JCSE International Journal of Computer Sciences and Engineering Open Access

Review Paper Vol.-7, Special Issue, 6, March 2019 E-ISSN: 2347-2693

Classification of Network Traffic Based on Zero-Length Packets: A Review

S.Kalpana[1*], T. Raghu Trivedi[2]

[1,2]Dept. of MCA, Sri Padmavathi College of Computer Sciences and Technology, Tiruchanoor-Tirupati, India

[*]*Corresponding Author: kalpanagnana1995@gmail.com, Tel.: +91-7997994522*

DOI: https://doi.org/10.26438/ijcse/v7si6.103105 | Available online at: www.ijcseonline.org

***Abstract* -** Network traffic visitor's classification is fundamental to network management and its performance. However, traditional traffic classifications, which were designed to work on a devoted hardware at very high line rates, may not feature well in digital software-primarily based surroundings. The advised fingerprinting scheme is strong to community conditions which include congestion, fragmentation, put off, retransmissions, duplications, and losses and to various processing abilities. Hence, its overall performance is largely independent of placement and migration problems, and consequently yields an appealing answer for virtualized software program-primarily based environments. We recommend an identical fingerprinting scheme for consumer datagram protocol traffic, which advantages from the equal blessings as the TCP one and attains very excessive accuracy as properly. Results show that our scheme effectively labeled about 97% of the flows on the dataset examined, even on encrypted facts.

***Keywords* -** Network traffic classification, Network monitoring and measurements, Machine learning, Network function virtualization, Software-defined networking.

I. INTRODUCTION

We introduce the sampling and classification scheme which we utilize to classify traffic. The attribute set relies on the observation that data exchange between applications follows a characteristic pattern. This pattern can be utilized to identify the generating application, i.e., different applications generate distinguishable patterns. Let us explain the concept through an illustrated example depicted in, which follows the data exchanged (sent and received APDUs) between a client and a server, running an illustrative application. For example, as can be seen in the first time-line of this figure, from the application layer point of view, the APDUs sent from the client to the server (and vice versa) are essentially typical requests and responses that a client may ask and a server may fulfill. The APDUs themselves are generally affected very little by the network characteristics 2 relies on two main procedures: capture and classification capture and machine learning based classifications

The capture procedure samples data packets traversing the network and stores them for further inspection. To obtain a representative sample set for each flow, complex sampling mechanisms that consume many Content addressable Memory (CAM) filtering rules are required. Even though hardware-based CAM, commonly used in Software-Defined Networking (SDN) switches, is able to search its entire memory space in a single operation, due to its typically narrow table size and lack of support at high rate updates to its rule set, the resulting capturing capabilities are limited. Virtual environments, on the other hand, may use software-based CAM (look-up table) that is able to both stores a large number of rule sets. Typically, an ML algorithm maps flows according to discriminative attributes, then, unknown traffic can be classified, according to the rules that were learnt. The chosen attributes are significant to both classification performance and its complexity. In particular, classifying a large variety of applications may require a large attribute set to attain sufficient accuracy. However, as the size of the attribute set increases, the computational complexity of the classification process increases. To mitigate this issue, a hierarchical approach may be considered, such that at each level, the classifier focuses on a different attribute set. The classification performance under statistical attribute set (e.g., max, min or average packet size) has been examined with various ML algorithms. Note that these attributes can be attained by properly sampled packets from each flow. Yet, since such statistics can be determined only after the flows' termination, it is not suitable for policy enforcements or security functions. In the pioneering work of Bernaille e, the authors argued that some of the commonly used features, such as packets inter-arrival time, are very sensitive to the network load, and hence, should not be used to classify traffic flows. Instead, the authors suggested using the direction and the size of the first packets in a flow to classify flows before termination.

II. RELATED WORK

In this section, Joseph Kampeas describes the previous research is Traffic Classification Based on Zero-Length Packets . Network traffic classification is fundamental to network management and its performance. However, traditional approaches for traffic classification, which were designed to work on a dedicated hardware at very high line rates, may not function well in a virtual software-based environment.

III. PROPOSED SYSTEM

Based on the observations and strategies described in the previous sections, we designed, carried out and tested an automatic classifier, relevant to real time site visitor's category. The device is created from a light-weight two-segment technique, which is appropriate for virtualized switches. In precise, within the first segment, the classifier learns the protocols' conduct from a set of a-APDU sequences retrieved from the 0-period messages of the training information. This mastering segment is carried out offline (i.e., a-priori). In order to label the flows in actual time, the classifier builds and learns a model for each range of a-APDU exchanges, such that at any given time, the classifier holds a label for each energetic flow. Accordingly, the classifier can have some fingerprints for each utility, primarily based on a single a-APDU entry, two entries, three entries, etc. On the one hand, very quick signatures do no longer keep enough statistics to distinguish among applications.

IV. ALGORITHMS

A. Basic Properties of TCP

Even though the Transmission Control Protocol (TCP), which is the prevalent Transport layer protocol for guaranteed in-order data delivery service, is very well known, in this sub-section we provide a brief reminder of some of TCP's features and in particular the Sequence and Acknowledgment numbers (denoted by seq# and ack#, respectively) that we rely on throughout this paper. In order to maintain a reliable and ordered connection between two hosts, TCP utilizes positive acknowledgment messages (ACK), timeouts and retransmissions to guarantee error-free delivery. Accordingly, when a data is not received properly (retransmission timer expires before an ACK is received), the data is retransmitted starting at the not yet acknowledged first byte in the flow. In order to maintain this mechanism, TCP allocates seq# and ack# fields in each TCP segment header. In particular, when data is sent through a TCP connection, these fields indicate which bytes have been sent and correctly received. This is used to ensure that no data is lost on its way to the destination. The receiver utilizes the seq# to reorder the segments as needed. The receiver replies with an ACK informing the sender which data segments have been received with no error. Specifically, the ack# indicates the next seq# the receiver expects to receive from the sender. In this way, both endpoint hosts keep track of the data flow in both directions. It is important to note that a receiver can piggyback an ACK on a data packet that it needs to deliver (i.e., in the forthcoming ACK both the seq# and the ack# can be increased simultaneously, one informing the receiving host the seq# of the sent segment, and the other acknowledging the received data, respectively).

B. Automatic Traffic Classifier

In this paper, we utilize ML and classification algorithms which are already widely used in a variety of applications, including traffic classification. To name a few, uses support vector machines for traffic classification. References give very good overviews of contemporary works using various ML algorithms for traffic classification as well. These methods mainly rely on two core phases. In the first, the classifier is taught how to map a set of attributes (flow-related in the context of this paper) to a set of (traffic) classes. Then, the classifier applies the rules it has learned to map the data. As the focus of this paper is on building a software-based

High-speed traffic classifier, using a novel feature selection process, which creates unique and easy to capture fingerprints from traffic data, and not on the actual classification method, in the sequel we review the J48 decision tree classification method that is used throughout the paper. This method can be efficiently implemented in software. Together with our feature set, this method provides very accurate classification results.

C. J48 Decision Tree:

J48 is an open source version of the C4.5 algorithm. In this method, a decision tree is generated from a set of labeled samples. Generally speaking, when building a decision tree, the set of samples is recursively split, one feature after the other. The key concept behind C4.5 is to maximize the information gain when deciding which feature will dictate the next branching level in the tree. Specifically, assume each labeled sample consists of a p-dimensional vector (x1, x2... xp), where xj represents the jth feature (an attribute), and each sample has a label, y0 or y1. The algorithm computes, for each feature, the conditional entropies given each value of the feature, and subtracts the weighted sum of these entropies from the total entropy of the set. Entropies are calculated on the empirical frequencies the result is the information gain for a feature, and the feature which maximizes this gain is chosen for the next level in the tree.

V. CONCLUSION AND FUTURE SCOPE

we recommended a classification method which samples only zero-length packets in a TCP waft, and is implementable with handiest a unmarried filtering rule. We added a novel fingerprinting mechanism, which expresses the protocols' behavior in a compact and green way, no matter the network parameters or the dimension time and region. Experiments the use of actual visitors showed very encouraging consequences, classifying a big kind of applications with a noticeably small errors ratio.

REFERENCES

[1] B. Pfaff, J. Pettit, T. Koponen, E. J. Jackson, A. Zhou, J. Rajahalme, J. Gross, A. Wang, J. Stringer, P. Shelar et al., "The design and implementation of open vswitch." in NSDI, 2015, pp. 117–130.

[2] M. Rifai, N. Huin, C. Caillouet, F. Giroire, D. Lopez-Pacheco, J. Moulierac, and G. Urvoy-Keller, "Too many sdn rules? compress them with minnie," in Global Communications Conference (GLOBECOM), 2015 IEEE. IEEE, 2015, pp. 1–7.

[3] C. Estan, K. Keys, D. Moore, and G. Varghese, "Building a better netflow," in ACM SIGCOMM Computer Communication Review, vol. 34, no. 4. ACM, 2004, pp. 245–256.

[4] N. Hohn and D. Veitch, "Inverting sampled traffic," in Proceedings of the 3rd ACM SIGCOMM conference on Internet measurement. ACM, 2003, pp. 222–233.

[5] N. Duffield, C. Lund, and M. Thorup, "Learn more, sample less: control of volume and variance in network measurement," Information Theory, IEEE Transactions on, vol. 51, no. 5, pp. 1756–1775, 2005. [6] S. Fernandes, C. Kamienski, J. Kelner, D. Mariz, and D. Sadok, "A stratified traffic sampling methodology for seeing the big picture," Computer Networks, vol. 52, no. 14, pp. 2677–2689, 2008.

[7] S. Zander, T. Nguyen, and G. Armitage, "Sub-flow packet sampling for scalable ml classification of interactive traffic," in Local Computer Networks (LCN), 2012 IEEE 37th Conference on. IEEE, 2012.

[8] N. Katta, O. Alipourfard, J. Rexford, and D. Walker, "Rule-caching algorithms for software-defined networks," Technical report, 2014.

[9] A. Vishnoi, R. Poddar, V. Mann, and S. Bhattacharya, "Effective switch memory management in openflow networks," in Proceedings of the 8th ACM International Conference on Distributed Event-Based Systems. ACM, 2014, pp. 177–188.

[10] M. Kuzniar, P. Pere ´ sˇ´ıni, and D. Kostic, "What you need to know about ´ sdn flow tables," in International Conference on Passive and Active Network Measurement. Springer, 2015, pp. 347–359. [11] Q. Maqbool, S. Ayub, J. Zulfiqar, and A. Shafi, "Virtual TCAM for data center switches," in Network Function Virtualization and Software Defined Network (NFV-SDN), 2015 IEEE Conference on. IEEE, 2015, pp. 61–66.

[12] R. McGeer and P. Yalagandula, "Minimizing rulesets for TCAM implementation," in IEEE INFOCOM, 2009, pp. 1314–1322.

[13] H. Zhu, S. Chen, L. Zhu, H. Li, and X. Chen, "Rangetree: A feature selection algorithm for c4. 5 decision tree," in IEEE 5th International Conference on Intelligent Networking and Collaborative Systems (INCoS), 2013, pp. 17–22.

[14] T. Karagiannis, K. Papagiannaki, and M. Faloutsos, "Blinc: multilevel traffic classification in the dark," in ACM SIGCOMM Computer Communication Review, vol. 35, no. 4. ACM, 2005, pp. 229–240.

[15] M. Tavallaee, W. Lu, and A. A. Ghorbani, "Online classification of network flows," in Communication Networks and Services Research Conference, 2009. CNSR'09. Seventh Annual. IEEE, 2009.

[16] L. Grimaudo, M. Mellia, and E. Baralis, "Hierarchical learning for fine grained internet traffic classification," in Wireless Communications and Mobile Computing Conference (IWCMC), 2012 8th International. IEEE, 2012, pp. 463–468.

[17] Y. Wang, Y. Xiang, and S. Yu, "Internet traffic classification using machine learning: A token-based approach," in Computational Science and Engineering (CSE), 2011 IEEE 14th International Conference on. IEEE, 2011, pp. 285–289.

[18] S. Zander, T. Nguyen, and G. Armitage, "Automated traffic classification and application identification using machine learning," in Local Computer Networks, 2005. 30th Anniversary. The IEEE Conference on. IEEE, 2005, pp. 250–257.

[19] A. Dainotti, W. De Donato, A. Pescape, and P. S. Rossi, "Classification of network traffic via packet-level hidden markov models," in Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE. IEEE, 2008, pp. 1–5.

[20] Y.-D. Lin, C.-N. Lu, Y.-C. Lai, W.-H. Peng, and P.-C. Lin, "Application classification using packet size distribution and port association," Journal of Network and Computer Applications, vol. 32, no. 5, pp. 1023–1030, 2009.

Authors Profile

Ms S.Kalpana has received her graduation degree in BSc. from Sri Surya Degree College, Affiliated to S.V. University , Chittoor, AP in the year of 2013 – 2016. At Present she is Pursuing Post graduate degree MCA, Master of Computer Applications from Sri Padmavathi College of Computer Sciences and Technology Affiliated to Sri Venkateswara University , Tirupati, AP, India.