ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 4, April 2017

Detecting Malicious Uniform Resource Locators using Logistic Regression- An Implementation

Sashaank Pejathaya Murali

B. Tech Student, Department of Information Technology, SSN College of Engineering, Chennai, India

ABSTRACT: The World Wide Web has become the most essential criterion for information communication and knowledge dissemination. It helps to transact information timely, rapidly and easily. Identifying theft and identity fraud are referred as two sides of cyber crime in which hackers and malicious users obtain the personal data of existing legitimate users to attempt fraud or deception motivation for financial gain. E-mails are used as phishing tools in which legitimate looking emails are sent making the genuine users identity with legitimate content with malicious URLs. It helps to steal consumers' personal data such as user names, account numbers, passwords and other financial account credentials. Spoofed mails are mails in which a hacker pretends to be a legitimate sender posing to be from a legitimate organization and lets the user divulge his personal credentials. Malicious URL, or malicious website, is a common and serious threat to cyber security. Malicious URLs host unsolicited content (spam, phishing, drive-by exploits, etc.) and lure unsuspecting users to become victims of scams and cause losses of billions of dollars every year. It is imperative to detect and act on such threats in a timely manner. To improve the generality of malicious URL detectors, machine learning techniques have been explored with increasing attention in recent years. In this paper, I propose a simple algorithm to distinguish malicious URLs from non-malicious ones.

KEYWORDS: Malicious URL Detection, Machine Learning, Internet security, Cyber security, Logistic Regression

I. INTRODUCTION

The internet serves as a medium for a large number of malicious activities such as spam attacks, phishing attacks, DoS attacks and etc. motivated under financial aspects. These attacks attract the common users to click links attached in legitimate looking or spam emails and make them to visit malicious sites. They initiate them to click and urge them to give their personal information. E-mails with malicious URLs may have legitimate content in the body of the mails that are unable to be detected by content based spam filters.

In this section, I present the key principles used by researchers to solve the problem of malicious URL detection. A variety of approaches have been attempted to tackle the problem of malicious URL detection. These approaches can be broadly grouped into two categories: (i) Blacklisting or Heuristics and (ii) Machine Learning.

1) Blacklisting or Heuristic Approaches: Blacklisting is a common and classical technique for detecting malicious URLs, which often maintains a list of URLs that are known to be malicious. Whenever a new URL is visited, a database lookup is performed. If the URL is present in the blacklist, it is considered to be malicious and then a warning will be generated; else it is assumed to be benign. Blacklisting suffers from the inability to maintain an exhaustive list of all possible malicious URLs, as new URLs can be easily generated daily, thus making it impossible for them to detect new threats. Despite several problems faced by blacklisting due to their simplicity and efficiency, they continue to be one of the most commonly used techniques by many anti-virus systems today. Common attacks are identified, and based on their behaviour, a signature is assigned to this attack type. However, such methods can be designed for only a limited number of common threats. A more specific version of heuristic approaches is through analysis of execution dynamics of the webpage .Here too, the idea is to look for a signature of malicious activity such as unusual process

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 4, April 2017

creation, repeated redirection, etc. These methods necessarily require visiting the webpage and thus the URLs actually can make an attack.

2) Machine Learning: These approaches try to analyse the information of a URL and its corresponding websites or web pages, by extracting good feature representations of URLs, and training a prediction model on training data of both malicious and benign URLs. There are two-types of features that can be used - static features, and dynamic features. In static analysis, we perform the analysis of a webpage based on information available without executing the URL .The features extracted include lexical features from the URL string, information about the host, and sometimes even HTML and JavaScript content. Since no execution is required, these methods are safer than the dynamic approaches. The underlying assumption is that the distribution of these features is different for malicious and benign URLs. Using this distribution information, a prediction model can be built, which can make predictions on new URLs. Dynamic analysis techniques include monitoring the behaviour of the systems which are potential victims, to look for any anomaly. These include which monitor the system call sequences for abnormal behaviour, and which mine internet access log data for suspicious activity. Dynamic analysis techniques have inherent risks, and are difficult to implement and generalize. In his paper, I shall focus on static techniques and mainly the simplest, logistic regression.

II. RELATED WORK

Analysis of URLs

Phishing URLs can be analysed based on the lexical features and host based features of the URL. The lexical feature analyses the format of the URL. URLs contain the host name and the path. The proposed methodology analyses host based features such as page rank and age of domain, various lexical features such as URL encoding, presence of suspicious characters, hexadecimal character or malicious IP addresses to hide them. It is useful as illegitimate users spoof their identities, pass authentication tests and during content analysis also they may get escape by avoiding spam keywords. Some emails may not contain any message in the body except some malicious links in it urging the users to click them leading to fraudulent websites.

1. ***Lexical Features:***

   Lexical features analyses the format of the URL. It includes the length of the host name, length of the URL, the number of dots, presence of suspicious characters such @ symbol, hexadecimal characters and other special binary characters such as (‘.’, ‘=’, ‘$’, ‘^’ and etc.) either in the host or path name. IP addresses and hexadecimal characters are used to hide the actual URLs. The URL can also be represented using hexadecimal base values with a ‘%’ symbol.

2. ***Host Based Features:***

   Host based features identify the location, owner and how malicious sites are hosted and managed. Some of the features are as follows

i) Age of domain:

Age of the domain is used to identify when malicious websites are hosted such that they have less age or relatively new to obtain the user credentials. They will be recently registered sending more mails and some domains may not be available even at the time of checking. It obtains the data in the number of months and some may be in years more recently. The WHOIS lookups on the WHOIS server is used to retrieve the domain registration date, and if the domain registration entry is not found on the WHOIS server, this feature will simply return true, deeming it suspicious.

***ii)* Page Rank:**

Page rank provides the rank for the webpage and higher the page rank, the more important the page is. Obviously phishing web pages have less age of domain and short lived. Hence they obtain a very low page rank or page rank does not exist. Page rank is a link analysis algorithm in which each document on the web is assigned a numerical weight from 0 to 10, with 0 indicating least popular and 10 indicating most popular. A score value of 1 is assigned when the page rank value for a particular webpage is not available.

The classifier has a training dataset of malicious phishing URLs and legitimate URLs. The probability occurrence of each feature in the dataset is calculated and their respective scores are obtained (The occurrence of features in the

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 4, April 2017

dataset are counted and the cumulative score is calculated. If Cumulative score > Threshold, it is considered a phishing URL, else a legitimate URL)

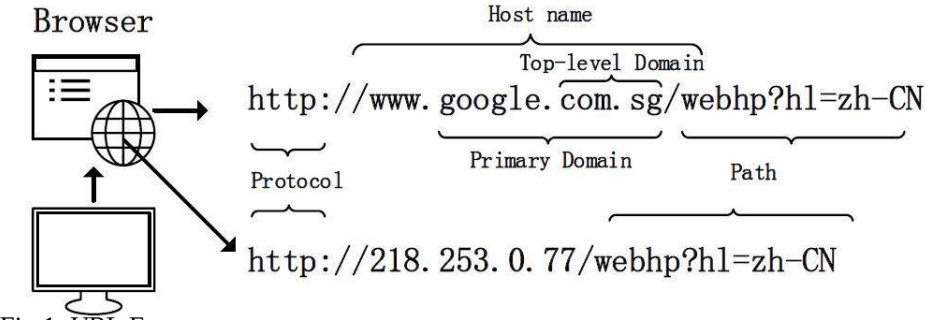

Fig 1: URL Features

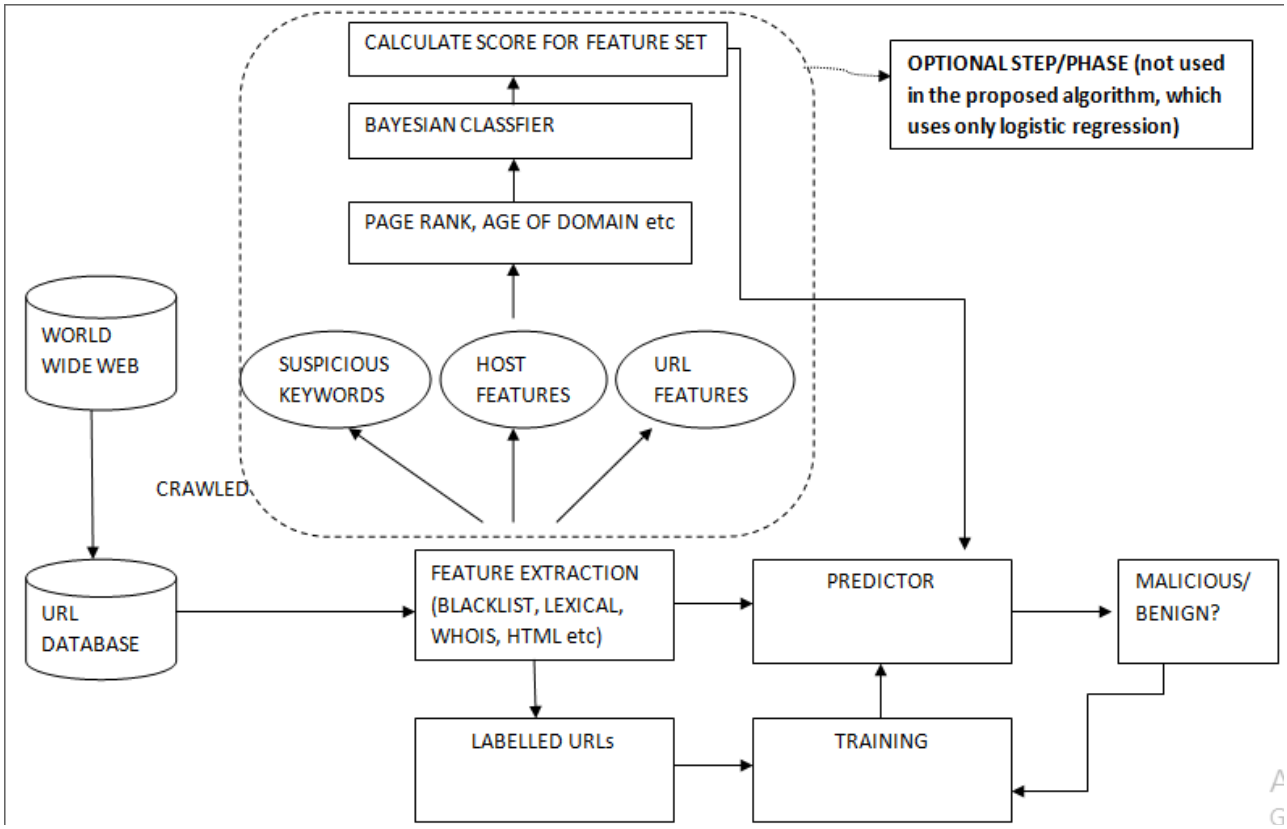

Fig 2: Malicious URL Detection

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 4, April 2017

II. PROPOSED METHODOLOGY

1. **Data Collection:**

i) **First Task:** The first task was gathering data. I found some websites offering malicious links while browsing. I set up a web crawler and crawled a lot of malicious links from various websites. (E.g.: vxvault.net)

ii) **Second Task:** The next task was finding out clear URLs. I used a data set that was already available, this time, and there wasn't a need for crawling.
I gathered around 500,000 URLs out of which around 90,000 were malicious and others were legitimate/clean.

2. **Algorithm:**

I used **logistic regression** because it is less time-consuming.

i) The first task was tokenizing the URLs. I wrote a tokenizer function in python for this. Some of the tokens we get are like *'virus','exe','wp','dat'* and so on.

ii) The next task was to load the data and store it in a list.

iii) Next, I vectorized the URLs. I used **tf-idf** scores instead of using bag of words classification since there are words in URLs that are more important than other expected words. I had the URLs converted into vectors.

iv) Finally, I converted it into test and training data and performed logistic regression on it.

III. EXPERIMENTAL RESULTS

In this algorithm, that I proposed I got an **accuracy of 98%.**
The URLs that I inputted were:

- oregonpreschool.org/wp-content/themes/spacious/js/schet_0612.exe
- google.com
- repsolt.pl/file/get.vbn
- wikipedia.com
- www.robsheehy.com/public_ftp/helpmerob/malware/Photo.scr

And my outputs were
['bad' 'good' 'bad' 'good' 'bad']

Tabulation of my results:

Input	Output
oregonpreschool.org/wp-content/themes/spacious/js/schet_0612.exe	**Bad**
google.com	**Good**
repsolt.pl/file/get.vbn	**Bad**
wikipedia.com	**Good**
www.robsheehy.com/public_ftp/helpmerob/malware/Photo.scr	**Bad**

And outputs a score of **0.98465 (98% accuracy)**

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 4, April 2017

```
C:\Users\sashaank\AppData\Roaming\Python\Python27\site-packages\sklearn\cross_validation.py:44: DeprecationWarning: This module was deprecated in version 0.18 in
favor of the model_selection module into which all the refactored classes and functions are moved. Also note that the interface of the new CV iterators are different
from that of this module. This module will be removed in 0.20.
  "This module will be removed in 0.20.", DeprecationWarning)
0.98465984089
 * Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
0.984207960235
['bad' 'good' 'bad' 'good' 'bad']
```

IV. CONCLUSION

Hackers bypass anti-spam filtering techniques by embedding malicious URL in the content of the messages. Hence the URL analyzer method with the help of minimized phishing feature set identifies the malicious URL in the emails. Malicious URL detection plays a critical role for many cyber security applications, and clearly machine learning approaches are a promising direction. In this article, I gave a comprehensive introduction on Malicious URL Detection using machine learning techniques. In particular, I proposed a simple algorithm using logistic regression for Malicious URL detection. Despite the extensive studies and the tremendous progress achieved in the past few years, automated detection of malicious URLs using machine learning remains a very challenging open problem. Future directions include more effective feature extraction and representation learning with more effective machine learning algorithms for training the predictive models.

REFERENCES

1.Colin Whittaker, Brian Ryner and MarriaNazif, “Large-Scale Automatic Classification of Phishing Pages”, In proceedings of NDSS, 2010.
2.Garera, S., Provos, N., Rubin, A.D. and Chew, M. “A Framework for Detection and Measurement of Phishing Attacks” In Proceedings of the 2007 ACM workshop on Recurring malcode, pp. 1-8, 2007.
3. Justin Ma, Lawrence K. Saul, Stefan Savage and Geoffrey M. Voelker, “Beyond Blacklists: Learning to Detect Malicious Web Sites from Suspicious URLs”, Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining pp.1245-1254, 2009.
4. Justin Ma, Lawrence Saul, K., Stefan Savage and Geoffrey Voelker, M. “Identifying Suspicious URLs: An Application of Large-Scale Online Learning”, In ICML ’09: Proceedings of the 26th Annual International Conference on Machine Learning, pp. 681-688, 2009.
5. Maher Aburrous, Hossain, M.A., KeshavDahal and FadiThabtah, “Experimental Case Studies for Investigating E-Banking Phishing Techniques and Attack Strategies”, Cognitive Computing,DOI 10.1007/s12559-010-9042-7, Vol. 2, pp. 242-253, 2010.
6.Doyen Sahoo, Chenghao Liu, and Steven C.H. Hoi: Malicious URL Detection using Machine Learning: A Survey.
7.PawanPrakash, Manish Kumar, RamanaRaoKompella and Minaxi Gupta, ‘PhishNet:Predictive Blacklisting to Detect Phishing Attacks’, Proceedings of the IEEE Infocom, pp.1-5, 2010.
8.Zhang, Y., Hong, J. and Cranor, L. Cantina: A Content-Based Approach to Detecting Phishing Web Sites. In Proceedings of the 16th international conference on World Wide Web, pp.639-648, 2007.

BIOGRAPHY

Sashaank Pejathaya Murali is a third year undergraduate student pursuing B.Tech in Information Technology in SSN College of Engineering, Chennai, India. His research interests are Machine Learning and Data Mining for Cyber and Information Security and Cyber Crime Analysis.