DETECTING PHISHING WEBSITES, USING MACHINE LEARNING

1st Yuvraj	2nd Ritik	3rd Atinder	4th Rajveer
CSE-IS	CSE-IS	CSE-IS	CSE-IS
Chandigarh University	Chandigarh University	Chandigarh University	Chandigarh University
Mohali	Mohali	Mohali	Mohali
21BCS3581@cuchd.in	21BCS3573@cuchd.in	21BCS4569@cuchd.in	21BCS11717@cuchd.in

***Abstract*—** Phishing sites have recently grown to be a serious threat to online security. Marketing emails, malware, ransom ware, exploits, etc. are hosted on phishing websites. Phishing websites frequently look like popular shopping sites to trick unsuspecting customers into contracting the sickness. Financial misfortune, a lack of private messaging, and reputational misfortune result from being a fraud victim. Finding a solution that effectively reduces particular security dangers is therefore important. Historically, blacklists have been used to identify phishing websites. Many well-known websites, such as G. Phis- Tank, host a higher class of blacklisted domains. Two features of the blacklisting strategy are lacking: the blacklists are not particularly comprehensive and do not catch recently established phishing sites. Machine intelligence techniques have advanced recently.

Keywords- Deep learning and machine learning a phishing website was attacked, and a dataset for phishing websites, characteristics of fraud websites, and a respectable website are all included. Hazardous domains' verification and classification.

1. Introduction

In phishing, a fraud method is used to steal a person's confidential and sensitive data online. The rise in this type of attack has caused the threat of identity theft to increase. Online transactions are often less secure than they used to be.

Due to the rise of the Internet, many businesses have started using it to manage their offline activities. Phishing attacks can appear in various forms of communication. Some of these include spam emails, short message services, and messages over the Internet.

In most cases, phishing attacks are carried out through an email that lures users into clicking a link that takes them to a fake website that aims to steal their financial information.

In this work, a deep reinforcement learning-based model for phishing website identification is created using an analysis of the supplied URLs. Changes in the URL structure can be automatically accounted for by the model. The difficulty of spotting phishing websites is one illustration of the conventional classification issue. We have developed a reinforcement learning approach using deep neural networks to address this categorization challenge.

14 lexical characteristics from the provided URLs were used to train our model on a balanced and labelled dataset of both legitimate and malicious URLs. Performance is measured using F-measure, recall, precision, and accuracy. The following are the paper's main contributions:

1) The detection of phishing websites can be modelled using reinforcement learning (RL), where an agent picks up the value function from the input URL and uses it to do the task of sorting.

2) The usage of reinforcement It is possible to map the sequential decision-making process for labelling using learning based on robust neural networks.

3) The effectiveness of a phishing URL classifier based on deep reinforcement learning should be assessed, and its outcomes should be compared to those of the existing phishing URL classifiers.

Sure, here's an introduction to detecting phishing websites using machine learning, broken down into points:

The goal of spoofing assaults, a popular sort of cyber-attack, is to steal sensitive data from consumers by deceiving them into entering their information on bogus websites that appear to be real.

1) Conventional techniques for identifying phishing websites focus on manually compiling a listof recognized phishing websites or examining a website's URL and content for phishing indicators. These methods are time-consuming and not always accurate.

2) By analyzing vast quantities of data to find patterns and anomalies that are suggestive of phishing, machine learning may be used to more accurately detect phishing websites.

3) The URL, content, and user behavior of a website, as well as outside data sources like blacklists and prior phishing attempts, may all be used to train machine learning models.

4) Machine learning systems for phishing detection frequently include supervised learning techniques like decision trees, random forests, and support vector machines as well as unsupervised learning algorithms like clustering and anomaly detection

5) Web browsers or other security solutions can use machine learning models to give real-time detection of phishing websites, adding an extra layer of security for users.

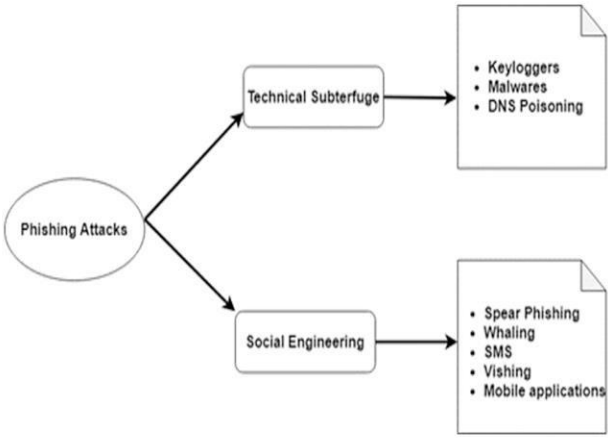
6) It's crucial to keep in mind, though, that machine learning models are not perfect and could result in false positives or false negatives. As a result, they should be used in conjunction with other security measures to offer complete defense against phishing assaults.

2. LITERATURE SURVEY

The phishing website detection model is anticipated to eliminate two types of misclassifications in order to successfully combat phishing attacks: False Negative Rate and False Positive Rate, respectively. Due to inaccurately describing valid websites as phishing, the first one prevents internet users from accessing legitimate websites, whilst the second one permits them to browse fake websites due to inaccurately labelling phishing websites as authorized.

Online blogs, SMS, short messaging services (SMS), VoIP systems where attackers use caller spoofing IDs, web 2.0 social media sites like Facebook and Twitter, peer-to-peer file-sharing services, and SMS are some of the leading.

social engineering techniques [3, 4]. How the process is carried out to trick the naive customer varies slightly depending on the sort of phishing. An email phishing attack occurs when a hacker sends a potential user an email that contains a link to a phishing website. Each type of phishing differs slightly in how the procedure is carried out to deceive the unwary customer. When an attacker uses social engineering to send a target email, it is called an email phishing attack..



2.1 Classification of phishing attack techniques

Phishing websites provide challenges for both organizations and individuals because of their resemblance to legitimate websites [5]. In Fig. 1, the various phishing attempts are depicted. Technical deception includes attacks like Key logging, DNS poisoning, and malware. In these assaults, the assailant uses a tool or technique to try to gain access. Users believe in the network, yet it has been compromised by attackers on the opposite side. Social engineering attacks include spear phishing, whaling, SMS, vising, and mobile apps, among others. In these attacks, attackers' prey on a particular set of people or businesses to trick them into visiting a phishing URL [6, 7]. In addition to existing attacks, the quantity of new attacks is rapidly increasing due to ongoing technological advancements. The many phishing attack types. Attacks like Key logging, DNS poisoning, and malware are examples of technical deception.

In these assaults, the assailant uses a tool or technique to try to gain access. Users believe in the network, yet it has been compromised by attackers on the opposite side. Social engineering attacks include spear phishing, whaling, SMS, vising, and mobile apps, among others. In these attacks, attackers' prey on a particular set of people or businesses to trick them into visiting a phishing URL [6, 7]. In addition to these strategies, new attacks are emerging quickly as a result of constant technological innovation.

3. Literature Review

The two methods for detecting phishing are as follows:

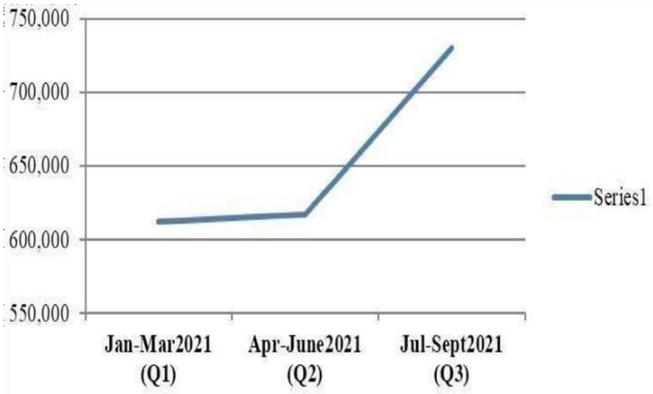
(1) Blacklisting is a method of determining whether a URL is hazardous by comparing it to a list of well-known phishing websites. If the URL is that one, there could not be any internet records of a recently built website because this method is essentially static.

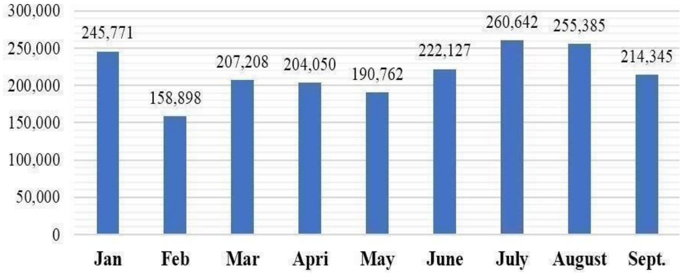
(2) Using several criteria to examine a certain URL. This method makes it easier to recognize phishing URLs. It parses the URL, gathers data from it, and applies a classifier to it to ascertain what a specific URL signifies. The classification in their proposed framework is based on the tied score of each phrase and lexical signatures developed based on the top five tufted scores.

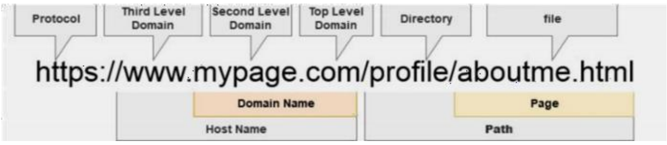
The vocabulary signature is then provided to a search engine like www.bing.com to dig for more details. The website is categorized as authentic approaches if a domain name with the same spelling as the one being analyzed appears in the search results. However, the questions endure on account of attackers steadily conceiving novel designs to exploit the existent antagonistic-phishing measures. Because the protection of connected to the internet consumers'.

4. Problem Overview

This study aims to explore the key gap in current research everything concentrating on machine intelligence and deep knowledge-located phishing site discovery because future scientists take care of using the recognize phishing Web page attacks testing structure knowledge and deep education institutions news cannot be missed, and the number of single site attacks resumes to rise at an alarming rate. The research break study procedures second hand in this place study chiefly devote effort to somethinghigh-quality-acted phishing site discovery model, site feature pick methods, dataset beginning, dataset magnitude, phish-authentic dataset percentages, allotment of Dataset Train-Test split percentage, the number of site countenance second hand, and run-opportunity study issues.







5. Proposed work

The component of this our domain name that is a domain name registrar as registered name. A domain name and subdomain are make up the hostname. The subdomain name component is simply modifiable and may be connected to any value by a phisher. The URL could also include a path and afile, which the phisher can simply change if he wants to. A URL's subdomain name and route are subject to the phisher's control. Once a domain has been discovered as fake and has only been registered once by an attacker, it is simple to stop users from viewing it. The URL's changeable subdomain and path are the source of the issue. Because of this, consumers, designers, and specialists in cyber security battle. Let's have a look at the following phishing URL.

Custom	http://
Domain name	myurl.com
way	/Sign-in/5b60fcc60b36d1c3d
Sub Domain	accounts
Sub Domain	Flipkart

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

URL so that a typical user would not be able to readily determine the real domain name.

URL so that a typical user would not be able to readily determine the real domain name

And it will be buried deep within the URL, for instance, the real Domain name in the aforementioned URL is "myurl.com."

5.1. A Model for deep reinforcement of learning

The training URL vectors' uniformities are captured by a generalized version of the suggested reinforcement-based learning model. Such that the

Learning agent receives the highest possible reward. The probability of the agent correctly forecasting remains the category of each test result. The agent uses this probability value to get information about its surroundings. The approximation function for the agent to learn from or profit from the environment is applied using a linear combine of feature vectors. There is no fresh information about the environment that the agent may learn to enhance prediction because the training method converges or the agent can constantly use the test data set to build the statistics needed for prediction.

The model is used to ignore the problems.

Information has been extracted from the vector space representation of the phishing URLs using deep neural networks. We use a softmax output layer, two fully linked layers with ReLU activation, one embedding layer, and two completely connected layers to create the DQN or agent of the reinforcement learning-based classifier. The gradient descent causes the policyparameter to be updated once the controller creates hyper-parameters. The rate of learning was 0.001. To increase accuracy, a 2-fold cross validation was carried out. In Figure 2, the network architecture is displayed. Both linear and nonlinear data may be used to train neural networks. Algorithm 1 presents the training and classification algorithm. The Real DQN learning method put forward to Mnih et al. [9] serves for the foundation for this approach.

Input: The training samples and their class labels

Output: The optimum Q-values

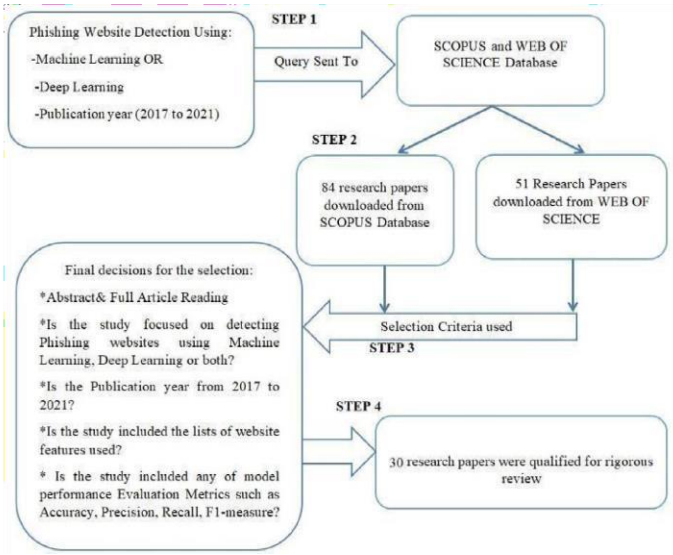
Algorithm 1: Training and classification algorithm

```
Initialize experience memory  ;
Initialize E-value function with random weights $\theta$;
Define episodes K;
Initialize target E-value function with random weights $\theta* = \theta$;
while episode 1 to K do
    Initialize s1 = v1;
    Initialize the early-processed sequence function $\varphi 1 = \varphi(s1)$ ;
    While t = 1 to T do
        action (at) based on -greedy,
        at =
        Random − action probability
        argmaxaQ((st), a; $\theta$) otherwise ;
        Observe rt for action at;
        Set st+1 = st, at, vt+1 ;
        Set pre-processed sequence function $\varphi t+1 = \varphi(st+1)$ ;
        Save ($\varphi 1$,at,rt,$\varphi t+1$, terminalj ) in M;
        Select Randomly sample from M and set
        yj usingequation.6;
        Perform gradient decent using equation.5;
        if terminalt = True then break;
    end
```

6.Experimental Setup

This section describes the experimental setup for the feature selection and feature extraction methodologies. The LSTM and CNN were built using a data set that included legitimate and fraudulent URLs. One million URLs from the data set were used to train the LSTM algorithm. Real websites from Common Crawl, a corpus of online crawl data, made up half of the data set, while phishing websites from Phish Tank, a website that acts as a repository for phishing URLs, made up the other half.. More than 10,000 photographs from both reliable and fraudulent websites were gathered to train the CNN. For the holdout cross-validation training and testing of the CNN, the picture information was split into two portions (30 and 70%, respectively). The raw data from URLs and images included a wide range of sizes and amounts of background information.

This data needed to be pre-processed in order to be ready for the model's training the images for the CNN architecture were just taken from the websites and cropped to fit the springy box. For the LSTM architecture, numerous website features were gathered and entered in Microsoft Excel as comma separated values.


STEP 1
Phishing Website Detection Using:
-Machine Learning OR
-Deep Learning
-Publication year (2017 to 2021)
Query Sent To
SCOPUS and WEB OF SCIENCE Database
STEP 2
84 research papers downloaded from SCOPUS Database
51 Research Papers downloaded from WEB OF SCIENCE
Final decisions for the selection:
*Abstract& Full Article Reading
*Is the study focused on detecting Phishing websites using Machine Learning, Deep Learning or both?
*Is the Publication year from 2017 to 2021?
*Is the study included the lists of website features used?
* Is the study included any of model performance Evaluation Metrics such as Accuracy, Precision, Recall, F1-measure?
Selection Criteria used
STEP 3
STEP 4
30 research papers were qualified for rigorous review

7. Experimentation details and results

The Ebbu2017 Phishing Dataset was used for the experiment for this study [5]. In their study on phishing URL detection, Sahingoz et al. produced and made the dataset available to the public [10]. Since there were no publicly available large phishing datasets, they developed a balanced dataset with both phishing and legitimate URLs. They developed their own script to use Yandex scan to scan the internet and collect phishing URLs from PhishTank3 and websites with better page ranks. The dataset contains 73,575 URLs, 36,400 of which are legitimate, and 37,175 of which are phishing. The training dataset is discrete and has deterministic classes. The proposed model can generate binary forecasts of test data observations.

https://tech.yandex.com

http://www.phishtank.com

Precision, recall, accuracy, and the F-measure have all been used to evaluate the performance of the suggested model. We must compute the True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) predictions in order to calculate these metrics. While FP and FN represent the misclassified data, TP and TN stand for the outcomes that were correctly classified. These 4 values allow us to compute the performance measures in the manner shown below:

• Precision = True Positive
True Positive +False Negative

In classification problem a precision value closer to 1 implies the predicted labels are closer to truth.

• Recall = T P
T P +F N

A recall value closer to all the testing
Samples could be predicted using the specified model.

• Accuracy = T P +T N
T P +T N+F P +F N

A accuracy score closer to 1 implies a high performance of the system.

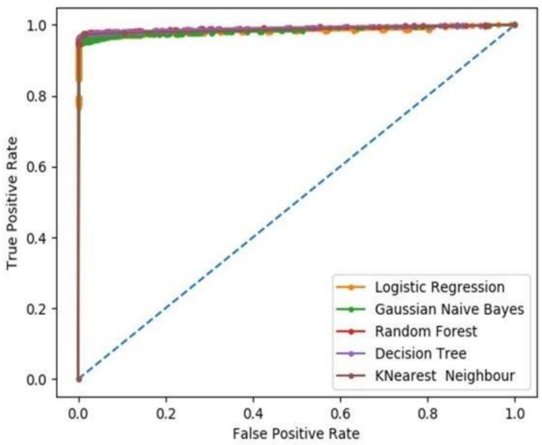
• F − Score = 2 ∗ Precision∗Recall
Precision+Recall
The harmonic mean of recall and precision is used to denote the model's resilience.

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

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

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

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



8. Conclusion

This research presents a framework for automated reinforcement learning-based url-based phishing detection. This deep learning application of the RL algorithm complements the current phishing detection methods and makes the system dynamic. This work lays the foundation for a more efficient, dynamic, and self-adaptive paradigm for phishing identification. Because this work is not optimized for real-world deployment, our next work will involve performance tuning for the Deep Q Learning Algorithm to optimise the Markovi Decision Process [18, [19] for ideal classification. Moreover, for the purposes of this experiment, we have only used the lexical aspects of the URLs; however, we would like to investigate how well our model performs when we include other advanced features like host-based features, content-based features, etc.

References

[1] Abdelhamid, N., Ayesh, A. and Thabtah, F., 2014. Phishing detection based associative classification data mining. Expert Systems with Applications, 41(13), pp.5948-5959.

[2] Chatterjee, M. and Siami Namin, A., 2018, July. Detecting web spams using evidence theory. IEEE 42nd annual computer software and applications conference (COMPSAC).

[3] Chatterjee, M., Siami Namin, A. and Datta, P., 2018, December. Evidence Fusion for Malicious Bot Detection in IoT. IEEE International Conference on Big Data (Big Data).

[4] Datta, P., Siami Namin, A. and Chatterjee, M., 2018, December. A Survey of Privacy Concerns in Wearable Devices. IEEE International Conference on Big Data (Big Data).

[5] Ebbu2017 Phishing Dataset. Accessed 5 April 2019. Available:https://github.com/ebubekirbbr/pdd/tree/master/input.

[6] Liu, W., Huang., G., Xiaoyue, L. Min, Z., and Deng, X., 2005., Detection of phishing webpages based on visual similarity. 14th international conference on world wide web (WWW).

[7] Nguyen, N. Siami Namin, A., Dang, T. 2018. MalViz: an interactive visualization tool for tracing malware. ISSTA.

[8] Mohammad, R.M., Thabtah, F. and McCluskey, L., 2012, An assessment of features related to phishing websites using an automated technique, IEEE Conference for Internet Technology and Secured Transactions.

[9] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., and Hassabis, D. (2015). Human-level control through deep reinforcement learning. Nature, 518(7540):529533.

[10] Sahingoz, O.K., Buber, E., Demir, O. and Diri, B., 2019. Machine learning based phishing detection from URLs. Expert Systems with Applications, 117, pp.345-357.

[11] Sartoli, S., and Siami Namin, A., 2017, A semantic model for actionbased adaptive security, Symposium on Applied Computing (SAC).

[12] Siami-Namini, S., Tavakoli, N., and Siami Namin, A., 2018, December. A Comparison of ARIMA and LSTM in Forecasting Time Series. International Conference on Machine Learning and Applications ICMLA.

[13] Siami-Namini, S. and Siami Namin, A., 2018, Forecasting Economics and Financial Time Series: ARIMA vs. LSTM, CoRR abs/1803.06386.

[14] Sutton, R.S. and Barto, A.G., 2018. Reinforcement learning: An introduction. MIT press.

[15] Tavakoli, N., Dai, Dong, and Chen Y., 2019, Client-side straggler-aware I/O scheduler for object-based parallel file systems, Parallel Computing.

[16] WHOIS: Search, Domain Name, Website, and IP Tools. https://who.is

[17] Zhang, Y., Hong, J.I. and Cranor, L.F., 2007, May. Cantina: a contentbased approach to detecting phishing web sites. In Proceedings of the ACM conference on World Wide Web.

[18] Zheng, J. and Siami Namin, A., 2018, A Markov Decision Process to Determine Optimal Policies in Moving Target, Proceedings of the ACM SIGSAC Conference on Computer and Communications Security.

[19] Zheng, J. and Siami Namin, A., 2018, Defending SDN-based IoT Networks Against DDoS Attacks Using Markov Decision Process, IEEE Conference on Big Data.

[20] Xiang, G., Hong, J., Rose, C.P. and Cranor, L., 2011. Cantina+: A feature-rich machine learning framework for detecting phishing web sites. ACM Transactions on Information and System Security (TISSEC).