ATLANTIS PRESS

Advances in Computer Science Research, volume 70

2nd International Conference on Mechatronics Engineering and Information Technology (ICMEIT 2017)

Optimization of a Hybrid Traffic Identification Model Based on DPI

Wenbei Duan [1, a,*], Yuanli Wang [1, b], Xiu Xiong [2, c]

Key Laboratory of Fiber Optic Sensing Technology and Information Processing, Ministry of Education, Wuhan University of Technology, China

[a]duanwbwhut@163.com, [b]2226342553@qq.com, [c]XiongXiu@whut.edu.cn

Keywords: Traffic identification, DPI, Machine Learning.

Abstract. In recent years, Internet Service Providers (ISPs) provide increasingly extensive services, and a large number of applications came into being. These applications provide users a lot of convenience, lead to the network traffic flow increasing, and make the complexity of flow components increasing. In this paper, we design a new DPI module based on OpenDPI. The experiments show that the hybrid model has been achieved the precision of >96%.

1. Introduction

Traffic identification is the primary means of analysing flow in the network. Analysis result is based on various protocols - TCP, UDP, or based on a number of applications - HTTP, P2P. In recent years, the number of network applications is growing rapidly, and the behavior of users on the network has become more and more complicated. Therefore, network traffic identification plays a vital role in network capacity planning engineering and intrusion detection. This paper mainly studies the traffic identification, hoping to improve the correctness of traffic identification, aiming at the existing tight bandwidth. So we design a new DPI module to optimize the OpenDPI.

2. Related work

2.1 Port-Based

The method of using of well-known port number to identify network traffic until 2000 are increasing trend. We can see that the port number-based method was very successful at that time [1]. But, Port-based has some disadvantages. Studies have shown that a new generation of P2P attempts to hide its traffic through the dynamic port number. We can't identify the application of dynamic port number with its port.

2.2 Payload-Based

Deep Packet Inspection (DPI) technology relies on the inspection of the payload of the packet [2]. DPI extracted data characteristics from the traffic, compared with the label experts to provide. This method is very effective for network traffic identification and has a high accuracy rate. However, the study shows that [3] DPI has many shortcomings and limitations. First, if the application label is not updated in a timely manner, the DPI does not recognize new or unknown attacks or applications. So it is necessary to keep up-to-date with the latest label libraries. This creates a great deal of danger because new programs or attacks appear every day. And it is impractical to update the label at any time. If an application is an encryption program, the DPI can't decrypt the load, and the original load can't be obtained. This method does not take effect.

2.3 Machine Learning-Based

In recent years, machine learning based on the Secure Transport Layer Protocol (TLS) has been gradually emerging [4]. This method requires only TCP / IP headers to count features, such as average packet size, stream length, and total number of packets. These statistical features make the classification based on flow characteristics have sufficient effective information to identify.

3. Traffic identification

3.1 Design of traffic identification framework

Considering the advantages of the three methods, we designed a hybrid flow identification system to ensure the efficiency and accuracy of system identification. We design a traffic identification framework as shown in Figure 1.

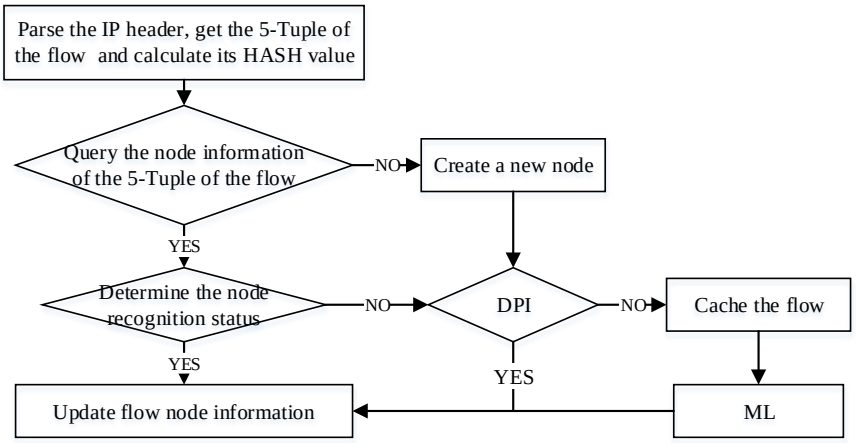

Figure 1. Flow identification framework

Process Description: When processing a message, the key of the Hash is calculated based on the five-tuple information (source IP, destination IP, source port, destination port, transport layer protocol) of the packet. If you do not query the node, the node will be newly created and will be added in the node pool. Then the node directly go into the DPI identification module. Determine whether the node has been identified: 1) if node's status was identified, then we will update the node information; 2) if node's status was not recognized, the node will go in the DPI identification process. DPI identification module is the core of the entire identification component, which is responsible for the accurate and rapid identification of some conventional non-encrypted protocols. After DPI identification, the number of unknown traffic flow will be greatly reduced. Then the flow will be cache, and reduced space-time complexity. Whether or not the flow is identified, it is necessary to update the node's status, then this can save time for subsequent traffic identification. The final identified protocols are presented as protocol numbers.

3.2 DPI recognition module

For OpenDPI can't accurately identify the application of this problem, we designed a new DPI to optimize it.

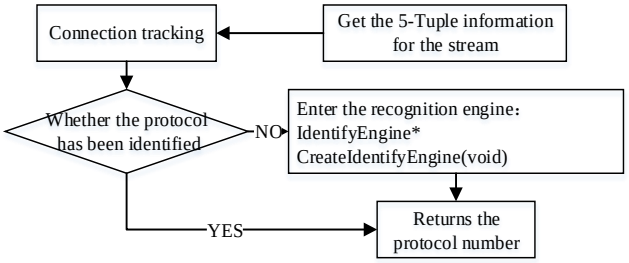

Figure 2. DPI identification

As shown in Figure 2, firstly, we find the node in the node pool according to the five-tuple information of the flow. If the node's status is "identified", we directly return the protocol number. If it is not recognized, it enters the recognition engine. Aiming to improve the efficiency of classification, the idea of classification processing is adopted for all packets, that is, packets are classified according to the transport layer protocol (TCP or UDP) of packets. And according to the characteristics of each agreement to its corresponding identification library on the corresponding categories. For example, the FTP protocol uses TCP packet transmission, so all the features of FTP is on the TCP class detection function. In this way, the FTP protocol may be detected when the transport layer of the packet is

transmitted using the TCP protocol. If the protocol is identified immediately exit the identification process.

3.3 Machine Learning

We choose the best-performing classifier on the basis of the eigenvalues chosen in Document [5]. The data used in the experiment were Cambridge data sets. For the evaluation of the classifier we use the following two metric

1)The precision reflects the proportion of the true positive sample in the positive case judged by the classifier.

$$precision = \frac{TP}{TP + FP} \tag{1}$$

2) The recall rate is the proportion of the correct classification in all the positive examples of the sample.

$$recall = \frac{TP}{TP + FN} \tag{2}$$

We use three classifiers (C4.5, Neural Networks, Naive Bayesian) to verify.

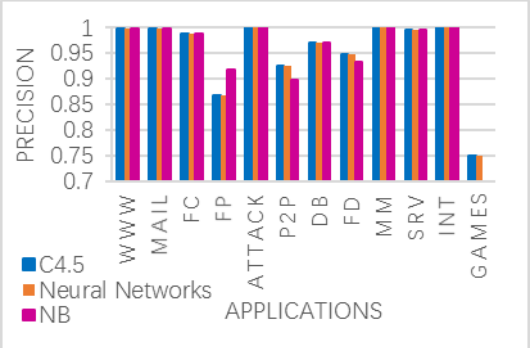

Figure 3. Precision in three classifiers

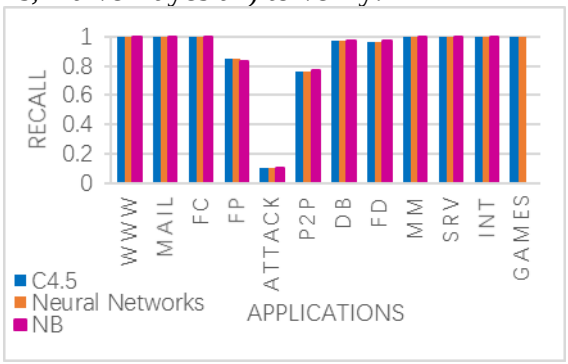

Figure 4. Recall of the three classifiers

As shown in Figure 3, it shows the precision in three classifiers. C4.5 and Neural Network in each application on the precision of the basic flat, but in ATTACK, DB, C4.5 precision is higher. For GAMES, Naive Bayesian precision is zero, while the other two classifiers are only 75% precision. And the precision of the classification of C4.5 and Neural Networks is slightly inferior to that of Naive Bayesian, and the precision of neural networks is the lowest in DB, ATTACK. Throughout the precision of the three classifiers, C4.5 precision is better than Naive Bayesian and Neural Networks. Figure 4 reflects the recall rate of the three classifiers in the dataset, and the C4.5 is essentially the same as the Neural Network. In the GAMES category, Naive Bayesian is zero, which is still due to too few examples of GAMES. But other classifications have recall rate of 100%, indicating that in this data set, C4.5 and Neural Network can class all the GAMES correctly. But if only the precision to measure, there will be a great bias. Comprehensive precision and recall rate of two indicators, in the data set, C4.5 classification effect is better.

4. Experiment Results

We selected four of the most common and better access protocols as training and testing samples. They are HTTP, FTP, BT and PPTV. To ensure the purity of traffic, there is only one protocol in each package. The flow of each protocol was captured by tcpdump. Table 1 show each packet flow statistics.

Table 1. Each packet flow statistics

Protocol	Number of Packet	Number of Flow	Byte
HTTP	180539	7703	168123752
FTP	3790213	24	434160564
BT	423047	946	553676250
PPTV	245332	347	146392334

The collected packets inevitably will contain a small amount of DNS, ICMP, ARP packets and some broadcast messages. But these can be correctly identified by DPI, so this will not affect the results. All the data packets are passed through the OpenDPI module and the hybrid model respectively. Comparison of the precision and recall rate of the hybrid model and the OpenDPI in real time online Table 2.

Table 2. The precision and recall rate of the hybrid model and the OpenDPI

Protocol	Precision		Recall	
	OPenDPI	DPI&ML	OPenDPI	DPI&ML
HTTP	97.2%	97.4%	96.1%	98.3%
FTP	100%	100%	100%	100%
BT	93.5%	94.1%	67%	83.7%
PPTV	95.8%	96.4%	74.9%	87.9%

Figure 5. The precision in real data

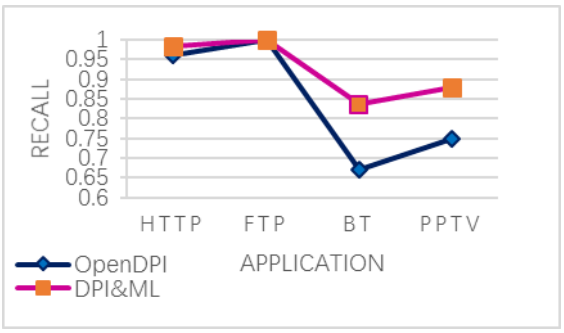

Figure 6. The recall in real data

As shown in Figure 5 and 6, it is clear that in BT and PPTV two categories, the hybrid module precision is higher than OpenDPI. And the hybrid recall rate is higher on all protocols, especially for the BT protocol and PPTV. In summary, the hybrid in the accuracy and recall rate has been significantly improved.

5. Conclusion

In this paper, we mainly design a hybrid traffic identification module, which mainly includes DPI and machine learning. We optimize the OpenDPI mainly for the HTTP protocol identification. The final results show we improve the accuracy of BT and PPTV recognition. The next step is to increase the time of this module, so that it can quickly and accurately identify traffic on-line.

References

[1]. Karagiannis T, Papagiannaki K, Faloutsos M, et al. BLINC: Multilevel traffic classification in the dark [J]. ACM special interest group on data communication. 2005(35): 229-240.
[2]. Haffner P, Sen S, Spatscheck O, et al. Proceedings of the 2005 ACM SIGCOMM workshop on Mining network data[C]. USA: ACM, 2005: 197-202.
[3]. Auld T, Moore A W, Gull S F, et al. Bayesian Neural Networks for Internet Traffic Classification [J]. IEEE Transactions on Neural Networks, 2007(18): 223-239.
[4]. Kim H, Claffy K, Fomenkov M, et al. Proceedings of the ACM CoNEXT Conference[C]. USA: ACM, 2008: 11-23.
[5]. Hongli Zhang, Gang Lu, et al. Feature selection for optimizing traffic classification [J]. Computer Communications. 2012(35): 1457-1471.