Phishing-Inspector: Detection & Prevention of Phishing Websites

Tanmay Doke, Pranav Khismatrao, Vaibhav Jambhale and Mr. Nilesh Marathe.

Department of Information Technology, Ramrao Adik Institute of Technology, Dr. D.Y. Patil Vidyanagar, Nerul, Navi Mumbai, Maharashtra – 400706, India

Abstract — With a tremendous boost in technologies & available learning material developing any website has become very easy. Due to this the number of websites are exponentially growing day by day. The traditional approach of comparing websites with Blacklist and whitelist is not so efficient. As attackers have become more intelligent regarding hiding and redirecting of the URL and thereby tricking the user into phishing attack without been getting detected. So there's a need to have a novel approaches based on Machine learning (ML) which would expose this phishing websites. In this paper, the proposed system is an extension to web browser which made use of ML algorithm to extract various features and thereby helping the user to distinguish between Legitimate website and Phishing website.

Keywords — phishing websites, machine learning, blacklist, whitelist, phishing attacks, anti-phishing extension, SSL certificate, classification, legitimate, security.

I. INTRODUCTION

Phishing is a social engineering attack or fraudulent attempt to steal user's identity & private information. Personal details such as credit card number and login credentials are often stealing from the user by sending an email from trusted looking dummy site. User's been unaware of such activities tends to open the link which is sent by the attacker and ends up giving their login credentials. Once the login credentials are exposed to the attacker, various activities such as removal of money from bank account or using fake identity to do various illegal things are encouraged to be done by attacker.

Websites are also designed in such a way that they tend to appear exactly similar with the legitimate site. Displaying some URL and redirecting to some other URL is so common nowadays. In today's scenario many users prefer to do online shopping. As online transactions are increasing day-by-day identifying such phishing websites is so important. About 42% of phishing attack is been observed in payment sector.

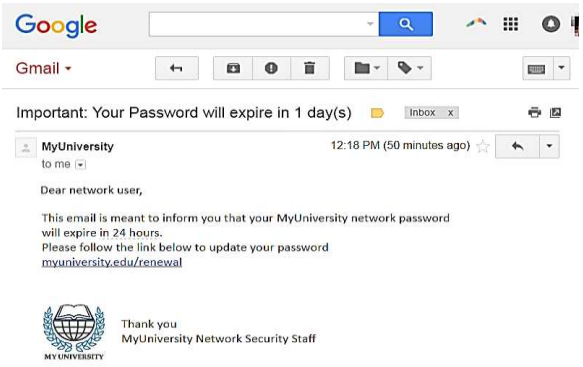

Fig.1 Web spoofing attack performed by attacker

Fig.1 depicts how a fake website which is been designed professionally like a legitimate website is accessed by the user. Thinking it as a legitimate website user will share his login credentials by clicking on that link, which may cause in installing some malware into his system. Once the attacker gets the credentials it can be used for any illegal purposes. There are many targets where phishing is done on large scale such as Facebook, Paytm, PayPal, Bank accounts, etc.

The remainder of this paper is structured as follows: in section II, survey on related work is reviewed. In section III, detailed explanation of

proposed methodology along with system flow diagram and analysis is described. In section IV, results are displayed. In section V, overall understanding and outcomes are summarized and concluded. In section VI, references are mentioned.

II. RELATED WORK

This section reviews various research done in order to detect and prevent phishing websites. Many researchers have applied different methods and algorithm in process to make their system more accurate. We have reviewed various work and gathered information which could help our system to result in more accurate results.

Jhen-Hao Li et al. [1] they have proposed an approach called Phish-Box to effectively collect phishing data and generate models for phishing validation and detection. It is a Two Staged Detection Model in which first data is collected from databases using ETL approach. Blacklist from Phish Tank is extracted and Alexa Toplist is extracted to check whether site is Phish site or legit site. Various features or parameters are considered like Host information, URL, Content, Screenshot. Second Stage is Voting Module.

Samuel Marchal et al. [2] they have designed a distinct concept explaining Intra-URL relatedness which is then carefully evaluated using a Features extraction from the letters and words that form a URL. The features extracted are then analyzed by using them as input in a machine learning Classification Model. 12 features are extracted from an individual URL based on Random Forest Classifier. A score is generated for every URL. It has accurate classification rate of 94.90.

P.A. Barraclough et al. [3] they have proposed an approach filed Online Phishing Detection. The entire system is made up 4 components: Six sets of data inputs, Algorithm to extract features, Website to be checked, An result phase. The 6 set of data inputs is applied to fuzzy if-then extractor algorithm. If the presence of phishing site is detected, a voice generating user warning signals are generated to alert the user of the website. User is alerted by giving red color dialogue box and notified by green color dialogue box.

Samuel Marchal et al. [4] this application was designed to be Client-Side only for Enhanced Privacy and extra protection. Resilience to dynamic Phish are some of the important features. The various data sources collected for the detection purpose are: Starting URL, Redirection Chain Landing URL, Logged Links and Webpage Contents. It is a Bowser Add-On and Brand Independent Application.

M. Zabihimayvan et al. [7] they have proposed a Fuzzy Rough Set theory tool to select most effective features from 3 data sets. For evaluating the FRS Feature Selection, they developed a generalizable phishing detection model in which the classifiers are trained by a separate data set consisting of 14000 website samples. 95% is the maximum F-measure gained by FRS Feature Selection.

III. PROPOSED WORK

A. Overview of our approach

The overall approach of our system is depicted in the figure. 1. The system is an Extension to the Web browser. The system initially extract's the URL of the website from the browser. Then the extracted URL is checked in dataset, which contains list of phishing website and legitimate website. If the URL is found in dataset labeled as legitimate website then the current website is declared as Legitimate site, but if the URL is found in dataset as phishing site then the current website is declared as phishing website. Same is the process for suspicious website. If the URL is not found in dataset lists, then the URL is passed as an input to the Phishing Detector machine learning algorithm. The algorithm used is Random forest algorithm and it uses 13 different kinds of features for decision making. The output of the ML algorithm is then given to the SSL certificate checker which checks the SSL certificate of the website After that output of SSL certificate is given to Domain creation and expiration date checker which checks for how much period the webpage is active. Based on the results the website is detected as Legitimate or Suspicious or Phishing.

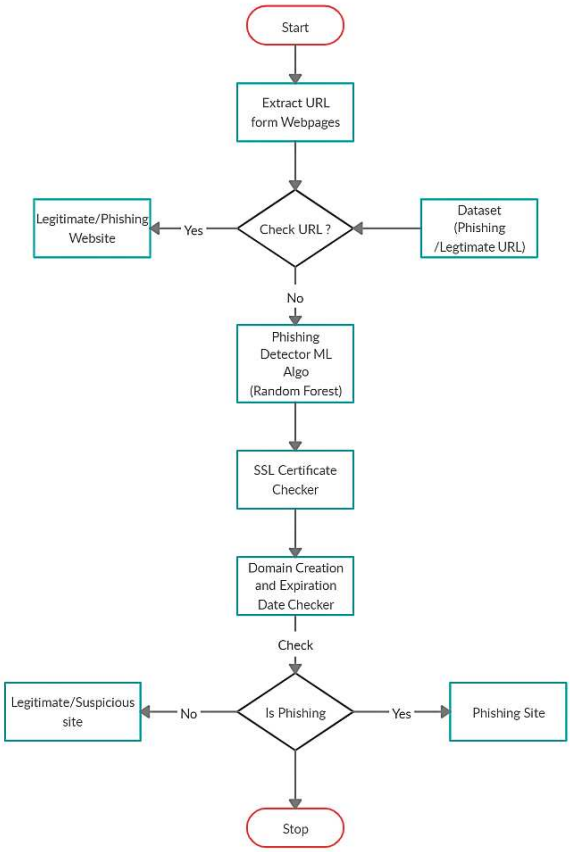

Fig 2. Architecture

B. Requirement Analysis

In order for the system to work smoothly the user's machine should satisfy the following software and hardware requirements.

1. Software Requirements

- Python 3.6 or above: Higher level language used for backend.
- Beautiful Soup: A package in python which is used for web scrapping.
- Scikit-learn: A package in python which has libraries for machine learning.
- JavaScript: Used for getting URL from the extension.
- Browser: A Google Chrome browser.

2. Hardware Requirements

- Hard disk with minimum of 250 GB.
- Minimum 4 GB RAM.
- Windows 7, 8, 8.1, 10.
- System Type: 32-bit Operating System and above.

C. Design Phase

1. Design Goals

- The system created should not be dependent on features specific to a target.
- To reduce the number of phishing attacks thereby helping the user to preserve confidential information.
- To detect real time generated phishing websites.
- To maintain privacy of user by not asking them their browsing history.
- The system can be used by business organization to conduct phishing campaigns.
- The warning messages generated by the system must be clear and easy to understand.

2. Design Choices

- The system made is totally client-side. It uses the URL features and domain information to compute the decision.
- The system uses some static and some non-static features for the detection purpose.
- The created system also relies on some predefined feature characteristics which are bound to present in most of the phishing websites.

3. Architecture and Implementation

The system is developed as an add-on to a web browser like chrome and is suitable windows operating system.

The add-on is created using JavaScript. When the user clicks on the icon on the navigation bar, a popup is displayed as shown in figure. Once the user clicks the check button the JavaScript method collects the URL of the current website. This URL is then passed to PHP file which executes the python file. This add-on is used to display the result.

The background process is developed in python. The main reason to use python is because of the

availability of machine learning libraries. The python code does the following functionality:

Checking the website in the available dataset: The dataset has websites labeled as 1, 0, -1. Here '1' stands for phishing website, '0' stands for legitimate website and '-1' stands for suspicious website. This functionality is first executed on execution of python file. The extracted URL is searched in the dataset and the label of the matched URL is returned. If the URL is not found in the dataset, then the other functions are executed else only this function is executed.

Phishing detection ML Algorithm: If the URL is not found in the dataset then the URL is passed to machine learning algorithm. The machine learning algorithm is used to compute the decision based on the features which are extracted from the URL. Random Forest Algorithm is used for this purpose. Total 13 features are extracted from the particular URL which are then used by the ML algorithm. The result is then used for further decision making process.

SSL Certificate Checker: This function is used to get the SSL certificate of the particular website. The function takes the hostname of the website and fetches the respective SSL certificate. Python inbuilt method of 'ssl.get_server_certificate' is used for this purpose. The result is again forwarded for decision making purpose.

Domain Creation and Expiration Date checker: This function is used to get the creation date and the expiration date of the domain used in the particular URL. Using these parameters, the period remaining for the domain to expire and the period for which the domain is active till current date is calculated. If both the period calculated comes to less than 1 year, then the chances of the website being a phishing or suspicious website is more. The age of domain is also calculated i.e. the period for which the domain is registered. If it is less than 1½ years than the result is more aligned towards the website being a phishing website. This is because the attackers mostly take domains with less period, as the websites get banned once found out to be phishing websites. Python inbuilt method 'whois' is used for this purpose.

Final Decision Making: The final decision of the website being a phishing website or legitimate website or suspicious website is taken based on following table

Table 1 Implementation Logic

ML Output	SSL Certificate	Domain Duration	Result
Phishing	Not Present	-	Phishing Site
Legitimate	Not Present	-	Phishing Site
Phishing	Present	Less	Phishing Site
Legitimate	Present	Less	Suspicious Site
Phishing	Present	More	Suspicious Site
Legitimate	Present	More	Legitimate Site

4. User Interface

The user interface is used to check the legitimacy of the website and the result is displayed through it. The user interface is simple and has minimal text. The navigation bar icon is the logo of the developed system. This icon is always present on the browser.

5. Machine Learning Algorithm

To differentiate between a legitimate and phishing website we use machine learning technique. The attackers always have some limitations while creating a phishing URL, this is because a URL is composed of registered domain and free URL. Registered domain has to be used as it, but the attackers can modify the free URL as they want. So machine learning approach is used to differentiate the legitimate and phishing sites. We have used supervised learning technique. In supervised learning the model is trained used labeled dataset. The dataset which is used to train the model consists of different classes for different values. Once the model is trained, it can be used to predict the class of the unlabeled data which is given as an input to the model.

For the training purpose we are using Random Forest algorithm. In this algorithm random

samples of data are selected from the dataset, using these samples separate decision trees are created for each and every sample. After this, prediction result for every decision tree is generated and the best decision tree is used for training the model. The reason for using random forest algorithm is the minimum overfitting risk and also it takes less time to train. Another advantage is the efficiency of the algorithm for large datasets and its high accuracy.

For the training purpose different features are extracted from the URL. A total of 13 features are generated.

- No of dots: It tells the count of the dots used in the URL.
- No of delimiters: It counts the different kinds of delimiters in the URL.
- Presence of '-', '@', '//': It checks the presence of these parameters in the URL.
- Count of sub-directory, sub-domain and queries: It counts the number of times the mentioned parameters used in the URL.
- Length of the URL, whether shortening service is used and whether IP address is used as URL are the other features being used for the training purpose of the model.

D. Analysis Phase

1. Dataset

The dataset which is used for the training purpose consists of 7030 URLs. These 7030 URLs consists of both legitimate as well as phishing URLs. The count of phishing URLs is 3536 and the count of legitimate URLs is 3494. A new dataset is created for the newly tested URLs. This dataset is different from the dataset which is used during the training of the model. The model can be trained using this newly generated dataset after some amount of time when the URL count in the dataset reaches a certain value.

2. Accuracy for phishing websites

Phishing websites were correctly predicted with accuracy of 93.16%. It had false positive rate of 7.38%.

3. Accuracy for legitimate websites.

Legitimate websites were correctly predicted with accuracy of 92.62%. False positive rate of 6.84% was observed.

Table 2 Classification Results for Random Forest

URL	Phishing Class	Legitimate Class	Precision	F-1 Score	Accuracy
Phishing	93.16% (TP)	6.84% (FN)	93%	93 %	93%
Legitimate	7.38 % (FP)	92.62 % (TN)			

4. Accuracy of the System

Accuracy of the system is defined as the count of the legitimate and phishing URLs which are correctly classified over total number of URLs present in the dataset. The calculated accuracy of the system is **93%.** True Positive Rate: It is defined as the percentage of phishing websites which are correctly classified. False Positive Rate: It is defined as the percentage of phishing websites which are incorrectly classified.

5. Feature Analysis

Fig 3. Displays various classification results for 6 classifiers. But accuracy percentage, TN rate and TP rate of Random forest is greater other classifiers.

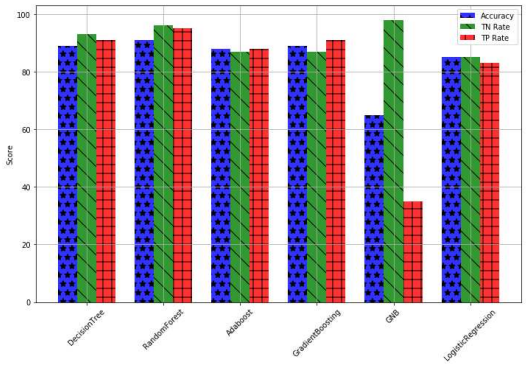

Fig 3. Classification Results for 6 Classifiers

IV. RESULT

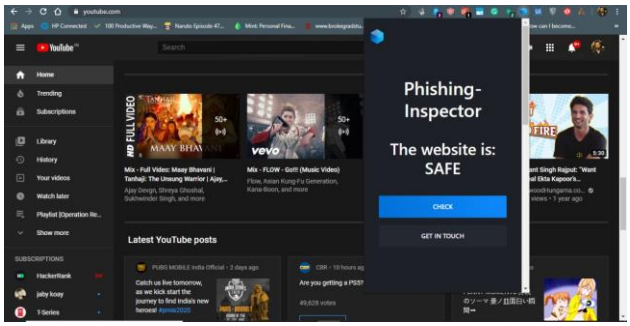

Fig 4. Safe Website

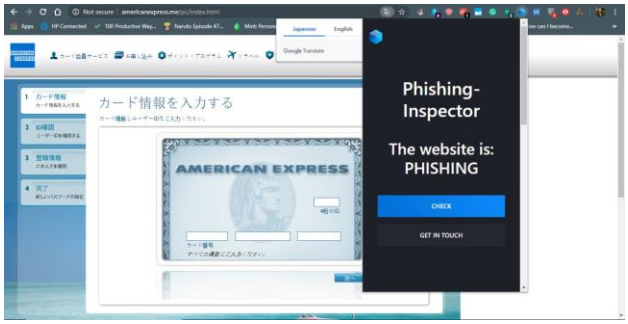

Fig 5. Phishing Website

Fig 4. Depicts the safe website. Fig 5. Depicts the phishing website which is caught by our phishing inspector. This website was not blocked by google chrome but here our system has detected it successfully. Such websites are fake as they try to trick the user to fall into their hands to perform illegal activities.

V. CONCLUSION

With the percentage of phishing attacks increasing this proposed system will enable the user for safe browsing and also for making safe transactions. It will also prevent sharing of confidential information of the user. Proposed system is able to differentiate between legitimate site and phishing site with ease so that anyone can use it without any hesitation. Education awareness should be spread across everyone so that there are less victims to such attack. Internet experts provides various security guidelines which must be followed while surfing on the internet. Users should not blind fondly open any URL link to encourage such attack. One must check the URL carefully and then only processed for further interaction where sensitive information is being asked.

VI. REFERENCES

[1] Jhen-Hao Li , Sheng-De Wang. PhishBox: An Approach for Phishing Validation and Detection. 2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress(DASC/PiCom/DataCom/CyberSciTec)

[2] Samuel Marchal , Jérôme François , Radu State , Thomas Engel.] PhishStorm: Detecting Phishing With Streaming Analytics. Engel IEEE Transactions on Network and Service ManagementYear: 2014 , Volume: 11

[3] P.A. Barraclough, G. Sexton & N. Aslam. Online Phishing Detection Toolbar for Transactions. Science and Information Conference 2015 July 28-30 2015 | London, UK Computer Science and Digital Technology University of Northumbria Newcastle Upon Tyne, NE 18ST, United Kingdom.

[4] Samuel Marchal ; Giovanni Armano ; Tommi Gröndahl ; Kalle Saari ; Nidhi Singh ; N. Asokan. Off-the-Hook: An Efficient and Usable Client-Side Phishing Prevention Application. IEEE Transactions on ComputersYear: 2017 , Volume: 66.

[5] Ludl, C., McAllister, S., Kirda, E., & Kruegel, C. (2007). On the effectiveness of techniques to detect phishing sites.In Detection of Intrusions and Malware, and Vulnerability Assessment (pp. 20-39). Springer Berlin Heidelberg.

[6] Y. Pan, X. Ding, "Anomaly based web phishing page detection", Proc. 22nd Annu. Comput. Security Appl. Conf., pp. 381-392, 2006.

[7] M. Zabihimayvan and D. Doran, "Fuzzy Rough Set Feature Selection to Enhance Phishing Attack Detection," 2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), New Orleans, LA, USA, 2019, pp. 1-6