Real Time Detection of Phishing Attacks in Edge Devices

Ushus Maria Joseph, Dr. Mendus Jacob
Research scholar
Lincoln University College Malaysia

***Abstract:-* Phishing is the duplicitous use of electronic communications to deceive and take advantage of users. Phishing attacks attempt to gain sensitive, confidential information such as usernames, passwords, credit card information, network credentials, and more. By posing as a legitimate individual or institution via phone or email, cyber attackers use social engineering to manipulate victims into performing specific actions—like clicking on a malicious link or attachment—or willfully divulging confidential information. Phishing is an antisocial and therefore anti sustainable activity, which has an important social and economic dimensions. Mobile devices are popular with hackers because they're designed for rapid responses based on minimal contextual information. The main objective of this work is to develop a model that can detect and prevent possible phishing attacks in real time. An intelligent phishing attack detection mechanism, when implemented to handheld devices increases the bandwidth of protection in the global digital village. Preventing phishing itself help attain sustainable development. Different algorithms like Decision trees and SVM are used to detect phishing attacks. This type of algorithm needs large computational power to run and the user data is sent to the server for the detection of phishing attacks. The main disadvantage is that our browsing data is sent to another server for analysis. In this situation a third party has access to our browsing data which will lead to a privacy issue and detection can be dependent upon various factors like network bandwidth etc. To overcome the privacy issue we can use real-time prediction on the client-side. For detecting phishing attacks in low computational devices we can use a quantized model.**

Keywords:-* Neural Networks, Phishing Attacks, Binary cross entropy loss value, Relu Activation function, TensorFlow lite, Quantization

1. INTRODUCTION

Phishing is the tricky utilization of electronic correspondences to beguile and exploit clients. Phishing assaults endeavor to acquire delicate, private data, for example, usernames, passwords, Visa data, network accreditations, and that's only the tip of the iceberg. By acting like a genuine individual or establishment through telephone or email, digital assailants utilize social designing to control casualties into performing explicit activities—like tapping on a malignant connection or connection—or determinedly unveiling classified data. Phishing is a kind of friendly designing assault regularly used to take client information like login subtleties, Visa numbers and other delicate data. Aggressors attempt to imitate a genuine organization or an individual to take data.

For example, this can be done by imitating your banking service to send you an email stating "your account is in trouble, you need to change your password" with a link to change the password. Someone who is unaware of the threat may click on the link to submit their login credentials to the attacker. This is the most common phishing attack and just one method among a lot of other phishing attacks. Phishing is an antisocial and therefore anti sustainable activity, which has an important social and economic dimensions. Mobile devices are mainstream with hackers since they're intended for fast reactions dependent on insignificant logical data.

Crooks trust on duplicity and making a desire to move quickly to make progress with their phishing efforts. During an emergency, individuals are tense. They need data and are searching for course from their managers, the public authority, and other important specialists. A great deal of arrangements has been proposed for the identification and avoidance of phishing assaults, still the danger isn't reduced. Email phishing assaults have spiked because of Corona infection. An effective phishing assault can have appalling ramifications for the casualties prompting monetary misfortunes and wholesale fraud. Be that as it may, ensuring ourselves against phishing is both a possible and fundamental advance. With the consistently expanding utilization of messages and development of advancements, hazard of losing important data to fraudsters has likewise been expanding This work focuses on detecting phishing attacks in real time with the help of neural networks.

2. REVIEW OF LITERATURE

Usually, Hackers will influence the users through phishing in order to gain access to the organization's digital assets and networks. With security breaches, cybercriminals execute ransomware attack, get unauthorized access, and shut down systems and even demand a ransom for releasing the access. Anti-phishing software and techniques are circumvented by the phishers for dodging tactics. Though threat intelligence and behavioral analytics systems support organizations to spot the unusual traffic patterns, still the best practice to prevent phishing attacks is defended in depth ***[1]***.

Phishing attacks have been increasing recently. Attackers use clever social engineering techniques to convince their victims into clicking a malware or deceptive login-based webpages. Most solutions for this particular problem focus more on helping desktop computer users than mobile device users. Mobile device users are more vulnerable than their desktop counterparts because they are online most of the time and they have device limitations such as smaller screen size and low computational power ***[2]***.

Phishing attacks are generally carried on using the four steps listed below:

1. Attackers configure a forged web site that impersonates the legitimate one. They also apply the Domain Name System and configure the web server.
2. A bulk pool of spoofed e-mails are dispatched to end users making them to input their personal details.
3. These victims open or click on the spoofed links assuming that they are from legitimate companies and organizations and provide their personal details.
4. This confidential information from the receivers are acquired and frauds are performed by the phishers ***[3]***.

Phishing websites is one such area where administrators need new techniques and algorithms to protect naïve users from getting exploited. Phishing is an attempt of fraud aimed at stealing our information, which is mostly done by emails. These phishing emails mostly come from trusted sources and try to retrieve our valuable information, for instance our passwords, bank details or even SSN. Many a times, these attacks come from sites where we have not even any type of account. ***[4]***.

A lot of solutions have been proposed for the detection and prevention of phishing attacks, still the threat is not alleviated. Blacklisting, Uniform Resource Locator (URL) based detection, static detection, and heuristics techniques are various methods used for detecting phishing attacks ***[5]***. Recent years have witnessed the increasing threat of phishing attacks on mobile computing platforms. In fact, mobile phishing is particularly dangerous due to the hardware limitations of mobile devices and mobile user habits. Existing schemes designed for web phishing attacks on PCs cannot effectively address the various phishing attacks on mobile devices. ***[6]***.

As compared to desktop users, mobile device users are at least three times more vulnerable to phishing attacks and the reason for this vulnerability is small screen size, lack of identity indicators, inconvenience of user input, switching between applications, habits and preferences of mobile device users. By exploiting the hardware limitations of these devices and careless behaviour of the users, an attacker can easily carry out phishing attack on mobile phones ***[7].***

3. OBJECTIVE OF THE STUDY

The main objective of this work is to develop a model that can detect and prevent possible phishing attacks in real time. An intelligent phishing attack detection mechanism, when implemented to handheld devices increases the bandwidth of protection in the global digital village. Preventing phishing itself help attain sustainable development. Different algorithms like Decision trees and SVM are used to detect phishing attacks. This type of algorithm needs large computational power to run and the user data is sent to the server for the detection of phishing attacks. The main disadvantage is that our browsing data is sent to another server for analysis. In this situation a third party has access to our browsing data which will lead to a privacy issue and detection can be dependent upon various factors like network bandwidth etc. Traditional machine learning models can be so huge that it runs on cloud servers, which makes it less efficient. The predictions can take so long that it may not be real-time always. This delay in prediction and warning can cause malware to affect the system before we can do anything. Reverse engineering is also affected in machine learning models. Attackers reconstructed the models for extracting design information from them. In order to do this prediction natively in the end-point system, the model must be compact and optimized enough to work in real-time minimizing the system resource usage. With the help of machine learning, each time a threat happens, the algorithm learns by itself and prevents the possibility of being exploited.

4. PROPOSED METHODOLOGY

To overcome the privacy issue we can use real-time prediction on the client-side. The main issue is that currently using detection algorithms needs large computational power for making predictions and this type of algorithm is not suitable for running on the client-side. For detecting phishing attacks in low computational devices we can use a quantized model. The quantized model size is very low and it is suitable for running in low computational devices and we can detect the phishing attacks in real-time. The main advantage of deploying the model on the client-side is that the user data is not sent to a third party and there is no privacy issue in the system, our browsing data is safe on the client-side. While the model is running on the client-side the speed of the detection is very fast compared to the data sending to the server for detection of phishing attacks.

4.1 FEATURES

After referring to available literature, we have selected and defined a set of features that capture the characteristics of phishing emails. [8, 9]

1.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”.

2. Using URL Shortening Services “TinyURL”

URL shortening is a method on the “World Wide Web” in which a URL may be made considerably smaller in length and still lead to the required webpage. This is accomplished by means of an “HTTP Redirect” on a domain name that is short, which links to the webpage that has a long URL. For example, the URL “http://portal.hud.ac.uk/” can be shortened to “bit.ly/19DXSk4”.

3. having “@” Symbol

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

4. Domain Registration Length

Based on the fact that a phishing website lives for a short period of time, we believe that trustworthy domains are regularly paid for several years in advance. In our dataset, we find that the longest fraudulent domains have been used for one year only.

5. HTTPS (Hyper Text Transfer Protocol with SSL)

The existence of HTTPS is very important in giving the impression of website legitimacy, but this is clearly not enough. Certificate Authorities that are consistently listed among the top trustworthy names include: “GeoTrust, GoDaddy, Network Solutions, Thawte, Comodo, Doster and VeriSign”. Furthermore, by testing out our datasets, we find that the minimum age of a reputable certificate is two years.

6. Favicon

A favicon is a graphic image (icon) associated with a specific webpage. Many existing user agents such as graphical browsers and newsreaders show favicon as a visual reminder of the website identity in the address bar. If the favicon is loaded from a domain other than that shown in the address bar, then the webpage is likely to be considered a Phishing attempt.

7.Port

This feature is useful in validating if a particular service (e.g. HTTP) is up or down on a specific server. In the aim of controlling intrusions, it is much better to merely open ports that you need. Several firewalls, Proxy and Network Address Translation (NAT) servers will, by default, block all or most of the ports and only open the ones selected. If all ports are open, phishers can run almost any service they want and as a result, user information is threatened.

8. The Existence of “HTTPS” Token in the Domain part of URL

The phishers may add the “HTTPS” token to the domain part of a URL in order to trick users. For example, http://https-www-paypal-it-webapps-mpp-home.soft-hair.com/.

9. URL of Anchor

An anchor is an element defined by the <a> tag. This feature is treated exactly as “Request URL”. However, for this feature we examine: If the <a> tags and the website have different domain names. This is similar to request URL feature. If the anchor does not link to any webpage, e.g.:

<a href=“#”>
<a href=“#content”>
<a href=“#skip”>
<a href=“JavaScript ::void(0)”>

10.Links-tags

Given that our investigation covers all angles likely to be used in the webpage source code, we find that it is common for legitimate websites to use <Meta> tags to offer metadata about the HTML document; <Script> tags to create a client side script; and <Link> tags to retrieve other web resources. It is expected that these tags are linked to the same domain of the webpage.

11.Server Form Handler (SFH)

SFHs that contain an empty string or “about: blank” are considered doubtful because an action should be taken upon the submitted information. In addition, if the domain name in SFHs is different from the domain name of the webpage, this reveals that the webpage is suspicious because the submitted information is rarely handled by external domains.

12.Abnormal URL

This feature can be extracted from WHOIS database. For a legitimate website, identity is typically part of its URL.

13. Redirect

The fine line that distinguishes phishing websites from legitimate ones is how many times a website has been redirected. In our dataset, we find that legitimate websites have been redirected one-time max. On the other hand, phishing websites containing this feature have been redirected at least 4 times.

14. On Mouse over

Phishers may use JavaScript to show a fake URL in the status bar to users. To extract this feature, we must dig-out the webpage source code, particularly the “onMouseOver” event, and check if it makes any changes on the status bar.

15. Using Pop-up Window

It is unusual to find a legitimate website asking users to submit their personal information through a pop-up window. On the other hand, this feature has been used in some legitimate websites and its main goal is to warn users about fraudulent activities or broadcast a welcome announcement, though no personal information was asked to be filled in through these pop-up windows.

4.2 NEURAL NETWORKS

A neural network is a computational model that has a network architecture. This design is comprised of counterfeit neurons. This design has explicit boundaries through which one can change it for playing out specific errands. A neural network has numerous layers. Each layer plays out a particular capacity, and the complex the organization is, the more the layers are. That is the reason a neural organization is additionally called a multi-facet perceptron.

The purest form of a neural network has three layers:
The input layer
The hidden layer/secret layer
The output layer

As the names recommend, every one of these layers has a particular reason. These layers are comprised of hubs. There can be various secret layers in a neural organization as per the prerequisites. The information layer gets the info signals and moves them to the following layer. It assembles the information from the rest of the world.

An example of a neuron showing the input (x_{1}-x_{n}), their corresponding weights (w_1 -w_{n}), a bias (b) and the activation function f applied to the weighted sum of the inputs.

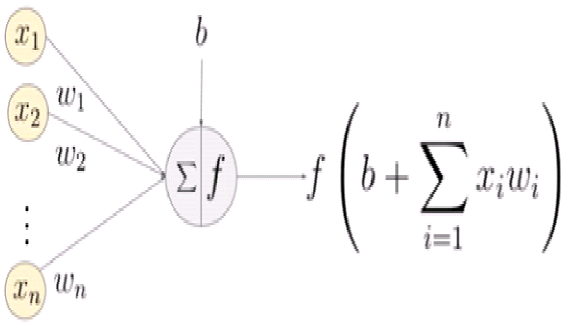

Fig 1: Working of a Simple Neuron

The secret layer plays out all the back-end assignments of estimation. A network can even have zero secret layers. Nonetheless, a neural network has at any rate one secret layer. The output layer communicates the end-product of the secret layer's estimation.

For computation, each neuron considers weights and bias. Then, the combination function uses the weight and the bias to give an output (modified input). It works through the following equation:

combination = bias +weights * inputs

After this, the activation function produces the output with the following equation:

output = activation(combination)

Information is fed into the input layer which transfers it to the hidden layer. The interconnections between the two layers assign weights to each input randomly. A bias added to every input after weights are multiplied with them individually. The weighted sum is transferred to the activation function. The activation function determines which nodes it should fire for feature extraction. The model applies an application function to the output layer to deliver the output. Weights are adjusted, and the output is back-propagated to minimize error. The model uses a cost function to reduce the error rate. You will have to change the weights with different training models. The model compares the output with the original result. It repeats the process to improve accuracy.

4.3 ReLU (Rectified Linear Unit) ACTIVATION FUNCTION

The rectified linear activation function or ReLU is a linear function that will output the input directly if it is positive, otherwise, it will output zero.

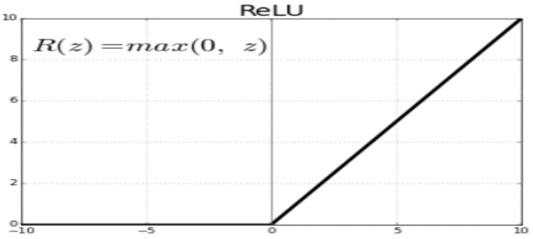

Fig 2: Representation of ReLU

Equation: f(x) = max (0, x)
Range: (0 to infinity)

Advantages of using ReLU are the function and its derivative both are monotonic, due to its functionality it does not activate all the neuron at the same time and
it is efficient and easy for computation.

4.4 BINARY CROSS ENTROPY LOSS FUNCTION

Binary cross entropy is a loss function that is used in binary classification tasks. These are tasks that answer a question with only two choices (yes or no, A or B, 0 or 1, left or right).

```
model. compile (optimizer='adam',
          loss=tf.keras.losses.binary_crossentropy,
          metrics=['accuracy'])
```

4.5 FEATURES OF NEURAL NETWORK MODEL TO MAKE INFERENCE AT THE EDGE

Light-weight**:** Edge devices have limited resources in terms of storage and computation capacity. Deep learning models are resource-intensive, so the models we deploy on edge devices should be light-weight with smaller binary sizes. [9]

Low Latency**:** NN models at the Edge should make faster inferences irrespective of network connectivity. As the inferences are made on the Edge device, a round trip from the device to the server will be eliminated, making inferences faster.

Secure**:** The Model is deployed on the Edge device, the inferences are made on the device, no data leaves the device or is shared across the network, so there is no concern for data privacy.

Optimal power consumption**:** Network needs a lot of power, and Edge devices may not be connected to the network, and hence, the power consumption need is low.

Pre-trained**:** Models can be trained on-prem or cloud for different deep learning tasks like image classification, object detection, speech recognition, etc. and can be easily deployed to make inferences at the Edge.

4.6 QUANTIZATION

When we save the TensorFlow Model, it stores as graphs containing the computational operation, activation functions, weights, and biases. The activation function, weights, and biases are 32-bit floating points. Quantization reduces the precision of the numbers used to represent different parameters of the TensorFlow model and this makes models light-weight. Quantization can be applied to weight and activations. Weights with 32-bit floating points can be converted to 16-bit floating points or 8-bit floating points or integer and will reduce the size of the Model. Both weights and activations can be quantized by converting to an integer, and this will give low latency, smaller size, and reduced power consumption. ***[10]***

```
converter = tf. lite. TFLiteConverter.from_keras_model(model)
tflite_model = converter. Convert ()
```

5. RESULTS

To evaluate the model accuracy 20% test data is used and the weights of the model are quantized for running in edge devices. Using this quantized model, we can run the model in the edge device and the predictions are real time. Accuracy is defined as the number of correct predictions divided by total number of predictions made. When the model training dataset is small and lightweight architecture there will arise situations like overfitting and underfitting in the model. When the hidden units in the model are increased it will affect the speed of the model running in the edge device and computational cost also increases. This model can be deployed either as a browser extension or an application.

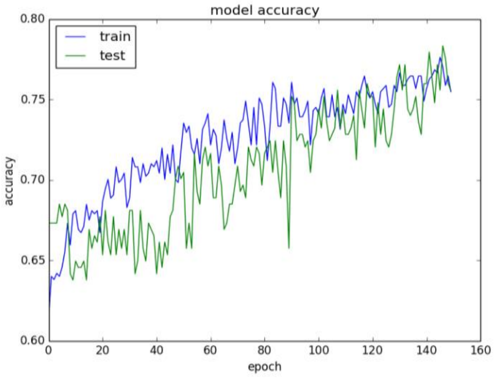

Fig 3:Model Accuracy using ReLU

6. FUTURE WORKS

As the Future works, to diminish the execution time and increment the productivity of the framework, the force of the Bayesian probabilities can be utilized. Also, different methodologies of Deep Learning, like recurrent neural networks, convolutional neural networks and LSTM can be tried for expanding the presentation of the framework.

BIBLIOGRAPHY

[1] D. S. F. L. S.-E.-U. H. Mohammad Nazmul Alam, "phishing attacks detection using machine learning approach," in *Third international conference on smart systems and inventive technnology*, 2020.

[2] Y. K. b. Jema David Ndibwile, "UnPhishMe: Phishing Attack Detection by Deceptive Login Simulation through an Android Mobile App," in *The 12th Asia Joint Conference of Information Security (AsiaJCIS)At: Seoul, South KoreaVolume: 978-1-5386-2132-5/17 $31.00 © 2017 IEEE*, 2017.

[3] M. M. K. S. H. S. U. S. N. Dr.Reshma banu, "Phishing Attacks Detection using Machine Learning Approach," in *Proceedings of the International Conference on Intelligent Computing and Control Systems (ICICCS 2019)*, 2019.

[4] J. S. S. S. J. S. S. S. IshantTyagi, "Detection of Phishing Attacks using Machine Learning," in *5th International Conference on Signal Processing and Integrated Networks (SPIN)*, 2018.

[5] A. K. J. Diksha Goel, "Mobile phishing attacks and defencemechanisms: State of art and open research: State of art and open research," *computers & security* , vol. 73, pp. 519-544, 2018.

[6] X. D. S. M. I. a. J. W. F. I. Longfei Wu, "Effective Defense Schemes for Phishing Attacks on Mobile Computing Platforms," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 8, 2016.

[7] X. D. a. J. W. Longfei Wu, "MobiFish: A Lightweight Anti-Phishing Scheme for," in *International Conference on Computer Communications and Networks (ICCCN)*, 2014.

[8] "https://www.kaggle.com/akashkr/phishing-url-eda-and-modelling," [Online]. Available: https://www.kaggle.com/akashkr/phishing-url-eda-and-modelling.

[9] https://www.kaggle.com/akashkr/phishing-url-eda-and-modelling.

[10] "www.tensorflow.org," [Online].

[11] Y. S. Tianrui Peng.Ian G Harris, "Detecting Phishing Attacks Using Natural Language Processing and Machine Learning," in *IEEE International Conference on Semantic Computing*, 2018.

[12] S. a. D.Soni, "A security model to detect smishing through sms content analysis and url behavior analysis," in *Future Generation computer systems-the International Journel of Escience*, 2020.

[13] G. a. K. Kuppusamy, "A phishing detection model with multi-filter approach," *Journel of king saud university-computer and information sciences,* vol. 32, no. jan 2020, pp. 99-112, 2020.

[14] S. I. B. a. D. B. Ozgur Koray Sahingoz, "Phishing Detection from urls by neural networks," *Computer science and Information Technology,* 2019.