Identification of Communication Devices from Analysis of Traffic Patterns

Hiroki KAWAI†, Shingo ATA†, Nobuyuki NAKAMURA‡, Ikuo Oka†,
†Graduate School of Engineering, Osaka City University
3-3-138 Sugimoto, Sumiyoshi-ku, Osaka 558-8585, Japan
Email: {kawai_h@c.info., ata@ oka@}eng.osaka-cu.ac.jp
‡Oki Electric Industry Co., Ltd.
2-6-8 Bingomachi, Chuo-ku, Osaka 541-0051, Japan
Email: {nakamura758}@oki.com

***Abstract*—Recently, variety of communication devices such as printers, IP telephones, network cameras are used widely, with the support of networking in consumer electronics. As a spread of IoT (Internet of Things), the number of embed devices are significantly increasing, however, such devices have lack of capability on security. It is therefore desirable that a network identifies these devices to take appropriate operations. In this paper, we propose an identification method of communication devices from monitoring patterns of traffic, here we use statistical metrics such as packet inter-arrival time or packet size, and we apply a machine learning for the identification. Through evaluations using real traffic, we show that our method can achieve over 90% of identification to 9 commiunication devices.**

I. Introduction

In recent years, many communication devices such as printers, IP phones, network attached storage (NAS) are connected to the network. Also, the use of such devices will be diversified widely as the progress of development and deployment of IoT (Internet of Things) applications and services, which leads another concern on the device management. In order to operate these devices safely and properly, it is necessary to accumulate log information on the real-time state of devices, which is typically done with a kind of device manager, however, such managers are mostly vender-specific, i.e., there is no interoperability between different devices and managers. The objective of this paper is to realize a unified framework for the management of various types of devices, by the identification of devices based on the measurement of traffic patterns generated by devices. Identification of the type of device (we refer as *device identification* in this paper) would be much important for achieving adequate treatment for the target device. Furthermore, detection of anomaly behavior in M2M (Machine-to-Machine) communication is much attracted, in which communication devices have less capability to follow unexpected events, as well as security attacks, due to lack of computational resource. One possible solution is to detect an anomaly behavior through the monitoring of communication patterns (i.e., packets exchanged between devices), based on the fact that the communication pattern during anomaly behavior is significantly different from the one in normal condition. Device identification would be useful for the initial step of modeling traffic pattern in the normal behavior.

From above background, in recent years, demands for device identification (particularly for embedded devices, IoT devices, etc.) are increasing, and would be much important in future.

Researches on device identification is something similar on application identification techniques, which have been studied so far. Application identification based on flow statistics can identify applications accurately whatever the traffic is encrypted or not [1]–[3]. Recent studies have shown that application can be identified from the statistics of flows with around 80% of overall accuracy even if the application traffic is encrypted. Techniques of application identification is recently extended for the support of on-time identification [4], [5], application identification in mobile devices [6] and identification of user behavior in the same application [7].

In these methods, flow statistics (we refer traffic features in this paper) such as packet size, IAT (packet inter-arrival time), flow duration, total bytes and packets of flow are calculated. Identification is performed by using supervised ML (Machine Learning) based algorithms.

In this paper, we also use traffic features and ML algorithm for device identification, however, our method is completely different from other studies as follows. We use only two types of traffic features, i.e., packet size and IAT, while other methods use other types of traffic features to improve the accuracy of identification. Unlike application identification, communications between devices are continuous, and they may be initiated from both side. Furthermore, the volume of traffic is relatively small compared to user applications. As a result, traffic features such as a total number of packets/bytes in flow, duration of flow, or average transfer rate cannot be applied for device identification, because these features are dependent on the length of sessions. Instead, we use m-quantile ($m = 30$) values of both packet size and IAT distributions to improve the identification accuracy, where other methods use $m = 4$. As a result of evaluation by real traffic, we show that our method can achieve device identification (9 devices) with around 90% of accuracy by using 30-quantile values of packet size and IAT, after monitoring first 200 packets.

978-3-901882-98-2

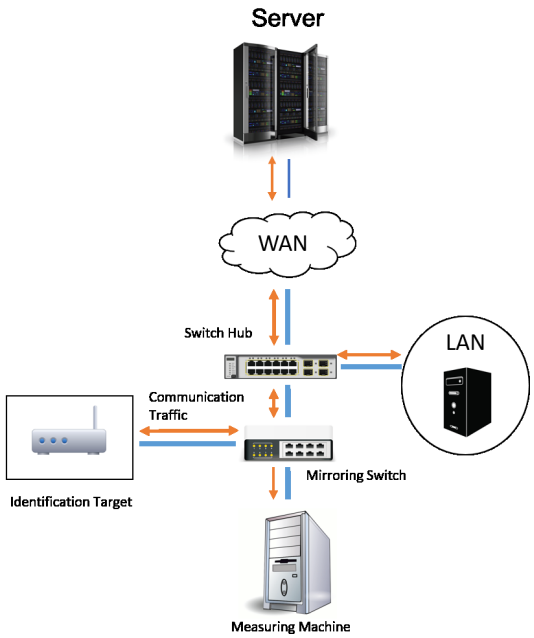

Fig. 1. Measurement environment

TABLE I
TARGET DEVICES FOR IDENTIFICATION

Device Type	Measurement Scenario
Wireless Access Point	Authentication with radius server
Digital Media Receiver 1, 2	Displaying video portal site
Network Camera 1, 2, 3	Playing real-time recorded video
Network Attached Storage (NAS)	Uploading and downloading files
Printer	Printing document
Consumer Game	Online matchup

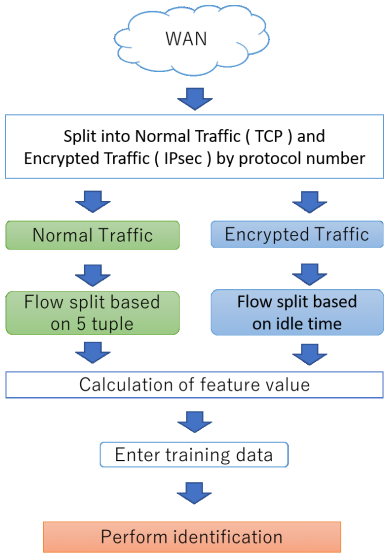

Fig. 2. Identification model

II. Measurement Environment and Procedure

In this section we describe the measurement environment and procedure for collecting, analyzing, and evaluating traffic exchanged by targeted devices.

A. Measurement Environment

The outline of the measurement environment is shown in Figure 1. We deploy an Ethernet switching hub with port mirroring, and connect a target device (Identification Target) intended to be identified. We also connect a measuring machine to the mirrored port. Identification Target communicates with a terminal located in the same network or a server of service provider (according to device type and measurement scenario). According to the measurement scenario, additional nodes (e.g., radius, DNS, dhcp servers) are also deployed if needed. Except the measurement of wireless devices, all devices are connected with wired cable to avoid any instability of measurement results due to wireless environment. Some devices communicates with a vendor-provided server (e.g., video portal, relay node, management server) located in the Internet and sometimes communications are established by using IPSec for security reasons.

B. Measurement Procedure

In this paper, we prepare total 9 devices shown in Table I as target devices for identification. Target devices are classified into 6 categories (Wireless, Digital Media Receiver, Network Camera, NAS, Printer, Game). We first provide a simple measurement scenario for every category. For each target device, we then prepare a measurement environment, i.e., connect the target device to the hub, deploy additional nodes if needed. Finally we manually operate nodes according to the scenario, and capture all packets of both inbound and outbound directions.

III. Proposed Device Identification Method

In this section, we propose a method for device identification, where we first explain the overview of identification, and provide detailed description for each part in the following subsections.

A. Method Overview

Following is the procedure of device identification method. In this paper, we divide monitoring data into two parts, the former is used as the training data for machine learning, and the latter is used as the evaluation data for device identification.

1) Identify monitored traffic to TCP/UDP or IPSec by checking the protocol number in the IP header.
2) Split a whole monitored traffic into flows by using 5-tuple (source/destination IP address, source/destination port number, protocol number)
3) Calculate *traffic features* for every flow
4) Perform device identification by using machine learning

B. Splitting Monitored Traffic into Flows

The procedure of flow split is shown in Figure 2. Since IPSec traffic has multiple flows into a single stream and flow information is encrypted, it is difficult to split IPSec traffic into

flows. We therefore suppose that a single flow is transferred over the IPSec tunnel at the same time, and we distinguish as a different flow if we find an idle longer than a predefined duration (30 sec in this paper).

For splitting into flows, we first check the protocol number in the IP header of monitored traffic. If the protocol number is 6 (TCP) or 17 (UDP), we consider the traffic is normal (non-encrypted), and classify into flows by using 5-tuples. Otherwise, (or the protocol number is 50 (ESP)), we consider the traffic is IPSec, and suppose to terminate the flow at the idle.

C. Calculation of Traffic Features

We define a *traffic feature* as a metric of flow statistics. For example, total number of packets/bytes in the flow, average packet size, average packet inter-arrival time (IAT), and so on. We use a set of traffic features to compose a multi-dimensional vector, as an input of the machine learning. However, some traffic features strongly depend on the conditional parameters. For example, the total bytes or packets in flow on file downloading directly depends on the characteristics of the content such as file size. To avoid any degradation of identification accuracy due to conditional parameters, it would be better to eliminate conditional dependent traffic features. However, the accuracy of the identification is also affected by the number of traffic features. Recent study has shown that there is a tradeoff between the accuracy of identification of machine learning and the total time for learning phase [8], and increasing the dimension of vector is highly related to the learning time. That is, the increase the number of traffic features may improve the accuracy of the identification.

From above reason, we first use only two types of traffic features, i.e., packet size and IAT, to remove any conditional dependencies. To improve the accuracy, in other words, to increase the number of traffic features, we use m-quantile values of packet size and IAT instead of using 4-quantile values using in most related works. In this paper we set $m = 30$ for the device identification after the 200 packets measurement. The size of m may vary based on the condition how many packet can we use for device identification. The smaller number of packets leads the smaller value of m.

All traffic features are calculated for every type of direction, from server to client (S $\rightarrow$ C), from client to server (C $\rightarrow$ S), and bi-directional (S $\leftrightarrow$ C). In total, we use 180 traffic features (30-quantile of packet size and IAT, 3 types of directions).

D. Identification using Machine Learning Algorithm

The calculated traffic features are input to the identificator, and the result is obtained by identification using the machine learning algorithm. For the SVM algorithm we use Weka [1] machine learning software.

IV. Identification Results

[1]http://www.cs.waikato.ac.nz/ml/weka/

TABLE II
IDENTIFICATION OF 6 DEVICES

Target Device	Identification Accuracy
Wireless Access Point	96.2%
Digital Media Receiver 1	98.4%
Network Camera 1	83.3 %
NAS	89.7%
Printer	90.6 %
Consumer Game	98.4%

In this paper, our final goal for the evaluation is to obtain the overall accuracy of device identification from 9 devices shown in Table I. However, to verify how similar devices (i.e., devices in the same category) can be identified correctly, we conducted following two types of evaluation scenarios.

1) **Identification of 6 devices:** we choose one device (Digital Media Receiver 1 and Network Camera 1) from every category, and evaluate how these devices are individually identified.
2) **Identification of 6 categories:** we use 9 devices but identification is performed by category-basis, i.e., we identify the name of category instead of the name of device. In this scenario, all network cameras are identified as the same category (Network Camera) for example.
3) **Identification of 9 devices:** we use all 9 devices and identify them individually, i.e., all network cameras are identified as different devices.

We show the results of above evaluation scenarios in following subsections.

As an evaluation metric, we calculate the accuracy, which represents the number of correctly identified flows divided by the total number of evaluated flows.

A. Identification of 6 Devices

The overall identification accuracy is 96.0%. Identification accuracy by devices is shown in Table II. The table shows that the identification accuracy of Network Camera 1 is relatively lower than others, because the number of flows used (in terms of both for training and evaluation data) is smaller. Network Camera has a tendency that a flow continues for a long time, once a terminal establish a connection to the device. Efficient collection of training data for long-lived flows is important.

B. Identification of 9 Devices

The overall identification accuracy is 88.3% in this case. Table III show the breakdown of identification accuracy. From this table, we can observe that the accuracies of Network Cameras 1 and 2 are low. Since these network cameras generate quite similar traffic, they are mis-identified each other. The identification accuracy of Printer is also low, because flows are often incorrectly identified as Digital Media Receivers.

One typical observation is that many flows of Digital Media Receiver 1, Network Cameras 2 and 3 are identified as Consumer Game.

TABLE III
IDENTIFICATION OF 9 DEVICES

Target Device	Identification Accuracy
Wireless Access Point	91.7%
Digital Media Receiver 1	97.1%
Digital Media Receiver 2	93.0%
Network Camera 1	75.6%
Network Camera 2	76.1%
Network Camera 3	89.2%
NAS	81.7%
Printer	69.8%
Consumer Game	78.9%

TABLE IV
IDENTIFICATION OF 6 CATEGORIES

Target Device	Identification Accuracy
Wireless Access Point	80.6%
Digital Media Receiver	96.1%
Network Camera	91.4%
NAS	78.3%
Printer	76.7%
Consumer Game	75.6%

C. Identification of 6 Categories

The overall accuracy is 88.1%, and individual accuracies are shown in Table IV. Compared to Table III the accuracy of Network Cameras becomes much higher. It is because in this case mis-identifications among different network cameras are treated as the same Network Camera category. The same tendency can be observed in Digital Media Receiver case.

D. Further Enhancement of Identification Accuracy by Using History of Traffic Features

Recall, one of the important things to improve device identification is to increase the number of traffic features. However, using large value of m in m-quantile is not always improve the accuracy. For the accurate identification, each value of m-quantile is needed to be stable, which means an enough number of packets statistically reasonable for each range, otherwise each value of m-quantile has wider range and then it leads potential mis-identifications.

For this problem, one effective approaches to increase the number of traffic features is to reuse the previously calculated traffic features in addition to the current traffic features.

We now suppose a real-time device identification which is conducted progressively, i.e., from the beginning, identification process is performed periodically for every, e.g., 50 packets arrival. The first identification is made at the time of the 50=th packet arrival, and traffic features are obtained by using statistics of first 50 packets. Next, the second identification is made at the 100-th packet arrival. Originally, the second identification is performed by using traffic features by using statistics of first 100 packets. In this case, we do no longer use traffic features calculated in the first identification. However, these traffic features may also be used for the second identification to improve the identification accuracy. In this respect, the total number of traffic features may be twice.

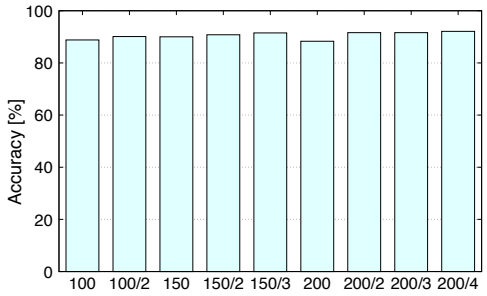

Fig. 3. Impact on Reuse of Traffic Features (9 Devices Identification)

The purpose to reuse the previously calculated traffic features have two objectives, (1) improve the overall identification accuracy by using the same number of packets, and (2) reduce the total number of packets needed to achieve the same accuracy. We evaluate both prespectives in this subsection. Figure 3 shows identification accuracy among different identification conditions. Here we assume device identification is made for every 50 packets and the final identification is made at 200-th packet arrival. Bars for 100, 150, 200 are the identification results by using only traffic features calculated from 100, 150, 200 packets, respectively. Suffix '/2', means that we reuse traffic features at 50-th packet additionally. Suffix '/3' means that we reuse both 50-th and 100-th, and '/4' means that we reuse all calculated traffic features (720 in total). As observed in these results, we can agree that the identification accuracy can be improve by the reuse of traffic features, especially, the improvement is up to 6% (88% to 94%) at 200-th packet identification.

V. CONCLUSION

In this paper, we have proposed a method for identification of communication devices (especially targeted for embed or IoT devices) based on the measurement of traffic pattern statistically. In order to realize concrete identification against degradation of identification accuracy due to conditional parameters, we use only m-quantile of packet size and IAT with $m = 30$. By applying SVM-based machine learning algorithm, we have shown that our device identification can achieve up to 96.0% in identification of 6 devices, 88.3% in 9 devices, and 88.1% in 6 categories. For future research topics, we will adopt further investigation of *key traffic features* which significantly impacts the identification accuracy, to reduce the total number of required packets and realize earlier device identification.

ACKNOWLEDGMENT

This work was supported by JSPS KAKENHI Grant Number JP15H02694.

REFERENCES

[1] Y. Okada, Shingo, N. Nakamura, Y. Nakahira, and I. Oka, "Comparisons of machine learning algorithms for application identification of encrypted traffic," in *Proceedings of the 10th IEEE International Conference on Machine Learning and Applications (ICMLA 2011)*, vol. 2, Honolulu, USA, December 2011, pp. 358–361.

[2] ——, "Application identification from encrypted traffic based on characteristic changes by encryption," in *Proceedings of the 1st IEEE International Communications Quality and Reliability Workshop (CQR 2011)*, Naples, Italy, May 2011, pp. 1–6.

[3] M. Korczynski and A. Duda, "Markov chain fingerprinting to classify encrypted traffic," in *Proceedings of the 33th IEEE Conference on Computer Communications (INFOCOM 2014)*, Toronto, Canada, May 2014, pp. 781–789.

[4] N.-F. Huang, G.-Y. Jai, H.-C. Chao, Y.-J. Tzang, and H.-Y. Chang, "Application traffic classification at the early stage by characterizing application rounds," *Information Sciences*, vol. 232, pp. 130–142, 2013.

[5] K. Yuichi, A. Shingo, N. Nobuyuki, N. Yoshihiro, and O. Ikuo, "Enhancing immediacy of identification with multi-stage application identification," in *Proceedings of the 7th IEEE International Conference on New Technologies, Mobility and Security (NTMS 2015)*, Paris, France, July 2015, pp. 1–2.

[6] T. Tsumuro, S. Ata, and I. Oka, "Statistical Analysis of Network Traffic by Applications of Smart Devices," in *Proceedings of the 18th Asia-Pacific Network Operations and Management Symposium (APNOMS 2016)*, Kanazawa, Japan, October 2016.

[7] Y. Iemura, S. Ata, and I. Oka, "Identification of user behavior based on time variation of traffic statistics," in *Proceedings of the 16th Asia-Pacific Network Operations and Management Symposium (APNOMS 2014)*, Hsinchu, Taiwan, September 2014.

[8] P. Rebentrost, M. Mohseni, and S. Lloyd, "Quantum support vector machine for big data classification," *Physical review letters*, vol. 113, no. 13, p. 130503, February 2014.