Network Node Classification Based on DNS Query Analysis

Kodai Watanabe[1], **Akira Sato**[1], **Shuji Sannomiya**[1,2], **and Yasuhi Shinjo**[1]
[1]University of Tsukuba, 1-1-1 Tennodai, Tsukuba, Ibaraki, Japan
[2]DDSNA, Inc., 1-1-1 Tennodai, Tsukuba, Ibaraki, Japan

Abstract— *The Domain Name System (DNS) is an essential component of the Internet; hence numerous researchers are working on data mining from DNS queries and replies. A local area network (LAN) includes not only normal clients and servers but also abnormal nodes, such as malware infected nodes and incorrectly configured nodes. Network administrators want to classify normal nodes and abnormal nodes. This paper proposes a classification method of network nodes though DNS query analysis with machine learning. This method does not use fully qualified domain names (FQDNs) but their hash values for protecting privacy. The proposed method uses conventional DNS record types and novel parameters referred to as pop rate and diff rate in the MeanShift algorithm. Experimental results show that the proposed method can classify the nodes in a LAN into clients and servers.*

Keywords: DNS, clustering, data mining.

1. Introduction

Monitoring the behavior of network nodes is important for detecting abnormal situations such as failures and cyberattacks. Efficiency and accuracy improvement of this detecting are important issues in the operation of large scale network systems including IoT (Internet of Things) networks. Such monitoring should have no side effect on the nodes to achieve safe and smooth communication. We recognize that a typical network node in a local area network (LAN) plays the role of a server or a client. The security requirements of servers are different from those of clients. Moreover, the traffic volume of a server is different from that of a client. LAN administrators often desire to know the numbers of servers and clients in their LANs. Based on these facts, it is important to classify network nodes into servers and clients to achieve such security management and effective routing.

The DNS is an essential component of the Internet. The DNS is used for name resolution, which translates a fully qualified domain name (FQDN) into an IP address, and for various purposes such as distributing the public keys of Secure Shell (SSH) servers. However, the DNS is misused through cyberattacks. For instance, malware-infected nodes are often controlled by embedding malicious commands in the TXT record of a DNS response [3]. It has been recently reported that the DNS traffic becomes 120000 queries per second and increases by approximately 8% every month because of such usage [8]. Numerous researchers are working on DNS queries and replies.

In this paper, we propose a network node classification method using DNS queries. We use machine learning and classify network nodes into normal nodes and abnormal nodes/clients and servers.

We have selected the MeanShift algorithm because this algorithm can determine the number of clusters automatically [6]. The DNS queries sent from the nodes in our campus LAN to a full-service resolver are provided to the algorithm. To preserve the privacy of users, we anonymize domain names by hashing them. We use the number of queries, the number of DNS record types, the frequency of the queries, and the difference in the FQDN as the parameters of learning. We normalize the values of these parameters in terms of percentage (from 0 to 100) and feed them to the learning algorithm. Results show that we are able to obtain the clusters of servers and clients.

The rest of this paper is organized as follows: Section 2 describes the related work that analyzes normal and anomalous DNS traffic. Section 3 presents the learning and classification based on the MeanShift algorithm. Section 4 describes the evaluation of the classification result. Finally, Section 5 provides the conclusion of this work.

2. Related Work

This section describes the related works on the analysis of normal or anomalous traffic. DNS traffic has already been analyzed from various viewpoints. In [5], the A record queries sent by DNS clients to the DNS server in the network of a university are considered and their contents are investigated. Results shows that (1) the terminals infected with mass-mailing worm send A record type queries more frequently compared to normal terminals and that (2) the FQDN in the A record queries have several features such as including a few keywords in the FQDN. In addition, the usage of TXT record queries is analyzed in [3] because it has been recently reported that the TXT record has been used for botnet communication. In these works, only specific records and the contents of FQDN and response are considerd. In contrast, all record types and hashed FQDNs are investigated in the present study.

The authors presented a characterization of DNS clients in [10]. They developed a method to gather queries into clusters analyzed the patterns in the hostnames looked up

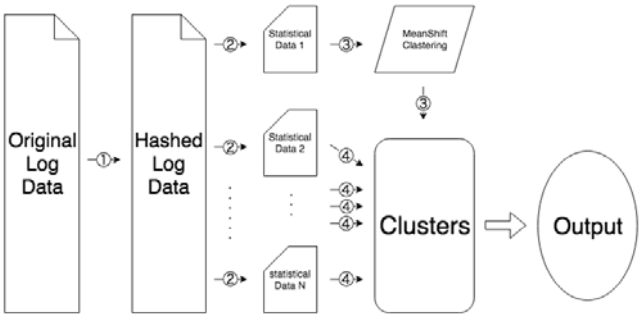

Fig. 1: Clustering flow.

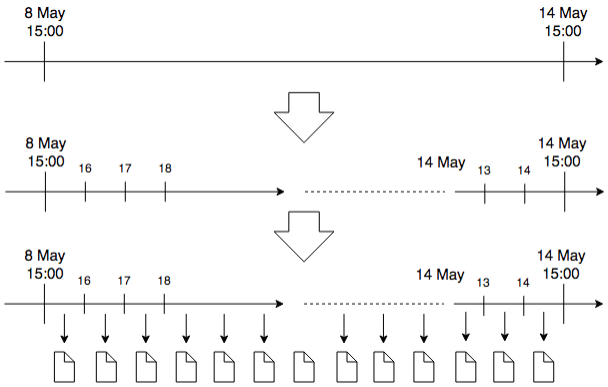

Fig. 2: Converting log data into statistical data, and window size. (step 2 of Fig. 1)

by individual clients. However, the purpose of this work was to identify clients' usages such as browsing and searching. Moreover, raw FQDN's were analyzed; this is problematic from the viewpoint of the privacy of network users. In contrast, the behavior of DNS clients for identifying a node's role is investigated with hashed FQDN in our work.

3. DNS Query Clustering

This section describes the procedure of the DNS query clustering based on the MeanShift algorithm. The procedure is composed of the following four steps:

1) Hashing an FQDN
2) Converting log data into statistical data for each DNS clients
3) Classifying statistical data for one hour using the MeanShift algorithm
4) Classifying other statistical data into the closest cluster created in the third step

These steps are illustrated in Fig. 1 where each encircled number represents the corresponding step, and they are detailed in the following sections.

3.1 Used Log Data of Queries to DNS Resolver

The log data of DNS queries in University of Tsukuba are used in this study. In the network of University of Tsukuba, some nodes are connected directly to the network while other nodes are indirectly connected through NAT servers. To discover the characteristics of each DNS client query, the log data of the queries sent from these DNS clients to a DNS full-service resolver in the network are considerd. The logs for one week from 15:00 on May 8, 2017 in Japan Standard Time are extracted from the log data stored in the DNS full-service resolver.

In the extracted log data, there are 29,209,880 queries from 12,141 DNS clients. In addition, responses, recursive queries, and recursive responses are recorded. We want to classify each DNS client's behavior, and we need to analyze active behavior. For this reason, we use only the data of the queries. To anonymize the extracted log data, all FQDNs are hashed so that the sites accessed by the DNS clients cannot be identified.

3.2 Converting Log Data into Statistical Data (Step 2)

The log data are continuously recorded. These data are temporally divided based on window size to discover the discriminative trends of queries in a certain period. A short window size may erase fruitful data while a long window size may make the early detection of anomalous terminals difficult. Window size is set as one hour in this study, as shown in Fig. 2.

3.3 Parameter

The parameters of machine learning are important because they may change results entirely. In this study, the following three statistical data are used as parameters:

1) All record types and the number of queries
2) Pop rate
3) Diff rate

3.3.1 All Record Type and the Number of Queries

A DNS query has several record types, and we use all record types as parameters. Certain record types are used frequently, while others are rare. To capture frequency of usage, the number of each record type is normalized in terms of percentage. For example, when a DNS client looks up five queries that consist of three A record queries, one AAAA record query, and one PTR record query, frequencies are calculated as $60\%(=3/5)$, $20\%(=1/5)$, and $20\%(=1/5)$, respectively. In this study, A, AAAA, PTR, TXT, SRV, DS, MX, SOA, NS, DNSKEY, SPF, CNAME, ANY, and SSHFP records are used for training data.

3.3.2 Pop Rate

This section provides the definition of pop rate. Fig. 3 shows how pop rate is calculated for each client. First, queries are devided into sections for every one minute. Second, the number of sections that have more than one

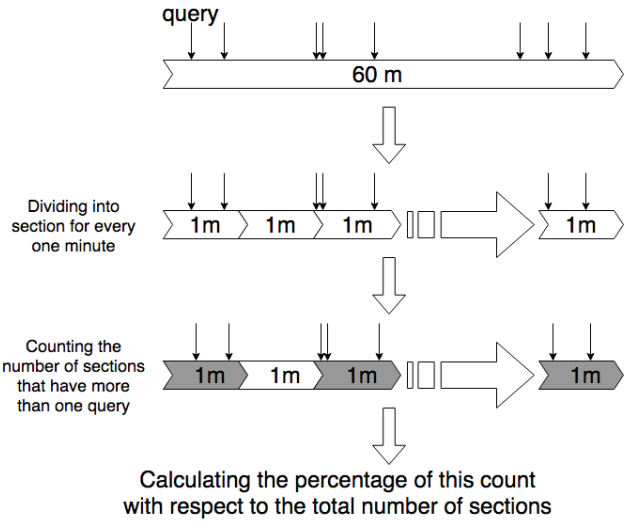

Fig. 3: Definition of pop rate

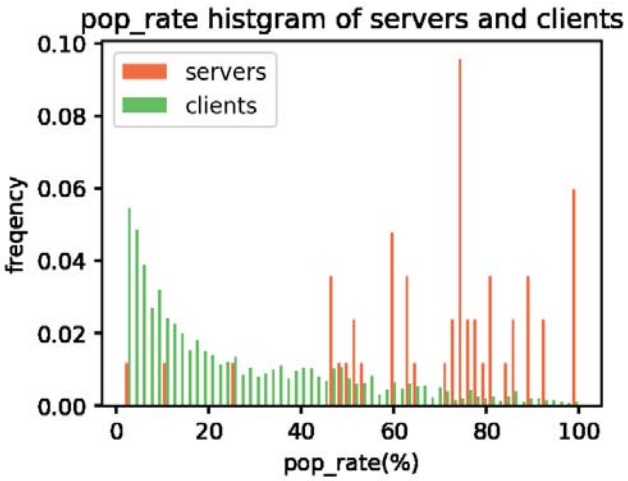

Fig. 4: Histogram comparing pop rates of servers and clients. (for one hour starting 15:00 on May 8, 2017)

query are counted. Finally, the percentage of this counted number is calculated with respect to the total number of sections. In Fig. 3, three sections have one than one query, and thus, pop rate is calculated as $5\%(= 3/60)$.

As pop rate will be higher when a network node communicates constantly, it is assumed that the pop rates of servers tend to be higher than those of clients. Fig. 4 shows the histogram of the pop rates of servers and clients from 15:00 on May 8, 2017 to 16:00 on May 8, 2017. The pop rates of most servers are larger, as shown in the Fig. 4. The labeling of nodes as servers or clients is explained in Section 3.4.

3.3.3 Diff Rate

Diff rate is defined as the different numbers of FQDNs. The calculation of the diff rate for each client is provided

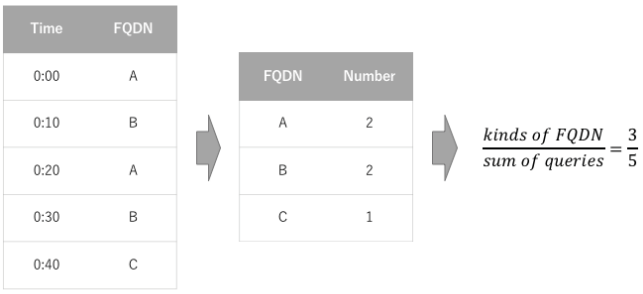

Fig. 5: Calculation example of diff rate

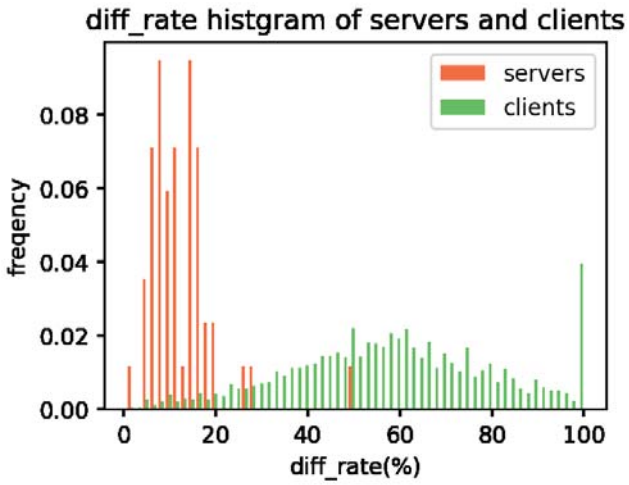

Fig. 6: Histogram comparing diff rates of servers and clients. (for one hour starting 15:00 on May 8, 2017)

below. Fig. 5 illustrates a diff rate calculation example.

$$\text{Diff\_Rate}(\%) = \frac{\text{Kinds of FQDN}}{\text{Sum of Query}} \times 100$$

Diff rate is small when a node looks up the same FQDNs. Based on this fact, it is expected that the diff rates of servers are small because as SSH server or a mail server is used by a specific person. Fig. 6 shows the histogram of the diff rates of servers and clients from 15:00 on May 8, 2017 to 16:00 on May 8, 2017. It is observed that the diff rates of most servers are less than those of clients. The labeling of servers and clients is explained in Section 3.4.

3.4 Clustering Based on MeanShift Algorithm and Labeling (Step 3)

First, DNS clients are classified using the MeanShift algorithm, whose training data are the parameters extracted from the log data for one hour. We select the MeanShift algorithm for clustering because it can automatically determine

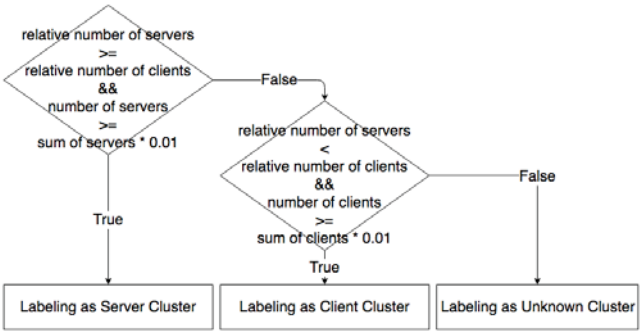

Fig. 7: Labeling flowchart

the number of clusters. Additionally, scikit-learn provides a flowchart that helps in selecting an appropriate machine learning algorithm [6], and the flowchart indicates that the MeanShift algorithm should be used.

Knowledge about the network is used to label the clusters derived from the MeanShift algorithm. In the campus network, certain IP address ranges are used only by wireless nodes that are almost clients and other IP address ranges are used as servers. Based on these facts, each cluster is labeled by investigating the IP addresses of the queries in the cluster. Fig. 7 shows the flowchart used to determine the label for each cluster. According to this flowchart, each cluster is labeled as a client, server, or unknown. Clustering requires a certain volume of training data, and we use the log data for one hour starting from 15:00 on May 8, 2017 because it is expected that a relatively large number of students (that is, the DNS clients) are using the network in this period.

3.5 Classification of DNS Queries (Step 4)

The data for other times is classified using the created cluster as test data. Then, we evaluate the result. Two groups of DNS queries are prepared. The first group is extracted from the log data of every one hour from 16:00 on May 8, 2017 to 15:00 on May 9, 2017. In other words, this group contains the training data and twenty three test data to check whether the classification depends on the time of the day. The second group is extracted from the log data of every one hour starting from 15:00 on every day from May 9, 2017 to May 14, 2017. That is, this group contains the training data and six test data at the same time of the day to check whether the classification depends on the day of the week.

4. Evaluation of Clustering and Labeling

This section describes the evaluation of the classification explained in the previous section.

4.1 Classification of Training Data

Table 1 shows the result of the classification, and it contains the identification number (ID), label, and the number of nodes for each cluster. According to the knowledge of the campus network, the number of server nodes and client nodes is 49 and 2407, respectively. Clusters No. 0, 1, 2, 3, and 4 are labeled as servers. Most servers belong to cluster No. 0 while a few client and unknown nodes also do. The remaining servers belong to the clusters labeled as No. 1, 2, 3, and 4, and a few client and unknown nodes also do. Even though it appears that cluster No. 2, 3, and 4 contain more clients than servers, the percentage of servers is more than that of clients, i.e. $0.02(\fallingdotseq 1/49)$ server and $0.005(\fallingdotseq 13/2407)$ client nodes exist in cluster No. 3. Cluster No. 5, 6, and 7 are labeled as clients. Most clients belong to cluster No. 5, and no server belongs to these clusters. Thus, these clusters are certainly considered as client clusters. The clusters labeled as unknown contain a few clients or unknown nodes.

Table 1: Classification result based on training data.

ID	Cluster Label	Server Nodes (51)	Client Nodes (2444)	Unknown Nodes
0	Server	42	11	69
1	Server	4	0	34
2	Server	2	7	36
3	Server	1	13	36
4	Server	2	37	159
5	Client	0	2228	2936
6	Client	0	45	165
7	Client	0	83	259
8	Unknown	0	4	25
9	Unknown	0	2	22
10	Unknown	0	6	7
11	Unknown	0	1	2
12	Unknown	0	0	2
13	Unknown	0	2	0
14	Unknown	0	0	1
15	Unknown	0	0	1
16	Unknown	0	0	1
17	Unknown	0	5	3
18	Unknown	0	0	1
19	Unknown	0	0	1
20	Unknown	0	0	1
Ratio(%)		100.0	96.4	

The ratio provided in the last row is the percentage of the number of nodes clustered appropriately with respect to the total number of nodes. In other words, the ratio of servers is true positive rate, and the ratio of clients is true negative rate. Because both ratios are high, it was seen that servers and clients have their own characteristics and training data can be classified using these characteristics. As the nodes in client clusters may contain abnormal nodes, true negative rate is only an indication of how correctly it is classified. The following discussion mainly focuses on true positive rate.

4.2 Characteristic Features of Each Cluster

Fig. 8 shows the average value of each parameter of each cluster as stacked bar graph.

Server clusters (No. 0, 1, and 2) have a high rate of PTR lookups. However, cluster No. 8 also exhibits a high rate, the

difference is that the nodes of this cluster look up several kinds of FQDNs. Normal clients do not send a large number of PTR records; however, it can be said that such clients exist. Cluster No. 3 and 4 consist of numerous A or AAAA records, and the nodes in the cluster search for the same FQDN in a short time. Further investigation of the servers in cluster No. 3 shows that a node that manages servers that are rented generates traffic for server management, such as security patch distribution. Thus, the node behaves like a client rather than a server. It is possible that the actual operation of the servers is different from the typical server operation.

The nodes in the client cluster No. 5 and 7 look up almost only A records. The pop rate of these clusters is approximately 50%, while their diff rates are different. According to this fact, it can be said that these two clusters are divided based on this difference. Cluster No. 6 cluster contains numerous A and AAAA records, and its diff rate and the pop rate are low.

In addition, there are clusters with characteristics different from those of the above mentioned clusters. Cluster No. 10, 12, and 19 contain multiple TXT record queries. In particular, in cluster No. 12 and 19, most queries consist TXT records and there are a few unknown nodes. This fact indicates that the nodes in these clusters may have been infected by malware. Even though cluster No. 13 and 20 contain multiple SOA record type queries, we cannot state what the nodes are doing. The unknown nodes in cluster No. 14 send a large number of ANY records while the unknown nodes in cluster No. 18 send NS records, however we cannot determine what these nodes intend to do. The sum of the parameters of cluster No. 15 is extremely high while its diff rate is considerably low. Based on this fact, we can state that the nodes in cluster No. 15 search for the same FQDNs in large quantities. The node in cluster No. 16 send numerous MX and PTR queries; therefore, this node may be working only as a mail server. Only a few nodes belong to these clusters, however, they are quite distinctive. Even though the actual purposes of the nodes of these clusters are uncertain, these are anomalies. Thus, our method can detect such anomalies.

Consequently, most servers and clients appear to be labeled correctly, and the clusters for classifying anomalous nodes are created.

4.3 Transition of True Positive Rate and True Negative Rate

Fig. 9 shows the transition of true positive rate and true negative rate from 15:00 on May 8 to 15:00 on May 9, 2017 (the first group). Both values are always over 90% regardless of the time period. The discussion on the low values of true positive rate is presented in the following section. Fig. 10 shows the transition of true positive rate and true negative rate from 15:00 on May 8 to 15 o ' clock

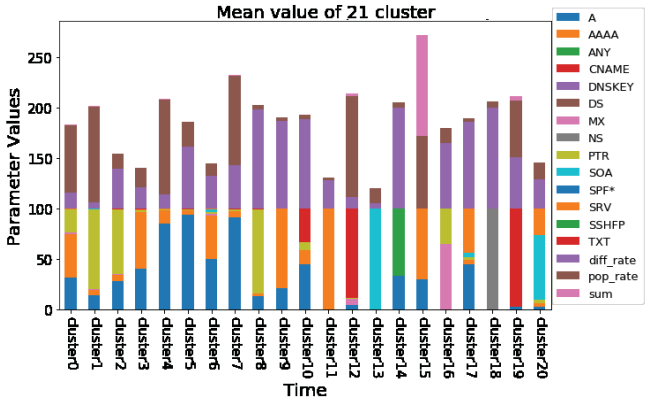

Fig. 8: Mean value of parameters of each cluster based on training data.

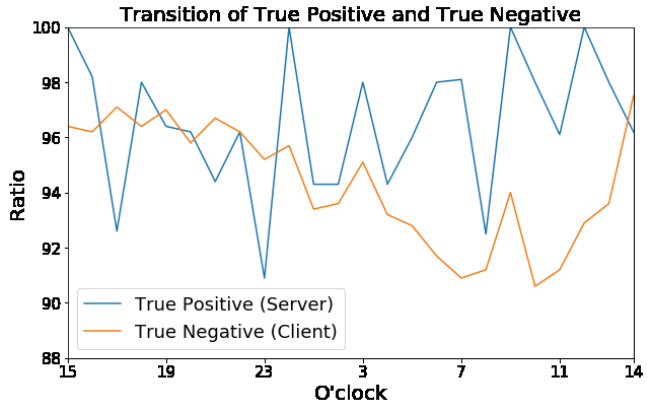

Fig. 9: Transition of true positive rate and true negative rate at each hour. (from 15:00 on May 8 to 15:00 on May 9, 2017)

May 15, 2017 (the second group). The minimum value of true positive rate is 98%. This indicates that clustering can be performed in the same time period. The discussion about true positive rate on a holiday is provided in the following section. True negative rate ais stable at 98%. Hence, clients can be identified regardless of weekdays and holidays.

4.4 Details on the Other Time

First, we discuss the cluster that has the highest true positive rate. Table 2 shows the classification result of the period starting from 0:00 on May 9. As shown in the table, all servers are classified into cluster No. 0, 1, and 2. In comparison with the result shown in Table 1, the number of servers in cluster No. 0 is less; however, cluster No. 1 contains more servers. Because most nodes in cluster No. 0, 1, and 2 are servers, the labeling of these cluster is relatively correct.

Second, we discuss the clustering result that has the lowest true positive rate. Table 3 shows the classification result of the period starting from 23:00 on May 8. As shown in the

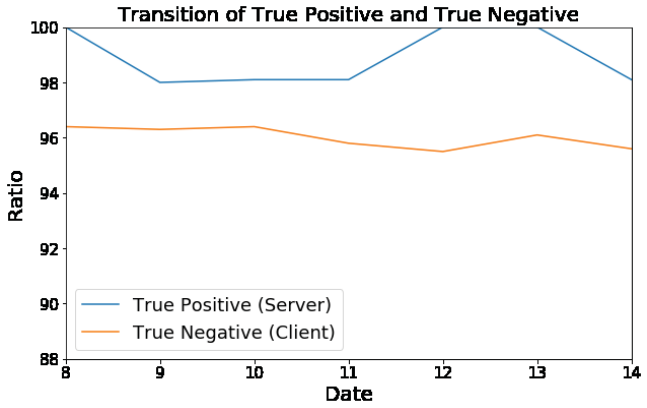

Fig. 10: Transition of true positive rate and true negative rate on each day. (from 15:00 on May 8 to 15:00 on May 15, 2017)

table, most servers are classified into cluster No. 0, 1, 2, and 3. The numbers of servers in these clusters are approximately the same as those shown in the Table 1. However, cluster No. 10, which is labeled as unknown, consists of four servers. The investigation of the parameter values of cluster No. 10 shows that there are a large number of TXT records in the cluster. Because there are cases in which servers send TXT record queries, this cluster may need to be labeled as a server. Moreover, a server node is classified into cluster No. 7 which is labeled as a client. Such nodes appear to work like a client without accepting any inquiries from their clients at only the period starting from 23:00 on May 8, e.g. its administrator conducts browsing, application execution, OS update, and so on. Except for these nodes, the classification appears to be correct.

Finally, we discuss the clustering result on a holiday. Table 4 shows the classification result for the period starting from 15:00 on May 14. Specifically, the training data were obtained on Monday (typically a weekday); however, it was a holiday in Japan. Most servers are classified into the No. 0, 1, 2, and 3 clusters. However, one server node is classified into cluster No. 7. This node is considered to be working as a client, similar to the servers discussed in the previous paragraph. As a whole, 98% of the servers are classified into server clusters, and this classification appears to be correct. Based on this fact, the classification can be performed regardless of the day of the week.

4.5 Discussion about Unknown nodes and Clusters

Unknown nodes classified into server or client clusters are considered to be in operation. However, there are a few unknown nodes classified into unknown clusters. We investigate these nodes. The characteristics of a few unknown clusters are quite different from those of other clusters.

Table 2: Classification result at 0:00 on May 9, 2017.

ID	Cluster Label	Server Nodes (54)	Client Nodes (2240)	Unknown Nodes
0	Server	34	16	73
1	Server	15	0	28
2	Server	5	7	56
3	Server	0	13	32
4	Server	0	39	173
5	Client	0	2036	2618
6	Client	0	55	153
7	Client	0	53	246
8	Unknown	0	6	18
9	Unknown	0	6	15
10	Unknown	0	3	11
11	Unknown	0	0	2
12	Unknown	0	0	2
13	Unknown	0	2	0
14	Unknown	0	0	0
15	Unknown	0	0	2
16	Unknown	0	0	2
17	Unknown	0	4	2
18	Unknown	0	0	1
19	Unknown	0	0	1
20	Unknown	0	0	0
Ratio(%)		100.0	95.7	

In other words, the nodes in these clusters are anomalous regardless of their maliciousness.

5. Conclusion

In this paper, we propose a network node classification method using DNS queries. We use machine learning and classify network nodes into normal nodes and abnormal nodes / clients and servers.

The DNS queries sent from the nodes in our campus LAN to a full-service resolver are provided to the MeanShift algorithm. To preserve the privacy of users, we anonymize domain names by hashing them. We use the number of queries, the ratio of DNS record types, pop_rate, and dif_rate as the parameters of learning. Pop_rate is the frequency of the queries, and diff_rate is the difference in the FQDN. Because pop_rate and diff_rate are characteristic parameters, they are important factor to understand the features of each node. Next, we classify training data based on the MeanShift algorithm. Each cluster is labeled as a server, client, or unknown. For this labeling, we use known data as server nodes and client nodes. We present a labeling flowchart. Finally, the data for other times are classified using the created clusters as test data. Test data contain the following two patterns: (1) Investigating whether results depends on time period (2) Investigating whether results depends on weekdays and holidays.

Each cluster of training data has characteristic features such as a high percentage of a specific record type. The Nodes belonging to these clusters are anomalies, and these nodes may lead to problems. True positive rate and true negative rate of clustering results are always over 90%

Table 3: Classification result at 23:00 on May 8, 2017.

ID	Cluster Label	Server Nodes (55)	Client Nodes (2677)	Unknown Nodes
0	Server	43	19	68
1	Server	4	0	22
2	Server	2	10	54
3	Server	1	21	36
4	Server	0	62	206
5	Client	0	2397	2821
6	Client	0	46	172
7	Client	1	105	306
8	Unknown	0	8	30
9	Unknown	0	2	26
10	Unknown	4	2	6
11	Unknown	0	0	2
12	Unknown	0	0	2
13	Unknown	0	2	0
14	Unknown	0	0	0
15	Unknown	0	0	1
16	Unknown	0	0	1
17	Unknown	0	3	4
18	Unknown	0	0	1
19	Unknown	0	0	2
20	Unknown	0	0	0
Ratio(%)		90.9	95.2	

Table 4: Classification result at 15:00 on May 14, 2017.

ID	Cluster Label	Server Nodes (52)	Client Nodes (2346)	Unknown Nodes
0	Server	40	19	64
1	Server	7	0	36
2	Server	3	8	27
3	Server	1	11	32
4	Server	0	49	166
5	Client	0	2123	2882
6	Client	0	35	148
7	Client	1	85	250
8	Unknown	0	3	27
9	Unknown	0	2	15
10	Unknown	0	6	10
11	Unknown	0	0	1
12	Unknown	0	0	2
13	Unknown	0	2	0
14	Unknown	0	0	0
15	Unknown	0	0	1
16	Unknown	0	0	1
17	Unknown	0	3	2
18	Unknown	0	0	0
19	Unknown	0	0	1
20	Unknown	0	0	0
Ratio(%)		98.1	95.6	

regardless time period in the first group. However, a few servers may be working as clients. The results of clustering are always approximately 98% regardless of weekdays and holidays in second group. Even though a few nodes may be running as servers, it can be said that they are almost correctly identified.

We will investigate the following challenges to investigate in future work: We must improve the labeling of certain clusters and label the clusters in smaller units. For this purpose, the nodes classified in incorrect clusters and anomalies of clustering must be examined.

References

[1] L. Bilge, E. Kirda, C. Kruegel, and M. Balduzzi, "EXPOSURE: Finding Malicious Domains Using Passive DNS Analysis," in *Conference: Proceedings of the Network and Distributed System Security Symposium, NDSS 2011, San Diego, California, USA, 6th February - 9th February 2011*, 01/01 2011.

[2] C. J. Dietrich, C. Rossow, F. C. Freiling, H. Bos, M. v. Steen, and N. Pohlmann, "On Botnets That Use DNS for Command and Control," *2011 Seventh European Conference on Computer Network Defense*, pp. 9–16, 2011.

[3] H. Ichise, Y. Jin, and K. Iida, "Analysis of via-resolver DNS TXT queries and detection possibility of botnet communications," *2015 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM)*, pp. 216–221, 2015.

[4] R. Moskovitch, Y. Elovici, and L. Rokach, "Detection of unknown computer worms based on behavioral classification of the host," *Computational Statistics & Data Analysis*, vol. 52, no. 9, pp. 4544–4566, 05/15 2008.

[5] Y. Musashi, R. Matsuba, and K. Sugitaniy, "Prevention of A-record based DNS Query Packets Distributed Denial-of- Service Attack by Protocol Anomaly Detection," *The Special Interest Group Technical Reports of IPSJ (IOT)*, vol. 2005, no. 83, pp. 23–28, 08/05 2005.

[6] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "2.3. Clustering scikit-learn 0.19.1 documentation."

[7] D. A. L. Romana, K. Sugitani, S. Kubota, and Y. Musashi, "DNS Based Detection of Spam Bots and Host Search Activity," *The Special Interest Group Technical Reports of IPSJ (IOT)*, vol. 2008, no. 87, pp. 1–6, 09/12 2008.

[8] Q. Xu, D. Migault, S. Sénécal, and S. Francfort, "K-means and Adaptive K-means Algorithms for Clustering DNS Traffic," in *Proceedings of the 5th International ICST Conference on Performance Evaluation Methodologies and Tools*, ser. VALUETOOLS '11. Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering; ICST, Brussels, Belgium, Belgium: ICST, 2011, pp. 281–290.

[9] T. Callahan, M. Allman, and M. Rabinovich, "On Modern DNS Behavior and Properties," *SIGCOMM Comput.Commun.Rev.*, vol. 43, no. 3, pp. 7–15, jul 2013.

[10] K. Schomp, M. Rabinovich, and M. Allman, "Towards a Model of DNS Client Behavior," in *Passive and Active Measurement*, T. Karagiannis and X. Dimitropoulos, Eds. Cham: Springer International Publishing, 2016, pp. 263–275, iD: 10.1007/978-3-319-30505-9_20.

[11] H. Zhang, M. Gharaibeh, S. Thanasoulas, and C. Papadopoulos, "BotDigger: Detecting DGA Bots in a Single Network," in *Proceedings of the IEEE International Workshop on Traffic Monitoring and Analaysis*. Louvain La Neuve, Belgium: IEEE, apr 2016, pp. 16–21.

[12] K. Ishibashi and K. Sato, "Classifying DNS Heavy User Traffic by Using Hierarchical Aggregate Entropy," in *2012 World Telecommunications Congress*, 2012, pp. 1–6.

[13] T.-S. Wang, H.-T. Lin, W.-T. Cheng, and C.-Y. Chen, "DBod: Clustering and detecting DGA-based botnets using DNS traffic analysis," *Computers & Security*, vol. 64, pp. 1–15, 01/01 2017.

[14] P. Mockapetris. DOMAIN NAMES - IMPLEMENTATION AND SPECIFICATION. https://www.ietf.org/rfc/rfc1035.txt. Network Working Group Request Request for Comments: 1035.