Hindawi
Journal of Computer Networks and Communications
Volume 2019, Article ID 4612474, 9 pages
https://doi.org/10.1155/2019/4612474

Research Article

Malicious Domain Names Detection Algorithm Based on *N*-Gram

Hong Zhao[1], **Zhaobin Chang**[1], **Guangbin Bao**[1] and **Xiangyan Zeng**[2]

[1]*School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China*
[2]*Department of Mathematics and Computer Science, Fort Valley State University, Fort Valley, GA 31030, USA*

Correspondence should be addressed to Hong Zhao; 594286500@qq.com

Received 21 November 2018; Accepted 15 January 2019; Published 3 February 2019

Guest Editor: Saman S. Chaeikar

Malicious domain name attacks have become a serious issue for Internet security. In this study, a malicious domain names detection algorithm based on *N*-Gram is proposed. The top 100,000 domain names in Alexa 2013 are used in the *N*-Gram method. Each domain name excluding the top-level domain is segmented into substrings according to its domain level with the lengths of 3, 4, 5, 6, and 7. The substring set of the 100,000 domain names is established, and the weight value of a substring is calculated according to its occurrence number in the substring set. To detect a malicious attack, the domain name is also segmented by the *N*-Gram method and its reputation value is calculated based on the weight values of its substrings. Finally, the judgment of whether the domain name is malicious is made by thresholding. In the experiments on Alexa 2017 and Malware domain list, the proposed detection algorithm yielded an accuracy rate of 94.04%, a false negative rate of 7.42%, and a false positive rate of 6.14%. The time complexity is lower than other popular malicious domain names detection algorithms.

1. Introduction

While rapid development of Internet has changed our lives positively, different types of malicious cyberattacks have been increasing simultaneously. According to the 36th issue of the 2018 "Network information security and dynamic weekly report" released by CNCERT, Chinese servers were attacked 178,156 times in 2018 by 3,724 malicious domain names such as Conficker, Trojan, and Srizbis, an year-over-year increase of 55.8% [1].

DNS (domain name system), one of the basic services for realizing the conversion between network domain name and IP addresses in the Internet [2], has been widely used in e-commerce, instant messaging, and network media. Almost all Internet applications are needed to use DNS to resolve domain names and achieve resource location [3, 4].

In order to achieve malicious purpose, attackers implant malicious programs through the vulnerabilities of system or service to infect the host, and the infected host is controlled by attackers remotely [5]. The infected host will issue resolution requests, using a large number of nonexistent domain names randomly generated by the DGA (domain generation algorithm) or domain flux [6] in a short time. These resolution requests of malicious domain name are forwarded multiple times among DNS servers and are discarded eventually due to the failure of domain name resolution. However, the record of the failure of domain name resolution is also forwarded multiple times, then returned to the infected host that initiates the domain name resolution request. A large number of resolution requests and resolution failure records of the malicious domain name are forwarded multiple times among the DNS servers, which increases the usage of network bandwidth and brings a heavy payload on the DNS servers. Moreover, it will affect the execution of normal domain name resolution tasks seriously as well. If the malicious domain name is not detected in an accurate and timely manner, the DNS servers may be down due to malicious domain name attacks, all Internet services relying on DNS servers will stop, and the results will be catastrophic. Therefore, accurate and timely detection of malicious domain name attacks has the significant impact on Internet security.

The remaining of this study is organized as follows. A number of related works are reviewed in Section 2. The proposed approach, system architecture, and progress to detect malicious domain names are presented in Section 3.

The experimental results and performance evaluation of the study are presented in Section 4, and the conclusion is presented in Section 5.

2. Related Works

From the perspective of domain name structure features and lexical composition, there are two main types of malicious domain names detection methods in the literature: domain name model [7–11] and domain name semantic [12–15] detection.

2.1. Domain Name Model Detection. There are many differences between the normal domain names and the malicious domain names in terms of behaviors and structures. Therefore, the legitimacy of domain names can be determined by analyzing the behavior and structure of domain names. For example, Truong et al. [16] used the DNS traffic characteristics to detect the DNS query flow of abnormal DNS servers. Zang et al. [17] proposed a malicious domain names detection algorithm based on AGD (algorithmically generated domain) by using cluster correlation that identifies the domain names generated by a domain generation algorithm or its variants. Features such as TTL, the distribution of IP addresses, Whois features, and historical information from the domain names in each cluster were extracted, and the support vector machine (SVM) algorithm was used to identify the malicious domain names. Sharifnya and Abadi [18] proposed a DGA-based botnet detection algorithm by clustering the hosts of the DNS request query and generating the potential candidate set to be tested. Their algorithm achieves malicious domain names detection by calculating the probability of the threatened candidate hosts in the candidate set to be tested. Kwon et al. [19] proposed a botnet detection algorithm based on DNS traffic features using PSD (power spectral density) testing technology, which detects malicious domain names by analyzing malicious behavior within large volumes of DNS traffic. Zhang et al. [20] proposed a botnet detection algorithm that combined the domain names' request behaviors with construction characteristics and conducted malicious domain names detection through SVM classification. Vinayakumar et al. [21] proposed a method that used big data computation platforms on a distributed cluster and deep learning algorithm to detect fraud and malicious activities where attackers used combinational method to avoid blacklist detection.

2.2. Domain Name Semantic Detection. The method of domain name semantic detection includes detection based on character matching and on content analysis. For example, Yadav et al. [22] proposed a malicious domain names detection algorithm by using linguistic features that measures information entropy of bigrams in all domains and statistical measurements such as Kullback–Leibler divergence, Jaccard index, and Levenshtein edit distance for identifying malicious domain names. Huang et al. [23] analyzed the differences between normal domain names and malicious domain names in character constitution. The statistical characteristics, resolution features, and similarity characteristics of domain names were extracted, and malicious domain names detection was achieved using a machine learning algorithm in the character and resolution features space. Zhang et al. [24] proposed a lightweight domain names detection algorithm based on morpheme features and natural language processing (NLP), which analyzed the domain name features such as the root, affix, Chinese spelling, and special noun abbreviation and used the C4.5 algorithm to construct a decision tree with recursive thinking. Zhang et al. [25] proposed a malicious domain names detection algorithm that analyzed the domain name features of character composition and the lexical hierarchical structure which included domain name length, double letters, and character frequency to distinguish malicious domain names. Zhao [26] proposed a high efficiency pattern matching detection algorithm for intrusion detection. The algorithm reduces computational time by constructing a hash table for attack pattern strings.

Each of the above malicious domain names detection methods has its own advantages. The domain name model detection methods have high detection accuracy rate and wide application range. However, this kind of detection method has a long data collection period, and it is difficult to obtain a large amount of resolution data from both the local domain name server and the root domain name server, thus resulting in high detection time overhead. Although the detection methods of domain name semantic have the advantage of low detection time overhead, this kind of detection method is based on domain name blacklist to design detection features and cannot effectively detect newly generated domain names.

To address the problems such as low detection accuracy rate and high detection overhead, a new method of malicious domain names detection based on *N*-Gram is proposed by combining domain name model and semantic features. In addition, unlike current methods that analyze the lexical composition and structure of the whole domain names, the new method divides the whole domain names into multiple domain name substrings and deeply analyzes the features of domain name substrings in terms of lexical composition and structure. In this algorithm, the domain names which are accessed with high frequency are chosen as the normal domain names sample, and the *N*-Gram method is applied to segment each domain name in the domain name whitelist sample to obtain the substrings on which domain name model and semantic features depend. Then, a test domain name is also segmented by the *N*-Gram method, and its substrings are compared with the domain name substrings in domain name whitelist substring set to determine whether it is a malicious domain name.

3. Proposed Approach

In the following sections, we provide more details for the components of the proposed algorithm.

3.1. System Overview. Figure 1 gives a flowchart of our proposed detection algorithm. Malicious domain names detection algorithm based on *N*-Gram consists of two main sections. (1) Domain name whitelist substring set construction. (2) Malicious domain names detection. There are two steps to construct the domain name whitelist substring set, namely, obtaining the substring statistics from the normal domain names segmented by the *N*-Gram method and calculating the substrings weight values. The domain names with high frequency of access excluding the top-level domain are segmented into multiple substrings according to its domain name level with the lengths of 3, 4, 5, 6, and 7 by using the *N*-Gram method. Then, substring statistics is calculated according to each domain name substring repetition occurrence number. The substring weight value is calculated according to the domain name substring occurrence frequency in the substring set.

In the process of malicious domain names detection, a test domain name is segmented also by the *N*-Gram method, and its reputation value is calculated according to the weight values of its substrings. Then, the malicious domain name is determined according to the reputation value.

3.2. Domain Name Whitelist Substring Set Construction. Alexa ranking is a service that Amazon provides to the public to evaluate the popularity of domain names [27]. Through the statistics and analysis of each domain name in the number of access, links, and other aspects, the domain name evaluation and ranking are produced according to the analysis results. Therefore, if a domain name ranks relatively high in the Alexa, it is more likely to be secure and normal.

Through the observation of a large number of normal domain names in Alexa 2013 [28], it is found that the domain name has the features of hierarchical structure on its composition form. Top-level domain is the domain name substring of the end of a domain name, including the country top-level domains (such as cn, jp, and gb) and international organization top-level domains (such as com, net, and org). Moreover, the total number of top-level domains is small, and there are unified naming rules in terms of its lexical structure and lexical composition [29]; namely, top-level domains are generally named on the basis of the name of country or region. For example, for a given domain name lut.edu.cn, cn is China's top-level domain on the Internet. Second-level domain (SLD) is the domain name substring that is adjacent to the top-level domain. For example, edu is a SLD, which represents education organization. Third-level domain (TLD) refers to the domain name substring that is adjacent to the right of SLD. For example, lut is a TLD, which is the abbreviation of the Lanzhou University of Technology. Therefore, domain name substrings at each level have a specific meaning in its construction.

In the process of domain name resolution, when the domain name resolution request is not recorded in local network DNS servers, the resolution request is forwarded to the superior network DNS servers, until it reaches the root domain DNS servers. After reaching the root domain, the resolution requests are forwarded to the DNS servers again where the top-level, second-level, third-level, and other level domain names are located, until the domain name resolution result is found. Given that the domain name resolution request is forwarded from the superior domain to the inferior domain, if the given inferior domain name does not exist, the domain name resolution fails. Then, the domain name resolution result or the cause of the domain name resolution failure will be returned to the host that initiates the domain name resolution according to the original path of the request.

From the domain name resolution process, it is noted that the deeper level a malicious domain name is at, the greater its forwarding number is, thus the heavier burden it creates on the system. Conversely, the closer a malicious domain name is to the top-level domain, the smaller its forwarding number is, and thus the easier it can be found. In addition, because of the small quantity, short length, and high popularity of top-level domains, they are readily identified. Therefore, malicious domain names are rarely found in the top-level domain, but often exist in the second, third, or fourth domain. Hence, this study focuses on each domain name substring excluding the top-level domain.

3.2.1. Substring Statistics. The character string in the text is segmented by a sliding window with a size of *N*, and multiple substrings of length *N* are obtained, each of which is called a gram. For example, the process of 5-Gram segmentation for a string "microsoftword" of length 13 is shown in Figure 2.

When the *N*-Gram method is applied to segment the text, the value of *N* will influence the number of gained substrings. If the value of *N* is too small, the number of domain name substrings obtained by segmentation will be large, which leads to enormous calculation quantity and storage space. If the value of *N* is too large, the number of domain name substrings obtained by segmentation will be small, which can lead to little effective lexical feature information obtained by segmentation, which is not conducive to extract domain name composition structure and semantic information. After excluding the top-level domains of top 100,000 domain names in Alexa 2013, domain name substring length proportions of each level are counted, as shown in Table 1.

As seen from Table 1, after excluding top-level domains of top 100,000 domain names in Alexa 2013, the proportion of the length of each level domain name in the [3, 7] interval is up to 97.63%. Therefore, the size of the sliding window *N* is set to 3, 4, 5, 6, and 7, and each domain name excluding top-level domains is segmented by the *N*-Gram method to form the domain name substring set.

For example, after removing the top-level domain "com" of wapseo.chinaz.com, the process of second-level and third-level domains is segmented by the *N*-Gram method as shown in Figure 3. The second-level domain substring set is {chi, hin, ina, naz, chin, hina, Inaz, china, hinaz, chinaz}. And the third-level domain substring set is {wap, aps, pse, seo, waps, apse, pseo, wapse, apseo, wapseo}.

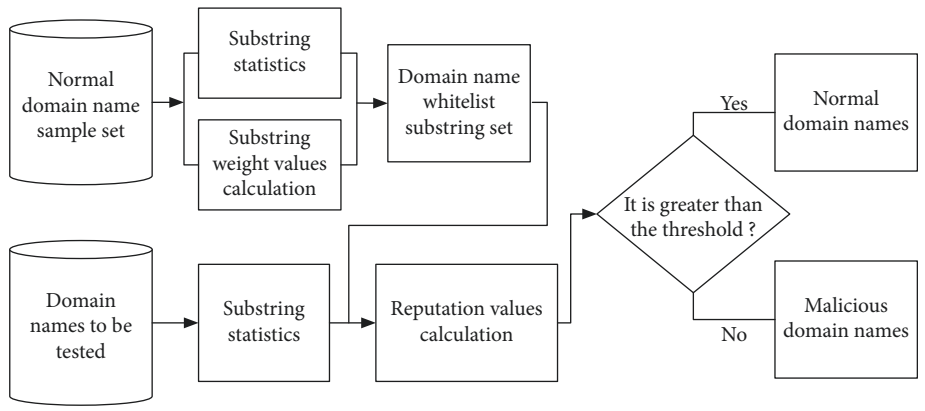

Figure 1: Flowchart of malicious domain names detection algorithm based on N-Gram.

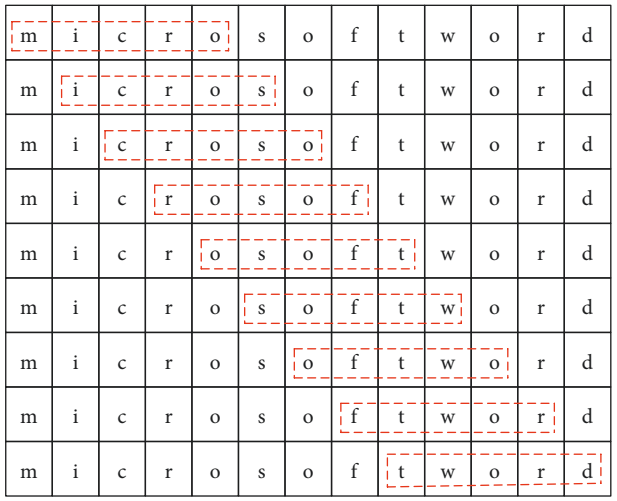

Figure 2: Principle diagram of 5-Gram segmentation.

Table 1: Each level substring length proportion.

Length	3	4	5	6	7
Proportion (%)	5.39	20.09	29.13	29.21	13.81

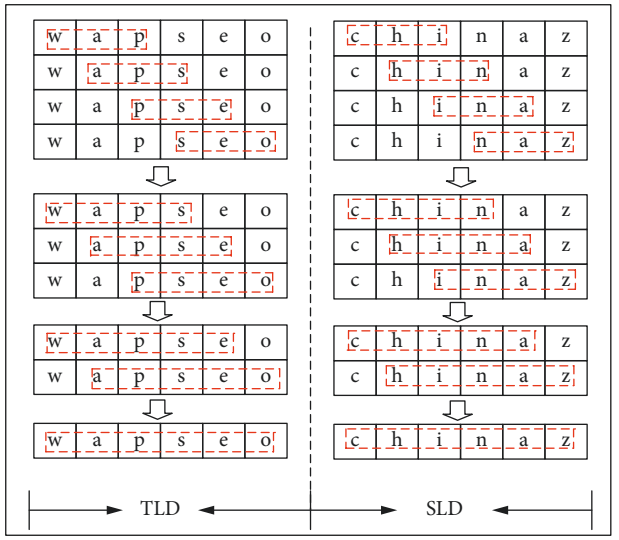

Figure 3: Process of domain name segmentation.

The top 100,000 domain names in Alexa 2013 are selected, and each domain name excluding the top-level domain is segmented into multiple domain name substrings according to its domain level with the lengths of 3, 4, 5, 6, and 7 by the N-Gram method. Furthermore, the duplicate domain name substrings are removed, and the 398,823 completely different domain name substrings are obtained as a domain name whitelist substring set. The number of domain name substrings at each domain level and of each sliding window size are calculated by the following formula:

$$\text{count}(j) = L - N + 1, \tag{1}$$

where count(j) ($j = 1, 2, \ldots, n$) represents the number of domain name substrings that are obtained from segmenting the jth-level domain of a domain name, L represents the length of jth-level domain, n represents the maximum level number of a domain name, and N represents the size of the sliding window whose value ranges from $\{N \in N^* | 3 \le N \le 7\}$.

When the size of the sliding window N is set to 3, 4, 5, 6, and 7, the number of completely different domain name substrings obtained is shown in Table 2. Where the number of domain name substrings with the length of 3 is 21,584, with the length of 4 is 84,431, with the length of 5 is 120,626, with the length of 6 is 116,908, and with the length of 7 is 55,274, with a total of 398,823 domain name substrings. And the domain name whitelist substring set is constructed according to these completely different domain name substrings.

3.2.2. Substring Weight Values Calculation. The extraction of the lexical features of the domain name turns into numerical calculation by calculating the weight values of 398,823 domain name substrings in the domain name whitelist substring set. The weight value of domain name substring is calculated by the following formula:

$$W_{N-\text{Gram}}(i) = \log_2\left(\frac{C_{N-\text{Gram}}(i)}{N}\right), \tag{2}$$

where $W_{N\text{-Gram}}$ ($N = 3, 4, 5, 6$, and 7) is the weight value of the ith domain name substring and $C_{N\text{-Gram}}(i)$ represents the

Table 2: The number of domain name substrings generated when $N = 3, 4, 5, 6,$ and 7.

Sliding window	Number
3	21,584
4	84,431
5	120,626
6	116,908
7	55,274
Total	398,823

total number of the occurrences of the ith domain name substring after the top 100,000 domain names are segmented in Alexa 2013.

398,823 domain name substrings are extracted from the top 100,000 domain names in Alexa 2013 by the N-Gram method, and each domain name substring weight value is calculated. Total score of each domain name to be tested is calculated according to these domain name substring weight values. Partial domain names substring weight values from the top 100,000 domain names in Alexa 2013 are shown in Table 3.

3.3. Malicious Domain Names Detection. In the process of malicious domain names detection, a test domain name is segmented by the N-Gram method, and its reputation value is calculated according to the weight values of its substrings in the domain name whitelist substring set. Finally, the judgment of whether a domain name is malicious is made according to its reputation value. The process of the malicious domain names detection is shown in Figure 4.

3.3.1. Reputation Values Calculation. Each domain name excluding the top-level domain is segmented into multiple substrings with the lengths of 3, 4, 5, 6, and 7 by the N-Gram method. The total weight value of the domain names to be tested is calculated according to the weight values of its substrings in the domain name whitelist substring set. Additionally, the total weight value is used as the reputation value (RV) to evaluate whether the domain name to be tested is a malicious domain name. The RV is calculated by the following formula:

$$\mathrm{RV}(l) = \sum_{i=1}^{m} W_{N\text{-}Gram}(i), \tag{3}$$

where $W_{N\text{-Gram}}$ ($N = 3, 4, 5, 6,$ and 7) is the weight value of ith domain name substring which is referenced from 398,823 domain name substring weight values (as shown in Table 3), $l\{l \in N^* | l > 0\}$ represents a domain name to be tested that the serial number is l, and m represents the total number of obtained domain name substrings whose domain name l is segmented, when the sliding window N is set to 3, 4, 5, 6, and 7. Since the substrings of the malicious domain names appear less frequently in the domain name whitelist substring set, the RV of malicious domain names is smaller. On the contrary, the RV of normal domain names is larger. Therefore, a simple technique of thresholding can be used to achieve the detection of malicious domain names.

Table 3: Partial domain names substring weight values from the top 100,000 domain names in Alexa 2013.

N-Gram	$C_{N\text{-Gram}}(i)$	$W_{N\text{-Gram}}(i)$
ing	3139	10.031
ter	2105	9.454
line	1270	8.310
blog	1194	8.221
direc	587	6.875
forum	452	6.498
ectory	341	5.828
ogspot	293	5.609
rectory	341	5.606
youtube	220	4.974
rketing	167	4.576

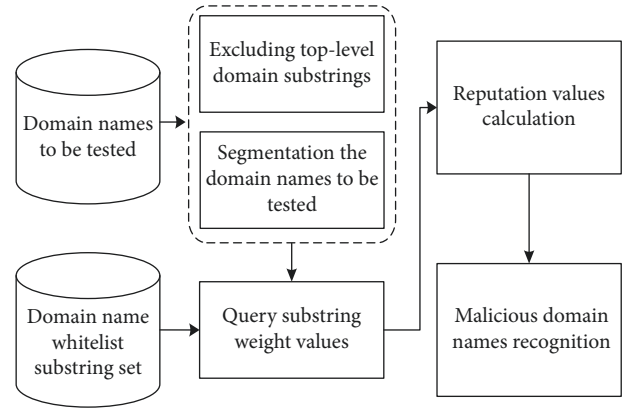

Figure 4: Flow of malicious domain names detection.

For example, after removing the top-level domain "com" and replacing the letter "o" in the normal domain name "taobao.com" with the number "0," the RV of the normal domain name "taobao.com" and the malicious domain name "ta0ba0.com," which are similar to the normal domain name "taobao.com," can be calculated. The RV is calculated as follows:

$$\begin{aligned}
\mathrm{RV}_{\mathrm{taobao}} &= W_{\mathrm{tao}} + W_{\mathrm{aob}} + W_{\mathrm{oba}} + W_{\mathrm{bao}} + W_{\mathrm{taob}} \\
&\quad + W_{\mathrm{aoba}} + W_{\mathrm{obao}} + W_{\mathrm{taoba}} + W_{\mathrm{aobao}} + W_{\mathrm{taobao}} \\
&= 2.736 + 3.807 + 2.321 + 3.222 + 0.807 \\
&\quad + 1.459 + 1.321 + 0.485 + 0.847 + 0.222 \\
&= 17.227, \\
RV_{\mathrm{ta0ba0}} &= W_{\mathrm{ta0}} + W_{\mathrm{a0b}} + W_{\mathrm{0ba}} + W_{\mathrm{ba0}} + W_{\mathrm{ta0b}} \\
&\quad + W_{\mathrm{a0ba}} + W_{\mathrm{0ba0}} + W_{\mathrm{ta0ba}} + W_{\mathrm{a0ba0}} + W_{\mathrm{ta0ba0}} \\
&= 0 + 0 + 0.415 + 0 + 0 + 0 + 0 + 0 + 0 \\
&\quad + 0 + 0 \\
&= 0.415.
\end{aligned} \tag{4}$$

By calculating the RV of the normal domain name "taobao.com" and the malicious domain name "ta0ba0.com," it can be seen that the RV of the normal domain name "taobao.com" is 17.227. When the size of N is 3, 4, 5, 6, and 7,

the normal domain name "taobao.com" is segmented into multiple domain name substrings which appear frequently in the domain name whitelist substring set. However, the RV of the malicious domain name "ta0ba0.com" is 0.415. When the size of N is 3, 4, 5, 6, and 7, the malicious domain name "ta0ba0" is segmented into multiple domain name substrings which appear with very small probability in the domain name whitelist substring set.

3.3.2. Threshold Setting for Malicious Domain Names Detection. In the process of malicious domain names recognition, the size of threshold decides the accuracy rate of the detection algorithm in this study. In order to attain the superior detection accuracy rate, the variable parameter threshold D is debugged on the same dataset and the optimal threshold D is selected. The corresponding relationship between the threshold D size and the detection accuracy rate is shown in Figure 5.

From Figure 5, it can be seen that the curve of detection accuracy rate shows a trend of left rising and right falling. When threshold D is 0.65, the detection accuracy rate reaches the optimal level of 94.04%, which refers to it as a better detection effect when the threshold D is 0.65. Therefore, threshold D in this study is set to 0.65.

In this study, the threshold for malicious domain names detection is set on the basis of the domain name whitelist substring set that is constructed by the top 100,000 domain names in Alexa 2013. If the domain name whitelist sample on constructing domain name whitelist substring set is replaced, the threshold for malicious domain names detection needs to be reset according to the above steps.

When the threshold is set, the RV of the domain name to be tested is calculated to judge whether the domain name to be tested is malicious based on the size of the RV and threshold for malicious domain names detection. If the RV of the domain name to be tested is greater or equal than the threshold for malicious domain names detection, the domain name is judged to be a normal domain name. If not, it is a malicious domain name.

4. Experimental and Result Analysis

To verify the performance of the proposed algorithm based on N-Gram, experiments on malicious domain names detection were conducted using large volumes of data collected and published by Alexa and other sources.

4.1. Experiments. The experimental environment is shown in Table 4.

4.2. Experimental Data

4.2.1. Domain Name Whitelist Substring Set. The top 100,000 domain names in Alexa 2013 are selected as the domain name whitelist sample set. Each domain name excluding the top-level domain is segmented into multiple domain name substrings according to its domain level with the lengths of 3, 4, 5, 6, and 7 by the N-Gram method. 398,823 completely different domain name substrings are extracted as the domain name whitelist substring set.

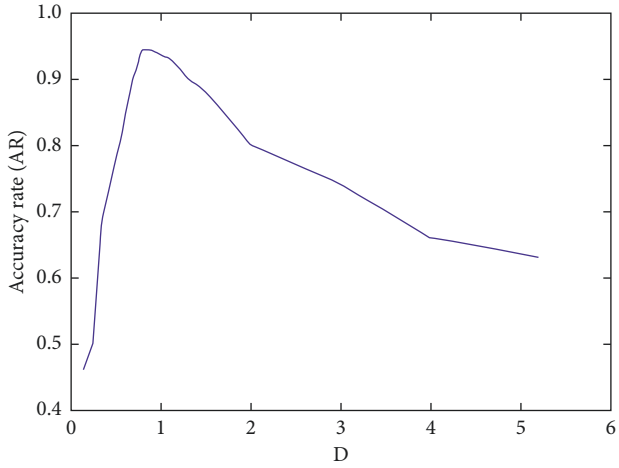

Figure 5: Relationship curves of accuracy rate with D.

Table 4: Experimental environment.

Parameters	Value
CPU	AMD A12-9700 2.5 GHZ
GPU	AMD R8 M435DX
Memory	8 GB
OS	64-bit Windows 10
Platform	Jupyter Notebook
Python	3.5

4.2.2. Domain Name Sample Set to Be Tested. In this study, 10,265 domain names from the Alexa 2017 and Malware domain list are collected and collated [30, 31]. The 8,000 domain names with the highest number of accesses in Alexa 2017 are taken as the normal domain name sample set, and the 2,265 domain names in Malware domain list (malicious domain names that are generated by the DGA [32], botnet [33], Conficker [34], and Spam [35]) are taken as the test sample set of malicious domain names.

4.3. Evaluation Standard. In order to evaluate the performance of the malicious domain names detection algorithm based on N-Gram in malicious domain names detection, the accuracy rate (AR), false negative rate (FNR), and false positive rate (FPR) are used. The evaluation standard is calculated based on the confusion matrix [36] of the experimental results, as shown in Table 5. Evaluation standard is calculated by the following formula:

$$\begin{aligned} \mathrm{AR} &= \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{FP}+\mathrm{TN}+\mathrm{FN}} \times 100\%, \\ \mathrm{FNR} &= \frac{\mathrm{FN}}{\mathrm{TP}+\mathrm{FN}} \times 100\%, \\ \mathrm{FPR} &= \frac{\mathrm{FP}}{\mathrm{TN}+\mathrm{FP}} \times 100\%, \end{aligned} \tag{5}$$

Table 5: Confusion matrix of TN, FN, FP, and TP.

Actual	Predicted	
	Negative	Positive
Negative	True negative (TN)	False positive (FP)
Positive	False negative (FN)	True positive (TP)

where TN represents the number of normal domain names that are correctly detected by the algorithm, FN represents the number of malicious domain names that are incorrectly reported as normal domain names, FP represents the number of normal domain names that are incorrectly reported as malicious domain names, and TP represents the number of malicious domain names that are correctly detected by the algorithm.

4.4. Result Analysis. Figure 6 shows the effect of threshold D on false negative rate and false positive rate, where Figure 6(a) shows the relation between the threshold D on the X-axis and the false negative rate on the Y-axis and Figure 6(b) shows the relation between the threshold D on the X-axis and the false positive rate on the Y-axis.

From the results of the two experiments presented in Figure 6, when the threshold D is less than 0.65 and gradually increases within this range, the curve of detection accuracy rate takes on ascend trend (Figure 5). The main reason is that the number of malicious domain names accurately detected by the algorithm increases and the number of malicious domain names incorrectly reported as normal domain names gradually decreases, which makes the detection accuracy rate increase gradually and the false negative rate decrease gradually (Figure 6(a)). When the threshold D is greater than 0.65 and it gradually increases within this range, the curve of detection accuracy rate takes on downward trend (Figure 5). The reason is that the number of normal domain names accurately detected by the algorithm decreases and the number of normal domain names incorrectly reported as malicious domain names gradually increases, which makes the detection accuracy rate decrease gradually and the false positive rate increase gradually (Figure 6(b)).

4.5. Comparison with Other Approaches. To verify the effectiveness of the proposed algorithm, experiments were also performed using the methods in the latest literatures [8, 10, 13, 14] with the same experimental conditions. Accuracy and computational cost are the measures for the effectiveness of the algorithms. Performance comparisons in terms of AR, FNR, FPR, and time overhead (TO) are shown in Table 6.

Our proposed method yielded a superior combinational result of accuracy rate and computational efficiency. Other methods either has lower accuracy rate or is computational more expensive. In addition, compared to the other methods that are based on machine learning techniques, our method is much easier to add new data when they become available. While the machine learning algorithms require a new training process of all the data, our method only needs modifications to the weight of relevant substrings.

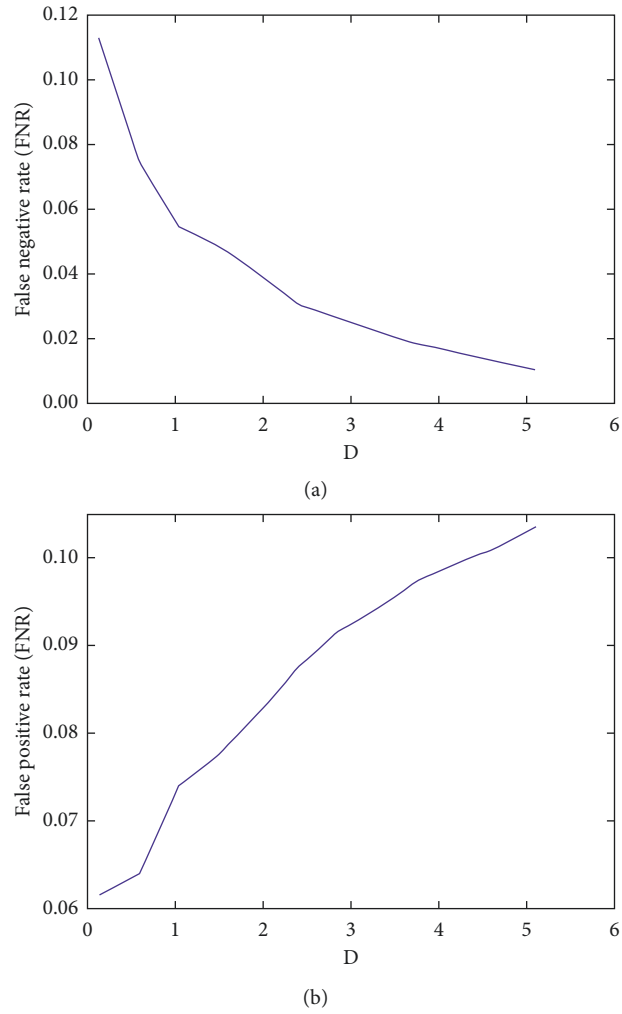

Figure 6: Threshold D effect on (a) false negative rate curve and (b) false positive rate curve.

Table 6: The performance comparison between our approach and methods in [8, 10, 13, 14].

Method	AR (%)	FNR (%)	FPR (%)	TO (s)
Shi et al. [8]	95.75	4.29	6.62	42.68
Ma et al. [10]	91.04	2.65	7.11	18.92
Wu et al. [13]	91.52	8.48	1.50	32.08
Song et al. [14]	93.47	4.42	7.43	38.27
Our approach	94.04	7.42	6.14	31.75

5. Conclusion and Future Work

This study proposes a new method for malicious domain names detection. The main contributions are as follows:

(1) The top 100,000 domain names in Alexa 2013 are taken as the domain name whitelist sample set to construct the domain name whitelist substring set

(2) The N-Gram method in the natural language processing technology is used to segment the domain names, and the malicious domain names are quickly recognized according to its occurrence number in the substring set

Compared to the malicious domain names detection methods proposed by [8, 10, 13, 14], our approach demonstrated a superior comprehensive performance of lower time overhead and higher detection accuracy rate. It has a good practical value in defending against the botnet, Spam, and remote access Trojan attack and can help security experts and organizations in their fight against cybercrime. However, our proposed approach is not comprehensive in homographic domain names (such as linkedin.com and 1inkedin.com, apple.com and app1e.com) detection aspects, but if the malicious domain names are generated randomly, our approach can detect them efficiently. These homographic domain names detection problems should be considered in the future.

Data Availability

The Alexa and Malware domain list datasets used to support the findings of this study are cited at relevant places within the text as references [28, 30, 31, 33–35].

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research work was supported by the National Science Foundation of China under Grant nos. 51668043 and 61262016, the CERNET Innovation Project under Grant nos. NGII20160311 and NGII20160112, and the Gansu Science Foundation of China under Grant nos. 18JR3RA156.

References

[1] National Internet Emergency Center, "36th internet security threat report," 2018, http://www.cert.org.cn/publish/main/44/index.html.

[2] W. Quan, C. Xu, J. Guan, H. Zhang, and L. A. Grieco, "Scalable name lookup with adaptive prefix bloom filter for named data networking," *IEEE Communications Letters*, vol. 18, no. 1, pp. 102–105, 2014.

[3] S. Yadav, A. K. K. Reddy, A. L. N. Reddy, and S. Ranjan, "Detecting algorithmically generated domain-flux attacks with DNS traffic analysis," *IEEE/ACM Transactions on Networking*, vol. 20, no. 5, pp. 1663–1677, 2012.

[4] W. Quan, C. Xu, A. Vasilakos, and J. Guan, "TB2F: tree-bitmap and bloom-filter for a scalable and efficient name lookup in content-centric networking," in *Proceedings of the IFIP Networking Conference*, pp. 1–9, Trondheim, Norway, June 2014.

[5] L. Bilge, S. Sen, D. Balzarotti, E. Kirda, and C. Kruegel, "Exposure," *Acm Transactions on Information and System Security*, vol. 16, no. 4, pp. 1–28, 2014.

[6] R. Sharifnya and M. Abadi, "DFBotKiller: DFBotKiller: domain-flux botnet detection based on the history of group activities and failures in DNS traffic," *Digital Investigation*, vol. 12, no. 12, pp. 15–26, 2015.

[7] B. Yu, L. Smith, M. Threefoot, and F. Olumofin, "Behavior analysis based DNS tunneling detection and classification with big data technologies," in *Proceedings of International Conference on Internet of Things and Big Date*, pp. 284–290, Rome, Italy, April 2016.

[8] Y. Shi, G. Chen, and J. Li, "Malicious domain name detection based on extreme machine learning," *Neural Processing Letters*, vol. 48, no. 3, pp. 1347–1357, 2017.

[9] S. Tian, C. Fang, J. Liu, and Z. Lei, "Detecting malicious domains by massive DNS traffic data analysis," in *Proceedings of the 8th International Conference on Intelligent Human-Machine Systems and Cybernetics*, pp. 130–133, Zhejiang, China, August 2016.

[10] Z. Ma, H. Chen, J. Yang, and X. L., "Novel network intrusion detection method based on IPSO-SVM algorithm," *Computer Science*, vol. 45, no. 2, pp. 231–235, 2018.

[11] P. Kintis, N. Miramirkhani, C. Lever, and Y. Chen, "Hiding in plain sight: a longitudinal study of combosquatting abuse," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, Dallas, TX, USA, August 2017.

[12] P. Zhang, T. Liu, Y. Zhang, J. Ya, and J. Shi, "Domain watcher: detecting malicious domains based on local and global textual features," in *Proceedings of the International Conference on Computational Science*, pp. 2408–2412, Zurich, Switzerland, June 2017.

[13] Z. Wu, J. Zhang, M. Yue, and C. Zhang, "Approach of detecting low-rate DoS attack based on combined features," *Journal on Communications*, vol. 38, no. 5, pp. 19–30, 2017.

[14] W. Song and B. Li, "A method to detect machine generated domain names based on random forest algorithm," in *Proceedings of the International Conference on Information System and Artificial Intelligence*, pp. 509–513, Hong Kong, China, June 2017.

[15] C. Xiong, P. Li, P. Zhang, Q. Liu, and J. Tan, "MIRD: trigram-based malicious URL detection implanted with random domain name recognition," in *Proceedings of the 6th International Conference on Applications and Techniques in Information Security*, pp. 303–314, Beijing, China, November 2015.

[16] D. Truong, G. Cheng, A. Jakalan, X. Guo, and A. Zhou, "Detecting DGA-based botnet with DNS traffic analysis in monitored network," *Journal Of Internet Technology*, vol. 17, no. 2, pp. 217–230, 2016.

[17] X. Zang, J. Gong, and X. Hu, "Detecting malicious domain name based on AGD," *Journal on Communications*, vol. 39, no. 7, pp. 15–25, 2018.

[18] R. Sharifnya and M. Abadi, "A novel reputation system to detect DGA-based botnets," in *Proceedings of the 3rd International Conference on Computer and Knowledge Engineering*, pp. 417–423, Mashhad, Iran, October 2013.

[19] J. Kwon, J. Lee, H. Lee, and A. Perrig, "PsyBoG: PsyBoG: a scalable botnet detection method for large-scale DNS traffic," *Computer Networks*, vol. 97, pp. 48–73, 2016.

[20] Y. Zhang, Y. Lu, and Y. Zhang, "Detecting domain flux through patterns of domain names' alphanumeric characters and querying behavior of hosts," *Journal of Xian Jiaotong University*, vol. 47, no. 8, pp. 54–60, 2013.

[21] R. Vinayakumar, K. Soman, and P. Poornachandran, "Detecting malicious domain names using deep learning approaches at scale," in *Proceedings of the 3rd International Symposium on Intelligent Systems Technologies and Applications*, pp. 1355–1367, Manipal, India, September 2017.

[22] S. Yadav, A. K. K. Reddy, A. L. N. Reddy, and S. Ranjan, "Detecting algorithmically generated malicious domain names," in *Proceedings of the 10th ACM Sigcomm Conference on Internet Measurement*, pp. 48–61, Melbourne, Australia, November 2010.

[23] K. Huang, J. Fu, J. Huang, and P. Li, "A malicious domain detection approach based on characters and resolution features," *Computer Simulation*, vol. 35, no. 3, pp. 287–292, 2018.
[24] W. Zhang, J. Gong, X. Liu, and X. Hu, "Lightweight domain name detection algorithm based on morpheme features," *Journal of Software*, vol. 27, no. 9, pp. 2348–2364, 2016.
[25] Y. Zhang, Y. Zhang, and J. Xiao, "Detecting the DGA-based malicious domain names," in *Proceedings of the International Standard Conference on Trustworthy Computing and Services*, pp. 130–137, Beijing, China, November 2013.
[26] L. Zhao, "Research on a high efficiency pattern matching algorithm for intrusion detection," *Computer and Digital Engineering*, vol. 45, no. 8, pp. 1592–1596, 2017.
[27] D. A. Orr and L. Sanchez, "Alexa, did you get that? Determining the evidentiary value of data stored by the Amazon Echo," *Digital Investigation*, vol. 24, pp. 72–78, 2018.
[28] Alexa top global sites, 2013, http://www.alexa.com/topsites.
[29] E. Casalicchio, M. Caselli, and A. Coletta, "Measuring the global domain name system," *IEEE Network*, vol. 27, no. 1, pp. 25–31, 2013.
[30] Alexa top global sites, 2017, http://www.alexa.cn/siterank/14.
[31] Malware domain list, 2017, http://www.malwaredomainlist.com.
[32] S. Schüppen, D. Teubert, P. Herrmann, and U. Meyer, "FANCI: feature-based automated NX- Domain classification and intelligence," in *Proceedings of the 27th USENIX Security Symposium*, pp. 1165–1181, Baltimore, MD, USA, August 2018.
[33] DNS-BH malware domain blacklist, 2016, http://www.malwaredomains.com.
[34] Phish Tank, 2013, http://www.phishtank.com.
[35] Blacklist provided by joewein.net (JWSDB), 2015, http://joewein.net/spam/blac-klist.htm.
[36] A. Aborujian and S. Musa, "Cloud-based DDOS http attack detection using covariance matrix approach," *Journal of Computer Networks and Communications*, vol. 2017, Article ID 7674594, 8 pages, 2017.