International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 05 | May 2019 www.irjet.net p-ISSN: 2395-0072

Advanced Phishing Identification Technique using Machine Learning

Rajatha[1], Shravya U Shetty[2], Swathi[3], Tanuvi[4], Vidya Vittal Shetty[5]

[1]Assistamt Professor, Dept. of Information Science Engineering, Sahyadri College of Engineering and Management, Karnataka, India
[2,3,4,5]Student, Dept. of Information Science Engineering, Sahyadri College of Engineering and Management, Karnataka, India

---

Abstract - *Phishing is one of the severe issue existing in the present day situation. Regardless of the development of counteractive action methods, phishing remains an imperative risk since the essential countermeasures being used are as yet dependent on receptive URL blacklisting. This method is not useful in light because of the short lifespan of phishing Web sites, making ongoing methodologies depending on constant or proactive phishing URL. This project mainly aims at providing security to user by identifying the malicious pages using Random Forest Algorithm and to create a Chrome Browser Extension that helps user to verify the webpages.*

Key Words: **Machine Learning, Feature Extraction, Classification, Random Forest Algorithm, Phishing, Chrome Browser Extension.**

1. INTRODUCTION

This chapter characterizes the introduction of the project Advanced Phishing Identification Technique using Machine Learning.

In recent years Machine Learning is creating a lot of hype. Machine Learning is a division of Artificial Intelligence in the field of Computer Science that provides computer with the ability to learn without being explicitly programmed and can learn from data and make predictions of data by exploring the study and constructions of algorithms. In Advanced Phishing Identification Technique using Machine Learning project, machine learning techniques are used to detect the Phishing websites.

During the training process, the algorithm begins with loading the dataset i.e. in the form of ARFF. The following stage incorporates changing over the dataset from ARFF to CSV which is the required format for the model to work on. The next step is to filter the unwanted features from the dataset so that the model is given with only the important features for predictions, this will help in reducing the false rate of predictions for the detection. The next step is to divide the dataset into training and testing, this is fundamental since it causes us to cross-check whether the model is foreseeing the right yield for the testing dataset from its learnt training dataset. The subsequent stage is to train the model with the training dataset i.e. with the help of Random Forest Classifier. Here the input for RFA to split the internal node is given with the value 7 and the verbose is set to true. The next step is to output the predictions of the trained model on the testing data to check whether the model is predicting accurate target value or not The following stage is to ascertain the accuracy of the training model. To know whether the model is anticipating the sites as safe and phishing.

The algorithm for testing incorporates opening the Chrome browser and typing the URL in the address bar. When the site page is being loaded then it is retrieved by the Chrome Extension, which with the assistance of PHP code pass it to the python file where the feature extraction is is accomplished for various highlights of URL like abnormal, address, JavaScript and HTML features. Now, we send the extracted feature values to RFA trained model for forecast of that site. On the off chance that the anticipated value returned is – 1, at that point the outcome showed is Phishing site and if the value returned is 1 then it is Safe site.

2. RELATIVE OVERVIEW

This chapter gives details of the various works carried out for the detection of phishing websites.

Samuel Marchal et al. [1], has explained about an mechanized phishing identification framework that can break down continuously any URL so as to distinguish potential phishing sites. Here, we characterize the new idea of intra-URL relatedness and assess it utilizing features extricated from Yahoo and Google web indexes. An extension for URL phishingness rating framework showing high certainty rate is proposed.

Shradha Parekh et al. [2], has explained that Phishing is an unlawful activity wherein people are misled into the wrong sites by using various fraudulent methods. The aim of these phishing websites is to confiscate personal information or other financial details for personal benefits or misuse. As technology advances, the phishing approaches used need to get progressed and there is a dire need for better security and better mechanisms to prevent as well as detect these phishing approaches.

Ishant Tyagi et al. [3], has explained that Phishing can be described as a way by which someone may try to steal some personal and important information like login id's, passwords, and details of credit/debit cards, for wrong

reasons, by appearing as a trusted body. Many websites, which look perfectly legitimate to us, can be phishing and could well be the reason for various online frauds. These phishing websites may try to obtain our important information through many ways, for example: phone calls, messages, and popup windows. So, the need of the hour is to secure information that is sent online and one concrete way of doing so is by countering these phishing attacks.

Mehek Thakera et al. [4], has clarified that Phishing is kind of major digital dangers presently, where the people's qualifications are gotten by an ill-conceived site. This paper suggests a framework which will identify old just as recently produced phishing URLs that have totally no past practices to make a decision after, utilizing Data Mining. A cloud-based classification model will be made for a similar where in different separated characteristics through the URL will be utilized as information.

3. SYSTEM OVERVIEW

System design represents a plan or drawing that shows the function along with working of a system.

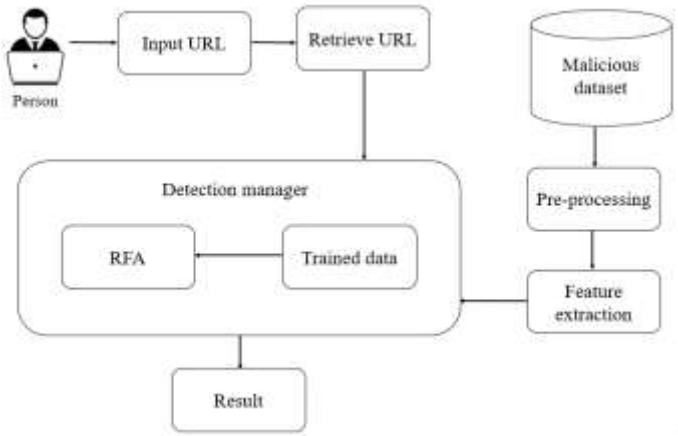

Fig -1: Architecture Diagram of Phishing Detection System

The system architecture of the Phishing Detection System is shown in Fig-1. The system will acknowledge the URL entered by the client and extract the features from the website. And afterward framework will give the extracted features to the detection manager. The detection manager contains the training data and Random Forest Algorithm (RFA) modules which thus takes the contribution from the malignant element analyzer. The training data contains malicious features which are extricated from the malignant dataset. The pre-processing manager takes data from the stored malignant dataset and then processes the data. The detection manager extricates the features from site and compares features that are present in the dataset. Feature extricated from dataset are maintained in a file and it is helpful in the test stage. This system is added as extension to the chrome browser with the goal that it functions as program Add-ons. At the point when the client enters any URL in the address bar, extension will check the URL and decides and gives the outcome as either phishing or normal page.

4. METHODOLOGY

This section tells about the implementation of the proposed system.

4.1 Algorithm for Training

Step 1: [Load the Dataset.]
Loading the file in ARFF format.
Step 2: [Conversion of ARFF file into required format.]
Converting ARFF file format into CSV format.
Step 3: [Pre-processing the features.]
Filtering the unrequired features from the malicious dataset.
Step 4: [Splitting the dataset.]
Splitting the dataset into train and test dataset.
Step 5: [Training the Model.]
Training the random forest classifier with training dataset.
Step 6: [Perform prediction.]
Printing what our model is predicting and the actual target.
Step 7: [Accuracy Calculations.]
Calculating train and test accuracy.
Step 8: [Confusion matrix.]
To know the true positive and true negative details of our trained classifier.
Step 9: [Plotting the feature importance graph and the overall performance graph.]
Predicting the contribution of each feature on the overall model's performance 5and also the performance of the system for different number of dataset.

4.2 Algorithm for Testing

Step 1: [Open Chrome Browser.]
Opening the browser.
Step 2: [Input URL.]
Enter the URL in address bar.
Step 3: [Retrieve URL.]
Browser extension will retrieve the URL from the address bar.
Step 4: [Extracting URL features.]
Extracting the features based on Address Bar, Abnormal, HTML and JavaScript, Domain based Features.
Step 5: [Apply Random forest for classification.]
Extracted features are sent to random forest classifier.
Step 6: [Check for Phishing Detection.]

The system checks for the similarity with the input URL and the trained model.

Step 7: [Output.]

Results either as Safe or Phishing site.

5. RESULTS AND ANALYSIS

This section tells us about the output of our project.

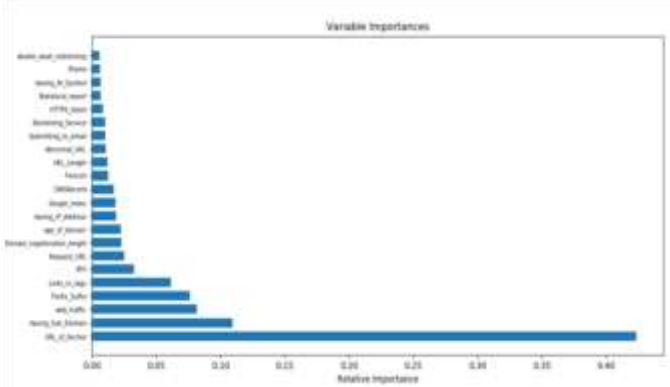

Fig-2: Relative importance of different number of features

The experimental results depends upon the performance of the feature's contribution to the system with training data. The accuracy of the system increased with the increased number of training data. More accurate results were obtained in good lighting conditions. Fig-2 depicts the relative importance of different number of features.

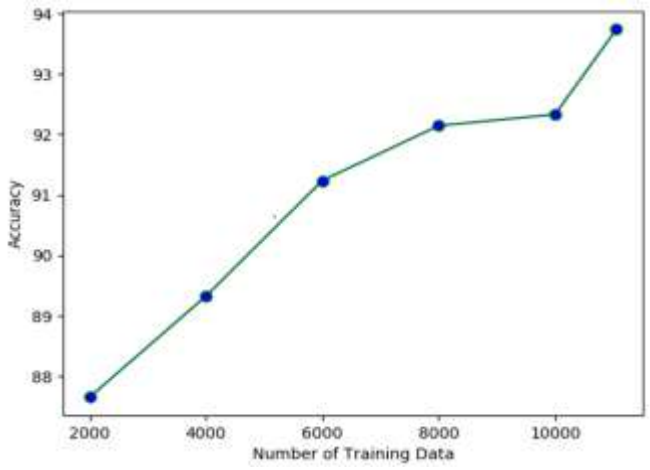

Fig-3: Phishing Detection Rate (%) for different number of dataset

Fig-3 depicts the Phishing Detection Rate (%) for different number of dataset. Here we have trained our model with different number of datasets and observed their accuracy rate.The accuracy increases with the increasing number of datasets.

6. CONCLUSION

The project is developed with a goal of providing security to the user to recognize web specific phishing pages continuously. The aim is satisfied with the assistance of Machine Learning techniques which will distinguish the malign pages efficiently. The methodology depends on URL relatedness. This relatedness reflects relationship among various URL based features, for example, Address based feature, Abnormal based feature, Domain based feature and HTML and JavaScript based features. Random Forest Algorithm is utilized for feature classification and prediction.

7. FUTURE WORK

The developed system provides better accuracy with some delay during training and testing. Future enhancements can be focused by implementing the project in such a way that when we click on the Browser Extension the phishing site has to be blocked instead of showing the popup message to the user.

REFERENCES

[1] Samuel Marchal, Jerome Francois, Radu State, and Thomas Engel “PhishStorm: Detecting Phishing With Streaming Analytics”, in IEEE Transactions on Network and Service Management, Vol. 11, No. 4, in 2014.

[2] Shradha Parekh, Dhwani Parikh and Srushi Kotak, “A new method for Detection of Phishing Websites: URL Detection", in IEEE 2nd International conference on Communication and Computational Technologies (ICICCT 2018) - Part Number CFP18BAC-ART; ISBN: 978-1-5386-1974-2, 2018.

[3] Ishant Tyagi, Jatin Shad and Shubham Sharma, “A Novel Machine Learning Approach to Detect Phishing Websites", in 5th Conference on Signal Processing and Integrated Networks (SPIN); ISBN: 978-1-5386-3045-7/18, 2018.

[4] Mehek Thaker and Mihir Parikh, “Detecting Phishing Websites using Data Mining", in 2nd International conference on Electronics, Communication and Aerospace Technology (ICECA 2018); ISBN: 978-1-5386-0965-1,2018.