Survey on Identification of Network Protocol using Collapsed Variational Bayesian Inference Algorithm

Sadiya S. Sheikh, Lalit Dole

Abstract— **The network consists of lots of heterogeneous data flowing through it. The network traffic should be classified for various networking as well as security issues, for instance network measuring, network monitoring or detection of some inobvious activity. For the identification of application protocols, the network traces are worked on to get the protocol message format specifications. This is important because currently most application protocols are proprietary on the internet that means they have no well documented protocol specifications. The proposed system implements a network trace based application protocol identifying system which gives the semantic information as result from protocol message formats with no previous knowledge of protocol specifications. In the proposed system, we find the statistical protocol message formats first by clustering n-grams with the relative semantics, and then classify the raw traces using the statistical formats. The system hopefully is capable in accurately and efficiently identifying the network traces of the target protocol. The system would hopefully work efficiently by the application of a Collapsed Variational Bayesian Inference algorithm for Latent Dirichlet Allocation as the application of Gibbs sampling was too time consuming due to its iterative nature in prior related work.**

Index Terms— **Collapsed Variational Bayesian Inference algorithm, Latent Dirichlet Allocation, network security, protocol identification, traffic classification.**

I. Introduction

The approach mainly concentrates on the extraction process of application-level specifications for network application protocols. Such a task works by analyzing static network traffic traces. As there is a little information available at the network level. Such information about application protocol specification is very helpful in various security-related things. For instance, they are helpful in intrusion detection systems for performing deep packet inspection, and these are used in black-box fuzzing tools easy implementation. It is also used in the automated generation of protocol fuzzers [23] to execute black-box testing of server programs which accept input from network.

As Internet is famous as a business infrastructure, many attacks on it, especially denial-of-service attacks such as TCP SYN flooding [2], Teardrop and Land [3] grows. Because of less security in TCP/IP, responsibility is a must for protecting the sites against network attacks. Although firewalls are used to prevent network attacks, they cannot prevent some specific attacks such as TCP SYN flooding. The deep knowledge of such protocol specifications for finding a number of security problems is invaluable. Consequently, intrusion detection systems (IDS) are increasingly deployed . In IDS, there is a deep packet inspection process done that parse the stream of network into segments or parts with semantics of application-level, and detection rules are applied to only some particular parts of the traffic where application protocol detection plays a major role. Various present methods for protocol detection used for deep packet inspection are time consuming. So the proposed approach can be used to lessen the burden of execution.

The protocol message format specifications are used at it’s maximum from network traces for the exact identification of application protocols. On Internet nowadays, many application protocols are proprietary, that means they have no well-documented protocol specifications in public. According to the expert observation nearly 50% of Internet traffic comes under unknown application protocols. Just for example, the control protocols do not have publicly proper available protocol specifications. The approach devised in this paper thus is different than other present approaches as it does not require protocol specifications for the corresponding protocol identification which automates the analysis with minimal manual efforts requirement.

A semantics-aware classification system, Securitas [1] that takes network traces as input and effectively identifies it’s target application protocol from mixed Internet traffic. There exist some similarity between application protocols and natural languages, as a protocol can be regarded as a language for two processes to communicate. Securitas works on a new idea that the n-grams of protocol traces similar to natural languages, exhibits frequency-rank distribution which is highly skewed and that can be leveraged for accurate protocol identification. We initially cluster n-grams with the same semantics to grasp the statistical protocol message formats, and then use them to classify raw network traces.

The proposed approach performs statistical inference methods and machine learning techniques on the syntax and

Sadiya S. Sheikh, *Department of Computer Science and Engineering, RTMNU/ G.H. Raisoni College of Engineering, Nagpur, India, 7620888488*
Lalit Dole, *Department of Computer Science and Engineering, RTMNU/ G.H. Raisoni College of Engineering, Nagpur, India.*

semantics of observed network traces efficiently as in Securitas with the application of a Collapsed Variational Bayesian Inference algorithm. Our proposed approach is an experiment of finding whether or not it is possible to apply the above mentioned algorithm for the betterment of the Securitas working or not. System generally involves three major phases

1. Modeling phase
   →Data collector
   →N-gram generation
   →Keyword identification
2. Training phase
   →Data collector
   →N-gram generation
   →Feature Extractor
   →Learning Module
3. Classification phase
   →N-gram generation
   →Feature Extractor
   →Classifier

A Collapsed Variational Bayesian (CVB) Inference algorithm for LDA is presumed to be easily implementable and more accurate than standard Variational Bayesian or Gibbs Sampling algorithm. The nucleus idea behind CVB is that topic variable dependence is considered. Computational efficiency is achieved in CVB with a Gaussian approximation, which was found to be so accurate that there is never a need for exact summation [11].

II. Literature Survey

Binpac [2] and GAPA [3] are generic protocol analyzers which require protocol grammars as input. Because of having protocol information helps in identifying and understanding applications which can communicate on ports which are non-standard. A number of systems [10, 11, 17, 18] have been proposed that study the network traces which generates by recording the client-server communication. Special meaning of the protocol can be indicated when the network traces are examined for the occurrence of common structures or bytes.

In [1], protocol identification was conducted based on the packet payloads of raw network traces. The system belongs to protocol signature-based (i.e., application fingerprint) category. Protocol signature-based methods conduct the analysis relying only on the study of the payloads of network traces. Such methods typically involve two ways to implement their functionalities, which includes manual analysis and automatic analysis. In [1], a semantics-aware classification system named Securitas was proposed, which takes network traces as input and effectively identifies the network traces of the target application protocol from mixed Internet traffic.

In [2], a system named FlowSifter was proposed which extracts application protocol field using a systematic framework. It invents a new model for grammar which was named as Counting Regular Grammars (CRG) and also invents for it, it's corresponding model of automata which was named as Counting Automata (CA). Both the models namely CRG and CA add counters which has for the regular grammars and finite state automata, update functions and transition guards. Due to these add on's, the ability to parse and extract fields from context sensitive application protocols is being provided.

In [3], the two very important factors for Network Intrusion Detection/Prevention Systems (NIDS/NIPS) are accuracy and speed. Due to failure of regular expressions to identify the vulnerability conditions, thus the accuracy of such systems have become a major problem. In contradiction to that the vulnerability signatures [10, 29] can exactly extract the vulnerability conditions with better accuracy. But the obvious challenging issue remains that with such a large rule set, finding a way to efficiently apply vulnerability signatures to high speed NIDS/NIPS. The paper [3] presents the first design named Netsheild that is a vulnerability signature based parsing as well as matching engine, by which better accuracy is achieved and that achievement also involves multi-gigabit. The following contributions were made: (i) introduced a candidate selection algorithm that matches thousands of vulnerability signatures at the same time efficiently with the need of only a little memory; (ii) introduced a parsing state machine that achieves fast protocol parsing which is automatically lightweight.

In [4], the second generation design is presented of a generic application-level protocol analyzer (GAPA) which together consists of both domain-specific language and the associated run-time. GAPA is designed to satisfy three nucleus goals: safety, response and analysis at real-time, and analyzer's rapid development. These goals are effective for those who monitor the network to implement protocol analysis. Therefore, GAPA was built to be emerged within the tools like Ethereal, Shield, etc. GAPA is good at preserving safety by the implementation of a safe memory for message parsing as well as analysis. For supporting online analysis, a serial processing model is used by GAPA where parsing is incremental. A similar syntax is used by GAPA same as of many RFCs protocol to speed up protocol development and to incorporate analysis of various common protocol as built-in abstractions.

In [5], there is a crucial step in the network traffic's semantic analysis which according to the high-level protocols is to parse the traffic stream. This converts raw bytes into structured, typed, as well as semantically meaningful data fields that provides a representation at high level of the network traffic.

The above discussed paper presents a system named binpac in which a language which is declarative and compiler are designed so that easily efficient semantic analyzers are made for complex network protocols that are robust too.

In [6], mapping of traffic to applications accurately is crucial for a large range of network measurement as well as management tasks. With the use of TCP or UDP header's well known default server network-port numbers, Internet applications have been identified. But this method is inaccurate.

In this paper, the method is adopted which extracts signatures of application from IP traffic payload content automatically. Mainly three statistical machine learning algorithms are applied to identify signatures for a variety of applications automatically. The results signify that the system is accurate and scales to allow on high speed links of the application identification. It was also encountered that content signatures work in the presence of encryption.

III. Conclusion

In this paper, a system is introduced which takes statically stored network packet traces as input and automatically infers the application of the network traces. The approach is completely based on network packet traces, and it neither requires protocol executable code nor requires any prior knowledge on protocol message format. It statically takes conversation, which is in a CSV format, as input which may be inaccurate as compared to the systems that performs dynamic analysis.

REFERENCES

[1] Xiaochun Yun, Yipeng Wang, Yongzheng Zhang, and Yu Zhou, "A Semantics-Aware Approach to the Automated Network Protocol Identification," in *IEEE/ACM* Transactions on Networking 2015.

[2] C. Meiners, E. Norige, A. X. Liu, and E. Torng, "FlowSifter: A counting automata approach to layer 7 field extraction for deep flow inspection," in *Proc. IEEE INFOCOM*, 2012, pp. 1746–1754.

[3] Z. Li *et al.*, "NetShield: Massive semantics-based vulnerability signature ma0tching for high-speed networks," in *Proc. ACM SIGCOMM*, 2010, pp. 279–290.

[4] N. Borisov, D. J. Brumley, and H. J. Wang, "A generic application-level protocol analyzer and its language," in *Proc. NDSS*, 2007.

[5] R. Pang, V. Paxson, R. Sommer, and L. Peterson, "Binpac: A yacc for writing application protocol parsers," in *Proc. 6th ACM SIGCOMM Conf. Internet Meas.*, 2006, pp. 289–300.

[6] J. St Sauver, "A look at the unidentified half of Netflow (with an additional tutorial on how to use the Internet2 Netflow data archives)," 2008 [Online]. Available: http://www.internet2.edu/presentations/jt2008jan/20080122-stsauver.pdf

[7] P. Haffner, S. Sen, O. Spatscheck, and D. Wang, "ACAS: Automated construction of application signatures," in *Proc. ACM SIGCOMM MineNet*, 2005, pp. 197–202.

[8] J. Kannan, J. Jung, V. Paxson, and C. E. Koksal, "Semi-automated discovery of application session structure," in *Proc. ACM SIGCOMM IMC*, 2006, pp. 119–132.

[9] J. Ma, K. Levchenko, C. Kreibich, S. Savage, and G. M. Voelker, "Unexpected means of protocol inference," in *Proc. ACM SIGCOMM IMC*, 2006, pp. 313–326.

[10] Y. W. Teh, D. Newman, and M. Welling, "A Collapsed Variational Bayesian inference algorithm for Latent Dirichlet allocation," in *Advances in Neural Information Processing Systems*. Red Hook, NY, USA: Curran & Associates, 2007.

[11] [11] Takuya Konishi, Takatomi Kubo, Kazuho Watanabe, and Kazushi Ikeda, "Variational Bayesian Inference Algorithms for Infinite Relational Model of Network Data," in *IEEE* Transactions on *Neural Networks And Learning Systems.*

[12] A. Este, F. Gringoli, and L. Salgarelli, "Support vector machines for tcp traffic classification," *Comput. Netw.*, vol. 53, no. 14, pp. 2476–2490, 2009.

[13] A. Finamore, M. Mellia, M. Meo, and D. Rossi, "KISS: Stochastic packet inspection classifier for UDP traffic," *IEEE/ACM Trans. Netw.*, vol. 18, no. 5, pp. 1505–1515, Oct. 2010.

[14] G. La Mantia, D. Rossi, A. Finamore, M. Mellia, and M. Meo, "Stochastic packet inspection for TCP traffic," in *Proc. IEEE ICC*, 2010, pp. 1–6.

[15] A. Tongaonkar, R. Keralapura, and A. Nucci, "Santaclass: A self adaptive network traffic classification system," in *Proc. IFIP Netw. Conf.*, 2013, pp. 1–9.

[16] Y. Wang *et al.*, "A semantics aware approach to automated reverse engineering unknown protocols," in *Proc. 20th IEEE ICNP*, 2012, pg. 1–10.

[17] T. Karagiannis, K. Papagiannaki, and M. Faloutsos, "Blinc: Multilevel traffic classification in the dark," *Comput. Commun. Rev.*, vol. 35, no. 4, pp. 229–240, 2005.

[18] L. Bernaille, R. Teixeira, and K. Salamatian, "Early application identification," in *Proc. ACM CoNEXT*, 2006, Art. no. 6.

[19] M. Iliofotou, M. Faloutsos, and M. Mitzenmacher, "Exploiting dynamicity in graph-based traffic analysis: Techniques and applications," in *Proc. 5th CoNEXT*, 2009, pp. 241–252.

[20] J. Zhang, C. Chen, Y. Xiang, W. Zhou, and A. Vasilakos, "An effective network traffic classification method with unknown flow detection," *IEEE Trans. Netw. Service Manage.*, vol. 10, no. 2, pp. 133–147, Jun.2013.

[21] F. Gringoli *et al.*, "GT: Picking up the truth from the ground for internet traffic," *Comput. Commun. Rev.*, vol. 39, no. 5, pp. 12–18, 2009.

[22] D. Oktavianto and I. Muhardianto, "Cuckoo Malware Analysis," Birmingham, U. K. :Packet, 2013.

[23] W. Cui, J. Kannan, and H. J. Wang, "Discoverer: Automatic protocol reverse engineering from network traces," in *Proc. 16th USENIX SS*, 2007, Art. Number 14.

Sadiya S. Sheikh (M.Tech 4th sem.) received the B.E. degree in computer science from Rajiv Gandhi College of Engineering, Research and Technology, Chandrapur, Maharashtra, India in 2014.

She is a student of M.Tech. 4th sem. and will hopefully complete the PG degree in 2016.

Lalit Dole(M.S.)
He is an Assistant Professor in Department of Computer Science with the Institute of G.H. Raisoni College of Engineering. He has published more than 15 research papers in refereed international journals and conferences like IJSR, IJETT, ISCO etc.

All Rights Reserved © 2016 IJARCET