Journal of Network and Computer Applications xx (xxxx) xxxx–xxxx

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

High performance traffic classification based on message size sequence and distribution

Chun-Nan Lu[a,*], Chun-Ying Huang[a], Ying-Dar Lin[a], Yuan-Cheng Lai[b]

[a] Department of Computer Science, National Chiao Tung University, Hsinchu, Taiwan
[b] Department of Information Management, National Taiwan University of Science and Technology, Taipei, Taiwan

ARTICLE INFO

Keywords:
Traffic classification
Packet size
Message size
Distribution
Sequence

ABSTRACT

Classifying network flows into applications is a fundamental requirement for network administrators. Administrators used to classify network applications by examining transport layer port numbers or application level signatures. However, emerging network applications often send encrypted traffic with randomized port numbers. This makes it challenging to detect and manage network applications. In this paper, we propose two statistics-based solutions, the message size distribution classifier (MSDC) and the message size sequence classifier (MSSC) depending on classification accuracy and real timeliness. The former aims to identify network flows in an accurate manner, while the latter aims to provide a lightweight and real-time solution. The proposed classifiers can be further combined to build a hybrid solution that achieves both good detection accuracy and short response latency. Our numerical results show that the MSDC can make a decision by inspecting less than 300 packets and achieve a high detection accuracy of 99.98%. In contrast, the MSSC classifier can respond by only looking at the very first 15 packets and have a slightly lower accuracy of 94.99%. Our implementations on a commodity personal computer show that running the MSDC, the MSSC, and the hybrid classifier in-line achieves a throughput of 400 Mbps, 800 Mbps, and 723 Mbps, respectively.

1. Introduction

Classifying a network flow into its source application is essential for application-aware network management. By associating network flows with source applications, network administrators can enforce various access control policies to better utilize the available network resources. However, it is not an easy task to correctly classify network flows into the corresponding applications because the use of obfuscation techniques such as port number randomization, payload encryption, and network tunneling. As a result, characterization of Internet traffic has become one of the major challenging issues in communication networks over the past few years (Azzouna and Guillemin, 2003).

A number of approaches have been proposed to classify network flows. The most primitive solution is port-based classification, which builds mappings from transport layer port numbers to applications. For example, map port 53 to DNS flows, port 20 and 21 to FTP flows, and port 25 to SMTP flows. The advantage of this solution is simple. However, it has an obvious flaw because an application is able to bypass the detection by using an unmapped port number or even masquerading an irrelevant well-known port number. One common case is the HTTP-tunneling, which is used to carry non-HTTP network flows over regular HTTP network flows using port 80. Therefore, port-based classification often fails to provide an accurate and reliable solution.

To overcome the drawback of port-based classification, researchers have proposed to detect network flows by finding specific signatures in payloads (Sen et al., 2004). Signature-based classification is considered to be more reliable. However, it did not solve all the issues. First, an application can employ encryption or encapsulation techniques to intentionally obfuscate packet contents; second, this solution requires precise and up-to-date signatures, which might not be applicable for proprietary applications; third, it is computation-intensive to compare characters in each payload against all the available signatures. These unresolved issues pushed research communities to seek for better solutions without inspection payloads.

Many recent approaches classify network flows based on statistical features. These solutions assume that an application would have certain unique statistical properties that can be obtained from empirical data and then used to classify flows into corresponding applications. Common statistical features include the volume, the duration, the burstiness, the payload size, and the jitter of network flows. Statistical-based traffic classification becomes a good alternative because it is

* Corresponding author.
E-mail addresses: cnlu.cs95g@nctu.edu.tw (C.-N. Lu), chuang@cs.nctu.edu.tw (C.-Y. Huang), ydlin@cs.nctu.edu.tw (Y.-D. Lin), laiyc@cs.ntust.edu.tw (Y.-C. Lai).

http://dx.doi.org/10.1016/j.jnca.2016.09.013
Received 29 September 2015; Received in revised form 21 April 2016; Accepted 29 September 2016
Available online xxxx
1084-8045/

possible to classify encrypted or obfuscated network flows.

Roughan et al. (2004) statistically abstracted application features based on application layer protocol attributes and used the features to classify network flows into a specific class-of-service, while Moore and Zuev, 2005 combined statistical analysis with the Bayes theorem to classify network flows. Selected features for the classifiers include the transport layer port number, the flow duration, the packet inter-arrival time, the payload size, and the effective bandwidth. Bernaille et al. (2006) adopted unsupervised clustering techniques to identify an application by using the sizes of the first five data packets of each TCP flow. The solution can make a decision in a pretty short time. However, the solution is sensitive to packet loss and out-of-order delivery.

Other researchers attempt to classify network flows based on observed application behaviors. They monitored and modeled application behavior profiles and then used the profiles to classify flows. Karagiannis et al. (2005) presented BLINC, which analyzed the communication patterns of transport layer host behavior at three levels of details: social, functional, and application, and then used these application features to classify network flows into groups.

However, the classification accuracy directly based on statistical features or observed behaviors are not satisfactory because of sophisticated application behaviors. Network behavior of one application may be similar to that of another application. For example, the behavior of an HTTP file transfer could be similar to that of an FTP transfer. In contrast, not all flows generated by an application behave similar. A BitTorrent client may simultaneously establish flows to retrieve the list of servers, look up resources, check peer status, and exchange files. Making good use of the scattered information can also help classification. Thus, to have a better classification accuracy, an approach, namely message size distribution classifier (MSDC) (Lu et al., 2012), was proposed to classify network flows into sessions and further obtain a complete picture of application behaviors.

MSDC contains two phases, i.e., flow classification and flow grouping. The former classifies network flows into applications by packet size distribution (PSD) and the latter groups related flows as a session by port locality. A flow is identified by the five-tuple information, which includes source IP, destination IP, source port, destination port, and transport layer protocol. When the PSD of one flow is determined, it is compared against the representative of each pre-selected application to decide which application the flow belongs to. Besides, flows are grouped as a session by checking port locality because underlying operation systems often allocate consecutive port numbers for flows of an application. If flows of a session are classified into different applications, an arbitration algorithm based on majority votes is then invoked to make corrections. Evaluations and online benchmarks show that MSDC can obtain accurate results and make a decision by inspecting at most 300 packets and the overall throughput exceeds 400 Mbps on a mainstream computer. Although MSDC can classify network flows accurately, it works in a not-so-fast manner. Therefore, we propose another lightweight and real-time solution called message size sequence classifier (MSSC).

MSSC classifies network flows into applications by message sequences observed during the activities between a pair of two endpoints. The packets exchanged between the two endpoints can be used to derive a sequence based on packet directions and packet sizes. Data exchanged between two endpoints must follow the protocol state machine and the protocol messages defined by involved network applications. MSSC compares the message size sequences (MSSes) of a flow among the representatives of all pre-selected applications to decide which application it belongs to. We also attempted to build a hybrid classifier by combing MSDC and MSSC to provide a balanced solution in terms of classification accuracy and response latency. Based on our analysis and evaluation, MSSC is able to respond by looking only at the very first 15 packets and have a better throughput of 800 Mbps and the hybrid classifier can achieve 723 Mbps.

The rest of this paper is organized as follows. In Section 2, we survey and review relevant researches on network flow classification. Section 3 describes the features that the proposed solutions used to classify network flows. The proposed MSDC and MSSC algorithms are then presented in Section 4. Section 5 provides an analysis for the proposed algorithms. Performance of the proposed solutions is discussed in Section 6. Finally, a conclusion is given in Section 7.

2. Related work

Various statistical-based network flow classification approaches have been proposed in recent years (Gomes et al., 2013). The advantage of these methods is the ability to classify an application without the need to inspect packet payloads. We classify all the approaches into two classes, i.e., the flow-level classification and the session-level classification. The former classifies each flow independently while the latter attempts to group network flows as sessions and then classifies network flows in a session-based manner.

2.1. Flow-level classification

Classifying network flows based on application behaviors is not new. Researchers assume that application behaviors are differentiable and the behaviors can be used to distinguish one application from another. Paxson (1994) modeled and analyzed individual connection characteristics, such as the number of bytes and packets transferred, connection duration, and packet inter-arrival time for different applications. The authors (Este et al., 2009) showed that the amount of information carried by the main packet-level features of Internet traffic flows, such as packet size and inter-arrival time, tends to remain rather constant irrespective of the point of observation and to the capture time.

Hereafter, more works endeavor to classify exclusively network traffic using statistics. They generally consist of two phases: training and classification. A representative model is first built using extracted statistical attributes of flows by learning the inherent structural patterns of datasets and the model is then used to classify network flows. Dewes et al (2003) analyzed and classified different Internet chat traffic using multiple flow characteristics such as flow duration, packet inter-arrival time, packet size, and bytes transferred. Roughan et al. (2004) used nearest neighbor and linear discriminant analysis to map applications to different Quality of Service classes using features such as average packet size, flow duration, bytes per flow, packets per flow, and root mean square packet size. Although Lin et al. (2009) also used the feature of packet size to classify network flows, they paid more attention on those packet sizes with larger proportions in a flow. When the packet size distribution and packet size change cycle of a flow is determined, it is compared against the representatives of all pre-selected applications and the flow is classified as the application having a minimum distance.

Some proposals utilized Machine Learning techniques to classify network traffic. The idea of applying Machine Learning techniques for traffic classification was introduced in (Frank, 1994). Machine Learning techniques are often divided into two phases, i.e., the training phase and the classification phase. Different Machine Learning techniques may perform different and often require distinct parameter configurations.

A number of works adopted probability models to identify and classify network traffic. With training data, probability models are derived for pre-selected applications, and flows are classified as the application having the maximum likelihood. These works assume that the application protocols exhibit consistent and observable structure and patterns in the series of packets they send. Wright et al. (2004) uses the left-right Hidden Markov Models (HMMs) with a large number of states and discrete emission probability distributions to identify TCP connections. Packet sizes and inter-arrival time are