Using Domain Top-page Similarity Feature in Machine Learning-based Web Phishing Detection

Nuttapong Sanglerdsinlapachai
Thai Computer Emergency Response Team
National Electronics and Computer Technology Center
Pathumthani, Thailand
nuttapong.sanglerdsinlapachai@nectec.or.th

Arnon Rungsawang
Massive Information & Knowledge Engineering
Dept. of Computer Engineering, Kasetsart University
Bangkok, Thailand
arnon@mikelab.net

***Abstract*—This paper presents a study on using a concept feature to detect web phishing problem. Following the features introduced in Carnegie Mellon Anti-phishing and Network Analysis Tool (CANTINA), we applied additional domain top-page similarity feature to a machine learning based phishing detection system. We preliminarily experimented with a small set of 200 web data, consisting of 100 phishing webs and another 100 non-phishing webs. The evaluation result in terms of *f-measure* was up to 0.9250, with 7.50% of error rate.**

Keywords-phishing; anti-phishing; machine learning; semantic similarity; domain top-page;

I. INTRODUCTION

The phishing attack is a fraud using a fake webpage to deceive users in giving their crucial information. These problems are increasing dramatically on the internet. According to the Anti-Phishing Working Group (APWG) survey [1], there were additional 9,635 phish pages in the second half of 2008. To mitigate the problem, preventing the users to visit those phish pages is an effective way. Some may filter out a lot of emails leading to the fraud sites which are the one of many attack vectors.

Blocking phish webs directly may be more efficient. There are two approaches to identify the phish pages. The first one is using a blacklist. Comparing the requested URLs with URLs in the list is a simple way to check that the target is legitimate or not. But the blacklist cannot cover all phish pages, because the fraudulent webs are newly created all the time.

The second one is called heuristic-base method. This approach aggregates various features synthesized from the target pages to judge whether it is a phishing or a real web page. Some researches such as [2] and [3] used the machine learning techniques to improve efficiency. This paper adapts CANTINA [4]'s heuristic features with a new additional attribute to six machine learning algorithms including Naïve Bayes (NB), Neural Network (NN), Support Vector Machine (SVM), Random Forest (RF), J48 Decision Tree and AdaBoost.

In Section II, we present the related work. We describe web features used for detecting the phishing and their concept in Section III. In Section IV, we define the evaluation metrics, show how to collect a dataset and prepare the experiments. Then we show our experimental results and discussion in Section V. We summarize our findings in Section VI and point out our future work.

II. RELATED WORK

Our related work in the literature can be separated into two groups. First group is the researches on finding features for detecting phish web pages, while the second group is on adapting the machine learning techniques to improve efficiency.

A. Heuristic Features for Detecting Phish Web

Early anti-phishing researches analyzed page source code or URL information to extract various features which could be used in comparison with known real page. CANTINA began to use an external resource, Google, to find real page and judge the suspect immediately. According to CANTINA's results, many researchers used this approach as a basis to develop new detection method. On the contrary, Kaigui et al. [5] presented another approach using external resources to identify the phish web. Their methods considered the credibility of the target site instead of finding the real page. All those heuristic features can become the attributes for training a computer to detect phishing automatically.

B. Using of Machine Learning Techniques

In 2006, Pan et al. [2] used a seven-featured SVM on 297 phish and 100 legitimate dataset. Their result's error rate was at 16%. To compare efficiency among many machine learning techniques, Daisuke et al. [3] used nine variant of learning methods with eight attributes from the heuristic features of CANTINA. Experimented using 1500 phish and 1500 legitimate web pages, the lowest error rate was 14.15% while the average was 14.67%. In addition, the highest *f-measure* is 0.8581 and the highest AUC, an area under the Receiver Operating Characteristic (ROC) curve, is 0.9342 in case of using AdaBoost.

In [3], the authors used only features from CANTINA. Adding or changing features may result in different efficiency. Following that hypothesis, this paper replaced some features of CANTINA with a new feature and tested with six different machine learning techniques. We used 100 phish pages and 100 legitimate pages dataset in our experiments.

978-0-7695-3923-2/10 $26.00 © 2010 IEEE
DOI 10.1109/WKDD.2010.108

III. PROPOSED ADDITIONAL PHISHING DETECTION FEATURES

In early works, the researchers defined many features for detecting the phishing web pages such as in [4] and [5]. Those features can depart to two main types: page information features and external resource features. The page information features use all information synthesized from URLs, source codes or web visual appearance to verify whether the page is a phishing or not. On the other hand, the external resource features consult the third parties to inform them with some clues for phishing identification. Major resources used to assist the classifiers are the web search engines such as Google or Yahoo. Other resources may be domain registrars, WHOIS services, and web categorized services. Most features can classify the webs by themselves, but there are some features that required legitimate pages' profiles for comparison. We analyzed those features presented in [4] and [5] to select them for use in our work.

A. Selected Features from CANTINA

According to CANTINA [4], there are eight features including their proposed Google assisted TF-IDF (term frequency/inverse document frequency). Their features are described as the following:

1) Age of Domain: This feature checks whether the domain name of the suspect is older than 12 months or not. To avoid web shutting down, the phishers may register their domain name for a short period before spreading emails with web link. It can infer that the web with a younger age has a higher potential to be a phish site.

2) Known Images: The phish pages must have very similar appearance like their targeted webs to deceive the users. If we compared the pictures between two sources, they should be very similar. To test this feature, the image processing must be used and all main pictures from target webs must be collected. These two processes make this feature hard to be implemented.

3) Suspicious URL: When the phishers try to trick the victims, the URLs of the phishing page may be modified to the pattern that is hard to check. As this feature, CANTINA find '@' or '-' signs in suspicious URLs which are often used to modify the URL.

4) Suspicious Links: Same as the above feature, this checks '@' or '-' in the content of links substituted for the URLs.

5) IP Address: For escaping from domain registration or user checking, the IP address is a simple way used to hinder from verification. Therefore, CANTINA checks whether the URL uses an IP address or not.

6) Dots in URL: CANTINA checks for five dots used in the URL. Many dots appearance may be caused by an attempt that the phishers use sub-domain to constuct a legitimate look of the URL or use a redirect script to bring the victim to another site.

7) Forms: To get victims' personal information, the phishers have to place the form with input blocks in their pages. This feature check the <*input*> tag in page's source codes especially the ones labeled with words such as *pin code* or *password*.

8) TF-IDF: This is the key feature of CANTINA [4]. It implements on assumption that the phishing sites' contents must be similar to the contents of real targeted webs. The authors of [4] used an information retrieval technique, TF-IDF, to get the content representative in form of only five words. Then they put all words with a domain name from URL to more reliable external resource like the search engine to assist. If that suspicious site is real, the search engine should be able to return its domain name.

Based on above features, we change *Forms* feature to be a filter for dataset selection. The reason is that we assume that the dangerous pages causing users lost their information must contain forms with input blocks. Furthermore we decide to leave *Known Images* out because this feature requires the updated collection of images from all sites which are the targets of the phishers. The last feature that we left out is *Age of Domain*. This feature has to contact a WHOIS service to get may-not-exist data of domain age. In addition, according to the APWG survey [1], more than 80 percents of phishing domains is compromised from existing webs. So the domain age may not be proper feature.

B. Domain top-page Similarity

From Kaigui's paper [5], the feature which we are interested is called "Web Category Comparison" or "WCC." The concept of this feature is to check whether a suspicious page is suited to be in its domain or not. That work used Yahoo! Directory [6] to categorize the suspect by comparing with its domain category. We decided to simplify this feature by using a similarity between suspicious page and top page of its domain instead. A cosine similarity was used to represent the similarity. The process to calculate a cosine similarity is described as follows:

- Find a term frequency vector of a suspicious page assign to vector *A* and vector *B* for its domain top-page.
- Calculate sizes of both vectors as in (1).

$$|V| = \sqrt{v_1^2 + v_2^2 + \ldots + v_n^2} \tag{1}$$

  where v_i is a number of the i^{th} term in document *V*.
- Then find a cosine similarity of two vectors via (2).

$$Cos(A,B) = \frac{A \cdot B}{|A| \times |B|} \tag{2}$$

The term $A \cdot B$ is the dot product of two vectors which equals to the sum of each orderly component pair products. If there is no domain top page, we set this feature to zero.

IV. EVALUATION APPROACH

For evaluating performance of our modified approach, we define the evaluation metrics commonly used to comparing classifiers. We also describe how to collect data for training and testing. Moreover, in this section, we will explain how to prepare our experiments.

A. Evaluation Metrics

For comparison of the classifiers, there are a number of evaluation metrics used commonly. These groups of metric relate to four types of the testing results consisting of:

- True Positive (*TP*) - The phish pages which were classed as phish page,
- False Negative (*FN*) - The phish pages which were classed as legitimate page,
- True Negative (*TN*) - The legitimate pages which were classed as legitimate page,
- False Positive (*FP*) - The legitimate pages which were classed as phish page.

For determining accuracy, the *f-measure* is chosen. This evaluation metric is defined as the harmonic mean of precision and recall. While precision p equals $TP/(TP+FP)$ and recall r equals $TP/(TP+FN)$, the *f-measure* is $2 \cdot p \cdot r/(p+r)$. Another evaluation metric used to measure accuracy is the error rate which is sum of incorrect classed page divided by all data. Therefore, the error rate equals $(FP+FN)/(TP+TN+FP+FN)$.

To compare an adjustment capability, the area under the ROC curve is used. By varying the threshold, the ROC is a curve plotted between TP rate ($TP/(TP+FN)$) and FP rate ($FP/(FP+TN)$). This metric show how many trade-off must be paid between security (TP rate) and annoyance (FP rate). If AUC was high, it indicated that trade-off between two rate values was small. However if AUC was low, the trade-off was high.

B. Dataset

The criteria for selecting URLs to create a dataset are according to Section III. Our phish pages were collected from the sites reported by CLEAN MX [7] during August 7th, 2009 to September 7th, 2009. We selected 100 phish pages with login-form for dataset. For legitimate pages, we use the Google with a key word "inurl:login+logon+ signin+signon", and choose another 100 URLs.

C. Experimental Set-up

In this paper, we divided our study into three experiments: to test CANTINA's reduced features, to test only the new feature, i.e. the proposed domain top-page similarity feature, and to test machine learning based method combining CANTINA's reduced features with the new feature. Each machine learning based experiment is performed by using four-fold cross validation.

1) Experiment I: In this experiment, we used five remaining features from CANTINA discussed in Section III to identify the phish pages. According to CANTINA, the weights of each feature for aggregration must sum into one. Therefore, we reweighed them based on their old weights as shown in "TABLE I." Considering the new weights, we can assume that this classification uses only TF-IDF because its weight is more than a half of all. The web pages that fail TF-IDF test will be classified as phish pages. For heuristic testing, we apply the machine learning techniques with the remaining features. To implement machine learning based experiment, we use the "WEKA" [8], an opensource software from the university of Waikato. The learning methods that we used are Naïve Bayes (NB), Neural Network (NN), Support Vector Machine (SVM), Random Forest (RF), J48 Decision Tree and AdaBoost. Next, we adjusted the configuration to meet the ones configured by [3] which used five hidden layer for Neural Network and 300 trees for Random Forest.

TABLE I. WEIGHTS OF FIVE REMAINING FEATURES FROM CANTINA

Feature	Weight	
	Old Weight	***New Weight***
Suspicious URL	0.01	0.02
Suspicious Links	0.00	0.00
IP Address	0.07	0.14
Dots in URL	0.13	0.27
TF-IDF	0.28	0.57

2) Experiment II: This experiment tested whether a new feature, i.e., domain top-page similarity, is suitable for detecting the phishing. Our heuristic test also tried to adjust a similarity theshold to find an acceptable point that classified web with the most accuracy.

3) Experiment III: To compare performance of our proposed model with new added feature, this experiment extends Experiment I by adding a domain top-page similarity into the feature vector and comparing the results.

V. RESULT AND DISCUSSION

According to previous section, we divided the results of the experiments into three parts related to each experiment.

A. Experiment I - Evaluation of Remaining Features

By using five remaining features from CANTINA, the results from four out of six machine learning methods outperform the heuristic method as shown in "TABLE II." The worst learning technique in this experiment is Naïve Bayes which cannot recognize any phish pages. The best one with the error rate 8.00% is Neural Network following by Random Forest (8.50%), Support Vector Machine (8.50%) and then AdaBoost (10.00%).

These results demonstrated that even if we altered features of CANTINA, the machine learning based method can outperform the heuristic method.

TABLE II. EVALUATION METRICS OF USING FIVE REMAINING FEATURES FROM CANTINA TO DETECT WEB PHISHING

Methods	Evaluation Metrics		
	f-measure	*The error rate*	*AUC*
Heuristic	0.8889	10.50%	-
NB	0.0000	50.00%	0.5000
NN	0.9170	8.00%	0.9340
SVM	0.9120	8.50%	0.9150
RF	0.9120	8.50%	0.9300
J48	0.8890	10.50%	0.8770
AdaBoost	0.8950	10.00%	0.9270

B. *Experiment II - Evaluation of Domain top-page Similarity*

The results shown in "TABLE III" were calculated from varying the threshold that was a boundary between phish and legitimate pages. We started at threshold which equals to 0.5 and stepped up and down with 0.25-length. When we noticed that the lower threshold gave a better evaluation value, we stepped with a half of 0.25-length and repeated until we find the finest point. With this dataset, our finest threshold is 0.125 which implies that the pages having the similarity less than 0.125 with the domain top-page will be classed as phishing.

TABLE III. EVALUATION METRICS OF USING DOMAIN TOP-PAGE SIMILARITY TO DETECT WEB PHISHING

Thresholds	Evaluation Metrics	
	f-measure	*The error rate*
0.750	0.7300	35.50%
0.500	0.7773	27.50%
0.250	0.8101	22.50%
0.200	0.8205	21.00%
0.125	0.8312	19.50%
0.100	0.8261	20.00%

C. *Experiment III - Evaluation of Feature-added Model*

When we added the domain top-page similarity as the extended attribute, the evaluation results of all machine learning algorithms were better as shown in "TABLE IV" comparing to "TABLE II." If we drilled down to each metric, the best accuracy was given by the Neural Network with 7.50% error rate and 0.9250 *f-measure*. But if we considered the AUC metric, the Random Forest gives the best result.

These experimental results pointed out that the more proper attribute for machine learning method, the more efficiency of detecting the phish pages.

The evaluation results of the method using TF-IDF plus search engine or domain top-page similarity as features, as experimented by our study, may change when the time is passed. The dataset must be recent to get the results more accurately.

TABLE IV. EVALUATING RESULTS USING EXTENDED DOMAIN TOP-PAGE SIMILARITY

Methods	Evaluation Metrics		
	f-measure	*The error rate*	*AUC*
NB	0.8100	22.50%	0.7780
NN	0.9250	7.50%	0.9560
SVM	0.9120	8.50%	0.9150
RF	0.9140	8.50%	0.9710
J48	0.8910	10.50%	0.9270
AdaBoost	0.9050	9.00%	0.9640

VI. CONCLUSION AND FUTURE WORK

Our research proposed a new attribute to improve efficiency of machine learning-based phishing detection. The new feature uses another part of concept, i.e., the domain top-page similarity, to test whether the page is phishing or not. It is easy to implement and can achieved with 19.50% error rate and 0.8312 *f-measure*. When we applied in learning methods, this additional proposed feature can boost accuracy to 0.9250 in term of *f-measure*.

In our future works, we plan to adjust existing feature extraction methods and feature weights, and seek for more relevant features to get a better result. Furthermore the method used to collect a dataset must be improved. Retrieved dataset should be able to use for testing a new algorithm at all time and have a large amount of data to guarantee that the developed method can be used in a realistic manner.

REFERENCES

[1] Electronic Publication: APWG's Global Phishing Survey: Trends and Domain Name Use in 2H2008, May 2009: http://www.apwg.org/reports/apwg_report_H2_2008.pdf

[2] Pan, Y., Ding, X., "Anomaly Based Web Phishing Page Detection". In *Proceedings of the 22nd Annual Computer Security Applications Conference on Annual Computer Security Applications Conference (ACSAC'06)*, Sep 2006.

[3] Daisuke, M., Hiroaki, H., Youki, K., "An Evaluation of Machine Learning-based Methods for Detection of Phishing". Australian Journal of Intelligent Information Processing Systems, vol. 10(2), 2008, pp. 20-39.

[4] Zhang, Y., Hong, J., Cranor, L. "CANTINA: A Content-Based Approach to Detect Phishing Web Sites". In *Proceedings of the 16th World Wide Web Conference (WWW'07)*, May 2007.

[5] Kaigui, B., Jung-Min, "Jerry" P., Michael, S. H., France, B., Janine, H., "Evaluation of Online Resources in Assisting Phishing Detection". In *Proceedings of the 9th Annual International Symposium on Applications and the Internet (SAINT2009)*, July 2009.

[6] Electronic Publication:Yahoo! Directory: http://dir.yahoo.com

[7] Electronic Publication: CLEAN MX realtime database: http://support.clean-mx.de/clean-mx/phishing.php

[8] Ian, H., Witten, F., Eibe, F., "Data Mining: Practical Machine Learning Tools and Techniques", 2nd ed., San Francisco: Morgan Kaufmann, 2005.