Traffic Classification Method Based On Data Stream Fingerprint

Kefei Cheng[1, a], Guohui Wei[2,b] and Xiangjun Ma[3,c]

[1] College of Computer Science Chongqing University of Posts and Telecommunication Chongqing, China

[2]College of Computer Science Chongqing University of Posts and Telecommunication Chongqing, China

[3]Chongqing Municipal Public Security Bureau Network Security Corps Chongqing,China

[a] chengkf@cqupt.edu.cn, [b]1539137612@qq.com

Keywords: Traffic classification, Renyi cross entropy, Data stream fingerprint, Similarity

Abstract. Traffic classification is a method for categorizing the computer network traffic into a number of traffic class based on various features observed passively from the traffic. In recent years, duo to the rapid development of the Internet, as well as the rapid increase of different Internet application, the requirement to distinguish between the different applications is rising. Many traditional methods like port based, packets based and some alternate methods based on machine learning approaches have been used for the traffic classification. In this paper, a new traffic classification method was proposed to utilize the data stream fingerprint information generated by an application. The proposed new method is compared with other network traffic classification methods. The experimental results show that the classification accuracy of the new method meet the actual needs.

1. Introduction

In recent years, more and more new network applications appear, especially in mobile Internet area. In order to use the bandwidth effectively and provide better network service, the traffic should be classified according to the difference of Internet application. Traditional network classification method based on port, or the payload for traditional web applications performs well. However, with the appearance of large numbers of new network applications, the accuracy of classification for the traditional networks is lower and lower. In this paper, a traffic classification method based on data stream fingerprint is introduced to improve the classification accuracy.

2. Related Work

At present, the mainly used method for network traffic classification are: IP traffic classification based on port, IP traffic classification based on payload, traffic classification based on host behavior, IP traffic classification based on machine learning.

2.1 Port based IP traffic classification

Traditional classification methods identify different types of applications according to the well-known port numbers(IANA designated port number). However, this approach has its limitations, some applications may not have their ports registered with IANA(for example, peer to peer applications such as Napster and Kazaa). An application may use some ordinary ports other than well-known ports to avoid access control restrictions of operating system. Although port-based traffic classification is the fastest and simplest method, several studies have shown that it performs poorly, less than 70% accuracy in classifying flows[1] [2].

2.2 Payload based IP traffic classification

Classification method based on the payload is known as examining whether the payload contains special label for traffic classification. This method is usually used for P2P traffic detection and network intrusion detection. But this method also has some defects: firstly, it can only classify the

unencrypted traffic, and has no effect on encrypted traffic or private agreement. Secondly, the analysis of content on the application layer directly leads to the issues of privacy infringement.

2.3 Traffic classification based on host behavior

Another new kind of method is based on the patterns of host behavior at the transport layer[3]. It pays attention to all the traffic generated by a specified host, and can accurately associate each host with the services it provides or uses. Authors in [4] investigated some fundamental characteristics of network applications, such as the huge network diameter and the presence of many hosts acting as both servers and clients, to classify the network traffic. However, this method is time-consuming and cannot classify a single flow, since it must gather information from server flows of each host before it can identify the role of a host.

2.4 Traffic classification based on machine learning

With the increasing demand of network classification, newer methods relying on traffic statistical characteristics was used to identify the application[5][6]. This methods firstly assume that traffic generated by some kind of applications has some unique features. However, due to the need for large-scale data sets for statistics, a traffic classification method based on machine learning is proposed in literature[7]. In many cases they could be hardly used for real-time network traffic classification due to the complexity of machine learning algorithms[8][9].

3. Methodology

This paper sums up the advantages and disadvantages of the current network traffic classification methods, a new traffic classification method based on data stream fingerprint is proposed. The framework of the proposed method is described in Fig.1, which can be divided into multi-steps as follows:

Step 1: Collecting data packet and then constructing the session flow.

Step 2: Extracting the data stream fingerprints in the session flow, and determining whether the current process is in the learning phase. If so, putting the data stream fingerprint into the sample data stream fingerprint database, then ending process. If not, go to the next step.

Step 3: Calculating the data stream fingerprint distribution probability of the current packets. Calculating the similarity between the current data stream distribution probability and the sample data stream fingerprint in the database, and obtaining the final results.

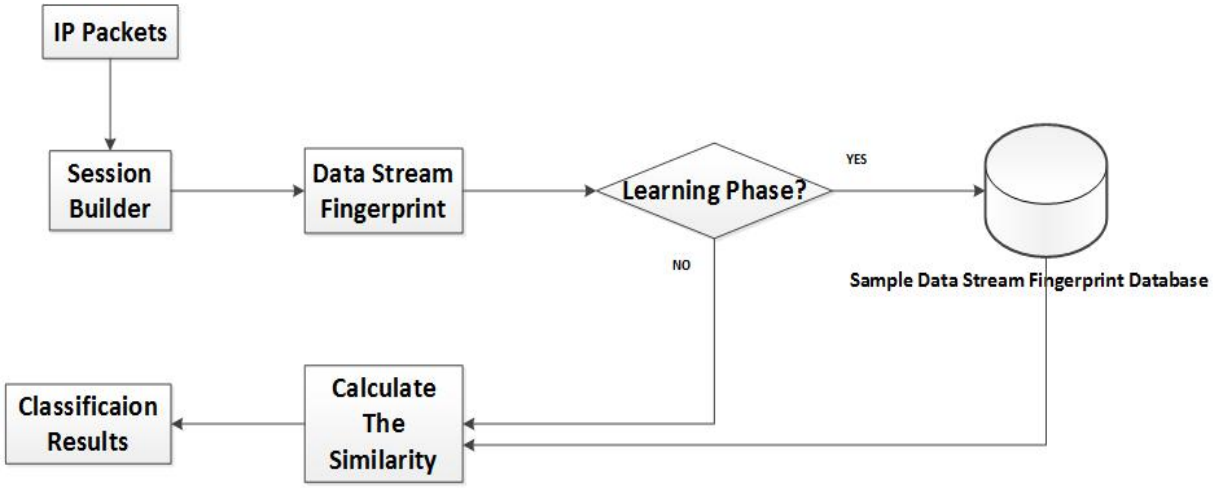

Fig.1 Framework of the proposed method

3.1 Pre-processing

Pre-processing is mainly used to collect the network packets, and then using the inspection packet header information to construct the session flow. A flow can be define as successive IP packets having the same 5-tuple: source IP, destination IP, source port, destination port, and transport layer protocol.

3.2 The data stream fingerprint

In this paper, we use a set of positive integers between 0 and 255 to describe the payload of a packet, which can be expressed in Eq.(1), where the variable *Payload* refers to the payload data, $X_i \in [0,1,2,\ldots,255]$ is the element that is composed of the variable *Payload*. We obtain the probability distribution by accounting the number of X_i from the *Payload*. In this paper, the probability distribution of X_i is called as Data Stream Fingerprint (DSF). The DSF of a packet can be expressed in Eq.(2), where $p = (p_0, p_1, p_2, \ldots, p_i, \ldots, p_{255})$ is the probability distribution of the variable *X*.

$$Payload = \{X_i \mid 0 \le i \le 255\} \tag{1}$$

$$DSF = \begin{pmatrix} X \\ p \end{pmatrix} = \begin{pmatrix} X_0 X_1 X_2 \cdots X_i \cdots X_{255} \\ p_0 p_1 p_2 \cdots p_i \cdots p_{255} \end{pmatrix} \tag{2}$$

3.3 Similarity of data stream fingerprint

In order to classify the current network traffic, we need to compare the similarity between current data stream fingerprint and the training sample data stream fingerprint. We use α orders Renyi cross entropy to compare the similarity. α orders Renyi cross entropy is defined as Eq.(3):

$$I_\alpha(p,q) = \frac{1}{1-\alpha}\log_2 \sum_\gamma \frac{p_\gamma^\alpha}{q_\gamma^{\alpha-1}} \tag{3}$$

Where *p* and *q* are two discrete stochastic variables, p_r and q_r are the probability function of *p*, *q*. One important property of Renyi cross entropy is that if the discrete stochastic variables have the same distribution, then $I_\alpha \to 0$. Then the entropy measure in Eq.(3) is asymmetric, which means that $I_\alpha(p,q) \ne I_\alpha(q,p)$. But in our method, the symmetric of Eq.(3) is suitable for the similarity of data stream fingerprint. So when chose $\alpha = 0.5$, the Renyi cross entropy can be rewritten into Eq.(4).

$$I_{0.5}(p,q) = 2\log_2 \sum_\gamma \sqrt{p_\gamma q_\gamma} \tag{4}$$

Method based on the Renyi cross entropy to classify a specific traffic can be expressed as follow:

$$\begin{cases} S = I_{0.5}(p,q) = 2\log_2 \sum_i^N \sqrt{p_i q_i} \\ p = (p_0, p_1, \ldots, p_i, \ldots, p_N) \\ q = (q_0, q_1, \ldots, q_i, \ldots, q_N) \\ N = 255 \end{cases} \tag{5}$$

Where *p* is one of the data stream fingerprint samples in the database while *q* is the data stream fingerprint of the traffic which will be classified. We can know from Eq.(5), if the *p* and *q* have the same or similarity data stream fingerprint distributions, the Renyi cross entropy *S* will be equal or close to 0. When the data stream fingerprint distributions of *p* and *q* have the larger difference, then

the Renyi cross entropy *S* is farther from 0. So, we need to set a suitable threshold β. When $|S| \leq \beta$, q is classified into *p*.

4. Experiment

4.1Experimental environment

The experimental environment is powerful computer configured with 4 Intel Core i5-4200U and 4G RAM. In this paper, we developed a frontend module and a backend module based on the Linux platform to implement the traffic classification method based on the data stream fingerprint. The frontend module is used to extracted the data stream fingerprint of network packets. While the backend module is used to calculate the similarity of data stream fingerprint and process the result of classification.

While implementing the traffic classification experiment, firstly we let the frontend module and backend module to enter the learning phase. In this paper, the learning phase time is set to 10 minutes. At the same time, the collected data during the learning phase is used as a training sample set. The set consists of 10425 data stream fingerprints such as dns, ftp, http, https, smtp.

4.2Performance Evaluation

When the learning phase completed, the two modules will come to the testing phase. This phase is mainly used to measure the similarity between the data stream fingerprint of the current packets and the training sample set. In this paper, the threshold of fingerprint similarity is selected as 0.1.

For the experimental purposes, the proposed new method is compared with other methods like Naive Bayes and Support Vector Machine. The experimental results are shown in the Table 1.

Table 1: Performance evaluation

Classification Method	Accuracy (%)	Error (%)	Time (sec)	Using Memory (MB)
Data Stream Fingerprint	91.64	9.85	0.12	62
Naïve Bayes	96.95	9.47	3.5	625
Support Vector Machine	97.76	8.13	11.85	624

Results show that the proposed new traffic classification method is lower than the other two methods in classification accuracy. In the literature[10] , the accuracy of the currently available network traffic classification methods is from 88% to 100%, so the proposed method can be used in real application. The accuracy of the proposed new method is low mainly due to the following two aspects:

Due to the proposed new traffic classification method is mainly applied to the classification of network traffic in public places, it is required that the new method is able to real-time classification of the network traffic. So in the case where the accuracy of the classification results meet available, we are more concerned about the time consumption of the classification process.

In this paper, the frontend module only extracts the data stream fingerprint of the first 500 packets in a session flow. This results in some cases, the extracted data stream fingerprint could not completely reflect the fingerprint feature of an entire session flow, and then decreases the classification accuracy.

5. Conclusion

In this paper, we propose a new traffic classification method based on data stream fingerprint of the packets in a session flow. In the experimental stage, we only extract the data stream fingerprint of the first 500 packets in a session flow as features to classify network traffic. This results in a decline of the classification accuracy. However, the experimental results show that the proposed method has the classification accuracy above 90%. The testing results also verify that the proposed method can

used for the traffic classification. In future works, we will collect the packets in the network by using the Poisson sampling method. In this way, the data stream fingerprint characteristics of the sampled packets can reflect the data stream fingerprint characteristics of an entire session flow. Then the classification accuracy of our proposed new traffic method will be further improved.

Acknowledgment

The research presented in this paper is supported in part by the Science and Technology Research Project of Chongqing Education Committee(KJ1400441), the Technology Support Demonstration Project of Chongqing Education Cloud(cstc2012jcsf-jfzhX0011).

References

[1] Erman J, Mahanti A, Arlitt M. Byte me: a case for byte accuracy in traffic classification[C]//Proceedings of the 3rd annual ACM workshop on Mining network data. ACM, 2007: 35-38.

[2] Moore A W, Papagiannaki K. Toward the accurate identification of network applications[M]//Passive and Active Network Measurement. Springer Berlin Heidelberg, 2005: 41-54.

[3] Mega J, Murata Y, Hirasawa M, et al. Using host profiling to refine statistical application identification[C]// INFOCOM, 2012 Proceedings IEEE. IEEE, 2012:2746-2750.

[4] Constantinou F, Mavrommatis P. Identifying Known and Unknown Peer-to-Peer Traffic[C]// Network Computing and Applications, 2006. NCA 2006. Fifth IEEE International Symposium on. IEEE, 2006:93-102.

[5] Crotti M, Dusi M, Gringoli F, et al. Traffic classification through simple statistical fingerprinting[J]. ACM SIGCOMM Computer Communication Review, 2007, 37(1): 5-16.

[6] Auld T, Moore A W, Gull S F. Bayesian neural networks for internet traffic classification[J]. Neural Networks, IEEE Transactions on, 2007, 18(1): 223-239.

[7] Nguyen T T T, Armitage G. A survey of techniques for internet traffic classification using machine learning[J]. Communications Surveys & Tutorials, IEEE, 2008, 10(4): 56-76.

[8] Tapaswi S, Gupta A S. Flow-Based P2P Network Traffic Classification Using Machine Learning[C]// Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2013 International Conference on. IEEE, 2013:402-406.

[9] Elnaka A M, Mahmoud Q H. Real-time traffic classification for unified communication networks[C]// Mobile and Wireless Networking (MoWNeT), 2013 International Conference on Selected Topics in. 2013:1-6.

[10] Donato W, Pescapé A, Dainotti A. Traffic identification engine: an open platform for traffic classification[J]. Network, IEEE, 2014, 28(2): 56-64.