Framework for Multi-Level Application Traffic Identification

Sung-Ho Yoon, Kyu-Seok Shim, Su-Kang Lee, and Myung-Sup Kim
Dept. of Computer and Information Science
Korea University
Sejong, Korea
{sungho_yoon, kusuk007, sukanglee, tmskim}@korea.ac.kr

***Abstract*— With the acceleration of the Internet speed and the vigorous emergence of new applications, the amount of Internet traffic has increased. In order to provide stable Internet service, efficient network management based on accurate traffic identification is critical. Although various methods for traffic identification have been proposed, not a single method identifies all types of Internet traffic. In this paper, we propose a framework for multi-level application traffic identification by combining several single methods.**

Keywords—traffic identification; traffic classification; multi-level; framework

I. INTRODUCTION

The volume of network traffic is continuously increasing because of new multimedia applications and advancements in Internet technology. In this type of situation, efficient network management is needed to provide Internet users with a stable Internet service [1]. The network policy is established based on traffic identification results, which analyze traffic sources such as applications, services, and protocols. The ultimate goal of a traffic identification method or system is to accurately and quickly identify all traffic in the target network.

Various methods are proposed as emphasizing the importance of traffic identification. However, because of traffic complexity and multiplicity, a single method cannot be guaranteed to identify all types of Internet traffic, as various applications and services are continuously emerging. In this paper, we propose a novel framework for multi-level traffic identification using various signature models. The contributions of this paper are as follows. First, the proposed framework can improve identification performance by combining several single signature models. These signature models are designed by our research group. Second, the framework has an extendable structure for further designed signature models. Currently, many methods using various traffic features are being proposed. In view of these upcoming methods, the framework includes independent and extendable detail signature identifiers. Finally, this framework includes not only identification modules, but also additional modules. The proposed framework consists of five detailed modules: the signature constructor, identifier, visualizer, signature maintainer, and utilizer.

The remainder of this paper is organized as follows. In Section 2, we review various traffic identification methods. In Section 3, we describe our framework in detail. Finally, we conclude our work and propose a future research direction in Section 4.

II. RELATED WORK

Traffic identification methods are evolving constantly in order to adapt to dynamic network environments. The most primitive method uses well-known ports assigned by The IANA (Internet Assigned Number Authority). In earlier decades, this method could identify Internet traffic with a high level of reliability and accuracy. However, the emergence of applications using random or dynamic port numbers to evade firewalls has reduced the accuracy of port-based identification methods to less than 70%. Thus, the port number no longer indicates a particular application or service [2].

The payload-based method identifies traffic by checking the existence of a certain bit string in the packet payload. It offers sufficient completeness and accuracy due to inspecting the packet payload directly. However, this method tends to encounter difficulties related to traffic encryption, computation complexity, and invasion of privacy.

The statistic-based method uses the statistical characteristics of traffic, such as the distribution of the packet size, direction, or interval, without inspecting the payload [3]. To accomplish this, it converts the packet to a flow unit. A flow is a set of packets having the same 5-tuple (source IP address, source port number, destination IP address, destination port number, transport layer protocol) and payload of first N packets. Although this method overcomes some of the issues of the payload-based method, it is difficult to distinguish between applications using the same communication engine or application-level protocol, as they have similar statistical characteristics.

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(2015R1D1A3A01018057), Next-Generation Information Computing Development Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT & Future Planning (2010-0020728) and KREONET-Emulab Testbed of Korea Institute of Science and Technology Information.

In order to overcome the limitations of applying a single method, multi-level methods combining several single methods have recently been proposed [4, 5]. A common assertion of multi-level methods is that they can improve the identification performance if several single methods are combined.

III. FRAMEWORK FOR TRAFFIC IDENTIFICATION

In this section, we explain a framework to identify Internet traffic. Figure 1 shows FORMULA (Framework for Multi-Level Application Traffic Identification), the method proposed in this paper. FORMULA is composed of five modules.

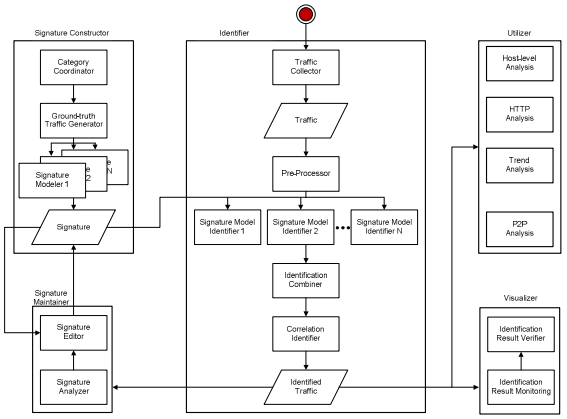

Fig. 1. FORMULA: Framework for multi-level application traffic identification

A. Signature Constructor

A signature is a unique characteristic of the target application that distinguishes it from other applications. To generate a signature, we select target applications and collect their traffic. Then, we define the class taxonomy of the target application. The collected traffic and the class taxonomy are used as input data for the signature generator. The signature generator extracts the signature based on various signature models such as header, payload, or statistic.

The ground-truth traffic generator collects traffic from the target application with labels such as the application or service name. This traffic is utilized not only in the signature generation process but also in the process of verifying the results to test the performance of the identification system. There are various methods of generating the ground-truth. First is DPI (Deep Packet Inspection) which uses the keywords of open protocols as the payload signature. Second method is agent-based; it installs the program collecting the socket information at the end-host that is generating the traffic.

The category coordinator defines the multiple and hierarchical class taxonomy of the target application. The definite taxonomy of application makes identification result more clearly. In addition, it enables an objective evaluation and comparison between the signature models. The class taxonomy is organized into one or more horizontal identification criteria, and each identification criterion has hierarchical attributes [6]. Thus, it is possible to identify the same traffic in multiple and hierarchical ways. The identification criteria used in this framework are service, application, protocol, and function, as shown in Figure 2. The service and application criteria have three-level hierarchical attributes and the protocol and function criteria have two-level hierarchical attributes.

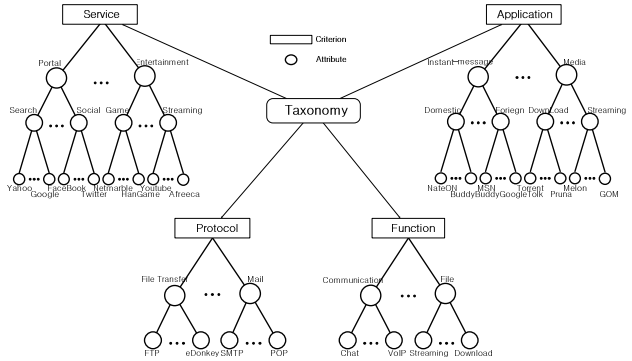

Fig. 2. Class taxonomy in the category coordinator

The proposed framework in this paper supports signature modelers of many types. Because application traffic behavior is becoming increasingly complex, and collectible traffic features are limited by the network environment, diverse signature models must be applied. The signature models applied to this framework are header, payload, and statistic. The header signature model uses the IP address of a specific application server or DNS query message [7]. The payload signature uses a regular expression which is converted from the bit string located behind the L4 protocol packet header [8]. The statistic signature uses a vector which is constructed from the order and size distribution of packets occurring in the same session [9].

B. Identifier

The traffic identifier uses the various signatures to analyze traffic occurring on the target network. In order to improve performance, it collects the raw traffic, reconstructs it into proper traffic units, and then conducts preprocessing, such as abnormal traffic and operating system analysis. The detail signature model identifiers are characterized by signature models that analyze the traffic in parallel, and the identification results are combined by the identification combiner. The integrated results are used for additional identification by correlating the identified and unidentified traffic.

The traffic identifier operates in a number of different ways. The real-time mode identifies traffic as soon as it is captured; the period mode runs the system at a given time interval, such as one or five minutes, on traffic captured during the previous interval; and off-line mode identifies the traffic in storage. We can select the mode based on the network and purpose of the traffic identification.

The traffic collector conducts port mirroring or tapping on the router or switch where all target network packets can be collected. The collected packet unit traffic is reconstructed by the flow unit. The reconstructed flow unit traffic provides more features because the request and response traffics are combined into a unit, and thus various signature models can be applied. In addition, it reduces the volume of storage and decreases the system overhead. The flow used in the framework is a bi-directional flow defined as a set of packets having the same 5-

tuple (source IP address/port, destination IP address/port, transport layer protocol). The flow not only includes statistical information, such as the start and finish time, number of packets, byte size, number of TCP flag packets and data packets, but also payload information. The payload information indicates the first N packets' payload, and can be adjusted by the available storage space and applicable range of the signature model. We define this traffic unit as a *flow_with_packet*; it is stored in the form of a hash data structure in order to minimize the generation and access time.

The signature model identifier is composed of detail identifiers based on several signature models. Because traffic identification methods vary from model to model, the identifier must also be operated individually. For example, the header signature model uses only the header information, so the function that inspects the header information is only applied to this identifier. A payload signature model additionally applies the function inspecting the payload. Each detail identifier is performed in parallel because the result of each signature model is different in terms of class criteria.

The results of the individual signature model identifiers are combined by the identification combiner. Because the class criteria are different depending on the signature models, the results of the detail identifiers must be combined, and all class criteria should be specified. If different identifiers have a result on the same class criterion, we call it a *collision*, and the integration algorithm chooses one. The integration algorithm is directed by the frequency of the results and the priority of the identifier in this framework. The proposed algorithm is revised from a weighted combination of machine learning techniques [10].

Let there be a set of identifiers $I = \{I_1, I_2, \ldots I_n\}$, where each is a signature model identifier. The result set is $R = \{R_1, R_2, \ldots R_m\}$, which is all possible identification results. If an identifier I_i identifies traffic x, the result vector is $I_i(x) = [I_{i,1}(x), \ldots, I_{i,j}(x), \ldots I_{i,m}(x)]$, where, if I_i identifies x as R_j, $I_{i,j}(x)$ is 1, otherwise 0. We construct a matrix $I(x)$ of all result vectors of the set of identifiers, as follows:

$$I(x) = \begin{bmatrix} I_{1,1}(x) & \ldots & I_{1,j}(x) & \ldots & I_{1,m}(x) \\ \ldots & & & & \\ I_{i,1}(x) & \ldots & I_{i,j}(x) & \ldots & I_{i,m}(x) \\ \ldots & & & & \\ I_{n,1}(x) & \ldots & I_{n,j}(x) & \ldots & I_{n,m}(x) \end{bmatrix} \quad (1)$$

$$F_j(x) = \sum_{i=1}^{n} I_{i,j}(x) \quad (2)$$

$$\mathrm{t} = \underset{j=1,m}{\operatorname{argmax}} F_j(x) \quad (3)$$

All values in each column of matrix $I(x)$ are added by $F_j(x)$, as shown in Equation (2). In other words, the yielded value indicates the number of identifiers with the same result. We choose the index having a maximum value using Equation (3); the result by the majority is decided as the final result. If t is greater than two, we decide the final result according to the predefined priority of the signature model identifiers.

The correlation identifier receives the result from the identification combiner, and then identifies the traffic using the correlations between identified and unidentified traffic. Specifically, it groups the traffic using certain correlated features. If the group includes identified traffic, the result of the traffic represents the group. In the system, four algorithms are used to correlate identifiers as shown in Figure 3. They are a server-client method that analyzes the traffic using a common 3-tuple (IP address, port, and transport layer protocol), an occurrence time method that analyzes the traffic from a single host in a certain time period, a host-host method that analyzes the traffic that occurs between two specific hosts, and a statistical method that analyzes the traffic using statistic information.

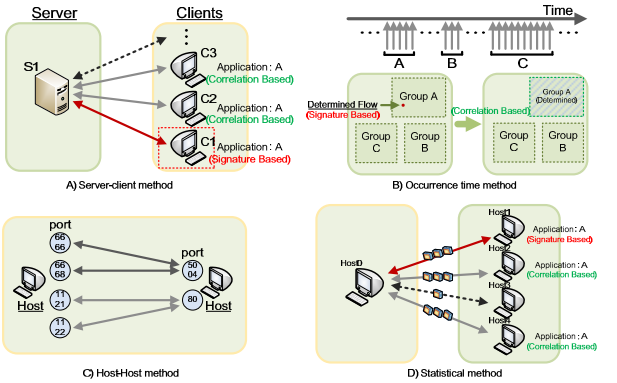

Fig. 3. Correlation identifer

C. Visualizer

The visualizer shows the results of the traffic analyzed by the identifier form various perspectives and provides information measured by the identifier. In addition, by measuring the accuracy of the traffic, it verifies the feasibility of the signature and the performance of the identifier. The result of this module is reported to the signature producer, the network manager, and the identifier operator immediately via Internet.

The identification result monitor displays the result using a variety of graphs and charts. The metric used to measure the result is completeness, as shown in Equation (4). This refers to the ratio of identified traffic to the total traffic. Because the identified traffic result is analyzed based on a multi-level and hierarchical classification taxonomy, the result table uses a tree structure. In addition, it provides time-line graphs for days, weeks, and months to show the change over time. Further, additional information, such as CPU usage, memory usage, and identification time, is provided to determine hardware status.

$$\text{Completeness} = \frac{Identified\ traffic}{Total\ traffic} \quad (4)$$

The identification result verifier verifies the result using ground-truth traffic. The metrics for measuring the accuracy of the result are overall accuracy, precision, recall, and F-measure. The overall accuracy is the ratio of the sum of all correctly identified traffic to the sum of all target traffic. Precision is the ratio of correctly identified traffic over all predicted traffic in an individual application; and recall is the ratio of correctly identified traffic over all ground-truth traffic for an individual application.

$$\text{Overall\_accuracy} = \frac{\sum TP}{Total\ traffic} \quad (5)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (6)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (7)$$

D. Signature Maintainer

The signature maintainer module analyzes the signature performance and status based on the verification results from the identification result verifier, and edits the signature list. The signature analyzer uses five maintenance measures to present the numeric value of signature performance.

$$\text{AcoS}(x) = \frac{TP(x)}{TP(x) + FP(x)} \quad (8)$$

$$\text{ExoS}(x) = \text{LUA}(x) - LUS(x) \quad (9)$$

$$\text{CorS}(x \rightarrow y) = \frac{supp(x \cup y)}{supp(x)} \quad (10)$$

TABLE I. SIGNATURE MAINTAINER METRIC

Matric	Significance
ComS (Completeness of Signature)	Amount of traffic in bytes and packets identified using the signature
FreS (Frequency of Signature)	Number of usage of the signature
AcoS (Accuracy of Signature)	Ratio of correctly identified traffic over traffic predicted by the signature using Equation (9)
ExoS (Expiration of Signature)	Time difference between the LUS (last usage time of signature) and LUA (last identified time of application) using Equation (10)
CorS (Correlation of Signature)	Signature pair used at the same time using Equation (11)

The signature editor edits the signature list according to the maintenance measures. A signature with a high value of ComS and FreS moves to the beginning of the list. A signature with a value of AcoS, ExoS, or CorS lower that a threshold is removed.

E. Utilizer

Utilizing the identification results in network management, requires an additional process because the result shows only the amount of application traffic in numeric values. The utilizer module provides various analyses for network managers, service providers, and Internet users. This framework provides four detail utilizers, such as a host-based analysis providing host behavior, trend analysis providing the actual usage of the Internet service, HTTP analysis providing the web service state, and P2P analysis providing complex behavior. Additional utilizers will be added.

IV. CONCLUSION AND FUTURE WORK

With the rapid development of the Internet in recent years, the importance of multi-level identifiers has received a growing emphasis. In this paper, we propose FORMULA (framework for multi-level application traffic identification) using various signature models. This framework consists of not only the essential parts, such as the signature constructor, identifier, and visualizer, but also additional parts, such as the signature maintainer and utilizer, for improving identification performance and utilization.

In future research, we plan to construct a system on a real network based on this framework; also we also plan to add various signature models.

REFERENCES

[1] H. Kim, K. C. Claffy, M. Fomenkov, D. Barman, M. Faloutsos, and K. Lee, "Internet traffic classification demystified: myths, caveats, and the best practices," in *Proceedings of the 2008 ACM CoNEXT conference*, 2008, p. 11.

[2] A. W. Moore and K. Papagiannaki, "Toward the accurate identification of network applications," in *Passive and Active Network Measurement*, ed: Springer, 2005, pp. 41-54.

[3] 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, May 2013.

[4] G. Szabó, I. Szabó, and D. Orincsay, "Accurate traffic classification," in *World of Wireless, Mobile and Multimedia Networks, 2007. WoWMoM 2007. IEEE International Symposium on a*, 2007, pp. 1-8.

[5] A. Callado, J. Kelner, D. Sadok, C. Alberto Kamienski, and S. Fernandes, "Better network traffic identification through the independent combination of techniques," *Journal of Network and Computer Applications,* vol. 33, pp. 433-446, 2010.

[6] K. Ji-hye, Y. Sung-Ho, and K. Myung-Sup, "Study on traffic classification taxonomy for multilateral and hierarchical traffic classification," in *Network Operations and Management Symposium (APNOMS), 2012 14th Asia-Pacific*, 2012, pp. 1-4.

[7] S.-H. Yoon and M.-S. Kim, "An efficient method to maintain the header signatures for internet traffic identification," in *Network Operations and Management Symposium (APNOMS), 2013 15th Asia-Pacific*, 2013, pp. 1-3.

[8] J.-S. Park, S.-H. Yoon, and M.-S. Kim, "Software Architecture for a Lightweight Payload Signature-Based Traffic Classification System," in *Traffic Monitoring and Analysis*. vol. 6613, J. Domingo-Pascual, Y. Shavitt, and S. Uhlig, Eds., ed: Springer Berlin Heidelberg, 2011, pp. 136-149.

[9] H.-M. An, M.-S. Kim, and J.-H. Ham, "Application traffic classification using statistic signature," in *Network Operations and Management Symposium (APNOMS), 2013 15th Asia-Pacific*, 2013, pp. 1-6.

[10] Y. Jinghua, Y. XiaoChun, W. Zhigang, L. Hao, and Z. Shuzhuang, "A novel weighted combination technique for traffic classification," in *Cloud Computing and Intelligent Systems (CCIS), 2012 IEEE 2nd International Conference on*, 2012, pp. 757-761.