Identification of Phishing Web Pages and Target Detection

Purnima Singh
M. E. Student (Computer Engineering)
MGM's College of Engineering
Navi Mumbai, India

Manoj D. Patil
Assistant Professor (Computer Engineering)
Datta Meghe College of Engineering
Navi Mumbai, India

Abstract- **Phishing attacks involve the imitation of web pages of legitimate organization in order to steal user identities. While there is urgent need to stop this kind of identity theft, current phishing methods are neither complete nor appropriately responsive. In proposed method, we first identify whether the given web page is phishing or not based on a large set of heuristic extracted from related work and if that web page is found to be phishing, we detect the most probable phishing target of that web page using Google Search API.**

Keywords: Anti-phishing, Phishing Target, Website features, Website Security.

1. INTRODUCTION

A phishing web page mimics a certain legitimate web page with an intention of luring people to visit the fake website and stealing their personal information such as usernames, passwords and the details of credit cards. The legitimate/true webpage mimicked by fake web page is defined as phishing Target and the fake web page as the Phishing page [2].

A lot of solutions have been developed to detect whether a web page is phishing page or not. However determining phishing target automatically is somewhat difficult for a machine although it is easier for human being [2]. Additionally a few phishing targets are less popular or new web pages, in which case, experienced professionals have difficulty distinguishing between phishing page and the target [1].

The need to automatically discover a target is important problem for anti-phishing efforts. If we correctly identify a target, we can confirm which web pages are phishing pages. We can also alert the target owners of phishing attacks so that they can take necessary action. Phishing has a huge negative effect on organization's revenues, customer relationships, marketing efforts and overall corporate image.

2. RELATED WORK

1. **Black/White Lists:** A white list contains URL's of legitimate sites and black list contains URL's of known phishing sites. Examples: Phish Tank [7], Site Checker, Google Safe Browsing [9].

2. **Visual Similarity:** In this method, the user has to register with the system the true web page. Then algorithms are applied to compute visual similarity [10].

3. **Content based Approach:** It detects phishing web page on the basis of the term frequency-inverse document frequency (TF-IDF) information retrieval system. TF-IDF score is calculated for each term in webpage and then taking five terms with highest scores, a lexical signature is generated. Then it is submitted to goggle to get search results. If page's domain name not falls into it, then it is phishing website [4].

4. **Semantic Link Network:** This approach first finds the associated web pages and then constructs an SLN from those web pages. Then a mechanism based on reasoning identifies whether the given page is phishing and find the target [2].

5. **Identity Discovery and Keyword Retrieval:** This method first find an identity based component to acquire identity of web page and then identity is used as query in search engine. If web page's

domain is not listed in search, Then It is a phishing web page [11].

6. **Web Communities:** A community on the web is defined as a set of sites that have more links to members of the community than to non-members. Members of such a community can be efficiently identified in a maximum flow minimum cut framework, where the source is composed of known members and the sink consists of well known non-members. A focused crawler that crawls to a fixed depth can approximate community membership by augmenting the graph induced by a crawl with links to a virtual sink node [1].

3. PROPOSED WORK

After reviewing previous work on phishing, we can extract many features that distinguish a phishing web page from a legitimate web page. Firstly, we examine if a webpage holds text fields, because a phishing webpage asks users to input personal information through those fields. If the webpage has at least one text field, we continue to extract other features. If the web page is found to be phishing then we detect phishing target of that web page.

3.1 Forms

This heuristic checks if a page contains text fields asking for personal data from people, such as password and credit card number. We scan the HTML for <input> tags that accept text and are accompanied by labels such as “credit card” and “password”. Most phishing pages contain such forms asking for personal data [4].

3.2 Age of Domain

The blacklist may succeed in protecting the users if it works on the domain level not on the URL level i.e. add the domain-name to the blacklist not the URL address. Thus, blacklisting those domains will in-turn adds the legitimate websites to blacklist as well. Even though the phishing website has moved from the domain, legitimate websites may be left on blacklists for a long time; causing the reputation of the legitimate website or organization to be harmed. Some blacklists such as “Google’s Blacklist” need on average seven hours to be updated. We find that the minimum age of the legitimate domain was 6 months. For this feature, if the domain created less than six months, it is classified as “Phishy”; otherwise, the website is considered “Legitimate” [3].

Proposed Rule:

Age of domain is ≥ 6 months → Legitimate

Otherwise → Phishy

3.3 Sub-Domain and Multi Sub-Domains

Assume that we have the following link http://www.hud.ac.uk/students/portal.com. A domain-name always includes the top-level domain, which in our example is “uk.” The “ac” part is shorthand for academic, “.ac.uk” is called the second-level domain, and “hud” is the actual name of the domain. We note that the legitimate URL link has two dots in the URL since we can ignore typing “www.”. If the number of dots is equal to three then the URL is classified as “Suspicious” since it has one sub-domain. However, if the dots are greater than three it is classified as “Phishy” since it will have multiple sub-domains [3].

Proposed Rule:

Dots in the domain part < 3 →Legitimate

Else if dots in domain part = 3 → Suspicious

Otherwise→ feature = Phishy

3.4 Long URL

Long URLs commonly used to hide the doubtful part in the address bar. Scientifically, there is no reliable length distinguishes phishing URLs from legitimate ones. The proposed length of legitimate URLs is 75[8]. However, the authors did not justify the reason behind their value. To ensure accuracy of our study, we calculated the length of URLs of the legitimate and phishing websites in our dataset and produced an average URL length. The results showed that if the length of the URL is less than or equal 54 characters then the URL classified as “Legitimate”. On the other hand, if the URL length is greater than 74 characters then the website is “Phishy” [3].

Proposed Rule:

URL length < 54 → Legitimate

URL length ≥ 54 and ≤75 →Suspicious

Otherwise → Phishy

3.5 DNS Record

For phishing sites, either the claimed identity in not recognized by the WHOIS database [6] or founded cord of the hostname is not founded. If the DNS record is empty or not found then the website is classified as “Phishy”, otherwise it is classified as “Legitimate” [3].

Proposed Rule:

No DNS record for the domain → Phishing

Otherwise → Legitimate

3.6 Abnormal URL

This feature can be extracted from WHOIS database [6]. For a legitimate website, identity is typically part of its URL [3].

Proposed Rule:

If the host name is not included in URL → Phishy

4. SYSTEM ARCHITECTURE

Our architecture consists of two parts:

4.1 Validate URL
4.2 Finding target for the same

4.1 Validate URL

Algorithm

1. *Text Box:* If input URL contains one or more than one text box apply target discovery algorithm If not then go to step 2
2. *Domain Age:* If domain age is less than 6 months, apply target discovery algorithm else go to step 3
3. *Google safe browsing [9]*: If there is negative feedback then apply target discovery algorithm else go to step 4
4. *Whois Information:* If Whois information is present then go to step 5 otherwise apply target discover algorithm
5. *URL length:* URL length < 54 goto step 6 otherwise apply target discovery algorithm.
6. *Multiple dots:* If more than 3 dots apply target discovery algorithm otherwise step 7
7. *URL source present on webpage:* If other then domain apply target algorithm otherwise step 8
8. Valid URL

4.2 Target Discovery for suspected Phishing Web Page

We use Google API in following way-

Step 1: Generate Google Search API Key and Include JavaScript

```
<script
src="http://www.google.com/jsapi?key=DOMAIN"type=
"text/javascript"></script>

<script type="text/javascript">

   google.load('search','1');

</script>
```

Step 2: Add HTML Container for Web Search

```
<divclass="data"id="web-content"></div>
```

When user will write a query, a request will be made to Google Search using Custom Search API and the results are fetched. These results are then copied into the DIV.

Step 3: JavaScript to call Google Search API

We use JavaScript to call the Google Search API and copy the results in our container DIV.

In short, we perform following functions:

1. Create an object to connect Google Web search using class google.search.WebSearch.
2. Set a call-back function that will get call once the results for the search are fetched.
3. Call the execute() method with search query as argument.
4. In call-back function, iterate through the results and copy it to container DIV.

5. ADVANTAGES OVER OTHER METHODS

In contrast to the blacklist method, a heuristic based solution can recognize newly created phishing websites. Also in list based methods; we have to update lists frequently. We are using already existing services like Google Safe Browsing so that we can get more accurate result to detect phishing web pages. This method is very cost effective because there is no need to buy server or server space and our consumed resources are free up to certain level.

6. CONCLUSION AND FUTURE SCOPE

The accuracy of the heuristic-based methods depends on picking a set of discriminative features that could help in distinguishing the type of website. We don't only detect the phishing web page; we also describe why that page is a phishing web page. This may enhance our knowledge about phishing web pages. Furthermore we also detect target of a phishing web page which is most challenging problem in anti-phishing field. We have used goggle search optimization for getting more accurate result. We can further filter the result by using various algorithms. In the near future, we will use the rules produced by different algorithms to build a tool that is integrated with a web browser to detect phishing websites on real time and warn the user of any possible attack.

7. References

[1] Liu Wenyin, Gang Liu, Bite Qiu, Xiaojun Quan, "Antiphishing through Phishing Target Discovery", IEEE Internet Computing, 2012.

[2] Liu Wenyin, Ning Fang, Xiaojun Quan, Gang Liu "Discovering Phishing Target Based on Semantic Link Network", Future Generation Computer Systems, 2010.

[3] Mohammad, Rami, McCluskey, T.L. and Thabtah, Fadi Abdeljaber, "Intelligent Rule based Phishing Websites Classification", IET Information Security, 2013.

[4] Y. Zhang, J. I. Hong, L. F. Cranor, “Cantina: A Content based Approach to Detecting Phishing Websites”, ACM Press, 2007.

[5] Antiphishing Working Group, www.antiphishing.org/reports/apwg-reports.

[6] WhoIS. [Online]. Available from: http://who.is/ .

[7] PhishTank. [Online]. Available from: http://www.phishtank.com/ .

[8] Horng SJ, Fan P, Khan MK, Run RS, Lai JL, Chen RJ, et al., “An efficient phishing webpage detector. Expert Systems with Applications”, An International Journal, 2011.

[9] Google Safe Browsing, www.en.wikipedia.org/wiki/Google_Safe_Browsing.

[10] Wenyin Liu, Xiaotie Deng, Guanglin Huang, Antony Y. Fu, “An Antiphishing Strategy Based on visual Similarity Assesment”, IEEE Internet Computing, 2006.

[11] G. Xiang and J.I.Hong, “A Hybrid Phish detection Approach by Identity Discovery and Keywords Retrieval”, 2009.