Machine Learning Algorithms for Detecting Phishing Websites

K P SENTHIL KUMAR M.E.,C.S.E.,
Assistant Professor, Dept of AI&DS,
Kings Engineering College, Chennai-600016.

ABSTRACT

The simplest method of obtaining sensitive information from unwitting people is through a phishing attack. The goal of phishers is to obtain crucial data, such as username, password, and bank account information. People working in cyber security are currently looking for reliable and consistent methods of detecting phishing websites. In order to distinguish between legal and phishing URLs, this article uses machine learning technology. It extracts and analyzes many aspects of both types of URLs. Algorithms such as Support Vector Machine, Decision Tree, and Random Forest are used to identify phishing websites. By evaluating each algorithm's accuracy rate, false positive and false negative rates, the study aims to identify phishing URLs and identify the best machine learning method.

Keywords

Phishingattack,Machinelearning

1. INTRODUCTION

Due to how simple it is to develop a phony website that closely resembles a legitimate website, phishing is now a top worry for security researchers. Although experts can spot fraudulent websites, not all users can, and as a result, some users fall prey to phishing scams. The attacker's primary goal is to obtain login information for bank accounts.Businesses in the US lose $2 billion annually as a result of their customers falling for phishing schemes [1]. The annual global cost of phishing was pegged at $5 billion in the third Microsoft Computing Safer Index Report, which was published in February 2014 [2].

Due to a lack of user awareness, phishing assaults are becoming more successful. Since phishing attacks take advantage of user vulnerabilities, it is highly challenging to mitigate them, but it is crucial to improve phishing detection methods. The "block list" method, which is the standard technique for detecting phishing websites, involves adding rejected URLs and Internet Protocol (IP) addresses to the antivirus database. Attackers modify the URL to appear authentic by obfuscation and many other straightforward ways, such as fast-flux, in which proxies are automatically constructed to host the website, algorithmic production of new URLs, etc., to dodge blacklists.

A zero-hour phishing assault can be detected using heuristic-based detection, which contains features that have been shown to exist in phishing attacks in reality, but the existence of these traits is not always guaranteed in such attacks, and the false positive rate for detection is very high [3].Many security experts are now focusing on machine learning techniques to overcome the limitations of blacklist and heuristics-based methods. Machine learning technology is made up of numerous algorithms that use historical data to forecast or make decisions about future data. This method uses an algorithm to examine a variety of genuine and blacklisted URLs and their characteristics in order to precisely identify phishing websites, including zero-hour phishing websites.

2. DATASET

From www.alexa.com and www.phishtank.com, respectively, URLs of trustworthy websites were gathered. The data collection includes 17058 benign URLs, 19653 phishing URLs, and a total of 36,711 URLs. Phishing URLs are labeled with a "1" whereas benign URLs are given a "0" designation.

3. FEATUREEXTRACTION

A Python application has been created to extract features from URLs. The features that we have culled for phishing URL detection are listed below.

1) IP address in the URL: If an IP address is present in the URL, the feature is set to 1, otherwise it is set to 0. The majority of trustworthy websites never use an IP address as the URL to download a webpage. The use of an IP address in a URL suggests that the attacker is attempting to steal sensitive data.

2) The @ sign in the URL: If the @ symbol is present in the URL, the feature is set to 1, otherwise it is set to 0. When phishers add a specific @ sign to a URL, the browser ignores everything before the "@" symbol and frequently skips to the true address after the "@" symbol.

3) Number of dots in Hostname: Phishing URLs have many dots in URL. For example http://shop.fun.amazon.phishing.com, in this URL phishing.com is an actual domain name, whereas use of “amazon” word is to trick users to click on it. Average number of dots in benign URLs is 3. If the number of dots in URLs is more than 3 then the feature is set to 1 else to 0.

4) If a domain name is separated by a dash (-), the prefix or suffix is set to 1; otherwise, it is set to 0. Legitimate URLs rarely employ the dash symbol. Phishers include the dash symbol (-) in the domain name to give users the impression that they are visiting a trustworthy website. For instance, the real website address is http://www.onlineamazon.com, but phishers can construct a phony version of it called http://www.online-amazon.com to trick unwary people.

5) URL rerouting: If "//" is included in the URL path, the feature is set to 1; otherwise, it is set to 0. The user will be moved to another website if the URL path contains the character "//" [4].If there is an HTTPS token in the URL, the feature is set to 1; otherwise, it is set to 0. In order to deceive users, phishers may append the "HTTPS" token to the domain portion of a URL. For instance, see [4] at https :// www. paypal-it-mpp-home.soft-hair.com.

6) Email submission of User Information: Phishers may use the "mail()" or "mailto:" functionalities to send User Information to their own personal email[4]. If the URL contains such functions, the feature is set to 1; otherwise, it is set to 0.

7) "Tiny URL" URL Shortening Services: This service enables phishers to conceal lengthy phishing URLs by making them brief. User traffic is being diverted to fraudulent websites. If the URL has been shortened using a service like bit.ly, then feature is set to 1, otherwise it is set to 0.

8) Host name length: The benign URLs' average length was discovered to be 25, and if the length is larger than 25, the feature is set to 1; otherwise, it is set to 0.

9) The presence of sensitive words in the URL: Phishing websites include sensitive words in their URLs to give consumers the impression that they are visiting a trustworthy website. The following words can be found in numerous phishing URLs:- "confirm," "account," "banking," "secure," "ebyisapi," "webscr," "signin," "mail," "install," "toolbar," "backup," "PayPal," "password," "username," etc.;

10) The number of slashes in the URL is discovered to be a 5. If the number of slashes in the URL is larger than 5, the feature is set to 1; otherwise, it is set to 0.

11) The URL contains Unicode characters: Phishers can utilize Unicode characters in URLs to fool users into clicking on them. The domain "xn--80ak6aa92e.com," for instance, is equal to "appe.com." The user can see the URL "appe.com," but when they click it, they are taken to the phishing website "xn--80ak6aa92e.com."

12) SSL Certificate Age: The use of HTTPS is crucial in creating the perception that a website is legitimate [4]. However, a benign website's SSL certificate must be at least one to two years old.

13) URL of Anchor: This feature was extracted by crawling the URL's source code. The a> tag specifies the URL of the anchor. The feature is set to 1 if the maximum number of hyperlinks in the a> tag are from another domain; otherwise, it is set to 0.

14) IFRAME: By crawling the URL's source code, we were able to extract this functionality. Using this element, one web page can be added to the primary webpage already there. The "I frame" tag can be used by scammers to render their content invisible, or without frame boundaries [4]. The user may enter critical information because the added webpage's border is invisible and appears to be a part of the original webpage.

15) Website Rank: We extracted each website's ranking and compared it to the top 100,000 websites in the Alexa database. The feature is set to 1 if the website's rank is more than 10,0000; otherwise, it is set to 0.

4. MACHINELEARNINGALGORITHM

one of the most used algorithms for machine learning. The decision tree method is simple to comprehend and use. A decision tree's job starts by selecting the best splitter from the qualities that are available for classification, which is referred to as the tree's root. The algorithm keeps growing the tree until it comes across the leaf node. In a decision tree representation, each internal node corresponds to an attribute, while each leaf node corresponds to a class label. This training model is used to forecast target values or classes. Gini index and information gain approaches are employed in the decision tree algorithm to calculate these nodes.

4.1 RandomForest Algorithm[6]

One of the most potent machine learning algorithms, the random forest algorithm is built on the idea of the decision tree algorithm. The forest with many decision trees is created by the random forest algorithm. High detection accuracy is provided by large numbers of trees.

The bootstrap approach is used in the tree creation process. In the bootstrap method, a single tree is constructed using randomly chosen dataset attributes and samples. Like the decision tree technique, the random forest approach uses the gini index and information gain methods to discover the best splitter from among the randomly chosen characteristics for categorization. This method will keep on until the random forest produces n trees.

Each tree in the forest makes a prediction for the target value, and an algorithm then determines the votes for each target prediction. High-vote projected target is finally taken into account as a final prediction by the random forest algorithm.

4.2 SupportVectorMachineAlgorithm[7]

Another potent algorithm in machine learning is the support vector machine. Each piece of data is displayed as a point in n-dimensional space using the support vector machine technique, which also creates a line that divides the data into two groups. This line is known as a hyperplane.

Support vector machines look for the nearby points, known as support vectors, and once they are located, they are connected by a line. Then, using a support vector machine, a separating line that bisects and is perpendicular to the connecting line is created.

Data should be perfectly classified with the biggest margin possible. In this case, the margin is the separation between the support and hyperplane vectors. Complex and nonlinear data cannot be separated in the real world, hence support vector machines employ a kernel method to convert lower dimensional space to higher dimensional space in order to tackle this difficulty.

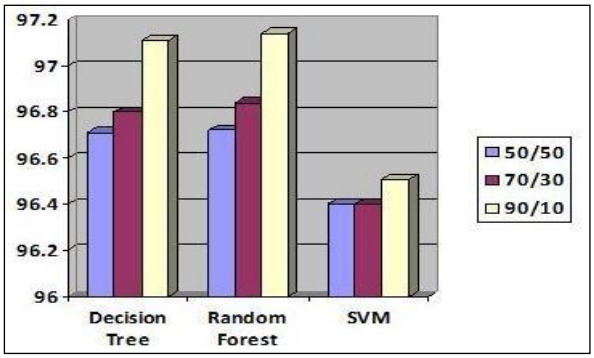

Fig.1Detectionaccuracycomparison

5. IMPLEMENTATIONANDRESULT

Machine learning methods have been imported using the Scikit-learn tool. The data set is split into training and testing sets in the following ratios: 50:50, 70:30, and 90:10. Each classifier is trained using a training set, and the performance of the classifiers is assessed using a testing set. The accuracy score, false negative rate, and false positive rate of classifiers have all been calculated in order to assess their performance.

Table1:Classifier'sperformance

Dataset Split ratio	Classifiers	Accuracy Score	False Negative Rate	False Positive Rate
50:50	DecisionTree	96.71	3.69	2.93
	Random Forest	96.72	3.69	2.91
	Support vector machine	96.40	5.26	2.08
70:30	DecisionTree	96.80	3.43	2.99
	Random Forest	96.84	3.35	2.98
	Support vector machine	96.40	5.13	2.17
90:10	DecisionTree	97.11	3.18	2.66
	Random Forest	**97.14**	**3.14**	2.61
	Support vectorm achine	96.51	4.73	2.34

The results demonstrate that compared to decision tree and support vector machine methods, the random forest algorithm provides greater detection accuracy, with a score of 97.14 and the lowest false negative rate.

The results also demonstrate that as more training datasets are used, the accuracy of phishing website identification increases. When 90% of the data is used as the training dataset, all classifiers function well.

Figure 1 displays the detection accuracy of all classifiers when 50%, 70%, and 90% of the data are used as training datasets. It is evident from the graph that the detection accuracy rises when 90% of the data are used as training datasets, and that the detection accuracy of the random forest classifier is highest compared to that of the other two classifiers.

6.CONCLUSION

This study uses machine learning technologies to improve the detection of phishing websites. Using the random forest technique, which has the lowest false positive rate, we were able to detect with 97.14% accuracy. Additionally, the results demonstrate that classifiers perform better when more data is used as training data.In the future, hybrid technology that combines the blacklist approach with the random forest algorithm of machine learning technology will be utilized to more reliably detect phishing websites.