Identification of P2P Flows Through Host Activity

John Hurley, Emi Garcia-Palacios, Sakir Sezer
The Institute of Electronics, Communications and Information Technology (ECIT),
Queens University of Belfast,
Northern Ireland,
jhurley03@qub.ac.uk, e.garcia@ecit.qub.ac.uk, s.sezer@ecit.qub.ac.uk

Abstract— **With the increasing quantity and varying nature of traffic crossing the internet, coupled with techniques such as fluctuating port numbers and transport layer encryption, the identification of individual packet flows is becoming more difficult. We introduce and investigate a new method for the detection of P2P flows based on the activity of the hosts (IP addresses) involved in the connection. Heuristics are generated that examine properties of these hosts and used to uniquely detect individual P2P and non-P2P flows. The identification strategy has been tested on two real network data traces from a core internet router with some classification accuracies showing higher than 99%.**

Keywords-component; P2P; Traffic Classification; Host Acivity

I. INTRODUCTION

Peer to Peer (P2P) file sharing is now a key contributor to the makeup of traffic on the internet. It has been suggested that the portion it occupies might be as high as 70% [1]. However, the use of P2P for illegal, malicious, and copyrighted data transfer as well as its attempt to utilise all available bandwidth – leading to reduced Quality of Service (QoS) for other applications – mean the detection and identification of P2P is a key task for Internet Service Providers (ISP). Identification and the subsequent classification of P2P is a fundamental part of many security and QoS policies for ISPs and network administrators.

However, the identification and classification of traffic (especially P2P traffic) is becoming more difficult. Port based classification [2] can no longer be relied on to identify packets as many applications now use random (ephemeral) ports. There are also cases were protocols use the well known port numbers of other applications to avoid detection. For example, the Kazaa P2P protocol is known to use the well known port of HTTP (80) to transfer data [3].

Further advances have lead to the classification of traffic in flows rather than on a 'per packet' basis. A flow is considered as all packets involved in the same process and determined by their 5 tuple value (source and destination IP address, source and destination port number, and transport layer protocol [4]). Therefore if one packet in the flow can be identified then all packets from within that flow can be classified the same.

An accurate method of detecting one packet from within a flow is by using application signatures [5]. These are ASCII strings or a series of bytes that appear within the transport layer payload of a packet. However, to avoid detection by this method P2P protocols can encrypt their transport layer payload. Encryption is known to exist in every packet on the Asian P2P network Winny [6], while it has also been introduced to other popular P2P networks such as BitTorrent [7]. As restrictions continue to be put on P2P protocols we believe that encryption will become standard in all P2P packets.

To identify encrypted packets, statistical data such as the lengths and timings of packets in a flow can be gathered [8]. However, identification and classification through such statistical methods can still be avoided by padding out packet lengths or transmitting packets in a format similar to another application (e.g. Gnutella P2P uses a HTTP format to transfer file data).

In this research we consider the activity of hosts (IP addresses) on the network to detect and classify P2P flows. Examining host activity identifies fundamental characteristics of P2P and non-P2P protocols. This means that detection cannot be avoided without changing how a protocol operates. We introduce a new approach to classification through host activity by focusing the classification towards individual flows. This approach has shown a high level of accuracy in its detection of P2P and web (non-P2P) flows.

Section 2 of this paper introduces existing research on the use of hosts for classification. Section 3 discusses our experimental setup with section 4 introducing some of the heuristics used for flow identification. Section 5 introduces our classification strategy with section 6 showing its accuracy when applied to real network traffic. Finally, section 7 concludes the paper.

II. RELATED RESEARCH

The activity of hosts on a network has been considered for P2P identification in various manners. On such tactic is to identify the service (listening) port used for P2P connections. Winny, the mainly Asian P2P network is identified in [6]. A decoy Winny host is deployed on the network and port/IP pairs are recorded for other Winny hosts. This exploits the property of P2P hosts using the same listening port (even if ephemeral) for receiving incoming requests. Once enough information is gathered the decoy host can be removed and new hosts discovered through continuing analysis of the network.

[9] also searches for P2P service ports by gathering information on how the hosts on the network interact with each

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7229
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7229

other. It considers what it describes as a network diameter. This is a connection between a series of hosts where host A initiates a connection to host B, B to C, C to D and so on. The longest number of hops witnessed from A to D is defined as the network diameter. Those with a large diameter are marked as P2P. This considers P2P to act in its own sub-network with each host receiving requests as well as initiating requests to further peers (acting as a client and a server).

The problem with the strategies mentioned above is that none focus on the identification of individual flows which is a requirement for ISPs and network administrators. They are designed to identify service ports. Because of this they are of limited use when the full activity of the network is unavailable (e.g. at a router when only data taking that path across the network is seen). They also generate large memory and processing costs as information on all hosts needs recorded and analysed for an extended period of time.

In [10] the observation that P2P hosts tend to connect together with one TCP connection while web flows have many concurrent connections to allow parallel downloads is used. It proposes that in P2P the number of distinct IP's connecting to a host should be equal to the number of source ports used for the connections. Results show a 10% false positive rating even with the use of some well known port numbers. This is inadequate for real time use.

BLINC [11] considers 3 levels for traffic classification – application level, functional level and social level. The social level analyses the host behaviour through how many other hosts an IP address interacts with. All three levels classify approximately 90% of all traffic, not just P2P, with about 95% accuracy. However, these results take into account the application and functional levels which make use of further flow statistics and examination of many different flow connections are at the one host. It is unknown how effective their 'social level' analysis is on its own when further information is unobtainable due to positioning on the network. The main aim of BLINC is to identify the activities at hosts through the flows it is active in.

Our research differs from BLINC and similar approaches in that we focus on the identification of individual flows by using the activity of the hosts involved. It is the activity of an individual flow that is of interest to ISPs and network administrators rather than defining the activity of hosts on the network.

We identify a flow by examining only 2 hosts (source and destination) and so it is possible to classify even if limited network activity is witnessed due to the position of the classifier within the network – we can classify even if we can only gather information on the activity of either the source or destination host. Further to this, our classification strategy can identify a flow even if only partial activity is noticed from a single host. This overcomes a problem inherent in other approaches.

Our approach is also carried out with no need for transport layer payload information (application signatures) or well known port numbers. We utilise the common property of P2P active hosts to have a set listening port to receive incoming connections but do not consider the actual number used. This restricts the possibility of port masquerading were a flow tries to disguise itself by using the well known port number of a different protocol. We believe adopting this approach presents new opportunities to use host based analysis for flow specific classifications. Because host based analysis concentrates on the fundamental workings of a protocol it will be much harder to avoid detection on a network than it is when only flow characteristics and properties are considered.

III. EXPERIMENT SETUP

P2P flows are analysed in [12]. Three types are described; mice (transfer less than 10 KB of data), elephant (transfer more than 5 MB of data), and buffalo (all in between flows). [12] states that 75.78% of web flows are mice while 92.93% of P2P flows are mice. It also measures the percentage of elephant flows as 0.81 for P2P and 0.04 for web traffic. However, this small percentage of elephant flows transfer 93.43% of all the bytes witnessed in P2P and 15.35% of the bytes in web traffic [12]. This information tells us that for every long, high bandwidth consuming P2P flow, there will be many small flows witnessed from the same host. In this research we concentrate on identifying the longer P2P flows – which will be of interest to an ISP as these are the high bandwidth consuming flows that transfer the file data. However, we do so by taking into account the short (mice [12]) TCP flows and the UDP flows that are connected to it. We consider a long flow to be any TCP connection were more than 20 data packets are passed.

The data used for experimentation in this research has been collected from a core router on the home user broadband network of a major European provider. Two traces of over an hour have been captured from different times of the day (morning and evening when internet usage should vary [10]) in March 2007. Each of the traces records approximately half of all the flows that pass across the router in that time period. The morning trace captures 13 GB of data with almost 40 GB captured in the evening when more users are accessing the internet. For experimentation and testing in this research samples of 20 million packets from the morning and evening traces are analysed. The morning trace is used to test proposed heuristics and decide on the most valid, while the evening trace is used for validation of the entire heuristic based classification strategy. Because a core router is used, not all of a host's activity will be viewed across it. However, our research aims to identify P2P and non-P2P flows even if limited activity of their hosts is witnessed.

The flows analysed in our experiments have been pre-processed using application signatures [5, 13]. Although it is the purpose of this research to find alternative classification strategies to application signatures, their use is still justified in pre-processing due to their high accuracy [5]. Well-known port numbers [2] were also used in situations where signatures couldn't generate classifications, for example, secure HTTP (HTTPS) which encrypts the data in its flow but is known to run over port 443. There may be some flows left unclassified by the pre-processing technique due to the use of unknown protocols, port fluctuation/masquerading, or payload encryption. However, because those flows that are classified

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7229
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7229

with the pre-processor should be accurate we can still use this information to test our classifier against. Three groups of flows are generated by the pre-processing phase; P2P protocols including BitTorrent, Gnutella, eDonkey, AppleJuice, FastTrack and DirectConnect; non-P2P traffic including WWW traffic, FTP, email and chat protocols; unknown flows that contain all flows that cannot be classified using application signatures or a selection of well known ports.

IV. Heuristics

In this section we describe how hosts are examined to generate flow specific classifications. Heuristics are proposed with the intention of exploiting fundamental characteristics of P2P and non-P2P activity. Because we focus on the protocols activity a P2P developer could not disguise itself by adding encryption or by fluctuating ports. The entire workings of the protocol would need to be redesigned and the networks would have to function differently to pass undetected.

To focus host activity towards flow classification we consider 3 host behavioural areas – the activity of the destination host of the flow, the activity at the source host of the flow, and the activity between both hosts (i.e. other flows witnessed between the same source and destination). Figure 1 introduces a key that we use to describe some of the heuristics.

A. *Destination host*

The destination host of a flow provides the best information about what the flow is active in. This is because the destination host is likely to be involved in the same activity as the flow. Therefore identification/classification of the destination host activity can give a classification of a flow travelling towards it. For example, a P2P flow will travel to a destination that is likely to be acting as client and a server (P2P host) while a HTTP web request will travel to a dedicated web server. Figures 2 and 3 give examples of how a destination host activity can be used to identify P2P and non-P2P flows. Also given is the effectiveness of these techniques when applied to our morning trace taken from a core internet router.

Figure 2 shows an inspected TCP flow travelling to a destination host that is involved in further incoming and outgoing connections (acting as a client and a server). We can hypothesise that hosts with this activity are likely to be involved in P2P connections and hence a flow that is travelling to one of these hosts is likely to be P2P. The accompanying graph of figure 2 shows this hypothesis to be true. All the flows travelling to a destination host with this behaviour are either P2P or unknown (many unknown are likely to be P2P which are unclassified by application signatures or ports). Almost 50% of all inspected BitTorrent and Gnutella flows can be classified using this destination analysis technique.

In the diagram of figure 3 a destination host is shown to have many distinct incoming TCP connections with no further outgoing connections. All of the incoming connections are full connections (all requests responded to). This is a rare activity in P2P hosts as [12] explains that P2P hosts have many short signalling and failed connections. The graph of figure 3 shows a small percentage of P2P flows are selected but these are far outweighed by the numbers of non-P2P flows.

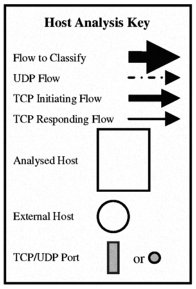

Fig. 1. Key for host analysis diagrams

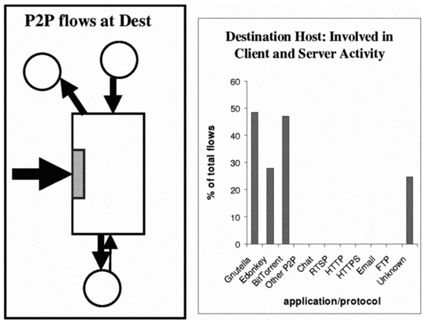

Fig. 2. Detecting P2P flows by destination analysis

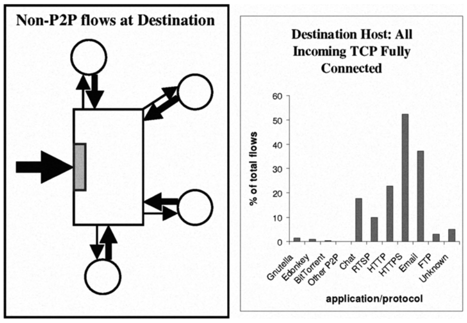

Fig 3. Detecting non-P2P flows at destination

B. *Source Host*

Source host analysis differs from destination analysis in that if a source host is predicted to be involved in a certain activity it doesn't mean all flows generated from that host are involved in the same activity. For example, the user may be browsing the web while a P2P download is currently taking place. Figures 4 and 5 give two techniques for identifying P2P and non-P2P flows through analysis of their source hosts.

Figure 4 considers the observation that the destination TCP port and UDP ports will often remain the same for P2P. Therefore if a source either receives or sends a UDP message on the same port as it is sending a TCP flow to, then it is likely that the TCP flow is P2P. The results of this heuristic do not signal as many flows as in the destination analysis but they do still show a large differential between P2P and other forms of web traffic.

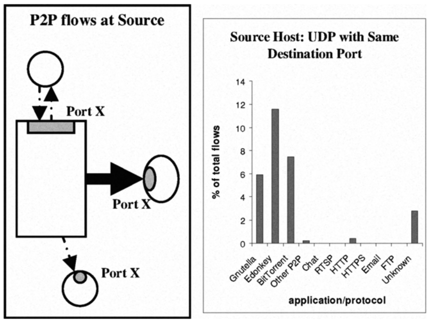

Fig. 4. Detecting P2P flows by source host analysis

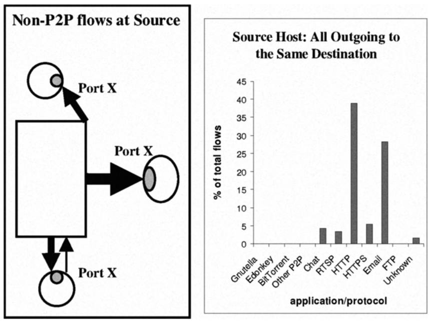

Fig. 5. Detecting non-P2P flows at source host

Figure 5 considers non-P2P hosts by identifying the source host as having all TCP flows outgoing and to the same destination port. This suggests that the user of this host computer is only involved in one type of web activity (same destination ports) and that this activity is not P2P (no attempted incoming TCP). This technique will not function when trying to detect flows where many applications are in use concurrently. However, the results of this test (graph of figure 5) show a high percentage of web flows, including a lot of HTTP, accurately classified with no P2P flows misclassified.

C. *Both hosts*

The two previous sub-sections focus on analysis of a specific host, however, we also consider further connections between the same hosts. Figures 6 and 7 show two examples of how this technique can aid the flow identification process for P2P and non-P2P traffic.

Figure 6 examines further bidirectional (incoming and outgoing) UDP flows between the same IP addresses as the analysed TCP flow. The graph of figure 6 shows this method to be excellent in the detection of Gnutella flows while selecting no non-P2P traffic. Gnutella is known to use UDP in a 'ping/pong' format [14] for searching and signalling throughout its network. If a ping is received at a host it is forwarded to all other hosts connected to it and a pong responded to the host that signalled it. Therefore if 2 hosts are on the Gnutella network and involved in a file transfer across TCP it is likely that at some point a UDP ping/pong will take place between the two.

Figure 7 examines further 2 way TCP connections between the same 2 hosts on different server port numbers. The graph (figure 7) shows that this method selects a lot of HTTPS flows. HTTPS is a secure connection normally used to send password or login details to a website. Therefore, it is common for a source host to connect to a destination server using port 80 for web requests and have a separate flow between the same two hosts using port 443 (HTTPS) for private information. The graph also shows a lot (almost 100%) of FTP flows are selected. FTP uses at least 2 channels for a file transfer. A control channel and a data channel. These channels run over different port numbers (normally 20 and 21).

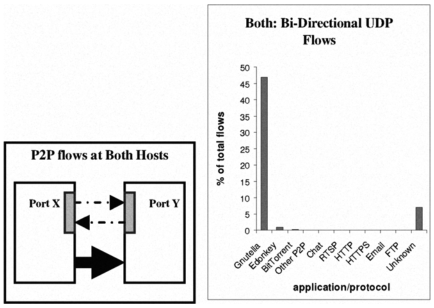

Fig. 6. P2P identification between both hosts

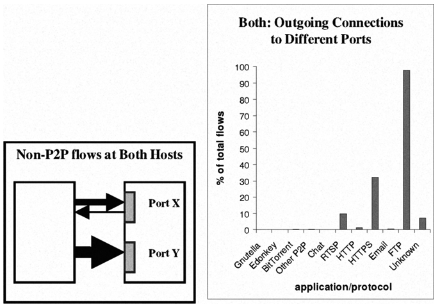

Fig. 7. Non-P2P flows between both hosts

V. CLASSIFIER

We propose a set of classification heuristics based on the host behavioural areas defined in section IV. A selection of these heuristics are then gathered to define a classification strategy. We propose an ordered system for the heuristics whereby the most successful classifiers are applied to trace data first and if a classification cannot be made then the next heuristic is applied and so on. This means that heuristics which appear further down the order will be applied to a set of flows that will have many P2P and non-P2P flows filtered out. Therefore heuristics which may generate many false positives when applied to all flows in the data trace may become more successful due to their application on a reduced flow set.

To determine the ordering of heuristics a ratio value is calculated for each by dividing the number of flow selections it makes from the protocol set it is defined to recognise (P2P or non-P2P) by the number of flows it selects from the opposite protocol set. The larger the value, the better the heuristic at selecting flows from its designated protocol set. If a ratio is calculated as infinite then the heuristic selects no flows incorrectly and is considered among those most effective.

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7229
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7229

We implement an algorithm that calculates these ratio values and orders the heuristics. It is run over our morning trace data (training trace). All heuristics are applied to all flows with a ratio calculated for each. The heuristic with the highest ratio value (or infinite) is added to position 1 in the classifier. The flows selected by this heuristic are removed from the trace data set and all remaining heuristics are applied to the reduced data set. Again the best is selected and added to position 2 in the classifier. This process is continued until all heuristics are ordered. We then apply the full classification ordering to the full trace data. Each heuristic is inspected relative to its ordering in the classifier. Those determined to classify too few flows correctly or have too many misclassification are removed from the strategy. A selection of 12 heuristics were chosen and ordered as follows – also included is the host behavioural area they should be applied to and the protocol set they are defined to detect.

- Both (web): multiple outgoing TCP flows to different ports than analysed flow destination – figure 7
- Source (web): all outgoing TCP to same destination port as analysed TCP destination port – figure 5
- Destination (P2P): initiating outgoing TCP flows to different ports than analysed TCP destination port – figure 2
- Destination (P2P): attempting outgoing TCP
- Both (P2P): bidirectional UDP between same hosts – figure 6
- Destination (P2P): UDP flows using same port as analysed TCP destination port
- Both (P2P): UDP using same port as destination TCP
- Destination (web): only incoming TCP with no outgoing TCP or UDP connections
- Source (web): 80% or more of outgoing TCP travelling to same destination port as the analysed flow
- Source (P2P): incoming TCP flows and UDP on same destination port as outgoing TCP – figure 4
- Both (web): over 10 TCP outgoing connections
- Destination (web): all incoming TCP complete connections with no fails – figure 3

VI. RESULTS

With the classification strategy in place we apply it in full to our morning trace from the core internet router that was used for training. We also apply the classification strategy to a further trace (from the evening) to gather independent results and show the heuristic are not 'trained' to work on one specific data trace from a specific time period.

Four calculations are used to show how effective the host based flow classification strategy is for detecting P2P and non-P2P activity. Firstly the 'success rate' calculates the percentage of the total number of P2P and non-P2P flows classified by application signatures and ports that are also selected by the host based heuristics. It tells how many of the protocol set's flows are correctly identified by our classification strategy.

The next 3 calculations show the accuracy of these flows that have been selected by the heuristics when compared to the application signature classification; 'classified correct' gives the percentage of flows selected by the heuristics that are correct according to the application signatures/ports; 'misclassified' gives percentage of the heuristic selections that are marked as the opposite protocol set by the signatures/ports (e.g. heuristic classifies as P2P, application signatures classify as non-P2P); 'unknown' gives the percentage of flows that are selected by heuristics but have been left unclassified by the application signatures and ports due to ephemeral port usage and possible encryption.

The results of applying all heuristics from section 5 to the two network traces are presented in tables 1 and 2. Table 1 shows a high percentage of the web flows that are selected by pre-processing have also been selected by the host analysis heuristics. Lower results are shown for P2P. However, we are still able to detect over 50% of the P2P flows from a core router were visibility of a host behaviour may be greatly reduced. The setup of many of the heuristics (section 5) means that a flow can be classified even if few other flows are witnessed at a host. For example, in the third heuristic described (section 5), only 1 more outgoing TCP to a different destination port needs to be witnessed to signal an analysed flow as P2P. The overall visibility of the destination host is irrelevant as long as a least one other flow matching the heuristic requirement passes our analysed router.

Out approach to P2P and non-P2P flow classifications is further justified when examining table 2. The accuracy levels for our heuristic selections are extremely high. For web classifications in table 2 a maximum of 0.54% flow misclassification is shown. The evening trace data (that was not used for heuristic training) is lower still at 0.03% of flows. The P2P heuristic classification accuracy shows a maximum of 0.15% of the selected flows are marked as non-P2P flows by the pre-processing. This is far less than the false positive ratings of other host based classification schemes (approximately 20% [9], 8-12% [10]).

TABLE I. SUCCESS OF HOST BASED HEURISTICS

Heuristic	Trace	Success rate
Web	**Morning**	93.13
	Evening	90.80
P2P	**Morning**	50.77
	Evening	51.57

TABLE II. ACCURACY OF HOST BASED HEURISTICS

Heuristic	Trace	Classified Correct	Misclassified	Unknown
Web	**Morning**	99.30	0.54	0.16
	Evening	99.13	0.03	0.84
P2P	**Morning**	93.13	0.07	6.80
	Evening	90.38	0.15	9.47

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7229
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7229

In the P2P results of table 2 the percentage of flows classified correctly seem smaller than for web traffic. This is due to the number of unknown flows selected by the heuristic classifiers. These are flows that have not been recognised by the pre-processing application signatures due to having unknown signatures or encrypted payloads.

Manual examination of the 'unknown' flows for web activity has shown that correct classifications have most likely been made but are not recognised by the pre-processing. The small percentage of unknown flows in the morning trace (0.16%) mainly consists of TCP flows travelling to port 8080. This is an alternative port for HTTP connections. Although these flows have not been found to contain an application signature it is still probable that these are HTTP flows and have been correctly classified by the heuristics.

Examining the unknown flows marked as P2P shows that almost all witnessed port numbers are unconnected with any application or protocol described in [2]. This use of ephemeral ports and lack of application signatures suggests that most of these flows are attempting to pass through the network undetected. Because of this, we believe that most of these unknown flows are P2P as this is the main internet activity that currently wishes to avoid detection on a network. This means that the unknown flows selected by the P2P heuristics account for its lower 'classified correct' rating and also suggests that the host heuristics are selecting many more P2P flows than can currently be classified by application signatures and ports. This highlights the advantage and reasoning behind applying heuristic host based classifications instead of just using application signatures.

VII. CONCLUSIONS

This paper has introduced a new method of detecting and separating P2P and non-P2P flows on a network by analysing the behaviour of the hosts at either end of the connection. The host behaviours are examined in 3 ways to maximise the classification potential when not all of a host's activity is witnessed across the analysed network point. These are the source host, destination host, and both hosts. This information is used to identify the protocol behind a specific flow rather than a specific host. A classification strategy was formed containing 12 heuristics and tested on real network traffic. The results showed extremely accurate classifications with a 0.54% misclassification rating the highest witnessed for web flows and 0.15% in P2P flows. The accuracy of the strategy described in this research shows much potential for the use of host activity to determine a flows protocol.

Future work will concentrate on further heuristics to increase the 'success rate' of P2P classifications. This will include the investigation and analysis of further host based heuristics. Further to this the classification scheme already proposed will be investigated for real-time purposes. If similar classifications can be made by monitoring hosts for a limited time period from the start of a flow then this strategy has potential to be used in a real-time, online network classification system.

REFERENCES

[1] Madhukar, C. Williamson, C. 2006. A Longitudinal Study of P2P Traffic Classification. In proceedings of the 14th IEEE International Symposium on Modeling, Analysis, and Simulation, pp. 179-188

[2] Iana port numbers, http://www.iana.org/assignments/port-numbers

[3] Kazaa, http://www.kazaa.com

[4] Ocampe, R. Galis, A. Todd, C. De Meer., H. 2006. Towards Context-Based Flow Classification. Autonomic and Autonomous Systems, ICAS '06, pp. 44-44

[5] Sen, S. Spatscheck, O. Wang, D. 2004. Accurate, Scalable In-Network Identification of P2P Traffic Using Application Signatures. In WWW2004

[6] Ohzahata, S. Hagiwara, Y. Terada, M. Kawashima, K. 2005. A Traffic Identification Method and Evaluations for a Pure P2P Application. PAM 2005: passive and active network management. LNCS, vol. 3431, pp. 55-68. Boston MA

[7] BitTorrent – A technical description of the bitTorrent protocol, http://www.cs.chalmers.se/~tsigas/Courses/DCDSeminar/Files/BitTorrent.pdf

[8] Bernaille, L. Teixeira, R. Akodjenou, I. Soule, A. Salamatian, K. 2006. Traffic Classification On The Fly. ACM SIGCOMM Computer Communication Review, Vol. 36, Issue 2

[9] Constantinou, F. Mavrommatis, P. 2006. Identifying Known and Unknown Peer-to-Peer Traffic. In FIFTH IEEE INTERNATIONAL SYMPOSIUM ON NETWORK COMPUTING AND APPLICATIONS, pp. 93-102

[10] Karagiannis, T. Broido, A. Faloutsos, M. Claffy, Kc. 2004. Transport Layer Identification of P2P Traffic. In PROCEEDINGS OF THE 4TH ACM SIGCOMM CONFERENCE ON INTERNET MEASUREMENT (IMC 2004), ITALY, OCTOBER 2004, PP. 121-13

[11] Karagiannis, T. Papagiannaki, K. Faloutsos, M. 2005. BLINC: Multilevel Traffic Classification in the Dark. Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications, Philadelphia, Pennsylvania, USA, August 22-26

[12] Basher, N. Mahanti, A. Mahanti, A. Williamson, C. Arlitt, M. 2008. A Comparative Analysis of Web and Peer-to-Peer Traffic. Proceeding of the 17th international conference on World Wide Web, China, pp. 287-296

[13] Karagiannis, T. Broido, A. Faloutsos, M. Claffy, Kc. 2004. File-sharing in the Internet: A characterization of P2P traffic in the backbone. Technical report. www.cs.usr.edu/~tkarag

[14] Gnutella – The Gnutella Protocol Specification v0.4, http://www9.limewire.com/developer/gnutella_protocol_0.4.pdf

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7229
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7229