Malicious URL Detection to Avoid Web Crime Using Machine Learning Techniques

DR A RAJESWARI
MONIKA S
NIVETHA G
SANGEETHA DEVI G

Velammal Engineering College
Department of Computer Science and Engineering
Chennai, Tamil Nadu, India
rajivelit@gmail.com
monikahamsa23@gmail.com
nivethag52@gmail.com
sangeetha06122000@gmail.com

Abstract. Today the foremost necessary concern within the field of cyber security is finding the intense issues that create loss in secure data. In recent years, most offensive strategies are applied by spreading malicious and phishing URLs. An accidental visit to a malicious website will trigger pre-designed criminal activity. The phishing website has evolved as a serious cyber security threat in recent times. Phishing may be a type of online fraud wherever a spoofed website tries to gain access to user's sensitive data by tricking the user into believing that it's a benign website. ML algorithms are one of the effective techniques for malicious website detection. The proposed system is enforced with the assistance of Gradient Boosting classifier, it considers 27 major features of the URL to detect whether the URL is legitimate or malicious based on varied discriminative options and attributes of the address. The model can find whether the address is safe or unsafe. It is found that the accuracy rate of gradient boosting algorithm is 98% and the accuracy rate of other existing algorithm is 96% or 95% respectively. Comparatively the proposed system outstand the performance of the existing system.

Keywords: Uniform Resource Locator (URL), Machine Learning (ML), Gradient Boosting.

1 Introduction

URL is the international address of documents and alternative resources on the globe Wide Web Addresses will be either benign or malicious. Malicious websites are well-known threats in cyber security. A Malicious website could be a spoofed website that tries to achieve access to a user's sensitive info by tricking the user into a basic cognitive process that it's a benign website. Therefore it's become a priority for cyber defenders to notice and mitigate the unfolding of malicious codes. This project aims at using machine learning techniques to notice the malicious net links and classify them. Malicious URLs are generated daily and have a brief era. Hence dataset assortment tends to be tedious work. Real-time malware detection is still an enormous challenge. It's unendingly growing in terms of numbers and maliciousness. URLs could be a kind of massive information with immense volume and high speed. Therefore it's not possible to coach a malicious address detection model on all the address information.

A malicious website may at times look alike an awfully widespread website and lures the user to represent the entice. Phishers steal personal info and money account details like usernames and passwords. Thus, the user ought to understand whether or not the website is safe or phishing. The matter statement is to notice the malicious links and specify whether or not a given link is Safe or

malicious. There is square measure and many solutions to notice phishing attacks like educating users, using blacklists, or extracting phishing characteristics found to exist in phishing attacks. Blacklisting approaches maintain a listing of URLs and the square measure is known to be malicious. Whenever a brand new uniform resource locator is visited, an information search is performed. Blacklisting suffers from the lack to take care of a complete list of all doablemalicious URLs, as new URLs are simply generateddaily, therefore creating is not possible for them to noticenew threats. In Heuristicapproaches, common attacks in square measure are known, and a signature is appointed to the present attackformat, the commonly used phases in the process of datamining for extracting knowledge.

2 Related Work

2.1 Blacklist Approach and Whitelist Approach

In [1], Pawan Prakash, Manish Kumar, Ramana RaoKompella, Minaxi Gupta (2010) projected a prognostic blacklist approach to sight phishing websites. it's currently called new phishing URL exploitation by heuristics and by exploitation of appropriate matching algorithm. Heuristics created new URL's by combining parts of the noted phished websites from the gettable blacklist. The matching algorithm then calculates the score of universal resource locater .If this score is over a given threshold worth it flags this site as phishing website. The score was evaluated by matching varied parts of the universal resource locater against the universal resource locater gettable at intervals the blacklist. In [2], scientist Min Kang and DoHoon Lee depicted approach that detected phishing supported user's on-line activities. This system maintained a white list as a vicinity of users' profile. This profile was dynamically updated whenever a user visited any site. Associate degree engine used here identified an online website by evaluating a score and then comparing it with a threshold score. The score was calculated from the entries gettable at intervals, the user profile and details of this site.

2.2 Visual Similarity Approach

In [3], A. Mishra and B. B. Gupta gave a hybrid solution that supported URL and CSS matching. during this approach, it will notice embedded noise contents like a picture in a very web page that is employed to sustain the visual similarity within the webpage. They adopted the methodology employed in [4] by Jian Mao, Pei Li, Kun Li, Tao Wei, and Zhenkai Liang to examine the CSS likeness and used it in their approach. the various types of visual options are - text content and text options. Text options are font color, font size, background color, font family then forth. This approach matches the visual options of various websites since the hacker copies the page content from the particular website. In [5] Matthew Dunlop, Sir Leslie Stephen coin, and David Shelly planned a browser-based mostly introduced known as goldphish to identify phishing websites. It uses the website logos to identify the pretend website. The hacker will use the $64000 brand of the target website toentice the web users. 3 stages to it are:

Brand Extraction: Goldphish is employed to extract the website brand from the suspicious website. Then it converts it into text exploitation optical character recognition (OCR) software package.

Legitimate website extraction: The text obtained issued as a question for the program. Generally, the search engine "google" is employed as a result of the invariable genuine websites in their prime results.

Comparisons: Suspicious website is compared with the top result obtained from the program supported by different options. If any domain is matched with the current website then it's declared legitimate as an alternative make it phishing website

2.3 Signature based Phishing URL Detection

Studies on phishing link detection using the signature sets had been investigated and applied very long time in the past. Most of those studies usually use lists of proverbial malicious URLs. Whenever a brand link is accessed, a query is executed. If the URL is blacklisted, it's thought about as malicious, and then, a warning is generated; otherwise URLs are thought of as safe. the most disadvantage of this approach is that it'll be troublesome to notice new malicious URLs that don't seem to be within the given list.

2.4 Machine Learning Based Phishing URL Detection

There are unit 3 sorts of machine learning algorithms that can be applied to malicious URL detection, ways including supervised learning, unsupervised learning, and semi supervised learningand also the detection ways area unit supported URL behaviors. In [6], a variety of malicious uniform resource locator systems supported by machine learning algorithms are investigated. Those machine learning algorithms embody SVM, provision Regression, Nave Bayes, call Trees, Ensembles, online Learning. In this paper, the 3 algorithms such as Gradient Boosting, RF and SVM, are used. The behaviours and characteristics of URLs are often divided into 2 main teams, static and dynamic. In their studies [7,8,9] authors given ways of scrutinizing and extracting static behaviour of URLs, as well as Lexical, Content, Host, and Popularity-based. The machine learning algorithms employed in these studies are unit online Learning algorithms and SVM. Malicious uniform resource locator detection

victimization dynamic actions of URLs are presented in [10,11]. Uniform resource locator attributes are unit extracted supported, each static and dynamic behaviour.

2.4.1 Decision Tree

One of the most popularly used algorithms in machine learning technology. Decision tree algorithm is simple to understand and also very easy to implement. Decision tree begins to work by selecting the best splitter from the available attributes for classification which is considered to be the root of a tree. This algorithm continues to build a tree until it finds the leaf node. A decision tree creates a training model which can be used to predict the target value or class in the tree model each internal node belongs to an attribute and each leaf node belongs to the class label. In this decision tree algorithm, gini index and information gain methods are being used to calculate these nodes.

$$\text{Entropy}(S) = \sum_{i=1}^{c} - P_i, \log_2 P_i$$

$$\text{Information Gain}(T, X) = \text{Entropy}(T) - \text{Entropy}(T, X)$$

$$\text{Gini} = 1 - \sum_{i=1}^{c} (p_i)^2$$

2.4.2 Random Forest

The random forest algorithm is one of the most dominant algorithms in machine learning technology and it is based on concept of decision tree. Random forest algorithm constructs the forest with a number of decision trees. The more number of trees gives high detection accuracy. Creation of trees is based on the bootstrap method. In the bootstrap method features and samples of the dataset are randomly selected with replacements to construct a single tree. Among randomly selected features, a random forest algorithm will select the best splitter for the classification, and like the decision tree algorithm; the Random forest algorithm also uses gini index and information gain methods to identify the best splitter. This process will get to continue until the random forest creates n number of trees. Each tree in the forest predicts the target value and then the algorithm will calculate the votes for each of the predicted targets. Finally, the random forest algorithm considers the high voted predicted target as the final prediction

$$\text{MSE} = 1/N \sum_{i=1}^{N} (f_i - y_i)^2$$

MSE – Mean Squared Error

N – Number of data points

f_i – Value returned by the model

y_i – Real value for data point i

2.4.3 Support Vector Machine

Support vector machine is another influential algorithm in machine learning. In the support vector machine algorithm, each data item is plotted as a point in n-dimensional space, and the support vector machine algorithm constructs a partitioning line for the classification of two classes, this separating line is well known as a hyper plane. The support vector machine finds the closest points called as support vectors and once it finds the closest point it then draws a line connecting to them. Support vector machine then constructs partitioning line which bisects and perpendicular to the connecting line. To classify data perfectly the margin should be maximum. Here the margin is the distance between the hyper plane andsupport vectors. In real-time, it is not possible to separate complex and non-linear data, to solve this problem support vector machine uses the kernel trick which transforms lower-dimensional space into higher- dimensional space.

3 Proposed Model

In this paper, we aimed to enforce phishingdetection by studying the URL of the webpage. URL is a complicated string that expresses syntactically and semantically expressions for a useful resource to be had over the Internet. In its maximum simple form, it's as follows <protocol>://<hostname><uri>, Fields together with the domain, subdomain, Top Level Domain (TLD), protocol, directory, file name, path, and query allow growing extraordinary URL addresses. These associated fields inside the phishing URLs are normally extraordinary from the legitimate ones on websites. Therefore, URLs have vital vicinity in detecting phishing assaults in particular for classifying the net web page quickly. It is discovered from the literature overview that powerful features received from the URL boom the accuracy of the classification.

3.1 Dataset

Dataset includes legitimate and malicious URLs. The URLs of legitimate websites were collected from Kaggle and the URLs of phishing websites were collected from Phistank websites. Benign URLs are labelled as “1" and phishing URLs are labelled as “-1".

3.2 Feature Collection

A phishing URL and also the corresponding page has many options which may be differentiated from a secure URL. So that we've collected many options based on major categories: URL-based, domain-based, content-based, site popularity-based and security-based options. The relevant options collected from these classes helps in differentiating phishing websites from legitimate websites.

3.3 Feature Extraction

The Feature Extraction phase is to derive different features of a URL. In this phase, we have extracted 27 features from the URL. These 27 features are identified from the major categories in the feature collection phase. The 19 are: Protocol, Domain, Path, TLD, Presence of HTTP, Rank, WHOIS registrar, age of the domain, domain registration length, SSL, Presence of IP in URL, URL Length, URL path length, Hostname Length, Tokens, No of special characters, No of slash, No of the hyphen, Dots in URL, Presence of security sensitive words, ASN, tiny URL, Presence of redirectionsymbol, email submission, iframe, links in tags, presence of anchor tag. From the 27 features extracted,
16 features which are Protocol, Domain, Path, TLD, Presence of HTTP, Presence of IP in URL, URL Length, URL path length, Hostname Length, Tokens, No of special characters, No of slash, No of the hyphen, Dots in URL, Presence of security sensitive words, Presence of redirection symbol belongs to URL-based category, 4 features that are email submission, iframe, links in tags, presence of anchor tag belongs to Content-based features, 4 features that are WHOIS registrar, age of the domain, domain registration length, Autonomous systemnumber belong to Domain-based category, 2 featureswhich are Rank and Tiny URL belong to Site popularity features and 1 feature SSL belongs to Security based features.

3.3.1 URL based features:

1.Protocol: Presents the protocol as a part of the link.
2.Domain: Presents the domain as a part of the link.
3.Path: Presents the path of the link.
4. TLD: Presents the highest level domain of the universal resource locator.
5. Presence of hypertext transfer protocol: If the HTTP token is in the link then the feature is set to one else to zero.
6. Presence of IP in link: If IP address is in URL then the feature is set to one else set to zero.
7. Length of link: If the length of the URL is larger than or equal to fifty-four then the feature is ready to one else set to zero.
8. Length of universal resource locator path: Presents the length of the path
9. Length of hostname: Presents the length of the hostname.
10. Tokens: Presents the number of tokens within the link.
11. No. of special characters: Presents the number of special characters in the URL.
12. No. of slash: If the amount of slashes within the link is on larger or equal to 5 then the feature is ready to one else set to zero.
13. No. of hyphens: Presents the number of hyphens within the link.
14. No. of dots in universal resource locator: If the amount of dots within the URL is greater than three then the feature is ready to one else set to zero.
15. Presence of security sensitive words: Presents the count of security sensitive words within the link.
16. Presence of redirection image: If the redirection symbol is in the link then the feature is ready to oneelse set to zero.

3.3.2 Content-based features:

1. Email submission: If mailto function is present in the URL then feature is set to 1 else to 0.
2. IFrame: If iframe tag is present in the URL then feature is set to 1 else to 0.
3. Links in tag: Presents the numbers of links found in the URL.
4. URL of the Anchor: Presents the numbers of anchor tags found in the URL.

3.3.3 Domain-based features:

1. WHOIS Registrar: Presents the name of the registrar from the WHOIS database.
2. Age of domain: If the age of the domain is smaller than six months then the feature is about to one else set to zero.
3. Domain registration length: If the domain registration length is a smaller than or capable of one year then the feature is set to one else set to zero.
4. Autonomous system number: Presents the Autonomous system number(ASN).

3.3.4 Site popularity features:

1. Rank: If rank of the website from Alexa database is greater than 1, 00,000 then feature is set to 1 else to 0.
2. Tiny URL: If the URL is crafted using shortening services then feature is set to 1 else 0
Security based features:
1. SSL: If the SSL certificate is not present, then the feature is set to 1 else set to 0.

3.4 Feature Selection

To find the best set of features, we have analyzed the robust features and selected those features which contribute more to the prediction output. From the collected 27 features, we have removed 6 non-robust features. Finally 21 features were selected to train and test the classifier. The 21 features are: Presence of http, Rank, age of domain, domain registration length, SSL, Presence of IP in url, URL Length, URL path length, Hostname Length, Tokens, No of special characters, No of slash, No of hyphen, Dots in url, Presence of security sensitive words, ASN, tiny url, Presence of redirection symbol, iframe, links in tags, presence of anchor tag.

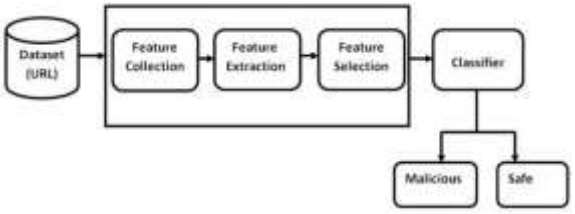

Fig.1 Flow of Proposed System

4 The Evaluation and Results

In The URL Detection model was trained and tested using Gradient Boosting, Random Forest and Support Vector Machine classifiers. The ratios in which we splitted the train and test dataset is 80:20. The models were tested against various metrics such as precision, recall, f1 score. Gradient Boosting yielded an accuracy of 98%, Random Forest model yielded an accuracy of 96% and SVM yielded an accuracy of 95%. Thus the Gradient Boosting model yielded a high accuracy compared to Random Forest and SVM. Therefore we applied Gradient Boosting classifier for prediction.

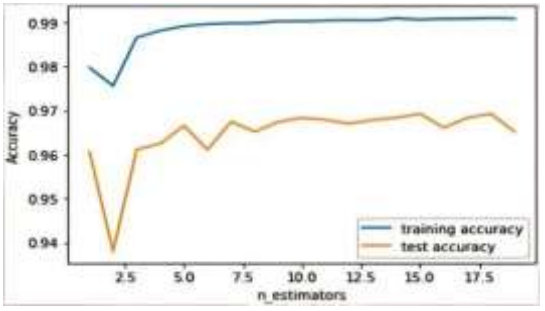

Fig.2 Random Forest Train vs. Test Accuracy Graph

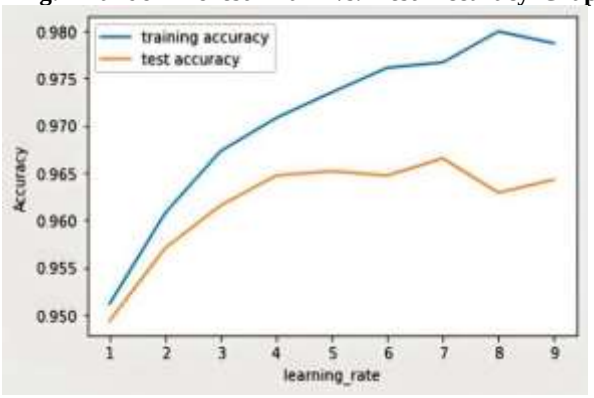

Fig.3 Gradient Boosting Train vs. Test Accuracy Graph (Learning Rate)

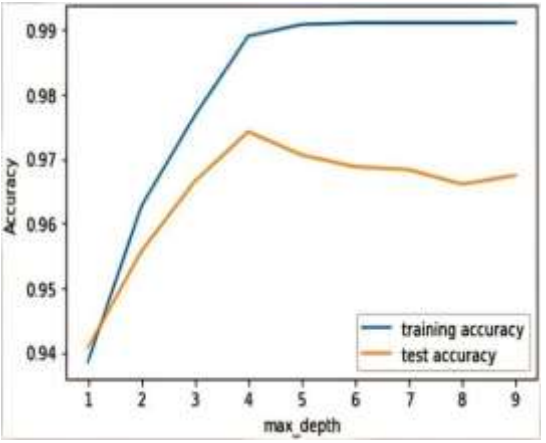

Fig.4 Gradient Boosting Train vs. Test Accuracy Graph (Maximum Depth)

5. Conclusion and Future Scope

Nowadays, the number of malicious websites has increased extensively. The proposed system thus aims to enlighten people all over the world who are unaware of the malicious URLs threats and thereby help people to get knowledge about these websites. The URL Detector lets people know about the URL features and behavior. Legitimate and malicious URLs were used as datasets and robust features were extracted. From Gradient Boosting, Random Forest and SVM classifiers used, Gradient Boosting gave an accuracy of 98%, Random forest gave an accuracy of 95% whereas SVM gave an accuracy of 67%. Thus Gradient Boosting classifier was used to detect malicious websites.

The proposed work can be further improved bycollecting a huge dataset and extracting many robust features. Malicious URLs are generated on a daily basis, thus collecting all the URLs was a tedious task. The system can be enhanced by training using a huge amount of datasets and using different classifiers and algorithms. The proposed system predicts the results whether safe or malicious websites, which can be enhanced by blocking all those websites.

References

[1] Pawan Prakash, Manish Kumar, Ramana Rao Kompella,MinaxiGupta,Purdue University, Indiana University "PhishNet: Predictive Blacklisting to Detect Phishing Attacks".

[2] JungMin Kang and DoHoon Lee "Advanced White List Approach for Preventing Access to Phishing Sites".

[3] A. Mishra and B. B. Gupta, “Hybrid Solution to Detect and Filter Zero-day Phishing Attacks”,ERCICA, 2014.

[4] Jian Mao, Pei Li, Kun Li, Tao Wei, and Zhenkai Liang, “Bait Alarm Detecting Phishing Sites Using Similarity in Fundamental Visual Features”, INCS 2013.

[5] Matthew Dunlop, Stephen Groat, and David Shelly, “GoldPhish Using Images for Content-Based Phishing analysis”, IEEE 2010.

[6] D. Sahoo, C. Liu, S.C.H. Hoi, “Malicious URL Detection using Machine Learning: A Survey”. CoRR, abs/1701.07179, 2017

[7] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, “Identifying suspicious urls: an application of large- scale online learning,” in Proceedings of the 26th Annual International Conference on Machine Learning. ACM, pp. 681–688, 2009.

[8] B. Eshete, A. Villafiorita, and K. Weldemariam, “Binspect: Holistic analysis and detection of malicious web pages,” in Security and Privacy in Communication Networks. Springer,pp. 149–166,2013.

[9] S. Purkait, “Phishing counter measures and their effectiveness– literature review,” Information Management & Computer Security, vol. 20, no. 5, pp. 382–420, 2012.

[10] Y. Tao, “Suspicious url and device detection by log mining,” Ph.D. dissertation, Applied Sciences: School of Computing Science, 2014.

[11] G. Canfora, E. Medvet, F. Mercaldo, and C. A. Visaggio, “Detection of malicious web pages using system calls sequences,” in Availability, Reliability, and Security in Information Systems. Springer, pp. 226–238,2014.

[12] D. Sahoo, C. Liu, S.C.H. Hoi, “Malicious URL Detection using Machine Learning: A Survey”. CoRR, abs/1701.07179, 2017.

[13] M. Khonji, Y. Iraqi, and A. Jones, “Phishing detection: a literature survey,” IEEE Communications Surveys & Tutorials, vol. 15, no. 4, pp.2091–2121, 2013.

[14] Hiba Zuhair, Ali Selamat, and Mazleena Salleh.Feature selection for phishing detection: a review of research,2016.