A Traffic Grouping Method using the Correlation Model of Network Flow

Young-Hoon Goo[1], Sung-Ho Lee[1], Seongyun Choi[2], Mi-Jung Choi[2] and Myung-Sup Kim[1]
Dept. of Computer and Information Science, Korea University, Sejong, Korea[1]
Department of Computer Science, Kangwon National University, Chuncheon, Korea[2]
{gyh0808, gaek5,tmskim}@korea.ac.kr[1], {seongyun, mjchoi}@kangwon.ac.kr[2]

Abstract— **Emergence of high-speed Internet and ubiquitous environment has led to a rapid increase of applications on the Internet and network traffic complexity. In order to provide reliable services and efficient management of network resources, it is essential to classify traffic with specific units. While various traffic classification methods are being studied, there is no single method to classify traffic completely yet. In this paper, we define the correlation model of network flow and propose a traffic grouping method based on it. The proposed correlation model of network flow for traffic grouping consists of the Similarity model and the Connectivity model. We define the Similarity model guideline and the Connectivity model guideline for the purpose of applying the proposed method effectively. By applying the proposed method to the actual application traffic classification, we demonstrate that the method has high accuracy and completeness.**

Keywords—traffic grouping; correlation model of network flow; Similarity model; Connectivity model; guideline of application

I. Introduction

Today's emergence of high-speed Internet and ubiquitous environment has led to a rapid increase of developed applications on the Internet and more network traffic complexity. In this situation, network traffic monitoring and analysis is essential for effective network management and stable service provision. The precise application-specific classification of Internet traffic must be preceded for reacting a various traffic analysis needs.

As a result, a lot of researches are being done to provide reliable services to users and maximize the utilization of network resources. In order to achieve this, a method that can accurately classify various applications traffic is necessary.

While various traffic classification methods are being studied, there is no single method to classify traffic completely yet. First, the header signature-based classification method [1] is not reliable for applications using various protocols or providing a function of setting two or more port numbers or using dynamic port numbers. The payload signature-based classification method [2] is the highest-performing method in terms of completeness and accuracy due to inspecting the packet payload directly. However, it cannot respond quickly to change of the application because the signature extraction work is manual work, and it requires a lot of time and expertise. Also, there may be a significant difference in the quality of the signatures depending on the ability of the network manager who extract signatures. The major drawback of this method is that it cannot classify encrypted traffic. The statistical signature-based classification method[3] overcomes some of the difficulties of the payload-based method. However, when using statistical information, there is a high probability that signatures will be generated that depend on the communication engine or application-layer protocol used by a particular application. All three of the above-mentioned classification methods only analyze flows matched with the signature, therefore a large number of signatures must be used in order to prevent a large number of false negative. This causes an increase in processing time. In the case of machine learning-based traffic classification methods [4,5], accuracy is strongly influenced by a large amount of sample data, and it is difficult to classify the application which is using the same application-layer protocol.

In this paper, we define the correlation model of network flow and propose a traffic grouping method based on it. The proposed method automatically calculates the correlation index and groups similar flows, hence classification speed is fast and encrypted traffic analysis is possible and it is possible to maximize completeness by continuative grouping.

The rest of this paper is organized as follows. Section II describes the proposed method. Section III describes the effective way to applying the proposed method. Section IV demonstrate the validity of proposed method through experiments. Finally, Section V presents conclusive remarks and a brief look for the future research directions.

II. Traffic Grouping Method Using the Correlation Model of Network Flow

This section describes the proposed method. The correlation model of the network flow consists largely of the Similarity model and the Connectivity model. Figure 1 shows the traffic grouping process of the proposed method.

First, the user selects some of the flows of the target of detection(application or malicious behavior) to be classified in the entire traffic as the SeedGroup(Group 0), and inputs SeedGroup and the entire traffic into the Similarity model. The SeedGroup can be selected through information obtained from

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (No. 2017-0-00513, No. R7117-17-0214)

978-1-5386-1101-2/17/$31.00 ©2017 IEEE

IDS or IPS alerts for malicious behavior or various signatures for applications. Next, in the Similarity model, the similarity index between the SeedGroup and the entire traffic is calculated, and the traffic having the similarity index exceeding a certain threshold value is grouped to output Group 1. Then, Group 1 is used as an input to the Connectivity model to calculate the connectivity index between Group 1 and non-grouped traffic, and the traffic having the connectivity index exceeding a certain threshold value is grouped to output Group 2. The user repeatedly inputs the corresponding Group i into the connectivity model until it is no longer grouped. Finally, the SeedGroup (Group 0) and the traffic from Group 1 to Group N are output as classification results.

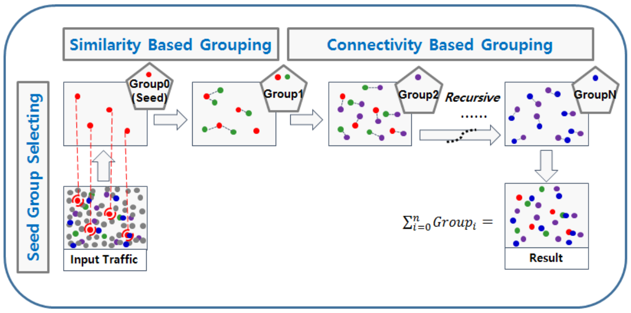

Fig. 1. Process of traffic grouping using the flow correlation

The Similarity index between two flows is calculated as a Euclidean Distance value by constructing a vector of statistical features of the flows. It is important to choose which statistical features are used to calculate the similarity index between flows so that only certain applications can be detected without duplicate detection of other applications. For this purpose, it may be necessary to increase the number of statistical feature used to distinguish only a specific application. Thus, we applied the various statistical feature set of flows for grouping certain application traffic without redundant detection. Finally, we select 21 statistical features of flow as attributes of the Similarity model. The 21 statistical features used are the value of maximum, minimum, average, and standard deviation of following four items and payload sizes considering the directionality of the first five packets. The four items above-mentioned are the inter-arrival time of all the packets in the flow, the inter-arrival time of the first 16 packets in the flow, payload size of all packets in the flow, payload size of the first 5 packets in the flow. In the case of TCP, the payload size considering directionality means a value obtained by multiplying the payload size by +1 in the packet transmitted from the client to the server, multiplied by -1 in the opposite case. In the case of UDP, the direction of the first packet is defined as the + direction. Equation 1 is an equation for calculating the similarity index between the flow f_x and the flow f_y.

$$Sim(f_x, f_y) = 1 - \frac{Euclidean\ Distance}{\sqrt{21}}$$
$$= 1 - \frac{\sqrt{\sum_{i=1}^{21}\left(a_i(f_x) - a_i(f_y)\right)^2}}{\sqrt{21}}$$

Eqs. 2. Calculation of the similarity index between flows

a_i denotes each attribute, and Min-Max normalization is applied to each attribute to equalize the scale of the value of each attribute, so the scale of value is adjusted to a value between 0 and 1. The calculated Euclidean Distance is again expressed as a value between 0 and 1 through Min-Max normalization, and the corresponding value is subtracted from 1 to express the degree of similarity between two flows. Finally, the similarity index between flows is expressed as a value between 0 and 1, and the more similar the two flows are, the closer this value to 1 is. In the Similarity model, the similarity index between the SeedGroup and the entire traffic is calculated, and the flows having the value exceeding a certain threshold are grouped.

The connectivity index between two flows is calculated using 4 attributes. These are the time of occurrence of flow, source IP address and destination IP address pair, source port and destination port pair, and Transport-Layer protocol. Equation 2 is an equation for calculating the connectivity index between the flow f_x and the flow f_y.

$$Conn(f_x, f_y) = \sum_{i=1}^{4}\left(w_i \times a_i(f_x, f_y)\right)$$
$$, where\ 0 \leq\ a_i(f_x, f_y) \leq 1, \sum_{i=1}^{4} w_i = 1$$

Eqs. 2. Calculation of the connectivity index between flows

$a_i(f_x, f_y)$ denotes the calculated value of each sub-connectivity index of each attribute of the Connectivity model, and w_i denotes the weight of a_i. The sum of the weights of each attribute is 1 and the weights of the attributes can be adjusted in consideration of the fact that the characteristics of the traffic generated according to the types of the targets to be detected may be different. The classification performance can be maximized by adjusting the weight according to the target.

$$a_{ST}(f_x, f_y) = 1 - \sqrt{\frac{dist(f_x, f_y)}{maxdist(F)}}$$

Eqs. 3. Calculation of the connectivity index of start time of flows

Equation 3 calculates the connectivity index of the start time between flows. $dist(f_x, f_y)$ denotes the start time difference of the flow, and $maxdist(F)$ denotes the maximum value among the start time differences of all flows of the target traffic. The meaning of Equation 3 is the value between 0 and 1 through the min-max normalization with the difference between the maximum and minimum values of start time of all flows and the start time difference of f_x and f_y. Also, because if the start time difference is too large for the flows of same detection target, there is the possibility of being misidentified as another application, it is finally expressed as a value between 0 and 1 through the square root to mitigate this problem. This value means that the closer to 0 means that the two flows are not similar, the closer to 1 means that the two flows are similar.

$$a_{IP}(f_x, f_y) = 1 - \left(\frac{prefixlenIP_{src}(f_x, f_y)}{32}\right)^2$$
$$+ \left(\frac{prefixlenIP_{dst}(f_x, f_y)}{32}\right)^2$$

Eqs. 4. Calculation of the connectivity index of the pairs of IP addresses of flows

Equation 4 calculates the connectivity index of the pair of source and destination IP addresses. In equation 4, $prefixlenIP_{src}(f_x, f_y)$ denotes the larger value of the number of the same prefix bits of the source IP address of f_x and the source IP address of f_y and the same prefix bits of the source IP address of f_x and the destination IP address of f_y. $prefixlenIP_{dst}(f_x, f_y)$ denotes the larger value of the number of the same prefix bits of the destination IP address of f_x and the destination IP address of f_y and the same prefix bits of the destination IP address of f_x and the source IP address of f_y. The result of the equation is a value between 0 and 1, and the closer to 0, the less similar the two flows, the closer to 1, the more similar the two flows.

$$a_{Port}(f_x, f_y) = 1 - \left(\frac{prefixlenPort_{src}(f_x, f_y)}{16}\right)^2 + \left(\frac{prefixlenPort_{dst}(f_x, f_y)}{16}\right)^2$$

Eqs. 5. Calculation of the connectivity index of the pairs of source port and destination port of flows

Equation 5 calculates the connectivity index of the pair of source and destination port. In equation 5, $prefixlenPort_{src}(f_x, f_y)$ denotes the larger value of the number of the same prefix bits of the source port of f_x and the source port of f_y and the same prefix bits of the source port of f_x and the destination port of f_y . $prefixlenIP_{dst}(f_x, f_y)$ denotes the larger value of the number of the same prefix bits of the destination port of f_x and the destination port of f_y and the same prefix bits of the destination port of f_x and the source port of f_y. The result of the equation is a value between 0 and 1, and the closer to 0, the less similar the two flows are, the closer to 1, the more similar the two flows. It is noted that the ports of flows that occur when an application is executed tend to have incremental and use a similar range of ports.

$$a_{PROT}(f_x, f_y) = \begin{cases} 0 : & f_x.PROT \neq f_y.PROT \\ 1 : & f_x.PROT = f_y.PROT \end{cases}$$

Eqs. 6. Calculation of the connectivity index of protocol of flows

Equation 6 calculates the connectivity index of protocol. If the protocol is the same, the value of $a_{PROT}(f_x, f_y)$ is 1, otherwise $a_{PROT}(f_x, f_y)$ is 0.

Finally, the connectivity index between two flows is expressed as a value between 0 and 1 by sum of multiplying the sub-connectivity index of the four attributes by the weights, as shown in Equation 2, and has a value close to 1 as the two flows become more similar.

In the Connectivity model, the Group 2 is generated by grouping non-grouped flows which have the connectivity index, that is equal to a certain threshold or more, by calculating the connectivity with Group 1, which is grouped in the Similarity model. The Group 2 is inputted into the Connectivity model to continuously group the traffic to be detected. Finally, it detects the target by calculating the connectivity with non-grouped flow repeatedly until it is no longer grouped.

III. Effective Way to Applying The Proposed Method

The Similarity model is similar to the statistical signature based classification method in that statistical information is used. The Connectivity model is similar to the header signature based classification method in that header information and flow occurrence time are used. However, the proposed method can maximize the completeness by using the Similarity model as the first step and the using the Connectivity model recursively in a cascade about remaining traffic that cannot be grouped by before step. Also, it can group the encrypted traffic. The Connectivity model groups flows similar to the header information of the target flow through calculating of the connectivity index, rather than grouping only the same IP, port, and protocol flow as the header signature. Our method can overcome the limitation of port based classification method because the method uses the similarity model as the first step. In addition, since it is not a signature-based classification method, there is no need to generate a signature. It is only need some flows to be used SeedGroup. If one flow to be used as SeedGroup is set as an input, the SeedGroup is used as the first clue of the target traffic classification, and the target grouping is performed. The flows to be used as SeedGroup can be selected through information obtained from IDS or IPS alerts for malicious behavior or various signatures for applications.

In order to apply proposed method effectively, we defined the guideline of correlation for target of detection. It is composed of the Similarity model guideline and the Connectivity model guideline. These guidelines should be constructed for each target of detection, and we can classify the target of detection by referring to them for high completeness and accuracy. It is also possible to classify traffic through this method, even when a user is not a skilled expert, such as a network administrator, when constructing a guideline of target of detection.

The Similarity model guideline for target of detection is a sophisticated threshold value that must be able to make accuracy to 100% even if any arbitrary of all ground-truth flows is used as SeedGroup. The Connectivity model guideline for target of detection is the set of weights of four attributes and threshold of the Connectivity model that reflect the characteristics of target of detection well. The reasons that such guidelines are defined are as follows.

First, in the Similarity model, the grouped flows are used as the inputs of the Connectivity model, and the Connectivity model is used recursively for grouping in a cascade. If there is a small number of false positives in the Similarity model, the false positives increase rapidly depending on the number of repetitions in the cascading Connectivity model. Therefore, in the Similarity model must achieve accuracy of 100%. For achieving this, we should set a sophisticated threshold value which must be able to make false positive is 0 even if a certain flow is used as SeedGroup for each detection target. Second, the Connectivity model can improve the accuracy and the completeness by making it possible to adjust the proper set of weights and threshold for each target of detection in consideration of the characteristics of the traffic generated depending on the type of application or attack.

IV. Experiment and Result

In this section, we demonstrate through experiments that this method has hgh accuracy and completeness, and that the guidelines for the correlation model for each target of detection clearly exist and can be found.

The experimental environment is as follows. First, we collected the ground-truth traffic and noise traffic of the target to be detected from several hosts. In this experiment, Torrent, Dropbox, Facebook, Kakaotalk, and Daumpot were selected as representatives of P2P application, file sharing application, SNS application, instant messenger application, and video streaming application respectively. We collected the ground-truth traffic and noise traffic for each application from 4 hosts.

To find the Similarity model guideline, do the following. First, select one of the ground-truth flows as Seed and group the target of detection using the Similarity model by increasing the threshold of the Similarity model from 0 to 1 by 0.0001. Then check the threshold at which false positive becomes zero in the results. This value is defined as "Proper Threshold". Repeat the above procedure by selecting all the ground-truth flows one by one as SeedGroup, and then check the maximum threshold values which made false positive is zero for all SeedGroups in each step. It is defined as "Maximum Proper Threshold". As shown in Figure 2, when threshold is set to 0, grouping based on the Similarity model, all flows are grouped, so the amount of true positive and false positive are both 100%. Also, as the threshold value approaches 1, true positive and false Positive decrease gradually, but false positive reaches 0 before true positive. In other words, the higher the threshold, the lower the completeness, but the higher the accuracy. Therefore, the Similarity model guideline should use the Maximum Proper Threshold which is the maximum value of the Proper Thresholds. This Maximum Proper Threshold guarantees 100% accuracy even if any arbitrary flow selected as SeedGroup. As the experiment is conducted, the Similarity model guidelines converge to higher values and become robust. Figure 3 shows the algorithm for finding the Maximum Proper Threshold for each experiment.

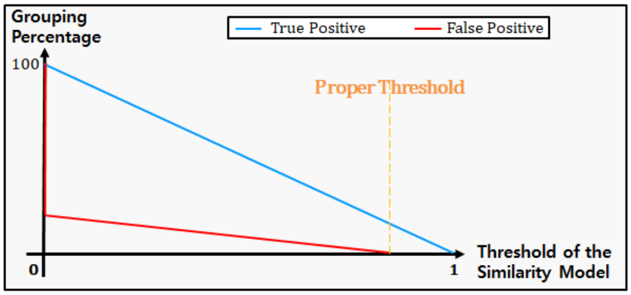

Fig. 2. True Positive and False Positive distribution of grouping result according to threshold change

To find the Connectivity model guidelines, we do the following. Firstly, we group the target of detection using the Similarity model with the Maximum Proper Threshold, then input the result of grouping into the Connectivity model. Next, we group the non-grouped flows using the Connectivity model with the all cases of combinations weights of attributes and the threshold value of the Connectivity model, and check the results in all cases. The total number of these combinations is 84. In the result, we identify the combination with the highest completeness and accuracy of 100%. This combination is the guideline of the Connectivity model and converges to a common value as the experiment is performed on the same detection target.

```
Find-Maximum-Proper-Threshold(n)
  Proper Threshold[n] = { 0 }
  Maximum Proper Threshold = 0
  S.I threshold = 0
  for i = 1 to n
      Set Flow ID i to SeedGroup
      for j = 0 to 1,  j += 0.0001
          S.I threshold = j
          Grouping using the Similarity model and check the False Positive
          if False Positive is 0
              Proper threshold[i] = S.I threshold
              break;
      if Maximum Proper Threshold < Proper threshold[i]
         Maximum Proper Threshold = Proper Threshold[i];
```

Fig. 3. Algorithm for finding the Maximum Proper Threshold

This experiment is conducted four times for each application using different traces to confirm that there are robust guidelines of Similarity model and Connectivity model for each application.

Table 1 shows the traffic information used in the experiment.

TABLE I. Traffic Information of the Experiment

Experiment		Flow	Experiment		Flow
1	Torrent	582	3	Torrent	732
	Noise	3189		Noise	2669
2	Torrent	924	4	Torrent	579
	Noise	2573		Noise	2486
1	Dropbox	39	3	Dropbox	44
	Noise	3624		Noise	4200
2	Dropbox	45	4	Dropbox	39
	Noise	4277		Noise	4791
1	Facebook	136	3	Facebook	158
	Noise	1554		Noise	2788
2	Facebook	132	4	Facebook	127
	Noise	776		Noise	3017
1	Kakaotalk	38	3	Kakaotalk	18
	Noise	2756		Noise	3432
2	Kakaotalk	20	4	Kakaotalk	26
	Noise	4787		Noise	5186
1	Daumpot	1082	3	Daumpot	778

	Noise	3922		**Noise**	4610
2	**Daumpot**	1099	**4**	**Daumpot**	1021
	Noise	1619		**Noise**	739

Table 2 shows the experimental results. Column 1 is the experiment number. Column 2 is the largest of all the thresholds where false positive becomes zero for all ground-truth flows in the Similarity model. Columns 3 to 7 represent the set of weights and threshold of the Connectivity model, and column 8 is percentage of true positive, column 9 means percentage of false positive.

TABLE II. EXPERIMENTAL RESULT

Ex		Sim. threshold	Con. threshold	Weight. IP	Weight. Port	Weight. Prot	Weight. Time	True Positive	False Positive
Torrent	**1**	0.996	0.6	0.2	0.4	0.1	0.3	93.0%	0%
	2	0.9759	0.6	0.2	0.4	0.1	0.3	99.2%	0%
	3	0.9949	0.6	0.2	0.4	0.1	0.3	99.8%	0%
	4	0.9870	0.6	0.2	0.4	0.1	0.3	95.4%	0%
Dropbox	**1**	0.9759	0.9	0.1	0.1	0.4	0.4	100%	0%
	2	0.9696	0.9	0.1	0.1	0.4	0.4	84.2%	0%
	3	0.973	0.9	0.1	0.1	0.4	0.4	100%	0%
	4	0.958	0.9	0.1	0.1	0.4	0.4	100%	0%
Facebook	**1**	0.9964	0.8	0.1	0.2	0.5	0.2	89.3%	0%
	2	0.9913	0.8	0.1	0.2	0.5	0.2	83.2%	0%
	3	0.9947	0.8	0.1	0.2	0.5	0.2	96.0%	0%
	4	0.9971	0.8	0.1	0.2	0.5	0.2	98.0%	0%
Kakaotalk	**1**	0.9898	0.7	0.2	0.2	0.1	0.5	95.0%	0%
	2	0.9753	0.7	0.2	0.2	0.1	0.5	87.5%	0%
	3	0.9744	0.7	0.2	0.2	0.1	0.5	100%	0%
	4	0.9756	0.7	0.2	0.2	0.1	0.5	95.7%	0%
Daumpot	**1**	0.987	0.5	0.1	0.5	0.2	0.2	90.7%	0%
	2	0.9959	0.5	0.1	0.5	0.2	0.2	83.5%	0%
	3	0.9898	0.5	0.1	0.5	0.2	0.2	86.8%	0%
	4	0.9955	0.5	0.1	0.5	0.2	0.2	86.6%	0%

From the first row of the table, it is the result of 4 lines each for Torrent, Dropbox, Facebook, Kakaotalk, and Daumpot.

For the result, it was confirmed that the threshold of the Similarity model of 100% accuracy for Torrent is 0.996. Thus, the Similarity model guideline for Torrent is 0.996. As a result of 4 torrent experiments, we can say there exist the definite Connectivity model guideline for Torrent, because the set of threshold and weights of the Connectivity model, which maximizes completeness and make accuracy to 100%, are constant at {0.6, 0.2, 0.4, 0.1, 0.3}, respectively. Likewise for Dropbox, the Similarity model guideline is 0.9759, and the Connectivity model guideline is {0.9, 0.1, 0.1, 0.4, 0.4}. For Facebook, the Similarity model guideline is 0.9971 and the of Connectivity model guideline is {0.8, 0.1, 0.2 , 0.5, 0.2}. For Kakaotalk, the Similarity model guideline is 0.9898, and the Connectivity model guideline is {0.7, 0.2, 0.2, 0.1, 0.5}. For Daumpot, the Similarity model guideline is 0.9959, and the Connectivity model guideline of is {0.5, 0.1, 0.5, 0.2, 0.2}. In this experiment, although the Connectivity model was used only once and the SeedGroup used only one arbitrary flow, it showed a high completeness while maintaining the accuracy of 100%. Therefore, we expect to have the more higher completeness when using multiple flows to be set to SeedGroup, and we can achieve completeness of 100% when using the Connectivity model repeatedly.

In this experiment, it is meaningful to prove that there exist a certain Similarity model guideline and Connectivity guideline for each target to be classified and that it can be found through experiments and can be made more robust by repeated experiments.

V. CONCLUSION AND FUTURE WORK

In this paper, we propose a traffic grouping method using the correlation model of network flow and suggest a way to use it efficiently. And the validity of this method is proved through experiments. As the future work, we plan to build guidelines for various applications, study on a chain of guidelines that make the completeness 100% through using the Connectivity model in a cascade, and apply the method to malicious detection.

REFERENCES

[1] S. H. Yoon, J. S. Park, Baraka D. Sija, M. J. Choi, and M. S. Kim, "Header Signature Maintenance for Internet Traffic Identification." International Journal of Network Management, vol.27, No.1, Jan. 2017, pp. 1-15

[2] J. H. Choi, and M. S. Kim, "Improved Processing Speed of Traffic Classification based on Payload Signature Hierarchy." In Proc APNOMS 2013, pp.1-6, Hiroshima, Japan, Sep. 2013

[3] H. M. An, S. K. Lee, J. H. Ham, and M. S. Kim. "Traffic Identification based on Applications using Statistical Signature free from Abnormal TCP Behavior," JOURNAL OF INFORMATION SCIENCE AND ENGINEERING, vol.31, no.5, pp.1669-1692, Sep. 2015

[4] T. Nguyen, and G. A. "A survey of techniques for internet traffic classification using machine learning.", IEEE Communications Surveys & Tutorial, vol. 10, no. 4, pp.56-76, Jan. 2009.

[5] S. H. Lee, and M. S. Kim, "Application Traffic Classification using TensorFlow Machine Learning Tool.", In Proc KICS 2016, pp.224-225, ChungAng Univ, Korea, Nov. 2009.

[6] F. Risso, M. Baldi, O. Morandi, A. Baldini, and P. Monclus, "Lightweight, Payload-Based Traffic Classification An Experimental Evaluation," IEEE International Conference on Communications, Beijing, China, May. 19-23, pp. 5869-5875, 2008.

[7] Liu, Hui Feng, Wenfeng Huang, Yongfeng Li, Xing "Accurate Traffic Classification", Networking, Architecture, and Storage, NAS 2007.

[8] H.-A. Kim and B. Karp, "Autograph: Toward automated, distributed worm signature detection," in USENIX Security Symp., vol. 286, 2004.

[9] X. Feng, X. Huang, X. Tian, and Y. Ma, "Automatic traffic signature extraction based on Smith-waterman algorithm for traffic classification," IEEE Int. Conf. Broadband Netw. Multimedia Technol. (IC-BNMT), pp. 154-158, 2010.

[10] C. MU, X.-h. HUANG, X. TIAN, Y. MA, and J.-l. Qi, "Automatic traffic signature extraction based on fixed bit offset algorithm for traffic classification," The J. China Universities of Posts and Telecommun., vol. 18, pp. 79-85, 2011.

[11] C. MU, X.-h. HUANG, X. TIAN, Y. MA, and J.-l. Qi, "Automatic traffic signature extraction based on fixed bit offset algorithm for traffic classification," The J. China Universities of Posts and Telecommun., vol. 18, pp. 79-85, 2011.