Hindawi
Security and Communication Networks
Volume 2021, Article ID 1028878, 9 pages
https://doi.org/10.1155/2021/1028878

Research Article

A Hybrid Association Rule-Based Method to Detect and Classify Botnets

Yuanyuan Huang,[1] **Lu Jiazhong**,[1] **Haozhe Tang**,[1] **and Xiaolei Liu**[2]

[1]*School of Cyberscurity, Chengdu University of Information Technology, Chengdu 610225, Sichuan, China*
[2]*Institute of Computer Application, China Academy of Engineering Physics, Mianyang, Sichuan 621900, China*

Correspondence should be addressed to Xiaolei Liu; liuxiaolei@caep.cn

Received 19 May 2021; Revised 4 August 2021; Accepted 25 August 2021; Published 17 September 2021

Academic Editor: Weiwei Liu

Nowadays, botnet has become a threat in the area of cybersecurity, and, worse still, it is difficult to be detected in complex network environments. Thus, traffic analysis is adopted to detect the botnet since this kind of method is practical and effective; however, the false rate is very high. The reason is that normal traffic and botnet traffic are quite close to the border, making it so difficult to be recognized. In this paper, we propose an algorithm based on a hybrid association rule to detect and classify the botnets, which can calculate botnets' boundary traffic features and receive effects in the identification between normal and botnet traffic ideally. First, after collecting the data of different botnets in a laboratory, we analyze botnets traffic features by processing a data mining on it. The suspicious botnet traffic is filtered through DNS protocol, black and white list, and real-time feature filtering methods. Second, we analyze the correlation between domain names and IP addresses. Combining with the advantages of the existing time-based detection methods, we do a global correlation analysis on the characteristics of botnets, to judge whether the detection objects can be botnets according to these indicators. Then, we calculate these parameters, including the support, trust, and membership functions for association rules, to determine which type of botnet it belongs to. Finally, we process the test by using the public dataset and it turns out that the accuracy of our algorithm is higher.

1. Introduction

Botnet is a group of centrally controlled bots on the Internet, and these computers using the botnet are called controlled hosts, which are often utilized by hackers to launch a large-scale cyberattack. These computers contain spams port scans, phishing sites, etc. The botnet host can also control the information stored in those computers, such as passwords of bank account and social accounts. In the meantime, hackers can also get the function of "access" of their computers easily. No matter it is the safe operation of the network or the users' data security protection, the botnets are perilous risks. However, current technology cannot recognize those botnets easily for they are usually controlled by hackers long distantly. In other words, users are often unaware of these hosts.

Nowadays, the main botnet detection algorithms are to detect network traffic. The existing detection methods have some shortcomings, however. For instance, if we only rely on bots' similarity detection method, the result is prone to get false. When it is difficult to determine the number of clusters by using the clustering algorithm, we need to establish a blacklist to complete the test. However, the blacklist depends on the bot's malicious attacks, and the efficiency of detection will be quite low.

According to different classification criteria, there are several classification methods of botnet detection: host-based detection methods, network-traffic-based detection method, and real-time detection methods. Host-based detection methods detect botnets by analyzing information logs and acting on the host. Because botnets will bring a series of changes while running, such as changing the registry, skipping the firewall, establishing a network connection, bypassing intrusion detection, and turning off antivirus software[1]. System changes caused by bots and legal programs are very different; thus, some research detected botnets by analyzing the host information [2–5].

Host-based detection method detects at a faster speed but with a lower cost. However, to apply host-based detection, the method needs to install specific software on each host, indicating its poor expansibility and adaptability. In addition, because of the various information in hosts, the formats of different operating systems are not the same information sources, which make this method difficult to be adopted. Therefore, the detection methods based on network traffic become the mainstream. Moreover, there is a method combining host and network traffic, and we can get the details from the literature [6, 7].

Methods based on network traffic detection can be divided into two types: active detection and passive detection. Active detection sends probe packets to the Internet to detect the existence of bots, while passive detection collects network traffic passively, analyzing and processing network traffic to detect botnets after that.

Active detection method has higher detection efficiency, it can detect whether there exists a botnet swiftly. However, the active detection method has some obvious shortcomings: the probe packets sent with the help of this method will add additional traffic to the network. It means that attackers can easily find them out and then avoid being detected.

In most of the time, passive detection technology can acquire network traffic from the measured network core, switch firstly, then analyze and process the collected traffic, and finally detect botnets. Passive detection technology will not generate extra traffic, and so attackers will not find it easily.

The detection method based on real-time reduces the detection time to a few seconds [8] without affecting the detection accuracy. Because of Botnets' long delay in the HTTP response, it can be used as a result of request relaying through the botnet proxy. This process usually takes extra time, and the nodes associated with the botnet agent have relatively limited calculating capability and network bandwidth. The real-time detection method may produce a relatively high false alarm rate because it may misclassify a legitimate web server as a malicious domain name.

2. Related Work

In our previous research [9], we have proposed an effective botnet detection method based on fuzzy association rules (FARR). This method can calculate the features of botnet traffic accurately, which can be used to recognize the normal traffic and botnet traffic, while the false alarm rate is relatively high.

Perdisciet et al [10] suggested that we should adopt the real-time tracking method (including queries) of DNS traffic, collect DNS responses by inserting monitors at some key positions in the ISP network, and analyze the traffic to facilitate searching for the coverage area of the botnet. The C4.5 decision tree classifier is used to classify the domain names. Different from the active detection method, the advantage of this method is that it does not create extra load on network resources to form active DNS queries and requests. It also makes it impossible for botnet attackers to detect these traces. However, such systems have a high false rate relatively. In addition, the detection delay of this technique is longer than any other task.

Tyagi and Aghila [11] proposed an analysis-based detection technique (ABDT) specifically for detecting botnets using a geographically dispersed set of proxy hosts with FFSN. HT Wang et al. [12] proposed a method for identifying botnets in real time by using a Local Spatial Geolocation Detection (LSGD) system, while also using Autonomous System Numbers (ASNs) to enhance localized geographic features. Huang et al. [13] proposed a Spatial Snapshot Fast-Flux Detection (SSFD) system based on two new spatial measuring methods: spatial distribution estimation and spatial service relationship evaluation. This system can capture the geographic location of the host, and the IP addresses from the response to the DNS response are mapping to the geographic coordinate system to detect the fast-flux botnets in real time and mitigate the harm caused by it.

Kang et al. [14] proposed a method of passive P2P monitor (PPM) which can identify the infected host's firewall or NAT. This method is derived from the authors of the study after the Storm Botnet, that is, the probability of establishing the coverage model (probability-based coverage model). The authors also utilized a verification tool (Firewall Checker, FWC) to verify the result of the identification. Research results suggest that 40% of the infected hosts are being used after a firewall or a NAT.

Saad et al. [15, 16] adopted the method of feature extraction of network traffic to detect P2P botnets, and this paper presents a dozen of feature values of network traffic, including the length of load average packet, the number of packets switching, and packet averaging intervals. Then they used machine learning methods to build a classifier to detect P2P botnets.

Wang et al. [17] proposed a fuzzy recognition algorithm to detect botnets. The paper points out that they regularly have DNS traffic and TCP traffic, and there are three steps to detect botnets: first, reducing the traffic to improve the detection efficiency; second, distributing the data packet whose feature is regarding the total number of DNS queries and the number of failures of DNS queries as the feature of DNS traffic. Meanwhile, we use TCP queries and response time distribution, the total number of TCP queries, and TCP data stream size distribution as TCP traffic's feature. Finally, we utilize the fuzzy recognition algorithm to detect domain names and IP addresses associated with botnets, thereby detecting botnets.

To solve the above problems, we propose a hybrid association rule algorithm to detect and classify the botnets. It includes global associations and fuzzy associations, and it also adds the detection of fast-flux botnets. Global associations can detect whether the data are a botnet, and fuzzy associations use global associations to determine what type of botnet it is. The results show that we can detect botnets quite well, to classify the botnets well.

3. Our Approach

According to the characteristics of traffic in a high-speed network environment, we propose a suspicious traffic filtering method based on real-time characteristics to reduce the total traffic that the system needs to process, the resources' consumptions, and improve the system performance. Taking the limitations of time-based detection methods currently into consideration, a hybrid association botnets detection method is derived according to the global association features extracted from the idea of bipartite graphs and combining local time features with fuzzy recognition. It includes global associations and fuzzy associations, which is shown clearly in Figure 1.

In global association, we analyze the global association between domain names and IP addresses, and we apply XGBoost machine learning algorithm with high detection speed and accuracy to botnet's detection to improve the accuracy of detection further. Consequently, it also enriches the botnet's dimension of the feature vector, breaks the limitations of the current research methods, improves detection efficiency and accuracy, and reduces detection false alarms and the rate of false alarms.

In fuzzy recognition, we have to divide the extracted features in a strict way. More specifically, different levels represent different degrees, the closer the botnet, the closer the level of optimization features to the botnet. According to the fuzzy algorithm principle of the maximum degree of membership, we can determine the attributes of the dataset. We also propose botnet association rules based on support and confidence formulae, and they can be used for mining association rules between botnets' features, which help us to determine the type of botnets and recognize normal and abnormal data.

3.1. Data Type. We set up botnet environment and collect data through public datasets. 36 normal datasets, 33 botnets datasets, 3 public botnets datasets [18], and 13 public botnets datasets [19] are collected together and shown in Table 1. We collected data and published three datasets for doing traffic analysis, and these abnormal datasets contain IRC, HTTP, P2P, fast-flux, and other botnets.

Besides, we utilize the real blacklist to access the traffic generated and the ISOT botnets' data collection and recombination and then utilize the TCPReplay tool to highly simulate the high-speed environment and replay the combined data stream packets. The high-speed network environment used in the test utilizes TCPReplay tool to simulate a 1 Gbps network and a 10 Gbps network, respectively. Table 2 shows the sources for collecting blacklists.

At this stage, the malicious domain names were collected for up to 48 hours. To increase the diversity of data, the top 500 popular domain names of Alexa [20] were selected for collection, and a 2.32 GB data stream package was chosen.

3.2. Traffic Filtering. The real-time suspicious traffic filtering method combines the advantages of black/white list and general real-time feature filtering of botnets to enhance the real-time and relative accuracy of filtering. In a complex network environment, the real-time detection of suspicious botnet domain names can further reduce and clean up complex DNS traffic, to provide an effective DNS data stream for the subsequent accurate detection, improve the system's speed of filtering DNS traffic, and reduce the overhead of system resources. Table 1 indicates the processing flow of the suspicious traffic filtering method based on real-time characteristics.

The real-time filtering methods for DNS traffic are here mentioned as follows:

(1) *Protocol Filtering.* The DNS parsing service uses port 53 for data transmission. Therefore, the first step is to use port 53 and the DNS packet header to filter DNS traffic.

(2) *Black and White Lists' Filtering.* The DNS traffic generated by most users on the Internet is harmless. A whitelist-based filtering method can filter a large amount of benign DNS access data, to reduce the data that the algorithm will use quickly and in real-time. Speaking of a specific fast-flux botnet real-time detection method that cannot distinguish the defects of the CND network and the fast-flux network, we use the first 100,000 domain names of Alexa, which can filter most CDN networks. The blacklist can directly filter malicious domain names, then alert the user and store it in the blacklist database, which provides technical support for mixed association botnet detection methods.

(3) *The Real-Time Feature Filtering of Botnets.* In real-time detection of botnets, feature vectors relatively are used to improve the real-time performance of the detection algorithm. However, due to the similarity between the CDN network and the fast-flux botnet, the real-time detection method has a high false positive rate and false negative rate. This article summarizes some general characteristics of real-time detection based botnets, relaxes filtering rules, eliminates false alarms, and filters suspicious fast-flux traffic, to provide accurate and effective data for the following algorithms, which can improve the detection performance and effectiveness.

3.3. Feature Extraction. We divide the crawling traffic into UDP and TCP flow by following UDP and TCP protocols so that we can count and analyze each flow and packet of datasets. A large number of bots will send a control message to the controlled host. Therefore, when the controlled host accepts messages, it will send it as a response to the bots, where there will be a lot of problems of traffic functions, for instance, a packet being sent successfully, packet transmission time intervals, large amounts of data emanating from the same port, but not containing specific ports.

The method proposed by Wang et al. [17] is inactive botnets, and it changes DNS intervals by the impact of bots. Based on this work, we propose a new method to analyze the TCP protocol of PSH and UDP protocol by utilizing the

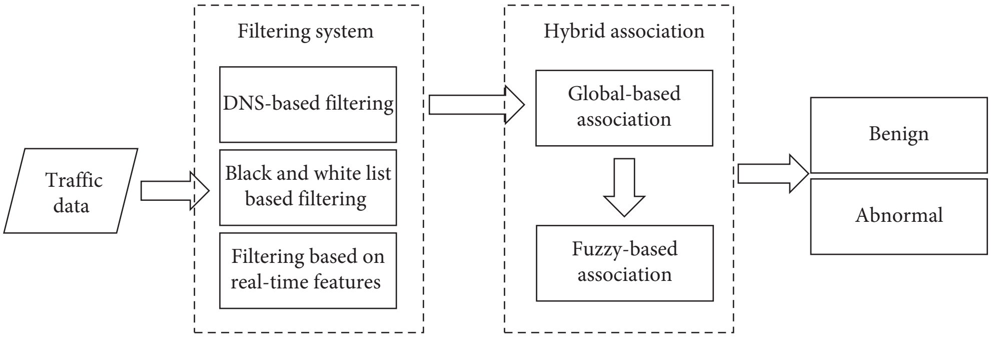

Figure 1: Botnet detection and classification process.

Table 1: Botnet type, number, and name.

Type	Amount	Botnet name
Normal	33	Normal (ISCX + ISOT)
IRC botnet	24	Neris, Rbot, Menti, Murlo, Tbot, IRC ISCX
HTTP botnets	7	Virut, Sogou
P2P botnets	12	NSIS.ay, SMTP Spam, Zeus (C & C), UDP Storm, Zeus, Zero access, Weasel
PS botnets	3	Zeus
Fast-flux botnets	3	Waledac

Table 2: Blacklist.

	Blacklist	Source
1	DNS Blackhole	http://www.malwaredomains.com
2	Spam	http://untroubled.org/spam
3	Phish	http://www.phishtank.com
4	Zeus malicious domain names	http://www.malwaredomainlist.com/forums/index.php
5	ZEUS tracker	https://zeustracker.abuse.ch/blocklist.php
6	Malicious domain names	http://www.abuse.ch/
7	Long-term malicious domain names	http://www.malwaredomains.com/wordpress/?p=1282

DNS response intervals. If there is no active botnet, there is no data needed to transmit, and PSH will be 0. The proportion of PSH in the dataset has changed, and DNS response will have a fixed interval. This will also affect the proportion of the TCP data's, and the source IP ratio will change greatly. When botnet's network node changes or controlled host strengthens defense, it will affect the success rate of data transmission. Some botnets transmit data through the C & C server, which relates to the existence of a specific port. Among the fixed TCP ports, the fixed UDP port, and interval DNS request, one is Boolean attribute, while another is a quantitative attributes.

In the TCP process, we need to analyze the following features, and Table 3 shows the related statistics.

We also select real-time function to further refine DNS traffic after the filtering of black/white lists. Paper [21] mentioned if any DNS A records TTL = 0, that domain will be marked as suspicious. If the TTL is not 0, we use the real-time characteristics in Table 4 to classify the domain into suspicious domain names or benign domain names. In each DNS response, both the A record and the NS record have a TTL field, which is used to specify the response retention time, or it means the effective intervals of the DNS cache. Although the RFC suggests calculating the minimum TTL in days, most legitimate high-availability websites use TTL values between 600 and 3600 seconds.

It is worth noting that in some fast-flux botnets, to change the IP address and IP of NS servers quickly, the attacker usually uses a TTL value of less than 300 seconds so that bots can connect to C & C hosts in time. In addition, to achieve the better load balancing and higher fault tolerance ability, the existing content distribution and Round-Robin DNS (RRDNS) networks usually have a smaller TTL value. Table 5 shows the TTL values of the types of network A's records.

Most benign Fully Qualified Domain Names (FQDNs) are mapped to even closer hosts and are part of the same ASN. Some fast-flux zombie hosts are geographically dispersed on the Internet randomly. Their characteristics belong to different autonomous systems. The A and NS records for domain names also occupy some more countries. Therefore, all IP addresses of a domain have the same ASN and country/region, which means that the domain is legal; otherwise, it may be suspicious. Table 6 lists and shows the number of ASN distributions and country distributions of benign and fast-flux domain names.

Table 3: Botnet features.

TCP protocol	PSH = 1 proportion the dataset TCP packet incoming and outgoing ratio ICMP success rate of sending Containing a specific TCP port, such as 6665,6667,8000,9000 Source IP proportion
UDP protocol	DNS request interval 90–110 s The same UDP port proportion of all ports The highest proportion of fixed UDP port

Table 4: Selection of global correlation features.

Category	Description
TTL value	A recorded survival time NS recorded survival time
The diversity of ASN	Diversity of ASN (autonomous domain number) of IP address in a record Diversity of ASN (autonomous domain number) of IP addresses in NS records
Number of IP addresses	Number of IP addresses in the a record Number of name server IP addresses in NS records

Table 5: TTL for fast-flux and high-availability networks.

Fast-flux botnets		High-performance benign network	
Domain name	TTL (s)	Domain name	TTL (s)
jaaphram.com	60	yahoo.com	1574
p-alpha.ooo.al	60	google.com	52
prtscrinsertcn.net	60	youtube.com	129
Entryrxshop.com	300	baidu.com	455
towardplian.com	120	163.com	444
gty5.ru	542	microsoft.com	3600
mp3for-you.com	60	huya.com	600

Table 6: ASN and country distribution number of fast-flux and normal domain names.

Fast-flux botnets			High-performance benign network		
Domain name	ASN	Country	Domain name	ASN	Country
leddamp.com	98	54	taobao.com	1	1
envoyee.com	112	31	renren.com	1	1
spampro.info	55	23	qq.com	1	1
leolati.com	102	30	baidu.com	2	1

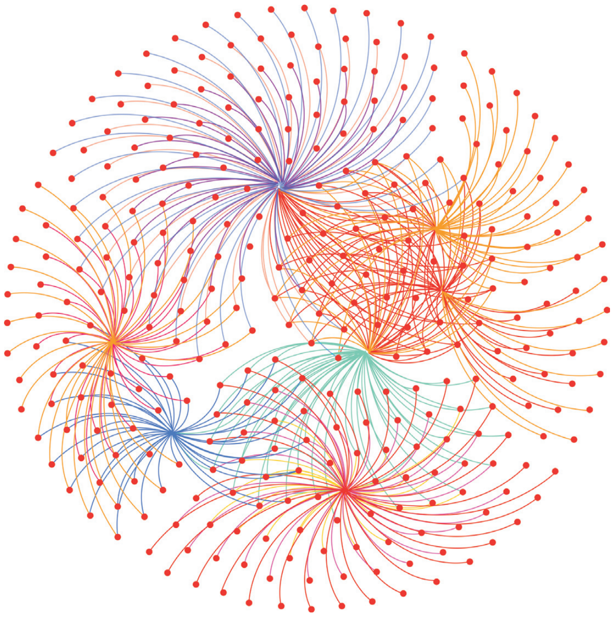

Figure 2: Association map of suspicious mapping of malicious domain names based on DNS Map.

In the fast-flux botnet, the IP address corresponding to the domain name is constantly changing. The global association mapping between the extracted domain name and the IP address is shown in Figure 2. The central points in the figure represent the domain name nodes, and the divergent nodes are represented as IP nodes. The global feature extraction is shown in Table 7.

3.4. Clustering Features and Dividing the Boundaries. Effective botnet's feature discretization is the key to the mining association rules; it is completely based on the method of part K to support the existence of inadequacies, especially when dealing with the difficulty in reflecting the actual distribution result from the high skewness of data effectively. There are excellent demarcation features on the division of the interval. Therefore, by using the FCM algorithm, we divide eight botnets features (including quantitative and Boolean attributes) into number of fuzzy sets; then, such fuzzy sets can convert between a set of elements and nonelements, to achieve softening the feature attributes of demarcation. When dealing with high skewness of data, FCM algorithm can effectively reflect the actual distribution of the data.

To classify the botnet features accurately, we must use various types of botnet datasets. In the FCM clustering of botnet matrix, 50 iterations, we divide it into five categories, and sizes of the center are divided into higher, high, medium, low, and lower, represented by numbers 5, 4, 3, 2, 1,

Table 7: Selection of global correlation features.

Category	Description
Number of nodes	Number of IP addresses Number of FQDN
Node degree	Maximum and average degrees of FQDN nodes
Betweenness	Betweenness FQDN node's largest IP node intermediary centrality

respectively. Based on the calculated botnets feature matrix and the center, the level of fuzzy sets can be determined by comparing the size of each of the centers. The largest center of fuzzy sets corresponding to the maximum level and the largest center of the corresponding elements of the matrix rows is the botnet ambiguity on fuzzy set maximum level. We list the feature PSH and ICMP's original datasets and classify those fuzzy sets. According to the different features of botnets classification, we can conclude the membership of each dataset's features, which means the features of botnets are different from each other. In other words, because of the different botnet's control nodes, the way of delivering messages, sending commands, and controlling the controlled host should be different.

We use fuzzy clustering to divide the quantitative features of botnets into five ranges, and we use the same quantitative feature values as the target dataset. In each interval in each class, the maximum and minimum values are taken as a maximum and a minimum range respectively. Thus, the quantitative attributes of Botnets could be divided into five ranges. In these fuzzy intervals, we can better respond to the actual distribution of botnet functions.

After analyzing all botnet datasets, we found that most botnets are at a lower level in PSH function. TCP packet incoming and outgoing ratio stay at two different points, one is at a high level while another is lower. We believe that different botnets have different proportions; IP source distribution is mainly in the higher level, and ICMP success rate is in a lower level, while the same UDP port functions are still in the lower level. DNS intervals, TCP, and UDP fixed port are distributed evenly.

With the basis of the conclusions acquired before, we divided the definition of this characteristic range dataset into five levels: higher, high, medium, low, and lower. These five levels represent the probability of Botnets as very high, high, medium, low, and very low, respectively. In Boolean features, we use different types of statistical methods to assess each level. Finally, we obtain a more accurate range of Botnet features, as it is shown in Table 8.

4. Association Rules for Botnet Recognition

In most botnets existing between features and feature necessary links, some are very closely linked while some are not, so our goal is to find features linked closely.

4.1. Botnet Fuzzy Association Rules. According to the level of the dataset to be divided, the numerical feature values will be converted into approximate Boolean feature values.

We set the botnet dataset feature dimension as p, class label dimension as q, then

$$\begin{aligned} X &= \left\{y_1, \ldots, y_p\right\}, \\ Y &= \left\{y_{p+1}, \ldots, y_{p+q}\right\}. \end{aligned} \quad (1)$$

Botnet datasets and class labels can be expressed as

$$S = \left\{s_j = (X, Y)_j = \left(y_1, \ldots, y_p; y_{p+1}, \ldots, y_{p+q}\right)_j = \left(y_{j1}, \ldots, y_{jp}; y_{j(p+1)}, \ldots, y_{j(p+q)}\right) \mid j = 1, \ldots, n\right\}. \quad (2)$$

Fuzzy sets of botnets features and class labels can be expressed as

$$\mathrm{MF} = \left\{A_m^k\left(y_{jm}\right) \mid j = 1, \ldots, n;\ m = 1, \ldots, p+q;\ k = 1, \ldots, k_{jm}\right\}. \quad (3)$$

Then,

$$F\mathrm{Sup}(X) = \sum_{j=1}^{n} \prod_{m=1}^{p} t_j\left(y_m\right) = \sum_{j=1}^{n} \prod_{m=1}^{p} \max_{k=1,\ldots,k_{jm}} \left\{A_m^k\left(y_{jm}\right)\right\}. \quad (4)$$

Fuzzy association rule "$X \Rightarrow Y$" fuzzy support is calculated by

$$F\mathrm{Sup} = \sum_{j=1}^{n} \prod_{m=1}^{p+q} t_j\left(y_m\right) = \sum_{j=1}^{n} \prod_{m=1}^{p+q} \max_{k=1,\ldots,k_{jm}} \left\{A_m^k\left(y_{jm}\right)\right\}. \quad (5)$$

Fuzzy association rule "$X \Rightarrow Y$" fuzzy confidence is calculated by

$$F\mathrm{conf} = \frac{F\mathrm{Sup}(X \cup Y)}{F\mathrm{Sup}(X)}. \quad (6)$$

Table 8: Botnet features of range.

Features	Higher	High	Medium	Low	Lower
PSH = 1	0–4.2%	18.5–32.7%	66.8–84.4%	94.3–100%	44.1–47.1%
TCP packet IN/OUT	122.8–160%	0–22.2%	25.1–58.2%	75.9–97.2%	218–230%
Source IP	18.5–36.4%	70.0–82.1%	0–8%	85.5–89.9%	57.5–59.4%
ICMP rate	0–8.5%	35.7–42.4%	27–25.2%	98.9–99.9%	92.9–97.2%
UDP port	0–9.7%	38.5–52.8%	66.7–71%	84.7–100%	22.3–36.3%

We define the minimum support as equal to 0.06 and minimum confidence equal to 0.2, which is a meaningful association rule. According to the different features of different memberships and formula, we calculate the botnet's association rules. Antecedents i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_9, i_{10} represent 10 botnet features; then, the i_{10} indicates the datasets' types: normal, Botnet, IRC, P2P, HTTP, Fast-Flux, Mix (IRC, P2P, HTTP, and PS), and part of fuzzy association rules. i_9 represents fast-flux botnet TTL < 300. i_{10} represents fast-flux botnet number of ASN distributions >2. As shown in Table 9, if X is A, then Y is B, $X = \{i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8\}$, $Y = \{i_{11}\}$, A represents the association rules, and B represents the properties of Y.

Table 9: Part meaningful association rules.

Rules	*F*Sup	*F*Conf
$i_6 = 1\hat{i}_7 = 1\hat{i}_8 = 1 = > = \mathrm{IRC}$	0.091	0.261
$i_6 = 1\hat{i}_7 = 1\hat{i}_8 = 0 = > i_{11} = \mathrm{IRC}$	0.091	0.200
$i_1 = 1\hat{i}_2 = 5\hat{i}_6 = 0\hat{i}_7 = 1 = > = \mathrm{IRC}$	0.091	0.261
$i_1 = 4\hat{i}_2 = 1\hat{c} = 1,\ i_7 = 1 = > = \mathrm{IRC}$	0.091	0.261
$i_6 = 0 = > = \mathrm{IRC}$	0.393	0.684
$i_7 = 1 = > = \mathrm{IRC}$	0.424	0.736
$i_8 = 1 = > = \mathrm{IRC}$	0.424	0.736
$i_6 = 0\hat{i}_7 = 0\hat{i}_8 = 0 = > = \mathrm{Normal}$	0.530	0.97
$i_1 = 2\hat{i}_2 = 5\hat{i}_6 = 0 = > = \mathrm{P2P}$	0.375	0.50
$i_1 = 2\hat{i}_2 = 1\hat{i}_5 = 1\hat{i}_6 = 1\hat{i}_8 = 0 = > i_{11} = \mathrm{Mix}$	0.109	0.42
$i_3 = 3\hat{i}_5 = 3\hat{i}_8 = 1 = > i_9 = \mathrm{HTTP}$	0.061	0.50
$i_3 = 5 = > i_{11} = \mathrm{botnet}$	0.472	0.566
$i_4 = 1 = > = \mathrm{botnet}$	0.303	0.606
$i_{11} = 1 = > = \mathrm{normal}$	0.470	0.939
$i_9 = 1\hat{i}_{10} = 1 = > = \mathrm{Fast\text{-}flux}$	0.236	0.710
$i_9 = 1\hat{i}_{10} = 0 = > = \ \mathrm{normal}$	0.391	0.452

Selecting meaningful association rules from the table of analysis, we can obtain flag i_6, i_7, i_8, which are used to determine the flag of botnets. Normal dataset features i_6, i_7, i_8 are all equal to 0, which is a high degree of confidence.

After calculation, we get *F*Sup $(i_7) = 0.72$, *F*Conf $(i_7) = 0.566$, which can be explained in the botnet. The same source IP distribution must be accounted for a large proportion of the whole IP, which explains the bots will always continue to send information to the target host.

Botnet featuresi_6, i_7, $i_8$for IRC botnet also have strong association rules. It can be used to identify IRC Botnets. In other features, we can find IRC Botnets being divided into two categories, the first is the lower the transmission effects are, the lower incoming packets outgoing ratio will be. The source IP distribution is obvious and there are DNS requests frequently, indicating that this type of IRC botnets has breakpoints in network nodes. In other words, controlled hosts add defensive measures to prevent bots' control. The second category is that the botnet has high efficiency of transmitting data. Transmission data packet also increased, indicating that this type of Botnets is very active.

When $i_3 = 3 \wedge i_5 = 3 \wedge i_8 = 1$, $i_9 = \mathrm{http}$ botnet and support = 6.1% and confidence = 50%. The proportion of http botnet explained in the same UDP port and the proportion of port number 0&161 have obvious characteristics. For identifying the http botnet, it has great help.

When $i_1 = 2 \wedge i_2 = 1 \wedge i_5 = 1 \wedge i_6 = 1 \wedge i_8 = 0, = i_9$mixed botnet. At this time, support = 11.5% and confidence = 42.8%.

It is noteworthy that when $i_4 = 1$, ICMP has a high success rate. It may also be associated with a strong botnet or a normal dataset.

4.2. Detection of Botnets Based on Hybrid Associations.

Hybrid association detection is a more accurate analysis and detection of filtered suspicious network traffic. First, we check whether there exists a botnet in the controlled network; then, the global feature correlation and local feature mixing methods are utilized to detect the botnet. For the lack of botnet detection, it greatly improves the detection accuracy and efficiency of botnets.

Based on the characteristics of global association and according to the idea of bipartite graph, we observe the global association relationship between the domain name and its mapped IP in a certain period of time. If the IP address hosting the malicious domain name (fast-flux domain name) hosts another unknown domain name, the unknown domain name may also be malicious. Because the fast-flux domain name will map a large number of IP addresses, and attackers can utilize more domain names to organize the fast-flux network. Using the association relationship, we can find the emerging and dying fast-flux domain names. We utilize the DNSMap tool to extract the global mapping relationship between domain names and IP addresses and then calculate the global correlation features, which can enrich the feature vector dimension of the fast-flux botnet.

According to the characteristics of botnets, local characteristics based on time are obtained by parsing DNS data packets and doing statistics on related data. Based on the real-time detection method, we analyze the DNS, and fast-flux domain names can be detected by obtaining 3-4 characteristics. However, the rapid development of existing CDN networks and RRDNS networks has shown the same trend as fast-flux networks' characteristics, so an amount of false alarms is generated during real-time detection. The time-based feature extraction method mainly counts

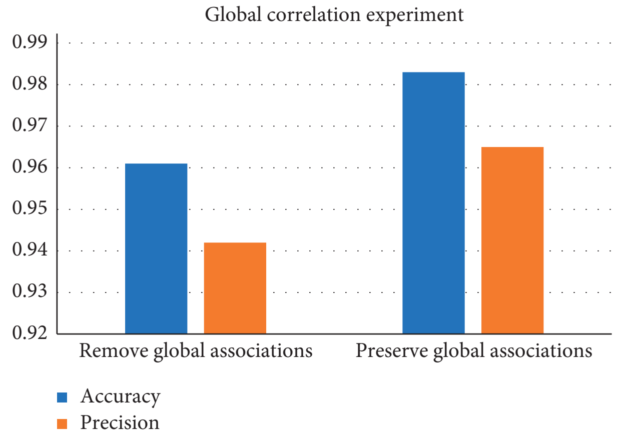

Figure 3: Accuracy and precision in the case of ten-fold cross-validation.

information such as the number and growth of IP addresses corresponding to botnet domain names over a period of time, together with the size and average value of TTL values corresponding to domain names. Therefore, although the time-based detection method uses time for detecting, it reduces the false alarm and false alarm rate of the system, which is acceptable in a large high-speed network environment.

During our experiments, we selected 7582 experimental data, which included 77 fast-flux botnet domain names and 7505 benign legal domain names. At the same time, the XGBoost learning algorithm is used for training the data and establishing the optimal classification model.

Then, this experiment verifies the extracted features by doing research on it, mainly to verify the validity of the features based on global correlation. We duplicate the extracted feature dataset into two duplicates, one of which removes the global associated features and only retains the time-based features, while another retains all features. Then we perform ten-fold cross-validation on the feature dataset by utilizing XGBoost machine learning algorithm. Figure 3 shows the accuracy and precision in the case of ten-fold cross-validation.

From Figure 3, we can find that after removing the global correlation features, the detection accuracy and precision have decreased in the case of using ten-fold cross-validation, which have dropped by 2.2% and 2.3%, respectively. At the same time, the number and the rate of false alarms have also decreased. It can be figured out that it is very helpful to improve the detection accuracy and efficiency based on global correlation features.

5. Method Comparison

According to our association rules, the test has already been done with the help of the public data [19]. The accuracy rate can reach 98.2%, and the comparison of various algorithms is shown in Table 10. TPR (true-positive rate) can be understood as how many of all positive classes in the result are predicted to be positive classes in our experiment (correct positive class prediction). FPR (false-positive rate) can be understood as how many of all negative classes in the result are predicted to be positive classes in our experiment (wrong positive class predictions). We can figure out that using our algorithm to identify the advantages of botnets receives great effects. Meanwhile, it can also mine in their rules deeply, especially for IRC and fast-flux botnets. The reason of it is that we are looking for functions similar to those of IRC and fast-flux botnets.

Table 10: Comparison of results.

	Public datasets scene 1, IRC botnet	
Method	TNR	FPR
BH	0.5	< 0.0
CA1	0.9	< 0.0
BCLus	0.5	0.4
FARR	0.9	0.1
Hybrid association	0.9	< 0.0
	Public datasets scene 2, IRC botnet	
Method	TNR	FPR
BH	0.99	< 0.0
CA1	0.9	< 0.0
BCLus	0.7	0.2
FARR	0.9	0.1
Hybrid association	0.9	< 0.0
	Public datasets scene 6, PS botnet	
Method	TNR	FPR
BH	0.99	< 0.0
CA1	0.9	< 0.0
BCLus	0.8	0.2
FARR	0.7	0.2
Hybrid association	0.8	0.1
	Public datasets scene 9, MIX botnet	
Method	TNR	FPR
BH	0.99	< 0.0
CA1	0.9	< 0.0
BCLus	0.6	0.3
FARR	0.8	0.1
Hybrid association	0.8	< 0.0
	Fast-flux datasets	
Method	TNR	FPR
BH	0.99	< 0.0
CA1	0.9	< 0.0
BCLus	0.6	0.3
FARR	0.8	0.1
Hybrid association	0.9	< 0.0

6. Conclusion

This paper proposes a new hybrid association method to detect and classify botnets. This method can well solve the boundary and classification problems of botnets and normal data. This new detection algorithm contains four steps. First, the collected traffic is filtered by using a suspicious DNS protocol, black/white lists, and real-time function monitoring, which can greatly reduce the detection overhead and improve the detection efficiency. The second is feature extraction for filtered traffic, and it ranges from time-related functions and domain name functions to basic traffic functions. The third is performing global correlation and using machine learning to identify botnets. Finally, we make the botnets fuzzy associated, determine the association rules by calculating the support and trust degree, and then classify those botnets after that. In botnets' classification, we use various functions between support and confidence degree to filter association rules. We can classify not only IRC and HTTP botnets, but new ones including P2P and fast-flux botnets.

Our next task is to study the double fast-flux botnet and detect it.

Data Availability

The normal and abnormal botnet traffic.pcap data used to support the findings of this study were supplied by Jiazhong Lu under license and so cannot be made freely available. Requests for access to these data should be made to Jiazhong Lu, ljz@cuit.edu.cn.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This paper was supported in part by the National Social Science Fund of China for supporting our project "Computational Communication Strategy for Chinese Technology Image on Twitter," no. 20CXW016, in part by the National Key R&D Program of China (no. 2017YFB0802300), in part by the Key Research and Development Project of Sichuan Province (nos. 20ZDYF0660, 2018GZ0204, 2018RZ0072, and 2021YFYZ0028), in part by the Foundation of Chengdu University of Information Technology under grant J201707 in part by the Project of Chengdu City (no. 2020RK00-00266-ZF), in part by the Director of Computer Application Research Institute Foundation (SJ2020A08), and China Academy of Engineering Physics Innovation and Development Fund Cultivation Project (PY20210160).

References

[1] T. Micro, *Taxonomy of Botnet Threats*, Trend Micro, Tokyo, Japan, 2006.

[2] E. Stinson and J. C. Mitchell, "Characterizing bots' remote control behavior," *Detection of Intrusions and Malware, and Vulnerability Assessment*, Springer-Verlag, in *Proceedings of the 4th International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, DIMVA' 07*, pp. 89–108, July 2007.

[3] L. Liu, S. Chen, G. Yan, and Z. Zhang, "BotTracer: execution-based bot-like malware detection," in *Proceedings of the 11th International Conference Information Security*, ISC, Taipei, China, September 2008.

[4] L. Liu, S. Chen, G. Yan, and Z. Zhang, "BotTracer: execution-based bot-like Malware detection," in *Proceedings of the 11th International Conference Information Security*, pp. 97–113, Taipei, Taiwan, China, September 2008.

[5] M. Szymczyk, "Detecting botnets in computer networks using multiagent technology," in *Proceedings of the Fourth International Conference on Dependability of Computer Systems*, pp. 192–201.

[6] K. Xu, D. Yao, Q. Ma, and A. Crowell, "Detecting infection onset with behavior-based policies," in *Proceedings of the5th International Conference on Network and System Security (NSS)*, pp. 57–64, ember.

[7] Y. Zeng, X. Hu, and K. G. Shin, "Detection of botnets using combined host-and network-level information," in *Proceedings of the 2010 IEEE/IFIP International Conference on Dependable Systems and Networks*, pp. 291–300.

[8] C.-H. Hsu, C.-Y. Huang, and K.-T. Chen, "Fast-flux bot detection in real time," in *Proceedings of the International Conference on Recent Advances in Intrusion Detection*, pp. 464–483, Springer-Verlag, Ottawa, Canada, September2010.

[9] J. Lu, F. Lv, Q.-H. Liu, M. Zhang, and X. Zhang, "Botnet detection based on fuzzy association rules," in *Proceedings of the 2018 24th International Conference on Pattern Recognition (ICPR)*, pp. 578–584, Beijing China, August 2018.

[10] R. Perdisci, I. Corona, D. Dagon, and W. Lee, "Passive analysis of recursive DNS traces," in *Proceedings of the Computer Security Applications Conference*, pp. 311–320, IEEE Computer Society, Honolulu, HI, USA, December 2009.

[11] A. K. Tyagi and G. Aghila, "Detection of fast flux network based social bot using analysis based techniques," in *Proceedings of the International Conference on Data Science & Engineering*, pp. 23–26, IEEE, Cochin, India, 18-20 July 2012.

[12] H. T. Wang, C. H. Mao, K. P. Wu, and H. M. Lee, "Real-time fast-flux identification via localized spatial geolocation detection," in *Proceedings of the Computer Software and Applications Conference*, pp. 244–252, IEEE, Izmir, Turkey, July 2012.

[13] S. Y. Huang, C. H. Mao, and H. M. Lee, "Fast-flux service network detection based on spatial snapshot mechanism for delay-free detection," in *Proceedings of the 5th International Symposium on ACM symposium on Information, Computer and Communications Security*, pp. 101–111, Beijing, China, January 2010.

[14] J. Kang, Y. Z. Song, and J. Y. Zhang, "Accurate detection of peer-to-peer botnet using Multi-Stream Fused scheme," *Journal of Networks*, vol. 6, no. 5, pp. 807–814, 2011.

[15] S. Saad, I. Traore, A. Ghorbani et al., "Detecting P2P botnets through network behavior analysis and machine learning," in *Proceedings of the 2011 9th IEEE Annual International Conference on Privacy, Security and Trust (PST 2011)*, pp. 174–180.

[16] D. Zhao, I. Traore, A. Ghorbani, B. Sayed, S. Saad, and W. Lu, "Peer to peer botnet detection based on flow intervals," in *Proceedings of the 27th IFIP TC 11 Information Security and Privacy Conference (SEC 2012)*, pp. 87–102, Heraklion, Greece, June 2012.

[17] K. C. Wang, C. Y. Huang, S. J. Lin, and Y. D. Lin, "A fuzzy pattern-based filtering algorithm for botnet detection," *Computer Networks*, vol. 55, pp. 3275–3286, 2011.

[18] Information Security Centre of Excellence, "UNB ISCX botnet DataSet [EB/OL]," 2014, http://www.unb.ca/research/iscx/dataset/ISCX-botnet-dataset.html.

[19] S. García, M. Grill, J. Stiborek, and A. Zunino, "Anempirical comparison of botnet detection methods[J]," *Computers & Security*, vol. 45, pp. 100–123, 2014.

[20] Amazon company, "Alexa top 500 global sites[EB/OL]," 2020, https://www.alexa.com/topsites.

[21] H. T. Lin, Y. Y. Lin, and J. W. Chiang, "Genetic-based real-time fast-flux service networks detection," *Computer Networks*, vol. 57, no. 2, pp. 501–513, 2013.