Rule-based User Behavior Detection System for SaaS Application

Jee-Tae Park
Computer Information and Science
Korea University
Sejong, Korea
pjj5846@korea.ac.kr

Ui-Jun Baek
Computer Information and Science
Korea University
Sejong, Korea
pb1069@korea.ac.kr

Myung-Sup Kim
Computer Information and Science
Korea University
Sejong, Korea
tmskim@korea.ac.kr

Min-Seong Lee
Computer Information and Science
Korea University
Sejong, Korea
min0764@korea.ac.kr

Chang-Yui Shin
Defense Agency for Technology and Quality
Daejeon, Korea
superego99@dtaq.re.kr

***Abstract*— SaaS is a cloud-based application service that allows users to use applications that work in a cloud environment. SaaS is a subscription type, and the service expenditure varies depending on the license, the number of users, and duration of use. For efficient network management, security and cost management, accurate detection of user behavior for SaaS applications is required. In this paper, we propose a rule-based traffic analysis method for the user behavior detection. We conduct comparative experiments with signature-based method by using the real SaaS application and demonstrate the validity of the proposed method.**

Keywords—SaaS, user behavior detection, rule based traffic analysis

I. INTRODUCTION

As the network environment expands, the importance of network management is increasing. In the field of network management, traffic classification is becoming increasingly important, and in particular, various researches are being conducted for intrusion detection, application identification [1-5]. Traffic classification has been variously performed in the past and it has resulted in a lot of performance improvement. However, due to the emergence of encrypted traffic and the advent of various types of application such as mobile application and cloud-based application, it has become difficult to apply the traditional method [3-5]. To solve this problem, recently, many researches have been conducted to classify encrypted traffic using learning-based analysis methods, and performance has been improved [5-8]. However, relatively few researches have been conducted on new types of applications such as cloud-based applications.

Cloud-based applications operate based on cloud computing and gradually increasing. Cloud-based applications are composed of PaaS (Platform as a Service), IaaS (Infrastructure as Service), and SaaS (Software as a Service) depending on the form and model of cloud service provision, and SaaS is the most used [9, 10]. SaaS refers to software that provides an application program running in a cloud environment in the form of a service. SaaS is a form of subscription by selecting a license and duration according to the required services to be used by the user. (e.g. Google-Apps, Microsoft Office 365).

SaaS service expenditure varies greatly depending on the license and user behavior. In addition, user behavior detection is an important in the field of network management and security because attackers can infer user behavior metadata and various information of the user can be known without decryption [12]. Therefore, it is necessary to accurately detect user behaviors that are related to each other for expenditure management of SaaS applications. In this paper, we focus on SaaS, a subscription type application and conduct SaaS traffic analysis. We propose a rule-based user behavior detection method that defines the detection rule based on target SaaS traffic analysis, and detects the behavior by using the defined rule.

The specific contributions of this paper can be summarized as follows:

- We describe the need for traffic classification researches for network management. In addition, we also explains the importance and necessity of user behavior detection research for subscription-type SaaS applications.
- We propose a rule-based user behavior detection system for SaaS applications. In order to verify the our proposed method, we performed several comparative experiments in terms of performance with the signature-based user behavior detection method. We achieved good performance for detection accuracy but, rule generation spends lots of time . We will conduct further research to reduce the rule generation time in the future.

This work was supported by the Technology Innovation Program grant funded By the Ministry of Trade, Industry & Energy (MOTIE, Korea) and the Korea Evaluation Institute of Industrial Technology (KEIT) (No. 20008902, Development of SaaS SW Management Platform based on 5Channel Discovery technology for IT Cost Saving) and "Regional Innovation Strategy (RIS)" through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (MOE) (2021RIS-004).

The rest of this paper is organized as follows. In Section 2, we will discuss the related work and, describe the proposed system and analysis method in Section 3. The experiments that have been conducted to evaluate the proposed method using the SaaS traffic (i.e. Microsoft Office 365) are presented in Section 4. Finally, we conclude the paper and outline future research directions in Section 5

II. Related Work

Traffic classification research has been actively conducted since the past, and representatively, port-based, and payload-based method. Port-based analysis is a method of classifying applications with fixed ports [1, 2]. The payload-based analysis method is a method of using the contents of the payload in a packet [1-4]. However, the use of encrypted traffic makes difficult to use the these traditional methods [1, 2].

In order to solve the problems of the traditional methods, machine learning and deep learning-based analysis methods have recently been performed [5-9]. The learning-based analysis method is a method of extracting features for each application and classifying target applications by learning the extracted features. For each research, the pre-processing, feature extraction, and feature selection method can be variously configured, and performance has been improved [5-7]. However, the learning-based analysis method has several problems. In that the performance varies greatly depending on the learning model and features, and the labeled traffic data is relatively scarce. In addition, a lot of time and computational power are required in the training and testing.

III. Proposed Method

A. User Behavior Definition

User behavior can be defined in various ways according to application type, usage form, provided service and function. In this paper, we define the 6 behaviors of the common process shown in Table 1: Application Start(AS), Login(LI), Feature Start, Feature End(FE), Logout(LO) and, Application End(AE). Feature is defined as a sub-application (e.g. PowerPoint, Word, Excel) used by users after login.

TABLE I. A Description of the User Behavior

Behavior	Description	Preceded Behavior
Application Start (AS)	SaaS software start	None
Login (LI)	Logging in the account	Application start
Feature Start (FS)	SaaS Feature start	Application start & Login
Feature End (FE)	SaaS Feature end	Application start & Login
Logout (LO)	Logging out the account	Application start & Login
Application End(AE)	SaaS software end	Application star

As shown in Table 1, application start and application end refer to the start and end of the target SaaS application, and login means login success after entering account information. Feature start and feature end refer to the start and end of the feature, and logout means logout of the user account. Each of the defined behaviors is related to each other, and there is an behavior that must be preceded before the behavior is performed. For example, for login, the application start should be preceded, and for feature start and logout, the application start and login should be preceded. The explanation of the defined behavior and information on the behavior that should be preceded are shown in Table 1.

B. Rule based Behavior Detection System

The structure of the proposed system consists two parts as rule generation based on traffic analysis and behavior detection using defined rules, and is shown in Figure 1.

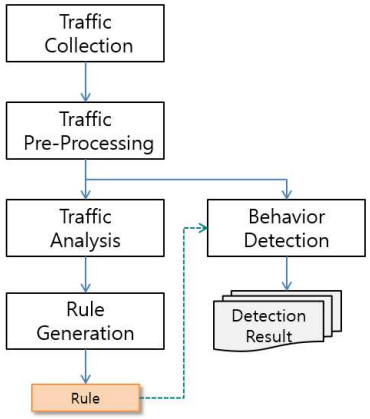

Figure 1. Entire Structure of Rule based Behavior Detection

Rule generation process through traffic analysis is shown in Figure. 2. In this paper, we define a set of packets with the same 5-tuple information as a flow. Target traffic is collected by using Wireshark (.pcap) for each defined behavior. Collected packet-based traffic is converted into flow-based traffic (.fwp) in preprocessing.

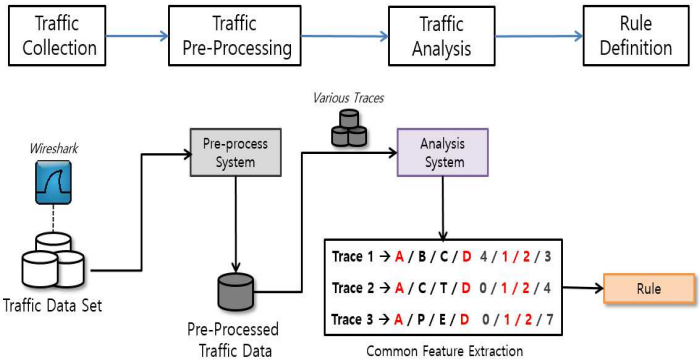

Figure 2. A Process of Rule Generation based on Traffic Analysis

Flow and packet-based analysis is performed on preprocessed traffic to extract flow and packet characteristics for each behavior. To define an accurate behavior detection algorithms, we collected traffic from various environments (e.g. multiple hosts, different collection time).

A description of the detection rule format is shown in Table 2 and consists of 4 detailed rules: SaaS, flow, packet, and status rule. SaaS rule represents target SaaS information and behavior to be detected, and the flow rule represents information on a flow to be detected among the generated traffic. Packet rule represents information on the packet to be verified in the flow to be detected, and the state rule represents an behavior that must be preceded to detect the target behavior.

TABLE II. A DESCRIPTION OF THE DETECTION RULE FROMAT

Rule	Information	Description	Example
SaaS Rule	SaaS	Target SaaS Information	Microsoft Office 365
	Behavior	Detecting Behavior	Application Start
Flow Rule	Client IP	Client IP Information	Any
	Client Port	Client Port Information	Any
	Protocol	Protocol	TCP
	Server IP	Server IP Information	13.xxx.xxx.xxx
	Server Port	Server Port Information	443
Packet Rule	Check Packet Number (CPN)	A Number of Check Packet in the Flow	4
	Direction	A Direction of Check packet in the Flow	CS *(Client to Server)*
	SNI	SNI Information of the Check Packet	www.office.com
State Rule	Behavior State	Preceded Behavior Information	none

The client is a host that has performed a user behavior, and the server represents the destination of the host. The Check Packet Number (CPN) represents the packet number to be verified, the Direction represents the direction of the packet to be verified, and consists of Client to Server (CS) and Server to Client (SC). SNI represents SNI information in a packet, and checks if the CPN and Direction rules are satisfied. The Behavior State represents information on behaviors that must be preceded for each behavior. For example, if a packet with SNI information of www.office.com in the 13.xxx.xxx.xxx flow of TCP 443 is used for application start detection in Microsoft Office 365, the detection rule is defined as an example in Table 2.

Behavior detection using defined rules is shown in Figure 3, and consists of three parts: traffic collection, pre-processing, and behavior detection. In behavior detection, a method of traffic collection is different with rule generation but the preprocessing is the same. We collected the traffic by using the Wireshark in rule generation, but the behavior detection process uses packet collection system based on packet mirroring. Packets are collected for multiple hosts in every minutes. Collected packets are converted into flow based traffic same as the rule generation and transmitted to the detection system.

In detection system, user behavior is detected by entering pre-processed traffic, predefined rules and previous detection results. The previous detection result indicates the detection result one minute ago, and the same process is performed every minute. User behavior detection is detected when all the rules defined for each behavior are satisfied with the input traffic and detection result is derived. Detection result consists of detection time, host, SaaS information, and detection behavior.

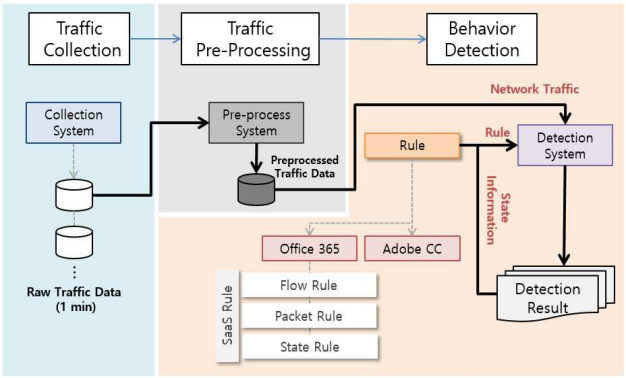

Figure 3. A Process of Behavior Detection

IV. EXPERIMENT

A. Experiment Environment

In this paper, to verify the validity of the proposed method, we conduct comparative experiments using the same traffic dataset as the existing signature-based user behavior detection method. We used Microsoft Office 365 as the target SaaS application in the experiment, and Table 3 presents the description of the traffic dataset.

TABLE III. A DESCRIPTION OF THE TRAFFIC DATASET

	Traffic Information		Application Information	User Behavior Sequence
	Flow	Packet		
Trace 1	228	3,841	**[SaaS]** Office 365 Web, Messenger	AS–LI–LO–AE
Trace 2	456	4,390	**[SaaS]** Office 365 (PowerPoint) Web, Music	AS–LI–FS-FE–AE
Trace 3	352	4,012	**[SaaS]** Office 365 (Word) Music, Messenger	AS–LI-FS-FE–LO-AE
Trace 4	229	2,890	**[SaaS]** Office 365 (Excel) Messenger, Video Streaming	AS–LI-FS-FE–LO-AE
Trace 5	582	6,552	**[SaaS]** Office 365 (Excel) Web, Messenger, Video Streaming	AS–LI-FS-FE–LO-AE

In Table 3, the traffic information indicates the number of flows, packets in each trace, the application information indicates the application name including the target SaaS application, and the User Behavior Sequence indicates the behavior sequence performed in each trace. The traffic used in the experiment was collected by using the target SaaS application and general applications such as web browser,

messenger, video streaming and music together, and the usage time, behavior, and application used together were set differently for each trace. In order to compare the user behavior detection results with actual behavior, we tagged the application information, host, behavior, and execution time in advance. We compare the tagged information with the detection result. If the time, application name, and host are detected correctly, it is classified as a true detection, if even one is detected incorrectly, it is a false detection, and if the performed behavior is not detected, it is classified as non-detection. We calculate the recall, precision and f-measure for our evaluation measurement.

B. Experiment Result

The experiment results are shown in Table 4. We conducted comparison experiments with the signature-based [11] using the 5 traces. As a result of the experiment, the recall of both the signature-based method and the rule-based method is 95-100%, and the precision is 98-100% in the rule-based method, but 85-100% in the signature-based method. Comparing the average recall, precision, and f-measure for each trace, the rule-based method shows higher performance than the signature-based method. However, the rule-based method also causes false detections, which means that the defined rule is not complete. Therefore, in order to reduce false detection in several traces, we need to refine the detection rules.

TABLE IV. . RESULT OF THE EXPERIMENTS

		Detection Result		
		Recall (%)	Precision (%)	F-measure
Signature-based Method [11]	Trace 1	100	85.12	91.96
	Trace 2	100	90.17	94.83
	Trace 3	100	94.12	96.97
	Trace 4	100	90.23	94.86
	Trace 5	95.45	89.10	92.16
	Average	99.09	89.74	94.15
Rule-based Method	Trace 1	100	100	100
	Trace 2	97.44	100	98.70
	Trace 3	100	95.23	97.55
	Trace 4	100	100	100
	Trace 5	100	96.40	98.16
	Average	99.48	98.54	98.88

In addition, in the rule-based method, it takes too much time and effort to generate a rule. When generating a rule to detect single behavior, it takes about 120 to 200 minutes on average. If we define more behaviors in other applications, rule generation time will increase significantly. It seems to occur because the traffic analysis and rule definition performed in rule generation are manually performed and the process is complicated. Therefore, we plan to conduct research to improve rule generation in the next research.

V. CONCLUSION

In this paper, we described the differences between SaaS and other application, and the need for SaaS user behavior detection research in the introduction. In this paper, we proposed a rule-based behavior detection method and defined user behaviors. The rule-based behavior detection method consists of the detection rule generation based on traffic analysis and the rule-based behavior detection. We conducted comparative experiments with signature-based method by using Microsoft Office 365. As a result of the experiment, when the rule-based method is applied, recall, precision, and f-measure of 98%, which is higher than the signature-based method. However, our proposed method requires a lot of time and effort in rule generation.

As a future research, we will refine and elaborate the defined detection rules to reduce the false detections, and define rules for other SaaS applications as well. We also plan to conduct additional research to reduce the rule generation time in the future.

REFERENCES

[1] M. Finsterbusch, C. Richter, E. Rocha, J.-A. Muller, and K. Hanssgen, "A survey of payload-based traffic classification approaches," IEEE Communications Surveys & Tutorials., vol. 16, no. 2, pp. 1135–1156 2013

[2] A. Dainotti, A. Pescape and K. Claffy, "Issues and Future Directions in Traffic Classification,", IEEE Network, Vol. 26, no. 1, pp. 35-40, 2012

[3] C. Modi, D. Patel, B. Borisaniya, H. Patel, A. Patel, and M. Rajarajan, "A Survey of Intrusion Detection Techniques in Cloud," Journal of Network and Computer Applications, vol. 36, no. 1, pp. 42–57, 2013

[4] M. Stoyanova, Y. Nikoloudakis, S. Panagiotakis, E. Pallis, and E. K. Markakis, "A survey on the Internet of Things (IoT) forensics: Challenges, approaches and open issues," IEEE Communications Surveys & Tutorials, vol. 22, no. 2, pp. 1191–1221, 2020

[5] R. Boutaba et al., "A Comprehensive Survey on Machine Learning for Networking: Evolution, Applications and Research Opportunities," Journal of Internet Services and Applications, vol. 9, no. 1, p. 16, 2018

[6] S. Zander, T. Nguyen, and G. Armitage. "Automated Traffic Classification and Application Identification using Machine Learning." The IEEE Conference on Local Computer Networks 30th Anniversary (LCN'05) l. IEEE, 2005.

[7] P. Wang, X. Chen, F. Ye, and Z. Sun, "A Survey of Techniques for Mobile Service Encrypted Traffic Classification using Deep Learning," IEEE Access, vol. 7, pp. 54024–54033, 2019

[8] A. W. Moore and D. Zuev, "Internet traffic classification using Bayesian analysis techniques," in Proc. ACM SIGMETRICS International Conference Measurement and Modeling. Computer Systems, pp. 50–60, 2005

[9] N.-F. Huang, G.-Y Jai, C.-H. Chen, and H.-C. Chao. "On the Cloud-based Network Traffic Classification and Applications Identification Service." 2012 International Conference on Selected Topics in Mobile and Wireless Networking. IEEE, 2012

[10] H. Wang, K. Tseng and J. Pan, "A novel statistical automaton for network cloud traffic classification," 2012 International Conference on Information Security and Intelligent Control, pp. 49-52, 2012

[11] M.-S, Lee, J.-T. Park, J.-W. Choi and M.-S Kim, "Microsoft Office 365 Service Detection based on Payload Signature", Proc. Symposium of the Korean Institute of communications and Information Sciences, pp. 357-358, 2020

[12] C. Hou, J. Shi, C. Kang, Z. Cao and X. Gang, "Classifying User Activities in the Encrypted WeChat Traffic," 2018 IEEE 37th International Performance Computing and Communications Conference (IPCCC), 2018, pp. 1-8 .