# Malicious URL's Detection using Machine Learning

Shreya Samir Labhsetwar[1], Tushar B. Kute[2]
[1]*Student, Dept.of Computer Engineering, S.P.P.U. University, Nashik, India*
[2]*Researcher, MITU Skillologies, Pune, India*

***Abstract*- Internet underpins a wide range of crimes, for example, spreading of Malwares and Misrepresentation of data usage. In spite of the fact that the exact inspirations driving these plans may vary, the shared factor lies in the way that clueless clients visit their locales. These visits can be driven by email, web list items or connections from other website pages. In all cases, in any case, the client is required to make some move, for example, tapping on an ideal Uniform Resource Locator (URL). In this paper, we address the identification of pernicious URL's using various machine learning algorithms specifically Support Vector Machines, Decision Trees, Random Forest and k-Nearest Neighbours and logistic regression. Besides, we embraced an open dataset including various URLs (examples) and their corresponding labels. Specifically, Random Forest and Support Vector Machines achieve the most astounding precision. The phishing issue is tremendous and there does not exist just a single answer to limit all vulnerabilities viably, hence the systems are actualized and implemented.**

**Index terms- Machine Learning, URL, Support Vector Machine, Decision Trees, Random Forest and k-Nearest Neighbours and logistic regression**

## I. INTRODUCTION

Phishing is the false endeavour to get delicate data, for example, usernames, passwords and charge card subtleties by camouflaging oneself as a reliable substance in an electronic communication. Typically completed by email parodying or texting, it frequently guides clients to enter individual data at a fake site which matches the look and feel of the authentic site.

Phishing is a social designing assault that goes for misusing the shortcoming found in the framework at the client's end. For instance, a framework might be in fact secure enough for secret key burglary yet the uninformed client may release his/her secret word when the assailant sends a bogus update secret phrase demand through fashioned (phished) site. For tending to this issue, a layer of insurance must be included the client side to address this issue.

Phishing is a case of social engineering systems being utilized to betray clients. Clients are frequently tricked by correspondences indicating to be from confided in gatherings, for example, social sites, sell off destinations, banks, online instalment processors or IT chairmen. Endeavours to manage phishing occurrences incorporate enactment, client preparing, open mindfulness, and specialized safety efforts.

A phishing assault is the point at which a criminal sends an email or the URL professing to be a person or thing he's not, so as to get delicate data out of the person in question. The individual as to his/her interest or a feeling of urgency, they enter the subtitles, similar to a username, secret phrase, or Visa number, they are probably going to assent. The ongoing case of a phishing trick that focused around billion's clients around the world.

The Fig.1 looks exactly like an Amazon sing-in form, but the URL is slightly changed. Filling the details here the attackers would get full access to the victim's Amazon account. The transaction fraud can take place just by accessing the client's details. Microsoft Outlook misrepresentation is the second-most focused on and Google drive being the third. Different targets are Facebook, bank logins and Paytm, PayPal and so on. So as to recognize these malicious sites, the web security network has created boycotting services. These boycotts are thus created by a system including manual announcing, honeypots, and web crawlers joined with webpage investigation heuristics. While URL blacklisting has been viable somewhat, it is fairly simple for the attacker to hack the entered URL.

Fig.1. Phishing Scam URL

Definitely, numerous malicious sites are not boycotted either on the grounds that they are excessively recent or were never or inaccurately assessed. A few investigations in the writing tackle this issue from a Machine Learning point of view. That is, they assemble a rundown of URLs that have been classified as either malignant or favourable and portray every URL through a lot of traits. Classification calculations are then expected to become familiar with the limit between the choice classes. The paper involves identification of pernicious URL's using a Machine learning approach. The rest of this paper is organized as follows. Section II describes the related work. III outlines the proposed work of the system. The experimental results are put forth in Section IV. Concluding remarks are given in Section V.

## II. RELATED WORK

Numerous specialists have recently been done in this field of phishing recognition. We have assembled the data from different such works and have significantly audited them which has helped us in inspiring our very own philosophies during the time spent making an increasingly secure and exact framework

### A. Blacklist Approach and Whitelist Approach

In [13], Pawan Prakash, Manish Kumar, Ramana Rao Kompella, Minaxi Gupta (2010) proposed a predictive blacklist approach to detect phishing websites. It identified new phishing URL using heuristics and by using an appropriate matching algorithm. Heuristics created new URL's by combining parts of the known phished websites from the available blacklist. The matching algorithm then calculates the score of URLs. If this score is more than a given threshold value it flags this website as phishing website. The score was evaluated by matching various parts of the URL against the URL available in the blacklist. In [14], Jung Min Kang and DoHoon Lee described approach which detected phishing based on user's online activities. This method maintained a white list as a part of users' profile. This profile was dynamically updated whenever a user visited any website. An engine used here identified a website by evaluating a score and then comparing it with a threshold score. The score was calculated from the entries available in the user profile and details of the current website.

### B. Heuristic Approach

In [7], Aaron Blum, Brad Wardman, Thamar Solorio proposed a work which focused on the exploration of surface level features from URLs to train a confidence 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA) weighted learning algorithm. The idea is to restrict the source of possible features to the character string of the URL and avoid having the vulnerability of extracting host-based information. Every URL is displayed as a vector of binary feature. These vectors are fed to the online algorithm where at time of testing, previously unseen URLs in the binary feature vector is then mapped to it. The learner continues this new vector and output into the final result, either phish or non-phish. In [15], Guang Xiang, Jason Hong, Carolyn P. Rose, Lorrie Cranor proposed CANTINA+, a comprehensive feature-based approach in the literature including eight novel features, which exploits the HTML Document Object Model (DOM), search engines and third-party services with machine learning techniques to detect phish. Also, two other filters are designed in it to help reduce FP and achieve good runtime speedup. The first is a near-duplicate phish detector that uses hashing to catch highly similar phish. The second is a login form filter, which directly classifies webpages with no identified login form as legitimate. In [8], Joby James, Sandhya L, Ciza Thomas proposed a work which with the combined help of blacklisting approach and the Host based Analysis applied certain classifiers which can be used to help detect and take down various phishing sites. The host based, popularity based and lexical based feature extractions are applied to form a database of feature values. The database is knowledge mined using different machine learning methods. After evaluating the classifiers, a particular classifier was selected and was implemented in MATLAB. In [9], APWGM published a case study citing the importance of the WHO is tool and how invaluable it has been for the rapid phishing site shutdown over the past few years all around the globe.

### C. Visual Similarity Approach

In [2], A. Mishra and B. B. Gupta presented a hybrid solution based on URL and CSS matching. In this approach it can detect embedded noise contents like an image in a web page which is used to sustain the visual similarity in the webpage. They used the technique used in [3] by Jian Mao, Pei Li, Kun Li, Tao Wei, and Zhenkai Liang to compare the CSS similarity and used it in their technique. The different types of visual features are - text content and text features. Text features are like font colour, font size, background colour, font family and so forth. This approach matches the visual features of different websites because the attacker copies the page content from the actual website. In [5] Matthew Dunlop, Stephen Groat, and David Shelly proposed a browser-based plug in called goldphish to identify phishing websites. It uses the website logos to identify the fake website. The attacker can use the real logo of the target website to trap the internet users.

## III. PROPOSED SYSTEM

Out of the previous works many algorithms were not been implemented. Here we propose a system which will give us maximum accuracies for every Machine learning algorithm.

A) Data Collection

In this security evaluation, there are few datasets which can be available to use in the machine learning. Here we used a dataset of around 31800 entries. Where the URL's were been classified on basis of their labels as a "Good" URL or a "Bad" URL respectively. This dataset helped to train the Machine for obtaining accuracy over various machine learning algorithm's and one could categorize the best fitted algorithm for the system. The input as well as output of the Dataset is in the String format.

B) Pre-processing

a. Remove Duplicate Data

Data duplicity and data redundancy is one of the problems causing aspect of the system. Hence to overcome this barrier we remove the duplicate entries in the dataset so that the size of the dataset remains small and helps in the classification of data through Machine Learning.

b. Data Normalization

It is another technique which sorts the records in a legitimate way. By utilizing normalization every one of the records are changed to its score esteem or in loads. By utilizing normalization process the records highlight is re-focused and rescaled. By re-focusing and rescaling those highlights hold to zero mean and unit change.

C) Classification

a. Support Vector Machines

Support Vector Machines depend on the idea of choice planes that characterize choice limits. A decision plane is one that separates between a set of objects having different class memberships. A schematic model is appeared in the representation below. In this model, the items have a place either with class GREEN or RED. The separating line characterizes a limit on the correct side of which all items are GREEN and to one side of which all articles are RED. Any new object falling to the right is named, i.e., grouped, as GREEN (or named RED should it tumble to one side of the isolating line). Support Vector Machines depend on the idea of choice planes that characterize choice limits. A choice plane is one that isolates between a lot of articles having diverse class memberships.

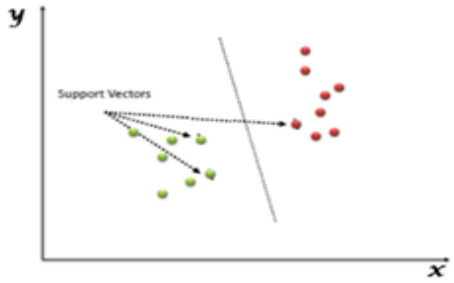


Fig. 2. Support Vector Machine

b. Random Forest

Random forests or random decision forests are a procuring system for gathering, backslide and various errands that works by structure an enormous number of decision trees at getting ready time and yielding the class that is the strategy for the classes or mean forecast of the individual trees. Random decision forests right for choice trees propensity for over fitting to their preparation set. Random forest classifier makes a lot of choice trees from randomly chosen subset of preparing set. It at that point totals

the votes from various choice trees to choose the last class of the test object.

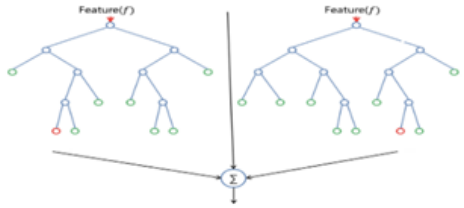


Fig. 3. Random Forest

c. Decision Tree

A Decision Tree is a Supervised Machine Learning algorithm which looks like an inverted tree, wherein each node represents a predictor variable(feature), the link between the nodes reparents a Decision and each leaf node represents an outcome. A decision tree represents a procedure for classifying categorical data based on their attributes. It is also efficient for processing large amount of data, so is often used in data mining application. The construction of decision tree does not require any domain knowledge or parameter setting, and therefore appropriate for exploratory knowledge discovery. Their representation of acquired knowledge in tree form is intuitive and easy to assimilate by humans.

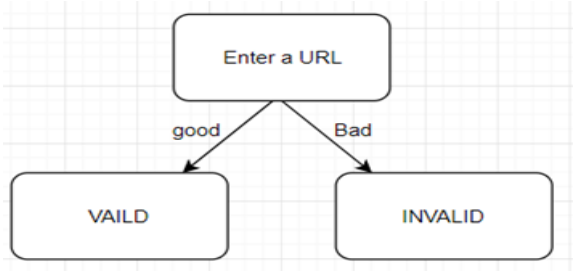


Fig. 4. Decision Tree

d. Logistic Regression

Logistic regression is a characterization calculation used to appoint perceptions to a discrete arrangement of classes. Dissimilar to straight regression which yields nonstop number qualities, logistic regression changes its yield utilizing the logistic sigmoid capacity to restore a likelihood esteem which would then be able to be mapped to at least two discrete classes. Logistic regression is a factual strategy for investigating a dataset in which there are at least one autonomous factor that decide a result. The result is estimated with a dichotomous variable (where there are just two potential results). In logistic regression, the reliant variable is double or dichotomous, for example it just contains information coded as 1 or 0.

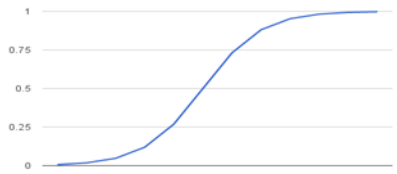


Fig. 5. Logistic Regression

e. k-Nearest Neighbours

K-Nearest Neighbours is a standout amongst the most fundamental yet basic grouping calculations in Machine Learning. It has a place with the directed learning space and finds extraordinary application in example acknowledgment, information mining and interruption recognition. It is broadly expendable, in actuality, situations since it is non-parametric, which means, it doesn't make any basic suspicions about the conveyance of information. In k-NN characterization, the yield is a class enrolment. An item is ordered by a majority vote of its neighbours, with the article being appointed to the class most regular among its k nearest neighbours. In the event that k = 1, at that point the article is just allotted to the class of that solitary nearest neighbour. In k-NN relapse, the yield is the property estimation for the item. This worth is the normal of the estimations of k nearest neighbours.

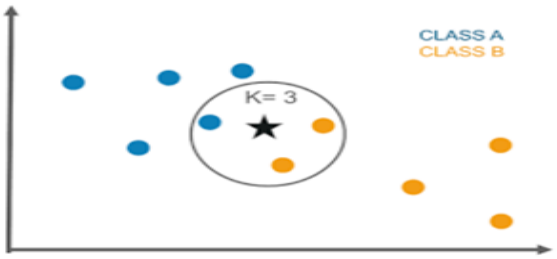


Fig. 6. k-Nearest Neighbours

## IV. EXPERIMENTAL RESULT

Given the enormous measure of information accessible, a non-stratified free irregular example with equivalent likelihood is taken from a lot of line numbers somewhere in the range of 1 and 20,000. For every one of the informational indexes, the URLs comparing to the column numbers in the example, would be added to the preparation set. This outcome all URLs utilized for learning the models. The test set

comprises of unique informational collections, one for every day, containing all URLs that have not been utilized for preparing. All numbers from the trial results have been accomplished by computing the forecast exactness for every one of the distinctive test sets. So as to assess the models, we utilize three unique measurements. Accuracy, Eq. 1, can be viewed as the general achievement rate of the technique regarding expectations. Time, Eq. 2, is stated as the time required for all the algorithms to run to completion.

$$accuracy = TP + TN/TP + TN + FP + FN \quad (1)$$

Time=End Time-Start Time (2)

- TP: number of true positives.
- TN: number of true negatives.
- FP: number of false positives.
- FN: number of false negatives.
- End time= time at the end of execution.
- Start time=Time at the start of execution.

Malicious websites are been identified by the help of various algorithms used to test the results. Support Vector Machine give the maximum accuracy followed by Random Forest algorithm, whereas k-nearest neighbours give the least accuracy. following Table shows the accuracies of the algorithm and time taken by the Algorithm to complete.

| SR.NO | Algorithm Name | Accuracy | Time |
|---|---|---|---|
| 1. | Support Vector Machine | 96.95% | 62.5s |
| 2. | Random Forest | 94.57% | 6.59s |
| 3. | Decision Tree | 94.27% | 3.47s |
| 4. | Logistic Regression | 92.38% | 0.16s |
| 5. | k-Nearest Neighbours | 91.95% | 0.02s |

TABLE 1

Vector Machine give the maximum accuracy followed by Random Forest algorithm, whereas k-nearest neighbours give the least accuracy. following Table shows the accuracies of the algorithm and time taken by the Algorithm to complete.

TABLE 1

The following Fig.7 shows Graphical representation of the Algorithm vs Accuracy and Fig.8 shows Graphical representation of Time vs Algorithm

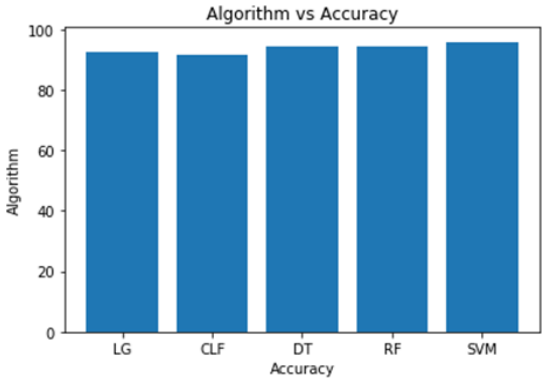


Fig. 7. Algorithm vs Accuracy

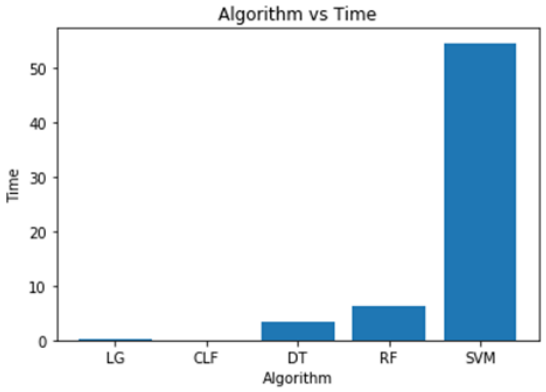


Fig. 8. Algorithm vs Time

The final GUI output of the system will be seen as shown in the Fig.9


Fig. 9. GUI Outlook

## V. CONCLUSION

The proposed framework empowers the web clients to have a sheltered perusing and safe exchanges. Its causes clients to spare their significant private subtleties that ought not be spilled. Giving our proposed framework to clients as expansion makes the procedure of delivering our framework a lot simpler. The outcomes focus to the productivity that can be accomplished utilizing the crossover arrangement of heuristic highlights, visual highlights

and boycott and white list approach and sustaining these highlights to Machine Learning. calculations. A specific challenge in this area is that crooks are continually making new systems to counter our barrier measures. To prevail in this specific circumstance, we need calculations that consistently adjust to new models and highlights of phishing URL's. What's more, along these lines we utilize internet learning algorithms. This new framework can be intended to benefit most extreme precision. Utilizing various methodologies inside and out will improve the exactness of the framework, giving a productive assurance framework. The downside of this framework is distinguishing of some negligible false positive and false negative outcomes. These disadvantages can be dispensed with by acquainting a lot more extravagant element with feed to the Machine Learning.

REFERENCES


[1] Ankit Kumar Jain and B. B. Gupta, "Phishing Detection Analysis of Visual Similarity Based Approaches", Hindawi 2017.

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

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

[4] Eric Medvet, Engin Kirda and Christopher Kruegel, "VisualSimilarity-Based Phishing Detection", ACM 2015.

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

[6] Haijun Zhang, Gang Liu, Tommy W. S. Chow, and Wenyin Liu, "Textual and Visual Content-Based Anti-Phishing a Bayesian Approach", IEEE 2011

[7] Aaron Blum, Brad Wardman, Thamar Solorio, Gary Warner; "Lexical Feature Based Phishing URL Detection Using Online Learning", Department of Computer and Information Sciences the University of Alabama at Birmingham, Alabama, 2016

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

[9] The Anti-Phishing Working Group, DNS Policy Committee;" Issues in Using DNS Who is Data for Phishing Site Take Down", The AntiPhishing Working Group Memorandum, 2011.

[10] Guang Xiang, Jason Hong, Carolyn P. Rose, Lorrie Cranor,"CANTINA+: A Feature-rich Machine Learning Framework for Detecting Phishing Web Sites", School of Computer Science Carnegie Mellon University, ACM Society of computing Journal, 2015.

[11] Joby James,SandhyaL,Ciza Thomas "Detection of phishing websites using Machine learning techniques", 2013 International Conference on Control Communication and Computing (ICCC).

[12] Mohsen Sharifi and Seyed Hossein Siadati "A Phishing Sites Blacklist Generator".

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

[14] Y. Zhang, J. I. Hong, and L. F. Cranor. Cantina: a content-based approach to detecting phishing web sites. In WWW '07: Proceedings of the 16th international conference on World Wide Web, pages 639– 648, New York, NY, USA, 2007. AC

[15] Frank Vanhoenshoven, Gonzalo N´apoles, Rafael Falcon†, Koen Vanhoof and Mario K¨oppen: Detecting Malicious URLs using Machine Learning Techniques.

[16] ] Vaibhav Patil, Pritesh Thakkar, Chirag Shah: Detection and Prevention of Phishing Websites using Machine Learning Approach