
macro avg	0.998	0.997	0.997	315124
weighted avg	0.998	0.998	0.998	315124

Fig. 3. Accuracy Score of Random Forest algorithm

We examine not just the accuracy of these methods while comparing them, but also additional parameters that are utilized to choose the best classifier. For each algorithm, we calculated the accuracy, precision, recall, and F1-Score for our dataset. We received the best scores when we employed the Random Forest algorithm, according to the results of these measures. As a result, we choose to use the Random Forest Algorithm to train the classifier. The screenshot in Fig. 3. shows the results of the algorithms together with the various performance measures [2].

iv. Browser Extension

Extensions are browser add-ons that help users get more out of their browser by introducing new functionalities and making it easier to use. A Google Chrome Extension is built using HTML, JavaScript, CSS, and JSON. A trained model is integrated with an extension in our project. When the loaded web page is malicious, the extension displays a popup [6]

v. Result

When we enter a URL in the Address bar, the static feature extractor will use python code to extract some of the features from the URL, and the result will be either 1 or -1, which will be kept in the array format. Then we put these features to the test on a random forest trained classifier.

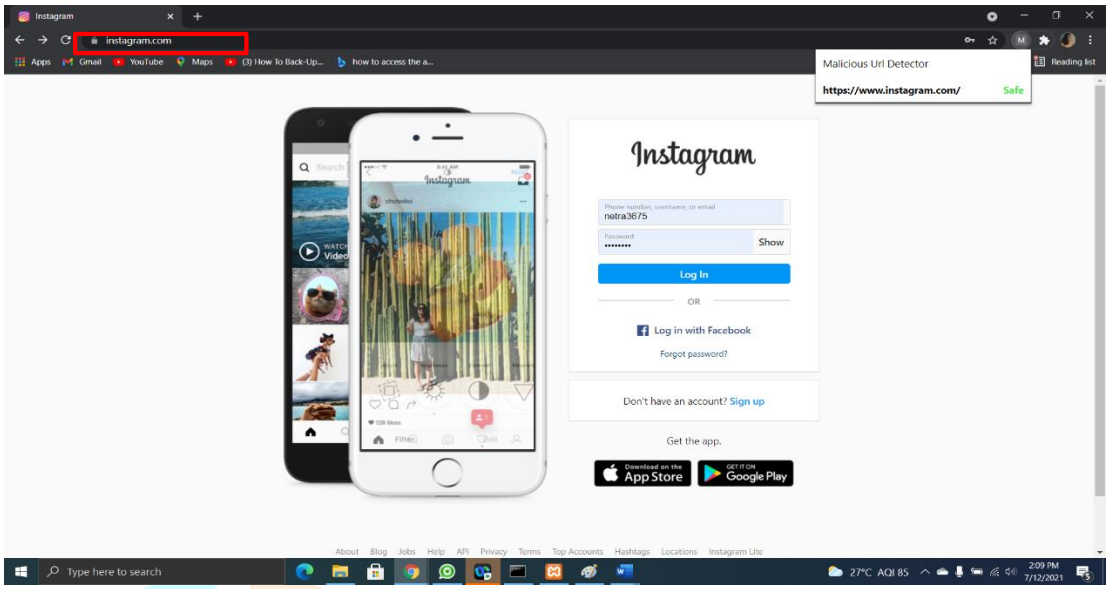

Fig. 4. Extension's result for a secure website

Fig. 4 is a resultant screenshot of legitimate website that is https://www.instagram.com because this web page is satisfy all the static features are safe so when we click on google chrome extension which will rise the popup as safe. As an example, it will display the safe for all legitimate websites.

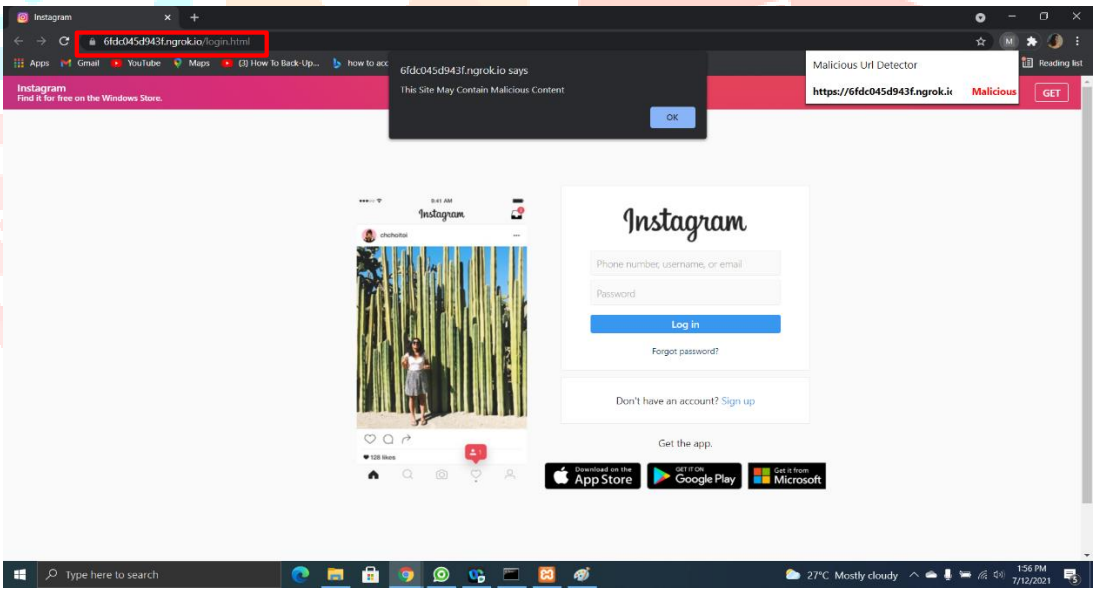

Fig. 5. Extension's result for a malicious website

Fig. 5 is a resultant screenshot of an Instagram malicious page or phishing page because this web page satisfies some of the suspicious static features and undergoes further classification in which web page satisfies that it includes hidden malicious script and automatically arises the popup that this page includes some malicious content. Our add-on will prevent online users from disclosing personal information to hackers in this manner.

V. CONCLUSION

This paper proposes the development of a malicious web page detection system that accepts URL as input. If the page is malicious, our model will display a popup window; otherwise, the browser will load legitimate websites. The random forest algorithm is used to classify the extracted features from the URL in this case. The Chrome Extension will keep the user's confidential credentials safe.

VI. REFERENCES

[1] D. R. Patil, J. B. Patil, "Survey on Malicious Web Pages Detection Techniques, Science and Technology", 2015 International Journal of u and e- Service.

[2] Seifert, Christian; Welch, Ian; Komisarczuk, Peter." Identification of malicious web pages with static heuristics" Telecommunication Networks and Applications Conference, 2008. ATNAC 2008. Australasian. 2008. p. 91-96.

[3] Y.-T. Hou, Y. Chang, T. Chen, C.-S. Laih and C.-M. Chen. "Malicious web content detection by machine learning", Expert Systems with Applications, In Press, Corrected Proof (2009).

[4] Ahmad Abunadi, Oluwatobi Akanbi, Anazida Zainal. "Feature extraction process: A Phishhing detection Approach". 2013 13th International Conference on Intelligent Systems Design and Applications (ISDA).

[5] Rajesh Kumar, Xiaosong Zhang, Hussain Ahmad Tariq, Riaz Ullah Khan, "Malicious Url Detection Using Multi-Layer Filtering Model", 2017.

[6] J. Ma, L. K. Saul, S. Savage and G. M. Voelker, "Beyond blacklists: learning to detect malicious web sites from suspicious URLs", Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, Paris, France, 2009.

[7] Anand desai, janavi jathakia, "Malicious Web Content Detection Using Machine learning", 2017 2nd IEEE International Conference on Recent Trends in Electronics Information & Communication Technology.