Journal of Network and Computer Applications 35 (2012) 992–1000

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

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

Generating regular expression signatures for network traffic classification in trusted network management

Yu Wang [a,*], Yang Xiang [a], Wanlei Zhou [a], Shunzheng Yu [b]

[a] School of Information Technology, Deakin University, Melbourne, 221 Burwood Highway, Burwood VIC 3125, Australia
[b] Department of Electronic and Communication Engineering, Sun Yat-Sen University, Guangzhou, China

ARTICLE INFO

Article history:
Received 1 August 2010
Received in revised form
12 January 2011
Accepted 9 March 2011
Available online 16 March 2011

Keywords:
Traffic classification
Application signature

ABSTRACT

Network traffic classification is a critical foundation for trusted network management and security systems. Matching application signatures in traffic payload is widely considered to be the most reliable classifying method. However, deriving accurate and efficient signatures for various applications is not a trivial task, for which current practice is mostly manual thus error-prone and of low efficiency. In this paper, we tackle the problem of automatic signature generation. In particular, we focus on generating regular expression signatures with a certain subset of standard syntax rules, which are of sufficient expressive power and compatible with most practical systems. We propose a novel approach that takes as input a labeled training data set and produces a set of signatures for matching the application classes presented in the data. The approach involves four procedures: pre-processing to extract application session payload, tokenization to find common substrings and incorporate position constraints, multiple sequence alignment to find common subsequences, and signature construction to transform the results into regular expressions. A real life full payload traffic trace is used to evaluate the proposed system, and signatures for a range of applications are automatically derived. The results indicate that the signatures are of high quality, and exhibit low false negatives and false positives.

1. Introduction

A range of issues related to trusted network management depend on *network traffic classification and application identification*, which is the process to discover what kind of applications are run by the end users, or more specifically what is the application-layer protocol of a particular network traffic flow. Based on the classification result, *Internet service providers* (ISPs) can deploy Quality of Service (QoS) and application-based auditing; *network administrators* are able to obtain the precise mappings between each application running inside the network and its traffic, so as to make sure that trusted service will get higher priority while others will be restricted; *network intrusion detection systems* (NIDSes) will be enhanced; and *trusted networking operators* will be able to employ the advanced application-specific confidentiality and integrity evaluation schemes.

Among a number of fundamentally different classification schemes, the payload-based signature matching approach is most widely applied in industry. This is because the approach is reliable and only consumes moderate computational power. However, deriving accurate and efficient protocol signatures for various applications is a challenging task. In practice, the signatures are mostly derived by professionals through manual analysis of packet level traces and protocol specifications (if available). The effort is not only highly time-consuming but also error-prone. The first challenge is the lack of publicly available documentations. Although there are standard RFCs for the public-domain protocols, a large number of proprietary protocols are short of open specifications. In addition, some applications have a variety of implementations, some of which do not fully comply with the specifications in the available documentation. Thus the signatures obtained may not span all the variants. Moreover, due to the rapid evolution of network applications, the signatures are also subject to change with time instead of staying fixed. Therefore the high cost manual signature generation process has to be repeated from time to time in order to keep the signatures up to date.

To address the signature generation problem, we propose a novel approach that automatically learns signatures from data. In the work, our method derives regular expression (regexp) signatures that not only provide sufficient flexibility and expressive power but also are widely supported by practical network intrusion detection systems (e.g., Snort and Bro) and traffic classification systems (e.g., l7-filter). In particular, we only consider a certain subset of the standard regexp syntax, including position constraints and choice operator, which are essential for identifying applications but missed

* Corresponding author. Tel.: +61 3 9251 7126; fax: +61 3 9244 6440.
E-mail addresses: y.wang@deakin.edu.au (Y. Wang), yang@deakin.edu.au (Y. Xiang), wanlei@deakin.edu.au (W. Zhou), syu@mail.sysu.edu.cn (S. Yu).

1084-8045/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jnca.2011.03.017

by previous studies of automatic approaches (Byung-Chul et al., 2008; Ye et al., 2009). The algorithm takes a labeled training data set as input and produces regexp signatures for matching the application classes presented in the data. It proceeds in four stages: the first stage is pre-processing where we extract the application session payload in a novel way, the next stage is tokenization, in which we find common substrings and incorporate several advanced features such as choice operators and position constraints, in the third step of multiple sequence alignment, we progressively align the payload byte sequences for each application class, in order to derive the common subsequence that matches all the flows from that class, and the final stage is signature construction where we transform the common subsequence into regular expression. We evaluate the approach using a real life full payload traffic trace and generate signatures for a range of popular applications. The results indicate that the signatures are accurate and close to the handcrafted signatures. The contributions of the work are as follows.

- We have proposed a novel approach to automate the protocol signature generation process. The signatures derived by our method are regular expressions with several essential features missed by previous works, including choice operators and position constraints.
- We have developed an advanced tokenization scheme, which is able to encode some important information of the protocol message format into the tokens.
- We have evaluated our approach with a real-life full-payload traffic trace and generated the signatures for some popular application.

The remainder of this paper is organized as follows. In the next section, we give a brief review of the related work. We continue in Section 3 by introducing the basic concepts and defining the type of signatures considered in this work. In Section 4, we present the details of the proposed signature generation approach. Evaluation methods and results are presented in Section 5, following by a discussion of the scope and limitation of our approach in Section 6. After that, we conclude our paper.

2. Related work

2.1. Network traffic classification

Traffic classification has attracted a lot of research interests recently. In this section we briefly discuss the background by dividing the approaches into four categories according to their bases. The first is the traditional *port-based* method that simply inspects the TCP or UDP port numbers and identifies the application protocols according to the Internet Assigned Numbers Authority (IANA) list of well-known ports and registered ports. The method has been proved to be highly inaccurate nowadays due to the violation of port assignment by more and more emerging applications such as tunneling and peer-to-peer (P2P) (Karagiannis et al., 2004; Sen et al., 2004). The second is *payload-based* method, which achieves better accuracy by performing deep packet inspection. Having the access to packet payload, one can either reconstruct and evaluate the application protocol sessions or simply match signatures in payload data. Early research efforts have focused on deriving hand-crafted signatures. Karagiannis et al. (2004) and Sen et al. (2004) have studied P2P traffic identification using fixed strings. Moore and Papagiannaki (2005) and Moore and Zuev (2005) have combined nine methods to classify a data set by hand. Similar techniques are used in practical tools such as Bro, Snort, Ethereal/Wireshark and l7-filter. The third category is *statistics-based* methods, which has been widely studied in the recent year. The rationale is that the traffic generated by different types of applications exhibits distinct characteristics, which can be automatically found using machine learning techniques. Earlier proposals (Auld et al., 2007; Erman et al., 2006, 2007; Moore and Papagiannaki (2005); Moore and Zuev (2005); Zander et al., 2005) mainly work offline since the flow features are generated from complete flows. This limitation is addressed by some papers (Bernaille et al., 2006; Crotti et al., 2007) that use early sub-flow statistics for real-time classification. The final class consists of other approaches such as the *connection pattern based* classifier (Karagiannis et al., 2005), whose basic idea is to compare the communication pattern of a particular host to the behavior patterns representing different activities/applications. A heuristics based system (Szabo et al., 2007) has been proposed, which integrates various methods such as port based, statistics based, connection pattern based, signature based, and so on. It is widely believed that an accurate traffic classification system will incorporate all sorts of information.

2.2. Automatic signature generation

A number of recent studies have been devoted to automatic application signature generation. Some papers (Haffner et al., 2005; Wang et al., 2010) have proposed using the supervised machine learning models as application signatures, which can be automatically learned from the data sets. Another work (Ma et al., 2006) has proposed three application protocol models and an unsupervised framework to achieve the same goal without the requirement of labeled training data. The main difference between their work and ours is that they generate model-based signatures while we derive explicit string-based signatures. Two systems LASER (Byung-Chul et al., 2008) and AutoSig (Ye et al., 2009) have been proposed to generate substring sequence signatures from labeled data. The former adopts the longest common subsequence (LCS) algorithm and the latter proposes a substring tree structure. Compared to their work, we generate regular expression signatures with the advanced features that are essential for matching application protocols; in addition we develop a sophisticated token extraction method and a token based sequence alignment algorithm.

The problem of automatic worm signature generation has also been well studied. Early systems such as Autograph (Kim and Karp, 2004) have focused on single substring patterns and the recent studies have proposed advanced types of signatures (Li et al., 2006; Newsome et al., 2005; Tang et al., 2009). The most related work to ours is polygraph (Newsome et al., 2005) and the bio-approach (Tang et al., 2009). The former generates substring sequences using suffix tree based token extraction and local alignment with greedy strategy, and the latter generates simplified regular expressions using a modified T-coffee algorithm. Generally speaking, worm identification is a two-class classification problem while general application identification is multi-class. The challenge in worm signature generation is the competition with adversary (attacker) while the challenge in general traffic classification is capturing the diverse nature of applications. As a result, the methods turn out to be quite different as well.

3. Signature generation problem

3.1. Objects and classes

The communication between processes in network hosts is organized into *application sessions*, which are also the basic processing *objects* in the context of traffic classification. Here a