An Analytical Machine Learning to Approach to detect Cyber Attacks and Malicious URLs

Sonam Chauhan[1]*, Trapti Saxena*[2]
[1]*MTech Scholar,* [2]*Assistant Professor*
Department of ECE, RKDF University, Bhopal, M.P, India

Abstract--- The primary use of a URL (Uniform Resource Locator) is as a website address. However, some URLs may also be used to host uninvited content that may lead to cyberattacks. These URLs are known as malicious URLs. The legitimate user may be exposed due to the incapacity of the end user system to detect and remove the malicious URLs. Furthermore, using malicious URLs could result in unauthorised access to user data by an advertiser. The primary reason that fraudulent URLs are detected is because they offer an attack pathway to the advertiser.

It is crucial to stop these activities using some novel methodologies. Numerous filtering techniques have been used in literature to identify dangerous URLs. Some of them involve blacklisting, heuristic classification, and other techniques. These conventional mechanisms rely on URL syntax matching and keyword matching. As a result, these conventional mechanisms are unable to deal effectively with developing technologies such as web access techniques. Additionally, these approaches are ineffective in detecting contemporary URLs like short URLs and dark web URLs. In this article, we propose a novel classification method to address the difficulties faced by the established mechanisms for identifying malicious URLs.

The proposed classification model is based on sophisticated machine learning techniques that take into account not only the syntactical nature of URLs but also their semantic and lexical meanings. The proposed method is anticipated to perform better than existing methods.

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

I. INTRODUCTION

On the current day's internet, billions of websites are identified by human understandable URLs. Adversaries who attempt to get unauthorised access to sensitive information may present harmful URLs as legitimate ones to unsuspecting users. Malicious URLs are those that serve as a gateway for uninvited activities. These malicious URLs can lead to unethical activities including the theft of private and confidential data and the installation of ransomware on users' devices, which causes enormous loss every year around the globe. Even security agencies are wary of the malicious URLs because they have the potential to compromise sensitive and private information belonging to public and private organizations.

The development of social networking platforms has led to several of them enabling users to publish illegal URLs. Many of these URLs are related to the promotion of businesses and self-promotion, but some of these unexpected resource locators can present a threat to inexperienced users. The inexperienced users who use the fraudulent URLs would face serious security threats started by the advertiser. Verifying URLs is very important to ensure that users are prevented from visiting harmful websites. There have been numerous suggested mechanisms to identify rogue URLs. Allowing the benign URLs that are requested by the client and preventing the malicious URLs before they reach the user is one of the fundamental characteristics that a mechanism should have. This is accomplished by warning the user that the website was dangerous and that caution should be exercised. In order to accomplish this, a system should consider the semantic and lexical properties of each URL rather than relying solely on their syntactic properties. Traditional methodologies can detect these URLs and prevent them before they reach the user, including Black-Listing [1] and Heuristic Classification[2].

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 alternative existing method is called "Heuristic classification"[2] and it is an improvement over "Black-Listing." Here, the signatures are compared and tested to determine whether there is a correlation between the new URL and the signature of an existing malicious URL. Although Black-Listing and Heuristic Classification may effectively categorise malicious and benign URLs, they are unable to keep up with the ever evolving attack techniques. Recent statistics[2] indicate that there is a 20–25% annual increase in attacks, and threats coming from newly created URLs are on the rise. One significant drawback of these techniques is their inability to categorise newly generated URLs.

Another collaborative effort has been started by the largest Internet companies, including Google, Facebook, and many startup companies, to create a single platform that works together for the common goal of protecting innocent users from bad URLs. Many of these web-based businesses use exhaustive data bases that may store millions of URLs and refine these URLs sets regularly.

The UBlock ad blocker is a particularly nice example to mention because, despite the fact that it updates manually on a periodic basis, the performance was good and the database contains current URLs. But is this really the only feasible answer to all the issues? The response is "no." Despite the method's greater accuracy, one of its significant drawbacks is the requirement for human intervention to update and maintain the URL list.

We propose a novel strategy that makes use of sophisticated machine learning techniques and might be used as a platform by internet users to overcome these constraints. In this article, we propose a technique to identify malicious URLs. Additionally, a number of feature sets for URL detection have been proposed that can be used with support vector machines (SVM). The 18 features make up the feature set, which includes attributes like token count, average path tokens, largest path, largest token, etc. We also suggest a generic framework that can be used at the edge of the network. That would protect the network's innocent users from hacker attacks.

The organization of this paper is as follows. Section II discusses various previous works of this area . Section III presents the proposed methodology. In Section IV, we briefly discuss about the expected outcomes. Section V gives the conclusion.

II. LITERATUREREVIEW

The constraints of the early classification methodologies, such as Black-Listing and heuristic categorization, must be overcome. Machine learning is one of the potential approaches to effectively classify URLs that has been studied across a variety of fields. [1] explains one of the many techniques to advance machine learning in URL detection. Without any tuning or feature selection, 89 percent of the population may be classified using supported machine learning techniques like the random forest model.

Some methods that demand detailed feature extraction for precise categorization. Uses the terms level and character level in [2]. Convolutional neural networks are very useful 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 method classified the URL with a precision of 94%, yielding better results. This methodology uses the URL detection at the character and word levels before completing the URL.

Usually, the problem will arise during the gathering of the data[4] by looking at theexample 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 theFeatures[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 like Facebook and Twitter primarily use these kind of basic features to determine whether to check anything out; technically speaking, these systems are referred to as recommendation systems.

We can derive four distinct categories of obfuscation techniques so that wecan simply identify the benign from malicious[7], they propose theeighteen manually selected features in order to identify the variance. Thefour 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) Domainmisspelled.

Another set of rules that can be used to determine the differences between URLs is known as a heuristic detection technique[9]. These rules are created by experts in the field of internet security. Security is delegated the responsibility to define what constitutes benign or harmful behaviour, which will aid in the challenge of finding malicious URLs. The malware was mostly operated on simulated environments like Sand Box, Virtual Machines, and some Emulators.

Going a little bit deeper will allow us to see how these features are evolving to become more effective in the decision-making process. The main benefit of selecting lexical features is that they are effective and can deliver lighter and more rapid detection.

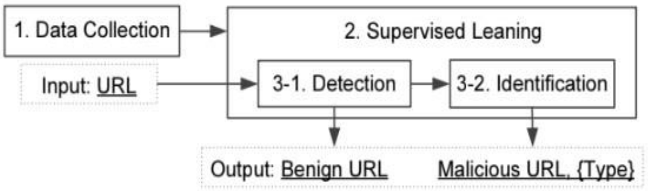

Figure. 1. Block Diagram of the proposed method.

The various machine learning techniques have been compared. Convolution neural networks have demonstrated superior performance than the Support vector machine algorithm and the Logistic Regression method, according to a detailed review of the results of numerous techniques in [6]. Convolution neural networks have produced results with a precision of roughly 96 percent over the other two machine learning techniques when compared to the remaining classification techniques. Extensive research on the Deep Learning Technique [3] provides insight into the Dynamic attack detection method in which javascript was embedded in the URL to get around the detection mechanisms; in the best case, this technique produces a false positive rate of less than 4.2 percent.

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 wecan use the Lexical Features Alone from theURLs. The [10] content-based approach a new Paradigm for URL Detection. Using[11] the parameters of theHTML, JavaScript, URL, Host Based Features can also help in determining theURLs. For the knowledgeof knowing which classification mechanism[13] should bedeployed in order to make complete useof 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 theDNS.

III. METHODOLOGY

Any method for computerised learning normally consists of two steps: In order to discover malicious URLs, one must first get the relevant feature representation, and the second is to use this representation to train a learning-based prediction mechanism. We have provided the feature representation of the URLs in the proposed technique. Analogously, the features and heart of this process are the machine's teaching mechanism. Every time the blood flows through the heart, refining takes place. The features of the URLs will similarly go through the machine learning engine, and based on the prior learning, the classification will develop.

In our case, we explicitly adhered to the Lexical Analysis. In addition to the third-party feature, geo ranking, we also gathered the features listed in the table below: 1. The test features are displayed and are broken down according to the categories that are involved.

The bulk of the new URLs are more likely to have the same structure as existing dangerous URLs, which is why lexical features[15] and some behavioural features of the URLs will be able to justify differentiating between the benign and malicious.

See Figure. 2 We briefly described the process of selecting and verifying the URLs. We obtained the information from the phish tank. The resource known as Phish Tank enables registered users to contribute new malicious URLs that are not already there. The processing phase of the machine learning scoring, which is in the second phase of the workflow, is when all the features are gathered and converted to numerical values known as metrics. utilizing the metrics obtained during the processing

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 theeffective learning rate and quite suitable for the featureextraction[16]

We utilized the term frequency and interspersed document frequency to weigh the importance of each decision. The chunk of the URLs is the term used. Any component of the URL, including the domain and the path, can be a token.

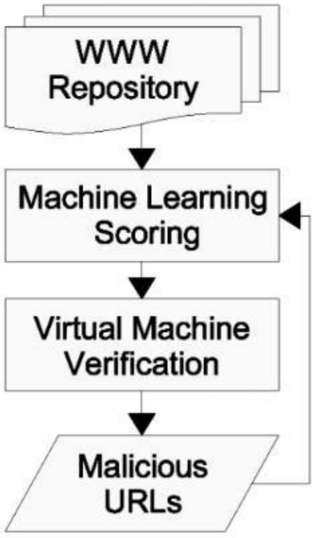

Figure. 2. URL selection and Verification WorkFlow

We are initially taking into account the Basic Minimum Feature established, which is associated with the URL Physical Structure and is not based on content-based properties. The reason for this is that other techniques, like heuristics, frequently use the content-based rules to deal with classification. There are numerous URL features that can be used to satisfy the lexical requirements; some of them include the botnet feature, WHOIS feature, host-based feature, black list feature, and many more[17].

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

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

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

Some of the metrics in the Black List features are real-valued, while others are binary. We already spoke about the Internet in the introduction, therefore we will now learn about the features of the Internet that were provided by the SpamAssassin Internet plugin. This also includes the presence of five more features, each of which denotes the presence of the relevant client-server-specific keywords. [18] This feature typically displays whether the given URL hostname contains any of the IP addresses as well as two more key characteristics related to the PTR record of the given hostname. A comparison of classifiers is shown in Table 1.

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%

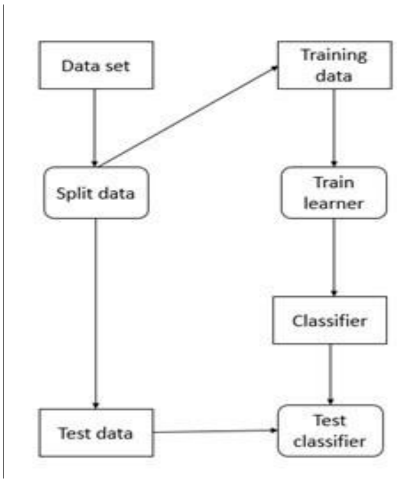

Figure 3: Machine Learning Classifier

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 theAnalyser 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 Domainname.

One of the key features of the URL is the seven dots present in the provided input. 8 The overall length of a URL is calculated as the sum of the lengths of all its tokens, including all input delimiters. [20] 9 Path Token count will explain how many tokens are included in the URL's path. 11 The average token length is calculated by adding the lengths of each token and dividing the result by the total tokens present in the URL.

In the sameway, 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, Lengthof theHost is counting the number of the characters in the host part of theURL. 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 theURL came in as the response from the DNS. The Interesting featureweare used here is the 16 Safe Browsing which is a Binary valued and it '1' indicates the Benign and '0' indicates the malevolent. Using the 17 Alexa 3rd party services wewill include the Rank Host feature that will parse [26,29]the features of theURL and evaluating the rank procedure to identify the various classes of URLs. But ranking will deteriorate the performanceof the model since the spammers can take the various features to inject theURL into the system. The best way is to trade- off between the ranking and feature selection.

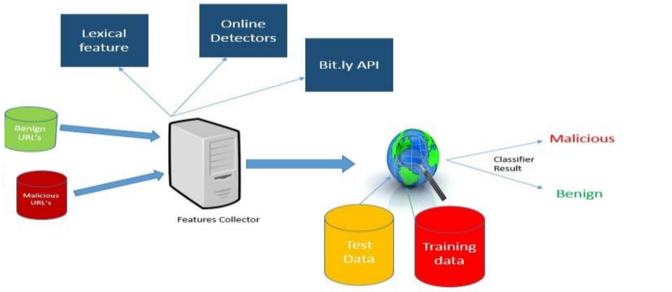

Figure. 4. Block Diagram of the proposed methodology.

IV. RESULTS

The work being presented is still in its early stages. This paper's goal is to explain our strategy in detail. One theory is that bad URLs could be detected by emulating the features of the language. The Classifying method based on the TF-IDF word association was used for the basic investigation. We are able to support the features that were taken from the URL bigrams, term frequency, and inverse term frequency, which will provide the bare minimum classification environment. The major task, however, and the point at which the proceeding state was completed, was classifying using the suggested features. The work that is being presented is an early attempt to identify dangerous URLs. In a subsequent effort, we plan to discover how the post processes the feature set and obtain the classification efficiency factors.

V. CONCLUSION

In this paper, we've described how a machine can evaluate URLs based on the specified features. In particular, we discussed the feature sets and a method for categorizing the given feature set for dangerous URL detection. Our proposed method can be supplemented with established methods when they are unable to detect the new harmful URLs on their own and is expected to produce improved results.

Here in this paper, we proposed a feature set that can categorize URLs. The work that needs to be done in the future is to fine-tune the machine learning algorithm so that it will use the given feature set to provide a better result. The additional concern is how we can manage the enormous number of URLs whose set features will change over time. To create a more resilient feature set that can adapt to changing circumstances, specific efforts must be made in that direction.

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 conferenceon Knowledge discovery and data mining. New York, NY, USA: ACM, 2009, pp.1245–1254

2. Mohammed Al-Janabi, Ed deQuincey, Peter Andras, “Using Supervised Machine Learning Algorithms to Detect suspicious URLs in online social networks”, Proceedings of theIEEE/ACM International Conferenceon 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 Mar2018.
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 themalicious shortURLs on Twitter. in 49th Hawaii International Conferenceon 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 thearxiv.
7. Anh Le, Athina Markopoulou, Michalis Faloutsos, “PhishDef: URL Names Say It All”, proceedings in IEEEin INFOCOM (International Conferenceon computer communications) DOI: 10.1109/INFCOM.2011.593499, Published in2011.
8. Rakesh Verma, Avisha Das, “What’s in a URL: Fast FeatureExtraction and Malicious URL Detection” proceeding IWSPA ‘17 Proceedings of the 3rd ACM on International Workshop onSecurity and Privacy Analytics Pages55-63.
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 WideWeb 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 WideWeb 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 HomePage: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 Conferenceon 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, WenkeLee, Nikolaos vasiloglou II, and David Dagon,” Detecting Malware Domains at the Upper DNS Hierarchy” Conference: Proceedings of the 20th USENIX conferenceon 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 Conferenceon Email and Anti-Spam, 21-22 August 2008, Mountain View, California, USA, sourceDBLP
16. Abubakr Sirageldin, Baharum B. Baharubin, and Low Tang Jung, “Malicious Web Page Detection: A Machine Learning Approach”, Advances in Computer Scienceand 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 conferenceon 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 IEEESymposium on Security and Privacy, DOI10.1109/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 OnlineActiveLearning with Application to Malicious URL Detection”, Proceedings of the 19th ACM SIGKDD International Conferenceon Knowledge Discovery and Data Mining, August 11-14, 2013, Chicago. 919-927. Research Collection School Of InformationSystems
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 AgentTechnology.
27. Frank Vanhoenshoven, Gonzalo Napoles, Rafael Falcon, Koen Vanhoy, and Mario Koppen, “Detecting Malicious URLs using Machine Learning Techniques”, IEEE, Symposium for the Computational Intelligencefor Defence and SecurityApplications.
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 Symposium2010.
30. Kurt Thomas, Justin Ma, Vern Paxson, Dawn Song, Chris Grier, “Design and Evaluation of a Real-Time URL Spam FilteringService”.