A Machine Learning Approach for Detection and Analysis of Malicious URLs

Krishna Kumar Sahu[1], Ajay Barapatre[2]
[1]Mtech Scholar, [2]Assistant Professor, Department of ECE, RKDF University, Bhopal, M.P, India
[1]kapilsahu.krish@gmail.com, [2]barapatre.ajay@yahoo.co.in

Abstract--- The primitive usage of URL (Uniform Resource Locator) is to use as a Web Address. However, some URLs can also be used to host unsolicited content that can potentially result in cyber attacks. These URLs are called malicious URLs. The inability of the end user system to detect and remove the malicious URLs can put the legitimate user in vulnerable condition. Furthermore, usage of malicious URLs may lead to illegitimate access to the user data by adversary. The main motive for malicious URL detection is that they provide an attack surface to the adversary. It is vital to counter these activities via some new methodology. In literature, there have been many filtering mechanisms to detect the malicious URLs. Some of them are Black-Listing, Heuristic Classification etc. These traditional mechanisms rely on keyword matching and URL syntax matching. Therefore, these conventional mechanisms cannot effectively deal with the ever evolving technologies and web-access techniques. Furthermore, these approaches also fall short in detecting the modern URLs such as short URLs, dark web URLs. In this paper, we propose a novel classification method to address the challenges faced by the traditional mechanisms in malicious URL detection. The proposed classification model is built on sophisticated machine learning methods that not only takes care about the syntactical nature of the URL, but also the semantic and lexical meaning of these dynamically changing URLs. The proposed approach is expected to outperform the existing techniques.

Keywords: Malicious URLs, Black-Listing, machine learning, URL Features, Cyber Crime.

I. INTRODUCTION

The human understandable URLs are used to identify billions of websites hosted over the present day internet. Adversaries who try to get unauthorized access to the confidential data may use malicious URLs and present it as a legitimate URL to naive user. Such URLs that act as a gateway for the unsolicited activities are called as malicious URLs. These malicious URLs can cause unethical activities such as theft of private and confidential data, ransomware installation on the user devices that result in huge loss every year globally. Even security agencies are cautious about the malicious URLs as they have the potential to compromise sensitive and confidential data of government and private organisations. With the advancement of social networking platforms, many allow its users to publish the unauthorized URLs. Many of these URLs are related to the promotion of business and self-advertisement, but some of these unprecedented resource locators can pose a vulnerable threat to the naive users. The naive users who use the malicious URLs, are going to face serious security threats initiated by the adversary.

The verification of URLs is very essential in order to ensure that user should be prevented from visiting malicious websites. Many mechanisms have been proposed to detect the malicious URLs. One of the basic feature that a mechanism should posses is to allow the benign URLs that are requested by the client and prevent the malicious URLs before reaching the user. This is achieved by notifying the user that it was a malicious website and a caution should be exercised. To achieve this, a system should take semantic and lexical properties of every URL rather than relying on syntactic properties of the URLs. Traditional methodologies such as Black-Listing[1], Heuristic Classification[2] has the ability to detect these URLs and block them before reaching the user.

Black-listing[1] is one of the basic and trivial mechanisms in detecting malicious URLs. Generally, Black-List is a database which contains the list of all URLs which are previously known to be malicious. 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. The update has to be made in black list whenever system comes across a new malicious URL. The technique is repetitive, time-consuming, and computationally intensive with ever increasing new URLs.

The other existing approach Heuristic classification[2] is an improvement to the Black-Listing. Here the signatures are matched and tested in order to find the correlation between the new URL and signature of existing malicious URL. Even though both Black-Listing and Heuristic Classification can effectively classify the malign and benign URLs, however, they cannot cope up with the evolving attack techniques. Recent statistics[2] imply that there is 20 - 25% growth in the attacks yearly and the threats that are coming from the newly created URLs are on the rise. One serious limitation of these techniques is that they are inefficient to classify the newly generated URLs.

One of the other collaborative work has been initiated by the top tier Internet companies such as Google, Facebook along with many of the startup companies to build a single platform that works all together for one cause of preventing the naive users from the malicious URLs. Many of these web-based companies use exhaustive data bases which can store as many as millions of URLs, and refine these URL

sets regularly. UBlock adblocker is a very good example here to mention, even though it is a manual procedure to update periodically, the performance was good and the database contains up-to-date URLs. But is this the feasible solution to all the problems? The answer is **NO**. Despite having the greater accuracy, the need for human intervention to update and maintain the URL list is one of the major limiting factors in this method.

To counter these limitations, we propose a novel approach using sophisticated machine learning techniques that could be used as a common platform by the Internet users. In this paper, we propose a technique in order to detect the malicious URLs. Various feature sets for the URL detection have also been proposed that can be used with Support Vector Machines (SVM). The feature set is composed of the 18 features, such as token count, average path token, largest path, largest token, etc. We also propose a generic framework that can be used at the network edge. That would safeguard the naive users of the network against the cyber attacks.

The organization of this paper follows: in Section II we have discussed various previous works of the field. Section III presents the proposed methodology. In Section IV, we have briefly discuss about the expected outcomes. Section V provides the concluding remarks and the future scope of the technique.

II. LITERATURE REVIEW

To overcome the limitations posed by the primitive classification methodologies like Black-Listing, Heuristic classification Research has been carried over the several areas and machine learning is one of the promising approaches to effectively classify the URLs [1] explains one of the several ways to leverage the machine learning in URL detection. Using the Supervised Machine Learning concepts such as Random Forest Model can classify at 89% without any tuning and feature selection.

Some Approaches in which detailed feature extraction required for precision classification. [2] uses the word level and character level Convolutional Neural Networks, as these underlying neural networks are quite handy in dealing with image data for computer vision tasks especially in deriving and learning from the salient features of the images from the raw pixel values. This approach produced better results by classifying the URL at a precision of 94%. This methodology uses the URL detection at the Character level and word level and finally to complete URL.

Usually, the problem will arise during the gathering of the data[4] by looking at the example we mentioned in the Introduction that UBlocker uses the manual updates of the URLs which is so hectic in reality, the general principle is to make the Automated model in order to collect the data, but they are so many difficulties in reality. Some of them are the URLs don't stay up for very long, some huge internet companies such as Google and Cisco, try to save the state of the Website and periodically this routine continuously follows. This is the reason why the research is going to extracting the Features[8] which are not too volatile, the main problem with the volatile features such as the size of the website, rate of requests to the websites are always kept on changing because since the internet is busty in nature so the growth was usually unpredictable.

For getting more insight about the URL, without digging too deep holes at one place some resources are quite helpful, one can use the Lexical Features as the classifying parameters in the Detection of Malicious URLs[5], by leveraging the Visible Attributes it is possible to Classify the Malicious Short URLs. The Social Network giants such as Twitter and Facebook use mainly these kinds of primitive features to Know whether to check, technically these systems are called Recommendation systems. We can derive four distinct categories of obfuscation techniques so that we can simply identify the benign from malicious[7], they propose the eighteen manually selected features in order to identify the variance. The four Obfuscation Features are the following (1) Obfuscating the host with an IP address, (2) Obfuscating the host with another domain, (3) Obfuscating with the large hostnames and, (4) Domain misspelled.

Another set of rules we can frame to determine the difference of the URLs, these kinds of rule-based determination are called as Heuristic detecting the methods[9], the rules are framed by the professionals in the field of the internet Security they are delegated to have the authority over to define, what is the behavior of Benign or behavior of Malicious this will help in the problem to search for the malicious URLs. Generally, the malware was run on simulated environments such as Sand Box, Virtual Machines and some Emulators.

By going a little bit deeper we can see how these features are turning to be, more effective in determining step. The major advantage of choosing the Lexical Features is they are effective and as well as can able to provide the lighter and super fast detection.

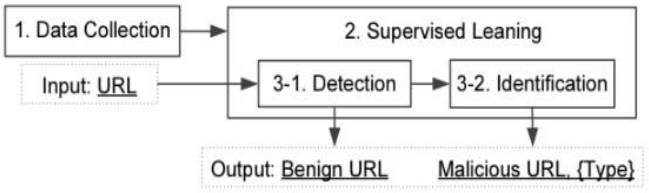

Figure. 1. A Framework of the proposed method.

The comparison has been made on the various machine learning techniques The detailed view of the results of various techniques has been elaborated in [6] stating that Convolution Neural Networks has shown good performance over the Support Vector Machine algorithm and Logistic Regression algorithm. Compared to the remaining Classification Techniques the Convolution Neural Networks has produced the precision of about 96% over the other two machine learning Techniques. The extensive research on the Deep Learning Technique [3] gives the insight about the Dynamic attack detection method in which the javascript was embedded to the URL to bypass the detection mechanisms the false positive rate produced by this technique is less than 4.2% in the best case.

The industry level work was carried out on [5] Twitter to detect the Malicious URLs in the Twitter website. Users in these kinds of websites believe one another and there will be more probability that users can blindly able to visit the site without any preprocessing. Twitter uses the Google safe browsing service, this method is of Kind of the Black-Listing service but the URL is changed quite Frequently. By [7] knowing how both the lexical and host-based features work, and how well we can use the Lexical Features Alone from the URLs. The [10] content-based approach a new Paradigm for URL Detection. Using[11] the parameters of the HTML, JavaScript, URL, Host Based Features can also help in determining the URLs. For the knowledge of knowing which classification mechanism[13] should be deployed in order to make complete use of the Features. We will see the Novel method[14] of using the DNS at the higher level hierarchy where the model will access the domains in the DNS.

III. METHODOLOGY

Any machine learning technique typically comprises of two steps: one is to obtain the appropriate feature representation that it could provide the determining insights in finding the Malicious URLs, and the second is to use this representation to train a learning-based prediction mechanism. In the proposed approach, we have provided the feature representation of the URLs. Analogically, Blood of this Process is the features and heart is the machine learning mechanism. Every time the blood passes through the heart the refining will happen. In the same manner features of the URLs will pass through the machine learning engine and then based on the previous learning the classification develops. In our case, we clearly followed the Lexical Analysis Features in addition to the 3rd party feature, geo ranking, altogether we gathered the feature set as shown in the TABLE:1. The test features are shown and the features are briefed with the categories that are involved in it.

The reason why lexical features[15] along with some of the behavioral features of the URLs will able to justify differentiation between the benign and malign is, majority of the new URLs are more likely to be having the same structure of existing malicious URLs.

In Figure.2 we briefly outlined the workflow of the Selection and Verification of the URLs. The source we took is the phish tank data. Phish Tank is the opensource that allow the registered users to add new malicious URLs that are not in the existing one. The Machine Learning Scoring that is in the second phase of the workflow will be the preprocessing phase where all the features are collected and converted to the numerical also called as metrics. Using the metrics that are obtained in the preprocessing.

Many methods are been proposed to fabricate the Classification Mechanism, Even though we are currently interested in just machine learning techniques, but out of all Convolutional Neural Networks(CNN) provided the better results this is because of the effective learning rate and quite suitable for the feature extraction[16]

To weight the importance of each token, we used the term frequency and inverse document frequency. The term token is the chunk of the URLs. A token can be any part of the URL including the domain and the path.

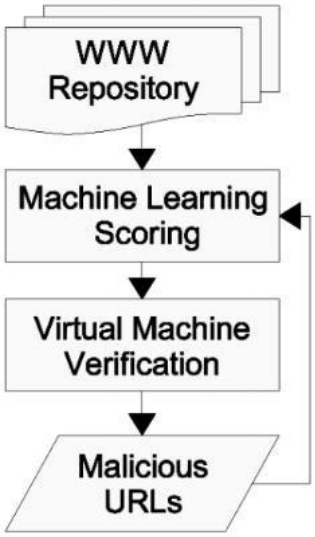

Figure. 2. URL selection and Verification WorkFlow

We are considering the Basic Minimum Feature set first which is related to the URL Physical Structure and that doesn't based on content-based properties. The reason being other techniques such as heuristics commonly uses the content based rules to deal with the classification. They are also several kinds of URL features can be used which will complement the Lexical Features some of them are Botnet features, WHOIS features, Host-Based features, Black List features and much more[17].

$$tf \cdot idf = tf(t, d) \cdot idf(t, D)$$

$$tf(t,d) = \frac{f(t,d)}{max\{f(w,d) : \omega \in d\}}$$

$$idf(t,D) = \log \frac{|D|}{|\{d \in D : t \in d\}|}$$

In the Black List Features, some of the metrics are of Real-Valued and some of them are Binary. We will know about the Botnet Features which was from the SpamAssassin Botnet plugin we have already discussed the Botnet in the introduction. This includes the presence of five other features which indicates the presence of the various corresponding client-server specific keywords. [18] This feature will usually represent whether the given URL hostname contains any of the IP address and also two more vital features which involve in the PTR record of the given Hostname. Table 1 shows comparison of classifiers.

Table 1: Comparison of Classifiers

Evalution Metric	Naive Bayesian	Decision Tree	K-Star
Accuracy	**95.41%**	89.9%	90.4%
F-Measure(Malicious)	**81.0%**	64.4%	76.3%
F-Measure(Legitimate)	**95.5%**	93.4%	94.0%
True Positive Rate	88.2%	80.9%	79.0%
False Positive Rate	93.6%	90.1%	93.0%
Positive Predictive Rate	74.9%	53.5%	73.7%
Negative Predictive Rate	97.3%	97.1%	94.8%

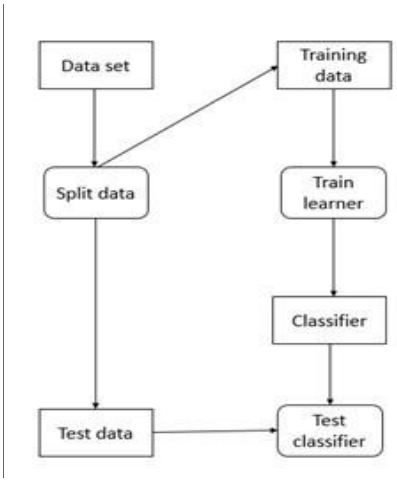

Figure 3: Machine Learning Classifier

Fig. 3 shows ML classifier. Here the set of 18 features which are based on the different reasoning is explained as follows, 1 Token count is a real-valued feature which takes the number of parts in the URL.[19] 2 An average token path is the average number of the Tokens that are present in the Path of the URL.3 Largest Path will infer the largest Token in the path with respect to the length. 4 Largest Token is the largest token among the overall URL which is also based on the length of the word which is nothing but the Token. [19]The Binary valued feature 5 IP Address presence will let the Analyser know whether the given URL contains the IP address which is in the Numerical. 6 Largest Domain Length will indicate the real-valued parameter, that indicates the Largest length of the token among the Domain name.

One of the key features of the URL is the 7 number of dots that are in the Given input. 8 Length of URL is the total length which is the sum of lengths of all tokens of the given input URL including every delimiter of the input. [20] 9 Path Token count will explain the number of tokens that are present in the Path of the URL. 11 Average token length of the URL is the sum of lengths of each token divided by the number tokens present in the URL. In the same way, 12 Average Domain Token Length is the sum of lengths of the domain tokens divided by the number of tokens in the Domain. [21] 13 The feature, Length of the Host is counting the number of the characters in the host part of the URL. 14 Security sensitive words are regarded as the some constrained set of words that usually appear in the Malicious URLs its impact will be on the Analyser. Fig. 4 shows block diagram 0f the proposed methodology. Autonomous System Number is the network parameter which will try to specify the path in which the URL came in as the response from the DNS. The Interesting feature we are used here is the 16 Safe Browsing which is a Binary valued and it '1' indicates the Benign and '0' indicates the malicious. Using the 17 Alexa 3rd party services we will include the Rank Host feature that will parse [26,29]the features of the URL and evaluating the rank procedure to identify the various classes of URLs. But ranking will deteriorate the performance of the model since the spammers can take the various features to inject the URL into the system. The best way is to trade- off between the ranking and feature selection.

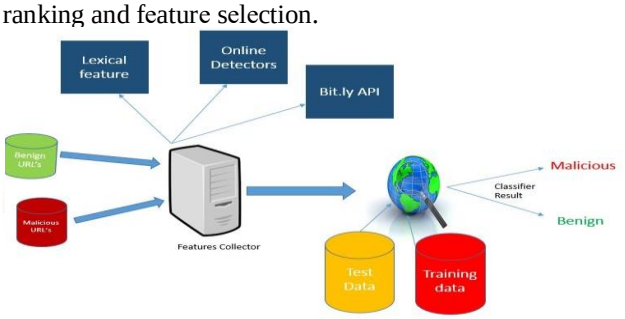

Figure. 4. Block Diagram of the proposed methodology.

IV. RESULTS

The presented work is still in its early state. The motive of this paper is to provide a brief about our approach. One assumption is that malicious URLs could be detected by extracting the lexical features. For doing the basic conduction we performed the Classifying method based on the TF - IDF word association. We can support the features that are extracted from the URL bigrams and term frequency and inverse term frequency will give the minimal classifying environment. But the classifying that uses the proposed features is the main task and we completed the preprocessing state. The presented work is an early effort in malicious URL detection, we will be covering the post process the Feature set and give the classifying coefficients which are used as the separating parameters as a future work.

V. CONCLUSION & FUTURE SCOPE

In this work, we have described how a machine can able to judge the URLs based upon the given feature set. Specifically, we described the feature sets and an approach for classifying the given the feature set for malicious URL detection. When traditional method fall short in detecting the new malicious URLs on its own, our proposed method can be augmented with it and is expected to provide improved results. Here in this work, we proposed the feature set which can able to classify the URLs. The Future work is to fine tuning the machine learning algorithm that will produce the better result by utilizing the given feature set. Adding to that the open question is how we can handle the huge number of URLs whose features set will evolve over time. Certain efforts have to be made in that direction so as to come up with the more robust feature set which can change with respect to the evolving changes.

REFERENCES

1. Justin. Ma, Lawerence. K. Saul, S. Savage, and G. M. Voelker, "Beyond blacklists: learning to detect malicious websites from suspicious URLs," in Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. New York, NY, USA: ACM, 2009, pp. 1245–1254

2. Mohammed Al-Janabi, Ed de Quincey, Peter Andras, "Using Supervised Machine Learning Algorithms to Detect suspicious URLs in online social networks", Proceedings of the IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017, https://dl.acm.org/citation.cfm?id=3116201 .
3. Hung Le, Quang Pham, Doyen Sahoo, Steven C.H Ho, "URL Net: Learning a URL Representation with Deep Learning for Malicious URL Detection", arXiv:1802.03162v2 Mar 2018.
4. Christophe Chong, Daniel Liu, Wonhong Lee, "Malicious URL Detection" Published at Stanford University, with Neustar, http://cs229.stanford.edu/proj2012/ChongLiu-MaliciousURLDetection.pdf.
5. R.k. Nepali and Y. Wang "You Look suspicious!!" Leveraging the visible attributes to classify the malicious short URLs on Twitter. in 49th Hawaii International Conference on System Sciences(HICSS) IEEE, 2016, pp. 2648-2655.
6. Doyen Sahoo, Chenghao Liu and Steven C.H.Hoi "Malicious URL Detection using Machine Learning A Survey", 2016, an article in the arxiv.
7. Anh Le, Athina Markopoulou, Michalis Faloutsos, "PhishDef: URL Names Say It All", proceedings in IEEE in INFOCOM (International Conference on computer communications) DOI: 10.1109/INFCOM.2011.593499, Published in 2011.
8. Rakesh Verma, Avisha Das, "What's in a URL: Fast Feature Extraction and Malicious URL Detection" proceeding IWSPA '17 Proceedings of the 3rd ACM on International Workshop on Security and Privacy Analytics Pages 55-63.
9. Y.Wang, W.-d Cai and P.-c Wei, "A deep learning approach for detecting malicious javascript code", Security and Communication Network, https://doi.org/10.1002/sec.1441, 2016.
10. Yue Zhang, Jason Hong, Lorrie Cranor, "CANTINA: A Content-Based Approach to Detecting Phishing Web Sites" International World Wide Web Conference Committee (IW3C2). www 2007, in may 8-12-2007, Banff, Alberta, Canada ACM 978-1-59593-654-7/07/0005 .
11. Davide Canali, Marco Cova, Giovanni Vigna, Christopher Kruegel, "Prophiler: A Fast Filter for the Large-Scale Detection" International World Wide Web Conference Committee (IW3C2), www, 2011 Hyderabad, India ACM 978-1-4503-0632-4/11/03.
12. Chai-Mei Chen, D.J. Guan, Qun-Kai Su, "Feature set identification for detecting suspicious URLs using Bayesian classification in social networks, Information Sciences, Elsevier, Journal Home Page: www.elsevier.com/locate/ins
13. Justin Ma, Lawernce K.Saul, Stefan Savage, Geoffrey M.Voelker, "Identifying Suspicious URLs: An Application of Large-Scale Online Learning" UC San Diego, Conference: Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009, Montreal, Quebec, Canada, June14-18, in 2009 https://www.researchgate.net/publication/221345258_Identifying_suspicious_URLs_An_application_of_large-scale_online_learning
14. Manos Antonakakis, Roberto Perdisci, Wenke Lee, Nikolaos vasiloglou II, and David Dagon," Detecting Malware Domains at the Upper DNS Hierarchy" Conference: Proceedings of the 20th USENIX conference on Security in August 2011 https://www.usenix.org/legacy/event/sec11/tech/full_papers/Antonakakis.pdf
15. Andre Bergholz, Gerhard Paab, Frank Reichartz, siehyun Strobel, Jeong-ho Chang, "Improved Phishing Detection using Model-Based Features " Conference: CEAS 2008 - The Fifth Conference on Email and Anti-Spam, 21-22 August 2008, Mountain View, California, USA, source DBLP
16. Abubakr Sirageldin, Baharum B. Baharubin, and Low Tang Jung, "Malicious Web Page Detection: A Machine Learning Approach", Advances in Computer Science and Its Applications, Springer-Verlag Berlin Heidelberg 2014
17. Colin Whittaker, Brain Ryner, Maria Nazif, "Large-Scale Automatic Classification of Phishing Pages", Conference: Proceedings of the Network and Distribution System Security Symposium, NDSS, 2010, Sandiego, California, USA.
18. Hyunsang Choi, Bin B. Zhu, Heejo Lee, "Detecting the Web Links and Identifying Their Attack Types", Proceedings of the 2nd USENIX conference on Web Application development Pages 11-11, Portland.
19. Adrienne Porter Felt, Richard Barnes, April King, Chris Palmer, Chris Bentzel, Parisa Tabriz, "Measuring HTTPS Adoption on the Web", Conference Paper, USENIX Security Symposium(2017), Vancouver, BC, https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/felt.
20. Kyle Soska, Nicolas Christin, "Automatically Detecting Vulnerable Websites Before They Turn Malicious", Conference Paper, USENIX Security Symposium, 2014, CA, ISBN 978-1-931971-15-7, https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/soska .
21. Birhanu Eshete, Fondazione Bruno Kessler, "Effective Analysis, Characterization, and Detection of Malicious Web Pages", Conference Paper, International World Wide Web Conference Committee(IW3C2) WWW 2013, Brazil, ACM 978-1-4503-2038-2/13/05.
22. Luca Invernizzi, Paola Milani Comparetti, "EvilSeed: A Guided Approach to Finding malicious Web Pages", Conference Paper, 2012 IEEE Symposium on Security and Privacy, DOI 10.1109/SP.2012.33.
23. Kyumin Lee, James Caverlee, Steve Webb, "Uncovering Social Spammers: Social Honeypots + Machine Learning", Conference Paper, 2010, SIGIR, Swiss
24. Pelin Zhao, Steven C.H.Hoi, "Cost-Sensitive Online Active Learning with Application to Malicious URL Detection", Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, August 11-14, 2013, Chicago. 919-927. Research Collection School Of Information Systems
25. Niels Provos Panayiotis Mavromatis, Moheeb Abu Rajab, Fabian Monrose, "All Your IFrames Point to Us", Conference:2008, 17th USENIX Security Symposium Association, https://www.usenix.org/legacy/events/sec08/tech/full_papers/provos/provos.pdf.
26. Hsing - Kuo Pao, Yan - Lin Chou, Yuh-Jyr Lee, "Malicious URL Detection Based on Kolmogorov Complexity Estimation", 2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology.
27. Frank Vanhoenshoven, Gonzalo Napoles, Rafael Falcon, Koen Vanhoy, and Mario Koppen, "Detecting Malicious URLs using Machine Learning Techniques", IEEE, Symposium for the Computational Intelligence for Defence and Security Applications.
28. Sangho Lee, Jong Kim "Warning Bird: Detecting Suspicious URLs in Twitter Stream", Conference (NIPA-2011-C1090-1131-0009), Network and Distributed System Security Symposium, 2012.
29. Adam Barth, Adrienna Porter Felt, Prateek Saxena, "Protecting Browsers from Extension Vulnerabilities" Conference Network and Distributed System Security and Symposium 2010.
30. Kurt Thomas, Justin Ma, Vern Paxson, Dawn Song, Chris Grier, "Design and Evaluation of a Real-Time URL Spam Filtering Service".