

A Framework for Tunnelled Traffic Analysis

Taner Yildirim, Dr. PJ Radcliffe,

School of Electrical & Computer Engineering, RMIT University, Australia

taner.yildirim@rmit.edu.au, pjr@rmit.edu.au

Abstract—Research in traffic classification is reaching into ever more difficult areas. Traditional techniques such as header and payload inspection are not providing sufficient information due to usage of non-standard ports and encryption. Promising alternative methods have been proposed based on the statistical behaviour of traffic flows. Although these methods can achieve quite high accuracies in non-encrypted traffic flows, traffic identification of encrypted traffic flows is still in its early stages. We argue that the results to date for encrypted traffic cannot help a network device such as a firewall make any useful decision, nor are there any indications that this may be achieved in the future. We propose a novel approach to cope with encrypted peer to peer network layer tunnels which are a particular problem in schools, universities, and larger corporate networks. First statistical techniques are used to identify the protocols present, a process that may take in the order of seconds. Next, based on the protocols discovered, and enterprise policies, a network device is advised to block, band-limit, or allow the whole tunnel, or a range of packet sizes within that tunnel. Preliminary research has concluded that VoIP traffic can be successfully handled by this approach and that advise to a network device can be practically useful. Work continues to apply these techniques to other protocols and mixes of protocols within a peer to peer tunnels.

Index Terms—Traffic Classification, VPN, Encrypted Traffic, IPSec, VoIP

I. INTRODUCTION

With the evolution of Internet there has been a rapid expansion in the number of applications and the way these applications use encryption, protocols and ports. In the face of these changes, traditional methods for traffic classification have become inadequate thus making network management ever more challenging.

Traditionally traffic classification is achieved by packet inspection. Packets are analysed from their headers or payloads. However, information extracted from packet headers can be misleading. For example, emerging applications use random port numbers instead of using standard ports. Some application types can reuse standard ports, such the web-browsing ports, for their own purposes. As the number of application types increase, algorithms used to identify packet payloads become more complex and the trade-off between computational complexity and accuracy becomes an issue.

Encrypted traffic packet inspection is quite difficult since the only obvious available parameters are packet lengths and inter-packet arrival times. These two parameters led the research community to study statistics of traffic flows rather than network specific or application specific properties of packets. Where the traffic flow is unencrypted, using just packet length and inter-arrival time allows traffic classification to achieve

high accuracies when Machine Learning (ML) Algorithms are used. A survey paper [1] identifies 18 significant works done in ML field from 2004 till 2007. Interestingly, most of the work done in ML algorithms is composed of non-encrypted traffic with protocols such as Voice over IP or Video. Streaming traffic within encrypted tunnels has not yet been investigated thoroughly.

Although not studied extensively, these scenarios are extremely relevant to recent networks. Most corporate networks employ Voice over IP systems as well as Video Conferencing and use VPN to secure the packets from intruders. VPNs are commonly used instead of leasing expensive dedicated lines but in terms of network management, VPNs are quite challenging. A VPN may be used to disguise large downloads or other forms of traffic that would normally be banned by enterprise policies as enforced by proxy servers, network optimizers, or other network devices. Successful monitoring and classifying of VPN traffic is thus quite critical to network management. The literature to date on encrypted traffic identification research has focused on SSH [2] traffic.

In this paper we discuss existing approaches for encrypted and non-encrypted traffic identification using ML techniques as well as other algorithmic approaches. We then propose some new algorithms for identifying traffic within encrypted tunnels and new research directions which have not been studied or only partially studied. The rest of this paper organized as follows. Section II reviews the relevant research for traffic classification using Machine Learning Algorithms. Section III reviews Encrypted Application classification techniques. In both cases, achieved accuracies are discussed. Section IV outlines the classification problem in virtual private networks. Section V discusses promising preliminary results and plans for a more comprehensive research program. Section VI concludes the paper and considers some further implications of identifying encrypted traffic.

II. CURRENT RESEARCH

A. Machine Learning for IP traffic Classification

The concept of Machine Learning is considered to be in the field of artificial intelligence, and hence it can be defined as a computing process of acquiring new knowledge depending on existing knowledge. As defined in [1] it is a process of structural pattern recognition from given data. Machine Learning techniques were first used in traffic classification for intrusion detection [3]. For our purposes we use three sub-fields of Machine Learning Algorithms, these are Supervised Learning

Algorithms, Clustering Learning Algorithms (Unsupervised) and Hybrid Learning Approaches.

1) *Supervised Machine Learning*: Supervised algorithms are the type of algorithms which are fed with a so called training set (i.e. a data set in which there exists a pair of data consisting of input and desired outputs). In this sense accuracy of supervised algorithms would depend on the provided data set since the following decisions will be made upon the provided set. If the network traffic considered as homogeneous (i.e. does not change its behaviour dramatically in time) this type of algorithms can be employed.

One of the promising methods is provided by Crotti et al [4], where supervised learning used with three extractable properties of IP packets; packet length, inter-arrival time and order of arrival. They used so called protocol fingerprints which are extracted from PDF vectors. Upon constructing the PDF vectors, they used anomaly scores from 0 to 1 in order to probabilistically estimate which incoming flow can be classified for a given PDF vector. Results show that with 91% accuracy they can identify HTTP, SMTP and POP3 protocols. Although promising, they have not addressed the issue of packet loss and jitter in their experiments. For applications like VoIP packet loss and jitter is quite critical. They also used HTTP as an application in which many new arising application types can be hidden. They also assumed that starting point of the flows can be caught, in the case of online deployment there is always a case of missing the beginning of the flows. Such capture may not be possible outside a VPN.

A supervised approach used by Li [5] uses C4.5 [6] decision algorithm. They categorized the Internet flow using classes such as web-browsing, mail, bulk ftp, attack, peer-to-peer, database, service, and interactive. They achieved 99% accuracy using 1875 flows as a training set. They also addressed the issue of accuracy versus complexity. Although mentioned there is no data available for VPN traffic in their paper.

In 2005 Moore [7] used Naive Bayes technique to classify application types. They used hand-classified data to be certain of categories. They also gathered various feature sets such as flow length, port number, inter-arrival time etc. However, authors made the assumption that features such as port number is available which would not be the case in tunnelled streams. Also, pre-classified traffic can be misleading since traffic patterns can change dramatically in time. Their internet traffic classifications include bulk data, p2p, database, interactive, WWW, mail, games and multimedia. They maintained 95% overall accuracy using Fast Correlation-Based Filter (FCBF) methods. However, we have not been able to see any application for VPN traffic scenarios.

All machine learning approaches concentrate on packet type identification and result in a probabilistic measure that a given packet serves a particular protocol. Such a probabilistic measure is not of great use within real network devices. Consider a VoIP stream where 90% of packets were correctly identified, the 10% misidentified and blocked would render VoIP unusable. Even in an FTP download the recovery from lost packets would be an unwanted addition to the network

traffic.

2) *Clustering Learning*: Clustering has been widely used for application classification. One of the earliest papers in this area is proposed by McGregor et al. [8], in which application types such as HTTP, FTP, SMTP, IMAP, NTP and DNS are studied. The authors did not attempt to classify each and every application by itself rather their scheme used bulk transfer, small transaction and multiple transactions as classification. Therefore this paper suggested similarity patterns in a given traffic. However, a better classifier should have the capability of decision making which would include accelerating and/or blocking individual application types such as VoIP. They used Expectation Maximization [9] algorithm for clustering.

One other paper by Zander and Armitage [10], used AutoClass [11] algorithm for clustering and they studied game traffic, Napster, AOL, HTTP, SMTP, Telnet and FTP traffic. Their classifier was application specific however; one of the issues about their proposal was that, given a number of application types their classifier found a high number of clusters. So, cluster to application mapping was a critical problem for the efficiency and accuracy of the classifier.

Another attempt using the clustering method was proposed by Bernaille et al [12], in which K-means algorithm was employed. K-means algorithm is a clustering algorithm which clusters the given data using a distance vector scheme. Their classifier was TCP based and therefore other transport protocols such as UDP could not be employed. However, in tunnelled networks transport protocol information is completely invisible. One other drawback of their proposal was the assumption that the classifier can always get the first few packets of the flow. However, this would not be the case all the time because in probabilistic methods a packet may be misclassified and so miss the start of a transaction. A good classifier should catch the applications at any time. They studied FTP, HTTP, Kazaa, NTP, POP3, SMTP, SSH, HTTPS, POP3S.

One recent paper by Erman et al [13] also used K-means algorithm for clustering and they studied web, p2p and ftp traffic. In their study they assumed that training flows can be identifiable from their payload and header information. One other work by Junior et al [14], just studied p2p traffic and tried to distinguish p2p traffic from the rest of the application types. However, we could not see any further implementations on VPN traffic.

Some researchers have studied several application types and some studied only one type. From the application type prospective we can state that file transfer, web, e-mail, p2p type protocols have been studied. Only a few researchers dealt with encrypted data however these consist of application level encryption protocols which are tunnels containing only one protocol. Such approaches will not suit Peer to Peer network layer VPNs such as IPsec which may contain several protocols and any one time. Also, some assume that start of each flow can be extracted and some assume that payload and header information can be still used for training phase of the algorithms. However, none of the studies deal with network

level VPNs which may contain several protocol types in one tunnel. In the most general VPN case no header or payload information is available for the classifier so even in the training phase might constitute an issue since the training is the core part of the machine learning algorithms.

Another interesting paper by Jeffrey et al [15] addresses the issue of mapping from cluster to application problem. The problem is to map the existing cluster on to the application type to be classified. If the number of clusters created by the algorithm is greater than the number of intended application types, classification becomes quite complicated. To solve this problem the authors fed the clustering algorithm with pre-labelled flows, which increased the overall performance. However, again they assume that pre-labelling is available. In VPN tunnelled flows we do not have any chance to have priori knowledge about the content.

All machine learning approaches fully depend on the training data which may not always represent the general behaviour of network traffic. The mentioned ML methods only generates a probabilistic measure that a packet serves a given protocol. Again such a decision is of little use to real network devices.

III. ENCRYPTED TRAFFIC CLASSIFICATION

Regarding encrypted traffic classification, our analysis suggests that most of the work focuses on either SSH or SSL tunnels. There appear to many challenges and the accuracy of identification is very low, also while application layer tunnels can hold only one application per session, network layer tunnels can hold more than one application per session. For example, Charles et al [16] propose an empirical method based on hidden Markov models which was applied to simulated HTTPS, SSH and AIM. tunnels. Application types were identified with only 20% accuracy.

A paper by Bernaille and Teixeira [17] created a model to detect SSL traffic. They did not include network layer tunneling and assume the ability to analyse the first few bytes of SSL traffic before encryption which is not possible for traffic put through a network layer tunnel their proposed accuracy is about 85%.

Alshammari and Zincir-Heywood [18] used two supervised machine learning algorithms AdaBoost [19] and RIPPER [20] to detect SSH traffic and classify applications inside SSH. They showed that it is feasible that SSH type traffic can be detected. However they only studied a small number of applications and dealt with only application layer encryption. Application types such as telnet, DNS and FTP are classified with 99% accuracy. However, we could not find classification proposals for delay sensitive applications such VoIP.

Marc and Neil [21], stated that application types of an encrypted connection using HTTP can be inferred. In short the existing work shows that although quite a few researchers are studying traffic types and encryption we could see little work related to classification of network layer VPN tunnelled protocols and their application types. Again the identification process results in a probabilistic measure that a packets serves a particular protocol. Again this kind of statistical information

is not usable by commercial devices such as firewalls or network optimizers.

IV. STATEMENT OF PROBLEM

We aim to identify encrypted, tunnelled traffic in such a way that network devices such as firewalls and network optimizers can make use of that decision. First we must identify where such a decision is important. There are two main types of tunnels to consider.

The first type is a peer to peer VPN that connects an individual computer to a remote service or other computer. This type of implementation can be quite problematic in a university or corporate network as protocols which are banned or band-limited can be hidden from firewalls, network optimizers, and proxy servers. For example real time video can be hidden in a tunnel and the network devices just mentioned cannot identify the traffic type.

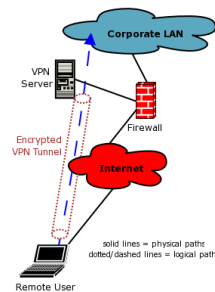


Fig. 1. Corporate VPN

The next type of VPN is the corporate VPN. In this case the nature of the traffic can be identified after the VPN server has decapsulated and decrypted the traffic stream. In this case traditional packet inspection techniques can be used.

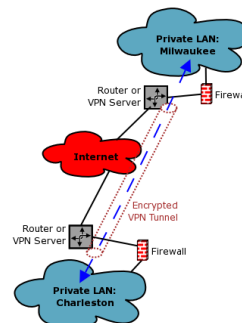


Fig. 2. Client Based VPN

The main application of traffic identification within encrypted tunnels is thus peer to peer tunnels which represent a real concern in some networks.

A. A New Classification Decision Process

In our previous discussion it was seen that current research uses packet statistics to give a probability that an individual packet serves a given protocol. The accuracies reported even

for non encrypted traffic are poor (around 90%), and the results for tunnelled and encrypted traffic are totally unusable. A network device cannot make a sensible decision based on such information.

We propose a new and novel approach that makes probabilistic information usable. First the identification process aims to determine what protocols are present in a peer to peer tunnel. Next advice is given to the network device to either allow, block, or band-limit the entire tunnel, or packets within a certain size range.

To see how this concept works, consider VoIP traffic as shown in Figure 3. VoIP packets fall between certain size limits and packets from this range are taken to be analysed for size and inter-arrival time. After a period of time a decision algorithm will decide that there is, or is not, VoIP in the tunnel. Similar decisions can be made about the presence of streaming video, web browsing, FTP, and bit torrent downloads. Allowing the decision making process to occur over a period of time allows the identification process to have much more information to work with.

The next step is to give advice to the network device. This advice is based on the mix of protocols discovered and enterprise policies. The advice options are to block a tunnel, band limit a tunnel, or allow a tunnel through. In the case of VoIP it is possible to apply these decisions to a range of packet sizes rather than the entire tunnel.

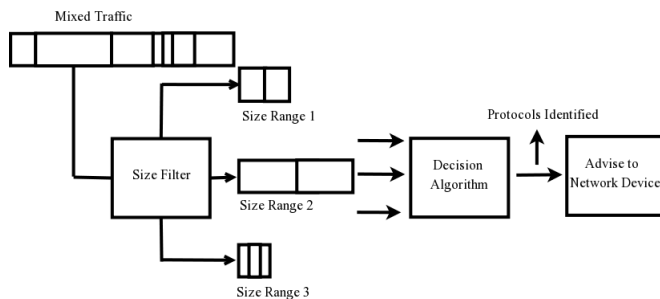


Fig. 3. Decision Process

V. PRELIMINARY ANALYSIS

As a preliminary analysis we studied VoIP traffic from Skype because, like all VoIP systems, it has some significant statistical characteristics which can be identified within several seconds. Available traces of Skype traffic were obtained from [22]. The new scheme depends only to packet size however using inter-arrival time may improve the system as well. We analyzed different voice codecs since codec usage changes the basic statistical structure of the traffic. The following table shows the mean and standard deviation of the packet sizes from each available codec as well as the theoretical sampling frequency.

In figures 4, 5 and 6 we show graphically the probability distribution and cumulative probability distribution of SVOP, iLBC and G729 codecs in terms of individual packet sizes. Although different codec usage changes the behaviour of

TABLE I
SKYPE CODECS

Codec	Mean	Standard Deviation	Sampling Frequency
SVOP	156.32	12.47	16 kHz
iLBC	89.49	8.18	8 kHz
G729	70.96	5.13	8 kHz

packet size distribution, most packets lie between 50-200 byte length. One other interesting result is that Skype traffic packet size is normally distributed. Looking at the standard deviations it is clear that most of the Skype traffic can be gathered if a suitable size window is applied. To find out the exact percentage of traffic which can be classified we can employ basic properties of normally distributed samples. Assume that μ is the mean and σ is the standard deviation of a normal distribution. For any μ and σ , 99.6% of observations fall within 3σ standard deviation of the mean that is $\mu + 3\sigma$ and $\mu - 3\sigma$. Following these fundamental properties we can conclude that for all different codecs in use it is possible to identify 99.6% of VoIP traffic in the range of 56-194 bytes. In figure 7 it is possible to see the overall picture of probability densities of the codecs we tested. We have also tested the available Skype traces against a FTP download trace retrieved from our local labs at RMIT university.

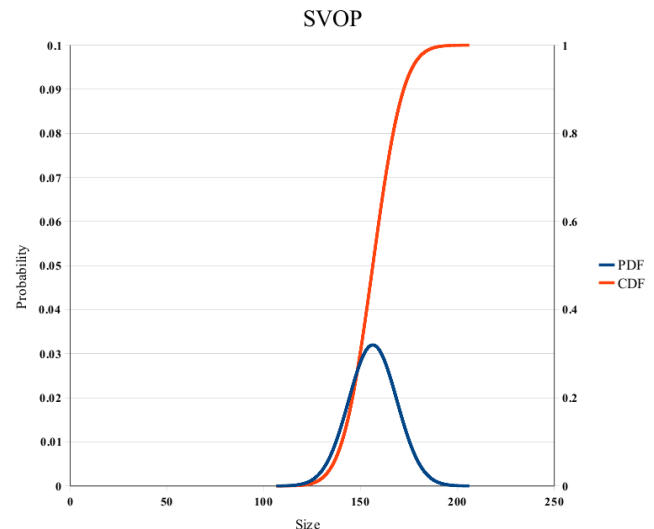


Fig. 4. Probabilistic Size Distribution of SVOP Codec

In figure 8 it possible to see that the density of FTP packets are lying in the MTU range. This also proves our point that VoIP sessions can be detected when FTP and HTTP sessions share a network layer tunnel. It is also important to clarify that VoIP traffic we tested was using UDP as a transport layer protocol. However, we acknowledge the fact that Skype traffic can also be transmitted with TCP. Testing TCP based Skype traffic against other TCP based applications is left as a future implementation. However, our initial trials showed that although Skype traffic can be encapsulated in a TCP datagram,

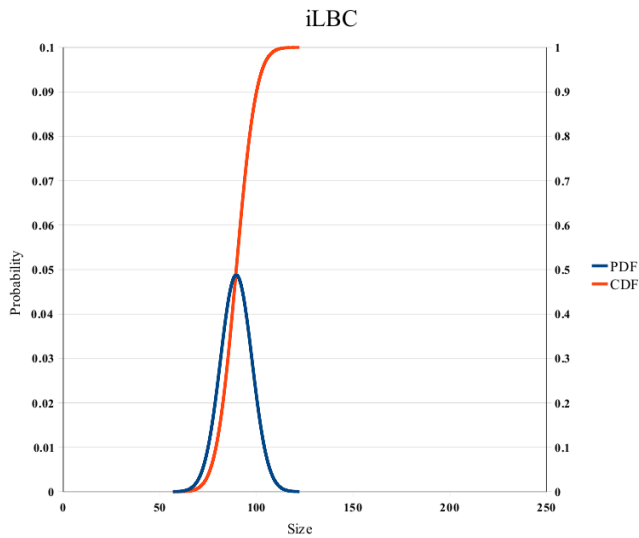


Fig. 5. Probabilistic Size Distribution of iLBC Codec

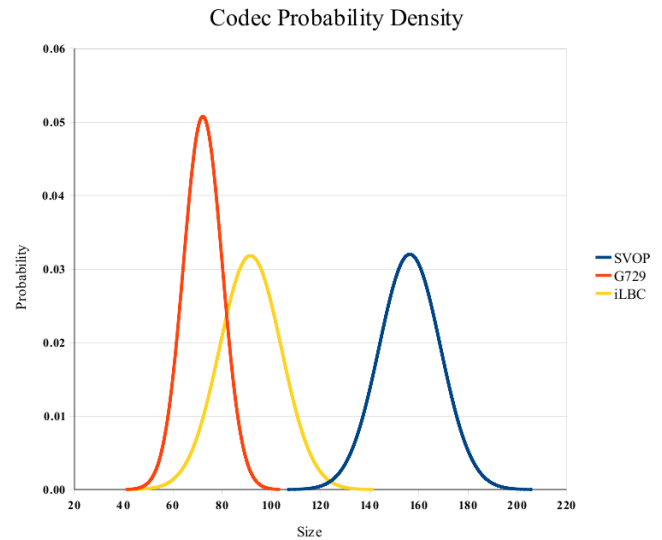


Fig. 7. Probability Density of All Codecs

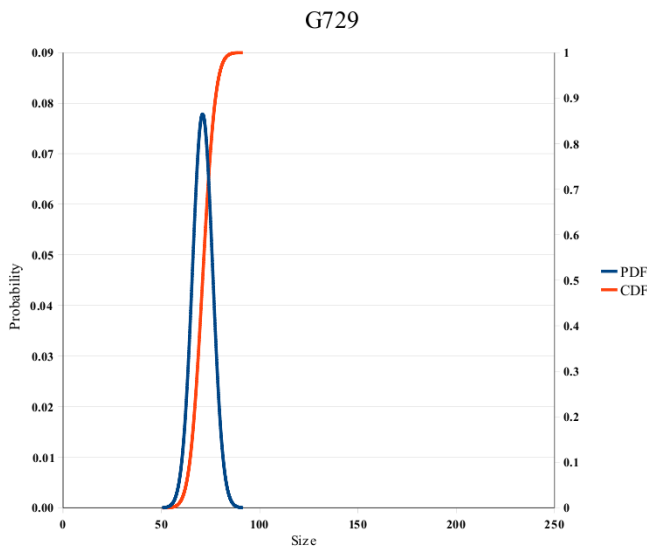


Fig. 6. Probabilistic Size Distribution of G729 Codec

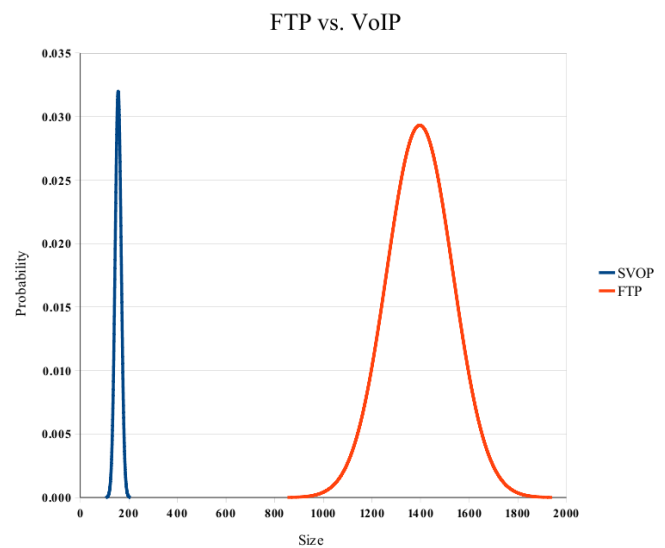


Fig. 8. Probability Density Comparison of FTP vs. VoIP

due to its signalling properties the size of the packets would still have distinct statistical properties.

The simulations done to date are very interesting and provide a solid basis for practical implementation in real network devices. We are currently implementing a system with peer to peer VPNs and a Network Optimizer between them. The aim is to allow VoIP packets to pass unhindered but band limit or perhaps even block download traffic such as bit torrent and FTP. It would appear that VoIP is easy to identify even when mixed with FTP but we intend to investigate identifying VoIP in the presence of many other protocols. We also aim to investigate learning methods for other protocols of practical interest to network management, in particular streamed video and download traffic.

VI. CONCLUSION

In this paper we have reviewed recent approaches to IP traffic classification with encrypted data flows, including application layer encryption such as SSL and network level encryption such as IPSec. The most promising of these approaches use some form of Machine Learning but current research has only been applied to application layer tunnels. There is still much work to be done before packets can be identified within network layer encrypted tunnels. In other words the identification decisions made by these systems cannot direct any useful activity by network devices such as network optimizers, firewalls and proxy servers. We have proposed a new two step approach suitable for peer to peer network layer tunnels that can generate advice that is useful

to network devices-

- Identify the protocols present over a period of time based on statistical properties. In our preliminary studies we used just the PDF of the packet sizes over several seconds.
- Given the protocol identification, and enterprise policies, advise the network device to block the tunnel, band limit the tunnel, or allow the tunnel through at full speed. In the case of streamed traffic such as VoIP this may be applied to a sub-range of packet lengths.

The identification algorithm used in this paper was very simple, just build a PDF of the packet sizes and identify protocol classes based on the observed peaks. Our preliminary results indicate it is possible to tell if a peer to peer tunnel contains VoIP, web browsing, download traffic, or some combination thereof. These results can help a network optimizer at a school or university tell if someone is hiding a download within a peer to peer network layer VPN. We expect it will be possible to identify streaming video of various types though it will probably not be possible to differentiate between different download protocols. This paper used one of the simplest identification algorithms and there is some interesting research to be done developing more sophisticated identification algorithms which have more accurate and faster protocol identification.

Traffic identification within a network layer encrypted tunnel raises a new set of security issues. Does the identification of traffic types pose a security risk? What methods can be used to confuse the identification and when is it appropriate to use such techniques? These very important topics are worthy of deeper reflection and consideration.

REFERENCES

- [1] T. T. T. Nguyen and G. Armitage, "A survey of techniques for internet traffic classification using machine learning," *Communications Surveys & Tutorials, IEEE*, vol. 10, no. 4, pp. 56–76, 2008. [Online]. Available: <http://dx.doi.org/10.1109/SURV.2008.080406>
- [2] T. Ylonen and C. Lonvick, "The secure shell (ssh) protocol architecture," RFC 4251, 2006. [Online]. Available: <http://www.rfc-editor.org/rfc/rfc4251.txt>
- [3] J. Frank and N. U. Mda-c, "Artificial intelligence and intrusion detection: Current and future directions," in *In Proceedings of the 17th National Computer Security Conference*, 1994.
- [4] M. Crotti, M. Dusi, F. Gringoli, and L. Salgarelli, "Traffic classification through simple statistical fingerprinting," *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 1, pp. 5–16, 2007.
- [5] W. Li and A. W. Moore, "A machine learning approach for efficient traffic classification," in *MASCOTS '07: Proceedings of the 2007 15th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 310–317.
- [6] J. R. Quinlan, *C4.5: programs for machine learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.
- [7] A. W. Moore and D. Zuev, "Internet traffic classification using bayesian analysis techniques," in *SIGMETRICS '05: Proceedings of the 2005 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*. New York, NY, USA: ACM, 2005, pp. 50–60.
- [8] A. McGregor, M. Hall, P. Lorier, and J. Brunskill, "Flow clustering using machine learning techniques," in *Lecture Notes in Computer Science*. Springer Berlin/Heidelberg, 2004, pp. 205–214.
- [9] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977. [Online]. Available: <http://dx.doi.org/10.2307/2984875>
- [10] S. Zander, T. Nguyen, and G. Armitage, "Automated traffic classification and application identification using machine learning," in *LCN '05: Proceedings of the The IEEE Conference on Local Computer Networks 30th Anniversary*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 250–257.
- [11] P. Cheeseman and J. Stutz, "Bayesian classification (autoclass): theory and results," pp. 153–180, 1996.
- [12] L. Bernaille, R. Teixeira, I. Akodkenou, A. Soule, and K. Salamatian, "Traffic classification on the fly," *SIGCOMM Comput. Commun. Rev.*, vol. 36, no. 2, pp. 23–26, 2006.
- [13] J. Erman, A. Mahanti, M. Arlitt, and C. Williamson, "Identifying and discriminating between web and peer-to-peer traffic in the network core," in *WWW '07: Proceedings of the 16th international conference on World Wide Web*. New York, NY, USA: ACM, 2007, pp. 883–892.
- [14] G. Junior, J. Maia, R. Holanda, and J. de Sousa, "P2p traffic identification using cluster analysis," in *Global Information Infrastructure Symposium, 2007. GIIS 2007. First International*, July 2007, pp. 128–133.
- [15] J. Erman, A. Mahanti, M. Arlitt, I. Cohen, and C. Williamson, "Semi-supervised network traffic classification," *SIGMETRICS Perform. Eval. Rev.*, vol. 35, no. 1, pp. 369–370, 2007.
- [16] C. V. Wright, F. Monrose, and G. M. Masson, "On inferring application protocol behaviors in encrypted network traffic," *J. Mach. Learn. Res.*, vol. 7, pp. 2745–2769, 2006.
- [17] L. Bernaille and R. Teixeira, "Early recognition of encrypted applications," *Passive and Active Network Measurement*, pp. 165–175, 2007.
- [18] R. Alshammari and A. Zincir-Heywood, "A flow based approach for ssh traffic detection," in *Systems, Man and Cybernetics, 2007. ISIC. IEEE International Conference on*, Oct. 2007, pp. 296–301.
- [19] Y. Freund and R. Schapire, "A short introduction to boosting," *Japanese Society for Artificial Intelligence*, vol. 14, no. 5, pp. 771–780, 1999.
- [20] W. W. Cohen, "Fast effective rule induction," in *In Proceedings of the Twelfth International Conference on Machine Learning*. Morgan Kaufmann, 1995, pp. 115–123.
- [21] M. Liberatore and B. N. Levine, "Inferring the source of encrypted http connections," in *CCS '06: Proceedings of the 13th ACM conference on Computer and communications security*. New York, NY, USA: ACM, 2006, pp. 255–263.
- [22] "Skype traces," <http://tstat.tlc.polito.it/traces-skype.shtml>, retrieved at 27 September 2009.