NETWORK ANOMALY DETECTION

FYP16021 Project Plan

by Tien Hsuan Wu
Supervised by Dr. S. M. Yiu

September 2016

TABLE OF CONTENTS

Abstract 2

I. Introduction 2

II. Related Studies 3

III. Objectives 4

IV. Methodology 5

A. Application Protocol Identification 5

B. Anomaly Detection 5

V. Project Schedule and Deliverables 6

References 7

ABSTRACT

Network has brought convenience to the world by allowing fast transformation of data, but it also exposes a number of vulnerabilities. With anomaly detection systems, the outliers of packets can be detected and computers are prevented from attacks. Some anomaly detection systems found in literature are based on data mining methods. Recently, as deep learning becomes a popular area of research, we propose using deep learning as the model for anomaly detection in this project.

I. INTRODUCTION

The Internet is evolving and it has revolutionized the world since the World Wide Web was invented. The usage of the Internet has become necessary in various areas. Through the Internet, we are able to gain access to remote hosts, retrieve data and operate on the hosts. This simplifies our day-to-day life, but without appropriate security measures, it is likely that the systems would be compromised, causing individuals and companies suffering from great loss. Intruders may gain unauthorized privileges, or simply overload the server to make it unavailable. Both of these may incur great loss for the system owners.

In order to protect the computers from being hacked, intrusion detection systems (IDS) can be installed. Some common open source IDS are Snort [1] and Suricata [2]. With IDS installed, whenever a system encounters unauthorized access, it can respond by refusing

such access request. Moreover, it can generate alerts for human to inspect if there is any system defect.

Intrusion detection systems can be classified into three categories: signature detection systems, anomaly detection systems, and hybrid systems [3, 4]. A signature detection system maintains a misuse database which contains the patterns of abnormal traffic. When a packet arrives, the system will compare it with the misuse database to determine whether such packet is normal. The advantage is that signature detection systems generate a low false positive rate when the misuse database is reliable. This is due to the fact that intrusions detected are supposed to have a high similarity with the abnormal packets. For an anomaly detection system, it uses the pattern generated from normal traffic as the baseline. Any pattern that deviates from the normal traffic is considered anomalous. The advantage is that it can detect unseen (zero-day) attacks. Hybrid systems combine both techniques used in signature detection systems and anomaly detection systems.

II. Related Studies

Data mining techniques and machine learning algorithms can be applied to intrusion detection systems, and these techniques have been extensively studied in the past decade. Clustering and classification are some techniques used in IDS. Münz, Li & Carle proposed an anomaly detection system using k-means algorithm which combines both classification and outlier detection [5]. In [6, 7], the authors combined the k-means clustering with naïve Bayes classification. In [8], the authors further utilized the result

from k-means clustering as new features for naïve Bayes classifier. In [9], naïve Bayes classifier is combined with decision tree algorithms.

The abovementioned methods operate on network features only, namely, the connection records. To take payload data from packets into consideration, we need some other techniques. PAYL is a histogram-based classification method that takes payload data as input. It builds a histogram from the input, with frequency of each byte pattern being a bin (see Fig. 1Fig.), and compares the histogram built from the data with baseline [10]. Deep learning techniques can also be implemented to detect anomalies. Wang [11] built a system that can classify TCP packets according to their application layer protocols, and suggested that misclassified packets may be anomalous.

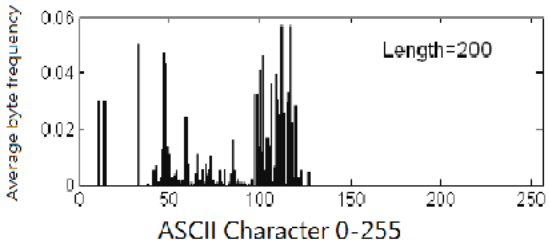

FIG. 1. Example of byte distribution for a 200-byte packet.

III. OBJECTIVES

The ultimate goal for this project is to build an anomaly detection system using neural networks and deep learning, and then study the effectiveness of different models and learning techniques. In order to achieve the goal, we divide the project into two phases: i)

application protocol identification and ii) anomaly detection. The next section discusses how to construct the network to achieve the goals in each phase.

IV. METHODOLOGY

A. APPLICATION PROTOCOL IDENTIFICATION

Given a network packet, we aim to determine what protocol is used in the application layer. We will use a simple feedforward network as an initial implementation, and modify the network to achieve better results. After the neural network is built, we can study the relationship between the byte features and their importance in protocol identification. Fig. 2 shows the result of feature learning from [11]. As the baseline (normal behavior) for different applications may vary, the result derived can help in the second phase.

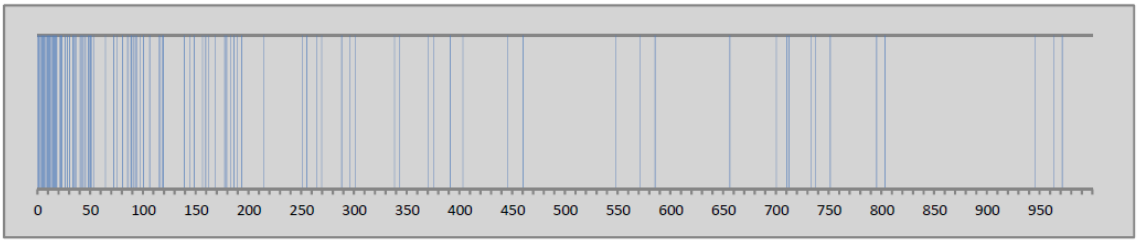

FIG. 2. The most important 100 locations of the data.

B. ANOMALY DETECTION

From the system we will have built in the earlier phase, we can modify the objective in order to detect anomaly. In addition to the payload data, we can also include the statistical features of the packets in the dataset. Furthermore, the structure of the neural network need not be the same. For example, we can use recurrent neural network to capture the information of previous packets. The system will be evaluated with real

dataset if it is obtainable. Another alternative is to evaluate the system with DARPA [12] dataset.

At the deadline of this project plan, we use Keras for deep learning development and Theano as the underlying platform. Keras is highly modular and allows fast prototyping [13]. It supports both convolutional layers and recurrent layers, which can be implemented in later phase of this project.

V. PROJECT SCHEDULE AND DELIVERABLES

September	• *Study the theory of deep learning* • *Familiarize myself with deep learning model constructions*
4 October 2016	Deliverables of Phase 1 (Inception) • Detailed project plan • Project web page
October - November	*Development for application protocol identification*
December	*Development for anomaly detection (I)*
11-15 January 2017	First presentation
24 January 2017	Deliverables of Phase 2 (Elaboration) • Preliminary implementation • Detailed interim report
February - Mid March	*Development for anomaly detection (II)*
Mid March – Mid April	*Study the performance of anomaly detection*
17 April 2017	Deliverables of Phase 3 (Construction) • Finalized tested implementation • Final report
18-22 April 2017	Final presentation
3 May 2017	Project exhibition
6 June 2017	Project competition (for selected projects only)

REFERENCES

[1] Cisco. (2016). *Snort*. Available: https://www.snort.org/

[2] Open Information Security Foundation. (2016). *Suricata*. Available: https://suricata-ids.org/

[3] A. Patcha and J.-M. Park, "An overview of anomaly detection techniques: Existing solutions and latest technological trends," *Computer networks*, vol. 51, pp. 3448-3470, 2007.

[4] S. Agrawal and J. Agrawal, "Survey on Anomaly Detection using Data Mining Techniques," *Procedia Computer Science*, vol. 60, pp. 708-713, 2015.

[5] G. Münz, S. Li, and G. Carle, "Traffic anomaly detection using k-means clustering," in *GI/ITG Workshop MMBnet*, 2007.

[6] R. Chitrakar and C. Huang, "Anomaly based Intrusion Detection using Hybrid Learning Approach of combining k-Medoids Clustering and Naïve Bayes Classification," in *2012 8th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM)*, 2012, pp. 1-5.

[7] Z. Muda, W. Yassin, M. Sulaiman, and N. I. Udzir, "A K-Means and Naive Bayes learning approach for better intrusion detection," *Information Technology Journal*, vol. 10, pp. 648-655, 2011.

[8] S. Varuna and P. Natesan, "An integration of k-means clustering and naïve bayes classifier for Intrusion Detection," in *2015 3rd International Conference on Signal Processing, Communication and Networking (ICSCN)*, 2015, pp. 1-5.

[9] D. M. Farid, N. Harbi, and M. Z. Rahman, "Combining naive bayes and decision tree for adaptive intrusion detection," *arXiv preprint arXiv:1005.4496*, 2010.

[10] K. Wang and S. J. Stolfo, "Anomalous payload-based network intrusion detection," in *International Workshop on Recent Advances in Intrusion Detection*, 2004, pp. 203-222.

[11] Z. Wang, "The Applications of Deep Learning on Traffic Identification," presented at the Black Hat, USA, 2015.

[12] R. Lippmann, J. W. Haines, D. J. Fried, J. Korba, and K. Das, "The 1999 DARPA off-line intrusion detection evaluation," *Computer networks*, vol. 34, pp. 579-595, 2000.

[13] F. Chollet. (2016). *Keras Documentation*. Available: https://keras.io/