Poster Abstract: Encrypted Malware Traffic Detection Using Incremental Learning

Insup Lee, Heejun Roh[†], and Wonjun Lee
Network and Security Research Lab.
School of Cybersecurity, Korea University, Seoul, Republic of Korea
[†] *Division of Applied Mathematical Sciences*, Korea University, Sejong, Republic of Korea
wlee@korea.ac.kr

***Abstract*—Even though the growing adoption of TLS protocol empowers web traffic to secure privacy, attackers also leverage the TLS to evade from detection, and this makes detecting threats from the encrypted traffic a crucial task. In this paper, we propose an effective encrypted malware traffic detection method that maintains sufficient performance level by periodic updates using machine learning. The proposed method employs incremental algorithms trained by 31 flow features from TLS, HTTP, and DNS. Experimental results show that the incremental Support Vector Machine with Stochastic Gradient Descent algorithm is suitable for the detection method amongst three algorithms, by off-line and on-line accuracy at a low false discovery rate.**

***Index Terms*—Encrypted Malware Detection, Transport Layer Security, Incremental Learning, Machine Learning**

I. Introduction

Transport Layer Security (TLS) has encrypted most web traffic recently, and attackers also use the encryption to hide their malware. Unfortunately, traditional detection mechanisms such as port-based approach and deep packet inspection do not effectively identify threats in the encrypted traffic [1]. After success in various domains, machine learning has attracted substantial attention as a countermeasure to the threats in encrypted traffic. Liu et al. [2] suggested a model named FS-Net applying a recurrent neural network, which learns raw packets without a manual feature engineering. On the other hand, Anderson et al. [3], [4] pointed out that the feature engineering is the most decisive part and showed that features from contextual flow data, which means HTTP and DNS features correlated with TLS, can improve the performance of classifying encrypted malicious flow. However, they only analyzed batch learning algorithms, the learning mechanism that trains all possible data at once.

We focus on two issues for effective encrypted malware traffic detection. First, as network traffic arrives continuously over time, the detection model needs to be updated periodically to sustain alarm accuracy. Incremental learning is more suitable for this sequential data processing than batch learning [5]. Second, although an engineered feature set is less efficient to obtain, it typically provides the most drastic increase in performance [4]. The contributions of this paper are as follows.

- We first propose a detection method that uses incremental algorithms trained by contextual flow information, which consists of 31 features related to TLS, HTTP, and DNS.
- We evaluate three incremental algorithms by the accuracy with a false discovery rate of 0.001% and consider intermediate models to provide a deeper insight.

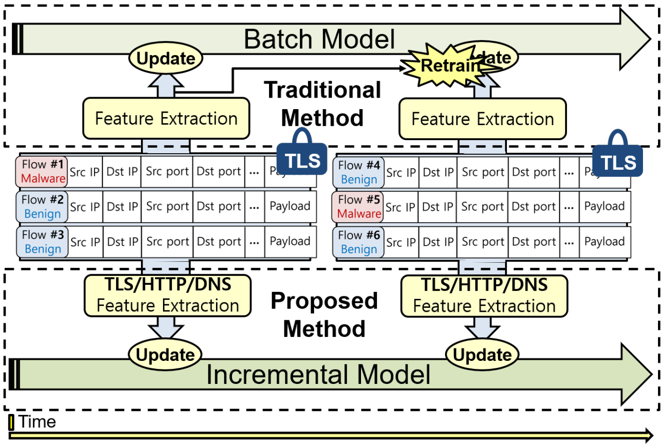
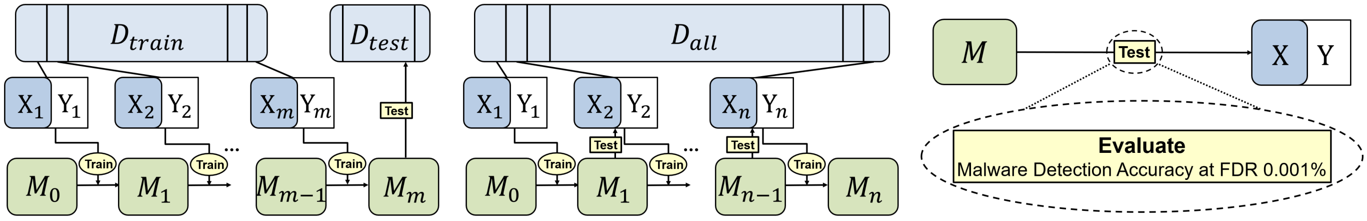


Fig. 1: Detection method using incremental learning.

II. Proposed Method

As shown in Fig. 1, we propose an incremental learning-based detection method that leverages contextual data extracted from TLS encrypted flows and maintains performance through periodic updates. Fig. 1 also shows two learning paradigms, batch and incremental. The incremental method trains with the newly entered data alone, while the batch method requires retraining the previously trained data. This characteristic makes incremental learning is more proper for easy model updates. A feature set from TLS, HTTP, and DNS is determined to improve performance, and the set consists of 31 features such as inter-arrival times, TLS Cipher Suite, HTTP Content-Type, and DNS response name. The feature set highlights more on TLS metadata compared to the previous research [3], whose feature set includes 27 features.

In evaluation, we measure an accuracy under a limited false discovery rate (FDR) and consider both the last updated model and the intermediate models. Since simple FDR can overwhelm due to the massive volume of network traffic [3], we observe the accuracy when FDR is limited to 0.001%. Fig. 2a and Fig. 2b show two evaluation settings called off-

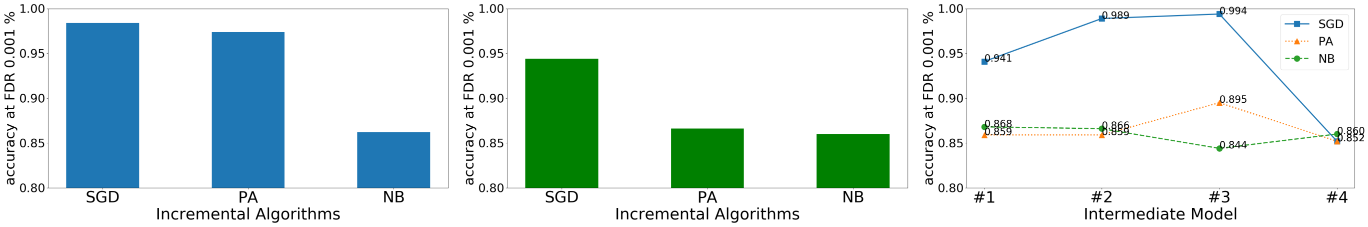


(a) off-line setting evaluation

(b) on-line setting evaluation

(c) evaluation metric

Fig. 2: Evaluation method of incremental learning. (a) measures a score of the last trained model; (b) measures an average score of intermediate models; (c) depicts a metric used in (a) and (b).



(a) off-line accuracy at FDR 0.001%

(b) on-line accuracy at FDR 0.001%

(c) accuracy of intermediate models

Fig. 3: Results from experiments.

line and on-line [5], and Fig. 2c indicates a metric in the both settings. Learning objective in this detection problem is to predict a target variable $y \in \{\text{benign}, \text{malware}\}$ given a set of selected flow features $\mathbf{x} \in \mathbb{R}^{31}$. When a chunk of flows $(\mathbf{X}, \mathbf{Y})$, where $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_{chunk\_size}\}$ and $\mathbf{Y} = \{y_1, y_2, \cdots, y_{chunk\_size}\}$, from data $D = \{(\mathbf{X}_i, \mathbf{Y}_i) | i \in \{1, 2, \cdots\}\}$ arrives, an incremental model updates like $M_i = \text{train}(M_{i-1}, (\mathbf{X}_i, \mathbf{Y}_i))$. The off-line setting is an evaluation of a last updated model, and the setting explicitly splits the data D_{all} into a training set D_{train} and a testing set D_{test}. In contrast, the on-line setting averages prediction performances of intermediate models and does not split the D_{all}. Instead, each chunk of flows is used for intermediate testing first and then for the model update.

III. Evaluation

Benign pcaps are self-collected whereas the malware pcaps are offered by malware-traffic-analysis.net [6]. An in-house analysis tool based on TShark extracts flows from the pcaps, then 31 features of each flow are selected, preprocessed, and used in training. We assume that the traffic arrives five times and the data to be labeled. Off-line setting accuracy is a result of the final model after five training steps, while on-line setting accuracy is measured by averaging the scores of four intermediate models considering trained sample numbers for each model. Scikit-learn library is used to implement three representative incremental algorithms, including Support Vector Machine with Stochastic Gradient Descent (SGD), Passive Aggressive (PA), and Gaussian Naive Bayes (NB).

Experimental results are shown in Fig.3. Although both SGD and PA achieve similar off-line accuracy in Fig.3a, they show a difference in on-line accuracy as the SGD achieves 0.944 while the PA achieves 0.866 in Fig.3b. It implies that the SGD not only shows high detection accuracy at FDR of 0.001%, but also converges on a performance level more rapidly than the other methods. Fig.3c shows the accuracy of intermediate models, a detailed view of on-line accuracy.

IV. Conclusion

In this work, we presented a TLS-encrypted malware traffic detection method, that uses incremental learning and trains 31 contextual flow features. Experimental results in terms of accuracy at 0.001% FDR in two ways, including off-line and on-line settings, showed that the incremental SGD-based method detects encrypted anomalies efficiently.

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2019R1A2C2088812). Wonjun Lee is the corresponding author.

References

[1] P. Wang, X. Chen, F. Ye, and Z. Sun, “A survey of techniques for mobile service encrypted traffic classification using deep learning,” *IEEE Access*, vol. 7, pp. 54 024–54 033, 2019.

[2] C. Liu, L. He, G. Xiong, Z. Cao, and Z. Li, “Fs-net: A flow sequence network for encrypted traffic classification,” in *Proc. of IEEE INFOCOM*, 2019.

[3] B. Anderson and D. McGrew, “Identifying encrypted malware traffic with contextual flow data,” in *Proc. of ACM workshop on artificial intelligence and security*, 2016.

[4] ——, “Machine learning for encrypted malware traffic classification: accounting for noisy labels and non-stationarity,” in *Proc. of ACM SIGKDD*, 2017.

[5] V. Losing, B. Hammer, and H. Wersing, “Incremental on-line learning: A review and comparison of state of the art algorithms,” *Neurocomputing*, vol. 275, pp. 1261–1274, 2018.

[6] B. Duncan, “Malware traffic analysis,” http:/malware-traffic-analysis.net/.