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 95.3% accuracy on detecting malicious Web sites with only 2.5% false positive rate.

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 95.3% accuracy on detecting malicious Web sites with only 2.5% false positive rate.
- We implement and compare various classification algorithms, e.g. SVM, Logistic Regression, and Decision Tree.

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 "differ-

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

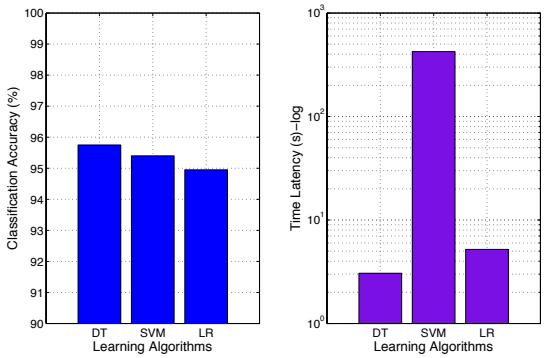

Figure 1: Classification Accuracy and Time Latency

ent", compared with legitimate sites. Analyzing lexical features enables us to capture the property for classification 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.

Host-based Features: Host-based features are based on the observation that many illegal sites choose less reputable hosting centers or disreputable registrars.

Site popularity Features: The reason for using these features is that malicious sites tend to be less popular than benign ones. One import feature among site population features is the number of links from outside.

2.2 Date Set

We randomly collect 29,276 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 7,071 samples from PhishTank[2], a collaborative site where people can submit and verify phishing URLs. Besides, we select 9,285 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 and select the best set of parameters. And then select 10,000 new URLs, including benign, phishing and malware ones, to test our project's performance.

Our project's goal is to learn a URL classification model to predict unknown URLs. Specifically, the task of the project includes three sub-tasks.

Feature Selection: Some extracted features are not informative and including them may cause negative results. So in our project, we need to adopt Chi-Square test to calculate and get the most informative features. Selected features will be organized in vector format and each element represent a feature.

Train and Classify: The last task is to train the classifier and use the trained classifier to predict unknown URLs. All the URLs, before training and classifying, need to be preprocessed into vector format. (i.e. <2,3,0,1,...> the i-th element is the value of i-th feature). The output of the classifier will be "benign", "malicious" (indicating that the URL is a phishing or malware site) or "unknown" (meaning that the classifier can't classify it with high confidence). We plan to choose SVM as learning method for its advantages in binary classification of high dimensional data.

3. EVALUATION

3.1 Feature Comparison

In order to find how much each feature improves/reduces the performance of our mechanism on classification accuracy, we separate the features into several groups and achieve the corresponding classification by using SVM, Logistic Regression, and Decision Tree libraries in Weka. From Figure 1, we can see that

3.2 Classifier 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				LG			
	Benign	Malware	Phishing	All	Benign	Malware	Phishing	All	Benign	Malware	Phishing	All
TPR	97.1%	97.2%	88.3%	95.8%	96.9%	96.3%	88.1%		96.7%	100%	80.8%	
FPR	6.2%	1.1%	1.3%	4.3%	6.4%	0.5%	2.2%		8%	0%	2.4%	

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 protect 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 95.3% accuracy on detecting malicious Web sites with only 2.5% false positive rate.

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