Malicious URL Detection using Logistic Regression

Malicious URL Detection using Logistic Regression

Mrs. Latha A.P(Assistant Professor).,P B Sashank (8th Sem), R Rohit (8th Sem), Rohit K Y(8th Sem), S R Karthik (8th Sem), Department of Information Science and Engineering, Dayananda Sagar College of Engineering, Bangalore.India

Abstract— Many web applications suffer from various web attacks as a result of lack of security consciousness. Therefore, it is necessary to improve the reliability of web applications by accurately detecting malicious URLs. In previous studies, keyword matching has always been used to detect malicious URLs, but this method is not effective. blacklists cannot be effective, and lack the ability to detect newly generated malicious URLs. When a new malicious URL is used by the user then we have to update the database which is time consuming and difficult. In this paper we focus on building a system for URL analysis and classification to primarily detect malicious URLs and reduce the cyber-attacks. The malicious URLs detection is treated as a binary classification problem. The proposed approach is that classifies URLs automatically by using Machine-Learning algorithm called logistic regression that is used to binary classification. WE use feature extraction and tokenization in the preliminary stage result of this is given to machine learning algorithm, namely logistic regression. Logistic regression algorithm classifies the data to be malicious or genuine one. In addition, genuine ones are tagged as good and if it is malicious it is tagged as bad.

Keywords—Malicious URLs, Blacklisting, Machine learning, Logistic Regression

1. INTRODUCTION

Web is regularly utilized by crooks for criminal operations. As the new technologies are emerging in the internet there are more of negative impacts than the positive impacts. The increasing use of the internet for such purposes increases the scope for cybercrimes activities. As accordance the number of users grow, there is comparative increase in attackers. As a result, attackers are exploiting the credentials of the end users without their content. Malicious or malware websites become one of the major threats for cyber security. Malicious URLs host content abnormalities, such as spamming, phishing attacks, exploiting users, threats, etc. They to cause huge monetary loss of billions of dollars every year worldwide. It is very important to detect and act on such attacks frequently for security. Most of the malwares uses internet to perform the attacks. Most of the attackers performs the cyber-attacks on Web using malware URLs since URLs are widely used all over the internet. So, a significant approach is required to detect malicious URLs and identify their attacks in order protect the user's data.

Normally the attacker performs his own URL it has its own structure. He may send this URL using many platforms unknowingly we may click on that URL and fell into his trap. He may send a message in email or use the twitter etc. After clicking the URL, the website may be malicious and we may loss the data. During banking transactions, it is very important to use the safe URL. To avoid problems like this it is very important to detect malicious URLs and block them.

Normally the method used was Generally, Black-List is a collection of data of malicious URLs which are previously known. A database lookup is performed every time the system come across a new URL. Here, the new URL will be matched and tested with every previously known malicious URL in the black list. Black List database should be updated as it is not dynamic. This technique is repetitive, time-consuming and difficult because of newly arising the malicious URLs. But blacklisting cannot detect the suspicious URLs which are not in the list so this method is not effective.

Other approach is Heuristic classification it is an improvement to the Black-Listing. Here the signatures are matched and tested to find the relation between the new URL and signature of existing malicious URL.

To overcome the disadvantages of the blacklisting method machine Learning methods are used because machine learning algorithms can be trained to learn on their own. Machine learning algorithms are effective in identifying a new type of malicious URL used by the attackers. Today's simple implementation of detection techniques is insufficient to address various URLs encountered in everyday life. However, attacks have being still taking place, as there has not developed any method to avoid users being exploited to malicious URL's. Thus, through our work we detect the malicious URLs in real time using machine learning algorithm based on logistic regression. URLs are tagged as good if the URL is safe and bad if the URL is unsafe so that users can know about it.

1. LITERATURE REVIEW

A Social Approach to Security: Using Social Networks to Help Detect Malicious Web Content [1]

The aim of this paper is to detect the malicious content present in the social media especially Facebook. It uses different information such as Facebook Heuristics, traditional Heuristics, twitter Heuristics and google safe browsing. All these are tested

and a score is given at the last based on the score it decides whether the link is malicious or legitimate. $Good_score and $bad score are the variables used to give for safe and unsafe content respectively. Application is tested in two components automated scan and the manual scan. Automated scan is for users whereas manual scan is for the researches to do research outside the Facebook. The success rate of this application is high. Any Facebook user can secure his account and stay away from malicious links. But it is limited to Facebook only and the overall performance of the model can be improved.

Malicious Web Page Detection Based on ON-LINE Learning Algorithm [2]

This paper proposes to use online learning methods to detect malicious URLs. On-line learnings methods are efficient when large number of data samples are there. In this work the three types of algorithms are used and compared the results with those three algorithms. The algorithms used are Perceptron, Passive-Aggressive (PA) Algorithm, Confidence-Weighted (CW) Algorithm. Among these algorithms Confidence-Weighted (CW) Algorithm has more accuracy compared to others. The content and format of web pages change constantly if the new one is different from the training samples then we have to add new webpage to the training sample to avoid this we use on-line learning algorithm. It uses the URL rather than the content of the web page. The disadvantage of this system is sometimes URLs may look safe but it may contain malicious content. The accuracy of the model should be improved.

Real Time Detection System for Malicious URLs [3]

This paper proposes on detecting the malicious URLs in twitter in real time. The data is collected from the real time in twitter. The tweets which have URL are collected from the twitter and the URLs are extracted. The short URLs are converted into long URLs. Next the URLs which have same domain are collected by the thread master module. Among the URLs the frequently used URL is connected to the crawler browser the crawler browser checks whether it is having any relation to other URL from database or not and if it finds any correlation then it checks for the entry point, domain name etc and marks the URL as suspicious. The URLs are detected in the real time so the users can be careful while clicking on the URLs. The Real Time Detection System uses a POPUP window Concept to detect the malicious URLs. The small window gives accurate and good results. But it cannot identify suspicious URLs that repeat after specified time spam.

Phishing URL detection using URL Ranking [4]

The approach used in this paper is hybrid approach it is by combing both the clustering and the classification algorithms. It uses Online learning for the classification. Clustering is performed on the dataset using the K-means algorithm. Three types of features are extracted from the URLs they are host-based features, lexical features and Cluster Label Feature. All the URLs categorized into 3 types Severe, Moderate, and Benign. After extracting the features from the URL falls into one of the categories. An internal scale gives the categorization of each URL if the URL is severe then it is scaled to red. If it is moderate then scaled to yellow and if it is severe it is scaled to red. It has a higher accuracy of 98.45%. The moderate URLs sometimes may be malicious so user may tend to the attacks.

Detection of Malicious URLs In Big Data Using Ripper Algorithm [5]

In this paper the method used to detect malicious URLs is using the Ripper algorithm. Firstly, dataset is created by collecting the different malicious and legitimate URLs. Some features are extracted from both the URLs and the dataset is split into training and testing dataset. The datasets are in .arff file format which is supported by WEKA. The dataset is pre-processed and RIPPER algorithm is used to train the training dataset. Ripper algorithm uses FOIL's information gain to choose the attributes that can generate best rule. URLs from testing dataset are tested using rule-based classifier by the parameters like true negative, true positive, false negative, false positive. Confusion matrix is created to analyze the above things. This model is accurate for large dataset. Sometimes the URLs are wrongly classified.

MALICIOUS URL DETECTION USING MULTI-LAYER FILTERING Model [6]

The aim of this research is to detect the malicious URLs by using multiple methods. model consists of 4 Layer filter composition. A URL is filtered and classified through four different layers. Each URL goes through 4 layers of filter composition. First layer in the model is Stratified filter it consists of the whitelist and blacklist filter. The normal URLs are stored in the whitelist files and malicious URLs are stored in blacklist files. To detect the type of URL, we traverse the list of Black and white to determine whether the URL in the black List or in the white list. The second layer filter in this model is a naive Bayesian filter. It trains the model by training the URL samples and use two-dimensional arrays c1 and c2 to store the probability of each value of malicious website and benign website. The third layer in the model is CART decision tree filter. The CART tree is constructed by Training samples with URLs, and the leaf nodes are deciding Nodes. the CART tree in the file system. This model is good at data. The last filter in the model is SVM filter. It combines all the above filters to determine the results.

Malicious URL Detection Based on Kolmogorov Complexity Estimation [7]

In this work malicious URLs are detected by using n estimation of the conditional Kolmogorov complexity of URL strings. For an URL Kolmogorov complexity is measured. The dataset used is a private dataset from a commercial company which can collect more than one million unclassified URLs in a typical hour. Two databases are there keep all the URLs that user have seen in the history: one for the malicious part, denoted by Dm, and the other for the legitimate part, denoted by Db. If the user enters a new URL to find the type of URL the URL is compared with Dm and Db. It can be combined with other URLs for better results.

Detecting Malicious Websites by Integrating Malicious, Benign, and Compromised Redirection Subgraph Similarities [8]

This paper uses graph mining approach to detect the malicious URLs. All the websites are converted at the time of access into a redirection graph. The vertices of the graph are URLs and edges are redirections between the two URLs. Subgraphs are extracted from each graph, and the similarities of many pairs of graphs are calculated on the basis of the number of subgraphs shared by them. To perform classification, we combine the similarities between its subgraphs and redirection subgraphs shared across malicious, benign, and compromised websites. We calculate the numerical value by comparing the similarities and the values are stored in the feature vector form. To that feature vector we apply random forest

algorithm. his approach has high classification accuracy, but the drawback is it has a high computational cost.

The Detection Method for Two-dimensional Barcode Malicious URLs Based on the Hash Function [9]

This paper proposed a new method that extracts the Eigenvalues of URLs embedded in two-dimensional barcode by H-SBH. The detection system used in this paper is Single-Block hash function H-SBH to extract eigenvalues of URLs. H-SBH deals the input value with 128-bit blocks. URLs are represented as binary string and divided into L 128-bit mini-blocks, then each H-SBH deals 128-bit data from the current block and 128-bit data obtained by the before block through calculation. Finally, after a series of processing, algorithm outputs a 128-bit hashed value. Each URL is given a fixed-length eigenvalue which is unique. Then the whitelist and blacklist database is created. The eigenvalues of secure URLs is stored in the whitelist database and the eigenvalue of malicious URLs is stored in the blacklist database. Black and whitelist databases are updated according. The detection system checks with the eigen values in the white list and blacklist database and matches both the database to detect the malicious URLS.

Detecting malicious URLs. A semi-supervised machine learning system approach[10]

The aim of the work is to detect the malicious URLs in the network traffic. After accessing the URL, it will be tested by the trained model and later it will be shown as malicious or benign. If the URL is considered as malicious then it be blocked. If the URL is not malicious then the browser will be open. The detecting system consists of three components, they are training algorithm, the model and cache database. To classify a URL the detection system uses a model trained using the one side class perceptron algorithm. Bayes algorithm is used for detecting malicious URLs. The main advantage of this model is it adopts to the changing malicious URL structure.

Intelligent Malicious URL Detection with Feature Analysis [11]

The purpose of this paper is to develop a system to detect malicious URLs and provide 41 features, that includes three type features, one type is domain-based features, another is Alexa-based features, and the other is obfuscation technique-based features are used to train the model. Domain based features include Domain, org, creation time, updation time, expiration time, count dns. Alexa based features include Three-month website popularity ranking, monthly, website popularity ranking, weekly, website popularity ranking and so on. obfuscation technique-based features include Average number of comments per line in JavaScript, Percent rate of no comment program in JavaScript, Average number of string functions in JavaScript, Size of script in JavaScript. To train the model finding the features and Removing redundant noise of Domain based features, Obfuscation-based features, and Alexa-based features is important. ANOVA and XGBoost are the algorithms used to train the model. XGBoost is based on the Gradient Boosting Decision Tree (GBDT). It adjusts the data of misclassified weights. XGBoost generalize loss values from square loss to a second-order deductible loss. This method has higher accuracy and high training speed.

Deep Approaches on Malicious URL Classification [12]

The approach used in this paper is deep learning. Deep learning extracts the features on its own. First the data is pre-processed. Tokenization is used to extract the features and then tokens are mapped to their ids. preprocessed data is sent to the neural networks. Different neural networks such as RNN, CNN, LSTM are created. The data is sent and trained using all these networks and the accuracy is measured among all these networks. The advantage of this technique is no need to specify the features the neural network itself extracts the features and trains according.

A Bi-Directional LSTM Model with Attention for Malicious URL Detection [13]

The authors proposed an attentional based BiLSTM model AB-BiLSTM for the Malicious URLs detection in this paper. To eliminate the task of extracting the important features manually the deep learning-based model AB-BiLSTM for malicious URL detection is proposed. AB-BiLSTM model learns the semantic relationships between the URL sequences. The model contains five layers the input layer, embedding layer, BiLSTM layer, attention layer, and output layer. Input Layer consists of the URL as input. IN embedding layer URL is broken down into whitespace-separated words and symbols. It contains two layers of hidden nodes from two separate LSTMs. Attention mechanism is to capture relevant features from the output of BiLSTM. Output layer gives the output. The model learns the features automatically and uses the attention mechanism to capture the important and relevant features to achieve greater accuracy.

3.Applications of the Project

Data is important nowadays. All the user's data should be protected and private data should be secured this work is helpful in securing the data. This work can be implied to reduce the cyber-attacks. Detecting malicious URLs will be helpful in reducing the crime related to net banking and money transfer. To protect the users from the attacks through URLs sent in emails and twitter the URL is to be tested whether it is safe or not after confirming the URL is safe user can go through the link. This detection mechanisms can be used in the websites to automatically detect the new URLs and to block the malicious URLs.

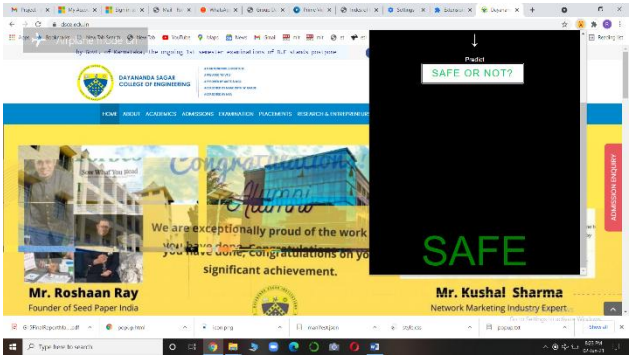
4.Tools Used in the Project

Many algorithms are used to detect the malicious URLs. But in our work, we are using Logistic regression algorithm. Through tokenization data is pre-processed. some important features are extracted and trained. The language used to implement the project is python. Jupyter notebook is the platform. The project is demonstrated through the webapp where we enter the URL and it gives output as malicious or not.

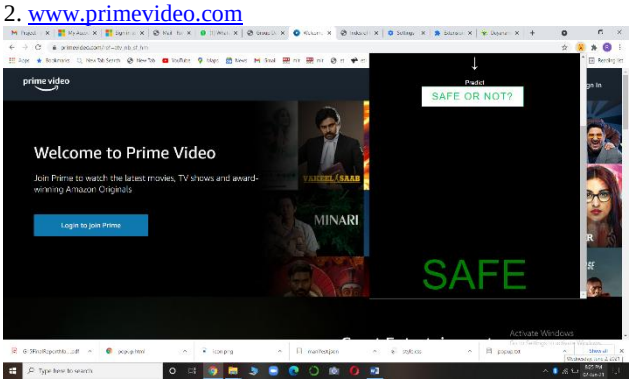
5. Results

The results of the model is either "safe" or "malicious". Few outputs of the developed model when tested with various URLs are as shown below

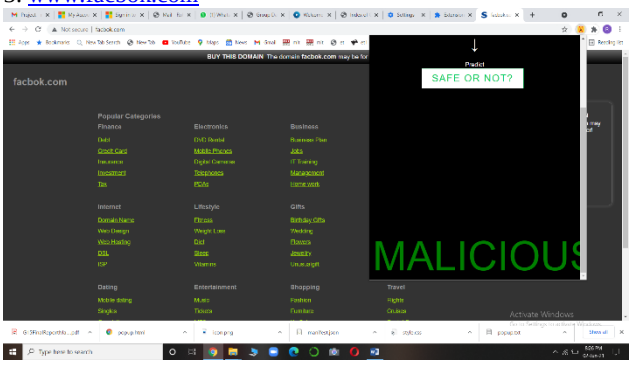
1. www.dsce.edu.in



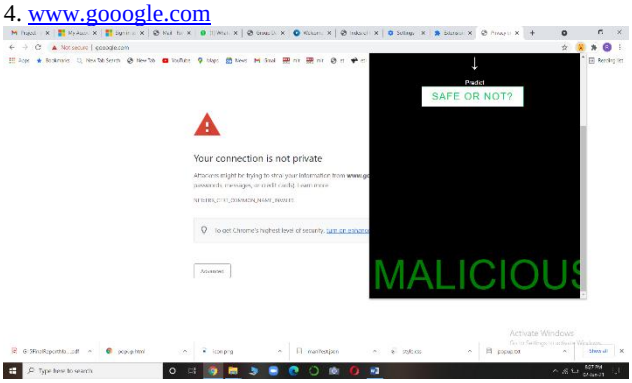
2. www.primevideo.com



3. www.facbok.com



4. www.goooogle.com



6.CONCLUSION

Malicious URLs and Web sites are the basis of most of the cybercrime activities over the internet. The problems that arise from the malicious sites are enormous and the end-users must be prohibited from visiting such sites. Thus, in this work we found out a technique through which URL Spamming can be avoided in real time through the help of machine learning algorithm-based o on logistic regression. We propose a system which contains a plugin-in as frontend which tags the URL as good and bad. This method not only detects the malicious but the user can know the type of URL to avoid cyber-attacks. Reducing the cyber-crime and protecting user data is the main objective of this work. We aim to reduce the number of cyber-attacks through URLs by this work. Future directions include more effective feature extraction and classification. Blocking all the unnecessary sites for children other than educational sites because the children may unknowingly go through the malicious links and may loose the data

.

7. ACKNOWLEDGEMENT

We acknowledge the efforts and hard work by the experts who have contributed towards development of the different home automation systems. We would also like to thank the Department of Information Science and Engineering, Dayananda Sagar college of Engineering for providing us an opportunity to bring our idea to an implementation level. We also like to thank our project guide Mrs. Latha A.P., Assistant Professor, ISE department for her support and guidance.

6.Refrences

1.M. Robertson, Yin Pan and Bo Yuan, "A social approach to security: Using social networks to help detect malicious web content," 2010 IEEE International Conference on Intelligent Systems and Knowledge Engineering, Hangzhou, 2010, pp. 436-441, doi: 10.1109/ISKE.2010.5680839.

2. W. Zhang, Y. Ding, Y. Tang and B. Zhao, "Malicious web page detection based on on-line learning algorithm," 2011 International Conference on Machine Learning and Cybernetics, Guilin, 2011, pp. 1914-1919, doi: 10.1109/ICMLC.2011.6016954.

3. N. S. Gawale and N. N. Patil, "Real Time Detection System for Malicious URLs," 2014 International Conference on Computational Intelligence and Communication Networks, Bhopal, 2014, pp. 856-860, doi: 10.1109/CICN.2014.181.

4. M. N. Feroz and S. Mengel, "Phishing URL Detection Using URL Ranking," 2015 IEEE International Congress on Big Data, New York, NY, 2015, pp. 635-638, doi: 10.1109/BigDataCongress.2015.97.

5. S. Thakur, E. Meenakshi and A. Priya, "Detection of malicious URLs in big data using RIPPER algorithm," 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), Bangalore, 2017, pp. 1296-1301, doi: 10.1109/RTEICT.2017.8256808.

6. R. Kumar, X. Zhang, H. A. Tariq and R. U. Khan, "Malicious URL detection using multi-layer filtering model," 2017 14th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), Chengdu, 2017, pp. 97-100, doi: 10.1109/ICCWAMTIP.2017.8301457.

7. H. Pao, Y. Chou and Y. Lee, "Malicious URL Detection Based on Kolmogorov Complexity Estimation," 2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology, Macau, 2012, pp. 380-387, doi: 10.1109/WI-IAT.2012.258.

8. T. Shibahara, Y. Takata, M. Akiyama, T. Yagi and T. Yada, "Detecting Malicious Websites by Integrating Malicious, Benign, and Compromised Redirection Subgraph Similarities," 2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC), Turin, 2017, pp. 655-664, doi: 10.1109/COMPSAC.2017.105.

9. J. Xuan and L. Yongzhen, "The Detection Method for Two-Dimensional Barcode Malicious URLs Based on the Hash Function," 2016 3rd International Conference on Information Science and Control Engineering (ICISCE), Beijing, 2016, pp. 702-705, doi: 10.1109/ICISCE.2016.155.

10. A. D. Gabriel, D. T. Gavrilut, B. I. Alexandru and P. A. Stefan, "Detecting Malicious URLs: A Semi-Supervised Machine Learning System Approach," 2016 18th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), Timisoara, 2016, pp. 233-239, doi: 10.1109/SYNASC.2016.045.

11. Y. -C. Chen, Y. -W. Ma and J. -L. Chen, "Intelligent Malicious URL Detection with Feature Analysis," 2020 IEEE Symposium on Computers and Communications (ISCC), Rennes, France, 2020, pp. 1-5, doi: 10.1109/ISCC50000.2020.9219637.

12. A. Das, A. Das, A. Datta, S. Si and S. Barman, "Deep Approaches on Malicious URL Classification," 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2020, pp. 1-6, doi: 10.1109/ICCCNT49239.2020.9225338.

13. F. Ren, Z. Jiang and J. Liu, "A Bi-Directional LSTM Model with Attention for Malicious URL Detection," 2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chengdu, China, 2019, pp. 300-305, doi: 10.1109/IAEAC47372.2019.8997947.

14. A. R. Nagaonkar and U. L. Kulkarni, "Finding the malicious URLs using search engines," 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, 2016, pp. 3692-3694.

15. W. Yang, W. Zuo and B. Cui, "Detecting Malicious URLs via a Keyword-Based Convolutional Gated-Recurrent-Unit Neural Network," in IEEE Access, vol. 7, pp. 29891-29900, 2019, doi: 10.1109/ACCESS.2019.2895751.

16. A. S. Popescu, D. B. Prelipcean and D. T. Gavrilut, "A Study on Techniques for Proactively Identifying Malicious URLs," 2015 17th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), Timisoara, 2015, pp. 204-211, doi: 10.1109/SYNASC.2015.40.

17. W. Chen, Y. Zeng and M. Qiu, "Using Adversarial Examples to Bypass Deep Learning Based URL Detection System," 2019 IEEE International Conference on Smart Cloud (SmartCloud), Tokyo, Japan, 2019, pp. 128-130, doi: 10.1109/SmartCloud.2019.00031.