A Study of Anti-Phising Methodologies and Phishing Detection Algorithms

Beatrice M. Cerda (email: cerdab1@gator.uhd.edu)
Shengli Yuan (email: yuans@uhd.edu)
Department of Computer Science and Engineering Technology
University of Houston - Downtown
Houston, Texas, USA

Abstract - Phishing is a way for attackers to convince users to believe the information provided is legitimate and thus voluntarily hand over personal or sensitive information. In many of the phishing attacks, phishers send emails to either random individuals in large groups, or in the case of spear phishing, a set of specific individuals. These emails contain languages that lure the victims or pressure them to click a special hyperlink, and take them to a carefully crafted website that resembles a legitimate one, where the victims are deceived to hand over their private or sensitive information.

We studied different techniques that detect phishing websites, ranging from list-based approach to behavior model approach, to methods using machine learning with natural language processing. We developed two simple algorithms to detect hyperlinks that lead to phishing websites. This work forms a foundation for further study in phishing detection.\(^1\)

Keywords - phishing, anti-phishing, hyperlink, machine learning.

1. INTRODUCTION

Phishing is a form of cyber-attack based on social engineering. In many of the phishing attacks, phishers send emails to either random individuals in large groups, or in the case of spear phishing, a set of specific individuals. In either case, these emails contain languages that attempt to lure the victims or pressures them to click a special hyperlink, and take them to a carefully crafted website that resembles a legitimate one, where the victims are deceived to hand over their private or sensitive information [1]. Efforts are underway to develop tools that help users to identify potential phishing traps, and to block the phishing websites [2]. Trainings, surveys, and psychology studies of user reactions to phishing websites are also being conducted [3].

There have been a number of phishing detection approaches proposed and developed, ranging from the traditional list-based approach where a list of phishing sites is recorded and referenced [5], to the more current detection and impersonated entity discovery techniques that utilize conditional random field, latent Dirichlet allocation (LDA), machine learning and natural language processing to facilitate the detection of zero-day phishing sites [6]. This paper will review the current trends in anti-phishing approaches, discuss popular methods, and briefly cover areas of special interest for future research. A heuristic hyperlink-based algorithm, and one based on machine learning were developed and explored.

This paper serves as a foundation for further study of phishing detection, and for inspiring interest in the areas of machine learning and natural language processing. The rest of the paper is organized in the following manner: Section II discusses related work in anti-phishing methodologies; Section III describes two algorithms we developed and implementations; Section IV analyzes the data retrieved from the simulations; Section V discusses further research and conclusion.

2. SURVEY OF RECENT WORK

Aravindhan et al. discussed the ways phishing can be performed and the tactics attackers use to obtain information [5], which include email-to-email, email-to-website, website-to-email, website-to-website, and browser-to-website. In all cases, phishers use social engineering tactics such as luring and intimidation to convince users to respond, and take them to a carefully crafted website that resembles a legitimate one, where the victims are deceived to hand over their private or sensitive information. A number of tools and websites are now available that help users to identify phishing sites. Here are some examples [5]

- Web of Trust
- PhishTank
- PineApp
- Mozilla Thunderbird

\(^1\) Part of the early work of this paper was presented to the Conference for Industry and Education Collaboration (CIEC 2018), San Antonio, TX, February 7-9, 2018.
- GeoTrust
- SpamAssassin
- ISP-provided tools such as Earthlink’s SpamBlocker

The list-based-approach uses lists to record the Uniform Resource Locators (URLs) that belong to either phishing or non-phishing web pages. Identification of URLs that are in the list can reach 100% accuracy. The challenge is the inability for lists to be updated timely. [5].

Heuristic approaches use one or more characteristics of a website to identify phishing. The characteristics may be the hyperlink, the URL, the web page content, or something else belong to the website. Machine learning techniques may be used to train the phishing detectors to locate specific characteristics. For example, for a hyperlink-based heuristic approach such as LinkGuard [7], we may use special characters, number of dots, an IP address and several other features in the hyperlink to determine whether the site is authentic or illegitimate (Figure 1). We can also take a hyperlink and extract the domain name to compare it against a page rank of other extracted domain names. If the ranking is largely different, then the extracted domain is considered a phishing site. Additional tools of this category include PhishChecker [1], CANTINA [2], CANTINA+ [8], and PILFER [9].

The certificate-based approach uses features taken from the public key certificate of a website. The features include subject name, domain name, date of initial validity, weather it was a self-signed certificate, along with many other characteristics. As stated in [10], most phishing sites are dynamic. This approach makes it costly for attackers to register and maintain their phishing sites.

The content-based approach may also check the images embedded in webpages. Using optical character recognition, it can convert images into text. In most cases, once the image is converted, the text is assessed and placed in a page ranking algorithm to decide if the website is authentic [5] [11]. GoldPhish is a tool that implements this approach.

CANTINA takes the concept of lexical signatures (this concept was originally used to fix broken links), and finds the top five most frequently used words in the web page. The top five words are then inserted into a search engine to find the authentic website [2].

In the message-ID based approach, the email header and the message-ID is extracted and evaluated using n-gram analysis (concept related to natural language processing), including various machine learning algorithms such as online confidence weighted learning along with a ten-fold cross validation on varying data sets. Testing proved detection rates can reach above 99% [12]. Phish-Identifier is a tool using this approach.

Social profile cloning is a form of impersonation where an attacker creates a phishing site containing a profile that resembles a target profile for the purpose of deceiving the victims. The social profile cloning approach helps detect such cloning thus the phishing sites [13]. Similarly in the phishing target discovery approach, when site cloning is detected, the owner of the legitimate website is warned [14] and [15].

In the behavior model approach, submission forms with random inputs to a web site and its corresponding responses are captured to build a Finite State Machine (FSM), i.e., the behavior model. This model is then verified with a set of heuristics to identify whether it is a phishing site. Phish Tester uses this approach. It works well with text-based sites but fails to catch illegitimate sites that contain embedded objects [16].

![Figure 1. Heuristic-based Approach for Phishing Site Detection Using Hyperlink Features [4]](image-url)

In the field of artificial intelligence, neuro-fuzzy refers to combinations of artificial neural networks and fuzzy logic [17] [18]. A hybrid neuro-fuzzy system is a fuzzy system using learning algorithms from gradients and/or heuristic learning strategies to find parameters such as fuzzy sets through inputs and outputs [19]. With the hybrid neuro-fuzzy approach, five inputs are used to detect phishing sites and improve accuracy:

- legitimate site rules
- user-behavior profile
- phish tank
- user-specific sites
- pop-ups from emails

A process is followed with 1) identifying and extracting features based on the five inputs; 2) developing a hybrid neuro-fuzzy model; 3) training the fuzzy inference; and 4) catch phish sites by output of neuro-fuzzy model [20] [21].

To detect target phishing sites, a machine learning and natural language processing based approach utilizes Conditional Random Field (CRF) and Latent Dirichlet
Allocation (LDA). The detector has three stages: 1) feature extraction; 2) phishing classifier; and 3) impersonated entity discovery [22][23][24].

A multi-tier phishing email classification was introduced in [25]. This approach uses relevant features built on message content and message header. Classifications stage one contains two different classifiers that classify suspicious email in parallel. If both results are positive, then it is labeled a phishing email. If both result are negative, they are both labeled legitimate site. If one or the other are positive, then it is considered misclassified.

3. PHISHING DETECTION ALGORITHMS

For this section, we developed and implemented two phishing detection algorithms employing techniques discussed earlier. The first one used the list-based approach combined with the hyperlink-based heuristic approach as described in [5] and [7].

When clicked or tapped on, a hyperlink embedded in an email or text message directs a user to an online resource such as a webpage where the attacker may have set up as a phishing site. A simple hyperlink takes in the following form [26]:

```plaintext
<a href="URL"> Anchor Text </a>
```

Within it the URL is broken down into parts. The first part identifies the protocol used, for example http or https. The second part identifies the IP address or the domain name where the said resource is located. In our algorithm, we first compared the hyperlink with a whitelist that recorded the known legitimate websites [27]. For a hyperlink that was not in the list, we looked for special texts within the extracted hyperlink, and flagged warning signs if we identified the following information:

- **Short domain names:** Companies like Bitly or Google offer the popular shortened domain name service. Attackers are using them to mask the real identities of their malicious websites. For example, if an attacker were to register with Bitly, their domain name would be: https://www.bitly.com/HJ63I78/. It would then be very difficult for a naïve user to identify a phishing site from this URL.

- **Protocol Identifier:** The two most common protocols used to surf the web is https which denotes a secure session, and http which is non-secure. If an email contains languages for serious business, yet the URL uses http as the protocol, our algorithm takes it as another warning sign for a possible phishing link.

- **IP address:** Legitimate site owners are motivated to use clear and easy-to-remember domain names in their URLs, while an attacker sometimes uses an IP address, instead of a domain name in order to deceive the victim.

- **ASCII encoded characters:** Sometimes a domain name has a % symbol after every character that requires decoding to obtain the IP address or the domain name. These domain names are normally very long, and attackers may use them to mask the real domain name, while legitimate URLs try to avoid encoded characters. Therefore encoded characters in a URL is a major warning sign.

- **Special character @:** Special characters in a URL such as @ can be used to redirect the visitor to a phishing site. For example, http://www.legit.com:xyz@www.phishing.net takes the visitor to www.phishing.net instead of www.legiti.com.

- **rDirl:** Redirects can be performed in many ways. The attackers may utilize the vulnerabilities of the target website to redirect users to a phishing site or to launch a CSS (cross site scripting) attack. After a URL is read and encounters the rDirl command, the website will be redirected to the link displayed following the statement.

- **Anchor Text:** We compare the URL against the Anchor Text because most attackers try to hide their domain name by placing an anchor text that reflects the target’s domain name. By inserting the names into two separate vectors, we were able to compare them and determine any discrepancies.

![Flowchart for our Phishing Detection Program using Machine Learning](image)

**Figure 2. Flowchart for our Phishing Detection Program using Machine Learning**

<table>
<thead>
<tr>
<th>Features</th>
<th>Categories</th>
</tr>
</thead>
<tbody>
<tr>
<td>Information in Email</td>
<td>Abnormal Based Features</td>
</tr>
<tr>
<td>Length of URL</td>
<td>Address Bar Based Features</td>
</tr>
<tr>
<td>Subdomain, MultiDomain</td>
<td>Address Bar Based Features</td>
</tr>
<tr>
<td>IP Present</td>
<td>Address Bar Based Features</td>
</tr>
<tr>
<td>Short URL</td>
<td>Address Bar Based Features</td>
</tr>
<tr>
<td>At Present</td>
<td>Address Bar Based Features</td>
</tr>
<tr>
<td>Double Slash</td>
<td>Address Bar Based Features</td>
</tr>
<tr>
<td>Favicon</td>
<td>Address Bar Based Features</td>
</tr>
<tr>
<td>Adding Prefix, Suffix</td>
<td>Address Bar Based Features</td>
</tr>
<tr>
<td>HTTPS Present</td>
<td>Address Bar Based Features</td>
</tr>
<tr>
<td>Domain Token Count</td>
<td>General Features</td>
</tr>
<tr>
<td>Path Token Count</td>
<td>General Features</td>
</tr>
<tr>
<td>Average Domain Token Length</td>
<td>General Features</td>
</tr>
<tr>
<td>Large Domain Token Length</td>
<td>General Features</td>
</tr>
<tr>
<td>Average Path Token Length</td>
<td>General Features</td>
</tr>
<tr>
<td>Website Forwarding</td>
<td>HTML &amp; JavaScript Based Features</td>
</tr>
<tr>
<td>Sess Present</td>
<td>Keyword Based Features</td>
</tr>
<tr>
<td>Length of Host</td>
<td>Lexical Features</td>
</tr>
<tr>
<td>Length of Path</td>
<td>Lexical Features</td>
</tr>
<tr>
<td>Number Dots Host</td>
<td>Lexical Features</td>
</tr>
</tbody>
</table>

**Table 1. Itemized list of features and their category used in training several networks.**

For comparison and possible improvement, we also developed a phishing detection program based on machine learning. This program had two parts, the Feature Extractor...
and the Deep Feed Forward Artificial Neural Network (Figure 2). First a list of URLs is fed into the Feature Extractor, who took a defined checklist of features and inspected each URL to see what features it had. Once it identified all the features of a URL, it then output a value for each of the features onto an Excel spreadsheet. 80% of this dataset was then used to train the Deep Feed Forward Artificial Neural Network, and the remaining 20% was used for testing. Table 1 contains the list of the features that were extracted and used for training and testing. The training was done using only one feature at a time, so that we could evaluate each feature individually and see how effective each feature was in detecting phishing URLs.

4. RESULTS AND ANALYSIS

For testing of the first program, we used samples of embedded links of legitimate sites as well as phishing sites compiled by journals and publications. The program correctly detected 80% of both types of links at high speed. This is a reasonable result given the simplicity of the first program.

For the second program, our models using only the individual features yielded an average accuracy below 50% (Figure 3). Initially we suspected this low accuracy was caused by the unbalanced dataset where some features had very low occurrences while others had very high occurrences. But further analysis revealed that there was no correlation between the accuracy and the occupancy of a feature. So the reasonable causes for this unsatisfactory result may be, 1. We missed one or more important features which could identify phishing site; or 2. Any single feature could not provide sufficient information for reliable phishing link detection. Further study are needed.

5. CONCLUSION AND FURTHER RESEARCH

Phishing attacks are serious threats to users. Finding an effective way to detect phishing sites is important. In this paper we surveyed the different approaches that detect phishing websites, ranging from list-based approach to behavior model approach, to the more advanced methods using machine learning with natural language processing. We developed two phishing link detection programs. The first one used the list-based approach combined with the hyperlink-based heuristic approach. This simple program was able to detect the majority of phishing sites at high speed during simulation. Yet its simplistic approach could lead to high false positive; and its performance also highly relies on the blacklist and whitelist being up-to-date. The second program used machine learning. The training was done using one feature at a time and the results were unsatisfactory. We believe that we either missed one or more important features, or any single feature cannot provide sufficient information for reliable phishing link detection. So for immediate future work, we will expand the feature list, and perform training using group of features rather than individual ones. We expect better accuracy from these updates. We will also explore better and more efficient heuristics for the first program.

In longer term, we will continue exploring ways to detect phishing sites efficiently, and to increase the true positives while keeping to a minimum the false positives, and all together removing any false negative.

6. REFERENCES


[15] Liu Wenyin, Gang Liu, Bite Qiu, and Xiaojun Quan, “Anti phishing through Phishing Target Discovery” Website Security –City University of Hong Kong.


[17] Hossain Shahrir, Mohammad Zulkernine “Trustworthiness testing of phishing websites: A behavior model-based approch” Future Generations Computer Systems School of Computing, Queen’s University, Kingston, Canada -2010


