Towards Peer-to-Peer Traffic Analysis Using Flows[1]

Myung-Sup Kim, Hun-Jeong Kang, and James W. Hong

Department of Computer Science and Engineering
POSTECH, Korea
{mount, bluewind, jwkhong}@postech.ac.kr

Abstract. One of the main problems with today's Internet traffic analysis is caused by the large number of network-based applications whose types and traffic patterns are more complicated than in the past. Today, peer-to-peer (P2P), streaming media, and game traffic are continuously increasing. The difficulty the traffic analysis is that this newly emerging traffic is not as simple as past well-known port based traffic. This paper focuses on analyzing P2P traffic, which is the most complicated traffic among newly emerging Internet traffic. We describe the properties of P2P traffic and explain why P2P traffic analysis is more difficult than other types of Internet traffic analysis. Next, we propose a new algorithm suitable for P2P traffic analysis. The main idea of our algorithm is that flow grouping based on their relationships will increase the accuracy of P2P traffic analysis.

1 Introduction

There are two main problems in traffic monitoring and analysis of today's Internet traffic compared to the past network environment. The first is how to capture and handle the huge amount of traffic data in a real-time manner generated from high-speed network links, such as 2.5 Gbps and higher. The second is how to analyze various and complex types of traffic generated by many different types of network-based applications, such as streaming media, P2P, and game applications.

On the first problem, there have been many efforts and good research results reported. To increase the performance of capturing, network cards specialized for monitoring purpose (for example, DAG card [1]) were developed. Flow formats such as NetFlow [2] and sFlow [3] were created. Many routers now have a function to export flow information with these popular formats. For real-time traffic monitoring architecture, RTFM [4] was introduced and influenced the development of many traffic monitoring systems [5, 6, 7]. We have also developed a real-time traffic monitoring and analysis system called NG-MON [8], where a clustering and pipelining architecture has been applied for enhanced scalability.

[1] This work was in part supported by the Electrical and Computer Engineering Division at POSTECH under the BK21 program of Ministry of Education and HY-SDR Research Center at Hanyang University under the ITRC program of Ministry of Information and Communication, Korea.

The second problem, which this paper addresses, is caused by the high number of network-based applications. So the types and patterns of current network traffic are not simple as in the past. In the past network environment, HTTP, FTP, TELNET, SMTP and NNTP traffic occupied almost all Internet traffic. Today, the proportion of these well-known ports based traffic is decreasing. Instead P2P, streaming media and game traffic are increasing. The difficulty with traffic analysis is that this newly emerging traffic is not as straightforward as the well-known port based traffic. Therefore, we need a new algorithm to analyze these new types of Internet traffic.

This paper focuses on P2P traffic that is the most complicated traffic among newly emerging Internet traffic. A few years ago new types of network-based applications emerged, which are different from traditional client/server based application architecture, such as Morpheus [10], Gnutella [11], Soribada [9], etc. Soribada is a Korean version of Napster. P2P applications have changed Internet traffic patterns and directions in many ways. These new types of applications require a new method to analyze them. In this paper, for the analysis of P2P traffic we describe the properties of P2P traffic and explain why P2P traffic is more difficult than other types of Internet traffic. Next, we propose a new algorithm suitable for P2P traffic analysis. The main idea of our algorithm is that flow grouping according to the relationship among flows will increase the accuracy of P2P traffic analysis.

This algorithm is composed of four crucial steps. The first step is to make the Application Port Table (APT) by off-line exhaustive search of existing P2P applications. The second step is the Important Port Number selecting method from each flow record, which is important in the decision of P2P applications. The third step is the use of Flow Relationship information in the analysis phase. The last step is the decision of P2P application name for each flow. For the validation of proposed algorithm we designed and implemented a P2P traffic analysis system and present the result of P2P traffic analysis in the Internet junction of our campus network.

The organization of this paper is as follows. The properties of P2P applications and some other new traffic analysis efforts are described in Section 2. Section 3 describes our proposed P2P traffic analysis algorithm. The design and implementation issues are described in Section 4. In Section 5, we describe the analysis results using the proposed algorithm. Finally, concluding remarks are given and possible future work is mentioned in Section 6.

2 Related Work

In this section, we give a definition of P2P traffic and describe its properties. We also describe the existing traffic analysis mechanisms.

2.1 Definition of P2P Traffic

For the analysis of P2P traffic, the first step is to define the nature of P2P traffic. In this paper we define P2P traffic as traffic generated by P2P applications. Then what is

a P2P application? Figure 1 describes the critical difference between traditional client/server applications and newly emerging P2P applications.

As seen in Figure 1, in the client/server architecture we can explicitly divide all hosts into two groups: a server group and a client group. The client usually sends requests for some services and the server replies for each request. Direct communication among clients never occurs. However, in the P2P architecture each host can act as a server and a client simultaneously. In other words, direct communication between peers is possible. A host sends a request to other peers to obtain a certain service, and the same host simultaneously receives requests from other peers.

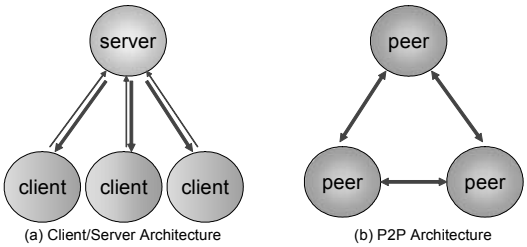


Figure 1. Client/Server Architecture and P2P Architecture

The traffic pattern is also different from that of the client/server architecture. The traffic pattern in the client/server model is directional and downstream from server to client. Otherwise, the P2P traffic is bidirectional. With these properties of P2P applications, P2P traffic can be defined.

2.2 Categorization and Properties of P2P Applications

We can categorize P2P applications into two types; they are instant messaging applications, such as MSN Messenger [12] and Yahoo Messenger [13], and file sharing applications such as Morpheus [10], Soribada [9] and eDonkey [14].

The major functions of the instant messaging applications are message delivery, single or multi-user chatting and file transfer. The major functions of file sharing applications are searching and file transfer. Besides these major functions many additional features are provided to differentiate themselves from other applications. There are many P2P applications and their number will continuously increase in the future.

Figure 2 describes a detailed communication sequence of two P2P applications: MSN Messenger [12] and Soribada [9], a Korean version of Napster. As Figure 2 shows, almost all P2P applications create multiple connections according to their different functions. Some applications use TCP and UDP simultaneously. These multiple connections make P2P traffic analysis very challenging.

We can summarize the properties of P2P applications from the traffic analysis point of view. First, there are many P2P applications. In Korea, the number of frequently used P2P applications are more than 20. The worldwide number of P2P applications is much greater than this number. Second, many P2P applications use multiple connections to support various functions. Some P2P applications use TCP and UDP simultaneously. Third, the protocol format or operation used in most P2P appli-

cations is unknown. Fourth, the port numbers used by P2P applications are dynamically generated and many of them are not registered at IANA; sometimes they use port numbers which are already registered at IANA for some other purposes.

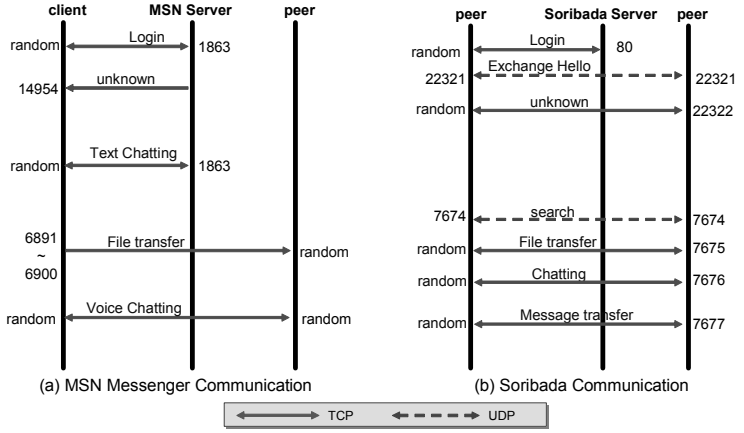


Figure 2. Communication Detail of P2P applications

2.3 Related Work on P2P Traffic Analysis

The analysis of P2P traffic is one of the important issues in the current Internet environment. Until now a few researches on the architecture of P2P applications [15], the traffic patterns and properties of some specific P2P traffic [16] have been performed. However, the method to identify P2P traffic among all Internet traffic and decide the application name of certain traffic is still very primitive. The only method currently used is the traditional method that decides the P2P application name by the port numbers. Figure 3 (a) illustrates the traditional traffic analysis method. The architecture of most traditional Internet applications is a client/server architecture. The traffic analysis of these client/server applications is very simple. According to the port number less than 1024 from the packet header information, we can decide the application name generating that packet.

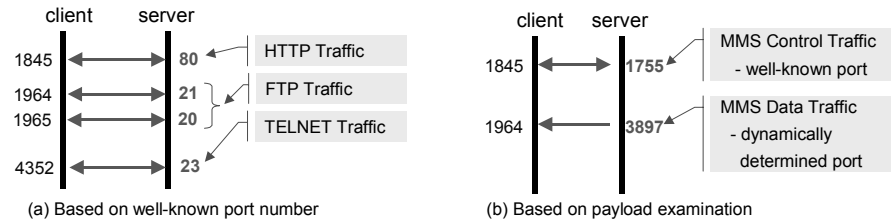


Figure 3. Existing Traffic Analysis Methods

However, streaming media traffic is not as simple as traditional client/server traffic. Streaming media applications also use client/server architecture but they usually establish two connections to communicate between hosts: one for control data trans-

fer and the other for video/audio data transfer. The port number used in a control session is a well-known fixed port. But the port number used for the data session is decided by the negotiation between the client and server. Mmdump[19] and SM-MON[20] introduced a payload examination based analysis for streaming media traffic analysis. Figure 3 (b) illustrates the method used in mmdump and SM-MON for MMS [17] traffic. They examine the payload of each control packet and detect the port number used in the data session. This method makes some overhead in the packet capture and analysis phase, because the entire packet should be captured and some processing is required in the examination of payload. This method is possible because there are few streaming media applications and protocols used worldwide, such as MMS [17] protocol and the RTSP [18] protocol. Moreover, the format of the RTSP protocol and its operation is open to the public.

Traditional well-known port based traffic analysis cannot be used in P2P traffic analysis because the port number used in the P2P application is usually over 1024 and they are using multiple connections. Further, for many P2P applications, port numbers are dynamically determined during the communication setup between the peers involved. The method of payload examination is not suitable because the number of P2P applications is large and usually the packet format and operation is not open to public. So we need a whole new method to analyze these multiple session and proprietary protocol based P2P application traffic.

3 P2P Traffic Analysis Algorithm

In this section, we present a new algorithm for P2P traffic analysis, which solves the problems that occur in the traditional well-known port number based analysis and payload examination based analysis. The main idea of the proposed algorithm is that flow grouping according to its corresponding applications will increase the accuracy of P2P traffic analysis. For example, the Web traffic typically uses port number 80 or 8080 for HTTP and 443 for HTTPS. The groups of flows generated by the Web server and client are obvious; the flows with port number 80, 8080, and 443 in the source or destination port can be grouped as Web traffic. In the case of P2P traffic, port number detection is more complex than Web traffic because P2P traffic applications are using port numbers over 1024 and the port number is often dynamically generated. If all P2P traffic can be selected among the entire range of traffic and then grouped according to its application name, then P2P traffic analysis will be performed with high accuracy.

For this purpose our proposed algorithm consists of four main processes, as illustrated in Figure 4. These processes are the Application Port Table (APT), the Important Port Selection, the Flow Relationship Map (FRM), and the P2P Application Decision. In our proposed algorithm we do not examine the payload of each packet; instead, we use only the header information of each packet.

The first step of the proposed algorithm is to construct the Application Port Table (APT). APT is constructed by the off-line exhaustive search of each P2P application using packet analysis tools. APT contains the P2P application names, their frequently used port numbers and protocol numbers. This information is used in the decision of

P2P application name of each flow in the P2P Application Decision process. The second step is the Import Port Number Selection process. In this step, the flow information is generated from the captured packets according to their 5-tuple information: source IP address, destination IP address, source port number, destination port number and protocol number. Then we select the important port number from the generated flow information. Because both source and destination port numbers of P2P traffic flow are usually over 1024, it is important to distinguish the important port number for the decision of P2P application from the randomly generated port number.

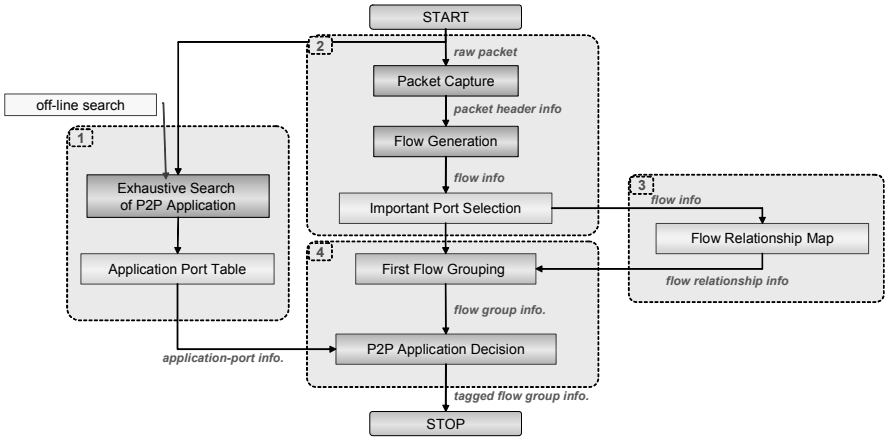


Figure 4. P2P traffic Analysis Algorithm

The third step is to construct the Flow Relation Map (FRM). Most P2P applications use multiple connections to support various functions so that it is possible to discover relationships between flows that belong to the same P2P application. The final step is to make group of flows according to the P2P application name using the results of the previous three steps.

3.1 Important Port Number Selection Method

This method comes from the fact that most of the Internet traffic is TCP traffic and most P2P applications use TCP. Figure 6 shows a normal TCP communication sequence. To establish a connection between a client and a server, the three-way handshaking mechanism is performed using SYN and SYN-ACK packets.

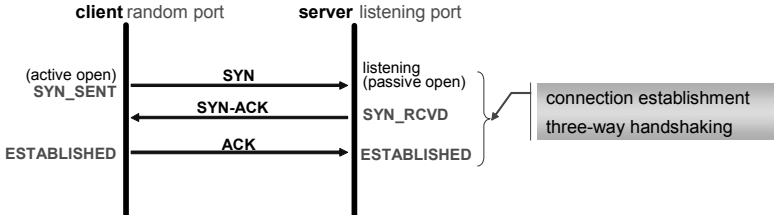


Figure 6. TCP Communication Sequence

In TCP communication, usually the server port number is fixed and not changed, but the client port number is randomly generated by operating system. Therefore, the server listening port is the important port for analyzing traffic. How can the server's listening port number be selected from the captured flow information? We utilize SYN and SYN-ACK packets in the three-way handshaking mechanism. The destination port number in the SYN packet is the server listening port. Likewise, the source port number in the SYN-ACK packet is the server listening port number. Using this information, we can determine the important port number from all the TCP flows.

In case of UDP flows we cannot apply the same method because there is no three-way handshaking mechanism like in TCP. Instead we can use the flow relationship between UDP packets to decide the important port number. We know by experiments that the patterns of UDP flows are very simple compared to TCP flows. So it is not so difficult to find relationships among UDP flows.

3.2 Flow Grouping using APT

To determine the P2P application name from the captured flow information we should know the P2P application names widely used by users. Through the exhaustive search of P2P applications using packet analysis tools such as tcpdump and ethereal [21] we construct the APT which contains the information about each P2P application, as illustrated in Table 1. The APT contains the P2P application names, frequently used TCP/UDP port numbers and one representative port number for each. As Table 1 shows, most P2P applications use multiple port numbers that are not mostly registered at IANA [22]. Some P2P applications use both TCP and UDP.

Table 1. An Example of Application Port Table

Application Name	TCP		UDP	
	representative port	well-known ports	representative port	well-known ports
MSN Messenger	1863	1863, 6981-6990, 14594		
Yahoo Messenger	5101	5101, 5050		
AIM/ICQ	5190	5190		
Soribada	22322	22322, 7675, 7676, 7677	22321	22321, 7674
eDonkey	4661	4661, 4662, 6667		
Shareshare	6399	6399	6777	6388, 6733, 6777

We select one port number among the frequently used port numbers by each P2P application and use it as the representative port number of that P2P application. If a P2P application uses TCP and UDP then two representative port numbers are assigned to each protocol respectively. This representative port number is used to indicate the groups of flows belonging to the same P2P application. In the final step of the proposed algorithm, the flows belonging to a P2P application are tagged with the corresponding representative port number. Therefore, all P2P flows are grouped by the tagged representative ports.

3.3 Flow Grouping using Flow Relation Map

The Application Port Table (APT) and Important Port Selection cannot give 100% of accuracy in the decision of P2P application name to all P2P flows. There are two reasons for this. First, there are too many P2P applications around the world to examine. Also, the complete examination to discover all used port numbers is difficult. In many cases, dynamically generated port numbers are the important port numbers. In such cases APT cannot provide flow-grouping information for this P2P traffic. Second, it is also possible that the same important port number is used by more than two P2P applications. In this case, we cannot decide which P2P application generates this flow without flow relationship information. Therefore, we propose the third step, the Flow Relationship Map (FRM), to increase analysis accuracy.

Currently, we are using a basic and simple relationship method among flows. First, the flows are grouped according to the combinations of source port, destination port, and protocol. We give a priority value to each combination of these three flow properties according to the weight of dependency, as illustrated in Table 2 (a). For example, the UDP flows with the source port number 22321 and destination port number 22321 are grouped with priority 100. This processing is called the property dependency grouping.

Table 2. Flow Dependency Table

	protocol	source port	destination port	priority
0				0
1			1	20
2		1		20
3		1	1	50
4	1			0
5	1		1	50
6	1	1		50
7	1	1	1	100

(a) Property Dependency Table

	source ip	destination ip	priority
0			0
1		1	10
2	1		10
3	1	1	100

(b) Location Dependency Table

After this property dependency grouping, all groups are linked with the weight. The weight value is decided by the priority values in the location dependency table, which is illustrated in Table 2 (b). The link weight between two groups is high when the source and destination IP addresses of flows in the two corresponding groups are highly dependent on each other. Otherwise, the weight is low. We call this processing the location dependency grouping.

By these two steps in the grouping method, the flows related to each other are grouped. And this group information is used in the P2P Application Decision process to increase the accuracy of analysis.

4 Design and Implementation of P2P Traffic Analysis System

In this section, we describe the design and implementation of the P2P traffic analysis system using the proposed method. The system was developed as a plug-in to the real-time traffic monitoring analysis system called NG-MON [8].

4.1 Design of P2P Traffic Analysis System

Figure 6 illustrates the overall design of the P2P traffic analysis system, which consists of three main modules. They are the APT module, the Important Port Selector module, and the Flow Relations Mapper module.

The Important Port Number Selector module consists of a Packet Capturer, a Flow Generator, and a SYN Packet Table. The Packet Capturer receives raw packets from a network link and generates packet header information from each raw packet. The packet header information is sent to the Flow Generator. If a packet is a SYN or SYN-ACK packet, it is stored in the SYN Packet Table. The SYN Packet Table keeps the TCP listening port information. To select an important port number from each flow, the Flow Generator looks up the SYN Packet Table.

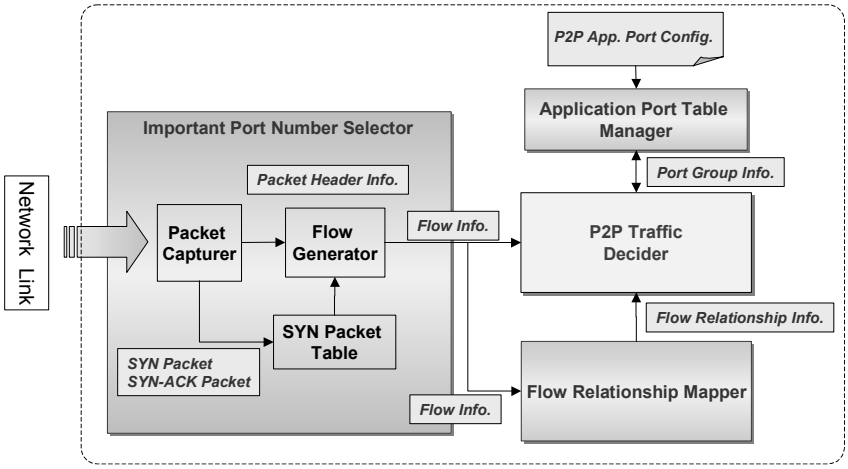


Figure 6. P2P Traffic Analysis System Design

The important port determined flows are sent to the P2P Traffic Decider and the Flow Relationship Mapper. The Flow Relationship Mapper module keeps track of the flow relationship and the history of flows from each host and provides this information to the P2P Traffic Decider Module.

Through the off-line search of each P2P application, we built a P2P port configuration file with XML. We used XML because it is easy to use and many XML-related libraries are provided in various languages, such as C/C++ and Java. The APT manager reads this configuration file and keeps the port group information. When the P2P Decider module receives a flow from the Important Port Number Selector module, it looks up the Flow Relationship Map and Application Port Table to decide the P2P application name where the flow belongs. Finally, the P2P Traffic Decider groups the flows by tagging each P2P flow with the corresponding representative port number. If the important port number of flow does not belong to a certain P2P application port group but this flow has high relationship with other flows which belong to that P2P application according to FRM, then the P2P Traffic Decider tags this flow with the same representative port number and updates the APT.

4.2 Integration of P2P Traffic Analysis System with NG-MON

NG-MON [8] is a real-time Internet traffic monitoring system for high-speed networks, developed at POSTECH. The P2P traffic analysis module is implemented as a plug-in module to NG-MON. Figure 7 illustrates the integration of the P2P traffic analysis module with the current NG-MON system.

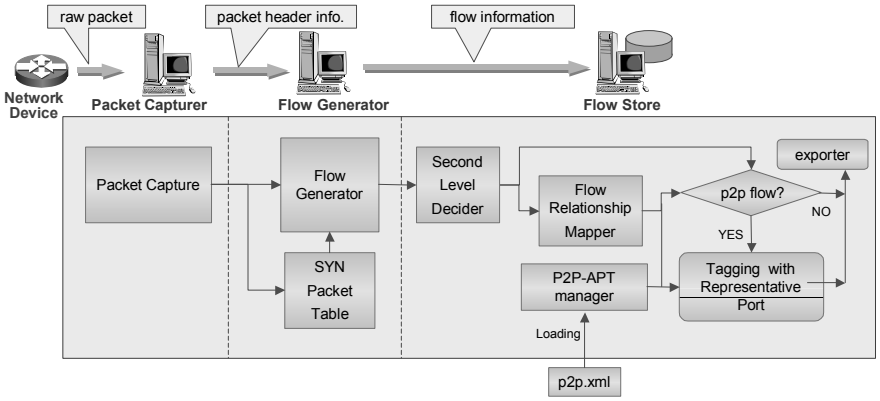


Figure 7. Integration of P2P traffic analysis system with NG-MON

The components of the Import Port Number Selector module are separated into the Packet Capturer, Flow Generator, and Flow Store phases. We use the Packet Capturer module of NG-MON as it is. The SYN Packet Table is added into the Flow Generator and the Second Level Important Port Number Selector is located in the Flow Store system. The APT manager, Flow Relationship Mapper, and P2P Traffic Decider are added in the Flow Store system. The P2P Traffic Decider Module is illustrated with the flowchart-like diagram in Figure 7. As a result of the P2P traffic analysis system, the tagged flow information with the representative port number is stored in the Flow Store system. The traffic analyzer determines the corresponding P2P application name of each P2P flow by the representative port number.

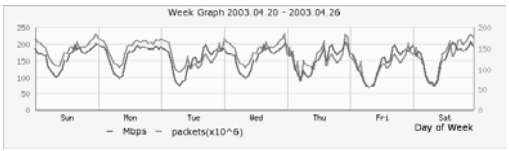
5 Result of P2P Traffic Analysis

We have deployed NG-MON with the P2P traffic analysis module in the Internet junction of our campus. Our campus Internet link is composed of two 100 Mbps Metro Ethernet links. Considering the bi-directional traffic, the maximum amount of traffic we analyze is 400 Mbps.

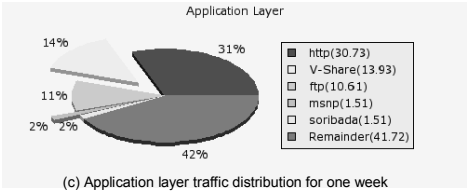
We can see the result of P2P traffic analysis at the application protocol view page of the NG-MON Presenter system. Figure 8 (a) shows an example of P2P traffic analysis results. NG-MON captures all the in/out Internet traffic and analyzes them from various points of view, such as throughput analysis per host and subnet, time series analysis of the throughput changes of each host and subnet. Figure 8 (b) and Figure 8 (c) is a result of NG-MON analysis during one week. Figure 8 (b) shows a time series graph of throughput and packet size changes during the tested period. The

total amount of captured data size is 11,493,562,602,529 bytes from 17,427,364,409 packets. The average bandwidth was 152.03 Mbps. The ratio of TCP and UDP among total IP packets was 82.3% and 9.9%, respectively.

(a) Application protocol view page



(b) Time series graph of Internet traffic for one week



(c) Application layer traffic distribution for one week

Figure 8. Analysis Result of P2P Traffic

Figure 8 (c) shows the application layer analysis result where our P2P analysis mechanism is applied. HTTP traffic occupies the largest part of the pie chart; it is 30.73% of total IP traffic. But the second is not FTP data; the proportion of FTP traffic is only 10.61% and the third. The second largest traffic is V-share P2P application traffic [22]. The fourth and fifth largest traffic is generated by the MSN messenger application [12] and the Soribada file sharing application [9]. We examined 20 popular P2P applications and made an Application Port Table (APT). The proportion of these 20 P2P applications was 32.53% of total traffic. The percentage will increase if we examine more P2P applications.

6 Conclusion

In this paper, we have presented a new algorithm for analyzing P2P traffic. First, we explained the properties of P2P traffic and the reasons why the existing analysis mechanism is unsuitable for P2P traffic analysis. The proposed algorithm consists of four main components: the Important Port Number Selection, the Application Port Table, the Flow Relationship Map and the P2P Traffic Decider. Using this proposed algorithm we designed a P2P traffic analysis system and implemented it as a plug-in to NG-NOM. Using this system we were able to analyze considerable amounts of unknown traffic which could not be determined by the traditional analysis method. The result of P2P traffic analysis on our campus Internet junction shows that the proportion of P2P traffic is steadily increasing.

The proposed algorithm can be improved still further including the Flow Relationship Map. By more experimental tests on our campus Internet junction, the efficiency of the proposed algorithm will be validated. In addition to the validation of our algorithm, we are going to apply proposed flow grouping algorithm to the analysis of other types of Internet traffic, such as game and streaming media traffic.

References

1. Ian D Graham and John G Cleary, "Cell level measurements of ATM traffic," Proc. of the Australian Telecommunications Networks and Applications Conference, pp. 495-500, Dec. 1996.
2. Cisco, White Papers, "NetFlow Services and Applications," http://www.cisco.com/warp/public/cc/pd/iosw/ioft/neflct/tech/napps_wp.htm.
3. P. Phaal, S. Panchen and N. McKee, "InMon Corporation's sFlow: A Method for Monitoring Traffic in Switched and Routed Networks", IETF RFC 3176, September 2001.
4. N. Brownlee, C. Mills and G. Ruth, "Traffic Flow Measurement: Architecture", IETF RFC 2722, October 1999.
5. N. Brownlee, "Traffic Flow Measurement: Experiences with NeTraMet", IETF RFC2123, March 1997.
6. Ken Keys, David Moore, Ryan Koga, Edouard Lagache, Michael Tesch, and k claffy, "The Architecture of CoralReef: An Internet Traffic Monitoring Software Suite," PAM Workshop 2001, April, 2001.
7. Argus, http://www.qosient.com/argus/.
8. Se-Hee Han, Myung-Sup Kim, Hong-Taek Ju and James W. Hong, "The Architecture of NG-MON: A Passive Network Monitoring System", LNCS 2506, DSOM 2002, October 2002, Montreal Canada, pp. 16-27.
9. Soribada, http://www.soribada.com/.
10. Morpheus,http://www.morpheus.com/
11. Gnutella, http://gnutella.wego.com.
12. MSN Messenger, http://messenger.msn.co.kr/.
13. Yahoo Messenger, http://kr.messenger.yahoo.com/.
14. eDonkey, http://www.edonkey2000.com.
15. Matei Ripeanu, "Peer-to-Peer Architecture Case Study: Gnutella Network", Technical Report TR-2001-26, University of Chicago, July, 2001.
16. Subhabrata Sen and Jia Wang, "Analyzing Peer-to-Peer Traffic Across Large Networks", IMW2002 Workshop, 2002, Marseille, France.
17. Microsoft, Windows Media Technology, http://www.microsoft.com/windows/windowsmedia/default.asp.
18. H. Schulzrinne, A. Rao, and R. Lanphier, "Real Time Streaming Protocol (RTSP)," RFC 2336, April 1998.
19. Jacobus van der Merwe, Ramon Caceres, Yang-hua Chu, and Cormac Sreenan, "mmdump-A Tool for Monitoring Internet Multimedia Traffic," ACM Computer Communication Review, Vol. 30, No. 5, 2000.
20. Hun-Jeong Kang, Hong-Taek Ju, Myung-Sup Kim and James W. Hong, "Towards Streaming Media Traffic Monitoring and Analysis", APNOMS 2002, September 2002, Jeju, Korea.
21. Ethereal, http://www.ethereal.com/.
22. V-share, http://www.v-tv.co.kr/.