A Method of Identifying the P2P File Sharing

Jian-Bo Chen

Department of Information & Telecommunications Engineering
Ming Chuan University
Taoyuan, Taiwan

Summary
In this paper, we propose a method to identify the P2P file sharing. In this method, we collect a large amount of network traffic and analyze the features of P2P file sharing by network layer and transport layers of OSI reference model. Four features are defined in this method, including quantity of packet count, percentage of TCP packet count, percentage of specific size of packet count, and percentage of duplicate destination port numbers. Based on our experiments, we can define the thresholds for each feature. Finally, we use four membership functions and a formula to identify the P2P file sharing.

Key Word:
P2P file sharing, feature, transport layer

1. Introduction

With the growth of the Internet, the network usage is increasing rapidly. Many applications are processed by the Internet. At the end of 1990, the function of end user devices was diversification, and the bandwidth of the Internet enlarged. The P2P (peer to peer) transmission is the most popular application on the Internet [1-3]. The idea of P2P is to alleviate the heavy traffic of a single server. The peers can act as both client and server which provide the contents to other peers. There are many types of P2P applications and architectures, but the most appropriate application is the P2P file sharing application. The P2P file sharing application can indeed improve the file transfer performance, but most problems of P2P file sharing application are network congestion and intellectual property rights [4-5].

This paper defines the features of P2P file sharing application according to the layer 3 and layer 4 information of OSI reference model [6-7]. The layer 3 is network layer, which contains the IP addresses of both source and destination. The layer 4 is transport layer, which contains the port number of each transmission site. The information of application layer does not need in our proposed method [8-9]. In addition to this information, we also collect the number of packets and the size of each packet. Based on this information, four features for P2P file sharing application are defined, including *quantity of packet count*, *percentage of TCP packet count*, *percentage of specific size of packet count*, and *duplication of destination port number*.

Based on these features, we want to know the thresholds for each feature. In this paper, we collect a large amount of traffic. These traffics include hosts running P2P file sharing or not. Then we can achieve the values of each feature. The thresholds of each feature can be determined by the experiments.
After determining the thresholds, almost all the file sharing can be identified. But some servers which are not running P2P file sharing may be considered as P2P file sharing. In order to avoid these kinds of error, we create a formula that adopts these four features, and all features has its own weights. Based on this formula, we can identify the P2P file sharing more accurately.

The remainder of this paper is organized as follows. In section 2, different types of P2P communications are addressed. In section 3, the features of P2P file sharing application are defined. The experimental results are given in section 4. Finally, in section 5, the conclusion is given.

2. P2P File Sharing

2.1 eDonkey/eMule

eDonkey/eMule is a decentralized architecture which does not rely on a central server to search and find files [10]. Its characteristic is fast searching, and it can search any file globally. It also allows peers to transmit any kind of file and provides the function to change to other peers for sharing the same file. Peers can download the same file from different peers in order to improve the performance of file sharing. When connected with another peer, the source peer will announce which other peers contain the same file. Consequently, the peer can download this file from other peers simultaneously.

2.2 Foxy

Foxy is the most popular P2P file sharing application in Taiwan [11]. Its architecture is like Gnutella, but its sharing unit is based on a folder. Peers share the files when they are on the shared folder. There are no bandwidth limitations for

Manuscript received November 5, 2010
Manuscript revised November 20, 2010

uploading and downloading for Foxy; it also does not need to find the seed to search for the shared files. This application can find the files and peers automatically. In order to enhance the searching performance, even if peer has no file to share, it will also respond to the source peer, which will waste lots of bandwidth. The advantage of Foxy is that it can upload and download files simultaneously. Therefore, the more other peers downloading, the more speed it can achieve.

2.3 BitTorrent

BitTorrent, sometimes call BT, cannot search shared files directly from other peers [12-13]. The peers must first find the seed (torrent file). The Tracker is contained inside the seed and records the network position, original source address, file hash values, and so on. BitTorrent downloads any shared file according the seed. The architecture of BitTorrent consists of the Tracker, seed, and peers. The Tracker is very important in BitTorrent architecture. It contains the information about which files are owned by which peers, so the peer can download the shared file from these peers. The seed always comes from some forum or exchange by peers. When a peer joins to BitTorrent, all he needs to do is to find the seed, then the peer can start to download any file.

3. Features Analysis

3.1 Quantity of packet count

When the P2P file sharing application is running, it will issue lots of packets in order to communicate with other peers. According to the observation of packet count for P2P file sharing, we can find that the amount of packet count is increasing. Because the peer must check both the availability of peers and the availability of files, it must send out many query packets to get the information. Normally, any other computer hosts, except servers, will not issue too many packets in the same time. Thus, we define our first feature according to the quantity of packet count. Firstly, we determine the quantity of packet count for normal hosts and hosts running P2P file sharing application over a set period of time. We define the threshold, when the packet count for one host is larger than this threshold, this host may be running P2P file sharing application.

$$T_a > \text{threshold} \quad (1)$$

where T_a is the total packet count for the specific host.
In Fig. 1, according to the experimental results, the packet count for the three P2P file sharing application are larger than 500. Thus, we can define the threshold as 500 packet count per second.

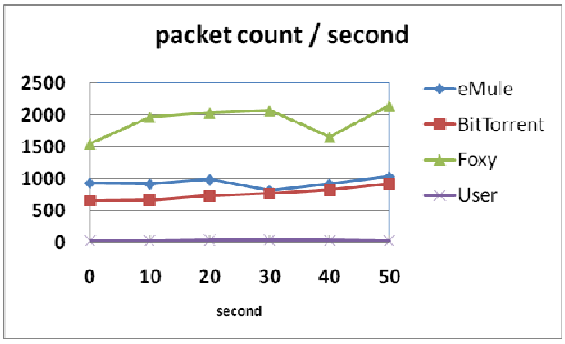

Figure 1. The comparison for packet count

3.2 Percentage of TCP packet count

From the observation of layer-4 packet types, the UDP packet count for normal users is always very small. Before this experiment, we collected the packet for a host running browser, running e-mail, connected to BBS, using instant messaging, and so on for several hours. The ratio of UDP packet count is very small which it approaches zero percent. This means that the TCP packet count is one hundred percent. Statistically, in this experiment, the percentage of TCP packet count with hosts running P2P file sharing application is always between 40% and 77%. Hence, the host running P2P file sharing application will decrease the percentage of TCP packet count. Here, the feature for percentage of TCP packet count is defined below.

$$T = \frac{T_t}{T_a} \quad (2)$$

where T_a is total packet count for the specific host, T_t is the TCP packet count for this host, and T is the percentage of TCP packet count.

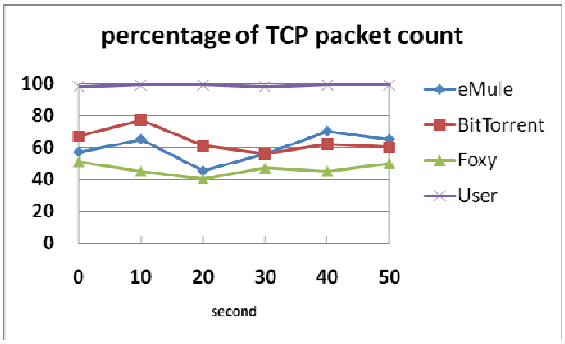

Figure 2. The comparison for percentage of TCP packet count

When comparing the percentage of TCP packet counts, the hosts running P2P file sharing are always less than 77% as shown in Fig. 2.

3.3 Percentage of specific size of packet

During the period of P2P file sharing, it can be observed that the packet sizes for TCP packets are almost 1500 bytes. The percentage of packet size larger than 1400 bytes is nearly 30%. In comparison with normal users, the percentage of packet size larger than 1400 bytes is far less than 30%. Thus, the number of large packet size is another important feature for P2P file sharing detection.

$$P = \frac{T_s - T_n}{T_t} \quad (3)$$

where T_t is total TCP packet count for the specific host, T_s is the number of TCP packets with size between 1400 to 1500 bytes, T_n is number of TCP packet with size between 1400 to 1500 bytes and the port number is well-known, and P is the percentage of specific packet size.

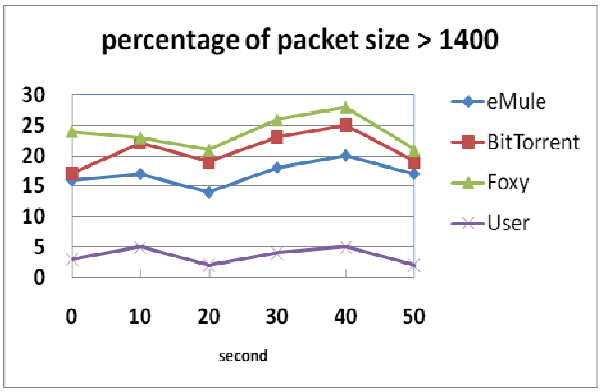

Figure 3. The comparison for percentage of specific size of packet count

The percentage of specific size of packet is shown in Fig. 3, where all the percentages of packet size larger than 1400 bytes are greater than 14%

3.4 Percentage of duplicate destination port number

After the handshakes between P2P file sharing peers, the host starts to share files with other peers. In order to improve the download performance, one host may download the same file from other peers. That is, the same source IP address will communicate with other destination IP addresses and different destination port numbers. Thus, we use the packet count for duplicate destination port number as the numerator, and the packet count for different IP address as the denominator. The value we calculate is another feature for P2P file sharing application.

$$D = \frac{T_{dp}}{T_{ip}} \quad (4)$$

where T_{ip} is the packet count for different IP address, T_{dp} is the packet count for duplicate destination port number, and D is the percentage of duplicate destination port number.

The percentage of duplicate destination port number is shown in Fig. 4, where all the percentages are greater than 16%.

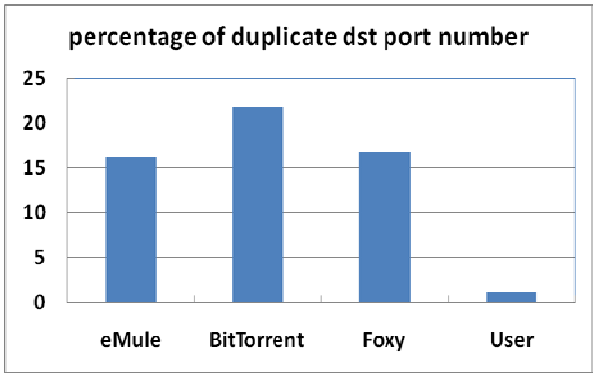

Figure 4. The comparison for percentage of duplicate destination port number

4. Experimental Results

In this section, we will use four membership functions to re-define the four features based on the thresholds. The larger value of membership function means that the host is more possible being the file sharing host.

4.1 Definition of membership functions

Now we define the four membership functions as shown in Fig. 5 to Fig. 8. For example, if the four features we collected are (1500, 70%, 25%, 15%), then we can get the four membership values (0.5, 0.25, 0.75, 0.25).

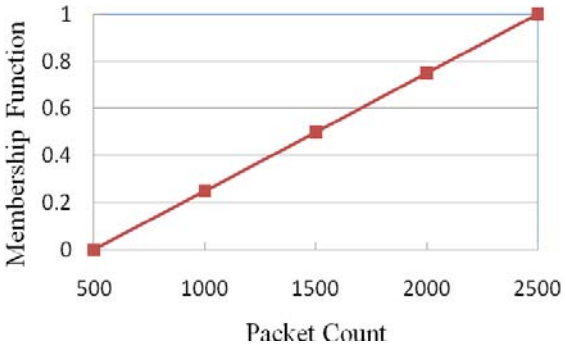

Figure 5. The membership function for packet count

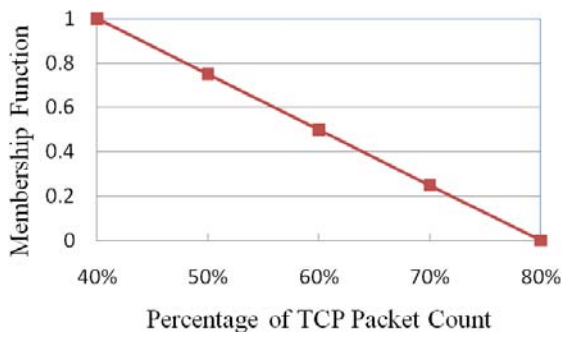

Figure 6. The membership function for percentage of TCP packet count

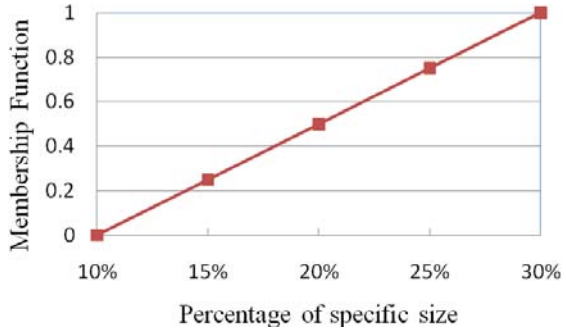

Figure 7. The membership function for percentage of specific size

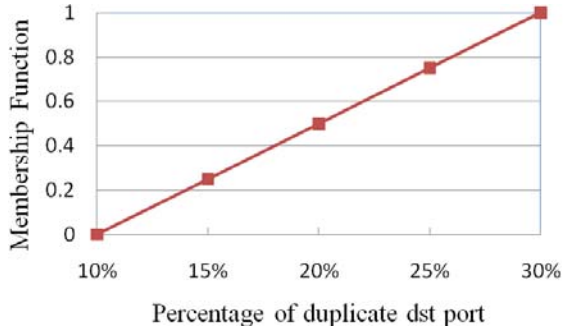

Figure 8. The membership function for percentage of duplicate dst port

4.2 The analysis of experimental results

If the four features we collected are defined as T_a、T、P、D, which T_a means the packet count, T means the percentage of TCP packet, P means the percentage of specific size, and D means the percentage of duplicate destination port number. The four membership are defined as f_1、f_2、f_3、f_4. Now we define the formula of decision value as below.

$$DV = \alpha * f_1(T_a) + \beta * f_2(T) + \gamma * f_3(P) + \delta * f_4(D) \quad (5)$$

Where DV is the final decision value, and α、β、γ、δ are the weighting values for these four membership functions, and $\alpha + \beta + \gamma + \delta = 1$. If the DV is larger than a certain thresholds, we can make the decision that this host is running P2P file sharing application.

Table 1 shown that when we adopt the formula in (5), almost all the different kinds of P2P file sharing can be successfully identified. The eDonkey/eMule can be identified by 87.8%, the Foxy can be identified by 94.3%, and the BitTorrent can be identified by 92.1%. The average accurate rate is about 91.1%.

Tabel 1. The success rate for P2P identification

P2P application	Success rate
eDonkey/eMule	87.8%
Foxy	94.3%
BitTorrent	91.2%
Average	91.1%

5. Conclusion

In this paper, four features for P2P file sharing application are defined. According to these experiments, four thresholds for these features are also defined. Based on these four features, we define four membership functions for these four features. Then we can get the four membership values. The formula we defined can be used to make the final decision. If the final decision value is larger than the threshold, we will identify that the host is running P2P file application.

References

[1] S. Sen, and Jia Wang, "Analyzing Peer-To-Peer Traffic Across Large Networks, " Trans. IEEE/ACM Networking, Vol. 12, No. 2, pp. 219-232, April, 2004

[2] Hamada, T., Chujo, K., Chujo, T., and Yang, X., "Peer-to-Peer Traffic in Metro Networks: Analysis, Modeling, and Policies, " Proc. Network Operations and Management Symposium, Vol. 1, pp. 425-438, April, 2004

[3] S. Saroiu, P. Krishna Gummadi, and Steven D. Gribble, "A Measurement Study of Peer-to-Peer File Sharing Systems," Proc. Multimedia Computing and Networking, January, 2002

[4] Spognardi, Alessandro Lucarelli, and Roberto Di Pietro, "A Methodology for P2P File-Sharing Traffic Detection," Proc. the Second International Workshop on Hot Topics in Peer-to-Peer Systems, pp. 52-61, July, 2005

[5] Matsuda T., Nakamura F., Wakahara, and Y. Tanaka, "Traffic Features Fit for P2P Discrimination," Proc. 6th Asia-Pacific Symposium on Information and Telecommunication Technologies, pp. 230- 235, 2005

[6] T. Karagiannis, A. Broido, M. Faloutsos, and Kc claffy, "Transport Layer Identification of P2P Traffic," Proc. 4th

ACM SIGCOMM Conference on Internet Measurement, pp. 121-134, New York, NY, USA, 2004
[7] Li Juan Zhou, Zhi Tong Li, and Bin Liu, "P2P Traffic Identification by TCP Flow Analysis," Proc. International Workshop on Networking, Architecture, and Storages, pp. 47-50, 2006
[8] Wang Jin Song, Zhang Yan, Wu Qing, and Wu Gong Yi, "Connection Pattern-Based P2P Application Identification Characteristic," Proc. Network and Parallel Computing Workshops, pp. 437-441, September, 2007
[9] S Sen, O Spatscheck, and D. Wang, "Accurate, Scalable In-Network Identification of P2P Traffic Using Application Signatures," Proc.13th International conference on World Wide Web, 2004
[10] The eMule protocol, http://www.emule-project.net/
[11] The Foxy protocol, http://tw.gofoxy.net/
[12] The BitTorrent protocol, http://www.bittorrent.com/
[13] Mong-Fong Horng, Chun-Wei Chen, Chin-Shun Chuang, and Cheng-Yu Lin, "Identification and Analysis of P2P Traffic – An Example of BitTorrent," Proc. First International Conference on Innovative Computing, Information and Control, pp. 266-269, 2006

Jian-Bo Chen received the MS degree in the department of electrical engineering in National Taiwan University, Taipei, Taiwan, Republic of China, in 1995, and PhD degree in the department of computer science and engineering in Tatung University, Taipei, Taiwan, Republic of China, in 2008. He is currently an assistant professor in the department of information and telecommunications engineering in Ming Chuan University, Taoyuan, Taiwan, Republic of China. His research interests include network management, network security, and load balance.