Jurnal
Teknologi

Full Paper

ONLINE PEER-TO-PEER TRAFFIC IDENTIFICATION BASED ON COMPLEX EVENTS PROCESSING OF TRAFFIC EVENT SIGNATURES

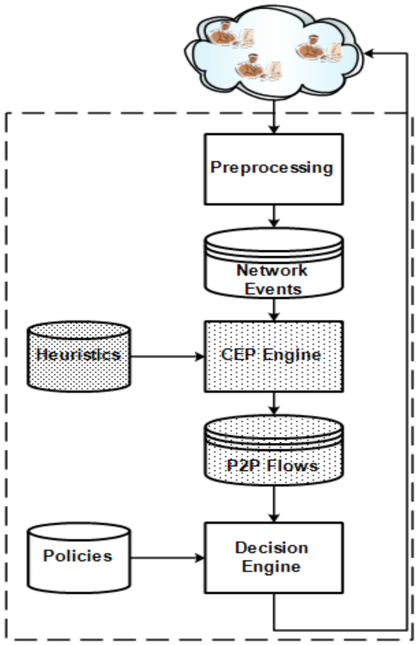
Joseph Stephen Bassi, Loo Hui Ru, Ismahani Ismail, Ban Mohammed Khammas, Muhammad Nadzir Marsono*

Faculty of Electrical Engineering, Universiti Teknologi Malaysia, 81310 UTM Johor Bahru, Johor, Malaysia

Article history
Received
12 May 2015
Received in revised form
20 November 2015
Accepted
15 June 2016

*Corresponding author
mnadzir@utm.my

Graphical abstract



Abstract

Peer-to-Peer (P2P) applications are bandwidth-heavy and lead to network congestion. The masquerading nature of P2P traffic makes conventional methods of its identification futile. In order to manage and control P2P traffic efficiently preferably in the network, it is necessary to identify such traffic online and accurately. This paper proposes a technique for online P2P identification based on traffic events signatures. The experimental results show that it is able to identify P2P traffic on the fly with an accuracy of 97.7%, precision of 98% and recall of 99.2%.

Keywords: Complex event processing, P2P, Traffic events heuristics

Abstrak

Aplikasi perangkai padan (P2P) adalah aplikasi jalur lebar yang menyebabkan kesesakan rangkaian. Sifat kebolehsamaran trafik P2P membuatkan kaedah pengenalpastian secara konvensional sia-sia. Bagi mengurus dan mengawal trafik P2P dengan cekap dalam rangkaian, adalah perlu untuk mengenalpasti trafik aplikasi tersebut secara dalam talian dan tepat. Kertas ini mencadangkan satu teknik untuk mengenal pasti trafik P2P talian berdasarkan tandatangan trafik peristiwa. Keputusan eksperimen menunjukkan bahawa ia dapat mengenal pasti trafik P2P dengan cepat dengan ketepatan 97.7%, kepersisan 98% dan perolehan kembali 99.2%.

Kata kunci: Pemprosesan acara kompleks, P2P, Peristiwa lalu lintas heuristik

1.0 INTRODUCTION

Data-only networks are now characterized with sophisticated systems comprised of multi-vendor equipment's, supporting multi-technology and capable of providing a wide range of real-time media applications at extremely high speeds [1]. This development has also encouraged the growth of Peer-to-Peer (P2P) applications on the network widely applied to bandwidth-heavy file sharing, online gaming and other applications, causing a concern to network administrators. Today, P2P file sharing networks account for more than 60% of the Internet traffic [2], with significant bandwidth consumption, aiding to the poor quality-of-service (QoS) for other network traffics. Hence, the issue of accuracy is one of the prevailing research topics in network management. Identifying P2P traffic especially by

Internet service providers (ISPs) is paramount for achieving appropriate QoS, which can be achieved through traffic shaping and traffic policing, enabling appropriate allocation of network resources to deliver optimal performance for end users.

A number of techniques have been proposed for P2P identification. Machine learning method [3-6] which make use of statistical flow features and heuristics methods [7, 8] that are based on host behaviours, are the most promising techniques. Nevertheless, these techniques are not for real-time (online) traffic identification because they are computational intensive and also require correlating past data samples. Thus, to detect network traffic on the fly, the system has to be able to detect traffic online. This is not only to improve QoS and adequate resource allocation, but also to boost security, accounting, traffic engineering, Class-of-Service (CoS) offerings and also provide a system with application-aware network flow processing.

In this paper, an approach to classify P2P traffic using Complex Event Processing (CEP) system is proposed. Traffic is classified based on transient or emerging patterns as they arrive. The targeted system uses CEP to classify network traffic as P2P or non-P2P by consolidating traffic connection heuristics. Our proposed system has been applied to UNIBS dataset [9] in order to evaluate the accuracy and performance of the system. Our proposed system has the ability to classify network traffic online with an accuracy of 97.7%, precision of 98% and recall of 99.2%.

The remainder of this paper is organized as follows: Section 2 presents related works on P2P identification. Section 3 presents the discussion on traffic events for P2P identification. Section 4 explains our proposed method. Section 5 presents our experimental results and discussions. Conclusion is in Section 6.

2.0 RELATED WORKS

Recent P2P applications have evolved to the use of arbitrary port numbers, port hopping, chunked file transfers and encrypted payloads as obfuscation means to avoid identification [10]. The detection of P2P applications have evolved from the use of dedicated known ports for centralized p2p architecture to dynamic ports for distributed and hybrid p2p architecture, which is a combination of centralized and distributed architecture [10]. One of the motivations for swift evolution of P2P and its applications is its high usage in file-sharing, gaming and multimedia applications today. Effective classification/identification of P2P traffic can enhance efficient network management and prudent utilization of network resources [8].

Conventional P2P traffics that use default port numbers can be easily detected and classified by matching their port numbers [11]. This method is simple and fast but is limited to classify today's P2P traffic since more and more P2P applications dynamically use arbitrarily port numbers and also hiding of their identity (masquerading). Reference [12] reported that only 30% P2P connections use default port numbers. Reference [10] reported that port-based examination is incapable of identifying 30–70% of the Internet traffic flows that they examined.

To complement the port-based P2P identification, signature-based [12] techniques have been proposed. This technique exploits specific strings in of packets payload to identify P2P traffic. Though this method have a high specificity, it performs poorly on encrypted payloads or unknown P2P traffic with unseen strings [6]. This method also requires high storage. Reference [13] has demonstrated that signature-based method achieves high accuracy of 96% for unencrypted p2p traffic, but the accuracy on encrypted P2P traffic is only between 30% and 70%. This method is not suitable to classify current or future networks P2P traffics that are mostly encrypted.

The use of machine learning and heuristic based techniques have been suggested to overcome the limitation and to complement other techniques i.e. signature-based and port-based. While machine learning methods classify network traffic based on extracted features from traffic, heuristic based methods use the communication patterns of connecting hosts [14-16].

1) The machine learning approaches mine traffic flow features such as flow duration, packet inter arrival time and packet size to classify network traffic. Reference [17] has presented nonlinear analysis to obtain self-similarities and long range correlation statistical features to classify classes of network traffic. However, this technique requires complex computation which is not suitable for online traffic identification.

2) The heuristic approach looks at the communication patterns established connecting hosts and compare them to the behaviours exhibited by different network applications traffic classes [6]. Reference [18] has proposed a three-class heuristic classification which is based on the connection patterns discussed in references [19], [20] and [21]. These methods are not suitable for online classification because the use of heuristics usually relies on features from off-line data to correlate peer connection patterns.

Distributed data processing systems for network traffics today are posed with the challenges of processing in-flight or streaming data with large volume, variety and high speed [22]. Attempting to store these data and mine them later produces excess computation and large memory requirement. The complex event processing engines provides the ability to process vast amount of streaming data with reduced latency, and also have the ability to include temporal, causal and structural relations between incoming events in-flight [23]. The benefits of using CEP engines for online data stream processing presented in reference [22] include:

1) Converting raw data into actionable information swiftly to either avoid losses, network state information or lose momentary evolving traffic in the network.
2) Identification of transient or emerging patterns, which cannot be identified with offline data mining techniques.
3) Removal of unwanted data in the pipeline to save memory, Central Processing Unit (CPU), storage and energy cost.

CEP acts by processing simple events generated by sources (event producer), extracting new knowledge in the form of composite events (complex events), and delivering them to interested sinks (event consumers). Event-based applications often involve a large number of sources and sinks, possibly dispersed over a wide geographical area. The ability of CEP to provide efficient processing of several heterogeneous events with very high throughput rates, scalability and adaptability [24].

Our proposed method identifies and aggregates simple traffic events into complex event heuristics using Complex Event Processing (CEP) system on the fly. This is to complement the existing methods of P2P traffic identification techniques. To the best of our knowledge no works have employed this technique for P2P classification.

3.0 NETWORK TRAFFIC EVENTS FOR P2P IDENTIFICATION

An event as defined in reference [25] is an occurrence within a particular system or domain; it is something that has happened, or is contemplated as having happened in that domain. An event in itself provides a little if any information to the end user. Complex event is when two or more events are combined (processed) to form a complex object with a higher degree of inference, or knowledge with value added information to end users. The processing of these single events depends on the detection of structural, temporal or special patterns [23].

P2P traffic have specific connection patterns which make it differentiable from other background traffic [18]. Some of these patterns include the concurrent use of TCP and UDP protocols and the random use of port numbers.

The idea of P2P heuristics for traffic classification based on transport and network layer headers is clearly stated in [20] and [21]. A flow is defined as packets with same five header tuples (source IP, destination IP, source port, destination port, protocol), while a pair is defined by either source or destination (IP, Port) of a packet [18].

A P2P host uses listening port to inactively awaits connections from other pairs after initiating a connection by advertising its (IP, Port) pairs to other host. This host in turn broadcasts the advertised (IP, Port) to other hosts on the network. These hosts use random source Port numbers to establish P2P connection to the listening host.

3.1 Events Definition

P2P connectivity can be modeled as a directed graph, represented by $\vec{G}$ = (V, E), having $v_i \in V$, and $e_{(s,d)} \in E$. $e_{(s,d)}$ is defined as number of active flows between source (V_s) and destination (V_d).

Events 1: For every advertised destination (IP, Port) pair of a host, if the number of distinct IPs connected to it equals to the number of distinct ports used to connect to it, they are marked as P2P connection.

$$e_{(s,d)} = \begin{cases} P2P, & s = d \\ non-P2P, & s \neq d \end{cases} \qquad (1)$$

Events 2: For every advertised destination (IP, Port) pair of a host, For every source (IP, port) or destination (IP, port) pairs, if the difference between the number of connected IP's and ports is less than n and the number of connected IP's is greater than m, where m and n are variables that will be described in Section 4.2.

$$e_{(s,d)} = \begin{cases} P2P, & s - d < n \,\&\, s > m \\ non-P2P, & otherwise \end{cases} \qquad (2)$$

4.0 PROPOSED SCHEME

A block representation of our proposed system is represented in Figure 1. Traffic events are sensed and fed into the proposed system for processing, and the result of the processing in the form of policies or control are fed back into the network in to timely tune/ adjust the system.

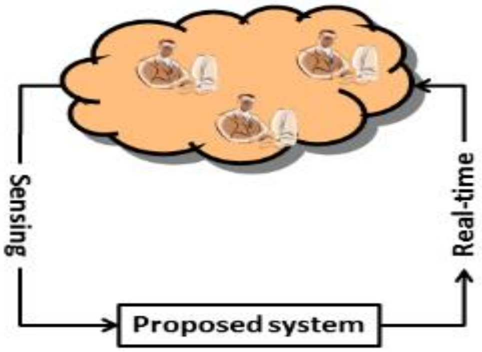


Figure 1 System block diagram

4.1 Monitoring Framework for In-network Monitoring

In order to support the in-network monitoring, we propose a P2P identification paradigm using CEP that will enhance traffic management functions. Figure 2 shows general architectural elements of the proposed in-network monitoring framework.

The system architecture is composed of three distinct layers: preprocessing layer, event detection layer and decision layer. Each of these layers, are segmental and flexible. This is to make the system flexible such that a change in any of the layer does not affect the other layers.

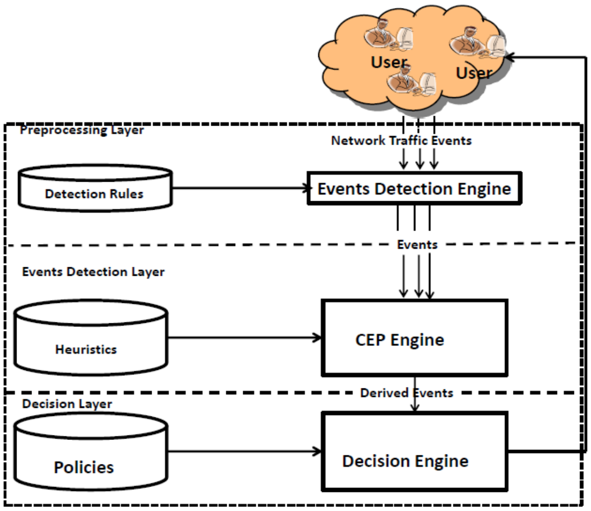


Figure 2 Proposed system architecture

4.1.1 Preprocessing Layer

The top level, the preprocessing layer acts as the screening point for real-time packets. Here, traffic packets are buffered, examined and filtered to identify suspicious packets based on set rules. Identified valid packets are transmitted to the next layer indicating the occurrence of an event. Other non-related processes are forwarded to egress port. This level is composed of a complete network monitoring/packet analyzing tool [26] which performs pre-processing of network packets based on a set of predefined rules.

4.1.2 Event Detection Layer

The event detection layer determines if the different events received from the preprocessing layer represent either P2P traffic or non-P2P traffic. This layer takes advantage of the events sent by the preprocessing layer, providing a higher level vision of the raw data transmitted through the network.

The event detection layer is a CEP engine Esper [27]. Esper performs the task of correlating the identified events to recognize special or time-based relationships among seemingly uncorrelated events that would have been detected by the packet analyzer. It performs this task on the basis of SQL-like queries Event Processing Language (EPL) that can be configured at run time. Queries are set based on network connection heuristics (as in Section 3.1). When sending the packet information/flow to the event detector (CEP), the order of the captured packets is maintained in the order of packet capture. This is necessary for evaluating sequence operators in the EPL queries.

4.1.3 Decision Layer

The resolutions of the detection layer in the form of derived events that are received by the decision layer. The decision layer is composed of a database with policies/decision list and the decision engine. The database stores the information/policies that describes how the system should act based on the derived events (complex events) being detected.

Appropriate decisions are implemented into the system in the form of control or policies by this level. The set of actions to be imposed into the system are determined to match the derived events with the appropriate decisions stored in the policies database of the decision engine. Decisions will be made based on the administrative policy, which is beyond the scope and will not be discussed in this paper. Readers are advised to refer to references [28] and [29] regarding network administration policies and functions.

4.2 Proposed Event Detection Algorithm

Combining events 1 and 2 yields our proposed complex event heuristic for the identification of P2P flows. Our algorithm is designed using Esper CEP system using [9] real traffic traces.

Algorithm 1 presents the procedure of event detection layer of our proposed monitoring architecture. Traffic identification starts upon receiving an incoming traffic flow (N). The number of distinct IP's (k) and ports (m) are calculated to identify the flow statistics. The events queries are the used to classify each flow based on classification conditions. The output of the classification will be in the form of derived events (P2P or non-P2P flows).

Algorithm 1 Complex event algorithm for P2P flow identification

```
$N$ : $All\ flow\ with\ 5\ tuples$ (time, sourceIP, sourcePort, destinationIP, destinationPort)
$W$ : $Advertised\ destination\ pair$
$k$ : $Distinct\ number\ of\ IPs$
$m$ : $Distinct\ number\ of\ ports$

For all $N$ do
    If $k = m$ then
        identify as P2P
    else if $(k - m < 2\ \&\ k > 5)$
        identify as P2P
    else
        Non-P2P
    end if
end for
```

5.0 RESULTS AND DISCUSSION

5.1 Evaluation Metrics

For evaluation, we supposed there are two traffic classes of P2P and non-P2P in internet traffic. A P2P classifier based on events heuristics is used to identify and classify if a flow is either P2P or Non-P2P. For our proposed technique, the experiments are evaluated using Accuracy, Recall, and Precision metrics as shown in Table 1.

Table 1 Evaluation Metrics

		Predicted class	
		P2P	**Non-P2P**
Actual Class	**P2P**	True Positive (TP) Correctly classified results	False Negative(FN) Missing results
	Non-P2P	False Positive(FP) Wrong classified result	True Negative(TN) Correct absence of result

1. Accuracy: the fraction of correctly identified P2P flow to all results given as

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad (3)$$

2. Recall: the fraction of accurately identified P2P to the sum of the correct and wrong classified results which is given as

$$Recall = \frac{TP}{TP+FN} \qquad (4)$$

3. Precision: the fraction of accurately identified P2P traffic to all positive results given as

$$Precision = \frac{TP}{TP+FP} \qquad (5)$$

5.2 Dataset

In order to evaluate the performance of our proposed technique, we implemented our proposed method using a real network dataset. The UNIBS dataset [9] is used in this research. The dataset was collected for three successive working days at the edge router of the campus network of the University of Brescia. The dataset pcap file containing packet header files with its associated ground truth. The dataset consists of Web (http and https), Mail (POP3s, and IMAPs) and P2P (bittorrent, edonkey, skype). For the purpose of this research, we divide the network dataset into two main classes: P2P and non-P2P applications. The composition of UNIBS dataset used is presented in Table 2.

Table 2 The composition of UNIBS dataset

Application	Number of flow	Flow ratio %
P2P	25990	32.9
Non-P2P	53008	67.1

5.3 Experimental Setup

In our experiment, labeled dataset [9] as traffic flows are injected into CEP as streams. The heuristics (H1, H2 and H3) in the form of event queries written in Java programming language are stored in database used by the event processor to detect events. These heuristics are host behaviours patterns exhibited during connection [6, 14, 15]. Figure 3 presents our proposed experimental setup.

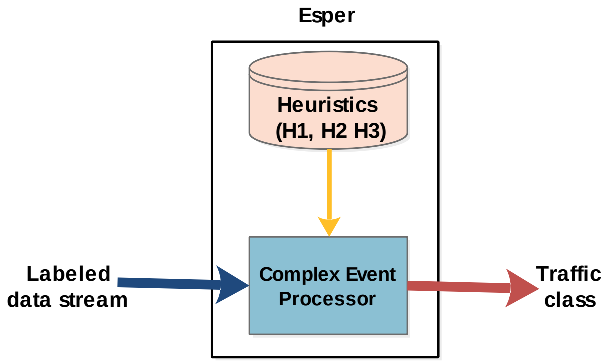


Figure 3 Proposed experimental setup

5.4 Heuristics Performance Measure

In order to analyze the performance of the proposed method, we evaluated and compared three schemes of event 1(H1), event 2 (H2), and Complex event (H3). The experiments were performed using the Esper 4.11.0 complex event processor.

The number of correctly classified P2P flows for each of the heuristics (H1, H2, and H3) against the ground truth (GT) and their composite plots are presented in Figures 4, 5 and 6, respectively. These figures illustrate the plots for each of the heuristics indicating the pattern of identification for each of the heuristics. The plots indicate that the proposed method have a better detection rate per time interval in comparison to H1 and H2. The complex event (H3) outperforms the single/simple events (H1 and H2) by identifying the packets with a higher accuracy, precision and recall. This is because the shortcoming of individual event heuristic is complimented by the complex heuristics. Table 3 presents a summary of the measurement metrics used for our methodology.

Table 3 Summary of results

Heuristic	Accuracy (%)	Precision (%)	Recall (%)
H1	80.96	99	77.4
H2	93.61	98	96.6
H3	97.7	98	99.2

The overall performance indicates that CEP system is promising for in-network monitoring, and also suitable to monitor the present/future complex and dynamic network systems.

Figure 7 summarizes the performance comparison of our proposed method using complex events with the use of single events as proposed by references [21] and [18]. Although the proposed method has the same precision with H2, it outperforms both H1 and H2 in accuracy and recall. This indicates that the CEP system can enhance the detection rate of simple event heuristics by combining (processing) them.

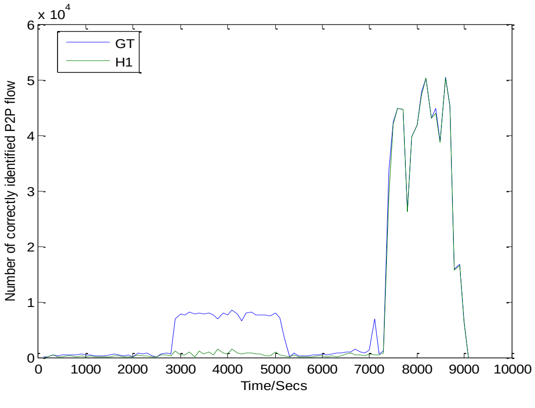


Figure 4 Plots of H1and GT

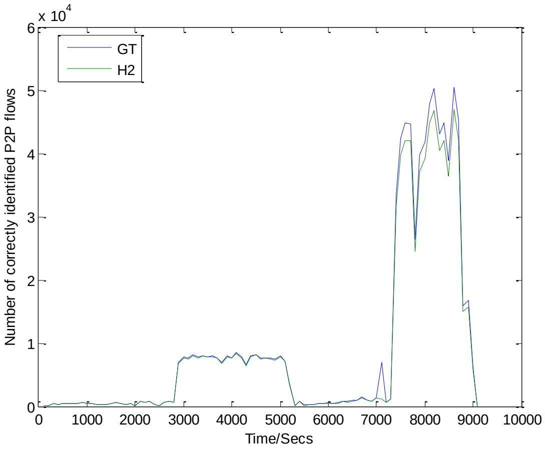


Figure 5 Plots of H2 and GT

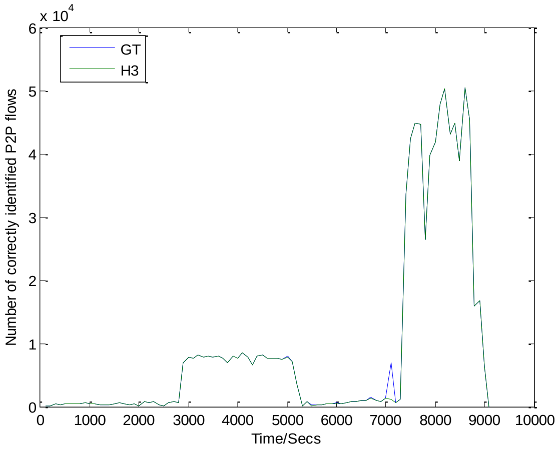


Figure 6 Plots of H3 and GT

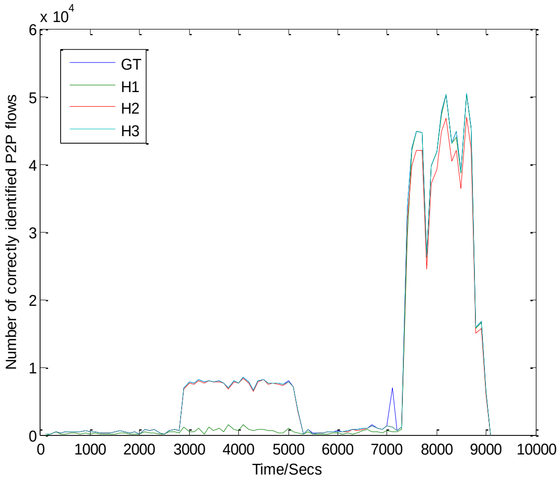


Figure 7 Composite plots of H1, H2, H3 and GT

The percentage of error plot for each heuristic is presented in Figures 8, 9 and 10. H3 outperforms the single events H1 and H2 with the lowest percentage error over time. This indicates that the complex event has a better error reduction rate compared to the simple singular events.

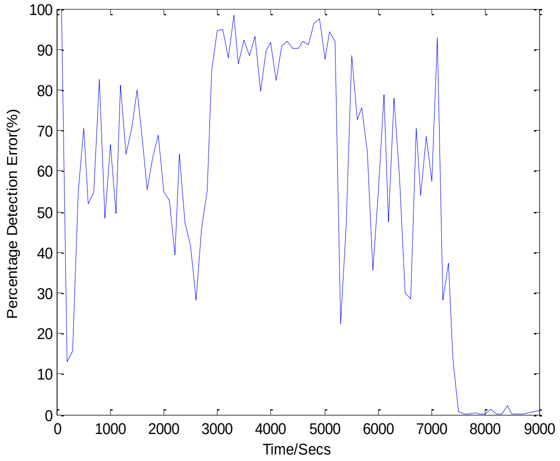


Figure 8 H1 Percentage error plot

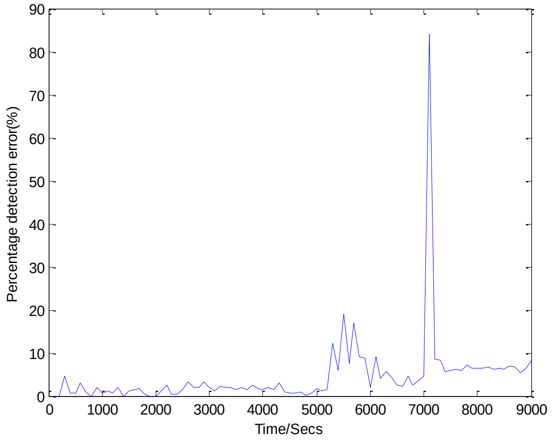


Figure 9 H2 Percentage error plot

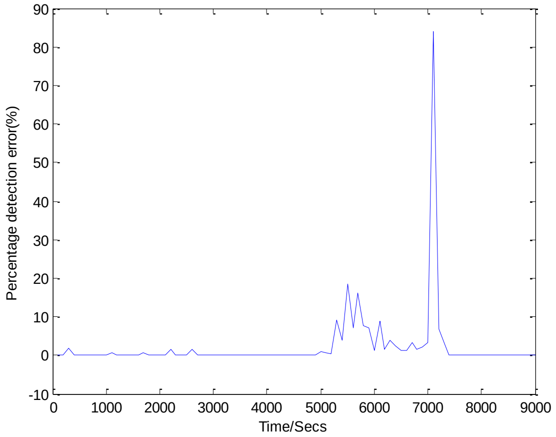


Figure 10 H3 Percentage error plot

6.0 CONCLUSION

This paper proposed an online P2P identification technique based on traffic events connection patterns. The system identifies network traffic on the fly based packet header information. In contrast to existing methods, our technique exploited the capabilities of the CEP system to classify P2P traffic.

The performance of our technique was estimated with real network traces. The experimental results show that we are able to accurately classify P2P applications with an accuracy of 99.7%, precision of 98% and recall of 99.2%. However, it has and a false discovery rate of 0.2%. In future we will use additional heuristics and additional information for the specific applications and achieve a better performance P2P classification.

Acknowledgement

This work is supported in part by Ministry of Science, Technology and Innovation ScienceFund Grant (UTM Vote No 4S095).

References

[1] N. Samaan and A. Karmouch. 2009. Towards Autonomic Network Management: an Analysis of Current and Future Research Directions. *Communications Surveys & Tutorials, IEEE*. 11: 22-36.

[2] J. Yan, Z. Wu, H. Luo, and S. Zhang. 2013. P2P Traffic Identification Based on Host and Flow Behaviour Characteristics. *Cybernetics and Information Technologies*. 13: 64-76.

[3] S. Deng, J. Luo, Y. Liu, X. Wang, and J. Yang. 2014. Ensemble Learning Model For P2P Traffic Identification. *Fuzzy Systems and Knowledge Discovery (FSKD), 2014 11th International Conference on*. 436-440.

[4] H. Liu, W. Feng, Y. Huang, and X. Li. 2007. A Peer-to-peer traffic Identification Method Using Machine Learning. *Networking, Architecture, and Storage, 2007. NAS 2007. International Conference on*. 155-160.

[5] N. Namdev, S. Agrawal, and S. Silkari. 2015. Recent Advancement in Machine Learning Based Internet Traffic Classification. *Procedia Computer Science*. 60: 784-791.

[6] W. Ye and K. Cho. 2014. Hybrid P2P Traffic Classification With Heuristic Rules And Machine Learning. *Soft Computing*. 1-13.

[7] J. M. Reddy and C. Hota. 2015. Heuristic-based Real-Time P2P Traffic Identification. *Emerging Information Technology and Engineering Solutions (EITES), 2015 International Conference on*. 38-43.

[8] Y. Wujian and C. Kyungsan. 2013. Two-Step P2P Traffic Classification with Connection Heuristics. *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2013 Seventh International Conference on*. 135-141.

[9] UNIBS. 2009. Available: http://www.ing.unibs.it/ntw/tools/traces/.

[10] A. Madhukar and C. Williamson. 2006. A Longitudinal Study Of P2P Traffic Classification. *Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, 2006. MASCOTS 2006. 14th IEEE International Symposium on*. 179-188.

[11] A. W. Moore and D. Zuev. 2005. Internet Traffic Classification Using Bayesian Analysis Techniques. *ACM SIGMETRICS Performance Evaluation Review*. 50-60.

[12] S. Sen, O. Spatscheck, and D. Wang. 2004. Accurate, Scalable In-Network Identification Of P2p Traffic Using Application Signatures. *Proceedings of the 13th international conference on World Wide Web*. 512-521.

[13] X.-B. Liu, J.-H. Yang, G.-G. Xie, and Y. Hu. 2009. Automated Mining Of Packet Signatures For Traffic Identification At Application Layer With Apriori Algorithm. *J Commun*. 29: 51-59.

[14] Z. Chen, Z. Liu, L. Peng, L. Wang, and L. Zhang. 2015. A Novel Semi-Supervised Learning Method For Internet Application Identification. *Soft Computing*. 1-13.

[15] L. M. Nair and G. Sajeev. Internet Traffic Classification by Aggregating Correlated Decision Tree Classifier.

[16] W. Ye and K. Cho. 2014. Hybrid P2P Traffic Classification With Heuristic Rules And Machine Learning. *Soft Computing*. 18: 1815-1827.

[17] F. Palmieri and U. Fiore. 2010. Insights into peer to peer traffic through nonlinear analysis. *Computers and Communications (ISCC), 2010 IEEE Symposium on*. 714-720.

[18] R. Zarei, A. Monemi, and M. N. Marsono. 2013.n Automated Dataset Generation for Training Peer-to-Peer Machine Learning Classifiers. *Journal of Network and Systems Management*. 1-22.

[19] M. Perényi, T. D. Dang, A. Gefferth, and S. Molnár. 2006. Identification and Analysis Of Peer-To-Peer Traffic. *Journal of Communications*. 1: 36-46.

[20] W. John and S. Tafvelin. 2008. Heuristics to Classify Internet Backbone Traffic Based On Connection Patterns. *Information Networking, 2008. ICOIN 2008. International Conference on*. 1-5.

[21] T. Karagiannis, A. Broido, and M. Faloutsos. 2004. Transport layer identification of P2P Traffic. *Proceedings of the 4th ACM SIGCOMM Conference On Internet Measurement*. 121-134.

[22] E. Olmezogullari and I. Ari. 2013. Online Association Rule Mining over Fast Data. *Big Data (BigData Congress), 2013 IEEE International Congress on*. 110-117.

[23] B. Tarnauca, D. Puiu, D. Damian, and V. Comnac. 2013. Traffic Condition Monitoring Using Complex Event Processing. *System Science and Engineering (ICSSE), 2013 International Conference on*. 123-128.

[24] B. Tarnauca, D. Puiu, S. Nechifor, and V. Comnac. 2013. Using Complex Event Processing for implementing a geofencing service. *Intelligent Systems and Informatics (SISY), 2013 IEEE 11th International Symposium on*. 391-396.

[25] O. Etzion and P. Niblett. 2010. *Event Processing In Action*: Manning Publications Co.,

[26] Wireshark. 2015, 3 October. *Packet Analyzer*. Available: https://www.wireshark.org/

[27] EsperTech. 2014. *Esper and NEsper*. Available: http://www.espertech.com/esper/index.php

[28] A. Farrel. 2011. *Network Management Know It All*: Elsevier,

[29] D. C. Verma. *Principles Of Computer Systems And Network Management*. Springer.