Mining DNS for Malicious Domain Registrations

Yuanchen He, Zhenyu Zhong, Sven Krasser, Yuchun Tang
McAfee, Inc.
4501 North Point Parkway, Suite 300
Alpharetta, GA 30022, USA
{yhe, ezhong, skrasser, ytang}@McAfee.com

***Abstract*— Millions of new domains are registered every day and the many of them are malicious. It is challenging to keep track of malicious domains by only Web content analysis due to the large number of domains. One interesting pattern in legitimate domain names is that many of them consist of English words or look like meaningful English while many malicious domain names are randomly generated and do not include meaningful words. We show that it is possible to transform this intuitive observation into statistically informative features using second order Markov models. Four transition matrices are built from known legitimate domain names, known malicious domain names, English words in a dictionary, and based on a uniform distribution. The probabilities from these Markov models, as well as other features extracted from DNS data, are used to build a Random Forest classifier. The experimental results demonstrate that our system can quickly catch malicious domains with a low false positive rate.**

I. Introduction

The Domain Name System (DNS) provides a convenient translation between domain names and IP addresses for Internet users and applications. Its widespread use makes it one of the most critical parts of the Internet infrastructure. DNS resolutions are triggered when a user browses a website, sends an email, or talks to friends via instant messenger service, etc.

In the past few years, DNS has been abused as a part of various attacks including phishing, spam, fast flux botnets, domain tasting, etc. DNS has become one of the weakest links exploited by attackers in different stages including the *domain registration stage* and the *resolution stage*.

- In the *domain registration stage*, arbitrary domain names can be registered at the official registrars. Such domain names, which many times contain meaningless strings, have been seen in attacks such as phishing or spam. For example, an adversary can put up a malicious webpage mimicking PayPal at http://www.paypal.com.example.com, which mimicks the actual PayPal page at http://www.paypal.com. The malicious webpage is then utilized to steal account information from users. To circumvent the domain blacklist, attackers change these meaningless domain names in the message body frequently. Domain tasting is another form of DNS abuse. Prior to 2009, domains could be registered free of charge for short periods of time. Attackers took advantage of this to seek out high value domains that users frequently access due to typos when trying to access popular sites. Also, domain tasting allowed attackers to cheaply use domain names for spam and phishing campaigns with no monetary disadvantage if the domain got blacklisted [3].
- In the *DNS resolution stage*, a particular malicious domain name can be resolved to a number of IP addresses to achieve its availability even part of the IP addresses are blocked by blacklists. Both DNS A record (providing the IP address for a given host) and NS records (providing the host name for the name server for a given domain) can be changed rapidly to provide a layer of redundancy, which complicates effective blocking of malicious machines.

All these attacks take advantage of the existing DNS, thus defending the DNS from being abused has become one of the foremost strategies to defend aforementioned attacks.

One way to determine whether websites contain malicious content it to simply fetch and analyze the content, either manually or using machine learning techniques. For example, PhishTank accepts submissions of suspicious phishing URLs, which are then verified by volunteers [1]. However, it takes considerable time for the human verification and a significant delay could be introduced such that a phishing site could vanish after conducting the crimes before it is verified and blacklisted.

Normal automatic malicious URL detection requires web crawling and webpage content analysis using machine learning techniques. Due to the large numbers of new domains registered every day (many of them being potentially malicious) and the generally short lifetime for such malicious domains, it is not cost effective to use the traditional web classification methods based on content. In order to avoid the large overhead of fetching the web content to detect DNS abuses as early as possible, researchers are facing two challenges in designing an efficient, low cost and effective approach:

- **Challenge 1:** *Detecting malicious DNS abuse behavior without introducing a significant amount of resource usage.*
- **Challenge 2:** *Classify a domain name without prior knowledge of its web content.*

To meet these two challenges, we present a novel approach that detects abnormal DNS behavior entirely based on the domain names and the related name server information. Our approach is very light-weight and performs in an efficient and effective way to detect malicious DNS behavior. This is accomplished without the need to fetch web content and is therefore not limited to domains for which content has been downloaded. In addition, this also renders this approach

COLLABORATECOM 2010, October 9-12, Chicago, USA

DOI 10.4108/icst.collaboratecom.2010.5

effective against malicious domains that have been registered but that are not set up for content distribution yet by the attacker.

II. Related Work

There are various related research efforts that attempt to address to some extent the vulnerabilities outlined above. The following sections outline the related work in the areas of domain/URL analysis and machine learning techniques utilized in the security space.

A. Malicious Domain/URL Detection

Most DNS abuses manifest themselves in the form of malicious URLs. These URLs could be phishing websites, spam websites, or websites that spread malware. Ma *et al.* perform Website classification based on inbound URLs [13][14]. Both lexical features and host-based features are extracted from each URL. Ma *et al.* consider both hostname and path. The host-based features considered are based on hostname information pertaining to IP addresses used, WHOIS properties, TTL of the DNS resource record, etc. The research investigates data on 15,000 benign URLs and 20,500 malicious URLs. Our approach differs in that we only consider the hostname (not the path) and that we utilize light-weight features that can be extracted from DNS zone data. Furthermore, our study is based on all data available in a complete top-level domain zone (.com) in DNS, granting us a global view of over hundreds of millions of domain names.

Cova *et al.* proposed techiniques to analyze rogue security software campaigns [7]. For the server side analysis of their approach, they use many network oberservable features including IP address, DNS names, other DNS entries pointing to the same IP, geolocation information, server identification string and version number, ISP identity, AS number, DNS registrar, DNS registrant, server uptime, etc. In our work, we only consider the zone based features. Their work focused on identifying rogue security software campaigns. Our work mainly focuses on domain classification.

Besides analyzing the domain names and URLs, content-based approaches have been proposed as well. Zhang *et al.* examine the detection of phishing websites based on the content of the URLs. The content is analyzed using a TF-IDF algorithm [19]. Anderson *et al.* investigated detection of spam by capturing the graphical similarity between rendered websites from the URLs in spam messages [4].

B. Proactive Detection with Machine Learning Techniques

Machine learning techniques have been adopted to proactively detect existing network attacks including spam, malware, phishing, etc. In our previous works, we extracted features from email messages and applied Support Vector Machines (SVM) and Random Forests (RF) classifiers for spam sender behavior analysis [16][17][18]. We also successfully applied Decision Tree (DT) and SVM on image header and file properties to identify image spam [12]. Sujata *et al.* used Logistic Regression (LR) classifier to model a set of heuristics including page rank, domain whitelist, obfuscation rules on URLs, and word-based features [10]. Fette *et al.* also proposed a similar approach to study URL characteristics and applied various classifiers including SVM, RF, and DT [8].

III. Features

The domain classification problem can be treated as binary classification problem where positive samples are malicious domains and negative samples are legitimate domains. In this work, two kinds of features are extracted for classification: textual features and zone features. Table I describes all features extracted in this work. Feature 1 to Feature 4 are the normalized markov transition probabilities for each domain name based on four transition matrices. Feature 5 to Feature 10 are the markov transition probability differences between any two markov transition probabilities for each domain name. Feature 11 to Feature 16 are the normalized markov transition probability differences between any two normalized markov transition probabilities for each domain name.

A. Textual Features

Many newly registered domains exhibit interesting textual sequence patterns. In many cases, a legitimate domain name consists of English words or looks like meaningful English words since these are easy to remember. Many newly registered malicious domain names are randomly generated and meaningless strings. To quantitatively describe this observation for the purpose of classification, we use several Markov Chain Models expressing the likelihood of textual sequences in domain names falling into different categories. In addition, we include other textual properties as features, such as the length of a domain name, the number of letters, the number of digits, and so forth. Refer to Table I for a complete list.

B. Zone Features

Typically, a legitimate domain will not change its hosting nameserver often while many malicious domains tend to change the hosting nameservers frequently. Also, our data shows that a large number of newly registered domains are malicious. Both observations indicate that such zone features carry discriminative power when used for classification. The zone features we consider in this work include the total number of nameservers that ever hosted a domain, the number of nameservers that hosted this domain but not host it anymore, the average/maximum/minimum string lengths of nameservers hosting it, and so on. Table I contains a comprehensive list of features used.

C. Feature Analysis

Two methods, signal-to-noise ratio (S2N) and the Random Forest (RF) based Gini coefficient [2][11], are used for preliminary feature analysis. S2N is the ratio of the strength of the signal and the strength of the noise. S2N is defined as

$$S2N_i = \frac{|\mu_i^+ - \mu_i^-|}{\delta_i^+ + \delta_i^-}$$

TABLE I

FEATURE SETS

Feature	Feature Name
1	normalized legitimate markov value
2	normalized malicious markov value
3	normalized English words markov value
4	normalized uniform distribution markov value
5	legitimate & malicious difference
6	legitimate & English words difference
7	legitimate & uniform distribution difference
8	malicious & English words difference
9	malicious & uniform distribution difference
10	English words & uniform distribution difference
11	legitimate & malicious normalized difference
12	legitimate & English words normalized difference
13	legitimate & uniform distribution normalized difference
14	malicious & English words normalized difference
15	malicious & uniform distribution normalized difference
16	English words & uniform distribution normalized difference
17	numbers in domain name
18	letters in domain name
19	hyphens in domain name
20	length of maximum number only substring
21	length of maximum letter only substring
22	length of maximum hyphen only substring
23	vowels count in domain name
24	Consonants count in domain name
25	#NS[1] hosted this domain
26	#non-active NS hosted this domain
27	non-active NS ratio
28	maximum days of any NS hosted this domain
29	minimal days of any NS hosted this domain
30	average days of all NS hosted this domain
31	#NS that hosted this domain less than 1 day
32	%NS that hosted this domain less than 1 day
33	#NS that hosted this domain less than 1 week
34	%NS that hosted this domain less than 1 week
35	#NS that hosted this domain between 1 week to 2 weeks
36	%NS that hosted this domain between 1 week to 2 weeks
37	#NS that hosted this domain between 2 weeks to 1 month
38	%NS that hosted this domain between 2 weeks to 1 month
39	#NS that hosted this domain longer than 1 month
40	%NS that hosted this domain longer than 1 month
41	#NS that hosted this domain in recent year
42	%NS that hosted this domain in recent year
43	Any NS newly host this domain in recent year

[1] NS = nameserver

where μ_i^+ and μ_i^- are mean values on the i^{th} feature of all positive/negative samples, δ_i^+ and δ_i^- are the corresponding standard deviations. A higher ratio indicates a feature is more likely to be informative.

We also analyze each feature with the decrease in Gini impurity from splitting on the feature in an RF classifier. In each node, Gini impurity can be calculated as

$$I_G = 2 \cdot \frac{N^+}{N} \cdot \frac{N^-}{N}$$

where N, N^+, and N^- are the number of total/positive/negative samples in the node. After each split, the sum of the Gini impurity values over two child nodes should be decreased compared to the Gini impurity on the parent node. The decreases are summed up for all splits on a feature in each tree and averaged over the RF as the Gini coefficient of the feature. The higher the Gini coefficient the more informative a feature is.

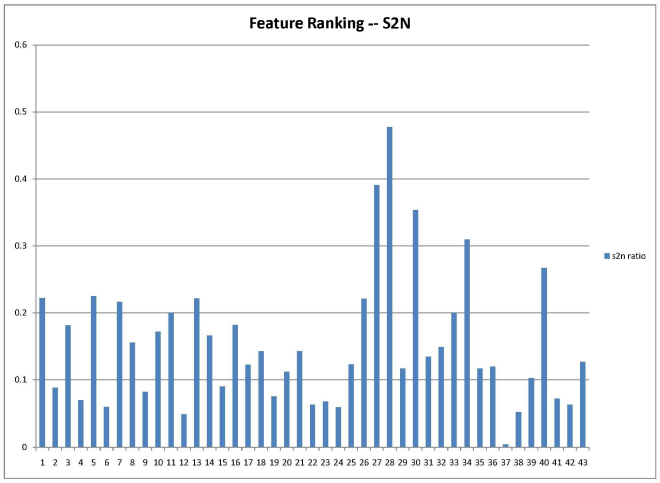

Fig. 1. s2n Feature Ranking

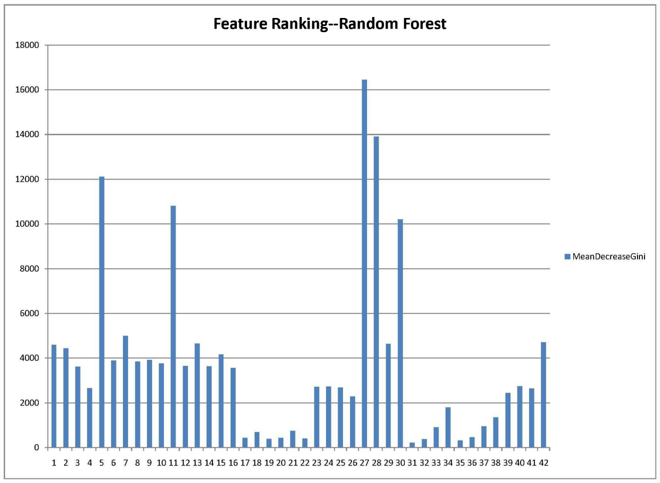

Fig. 2. Random Forest Feature Ranking

Fig. 1 depicts S2N values for all features while Fig. 2 shows Gini coefficient values. From those two feature ranking methods, the maximum days for a nameserver that ever hosted the domain (Feature 28 in Table I) and the non-active nameserver ratio (Feature 27 in Table I) are the most informative features. In general, the domain registry features are more informative than the textual features. For the textual features, the difference of legitimate Markov probability and malicious Markov probability (Feature 5 in Table I) is the most informative.

IV. CLASSIFICATION MODELS

A. Markov Model

Markov model is used to find the informative sequence patterns to discriminate malicious domains from legitimate domains. A Markov chain is a sequence of random variables X_1,X_2,X_3,... with the Markov property [15]. Markov property means "absence of memory" of a random process, the conditional probability distribution of future states of the process depends only on the present state and not on the past states. The Markov property is mathematically defined as

$$P\left(\mathbf{x_{n+1}} = x_{n+1}|\mathbf{x_n} = x_n, \mathbf{x_{n-1}} = x_{n-1}, ..., \mathbf{x_0} = x_0\right)$$
$$= P\left(\mathbf{x_{n+1}} = x_{n+1}|\mathbf{x_n} = x_n\right)$$

for every choice of n and value x_n.

A Markov chain of order m (or a Markov chain with memory m) where m is finite, is mathematically defined as

$$P\left(\begin{array}{l} \mathbf{x_{n+1}} = x_{n+1}| \\ \mathbf{x_n} = x_n, \mathbf{x_{n-1}} = x_{n-1}, ..., \mathbf{x_0} = x_0 \end{array}\right)$$
$$= P\left(\begin{array}{l} \mathbf{x_{n+1}} = x_{n+1}| \\ \mathbf{x_n} = x_n, \mathbf{x_{n-1}} = x_{n-1}, ..., \mathbf{x_{n-m+1}} = x_{n-m+1} \end{array}\right)$$

for every choice of n, m, and values $x_n, x_{n-1}, ...x_{n-m+1}$.

In this work, a second order Markov model ($m = 2$) is used to calculate the transition probability for the domain name. The second order Markov model transition probability is the conditional probability that the third character occurs in a three character-length sequence, given the occurrence of the first two characters. We build two transition matrices from malicious domain names and legitimate domain names separately according to the training dataset. If cross validation is used, the two matrices are generated in each fold separately without utilizing the validation dataset. Besides, we also generate a unique distribution transition matrix for all possible transitions for domain names, and a letters only transition matrix from an English dictionary. For a domain name, the probability from each transition matrix can be calculated and used as a feature for classification.

B. Logistic Regression

Logistic regression is one of the most commonly used statistical techniques. It is a type of predictive model that can be used when the target variable is a categorical variable or is in the form of a binomial proportion. Like linear regression, it estimates the relationship between input features and the target variable. Logistic regression, as a generalized linear model, has been widely used for binary classification problems [9].

C. Decision Tree

Decision Tree is one of the most popular classification algorithms current used in data mining and machine learning. A decision tree is a classifier expressed as a recursive partition of the instance space [5]. One advantage of decision tree is it can produce human-readable rules, and those rules can be used for classification.

D. Random Forest

Random forest consists of many independent decision trees, where each tree is grown using a subset of training samples randomly selected with replacement [6]. For each tree modeling, the splitting condition at each node is generated using a subset of the possible attributes randomly selected without replacement. In order to classify a prediction sample, using the each of the trees votes for the class label and the majority class label is assigned to the prediction sample.

TABLE II

DOMAIN DATA ON 03/14/2010

#domains	319,526
#malicious domains	140,554
#legitimate domains	178,972
#textual features	24
#domain registry features	19

TABLE III

CLASSIFICATION RESULT

Method	**AUC-7 CV**	**TP with 1% FP**
RF-US	0.87349	0.46436
RF-MV	0.88819	0.46463
DT	0.72996	0.25501
LR	0.81605	0.27700

V. EXPERIMENTS

A. Experiment Design

Table II describes the dataset used in this study. The top level domain for all the domains in the dataset is ".com." There are 319,526 domains, and the domain registry information is collected until March 14, 2010. For each domain we extract the 24 textual features and 19 domain registry features as described above. The task is to build a classifier to discriminate malicious domain names (labeled 1) from legitimate domain names (labeled -1). 7-fold stratified cross-validation is used to evaluate the real classification accuracy. The dataset is divided into 7 subsets. Each subset has approximately the same size, and approximately the same ratio of legitimate domains over malicious domains. In each fold, 6 subsets are combined for training, and 1 subset is used for validation. The validation accuracy for 7 folds are aggregated as the estimation of real classification accuracy.

We study three classification techniques, logistic regression, decision tree, and random forest. All of the three techniques are very popular to be used in many supervised learning applications. Compared to overall accuracy, we are more interested in identifying more malicious domains (TP) with very low FP rate, since the effect of a FP (wrongly classify a legitimate domain as a malicious domain) is more critical than a FN (miss a malicious domain).

The R package randomForest is used for random forest modeling. Two bias strategies are studied. One is undersampling, given the number of legitimate domains is n^- in training dataset, we use all n^- legitimate domains and random select $\frac{n^-}{20}$ malicious domains for training. Another strategy is bias the cutoff of the voting from trees. In our experiment 100 trees are built. A domain is predicted as malicious if and only if it is predicted as malicious by 98 or more trees. Otherwise it is classified as legitimate.

The R package rpart is used for decision tree modeling. The R package glmnet is used for logistic regression modeling. For these two techniques, simple weighting strategy is used so that FP cost is 100 and FN cost is 1.

B. Result Analysis

The modeling results are reported in Table III. The Area Under the Receiver Operating Characteristic Curve (AUC)

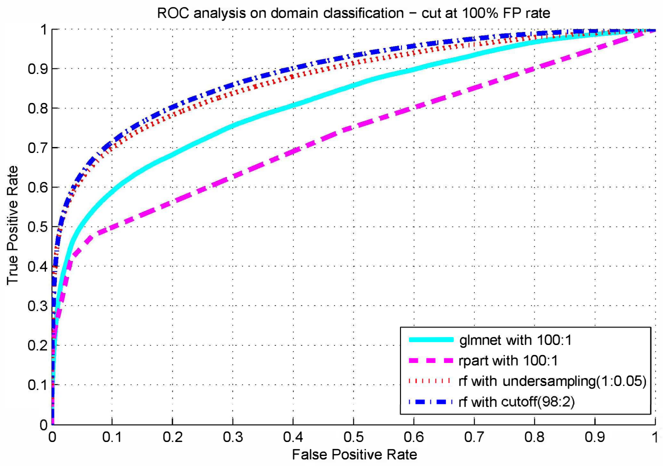

Fig. 3. ROC analysis-cut at 100% FP rate

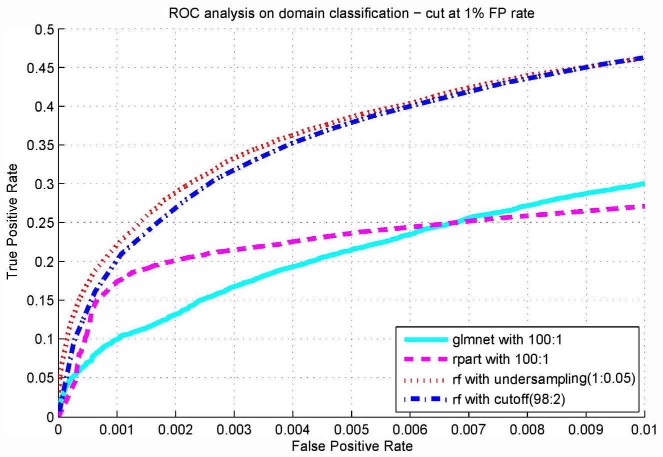

Fig. 4. ROC analysis-cut at 1% FP rate

matric is used for performance evaluation. RF-US denotes random forest modeling with the undersampling bias strategy. RF-MV denotes random forest modeling with major voting. DT denotes decision tree modeling, and LR denotes logistic regression modeling.

We observe that RF-MV is most accurate from AUC perspective. RF-US is very close. Compared to RF, LR and DT have much lower AUC values.

However, AUC values alone cannot justify the accuracy of the classifiers. In our real malicious domain detection system, a classifier with FP-rate larger than 1% is not acceptable. So we also evaluate TP-rate at a very low FP-rate (E.G. 1% FP-rate). We observe that RF-MV gives the highest TP-rate at 1% FP-rate, and RF-US has almost same performance as RF-MV, but for the even less FP-rate (less than 1%) RF-US has higher TP-rate than RF-MV.

Between LR and DT, DT has the smaller AUC value and less TP-rate at 1% FP, but for an even lower FP rate (FP rate less than 0.6%), DT performs better than LR.

Fig. 3 depicts ROC curves for all of the four classifiers. It shows that RF with both bias strategies have better overall performance than two other classifiers. Fig. 4 displays the ROC curves cut at 1% FP-rate. It clearly demonstrates that RF-US has the highest TP-rate at low FP-rate area. For example, at 0.3% FP-rate, RF-US can detect over 30% malicious domains.

VI. Conclusion

There are millions of new domains registered everyday. And our observation is that the majority of the newly registered domains are malicious. It is challenging, if not infeasible, to keep track of malicious domains by Web content analysis due to the large number of domains. One interesting pattern in legitimate domain names is that many of them consist of English words or look like meaningful English which are easy to remember, while many malicious domain names are randomly generated. So some character combinations rarely appear in malicious domain names may appear more often in legitimate domain names, and vice versa. In this work second order Markov models are used to transform this simple observation into useful features for classification. Four transition matrices have been built from known legitimate domain names, known malicious domain names, English words in a dictionary, and uniform distribution. The probabilities from these Markov models, as well as other features extracted from DNS data, like the number of nameservers which ever hosted a domain, the average string length of these nameservers, etc., are used as the input feature space for RF modeling and classification. The experimental results demonstrate that this very light-weight approach can detect many malicious domains with a low FP rate.

References

[1] "Opendns, phishtank," http://www.phishtank.com.

[2] "Random forests," http://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm#giniimp.

[3] G. Aaron, D. Alperovitch, and L. Mather, "The relationship of phishing and domain tasting," September 2007, anti-Phishing Working Group.

[4] D. S. Anderson, C. Fleizach, S. Savage, and G. M. Voelker, "Spamscatter: Characterizing Internet Scam Hosting Infrastructure," in *Proceedings of the USENIX Security Symposium, Boston, MA*, August 2007.

[5] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees*. Monterey, CA: Wadsworth and Brooks, 1984.

[6] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[7] M. Cova, C. Leita, O. Thonnard, A. Keromytis, and M. Dacier, "Gone rogue: An analysis of rogue security software campaigns," in *Proceedings of European Conference on Computer Network Defense (EC2ND)*, 2009.

[8] I. Fette, N. Sadeh, and A. Tomasic, "Learning to Detect Phishing Emails," in *Proceedings of the 16th International World Wide Web Conference, Baff, Alberta, Canada*, May 2007.

[9] J. Friedman, T. Hastie, and R. Tibshirani, "Regularization paths for generalized linear models via coordinate descent," 2009.

[10] S. Garera, N. Provos, and M. Chew, "A Framework for Detection and Measurement of Phishing Attacks," in *Proceedings of the ACM Workshop on Rapid Malcode (WORM), Alexandria, VA*, November 2007.

[11] C. W. Gini, "Variability and mutability, contribution to the study of statistical distributions and relations," *Studi Economico-Giuridici della R. Universita de Cagliari*, 1912.

[12] S. Krasser, Y. Tang, J. Gould, D. Alperovitch, and P. Judge, "Identifying image spam based on header and file properties using C4.5 decision trees and support vector machine learning," in *Proc. of the 2007 IEEE Systems, Man and Cybernetics Information Assurance Workshop (IAW 2007)*, 2007, pp. 255–261.

[13] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Beyond Blacklists: Learning to Detect Malicious Web Sites from Suspicious URLs," in *Proceedings of ACM SIGKDD Conference, Paris, France*, June 2009.

[14] J. Ma, S. Savage, L. K. Saul, and G. M. Voelker, “Identifying Suspicious URLs: An Application of Large-Scale Online Learning,” in *Proceedings of the International Conference on Machine Learning, Montreal, Quebec*, June 2009.

[15] S. Meyn and R. L. Tweedie, *Markov Chains and Stochastic Stability*. New York, NY, USA: Cambridge University Press, 2009.

[16] Y. Tang, Y. He, and S. Krasser, “Highly scalable SVM modeling with random granulation for spam sender detection,” in *Proceedings of IEEE Seventh International Conference on Machine Learning and Applications, ICMLA*, 2008, pp. 659–664.

[17] Y. Tang, S. Krasser, Y. He, W. Yang, and D. Alperovitch, “Support Vector Machines and Random Forests Modeling for Spam Senders Behavior Analysis,” in *Proceedings of IEEE Global Communications Conference (IEEE GLOBECOM 2008), Computer and Communications Network Security Symposium, New Orleans, LA*, 2008.

[18] Y. Tang, S. Krasser, P. Judge, and Y. Zhang, “Fast and effective spam IP detection with granular SVM for spam filtering on highly imbalanced spectral mail server behavior data,” in *Proceedings of The 2nd International Conference on Collaborative Computing (CollaborateCom 2006)*, 2006.

[19] Y. Zhang, J. Hong, and L. Cranor, “CANTINA: A Content-Based Approach to Detecting Phishing Web Sites,” in *Proceedings of the International World Wide Web Conference(WWW), Banff, Alberta, Canada*, May 2007.