Learning based Malicious Web Sites Detection using Suspicious URLs

Haotian Liu, Xiang Pan, Zhengyang Qu
Department of Electrical Engineering and Computer Science
Northwestern University, IL, USA
Email: {haotianliu2011, xiangpan2011, zhengyangqu2017}@u.northwestern.edu

ABSTRACT

Malicious Web sites largely promote the growth of Internet criminal activities and constrain the development of Web services. As a result, there has been strong motivation to develop systemic solution to stopping the user from visiting such Web sites. In this paper, we propose a learning based approach to classifying Web sites into 3 classes: *benign, phishing, and malware.* Our mechanism only analyzes the Uniform Resource Locator (URL) itself without accessing the content of Web sites. Thus, it eliminates the run-time latency and the possibility of exposing users to the browser-based vulnerabilities. By employing learning algorithms, our scheme achieves better performance on generality and coverage compared with blacklisting service. Through extensive evaluation, the resulting classifiers obtain 97.53% accuracy on detecting malicious Web sites.

General Terms

Theory and Performance

Keywords

Machine Learning, Web Security

1. INTRODUCTION

Web applications all around the world become popular and bring people convenience, while there is a rapid growth in the number of attacks from various criminal enterprises, such as financial fraud and spam-advertised commerce. The common thread among those attacks is the requirement that unsuspecting users visit the sites, clicking the target Uniform Resource Locator (URL).

If we can be informed of the properties of the target URL in advance, in other words, whether it is dangerous or not, such problems will be largely resolved. Thus, many security communities have provided blacklisting service, which is based on various techniques including manual reporting, and Web crawlers with site analysis heuristics. However, a large portion of malicious web sites are too new to be checked or have not been blacklisted, due to the limited coverage capacity of blacklist compared with the huge number of Web sites. Besides, some client-side systems analyze the content of Web sites when they are visited, which employs run-time latency and exposes users to the browser-based vulnerabilities.

To address this problem, we develop a mechanism based on machine learning. Given the properties achieved by some techniques, it is capable of classifying URL intelligently without the client-side latency and approaching Web content on demand. Our work makes the following contributions:

- We formulate the model and extract features which are effective in URL classification.
- Our mechanism can predict whether the target URL is malicious precisely without detecting web content that incurs run-time latency. It achieves 97.53% accuracy on detecting malicious Web sites.
- We implement and compare various classification algorithms, e.g. Support Vector Machine (SVM), Logistic Regression (LR), and Decision Tree (DT).

The rest of this paper is organized as follows: We discuss details of our approach to URL classification and the corresponding implementation in section 2. Section 3 presents the evaluation of our mechanism. We review related works in section 4. Finally, section 5 concludes this paper.

2. APPROACH & IMPLEMENTATION

URLs of the websites are separated into 3 classification:
Benign: Safe websites with normal services
Phishing: Website performs the act of attempting to get information such as usernames, passwords, and credit card details by masquerading as a trustworthy entity in an electronic communication.
Malware: Website created by attackers to disrupt computer operation, gather sensitive information, or gain access to private computer systems.

2.1 Feature Extraction

Given single URL, we extract its features and categorize them into 3 classes:

Lexical Features: Lexical features are based on the observation that the URLs of many illegal sites look "different", compared with legitimate sites. Analyzing lexical features enables us to capture the property for classification

Table 1: Lexical Features

No.	Feature	Type
1	Length of hostname	Integer
2	Length of entire URL	Integer
3	Number of dots in URL	Integer
4	Top-level domain	Integer
5	Domain token count	Integer
6	Path token count	Integer
7	Average domain token length	Real
8	Average path token length	Real
9	Longest domain token length	Integer
10	Longest path token length	Integer
11	Brand name presence	Binary
12	IP address presence	Binary
13	Security sensitive word presence	Binary

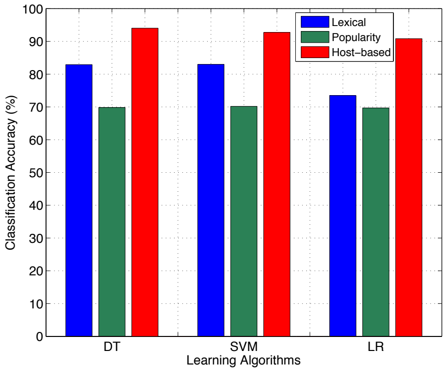

Figure 1: Detection Accuracy for each group of features

purposes. We first distinguish the two parts of a URL: the hostname and the path, from which we extract bag-of-words (strings delimited by '/', '?', '.', '=', '-' and '_') . Then we get the properties listed in Table I. Based on our study on 7071 URLs of phishing websites, 20976 URLs of benign websites, and 9285 URLs of malware websites, we find that phishing website prefers to have longer URL, more levels (delimited by dot), more tokens in domain and path, longer token. Thus, we choose the features 1, 2, 3, 5, 6, 7, 8, 9, 10 in Table I. Moreover, the top-level domain contains key information of the website, such as whether the website belongs to a commercial organization and in which country the website is registered. We extract the top-level domain in URL and transform the String to Integer by checking a hash map. Besides, phishing and malware websites could pretend to be a benign one by containing popular brand names as tokens other than those in second-level domain. Considering phishing websites and malware websites may use IP address directly so as to cover the suspicious URL, which is very rare in benign case, we extract feature 12 in Table 1. Also, phishing URLs are found to contain several suggestive word tokens (confirm, account, banking, secure, ebayisapi, webscr, login, signin), we check the presence of these security sensitive words and include the binary value in our features.

Site popularity Features: Intuitively, malicious sites are always less popular than benign ones. For this reason, site popularity can be considered as an important factor to measure a site' s reputation. In URLhelp, we adopt three features to describe sites' popularities. The first feature is the number of links pointing to that site, which can be acquired from Google. The second feature is the real traffic rank of that site. Since some malicious sites may adopt tricks like "link farm" to increase the number of links pointing to themselves, the second traffic feature is necessary: it is very hard to increase real traffic by using such tricks. Traffic rank feature can be acquired from Alexa.com. The third feature is a boolean feature, which indicates whether the domain is within a well reputable sites list. This site contains 1,000,000,000 domains with good reputation. It can be accessed from Amazon.com.

Host-based Features: Host-based features are based on the observation that malicious sites are always registered in less reputable hosting centers or regions. In addition, attackers are not inclined to leave sufficient information when registering a server for initiating attacks. Therefore, we include five host-based features in URLhelp: 1). the domain's autonomous system number; 2). the country its corresponding IP belongs to; 3). the number of registration information. 4) the number of resolved IPs and 5). if the domain contains valid PTR record. Some papers also use some other whois features such as the registration date, update date and expiration date. Even though we extract and store those features, our feature selection process didn't consider them as relevant features for determining if a site is malicious or not.

2.2 Date Set

We randomly collect 22,190 benign URLs from DMOZ Open Directory Project[1]. DMOZ is one of the largest human-edited directory of the world. It classifies URLs into different categories. Thus, random selection can guarantee our dataset ranging over different areas. As for phishing URLs, we collect 5,703 samples from PhishTank[2], a collaborative site where people can submit and verify phishing URLs. Besides, we select 9,220 URLs from DNS-BH project[3], a site creating and maintaining a list of domains known to be used to propagate malware and spyware. People can download the list for free.

2.3 Training

All of URLs in the dataset are labeled. We use 5-fold method to train-test our systems. Before training, we preprocess the features not consistent with others. For example, the range of traffic is much larger than that of other features. We map the feature into a much smaller range and it turns out to significantly increase the accuracy. We also use Chi-Square test and virtualization tool in Weka to select most informative features. After selecting features, we use three learning algorithms-J48 decision tree, logistic regression and support vector machine to train dataset.

3. EVALUATION

3.1 Feature Comparison

[1] http://www.dmoz.org/
[2] http://www.phishtank.com/
[3] http://www.malwaredomains.com/

Table 2: True Positive Rate (FPR) & False Positive Rate (FPR)

	DT			SVM			LR		
	Benign	Malware	Phishing	Benign	Malware	Phishing	Benign	Malware	Phishing
TPR	98.3%	99.9%	90.7%	98.9%	84.8%	44.0%	97.5%	100%	83.2%
FPR	3.6%	0.2%	1.1%	30.6%	0.7%	0.3%	6.4%	0%	1.8%

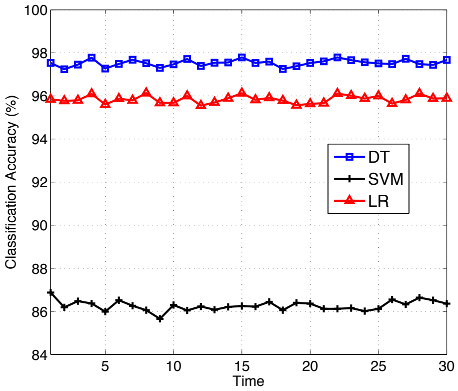

Figure 2: Classification Accuracy and Time Latency

In order to find how much each feature improves/reduces the performance of our mechanism on classification accuracy, we separate the features into 3 groups as lexical features, popularity features, and host-based features. Given each feature group, we utilize 3 learning algorithms as Decision Tree (DT), SVM, and Logistic Regression (LR) and get the classification results. From Figure 1, we can see that host-based features contributes the most to the classification performance among the 3 groups of features. Moreover, the effect of lexical features on classification accuracy will be largely reduced when LR is utilized.

3.2 Learning Algorithm Comparison

Given the features we extract and collect, we are motivated to compare the performance of the 3 learning algorithms. We conduct 30 tests on each algorithm and 5-fold cross validation is used in each test. Based on the results in Figure 2, we use t-test to compare the performance. At significance level of 0.05, DT is the most accurate learning algorithm. The average classification accuracy and runtime latency for DT, SVM, and LR are 97.53% (1.9s), 86.26% (579.4s), and 95.84% (5.7s). Clearly, the DT algorithm achieves the best classification accuracy 97.53% with the least run time latency, which is caused by our careful feature selection. Compared with DT, the time latency of SVM is too high and the classification accuracy of LR is relatively low. So we choose DT learning algorithm in our system.

Besides, we conduct extensive evaluation of the performance on each class of URLs and the whole data set by recording the true positive rate (TPR) and false positive rate (FPR) when each algorithm is utilized. In Table 2, we can see that even if DT has the best performance in general, LR can obtain the 100% TPR and 0% FPR when dealing with malware Web sites. Thus, a hybrid learning algorithm combining DT and LR could further enhance the accuracy of URL classification, which belongs to our future work.

4. RELATED WORKS

Garera et al. use logistic regression over 18 hand-selected features to classify maclious URLs [1]. The features include the presence red flag key works in the URL, which are based on Google's page rank and web page quality guidelines. Zheng et a. propose a approach to classify phishing URLs by thresholding a weighted sum of 8 features [2], including 3 lexical features, 4 content-related features, and 1 WHOIS-related feature.

The authors use statistical methods in machine learning to classify phishing emails [3], where the classifier examines the properties of URLs contained in a message (number of domains, number of dots in URL). Bergholz et a. further improve the accuracy of the mechanism in [3] by introducing models of text classification to analyze email content [4].

5. CONCLUSION

We propose a learning based approach to separating Web sites into 3 classes: benign, phishing, and malware. The analysis is only based on URL itself without accessing the target website, which removes the run-time latency and protects user from being exposed to browser-based vulnerabilities. We argue that this approach is complementary to both blacklisting and the systems based on evaluating site content and behavior. By carefully selecting features and learning algorithms, our system achieves 97.53% accuracy on detecting malicious Web sites.

6. REFERENCES

[1] S. Garera, N. Provos, M. Chew, and A. D. Rubin. A Framework for Detection and Measurement of Phishing Attacks. In Proceedings of the ACM Workshop on Rapid Malcode (WORM), Alexandria, VA, Nov. 2007. http://web.cs.jhu.edu/ sdoshi/index_files/p1-garera.pdf

[2] Y. Zhang, J. Hong, and L. Cranor. CANTINA: A Content-Based Approach to Detecting Phishing Web Sites. In Proceedings of the International World Wide Web Conference (WWW), Banff, Alberta, Canada, May 2007. http://www.cs.cmu.edu/ jasonh/publications/www2007-cantina-final.pdf

[3] I. Fette, N. Sadeh, and A. Tomasic. Learning to Detect Phishing Emails. In Proceedings of the International World Wide Web Conference (WWW), Banff, Alberta, Canada, May 2007. http://www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA456046

[4] A. Bergholz, J. H. Chang, G. Paaß, F. Reichartz, and S. Strobel. Improved Phishing Detection using Model-Based Features. In Proceedings of the Conference on Email and Anti-Spam (CEAS), Mountain View, CA, Aug. 2008. http://www.ceas.cc/2008/papers/ceas2008-paper-44.pdf