2020 International Conference on Computer Communication and Informatics (ICCCI -2020), Jan. 22-24, 2020, Coimbatore, INDIA

Phishing Website Classification and Detection Using Machine Learning

Jitendra Kumar
Centre for Development of Advanced Computing & NIT Trichy
Bengaluru,India
jitendra@cdac.in

A. Santhanavijayan
National Institute of Technology,Trichy
Trichy, India
vijayana@nitt.edu

B. Janet
National Institute of Technology,Trichy
Trichy, India
janet@nitt.edu

Balaji Rajendran
Centre for Development of Advanced Computing
Bengaluru,India
balaji@cdac.in

Bindhumadhava BS
Centre for Development of Advanced Computing
Bengaluru,India
bindhu@cdac.in

***Abstract*— The phishing website has evolved as a major cybersecurity threat in recent times. The phishing websites host spam, malware, ransomware, drive-by exploits, etc. A phishing website many a time look-alike a very popular website and lure an unsuspecting user to fall victim to the trap. The victim of the scams incurs a monetary loss, loss of private information and loss of reputation. Hence, it is imperative to find a solution that could mitigate such security threats in a timely manner. Traditionally, the detection of phishing websites is done using blacklists. There are many popular websites which host a list of blacklisted websites, e. g. PhisTank. The blacklisting technique lack in two aspects, blacklists might not be exhaustive and do not detect a newly generated phishing website. In recent times machine learning techniques have been used in the classification and detection of phishing websites. In, this paper we have compared different machine learning techniques for the phishing URL classification task and achieved the highest accuracy of 98% for Naïve Bayes Classifier with a precision=1, recall = .95 and F1-Score= .97.**

Keywords—domain name, lexical analysis of URL, malicious URL classification and detection, phishing website classification and detection, phishing attacks

I. Introduction

Phishing is a form type of a cybersecurity attack where an attacker gains control on sensitive website user accounts by learning sensitive information such as login credentials, credit card information by sending a malicious URL in email or masquerading as a reputable person in email or through other communication channels. The victim receives a message from known contacts, persons, entities or organizations and looks very much genuine in its appeal. The received message might contain malicious links, software that might target the user computer or the malicious link might direct the user to some forged website which is similar in look and feel of a popular website, further victim might be tricked to divulge his personal information e.g. credit card information, login and password details and other sensitive information like account id details etc. Phishing is the most popular type of cybersecurity attack and very common among the attackers. Phishing attacks are generally easy as most of the victims are not well aware of the intricacies about the web applications and computer networks and its technologies and are easy prey for getting tricked or spoofed. It is very easy to phishing unsuspecting users using forged websites and luring them for clicking the websites for some prize and offers than targeting the computer defense system. The malicious website is designed in such a way that it has a similar look and feel and it appears very genuine in its appearance as it contains the organization's logos and other copyrighted contents. As many users unwittingly clicking the phishing websites URLs and this results in huge financial and loss of reputation to the person and to the concerned organization. The phishing email might contain a PDF or Word document as a malicious attachment. The malicious document installs the malware in the system once opened. The cybercriminals use compromised email accounts for sending phishing emails as it is much easier than changing headers of the SMTP text message.

A. Phishing Attacks involving URLs

- Deceptive Phishing [16]: This is the most common type of phishing attack wherein a cybercriminal impersonates a known popular entity, domain or organization and attempt to steal sensitive private information from the victim such as login, password, bank account detail, credit card detail, etc. This type of attack lacks sophistication as it does not have personalization and customization for the individuals. For an example, emails containing Phishing URL is disseminated in bulk to large users as a volume of mail is very high the cybercriminal would expect that many users will open the emails and visit the malicious URLs or open the infected attachments. The idea behind this type of phishing is deception and impersonation. This type of email mostly creates panic and urgency for the victims to divulge sensitive information. The email subject will be such that it might create urgency such as "Your account has been hacked, change your password immediately!", "Your bill is overdue-pay immediately of pay fine!" or other similar messages, once a user open such messages or visit the URLs the damage is done.

- Spear Phishing [16]: In this type of phishing emails containing malicious URLs contain an abundance of personalization information about the prospective victim. The email might contain the name, company name, designation or his friends, colleagues and other social information of the recipient. The proliferation of the company website, personal website and social

978-1-7281-4514-3/20/$31.00 ©2020 IEEE

media enables cybercriminals to get such details and assist them in forging a very convincing email.

- Whale Phishing [16]: This type of phishing targets business leaders such as CEO of top-level management employees to spear phish a "whale", here top-level executive such as CEO. The main aim of this type of phishing is to gather confidential information from the CEO and impersonate as CEO. This attack can render maximum damage to company financial prospects, market value, and reputation.
- URL Phishing [16]: The scammer or cyber-criminal uses a URL link to infect the target. The people are social and will be very eager to click the link for accepting the friend requests and maybe even ready to share their contact such as email. Most of the time email or SMS works, hidden links, tiny URL, misspelled URL works as a conduit for such an attack.

Some security threat intelligence companies which detect and publish malicious web URL or IPs and provide blacklist database, hence helping in preventing the others from the harmful effect of phishing.

II. Related Work

Samuel Marchal et. al., [1] introduces a system PhishStorm and uses relatedness of the parts of URL as a metrics, the word intra-URL relatedness measures the relationship among the words blended into a URL and particularly into the part of the URL that can be freely defined and the registered domain and claim 97 % classification accuracy. Mohammed Nazim Feroz et. al., [2] described the usage of lexical, host-based features, cluster ID as a feature and URL reputation features for the classification problem and achieves 93-98% accuracy. Mahdieh Zabihimayvan et. al. [3] used a technique Fuzzy Rough Set (FRS) to select the most effective features, the author further claimed the maximum F-measure gained by FRS feature selection is 95% using Random Forest classification. Moitrayee Chatterjee et. al., [4] proposed a model based on deep reinforcement learning to model and detect malicious URLs. The proposed model can adapt to the dynamic behaviour of the phishing websites and thus learn the features associated with phishing website detection. Chun-Ying Huang et. al., [5] highlighted the evolving nature of the phishing URL as most of the existing solutions detect mimicked phishing pages by either text-based features or visual similarities of webpages and it can be easily bypassed and proposed a technique to identify the real domain name of a visiting webpage based on signatures created for web sites, site signatures, including distinctive texts and images, can be generated by analysing common parts from pages of a website. The authors claimed that the method achieves high accuracy and low error rates. Aaron Blum et. al., [6] explored the possibility of utilizing confidence weighted classification combined with content-based phishing URL detection to produce a dynamic and extensible system for detection of present and emerging types of phishing domains, and authors further claims the system can detect emerging threats and can provide an increased protection against zero-hour threats, unlike traditional blacklisting techniques which function reactively. Mohammed Al-Janabi et. al., [7] discussed the threat of malicious URL in social networks and the requirement of automated methods to detect and eliminate such content and used random forest classification with a combination of features derived from a range of sources. The authors also demonstrated that a random forest model without any tuning and feature selection produced a recall value of 0.89 and after applying parameter tuning and feature selection the classifier performance improved with 0.92 in the recall. Erzhou Zhu et. al., [8] highlighted the problem of overfitting in the phishing classifier and suggested FS-NN, an effective phishing website detection model based on the optimal feature and neural network. Ankesh Anand et. al., [9] mentioned the class imbalance problem existing in the phishing website detection. As most of the time the benign URLs outnumbers the malicious URLs the authors suggested oversampling of the minority class and generate synthetic URLs and made them a part of the training set. Justin Ma et. al., [10] used lexical and host-based features of their URLs and online classifier for the detection of a phishing website. Youness Mourtaji et. al., [11] explored black-list based, Lexical based, content-based and security and identity-based methods and constructed a model combined with machine learning classifiers for phishing website detection. Akihito Nakamura [12] et. al., emphasized compared phishing mitigation techniques, such as blacklist, heuristics, visual similarity, and machine learning and concluded that these techniques have limitations in dealing with zero-hour attacks and proactive detection of phishing websites. The authors proposed suspicious domain names generation and to predicts likely phishing web sites from the given legitimate brand domain name and scores and judges suspects by calculating various indexes to detect phishing websites.

III. Lexical Structure of a URL

The lexical structure of a URL as shown in Fig.1 could reveal the hidden information about a URL. A URL starts with a protocol name such as HTTP or HTTPs. The FQDN (fully qualified domain name) is the complete domain name of the server hosting the web site, which later translates into an IP address using DNS servers. The domain name consists of a second-level domain (SLD) which is suffixed with the top-level domain (TLD) to which it is registered. The domain name is a registered name that is registered with a domain registrar and unique across the Internet.

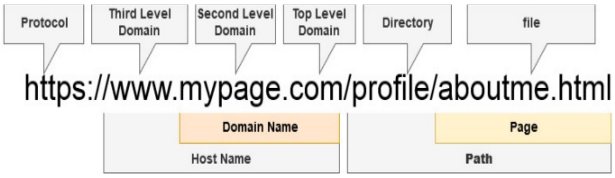

Fig. 1. Lexical structure of a URL

The domain name portion is a registered name with Domain name Registrar. The hostname consists of a subdomain name and domain name. A phisher can easily modify the subdomain name portion and can associate it with any value. The URL may also contain path and file which can also be easily modified by the phisher if he wishes so. The subdomain name and path of a URL can be controlled by the phisher. An attacker can register a domain only once and this domain is identified as fraudulent, it is easy to prevent the user from visiting such domain. The problem lies with the variable parts of the URL i.e. subdomain and path. This is the reason the cybersecurity experts, designers and users struggle to provide a feasible solution to mitigate URL phishing attacks.

Let’s consider a phishing URL given below

“http://amazon.com-verification-accounts.darotob.com/Sign-in/5b60fcc60b36d1c3d”

The lexical analysis of the above URL reveals parts as shown in Fig. 2. The attackers obfuscate the URL in such a way that the actual domain name might not be easily revealed to the normal user and it will be nested deep inside the URL, e.g. in the above URL actual Domain name is “darotob.com”, but at first glance, it might look like “amazon.com”. For a normal user not aware of the intricacies of the web technologies, “amazon.com” at the beginning of the URL, can provide the assurance and trust about the website and he might tempt to connect it and might share his confidential and sensitive information to the fraudulent website. This is a very common fraudulent technique used by the attackers. The cybercriminals can employ a technique, Cybersquatting [18] (another name is domain squatting), where he registers a domain name with a bad intention to make a profit from the goodwill of a brand name or trademark belonging to other companies or organizations. For example, the name of a renowned brand is “greatcompany.com”, the phisher can register “greatcompany.net”, “greatcompany.org”, “greatcompany.biz” etc. for fraud. Cybercriminals can also adopt Typosquatting [18] (also known as URL hijacking), which a varying form of cybersquatting, which relies on typographical mistakes unknowingly made by the users while typing the website address into a web browser, such typographical errors are hard to notice while quick casual reading. The URLs created by Typosquatting look very similar to the well-known trusted domains. The user occasionally might type the incorrect web address or click a link might look very similar to the trusted domains and this might lead him to visit a phishing website owned by a phisher. A very famous example of Typosquatting is “goggle.com”, which is a phishing website and extremely dangerous. Another example of Typosquatting is “yutube.com” which is a Typosquatting equivalent of “youtube.com”.

Protocol	http://
Domain Name	darotob.com
Path	/Sign-in/5b60fcc60b36d1c3d
Sub Domain	com-verification-accounts
Sub Domain	Amazon

Fig. 2. Different parts of the URL

IV. DATA SET FOR THE EXPERIMENT

We used the dataset [17] for our experiment. We found the dataset to be imbalanced as it contained more than 1 million URLs, but out of which it contained only about 57000 to 60000 phishing/malicious URLs and rest were benign URLs. Hence, we created two different sets, the first set containing about 57000 malicious URLs and the second set containing about 57000 benign URLs from the given dataset. Further, we created a final dataset of 100000 URLs for training after randomly mixing URLs from both sets. This strategy solved the data imbalanced problem and solved the issue of biased training and helped in bringing randomness in the dataset and further solved the issues related to variance or overfitting.

V. FEATURE ENGINEERING

The machine learning techniques have been used in our experiment, the lexical structure of a URL has been discussed in Section III which highlighted how an attacker can morph a URL for the phishing purpose. In this section, we will discuss data pre-processing techniques for feature selection and feature extraction for the phishing URL detection problem. The features which have generally been used in phishing classification problem can broadly be grouped in four categories as below

- URL lexical structure-based features
- Domain name related features
- Page based features

The feature extraction is the process of finding the most significant features for the given problem. The feature extractor first analyses the URL of the website to find the URL based features. Based on the distinctive points of a URL, we have extracted the following URL-based features.

- Length of the URL
- Number of dots in the URL
- Number of hyphens in the domain
- Presence of security sensitive words in the URL
- Length of the directories in the path of the URL
- Number of sub directories in the path of the URL
- Presence or absence of IP in the URL
- Count of tokens in the path
- Largest path token length
- Average path token length
- Length of the file
- Total number of dots in file
- Total number of delimiters in file
- Length of arguments
- Number of arguments
- Length of largest argument value
- Maximum number of delimiters in arguments

Domain based features extracted:

- Domain length
- Count of tokens in the domain
- Length of largest token of largest token in the domain
- Average domain token length
- Suspicious top-level domain

Page related features extracted:

- Age of the domain in months

- Domain expiry age in months
- Domain updating age in day
- zip code of the address of domain holder

VI. Expermental Setting

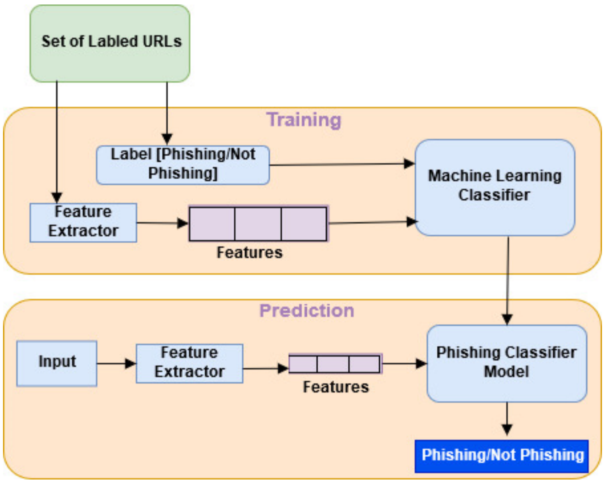

Fig. 3. Block diagram for the proposed work

Fig 3 depicts the block diagram for the overall approach, the training phase of the classifier takes a labeled dataset of phishing and non-phishing URLs and returns a trained model. The trained model is used for prediction for new instances.

Pseudo code for Phishing Classification and Detection

/* "phishing_dataset" contains an equal number of phishing and benign URLs, and "test_url" is a URL having an unknown status about being Phishing or Benign */

```
Create_Model(phishing_dataset)
{
  /* Create the feature matrix */
  phish_matrix=create_vector(phishing_dataset)
  /*Split the dataset into training dataset and testing dataset*/
  X_train,X_test,y_train,y_test=split (phish_matrix, test_size=0.3)
  /* use the ML classifier for training such as Logistic Regresion, Naïve Bayes and Random Forest etc.*/
  ml_classifier=ML_Classifier()
  classifier_model=ml_classifier.fit (X_train,y_train)
  prediction= classifier_model.predict(X_test,y_test)
  accuracy=classifier_accuracy(y_test, prediction)
  /* dump model to a file in disk */
  dump (classifier_model, file)
  }
  Phishing_Detection (test_url)
  {
  /* retrieve model from the file */
  classifier_model =retrieve_model(file)
  test_url_vector= create_vector(test_url)
  predict_new_url= classifier_model,predict(test_url_vector)
  /*
   returns whether the URL is a malicious / benign
  */
  return predict_new_url
  }
```

We trained different classifiers e.g. Logistic Regression, Naïve Bayes Classifier, Random Forest, Decision Tree and KNearest Neighbor.

VII. Results Analysis

The dataset used for the training is obtained from Internet sources [17] which is further a manual collection of URLs from [13] [14] and [15]. We created a dataset containing an equal number of labeled phishing and non-phishing websites as mentioned in Section IV. We further performed random mixing of the dataset to remove any order in the dataset. The dataset contained 100000 examples and is further split in the ratio 7:3 for training and testing. We trained the classifiers based on the features extracted from the lexical structure of the URL as mentioned in Section V. The performance evaluation of a classifier is done based on the metrics shown in Fig 4.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\textit{Precision} = \frac{TP}{TP + FP}$$

$$\textit{Recall} = \frac{TP}{TP + FN}$$

$$\textit{F - measure} = \frac{2 * \text{Recall} * \text{Precision}}{\text{Recall} + \text{Precision}}$$

Fig. 4. Performance evaluation metrics

The experimental result of performance metrics of the different classifiers for the Phishing URL Classification task is as shown in the table in Fig. 5. We used Scikit-Learn [19] in our experiment for classification and predictive analysis.

Classifiers	**Precision**	**Recall**	**F1-Score**	**Accuracy in %**
Logistic Regression	1	.96	.98	97.7
Random Forest	1	.96	.98	98.03
Gaussian Naïve Bayes	1	.95	.97	97.18

Decision Tree	1	.96	.98	98.02
KNearest Neighbor	.99	.97	.98	97.99

Fig. 5. Performance metrics results of different classifiers

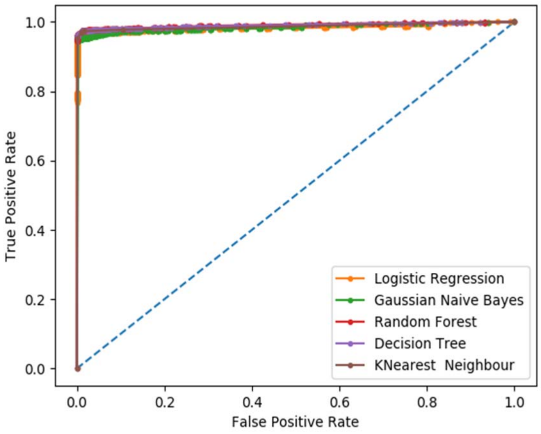

Fig. 6. ROC(Reciever Operating Curve) plot of different classifiers

CLASSIFIERS	AUC (AREA UNDER CURVE)
Logistic Regression	0.983
Random Forest	0.983
Gaussian Naïve Bayes	0.991
Decision Tree	0.989
KNearest Neighbor	0.987

Fig. 7. Area under the curve (AUC) of different classifiers

The performance metrics data as shown in Fig.5, Fig.6, and Fig. 7 shows that all used classifiers in the experiment are suitable for the Phishing URL detection tasks. The classifiers Random Forest and Gaussian Naïve Bayes classifiers result in better accuracies of about 98 %. Fig. 6 shows that the AUC (area under the curve) for all the classifiers for the Phishing URL classification task is almost the same but Gaussian Naïve Bayes has the highest numerical value of .991, as shown in Fig. 7. Hence, for the Phishing URL classification, all the used classifiers in the experiment perform well as AUC (area under the ROC curve) is almost same for all classifiers, but particularly Naïve Bayes Classifiers is more suitable as it has highest AUC value in all the used classifiers in our experiment.

VIII. CONCLUSION

In this paper, we have explored how well to classify phishing URLs from the given set of URLs containing benign and phishing URLs. We have also discussed the randomization of the dataset, feature engineering, feature extraction using lexical analysis host-based features and statistical analysis. We have also used different classifiers for the comparative study and found that the findings are almost consistent across the different classifiers. We also observed dataset randomization yielded a great optimization and the accuracy of the classifier improved significantly. We have adopted a simple approach to extract the features from the URLs using simple regular expressions. There could be more features that can be experimented and that might lead to improving further the accuracy of the system. The dataset used in this paper contains the URLs list which may be a little old, hence regular continuous training along with a new dataset would enhance the model accuracy and performance significantly. In our experiment we have not used the content-based features as the main problem with the content-based strategy for detecting phishing URLs is the non-availability of phishing web-sites and the life span of the phishing website is small, and it is difficult to train an ML classifier based on its content-based features. In the future, we would like to incorporate a rule-based prediction based on the content analysis of a URL. Hence, the combination of classification based lexical analyzer along with a rule-based URL content analyzer for phishing URL detection would provide a comprehensive solution.

REFERENCES

[1] Samuel Marchal, Jérôme François, Radu State, and Thomas Engel, “PhishStorm: Detecting Phishing With Streaming Analytics,” IEEE Transactions on Network and Service Management, vol. 11 , issue: 4 , pp. 458-471, December 2014

[2] Mohammed Nazim Feroz,Susan Mengel, “Phishing URL Detection Using URL Ranking,” IEEE International Congress on Big Data, July 2015

[3] Mahdieh Zabihimayvan, Derek Doran, “Fuzzy Rough Set Feature Selection to Enhance Phishing Attack Detection,” International Conference on Fuzzy Systems (FUZZ-IEEE), New Orleans, LA, USA, June 2019

[4] Moitrayee Chatterjee,Akbar-Siami Namin, “Detecting Phishing Websites through Deep Reinforcement Learning,” IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC), July 2019

[5] Chun-Ying Huang,Shang-Pin Ma,Wei-Lin Yeh,Chia-Yi Lin,Chien-Tsung Liu, “Mitigate web phishing using site signatures,” TENCON 2010-2010 IEEE Region 10 Conference, January 2011

[6] Aaron Blum,Brad Wardman,Thamar Solorio,Gary Warner, “Lexical feature based phishing URL detection using online learning,” 3rd ACM workshop on Artificial intelligence and security, Chicago, Illinois, USA, pp. 54-60, August 2010

[7] Mohammed Al-Janabi,Ed de Quincey,Peter Andras, “Using supervised machine learning algorithms to detect suspicious URLs in online social networks,” IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017, Sydney, Australia, pp. 1104-1111, July 2010

[8] Erzhou Zhu,Yuyang Chen,Chengcheng Ye,Xuejun Li,Feng Liu, “OFS-NN:An Effective Phishing Websites Detection Model Based on Optimal Feature Selection and Neural Network,” IEEE Access(Volume:7), pp. 73271-73284, June 2019

[9] Ankesh Anand,Kshitij Gorde,Joel Ruben Antony Moniz,Noseong Park,Tanmoy Chakraborty,Bei-Tseng Chu, “Phishing URL Detection with Oversampling based on Text Generative Adversarial Networks,” IEEE International Conference on Big Data (Big Data), December 2018

[10] Justin Ma,Lawrence K. Saul,Stefan Savage,Geoffrey M. Voelker, “Learning to detect malicious URLs,” ACM Transactions on Intelligent Systems and Technology (TIST) archive Volume 2 Issue 3, April 2011

[11] Youness Mourtaji,Mohammed Bouhorma,Alghazzawi, “Perception of a new framework for detecting phishing web pages,” Mediterranean Symposium on Smart City Application Article No. 11, Tangier, Morocco, October 2017

[12] Akihito Nakamura,Fuma Dobashi, “Proactive Phishing Sites Detection,” WI '19 IEEE/WIC/ACM International Conference on Web Intelligence), pp. 443-448, October 2019

[13] https://www.phishtank.com/developer_info.php, [Online]. Available: https://www.phishtank.com/developer_info.php [Accessed: 27- September- 2019].

[14] https://openphish.com/, [Online]. Available: https://openphish.com/ [Accessed: 27- September- 2019].

[15] https://majestic.com/reports/majestic-million [Online]. Available: https://majestic.com/reports/majestic-million [Accessed: 27- September- 2019].

[16] Preethi, '14 Types of Phishing Attacks That IT Administrators Should Watch For', [Online]. Available: https://blog.syscloud.com/types-of-phishing [Accessed: 10- November- 2019].

[17] ‘https://github.com/rlilojr/Detecting-Malicious-URL-Machine-Learning/blob/master/dataset.csv’,[Online]. Available: https://github.com/rlilojr/Detecting-Malicious-URL-Machine-Learning/blob/master/dataset.csv Accessed: 10- November- 2019].

[18] Ebubekir Büber, ' Phishing URL Detection with M', [Online]. Available: https://towardsdatascience.com/phishing-domain-detection-with-ml-5be9c99293e5 [Accessed: 10- November- 2019].

[19] scikit-learn, , Machine Learning in Python, [Online]. Available: https://scikit-learn.org/stable/ [Accessed: 10- November- 2019].