Spoofing website alarm: An extension using supervised learning

Abhishek Tiwari, Siddhesh Nakipraj, Vinnie Varghese, Keerti Kharatmol
Student, Student, Student, Professor (Guide)
Computer Engineering,
K.C College of Engineering & Management Studies & Research (Mumbai University), Thane, India

Abstract: for a common internet user, a web *browser* is one of the regularly used programs on the computer and they have no idea about back end, sometimes they visit the phishing website and provide their confidential information. The information extracted may contain valuable credentials which can harm the common user. Our aim is to create an extension for browser which will act as middleware between user and malicious website and will be using machine learning technique to solve this issue. So now when the user tries to visit a phishing website user will be alarmed and will ensure safe browsing.

Index Terms **-Machine learning, Chrome, Extension, Phishing, Malicious.**

I. Introduction

Phishing is defined as mimicking a creditable company's website aiming to take private information of user. India among top three countries most targeted for phishing.65% of phishing occurs due to fake websites. We will recognize black lists website using intelligence systems. RBI has registered total 12320 cases of phishing in 2018. As per the survey conducted in the year 2017 "Survey on Malicious Web Pages Detection Technique", the authors stated that the web attacks are increasing year by year. In the year 2017, each month noted a total of approximately 1300,000 phishing attacks, which summed to a loss of $9 billion. Based on this data we can conclude that there is an urgent need to prevent such issues. According to [4], 70% of successful phishing attack happens through social network. Also the lack of awareness about the websites ''A personalized whitelist approach for phishing webpage detection," research paper proposed a personalized whitelist approach combined with a support vector machine (SVM) classifier. The phishing pages which are not listed in the whitelist are passed to SVM for further analysis. Blacklist/whitelist-based detection is easy to implement therefore it is widely use in various tools. But due to short life span of phishing website it leads to a less accurate result. Hence, we are using machine learning technique which identifies spoofing websites by the URL features. For this we are using chrome extension as a tool for implementation, because it is widely used.

The goal is to ensure safe browsing irrespective of the website which the user wishes to visit. Even if the user decides to visit

II. Literature Survey

As per the survey conducted in "Survey on Malicious Web Pages Detection Technique", the publisher stated that the web attacks are on a rise. In the year 2012, each month noted a total of approximately 33,000 phishing attacks, which summed to a loss of $687 million. . With such a rise in the number of spoofing attacks, there is a dire need of a system which prevents such types of spoofing attacks [1]. Most users are unaware of such phishing websites and hence they end up giving their valuable credentials and information. . So to prevent such type of attacks, a tool is needed which examines the URL entered by the user and checks if the website is safe or not. In the paper "Identifying Vulnerable Websites by Analysis of Common Strings in Phishing URLs" [2], the authors mention about the hike in the spoofing websites and the technique they used to find and prevent it. For the detection of a safe or a malicious webpage, they implemented Largest Common Substring (LCS) method. They developed a database of phishing websites of their own and tested the LCS on the new website. In the paper "On URL Classification", the authors stated about how URLs can be used to use the user's computer resources for different attacks like phishing, denial of service. Also, they compared various methodologies including traditional ones and the new trends in the field of machine learning. The results show that machine learning method are better than traditional method for detection [4].

Now, as we have to examine whether the website is malicious or benign, we will have to extract the features of the website. In the paper "Feature Extraction Process: A Phishing Detection Approach", the author talks about how the features can be extracted from a URL.

The author find 17 features which can be extracted from the URL based on that which URL can be declared as phishing or not.

III. Methodology

To overcome less accuracy rate intelligence system we will implement machine learning techniques to achieve high classification accuracy rate intelligence system. It will be quite faster than other techniques and it can deal with different websites for phishing detection Phishing attacks are increasing year by year, using a search in database of phishing website list is not enough. To provide protection from new phishing attacks, machine learning provides the best alternative for the same. We used the UCI Dataset of Phishing Website to train the classifier. Then, whenever a user enters the URL or visit a website, the features of that URL are extracted and the URL is tested on the trained classifier to obtain the result. Following steps involved in the implementation:

A. Obtaining Dataset

The dataset was obtained from the UCI - Machine Learning Repository [5] which contains the Phishing Web Site Dataset. This dataset is composed of 11055 entries of websites which are classified as phishing and benign. These entries each have 30 features of the website used.

B. Feature Selection

From the dataset, out of the 30 features present, it was difficult to extract all the features. This is because many features required some standard databases which is not accessible to us. Also extraction of some of the features become impossible because it required data from the server .Hence, we shortlisted our dataset which contain 22 important features. Some of them were:

- Using the IP Address

If an IP address is used as an alternative of the domain name in the URL, such as “http://125.98.3.123/fake.html”, users can be sure that someone is trying to steal their personal information. Sometimes, the IP address is even transformed into hexadecimal code as shown in the following link “http://0x58.0xCC.0xCA.0x62/2/paypal.ca/index.html”

- URL’s having “@” Symbol

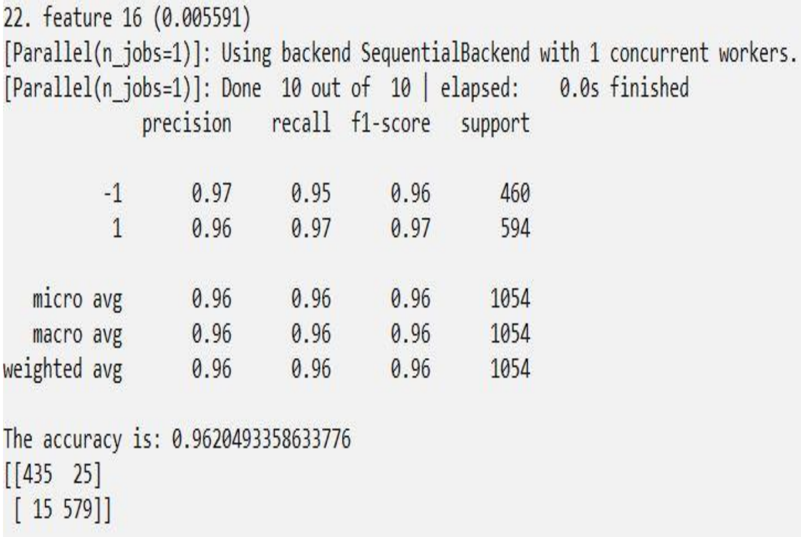
Using “@” symbol in the URL leads the browser to ignore everything preceding the “@” symbol and the real address often follows the “@” symbol.

C. Choosing Classification Algorithm

To find the URL, as either safe or not, we considered the following algorithm:

Random forest:- Random Forest is a flexible, easy to use machine learning algorithm that produces, even without hyper-parameter tuning, a great result most of the time. It is also one of the most used algorithms, because its simplicity and the fact that it can be used for both classification and regression tasks. Random Forest adds additional randomness to the model, while growing the trees. Instead of searching for the most important feature while splitting a node, it searches for the best feature among a random subset of features. This results in a wide diversity that generally results in a better mode.

For this algorithm, we calculated the accuracy, recall, F1- Score, precision. As per the result of these metrics, we got the best score with the RF algorithm. Hence, we train the classifier with the RF Algorithm. Below is the screenshot of the results of the algorithms.



```
22. feature 16 (0.005591)
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.
[Parallel(n_jobs=1)]: Done  10 out of  10 | elapsed:    0.0s finished
             precision    recall  f1-score   support

          -1       0.97      0.95      0.96       460
           1       0.96      0.97      0.97       594

   micro avg       0.96      0.96      0.96      1054
   macro avg       0.96      0.96      0.96      1054
weighted avg       0.96      0.96      0.96      1054

The accuracy is: 0.9620493358633776
[[435  25]
 [ 15 579]]
```

Figure1.Accuracy for algorithm

Also, for Random Forest, it is possible to use different values for minimum split size. Minimum split size denotes the least number of samples which are needed to split an internal node. The default number of nodes required for splitting an internal node is 2. By considering the different split sizes of 3 to 11, we got the results as shown in Table 1. In the table, TP-True Positive, TN-True Negative, FP-False Positive, FN-False Negative:

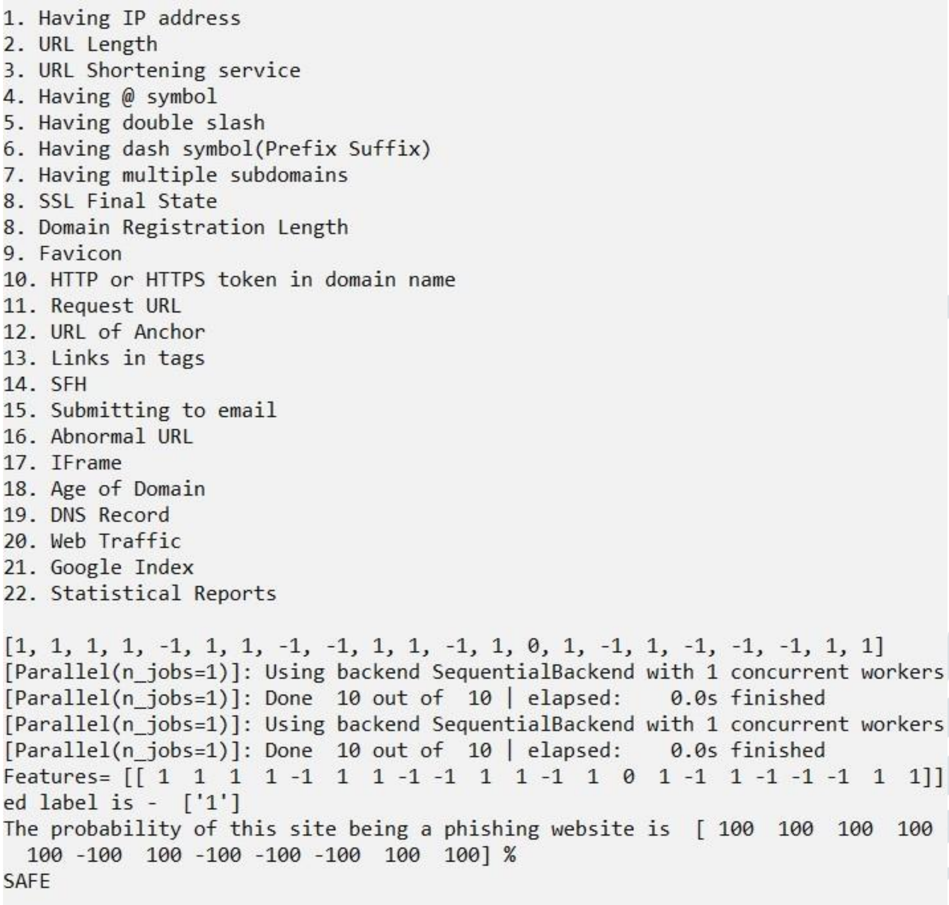
On observing the above table, minimum split size was selected as 7. So we applied the Random Forest Algorithm with the minimum split size of 7.

IV.RESULTS

When the URL of the website is entered in the extension, the program extracts the features of the URL and then they are tested on the trained classifier of Random Forest.

The screenshots of the result are as shown below.

1) Figure2 shows the result for safe website and the net result obtained was safe by chrome extension

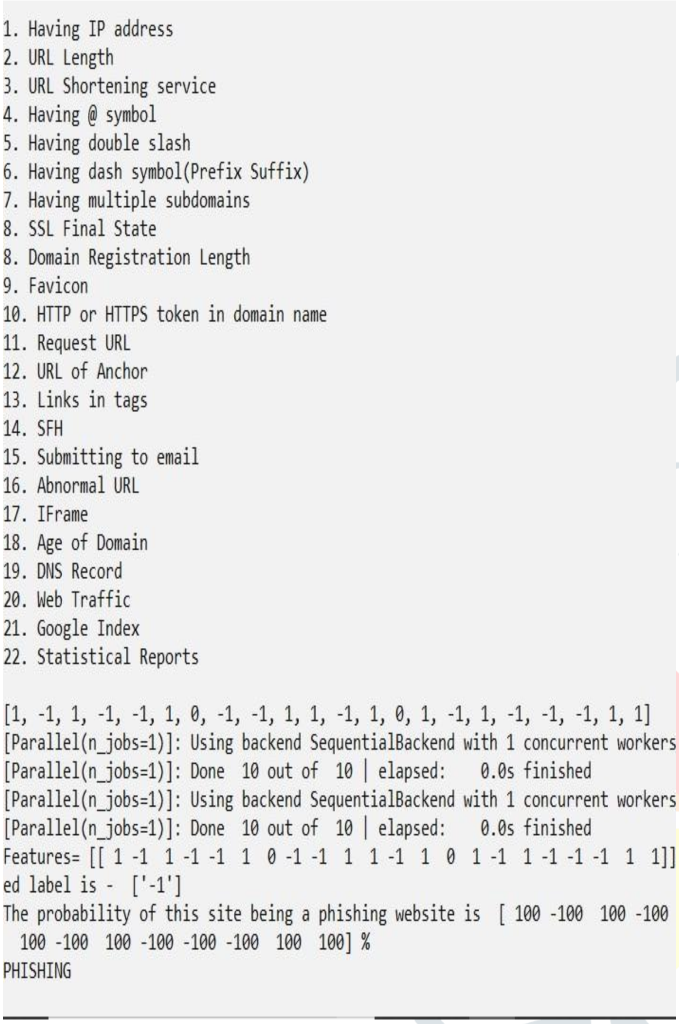


```
1. Having IP address
2. URL Length
3. URL Shortening service
4. Having @ symbol
5. Having double slash
6. Having dash symbol(Prefix Suffix)
7. Having multiple subdomains
8. SSL Final State
8. Domain Registration Length
9. Favicon
10. HTTP or HTTPS token in domain name
11. Request URL
12. URL of Anchor
13. Links in tags
14. SFH
15. Submitting to email
16. Abnormal URL
17. IFrame
18. Age of Domain
19. DNS Record
20. Web Traffic
21. Google Index
22. Statistical Reports

[1, 1, 1, 1, -1, 1, 1, -1, -1, 1, 1, -1, 1, 0, 1, -1, 1, -1, -1, -1, 1, 1]
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers
[Parallel(n_jobs=1)]: Done  10 out of  10 | elapsed:    0.0s finished
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers
[Parallel(n_jobs=1)]: Done  10 out of  10 | elapsed:    0.0s finished
Features= [[ 1  1  1  1 -1  1  1 -1 -1  1  1 -1  1  0  1 -1  1 -1 -1 -1  1  1]]
ed label is -  ['1']
The probability of this site being a phishing website is  [ 100  100  100  100
  100 -100  100 -100 -100 -100  100  100] %
SAFE
```

Figure 2. Result of the extension for a safe website

Figure3 shows the result of phishing website and the net result obtained was not safe by chrome extension



```
1. Having IP address
2. URL Length
3. URL Shortening service
4. Having @ symbol
5. Having double slash
6. Having dash symbol(Prefix Suffix)
7. Having multiple subdomains
8. SSL Final State
8. Domain Registration Length
9. Favicon
10. HTTP or HTTPS token in domain name
11. Request URL
12. URL of Anchor
13. Links in tags
14. SFH
15. Submitting to email
16. Abnormal URL
17. IFrame
18. Age of Domain
19. DNS Record
20. Web Traffic
21. Google Index
22. Statistical Reports

[1, -1, 1, -1, -1, 1, 0, -1, -1, 1, 1, -1, 1, 0, 1, -1, 1, -1, -1, -1, 1, 1]
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers
[Parallel(n_jobs=1)]: Done  10 out of  10 | elapsed:    0.0s finished
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers
[Parallel(n_jobs=1)]: Done  10 out of  10 | elapsed:    0.0s finished
Features= [[ 1 -1  1 -1 -1  1  0 -1 -1  1  1 -1  1  0  1 -1  1 -1 -1 -1  1  1]]
ed label is -  ['-1']
The probability of this site being a phishing website is  [ 100 -100  100 -100
  100 -100  100 -100 -100 -100  100  100] %
PHISHING
```

Figure 3.Result of the extension for a phishing

IV.CONCLUSION AND FUTURE SCOPE

In this paper we have proposed the development of Chrome Extension for identification and detection of phishing Websites using Random forest algorithm. This concept displays the safe way to use a website which will keep the user security. Otherwise the user might end up giving his valuable information to the phishers, which they can use for blackmailing or threatening to users.

The future scope of this idea is very vast. This will not allow the users to browse unsafe websites.This will make browsing more secure and safe in the coming future.

V. ACKNOWLEDGEMENT

We would like to thank our guide Prof. (Mrs.) , Keerti Kharatmol, Associate Professor, K. C College of Engineering & Management studies & Research, Thane for initiating us into this field of research and for providing me with the necessary guidance, great encouragement throughout the preparation of this paper. We record our deep indebtedness to them for their support. We take this opportunity to express our gratitude to the Staff members, Non-Teaching Staff members and Research Scholars of the Department of Computer Engineering, K. C College of Engineering & Management studies & Research, Thane, for their timely help and encouragement.

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] B. Wardman, G. Shukla and G. Warner, ;"Identifying vulnerable web-sites by analysis of common strings in phishing URLs,"; 2009 eCrime

Researchers Summit, Tacoma, WA, 2009, pp. 1-13.

[3] Xiang et al., "A Feature-Rich Machine Learning Framework for Detecting Phishing WebSites, ACM Transactions on Information and System Security "2011.

[4] Abdul Ghani Ali Ahamad and Nurul Amirah Abdulla" Real time detection of phishing websites "2013 IEEE

[5] A. Abunadi, O. Akanbi and A. Zainal, "Feature extraction process: A phishing detection approach," 2013 13th International Conference on Intellient Systems Design and Applications, Bangi, 2013, pp. 331-335. doi: 10.1109/ISDA.2013.6920759

[6] "UCI" machine learning repository data set" archive.ics.uci..edu 2017
Online available http://archive.uci.edu/ml/dataset/phishing+website