Boosting statistical application identification by flow correlation

Mohamad Jaber, Roberto G. Cascella and Chadi Barakat
INRIA - France
Email: {mohamad.jaber, roberto.cascella, chadi.barakat}@inria.fr

***Abstract*—In this paper, we propose a new online method for traffic classification that combines the statistical and host-based approaches in order to construct a robust and precise method for early Internet traffic identification. We use the packet size as the main feature for the classification and we benefit from the traffic profile of the host (i.e., which application and how much) to decide in favor of this or that application. This profile is updated online based on the result of the classification of previous flows originated by or addressed to the same host. We evaluate our method on real traces using several applications. The results show that leveraging the traffic pattern of the host ameliorates the performance of statistical methods. They also prove the capacity of our solution to derive profiles for the traffic of Internet hosts and to identify the services they provide.**

I. Introduction

The identification of Internet traffic applications is very important for ISPs and network administrators to protect their resources from unwanted traffic and prioritize some major applications. Statistical methods [1]–[3] are preferred to port-based ones and deep packet inspection (DPI) since they also work for dynamic port numbers and encrypted traffic. These methods combine the statistical analysis of the application packet flow parameters, such as packet size and inter-packet time, with machine learning techniques. Other approaches [4] rely on the way the hosts communicate and their traffic patterns to identify applications.

In this paper we develop a new online method that combines the statistical properties of a flow with the traffic profile of the end-points to construct a robust and precise method for early Internet traffic identification. First, we define the host profile and we determine the host-based probability that a flow is of a given application in both the incoming and outgoing direction. Then, we show how to use these profiles later as an initial guess before the classification of future flows. The host profiles are updated after each classification using an exponential weighted moving average filter to absorb any transient behaviour; the way the profile accounts for past classified flows depends on some discounting parameter, which can be decided by the network administrator. Finally, we use two real traces to test our method and to show how to characterise the traffic pattern of each host in the traces.

II. Method Description

The novelty of our approach consists of using this traffic pattern to predict future flows that involve the same host. In this section, we first discuss how a monitor computes the probability that a flow of packets between two hosts is of a certain application solely using the traffic patterns of these hosts. Then we discuss our iterative classification of the flows for each packet size independently. Each flow corresponds to a sequence of N packets Pkt_k, where k indicates the position of the packet in the flow independently of its direction.

Let F denote a function that associates a packet flow between a source S and destination D to an application $A(i)$, with $1 \leq i \leq N_A$ and N_A the number of monitored applications. Let $P(F_S = A_S|S)$ (or $P(F_D = A_D|D)$) be the probability that, given the host traffic profile, the flow is of an application A_S for the source (or A_D for the destination). Then, the probability $P(F = A(i))$ that the flow is of application $A(i)$ is computed as follows:

$$\begin{aligned} P(F = A(i)) &= P(F_S = A_S \cap F_D = A_D | A_S = A_D) \\ &= \frac{P(F_S = A(i)|S) * P(F_D = A(i)|D)}{\sum_{j=1}^{N_A} P(F_S = A(j)|S) * P(F_D = A(j)|D)} \end{aligned}$$

We compute the probability by considering the cases when the prediction for each host is in accordance by considering the traffic profiles of S and D separately. The equation also holds when the monitor only records the traffic profile of one of the two hosts. In fact, if we assume a uniform probability for the other host, e.g., $P(F_D = A_D|D) = \frac{1}{N_A}$, then, the equation simplifies to $P(F = A(i)) = P(F_S = A(i)|S)$. This host-based probability $P(F = A(i))$ is then used our statistical classification method. The method detailed in [3]. consists of three main phases: the model building, the classification, and the application probability or labeling.

The model building phase consists of constructing the sets of classes (clusters) by using a training data set. In this phase we compute $P(C(j)|A(i))$, i.e., the per-class probability, knowing the application $A(i)$. In the classification phase, each flow is affected in a class among the classes of the training data set according to the similarity based in the Euclidian distance. Finally, the labelling phase consists of assigning a flow to an application. In this last phase, we combine iteratively the results of the classification for each single packet size and we calculate the probability ($P(A(i))$ that a flow belongs to an application $A(i)$ given the prediction from the host profiles and the classification results of the first N packet sizes (i.e., class $C(j(1))$ for the first packet size, class $C(j(2))$ for the second packet size and so on).

TABLE I: Traces Description

Source and Date	Application	training	testing
Brescia University April 2006 [2]	HTTP	8000	17,263
	SMTP	8000	19,835
	POP3	8000	19,935
Brescia University Fall 2009 [5]	HTTP	500	30422
	HTTPS	500	3608
	EDONKEY	500	3702
	BITTORENT	500	3608

$$P(A(i)) = P(A(i)|Result \cap P(F = A(i))) = \frac{P(F = A(i)) * \prod_{k=1}^{N} P(C(j(k))|A(i))}{\sum_{i=1}^{N_A}[P(F = A(i)) * \prod_{k=1}^{N} P(C(j(k))|(A(i))]}$$

$P(F = A(i))$ is the probability that a flow between a source and a destination comes from application $A(i)$ based on their traffic profiles. $P(C(j(k))|A(i))$ is the probability that Pkt_k of a flow belongs to the class $C(i)$ knowing the application $A(i)$. N_A is the total number of applications.

The prior distribution is updated after each classification of a new collected flow. Let $P_{(n-1)}(A(i))$ be the prior probability for application $A(i)$ computed from the past $(n-1)$ flows that the monitor affects to the application $A(i)$ with probability $P(F = A(i))$ for each application, then the posterior probability for each application is:

$$P_{(n)}(A(i)) = \lambda * P_{(n-1)}(A(i)) + (1 - \lambda) * P(F_n = A(i))$$

$P(F_(n) = A(i))$ is the result of the classification of flow n and λ, $0 \leq \lambda \leq 1$, represents the discounting factor for past classifications. When λ is close to 0, the profile is computed by associating a higher weight to the most recent flows. When λ is close to 1 the profile is calculated over a longer period, which means that the profile is determined in equal measure by all previous classified flows. When $\lambda = 1$ the profile corresponds to the initial prior distribution, which in our case assigns a uniform probability to all applications.

The traffic profile of a host is defined based on type of previous flows. This requires that the monitor collects statistical information about a flow, classifies the flow, and stores the result of the classification to track the activity of a host. The traffic profile, so computed, gives an indication of the preferred applications that run at the host.

III. Experimental results

We use two real traces to validate our method, see Table I for details. We define the *Precision* as the ratio of flows that are correctly assigned to an application, $TP/(TP+FP)$, and the overall precision is the weighted average over all applications given the number of flows per application. Fig. 1 and 2 plot the total precision of trace I and trace II versus the number of packets used for the classification respectively. The different lines in the plot correspond to the precision of the classifier when different values of the discounting factor λ are used.

For Trace I and II and for all the selections of λ, we have better performance compared to the classification without host profile information ($\lambda = 1$). For Trace I, we can observe in Fig. 1 that a value of $\lambda = 0.9$ gives the best performance for the classifier. We obtain a precision of 94% already after two

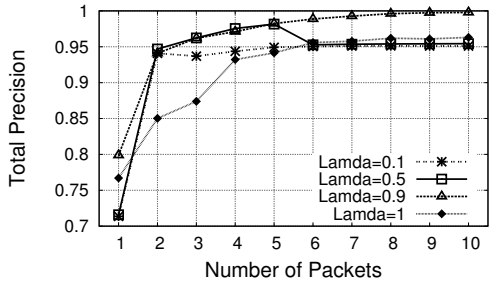

Fig. 1: Total precision versus the number of packets (Trace I)

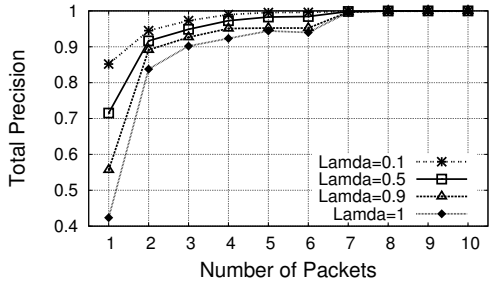

Fig. 2: Total precision versus the number of packets (Trace II)

packets, 97% after four packets and 99.9% when 10 packets are used for the classification. For Trace II We can observe in Fig. 2 that a small value of λ increases the precision already after the first packet. However, large values of λ require more packets to classify correctly the flows. The precision for all values of λ converges to 99.99% after 7 packets. These results show that the profile of the host gives an early characterisation of a flow because of the traffic pattern of the host. For instance, we can consider that a host that is browsing the web is more prone to have a sequence of HTTP connections.

IV. Conclusion

In this paper we present our new method for Internet traffic identification that combines the statistical and host-based approaches. The statistical parameters that we use are the size and direction of the first N packets. The novelty of our approach consists in leveraging the host profile to refine the classification. First we define the profile of the host. Then we show how the profiles of the source and destination hosts are used to assign a prediction probability to the new flow.

We evaluate our solution on two real traces and we profile the hosts with the same IP prefix. We test our method for different values of the discounting factor λ and discuss the optimal choice based on the traffic pattern of the host. The results show a great improvement for the classification of applications when the host profile is used. In particular, the classifier reaches a precision of 0.99.

References

[1] A. Moore and D. Zuev, "Internet traffic classification using bayesian analysis techniques," in *ACM Sigmetrics*, 2005.

[2] M. Crotti, M. Dusi, F. Gringoli, and L. Salgarelli, "Traffic classification through simple statistical fingerprinting," in *ACM CCR*, vol. 37, pp. 5–16.

[3] M. Jaber and C. Barakat, "Enhancing application identification by means of sequential testing," in *NETWORKING*, Aachen, Germany, 2009.

[4] T. Karagiannis, K. Papagiannaki, and M. Faloutsos, "Blinc: Multilevel traffic classification in the dark," in *ACM SIGCOMM*, 2005.

[5] T. II, "Brescia university," http://www.ing.unibs.it/ntw/tools/traces/, 2009.