International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-5 Issue-5, October 2015

Survey on Detection of Malicious Web Pages and URLs Using Machine Learning

Samkeet Shah, Dakshil Shah, Lakshmi Kurup

Abstract- Web based security threat is rising every day. Web pages serve as one of the primary ways for interaction with and for the users. However, certain web application or websites are directed to mislead the user and try to gain access to the user's system in order to steal sensitive personal information. The old legacy based approaches on malicious web pages or URLs detection consist of using blacklist that check the URL against an existing database of flagged and suspicious links. The World Wide Web has progressed significantly, with the active use of JavaScript, ActiveX, Flash Player and related technologies. The heavy use of these technologies has improved the user experience and available services on web pages. Attackers tend to find security loopholes into these technologies and use them to their advantage. This method however fails to detect ever evolving attack methods. Thus there is a need to use methods that can adopt to and evolve simultaneously with the advancing threats. Hence, in this paper we have reviewed various types of web based attacks and machine learning techniques to detect malicious web pages and URLs.

Keywords- Machine Learning, Malicious Webpages, Web Security

I. INTRODUCTION

The use of malicious web content as an instrument to perform attacks on the Internet is on the rise. The increase in the number of Internet users and number of devices used to browse web pages has made attacks on web clients more lucrative. Due to this, mechanisms to counter these attacks by early detection are needed.

II. MACHINE LEARNING ALGORITHMS

A learning approach where there is no external critic's feedback and only input vectors can be used for learning is referred to as unsupervised learning. The system evolves to extract features in the input patterns. On the other hand, if the target value is known beforehand, then the system evolves by adjusting the weights to reach the final results. Such a system of learning is called Supervised Learning.

A. K means clustering

K Means clustering is an unsupervised learning algorithm which classifies data using k clusters with a centroid defined for each cluster.

Manuscript published on 30 October 2015.
*Correspondence Author(s)
Dakshil Shah, Department of Computer Engineering, Dwarkadas J. Sanghvi College of Engineering, Mumbai (Maharashtra). India.
Samkeet Shah, Department of Computer Engineering, Dwarkadas J. Sanghvi College of Engineering, Mumbai (Maharashtra). India.
Prof. Lakshmi Kurup, Department of Computer Engineering, Dwarkadas J. Sanghvi College of Engineering, Mumbai (Maharashtra). India.

Each point in the dataset is associated with the nearest centroid. The positions of the centroids are now recalculated until the centroids no longer move. The result is a separation of the objects into groups.

B. Naive Bayesian Classification

The Naive Bayes Classifier technique is based on the Bayes theorem in probability. It is suited when the dimensionality of the inputs is high. Consider a training set T, containing samples, each with their class labels. Let there be k classes- C1, C2...Ck. Each sample can be represented as an n-dimensional vector X={x1,x2..xn}, which depicts n measured values of n attributes. Given the sample X, the Naïve Bayes Classifier will predict the class which the sample belongs to. X is predicted to belong to class Ci if:
$P(Ci|X) > P(Cj|X)$ where $1 \leq j \leq m$, j not equal to i.
By Bayes theorem, we have:
$P(Ci|X) = (P(X|Ci)\ P(Ci))/(P(X))$, which is to be maximized.

C. K nearest neighbor

K nearest neighbor algorithm is a supervised learning algorithm based on classification of objects on the basis of closest training examples in the feature space. An object is classified by a majority of its neighbors such that K is always a positive integer. The neighbors are selected from a set of objects for which the correct classification is known, such as the training set. The parameter K is pre decided and the distance between the query instance and training samples is calculated using a suitable distance measure algorithm. These distances are sorted and the nearest neighbor is determined based on the Kth minimum distance [1].

D. Support Vector Machines (SVM)

The data is separated into 2 categories of performing classification and constructing an N-dimensional hyper plane in SVM. If the classes are separable by hyperplanes, an optimal function can be determined. The model uses a sigmoid function and is related to multilayer perceptron network where the hyperplane is expressed by its normal vector w and bias b as f(x)= sgn((w,x),b)[1]. The optimal hyperplane is found such that it separates clusters of vectors so that one category of the target variable is on one side and the other category is on the other side.[9]

E. Decision Tree

A decision tree is a flowchart like tree structure with internal nodes denoting a test on an attribute, branches representing an outcome of the test and each terminal node holding a class label.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

If we have a tuple X, whose class label is unknown, the attribute values are tested using the decision tree by tracing a path from the root node to a terminal node which has a class prediction for that tuple. [10]

F. Random Forest

It is an ensemble learning method for classification and regression that builds many decision trees at training time and combines their output for the final prediction. Random forest is a collection of CARTs (Classification and Regression Trees). It improves the overall accuracy of decision tree algorithm by encouraging diversity among the tree.

III. TYPES OF WEB ATTACKS

A. Cross Site Scripting (XSS)

Cross site scripting is an attack that is intended to be run on the client side by injecting and executing malicious script using the web application input or vulnerabilities in the web based application. However, the attacker cannot directly affect the victim. The attacker must first be able to inject a malicious script that is accepted into an invalidated input field on the website. When the user visits the web page and enters the input the malicious script will be executed. Effects of cross site scripting range from basic color and change in dimensions of the window to modification of database contents. JavaScript can be used to set and delete cookies. Thus malicious JavaScript code when executed can compromise a session of the user, in turn resulting in session hijacking.

B. Drive-by-Downloads Attack

This type of attack exploits the vulnerabilities in the user's browser or plugins/extensions installed in the browser or it can by embedded into a file that the user downloads from that webpage. Upon execution the file containing malicious code is activated or malicious JavaScript code executes as soon as user visits the page. It then downloads malicious code to the user's computer by exploiting the vulnerabilities found as stated above and gains access to the system. Such an infected system later becomes a part of a botnet.

C. JavaScript Obfuscation

Obfuscation is a mechanism of hiding the meaning or intent of JavaScript code. It may be used to prevent others from copying code or as a means to hide malicious code. The various methods used are:

- Renaming variables to meaningless names
- Removing whitespaces and line breaks
- Self-generating code which on the first pass generates the actual code
- Using character codes and string manipulation

Example :HTML/Framer

```
(function () {
   var qk = document.createElement('iframe');
   qk.src = 'http://mySite.com/wp-includes/file.php';
   qk.style.position = 'absolute';
   qk.style.border = '0';
   qk.style.height = '1px';
   qk.style.width = '1px';
   qk.style.left = '1px';
   qk.style.top = '1px';
   if (!document.getElementById('qk')) {
      document.write('<div id=\'qk\'></div>');
      document.getElementById('qk').appendChild(qk);
   }
})();
```

Obfuscated code:

```
eval(function(p,a,c,k,e,d){e=function(c){return
c.toString(36)};if(!''.replace(/^/,String)){while(c--
){d[c.toString(a)]=k[c]||c.toString(a)}k=[function(e){return
d[e]}];e=function(){return'\\w+'};c=1};while(c--
){if(k[c]){p=p.replace(new
RegExp('\\b'+e(c)+'\\b','g'),k[c])}}return p}('(i(){g
1=3.h(\'d\');1.f=\'c://b.7/8-
9/a.e\';1.2.t=\'p\';1.2.q=\'0\';1.2.s=\'4\';1.2.o=\'4\';1.2.n=\'4\';1.
2.j=\'4\';k(!3.6(\'1\')){3.l(\'<5
m=\\\'1\\\'></5>\');3.6(\'1\').r(1)}})();',30,30,'|qk|style|docum
ent|1px|div|getElementById|com|wp|includes|file|mySite|http
|iframe|php|src|var|createElement|function|top|if|write|id|left|
width|absolute|border|appendChild|height|position'.split('|'),0
,{}))
```

As the code is obfuscated, a simple inspection of the code does not reveal the malicious intent of the code which is hidden.

D. Click jacking

Click jacking is a web based attack. A malicious web page tricks users into clicking on an element on another page while the intention of the user is to click on the top level page. The element may be invisible or barely visible. An unintended action is performed such as redirection, downloading malware.

E. Phishing

Phishing is a form of a social engineering attack. In the context of websites, a phishing attack may involve cloning of a legitimate website and requesting users to submit sensitive information or can be used to deploy a payload. Social Engineering Toolkits such as the one provided by Backtrack 5, allows an attacker to clone a website and receive data entered by a user on the cloned site. Detection is possible by analyzing the URL and contents of the webpage.

IV. RELEVANT WORK

Ma et al. [5] has categorized the features that can be gathered from URLs as either lexical or host based. Malicious URLs appear different from ordinary URLs. Based on this, patterns may be inferred. The length of the hostname, length of the URL and tokens in the hostname. Host based features are used as web sites may be hosted on servers present in locations with a history of malicious activities, owner reputation and management of the website. Using the IP address, a blacklist of IP addresses can be checked and hosting location can be obtained.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Using the WHOIS properties, details of the domain name such as registrar and owner may be obtained.The hierarchical nature of subdomains can be used to detect a phishing or suspicious pattern [7]. The URL can be split along each separator or ".". A malicious webpage may try to seem more legitimate by having a URL similar to the actual url. If mybank.com is a legitimate URL and mybank.com.sample.com is the misleading URL, it is possible to detect the malicious URL as the legitimate URL is actually a subdomain of sample.com and is meant to confuse a user.

Sehun et al. [3] have used a hybrid approach of misuses detection and anomaly detection to detect malicious webpages. The process works in two phases; the first phase involves detection of known web pages using misuse detection. The misuse detection model is built on a decision tree which is generated using the C4.5 algorithm. The C4.5 algorithm chooses that attribute that most effectively splits the data, such that the resultant classes are enriched with one class or another. Following this, the unknown webpages detection is implemented using anomaly detection technique, which uses principle of non-conformity to classify objects. Hence anomaly detection trains the system with known patterns of malicious web pages, and then looks for abnormalities and deviations in the new webpages. The results achieved using this two phase approached is shown have a higher detection and classification rate compared to the individual methods. Their two-phase system used 171 features extracted off webpages and achieved an overall accuracy of 98.9%.

Marco Cova et al. [4] have implemented JSAND (JavaScript Anomaly-based Analysis and Detection) along with Naïve Bayes classifier to detect and classify malicious content and webpages. Their model works in two phases, training mode and detection mode. Training mode involves training the network to set a threshold values for each of the following features listed below. Their experimental results included testing their tool on 140,000 webpages. The features defined are Number and target of redirections, Browser personality and history-based differences, Ratio of string definitions and string uses, Number of dynamic code executions, Length of dynamically evaluated code Number of bytes allocated through string operations, Number of likely shellcode strings, Number of instantiated components, Values of attributes and parameters in method calls and Sequences of method calls. Changes in their values is assessed with respect to the values that have been trained initially in the network. JSAND is used to detect anomalies in the values of exploit features and uses Naïve Bayes classifier to classify the exploit to the particular exploit class.

Da Huang et al. [6] have used a lexical analysis combined with their own version of mining algorithm, Greedy Selection Algorithm (GA) detect malicious URLs. URLs are considered to be a sequence of URL segments. One disadvantage of the work by Ma et Al is in real time applications, as they don't explicitly extract human interpretable URL patterns. Each segment represents the different parts of the main URL, such as a domain name or a file name. Since it is difficult to extract meaningful data of a sequence directly, they've initially broken down sequence into segments.In the proposed GA algorithm, the authors join two URL patterns p1 and p2 only if the quality of the combined pattern is better than each individual pattern. In order to reduce the redundancy associated with generation of multiple URLs, they generate URL only when a valid URL pattern is generated. To finally assess the overall quality of the URL they've used two parameters, malicious frequency and white frequency, where malicious frequency is the number of unique domains covered by the generated URL pattern and white frequency is the number of unique URL patterns it covers in the data set.

Ram B. Basnet et al [7] have proposed multiple feature based malicious web page and URL detection techniques using machine learning algorithms such as linear SVM, Random Forest, J48 (Decision Tree) and Naive Bayes. they have used the following datasets, PhishTank for data on phishing URLs, DMOZ Open Directory Project [12] database and Yahoo's public directory as a source of legitimate webpages. Once the web pages are crawled, their features are extracted and classified primarily into two types, URL based features and Content based features. Next, in order to find the importance of each individual feature, we compute the F-score (Fisher score) of all the features. F-score is a simple but effective criterion to measure the discrimination between a feature and the label. A larger F-score value indicates that the feature is more discriminative. [7] The entire process first requires the classifier to be trained and then each individual algorithm is implemented and it is found that Random Forest has the lowest error and false detection rate in batch algorithms. However, online algorithms offer the distinctive advantage of automatic updating of the learning model rather than having restart from scratch to train it with new values. The overall accuracy of the feature based model is 99.9% accuracy in phishing detection with an error or false negative detection rate of 0.06%

Wang et al. [8] have proposed a malicious JavaScript analysis framework based on SVM. The dataset is prepared by collecting benign and malicious JavaScript from the site, followed by cleaning the data and feature extraction. The data is then normalized and scaled to [1,0]. Using WEKA, the model is trained using SVM. Their findings show that SVM achieves and overall accuracy of 94.38% on the training set when compared to Naïve Bayes and ADTree

Hyusang Choi et al [11] et al. have used a dataset containing 40,000 harmless URLs and 32,000 malicious ones. Their proposed model uses a combination of machine learning algorithms with discriminative features obtained using lexical analysis. In addition to that their model is robust against evasion methods like redirection, fast-flux hosting and link manipulation. The model works as follows; initially they use SVM classifier on the extracted features to detect malicious URLs like a binary classifier. Following this, they use RAkEL, a high performance multi-label learning method that accepts any multi-label learner as a parameter. RAkEL creates m random sets of k label combinations, and builds an ensemble of Label Powerset classifiers from each of the random sets [11].

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

C4.5 algorithm is used as the single label classifier and LP as a parameter of the multi-label learner. A variation of kNN algorithm ML-kNN is used to determine the label set for unknown instances. The overall malicious webpages detection accuracy is 98% with their model.

Table I. Comparison of Relevant works

Relevant Work	Method used	Attack Detected	Advantages	Disadvantages
Ma et al.[5]	Logistic Regression, Perceptron and Lexical URL features	Malicious URL	feasible to automatically sift through comprehensive feature sets	Cannot be used in real time application as they do not explicitly extract human identifiable features
Sehun et al.[3]	Anomaly detection techniques, C4.5 algorithm	Malicious webpages	solves the disadvantages of misuse detection and an anomaly detection methods	proposed method had a relatively high false positive rate
Marco Cova et al.[4]	JSAND, feature extraction	Drive-by-download	reliable detection of malicious code by emulation of exercise behavior and analysis can be parallelized	detection of binary shellcode weak. Significant false positives and negatives detected
Da Huang et al[6]	Greedy selection algorithm, segmentation	URL of malicious websites	More general lexical feature extraction, hence availability of better flexible tokens for efficient classification	high memory consumption by GA due to excessive number of lexical strings to be stored during the runtime
Ram B. Basnet et al[1]	Random Forest, SVM, F-score	Phishing URL	Extremely high accuracy of 99%	classification performance of classifier degrades when new data sets are used for testing, while it is trained on the older data set
Wang et al.[8]	SVM	Malicious JavaScript code	SVM is shown to have better accuracy in Binary Classification	SVM classification accuracy is affected if value of nuclear radius isn't chosen properly
Hyusang Choi et al[11]	Link Popularity, ML-kNN, RAkEL, SVM	Phishing, Spam	ability to detect wide range of attacks	evading all the features in the method would cost much more

V. CONCLUSION

In this paper, we have primarily reviewed the use of machine learning algorithms in detection of malicious web pages and URLs. We have also given an overview of different machine learning algorithms and the different types of web attacks. Based upon the above researches, we can state that feature extraction using lexical analysis is one of the most effective and commonly used techniques in the process of detection of malicious web pages and URLs. JavaScript Obfuscation was originally intended to protect source code, however due to its property of hiding information, it has been misused to carry out attacks. JavaScript Obfuscation attack is the most prevalent type of attack due to the extensive application of JavaScript on web pages, which in turn makes it easy for attackers to use JavaScript as the most common means of attack. Given the extensive use of the Internet and ever evolving attacks, more robust methods for prevention and detection of web based attacks is needed. Thus we have summarized the relevant work with findings from various authors and their corresponding researches.

REFERENCES

1. Ramana, Bendi Venkata, M. Surendra Prasad Babu, and N. B. Venkateswarlu. "A critical comparative study of liver patients from usa and india: An exploratory analysis." International Journal of Computer Science Issues 9.2 (2012): 506-516.
2. Chong, Christophe, Daniel Liu, and Wonhong Lee. "Malicious URL Detection."
3. Sehun Yoo, Sehun Kim “Two-Phase Malicious Web Page Detection Scheme Using Misuse and Anomaly Detection” International Journal of Reliable Information and Assurance Vol.2, No.1, 2014

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

4. M. Cova, C. Kuregel, and G. Vigna. Detection and analysis of drive-by-download attacks and malicious JavaScript code. In Proc. of the International World Wide Web Conference (WWW'10), Releigh, North Carolina, USA, pages 281–290. ACM, 2010.
5. Ma, Justin, et al. "Identifying suspicious URLs: an application of large-scale online learning." Proceedings of the 26th Annual International Conference on Machine Learning. ACM, 2009.
6. Da Huang · Kai Xu · Jian Pei "Malicious URL detection by dynamically mining patterns without pre-defined elements". Springer, WorldWideWeb (2014)
7. R. B. Basnet and A. H. Sung, "Learning to Detect Phishing Webpages", Journal of Internet Services and Information Security (JISIS), vol. 4, no. 3, (2014), pp. 21-39.
8. WANG, Wei-Hong, et al. "A Static Malicious Javascript Detection Using SVM." Proceedings of the International Conference on Computer Science and Electronics Engineering. Vol. 40. 2013.
9. L. Rokach and O. Maimon, "Decision trees," in, O. Maimon andL. Rokach, Eds. Springer US, 2005, pp. 165-192.
10. J. Dukart, "Support Vector Machine Classification Basic Principles and Application," pp. 19-19, 2012.
11. Hyusang Choi, Bin B. Zhu, Heejo Lee "Detecting Malicious Web Links and Identifying Their Attack Types"
12. DMOZ. Netscape open directory project. http://www.dmoz.org.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)