Analysis of A Novel Phishing Detection Approach

Brenden Morton and Jesse Chehal

Dept. of Electrical and Computer Engineering, University of Central Florida, Orlando, Florida, 32816-2450

***Abstract* —** Internet phishing is a common attack that indiscriminately targets sensitive user data. Phishing detection algorithms have been developed to identify and protect from phishing attempts. The paper under review [1] describes a novel phishing detection model designed to have both quick access time and high phishing detection, which competing phishing detection algorithms at the time of the paper's publication could not compete with. We will replicate the model proposed by the authors and discuss our findings to judge the effectiveness of their model.

I. Introduction

Phishing is the act of spoofing to deceive users or even companies to obtain sensitive data. This data can include credit card information, bank account credentials, social security numbers and any other sensitive data that is intended to be kept private. This cybersecurity vulnerability tends to attack its targets over emails, text messages and fraudulent web pages. It is evident that internet phishing attacks rely on internet users. With the extreme changes seen over the past few years with the COVID-19 pandemic and the ensuing overall worldwide shift to a hybrid-remote work and life, phishing attacks have had a much wider target audience to feed on. According to Shi on Barracuda Networks [2], phishing emails saw an almost 700% increase just in February 2020. Two years later and phishing attacks are still going strong. Open up the spam folder on GMail or on Outlook and it is clear that phishing emails are being detected. Some, however, still slip through filters and end up in your inbox.

We will briefly describe a simple example of internet phishing to provide some background. A good depiction of a phishing life cycle can be found in the author's paper [1]. An attacker will first create a webpage that is very similar, if not identical, to a legitimate site (i.e. www.apple.com). This site however, will be modified such that the hyperlinks, log-in fields, etc can be leveraged to have a susceptible individual leak their private information. After an attacker creates one of these sites, they will host it on the web. These attackers can then use a variety of different methods to advertise this fraudulent phishing site such as email. In the case of email, the attacker will email hundreds if not thousands of random people an email containing the link to their phishing site. Some users will click on the site thinking the email is legitimate and then be served a normal and legitimate looking webpage. They will log-in, provide credit card info, etc. while thinking the site is legitimate, when in reality the attacker has obtained your sensitive private information. This attack is quite simple for an attacker to orchestrate.

Phishing detection is a critical field in cybersecurity that involves identification of phishing attacks using different approaches and then prevention of these attacks through analysis and characterization of the identified phishing attempt. However, this is easier said than done. Phishing attacks are constantly evolving to avoid detection and to increase their effectiveness.

The purpose of this paper is to assess the overall effectiveness of the novel phishing detection approach designed by Jain and Gupta [1]. The author's proposed framework utilizes an auto-updated whitelist of websites that are determined to be safe and utilizes hyperlink features of websites to classify the integrity of a site as either phishing or legitimate. As claimed by Jain and Gupta [1], this approach ensures both fast access time and high phishing detection rate as compared to other phishing detection approaches including black-list based approaches, which have fast access time but suffer from low detection rate, and

machine learning based approaches which have high detection rate but slow access time.

To properly assess the effectiveness of the approach under review [1], we will implement their proposed model and then use sites that have been determined to be phishing and valid legitimate sites as inputs to the model. We will then compare the results we get with the results that the authors [1] obtained.

It is worth noting that this article, and the proposed phishing detection framework, was released publicly in 2016. Given that more than 5 years has passed since the publication of this paper, phishing sites have had some time to evolve to avoid detection through this algorithm. Given that this approach [1] focuses on the features of a hyperlink set from a given webpage, those employing phishing sites would really only need to slightly modify the code on the site, more specifically the HTML and the hyperlinks, href tags, residing on the page. This time period since the publication of the paper will be a useful indicator to assess the effectiveness of this approach given the reasons previously stated.

II. Overview of Paper [1] and author's implementation

We will begin with an overview of the phishing detection model described in the paper under review [1]. This paper lays out an anti-phishing system that claims to have both fast access time and high detection rate which is not the case for other alternative phishing detection models.This model can be broken down into two main parts: auto-updated whitelist and the phishing identification algorithm.

We will briefly describe the structure of the whitelist. The whitelist is a central authority in this proposed model, though it is quite simple in structure. The whitelist is essentially a lookup table with the following format as seen in Table 1 below:

Key	Value
Domain	IP

Table 1. Representation of a single entry in the whitelist.

Each entry in the whitelist structure, simply contains the domain of the site and the IP resolved from the DNS query of the domain.

The phishing identification algorithm proposed by Jain and Gupta [1], solely uses the hyperlink set extracted from a given web page under review for classification. When a website is pushed into the phishing identification algorithm, the hyperlinks residing on the page in href tags are extracted into a set. The phishing detection algorithm uses this set of hyperlinks to perform some feature-based analysis to characterize a site as either phishing or legitimate. There are three features that are analyzed: number of hyperlinks, number of NULL hyperlinks and the number of links that point to a foreign domain.

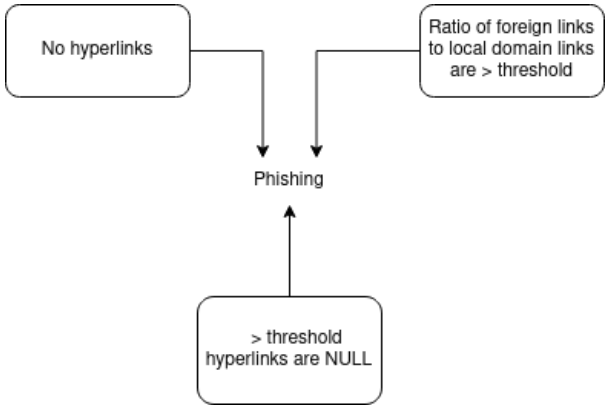

Fig 1. Phishing identification algorithm from Jain and Gupta [1]

Figure 1 above depicts the flowchart for the phishing detection algorithm. If all of these

conditions are not met, then the site is classified as legitimate and this site is added to the whitelist of safe sites that they use. Otherwise, if any of the conditions depicted in Figure 1 are satisfied, then the site is determined to be phishing and is not added to the whitelist.

The other part of the phishing detection model proposed by Jain and Gupta [1], is the URL and domain name matching. This module is the first stage that a webpage encounters when tested in the proposed model. Each time a page is accessed (input into the model), it is queried in the whitelist. If the page is not in the whitelist, then it immediately enters the phishing identification algorithm. Otherwise, it moves on to a domain matching condition. Before this condition, a third-party service is used to perform a DNS query on the site to find out the current URL to domain translation. This is a critical part of the model since DNS poisoning is common and this ensures that the whitelist is up-to-date and not tainted. With the DNS query response and the whitelist lookup result, the domain can be matched. If the domain is not in the whitelist, then the program, then the site is sent to the phishing identification algorithm for further analysis. If the webpage is in the whitelist already and the DNS query response (IP) and the webpage's IP match then the page is considered Legitimate. If they do not match, then the page is considered Phishing.

These two modules, the URL and DNS matching module and the Phishing Identification module make up the proposed phishing detection model [1]. As noticed, a simple yet crucial auto-updated whitelist and the hyperlink features of a webpage are used together to identify if a site is phishing or not.

III. Implementation

Now that the model has been given a proper overview, we will discuss how we implemented the proposed phishing detection model. The author's state that they used Java 7 (JDK 7) to implement their proposed model. However, we decided to use the Python interpreted programming language; more specifically Python 3.8.10. The choice to use Python over Java was mainly determined by the ease of Python over Java. There is no need to use the same programming language as the author's since speed, efficiency, or any programming language metrics including compile-time are not under review. Instead, the proposed model described in the paper is of interest. Thus, Python or any programming language is suitable for implementation of the proposed model.

The author's [1] implementation of their phishing detection model included a browser plug-in to detect in real-time phishing sites. In our implementation, we decided not to go this route mainly to speed up the testing time. Instead, we use a list containing a combination of phishing sites and legitimate sites to simulate internet activity. This has no effect on determining the effectiveness of the proposed phishing detection model since a browser plug-in is real-time internet activity consisting of browsing various sites and our implementation of simulated internet activity by having a predetermined list of sites is achieving the same goal as the author's implementation: testing the effectiveness of their proposed phishing detection model.

By using the descriptions and block diagrams from the paper [1], we were able to implement the author's proposed phishing detection algorithm. Our implementation can be found on GitHub by following this link:
https://github.com/brend3n/CAP6135-Phishing

This link will also be provided in the references section of this paper.

VI. Dataset

To properly test our implementation, a combination of both confirmed phishing sites as well as confirmed legitimate websites are needed. The authors Jain and Gupta used the PhishTank [3] website for acquiring a list of confirmed phishing sites. PhishTank [3] provides a real-time list of user submitted websites that are then classified as either Phishing or not Phishing. PhishTank is operated by Cisco, more specifically by Cisco Talos Intelligence Group. We chose to use the same data source as the authors because this data source not only provided an easy way to obtain regularly updated phishing site data, but also closely follows the authors implementation. To use the dataset from PhishTank, a function in our code makes a GET request to an api call that returns the phishing data in a JSON format. This format is then easy to parse in Python 3.8.10.

In addition to phishing site data, we also need legitimate sites to test on. To do this, we used a site called Moz [4]. Moz provides a list of the top 500 most popular sites accessed worldwide. This site also provides what they call a Domain Authority score which judges the security of the domain for that site. All of the sites from Moz had exceptional Domain Authorities. The Domain Authority rating along with the fact that these are 500 of the most popular sites worldwide, ensures that these are indeed safe legitimate websites. In addition to Moz, we used a public GitHub repo from Ben Sooter [5]. This repo [5] contained about 1000 of the most popular websites. This additional legitimate site data was needed to match the data sample size from the author's implementation of their proposed phishing detection model.

The combination of the phishing sites retrieved from PhishTank and the legitimate sites obtained from Moz, we have a good dataset to test the proposed phishing detection model.

V. Results

To properly compare the results of our implementation of the author's proposed phishing detection model with the results from the authors, we will replicate the same graphs, tables, and charts the author's included in the results section of their paper. We will omit the graph included in the paper [1] that depicts the growth of the whitelist since that is not necessarily judging the effectiveness of their proposed phishing detection model. Similarly, we will omit the table comparing various alternative anti-phishing solutions to the author's proposed anti-phishing solution since this is not judging the overall effectiveness of their detection model.

While in the testing stage of our implementation of the author's proposed phishing model, it was evident that some of the phishing sites that we were running through our program were hanging around when attempting to make a GET request on the website's page content. The most likely case for this can be attributed to the site being taken down or the creator's of the phishing site taking it down themselves. As a result of this, the GET request for the site would just hang and no content would be received. To circumvent this issue, we added a 5 second timeout for the GET requests. The value of 5 seconds was chosen arbitrarily and has been a suitable timeout value for the purposes of our implementation. Another issue that we began to notice is SSL certificate errors, invalid domain name errors, etc. Clearly, the SSL certificate errors are warning us, the user, that the site to be reached is insecure and a possible threat. We could not find a way around this. In addition to this, there were DNS query responses that were invalid. This could be attributed to sites no longer being hosted. These two issues were circumvented by wrapping the rest of the code in a try-except block. This filtered out these exceptions in the code. Because we have such a large dataset of both valid legitimate websites and phishing sites, it is fine if we obtain these errors.

For the remainder of this section, we will assess the author's proposed phishing detection model by comparing their results with the results that we obtained. We mirrored the calculations and results shown so that we can properly compare the results we obtained to the results from the authors.

A. Comparison of Table 2

First, we will compare the results from Table 2 from the paper [1]. Table 2 depicts the ratio of foreign domain hyperlinks over the total number of hyperlinks for different thresholds. The purpose of this is to determine a suitable threshold (%) value that maximizes the percentage of phishing sites detected while at the same time minimizes percentage of legitimate web pages being classified as phishing based on foreign domain hyperlink ratios. The tables below, Table 2.1 and Table 2.2, compare the values that we obtained from our implementation with the author's results for the ratio of foriegn domain pointing hyperlinks to total hyperlinks:

Threshold	Author Phishing (%)	Our Phishing (%)
10	77.92	96.65
20	75.64	93.73
30	73.05	93.02
36	71.42	93.09
40	68.99	92.84
50	62.01	92.49
60	49.02	92.39
70	40.90	90.60
80	31.98	90.29
90	20.12	89.57

Table 2.1: Threshold vs ratio of foreign pointing links for phishing pages

Threshold	Author Legitimate (%)	Our Legitimate (%)
10	31.11	86.27
20	19.75	74.20
30	6.91	68.32
36	1.48	67.02
40	1.48	66.08
50	0.98	59.22
60	0.49	53.09
70	0.25	50.83
80	0	49.47
90	0	48.08

Table 2.2: Threshold vs ratio of foreign pointing links for legitimate pages

Looking at Table 2.1, it is clear that the percentage of phishing sites for each threshold is much greater. This is also the case for the results found in Table 2.2. Despite the given threshold, phishing sites seem to be over 90% on average and legitimate sites seem to be around 60% on average.

From our results, we found that both phishing sites and legitimate sites have more foreign domain links on their pages as compared to the results found by the authors in 2016. Given the drastic changes in the internet over the course of the past 6 years, it is very possible that the legitimate websites that the average internet user accesses can point to an increased number of foriegn domain links as compared to the environment of the internet in 2016. Social media has become more and more prevalent in our society and with social media, users link to many other domains. This can be a valid reason for this large increase in the foreign domain percentages.

B. Comparison of Table 3

This section will involve the comparison of Table 3 from the paper [1]. Here, some basic measurements were taken from the extracted hyperlink set of each paper from the dataset. The main measurements under analysis are mentioned above in Part II of this paper: the number of webpages with a hyperlink set of size 0, the number of webpages that contain at least one null link, and the number of pages that point to a foriegn domain over some threshold value. These measurements are the basis for the phishing identification algorithm as described by Jain and Gupta As seen below, there are 4 tables. In the paper [1], the authors represent their results using whole values referring to the frequency of the measurements. Although this is clear quantitative information, we believed it would also be important to see percentages for this data. So, we included both the frequency representation of the data (Table 3.1a and Table 3.1b) as well as the percentages of these frequencies (Table 3.2a and Table 3.2b). Table 3.1a contains the data found from the author and table 3.2b shows the same data but from the results of our implementation. Similarly, Table 3.2a shows the results of the author and Table 3.2b shows our results.

Type	Total Instances	# No Links	# Null Links	# >= foreign domain threshold
Phishing	1120	279	245	440
Legit.	405	0	0	7

Table 3.1a: Author's results Hyperlink features at 20% threshold for foriegn links total number of instances

Type	Total Instances	# No Links	# Null Links	# >= foreign domain threshold
Phishing	1132	392	506	1061
Legit.	628	145	390	466

Table 3.1b: Our results Hyperlink features at 20% threshold for foriegn links total number of instances

From looking at the data in Table 3.1a and Table 3.1b, we can see that there are clear increases in the number of the measurements found for both legitimate sites and phishing sites. There is a moderate increase for the measurements found for the phishing sites and a massive increase in the value for legitimate sites. We will discuss more about our interpretations of the results in a later section of this paper.

We will now show the results in percentage values for a different perspective of the data.

Type	Total Instances	# No Links (%)	# Null Links (%)	# >= foreign domain threshold (%)
Phishing	1120	24.9	21.9	39.3
Legit.	405	0	0	0.17

Table 3.2a: Author's results Hyperlink features at 20% threshold for foriegn links percentage values

Type	Total Instances	# No Links (%)	# Null Links (%)	# >= foreign domain threshold (%)
Phishin	1132	34.6	44.7	93.7

g				
Legit.	628	23.1	62.1	74.2

Table 3.2b: Our results Hyperlink features at 20% threshold for foriegn links percentage values

To reiterate, the data in Table 3.2a and in Table 3.2b reference the same data in Table 3.1a and in Table 3.1b, the only difference is that now we are looking at the percentage values of the previous tables.
We will discuss more about our interpretations of the results in a later section of this paper.

C. Comparison of Table 4

In this section, we compare the results of Table 4 in the paper [1] under review with our results. This section looks at the accuracy of the model in determining whether or not sites are being classified correctly. More specifically, we are looking at the true positive rate, false positive rate, true negative rate, and the false negative rate.

As a review, we will describe each of these metrics as described in the paper [1]. The true positive rate is the at which phishing websites are correctly classified as phishing websites. The false positive rate is the rate at which phishing sites are incorrectly classified as legitimate sites. The false negative rate is the rate of legitimate sites that are incorrectly classified as phishing. Lastly, the true negative rate is the rate at which legitimate sites are correctly classified as legitimate. Additionally, there is a metric for accuracy which measures how accurate the classification of phishing and legitimate sites is. These calculations can be found in the paper [1]. More specifically, these calculations are referenced as Equations 2-6 from the paper [1].

Data	Author's	Our Results

Measurement	Results	
Total Phishing	1120	1128
Total Legitimate	405	573
Phishing website classified as phishing	964	1052
Phishing website classified as legitimate	156	76
Legitimate website classified as legitimate	398	188
Legitimate website classified as phishing	6	385
True Positive Rate	86.07 %	93.26%
False Negative Rate	1.48%	67.19%
Accuracy	89.38%	72.90%

Table 4: Comparison of the author's [1] results of their proposed phishing model at a 36% threshold.

As seen in Table 4 above, we compare our results with the results from the authors. At a glance, it seems that most metrics are similar. At a deeper inspection, we can see that the rate at which legitimate sites are being classified as phishing, also known as the false negative rate, is very high. The reason for this, as observed in previous sections of this paper, is that the number of foreign links on both legitimate and phishing sites have increased significantly. Though, the true positive rate is

slightly higher. The model seems to be classifying phishing sites quite accurately but it is misclassifying legitimate sites correctly. From this alone, there seems to be a need for some tweaking of the model to take into account changes in the hyperlink sets of legitimate sites.

D. Comparison of Figure 5

In the paper [1], Figure 5 is the graphical representation of the data from Table 2.1 and Table 2.2 in this paper. Below is Table 5 which is just a tabular aggregation of the data that we will then plot for comparison to the figure from the paper [1].

Threshold (%)	Phishing	Legitimate
10	96.65	86.27
20	93.73	74.20
30	93.02	68.32
36	93.09	67.02
40	92.84	66.08
50	92.49	59.22
60	92.39	53.09
70	90.60	50.83
80	90.29	49.47
90	89.57	48.08

Table 5: Hyperlink threshold versus ratio of webpages pointing to a foreign domain

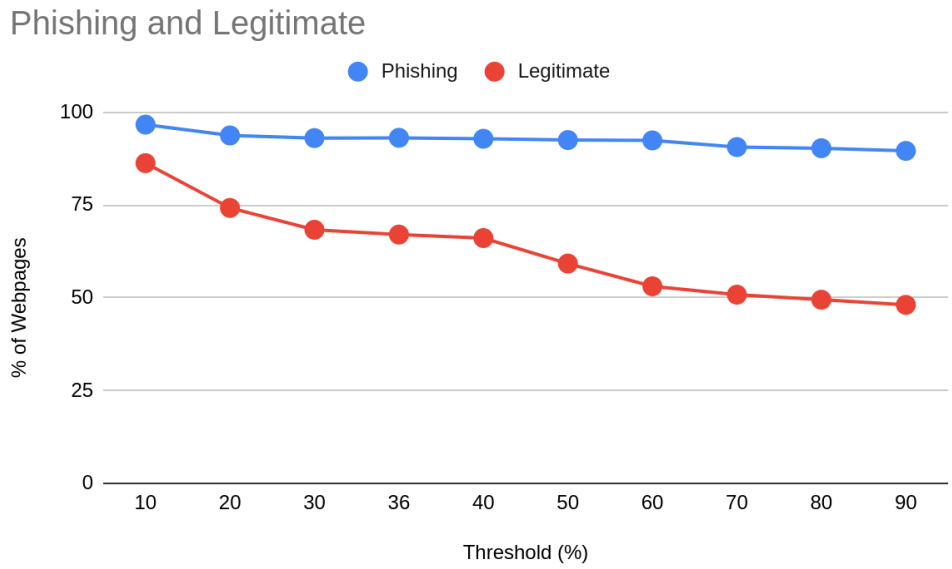

Fig. 2: Our results showing the hyperlink threshold versus percent of webpages pointing to a foreign domain. This is the same data representation as seen in Fig. 5 from the paper [1].

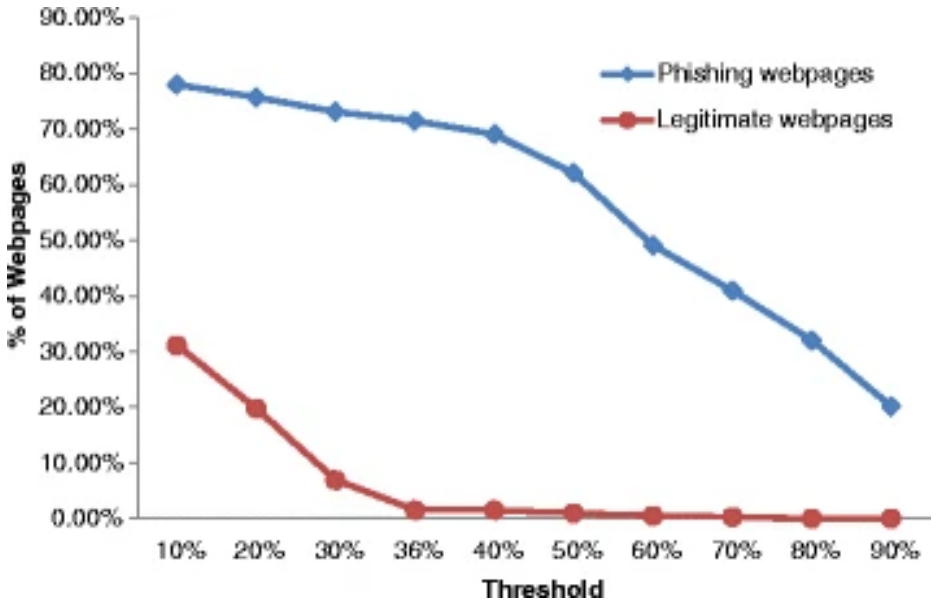

Fig. 3: The paper's [1] results showing the hyperlink threshold versus percent of webpages pointing to a foreign domain. This is the same data representation as seen in Fig. 5 from the paper [1].

As seen in Fig. 2 and Fig 3. above, the results for the legitimate websites are highlighted in red and the phishing sites are reflected by the blue lines. Comparing the graphs, it is clear that the results obtained in our implementation are skewed for both

metrics. First, the results for phishing sites is nearly constant in our implementation's results and the results for legitimate sites are somewhat similar to the author's [1] results but shifted up by roughly 50%. The reason for this is identical to the reasons previously mentioned, that being the increased number of foreign domain pointing hyperlinks from both legitimate and phishing sites from our dataset.

E. Additional Results

In addition to the results found in the paper [1], we thought it would be important to see the breakdown of the classification metrics mentioned earlier in this paper for the various threshold values chosen to test on. To be clear, these metrics are the true positive rate, false positive rate, false negative rate, and true negative rate.

Threshold (%)	True Positive Rate	False Positive Rate	False Negative Rate	True Negative Rate
10	96.91	3.09	86.41	13.59
20	93.99	6.01	74.36	25.64
30	93.29	6.71	68.66	31.34
36	93.26	6.74	67.19	32.81
40	93.19	6.81	66.43	33.57
50	92.76	7.24	59.41	40.59
60	92.48	7.52	53.29	46.71
70	91.40	8.60	51.25	48.75
80	91.62	8.38	49.89	50.11
90	90.72	9.28	48.5	51.50

Table 6: Comparison of metrics found in our implementation's results.

As seen in Table 6 above, we have the breakdown of these metrics for each of the thresholds analyzed. It is worth noting that these threshold values were the same values that the paper [1] utilized. As seen in this table, the true positive rate stays above 90% for all of the threshold cases. This is indicative of the model classifying phishing pages with a high degree of accuracy. However, it is also clear that the False negative rate is still quite high. This is the rate at which legitimate websites are classified as phishing sites. It is obvious that when the threshold value is low, 10% for example, the percentage of foreign domain links for a legitimate site or any site is much easier to be reached than it is for a higher threshold. The author's data in Table 4 of their paper [1] which can be seen in Table 4 of this paper, shows that at a threshold of 36%, the true positive rate is about 86.07% and the false negative rate is at around 1.48%.

Looking at Table 3 from the paper [1], we see that the legitimate sites tested in the model only failed on crossing the foreign domain threshold. Our results show that legitimate sites are now showing a greatly increased number of foriegn domain links and also no hyperlinks in some cases as well as null links. This shift has increased the false negative rate. As mentioned a few times in this paper, the differences in the environment of the internet when this paper was first published, 2016, and now are quite different. Social media sites have continued to dominate the internet but other sites have also become the norm. This vast amount of time, relative to the life of the internet itself, can be the driving force for these large disparities between the data.

VI. Interpretation and Discussion of Results

Now that we have shown the data that we collected from our implementation of the phishing detection model, we can now dive into the interpretation of the results.

To begin, it is clear that there are some disparities in the data that we collected compared to the data the author's published. Throughout this paper, we have seen that one of the metrics that the author's used heavily in the classification of a webpage is the number of foriegn domain links on a given webpage. The authors appear to use this metric to help determine if a page is phishing. It makes sense how this can be used to determine if a page is phishing; the owner of a phishing site wants the user to be on their domains where they can obtain the private user data that susceptible users leak themselves. The more links that point to domains other than the domain of the webpage can then be inferred to be additional domains that belong to the phishing mastermind. However, the nature of the internet has changed a lot since 2016. Additionally, with the work-from-home era still in full-swing, many people have been becoming interested in the internet for both work and leisure. These changes in the nature of the internet have made some of these metrics questionable for their use in the classification of a site as phishing or legitimate.

We observed that foreign domain pointing links were the most prevalent for determining a site as phishing. This can be seen in tables 3.1a/3.1b and 3.2a/3.2b. This metric follows the results of the author, however, the ratios of these metrics triggering the classification as a phishing site was much greater than that of the author's results. The other metrics were also inflated in our results as compared to the authors. Based on this, we feel that the metrics used seem to overinflate and thus incorrectly classify legitimate sites as phishing. This in turn results in a high False Negative rate. The author's were obtaining a False Negative rate of about 1.5% whereas we are obtaining a False Negative rate of about 67% (Table 6) at the same threshold value of 36%. However, the True positive rate is very high at around 93% in our results. This is comparable to the roughly 86% true positive rate the author's obtained. From this data, we can see that the classification of phishing is quite accurate. From the phishing sites that we sampled, this model seems to be able to accurately classify phishing sites, but has a very high false negative rate for correctly classifying legitimate sites.

One thing that we realized that could have played a role in skewing the results was the sample size of the legitimate webpages. According to the paper [1], the author's ran their model with only about 405 legitimate webpages as compared to the roughly 1102 phishing webpages. There were about twice as many phishing sites than the legitimate sites in the calculations. This could have skewed the results for the classification results. Since the authors do not include a more aggregated data sample size, it is hard to compare our results with a more unified sample size to the authors. This would have been important to observe since, again, if the sample sizes for the legitimate pages are equal, then the results would not be oversaturated by the data with the larger sample size.

The main culprit for these large disparities for the data is likely due to the authors' dataset for legitimate pages. As mentioned in the paper [1], the authors installed a chrome extension on the computers in the graduate laboratory on their campus. Thus, these computers were used by graduate students for academically related activities and internet traffic. It is likely that the sites these graduate students visited were likely to be domains from the university and other universities. We used some of the world's most popular websites as legitimate sites. The difference here is that the university sites or sites visited in the graduate laboratory like are for research, school, etc where the domains visited are likely the same and not pointing to many foreign domains. However, for large corporations and companies that operate globally, it is likely that they implement servers in many different locations

around the world making the whitelist approach ineffective for legit sites. Thus, it is clear that a wide range of smaller websites or local domains that the graduate students visited better suited the model for hyperlink extraction that Jain and Gupta proposed. For instance, less links pointing to foreign domains. The problem with this is that large legitimate sites do not work well with these features analyzed and are not very comparable to smaller sites. Thus, phishing sites are still accurately classified but legit sites cannot be.

VII. Conclusion

After implementing the phishing detection model described in the paper [1] by Jain and Gupta, it is clear that the effectiveness of this model in the current age of the internet has deteriorated. More specifically, the correct classification of legitimate web pages has diminished greatly to the point where more than half of legitimate web pages are being classified incorrectly as phishing sites. On the other hand, phishing webpages are being classified correctly at a solid rate of about 90%. As discussed many times already, the nature of the internet has evolved drastically since the release of the phishing detection model [1]. The model itself is accurate at detecting phishing sites but it is not effective at filtering out legitimate sites with the hyperlink features the authors rely on in the classification of a site. As one could imagine, it is possible to improve this phishing detection model through a modern analysis of the current trends in web page hyperlink features through research. Though other features appear to be necessary to do so rather than just the hyperlinks of a web page. As mentioned before, this phishing detection model needs more features to better classify legitimate sites since there legitimate sites come in many different shapes and sizes. All in all, this paper provides a solid understanding on one method of phishing detection and provides a simple yet effective way of singling out phishing sites.

References

[1] Jain, A.K., Gupta, B.B. A novel approach to protect against phishing attacks at client side using auto-updated white-list. *EURASIP J. on Info. Security* **2016,** 9 (2016). https://doi.org/10.1186/s13635-016-0034-3

[2] Shi, F., 2020. *Threat Spotlight: Coronavirus-Related Phishing*. [online] Barracuda. Available at: <https://blog.barracuda.com/2020/03/26/threat-spotlight-coronavirus-related-phishing/> [Accessed 10 April 2022].

[3] Phishtank.org. 2022. *PhishTank | Join the fight against phishing*. [online] Available at: <https://phishtank.org/> [Accessed 10 April 2022].

[4] Moz. 2022. *Top 500 Most Popular Websites*. [online] Available at: <https://moz.com/top500> [Accessed 10 April 2022].

[5] Sooter, B., 2016. *GitHub - bensooter/URLchecker: Test URL Filtering against a list of sites.*. [online] GitHub. Available at: <https://github.com/bensooter/URLchecker> [Accessed 16 April 2022].

[6] Huss, N., 2022. *How Many Websites Are There in the World? (2022) - Siteefy*. [online] Siteefy. Available at: <https://siteefy.com/how-many-websites-are-there/> [Accessed 18 April 2022].

[7] Morton, B. and Chehal, J., 2022. *GitHub - brend3n/CAP6135-Phishing*. [online] GitHub. Available at: <https://github.com/brend3n/CAP6135-Phishing> [Accessed 22 April 2022].