Secrecy and Sensitivity: Privacy-Performance Trade-Offs in Encrypted Traffic Classification

Spencer Giddens* **Raphael Labaca-Castro***
Dan Zhao* **Sandra Guasch*** **Parth Mishra*** **Nicolas Gama***

*SandboxAQ, Palo Alto, USA
{spencer.giddens, raphael.labaca, dan.zhao, sandra.guasch,
parth.mishra, nicolas.gama}@sandboxaq.com

Abstract

As datasets and models grow in size and complexity to increase performance, the risks associated with sensitive data also grow. Differential privacy (DP) offers a framework for designing mechanisms that provide a degree of privacy that can help conceal sensitive features or information. However, different domains and applications can naturally exhibit different rates of trade-offs between privacy and performance depending on their characteristics. In contrast to well-studied areas (e.g., healthcare), one relatively unexplored domain is network traffic analysis where the data contains sensitive information on users' communications. In this paper, we apply DP to various machine learning models trained to classify between encrypted and non-encrypted packets from network traffic; we emphasize that our goal is to examine a relatively unexplored area to analyze the trade-offs between privacy and performance when the data contains both encrypted and un-encrypted observations. We show how varying model architecture and feature sets can be a relatively simple way to achieve more optimal performance-privacy trade-offs; we also compare and contextualize reasonable privacy budgets from our analysis in the network traffic domain against those in other more well-studied domains.

1 Introduction

Network traffic analysis is a key component of infrastructure security—proper identification of network protocols can facilitate network sizing and enable the detection of anomalous connections, revealing ongoing attacks or insecure protocols within the network. A unique challenge for training machine learning (ML) models on network traffic data lies with the data itself, which contains sensitive information such as IPs, ports, protocols, or clear-text payloads. If these models are shared across different parties, it is imperative that no sensitive information on the underlying data is leaked.

In this paper, we explore the trade-offs of applying differential privacy (DP) [9] to protect the privacy of the training data in the context of network traffic classification; by varying the choice of model architecture and features used, we study the privacy-performance trade-offs of training both the DP and non-DP versions of models and show how these changes, along with the underlying domain and data characteristics, can considerably impact the selection of reasonable privacy budgets.

Machine Learning for Systems Workshop at 37th NeurIPS Conference, 2023, New Orleans, LA, USA.

2 Background

2.1 Network Traffic

Data is carried over computer networks in the form of discrete network packets where each packet carries protocol-dependent headers along with information in its payload. These packets then constitute a tuple-like structure consisting of multiple fields of information including source and destination IP addresses, ports, and other data relevant to network protocols. These packets are our fundamental unit of information as we classify between encrypted and un-encrypted/plain traffic. We note that if a compressed file is sent over an un-encrypted protocol we consider it plain.

2.2 Differential Privacy (DP)

Differential privacy, intuitively, ensures that for any given individual in a dataset, the output of a DP-satisfying mechanism will be similar whether the individual's data is included in the mechanism input or not. Those protected by DP guarantees (i.e., the entity whose presence is to be concealed) are known as privacy units and can represent any entity in the data (e.g., a user). In this paper, our privacy unit is a network packet or the information in said packet. We first formalize the notion of DP.

Definition 1 ((ε, δ)-differential privacy)**.** [8] *A randomized mechanism $\mathcal{M}$ satisfies (ε, δ)-*DP *if for all $S \subset \text{Range}(\mathcal{M})$ and all neighboring datasets D and $\tilde{D}$ (datasets differing by a single individual),*

$$P(\mathcal{M}(D) \in S) \leq e^{\varepsilon} P(\mathcal{M}(\tilde{D}) \in S) + \delta, \tag{1}$$

*where $\varepsilon > 0$ and $\delta \in [0, 1)$ are privacy budget parameters. When $\delta = 0$, we denote this as ε-*DP.

The degree of similarity between neighboring datasets for a DP mechanism is governed by ε; smaller values correspond to more privacy and vice versa. The parameter δ is commonly viewed as the probability with which ε-DP fails and is usually set on the order of $o(1/\text{poly}(n))$, where n is the size of the dataset. DP's popularity can be largely attributed to its strong theoretical properties, relative ease of use, and overall flexibility. We refer to [10] for a more comprehensive review.

2.3 Related Work

Ad hoc anonymization methods alone have been insufficient to ensure privacy for sensitive datasets [15, 1]. Sweeney [21] linked public voter records to anonymized health records from Massachusetts state employees to identify then-governor William Weld's health records. A similar attack [15] shut down the Netflix Prize competition after individuals in the anonymized competition dataset were partially de-identified. Even summary statistics of anonymized data have proven insecure as attacks on 2010 US Census statistics were able to reconstruct 46% of the records [6].

Models trained on sensitive data are also susceptible to attacks. [19] showed that, by using a black-box "target model" to synthesize training data and using those to train "shadow models" replicating the target model, an attacker can use the shadow models and synthesized data to infer membership of a given record in the training dataset for the target model. [24] demonstrated (approximate) attribute inference is also possible using membership inference as a subroutine. If attackers have additional information (e.g., model parameters), more attacks are possible [19, 24].DP makes no assumptions on the methods used by attackers to reveal an individual's presence in a sensitive dataset. [22] proved that an attacker's membership inference (MI) advantage Adv_{MI} (i.e., the difference between the attacker's true and false positive rates) is bounded by $\text{Adv}_{\text{MI}} \leq e^{\varepsilon} - 1$. Even for larger ε where theoretical MI advantage bounds no longer hold, [13] demonstrated that DP still limits state-of-the-art MI attack success rates in practice. [2] explored the use of DP to train a deep neural network to classify encrypted network traffic into classes of interest but do not attempt to classify between encrypted and plain data or study performance-privacy trade-offs using reasonable privacy budgets.

2.4 DP in Network Analysis

As described in § 2.2, DP provides mathematically rigorous privacy guarantees to the data which, in this context, we use to try and protect sensitive data that is commonly exchanged throughout our networks. One problem with network data is that sensitive information might be exposed while travelling the network. If attributes can be inferred from the data, user information such as

visited websites, financial transactions, or even passwords can be revealed. Other domain-specific peculiarities include the possible existence of more efficient DP mechanisms when portions of the network data are already encrypted or applying DP to an online stream of network packet time-series data among others—all relatively unexplored areas which we leave as part of our future work.

3 Methodology

The goal is to train a binary classifier using two sets of features, with and without DP, to distinguish between encrypted and un-encrypted network traffic. These two sets of features can be characterized by two approaches towards where the most useful network data lies: namely *header-based*, in which information about the network packet header is extracted, and *payload-based*, that focuses in calculating metrics that characterize the network payload. We train *vanilla* versions of these models (without DP guarantees) as baselines to compare against their differentially private versions.

3.1 Approach

To classify network traffic into *encrypted* and *non-encrypted* (or plain) data, we pursue two strategies, each characterized by a different set of explanatory variables in classifying network data: a *header-based* approach and a *payload-based* approach. In the former, a number of features are extracted from the header of the packet, while, in the latter, information from the payload itself is used to calculate randomness metrics as our features.

Our privacy-preserving versions of both random forest and logistic regression models are implemented in Python via IBM's DP library, *diffprivlib* [12]. These are produced with ε-DP guarantees for various values of ε. As the dataset considered for this paper is already public, our main focus is to explore the privacy-utility trade-off to determine a reasonable domain-specific value for the privacy budget ε that balances DP guarantees and model performance. To have a fair comparison between vanilla and DP models, especially given the additional privacy budget allocation that would be necessary for hyper-parameter tuning, we train each model with its default settings.

Header-based. In this approach, the features are extracted from the header of the network packets in the dataset 3.2 including multiple fields of network protocols found in the network and transport layers. These features are calculated through a custom network dissector tool, which provides a serialized representation before entering into a pre-processing pipeline that processes the data in a way to mitigate inconsistencies produced during the data capturing process from network interfaces such as invalid or incomplete packets.

Payload-based. A payload-based approach is characterized by the hypothesis that the entropy of encrypted data will be higher than that of equal-length plain data. Inspired by [4], we use the statistics from randomness tests conducted on the payload data as features to train our classifiers. The payload is first extracted using a custom extraction algorithm before being passed into a module that conducts randomness tests on the payload including entropy, chi-squared, and arithmetic average, which are used as the key features in our payload-based approach.

Model Choices. We train random forest, decision tree, AdaBoost, and logistic regression models as our base models. Due to our specific domain and dataset, we favored tree-based approaches that tend to be well-suited with both numerical and categorical data. Likewise, we also chose to evaluate an AdaBoost algorithm since weak learners behave similar to decision trees using a single split. Due to the nature of the network data and its heterogeneity across different network environments it might be useful to leverage its iterate methods to improve overall performance. Finally, we explore the logistic regression model as a simple binary algorithm to benchmark against previous models accordingly.

3.2 Dataset

The dataset [18] consists of network data captures in PCAP format collected between July 3rd at 9AM to July 7th at 5PM in 2017 and has been used relatively frequently in the field of network traffic applications [17, 20, 23]. The data is labeled as encrypted or un-encrypted (plain) before being divided into train/test splits. The training set consists of $\sim$1.26 million packets while the test set has $\sim$350,000 with approximately equal representation between the encrypted/un-encrypted classes.

4 Evaluation

We evaluate the performance and trade-offs of our models with and without DP using both the feature and payload-based approaches with a 70%/30% train-test data split. Vanilla (without DP) random forests, decision trees, AdaBoost, and logistic regression models are trained; these classifiers were then evaluated via prediction accuracy and F1-score on the test set. Then, for each $\varepsilon \in \{10^{-7}, 10^{-6.5}, \ldots, 10^{3}\}$, we train 30 random forests and logistic regression models satisfying ε-DP, and calculate their accuracy and F1-scores on the test set for comparison.

Header-based approach. Figure 2 shows the results of our DP simulations. As expected, the average performance of the privacy-preserving models approaches the performance of non-DP models as ε increases. For the DP version of logistic regression, its performance approaches that of the non-DP version starting from $\varepsilon = 10^2$ onward. While theoretical privacy guarantees at these ε values are weak, [13] demonstrated that even at this large of a privacy budget, practical privacy benefits can still be realized. For the header-based approach, the random forest model appears to be more promising with only a slight performance loss. The performance metrics level off beginning around $\varepsilon = 10^{-4}$. Though at this ε we see on average a 10% performance loss, the privacy guarantees at this level are strong. Based on [22], an attacker's membership inference advantage can be at most $e^{0.0001} - 1 \approx 0.0001$, guaranteeing that an attacker's ability to infer whether any given network packet is in the training set is barely better than random guessing. Based on these results, a DP version of the random forest with the header-based approach works best.

Payload-based approach. In Figure 2, for the payload-based approach, in contrast to the header-based comparisons, both the DP logistic regression and DP random forest reach their best performance levels at smaller values of ε. DP logistic regression begins performing similarly to its vanilla counterpart around $\varepsilon = 10^{-3}$, while the performance of DP random forest levels off at around $\varepsilon = 10^{-4.5}$. In fact, for $\varepsilon \geq 10^{-2}$, DP logistic regression even outperforms DP random forest for the payload-based approach. These values of ε all represent strong theoretical privacy guarantees against membership inference attacks.

4.1 Privacy budget ε comparisons with other domains

We compare the minimum ε reasonable privacy budgets from our network traffic domain to DP models in more common, well-studied domains in finance and healthcare (Table 1). For our purposes, we define a "reasonable privacy budget" to be a value ε at which an ε-DP classifier achieves performance that is both better than a baseline fully-random classifier and as close as possible to the performance of its analogous vanilla (non-DP) classifier. For example, a reasonable privacy budget for the DP random forest classifiers in Figure 2 would be $\varepsilon \geq 10^{-4}$. To ensure a fair comparison, we took a random sub-sample of our network traffic data of approximately the same size and class distribution and re-ran our DP model simulations. In this circumstance, the payload-based approach achieves reasonably good utility for ε values comparable to the finance domain, while the header-based approach struggles to obtain better-than-baseline performance for any ε in [5, 11], possibly due to the dimensionality of feature sets used. Though the healthcare domain results are not directly comparable to ours due to differences in the types of differential privacy and ML models used, we believe these comparisons give additional perspective and highlight the importance of better understanding reasonable privacy budgets with respect to the particular data-generating process and characteristics of each domain.

5 Conclusion

In this paper, we explored the performance/privacy trade-offs in the network security domain and assessed how these trade-offs vary with the choice of features and model architecture as well as against other more well-studied domains. A better understanding of these trade-offs between performance and privacy guarantees can derive easy and efficient ways to protect sensitive data and still preserve performance. Our future work hopes to build upon this by developing more efficient privacy-preserving mechanisms such as studying DP guarantees for only protecting/concealing un-encrypted observations in datasets with both encrypted and un-encrypted data.

References

[1] S. Ahn. Whose genome is it anyway?: Re-identification and privacy protection in public and participatory genomics. *The San Diego law review*, 52:751, 2015.

[2] I. Akbari and E. Tahoun. Privpkt: Privacy preserving collaborative encrypted traffic classification. *ResearchGate*, 2020.

[3] B. K. Beaulieu-Jones, W. Yuan, S. G. Finlayson, and Z. S. Wu. Privacy-preserving distributed deep learning for clinical data, 2018.

[4] S. Cha and H. Kim. Detecting encrypted traffic: a machine learning approach. In *Information Security Applications: 17th International Workshop, WISA 2016, Jeju Island, Korea, August 25-27, 2016, Revised Selected Papers 17*, pages 54–65. Springer, 2017.

[5] K. Chaudhuri, C. Monteleoni, and A. D. Sarwate. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12(29):1069–1109, 2011.

[6] D. Desfontaines. Demystifying the us census bureau's reconstruction attack. `https://desfontain.es/privacy/us-census-reconstruction-attack.html`, 2021. Accessed: September 5, 2023.

[7] D. Dua and C. Graff. Uci machine learning repository, 2017.

[8] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor. Our data, ourselves: Privacy via distributed noise generation. In S. Vaudenay, editor, *Advances in Cryptology - EUROCRYPT 2006*, pages 486–503, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[9] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In S. Halevi and T. Rabin, editors, *Theory of Cryptography*, pages 265–284, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[10] C. Dwork and A. Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3–4):211–407, aug 2014.

[11] A. Friedman and A. Schuster. Data mining with differential privacy. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '10, page 493–502, New York, NY, USA, 2010. Association for Computing Machinery.

[12] N. Holohan, S. Braghin, P. Mac Aonghusa, and K. Levacher. Diffprivlib: the IBM differential privacy library. *ArXiv e-prints*, 1907.02444 [cs.CR], July 2019.

[13] B. Jayaraman and D. Evans. Evaluating differentially private machine learning in practice. In *28th USENIX Security Symposium (USENIX Security 19)*, pages 1895–1912, Santa Clara, CA, August 2019. USENIX Association.

[14] D. S. Kermany, M. Goldbaum, W. Cai, C. C. S. Valentim, H. Liang, S. L. Baxter, A. McKeown, G. Yang, X. Wu, F. Yan, J. Dong, M. K. Prasadha, J. Pei, M. Y. L. Ting, J. Zhu, C. Li, S. Hewett, J. Dong, I. Ziyar, A. Shi, R. Zhang, L. Zheng, R. Hou, W. Shi, X. Fu, Y. Duan, V. A. N. Huu, C. Wen, E. D. Zhang, C. L. Zhang, O. Li, X. Wang, M. A. Singer, X. Sun, J. Xu, A. Tafreshi, M. A. Lewis, H. Xia, and K. Zhang. Identifying medical diagnoses and treatable diseases by image-based deep learning. *Cell*, 172(5), Feb. 2018.

[15] A. Narayanan and V. Shmatikov. Robust de-anonymization of large sparse datasets. In *2008 IEEE Symposium on Security and Privacy (sp 2008)*, pages 111–125, 2008.

[16] T. J. Pollard, A. E. Johnson, J. D. Raffa, L. A. Celi, R. G. Mark, and O. Badawi. The eICU collaborative research database, a freely available multi-center database for critical care research. *Scientific Data*, 5, 2018.

[17] T. Shapira and Y. Shavitt. Flowpic: Encrypted internet traffic classification is as easy as image recognition. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 680–687. IEEE, 2019.

[18] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani. Toward generating a new intrusion detection dataset and intrusion traffic characterization. *ICISSp*, 1:108–116, 2018.

[19] R. Shokri, M. Stronati, C. Song, and V. Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 3–18, 2017.

[20] R. Singh and G. Srivastav. Novel framework for anomaly detection using machine learning technique on cic-ids2017 dataset. In *2021 International Conference on Technological Advancements and Innovations (ICTAI)*, pages 632–636. IEEE, 2021.

[21] L. Sweeney. Only you, your doctor, and many others may know. *Technology Science*, September 2015.

[22] S. Yeom, I. Giacomelli, M. Fredrikson, and S. Jha. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st Computer Security Foundations Symposium (CSF)*, pages 268–282, 2018.

[23] A. Yulianto, P. Sukarno, and N. A. Suwastika. Improving adaboost-based intrusion detection system (ids) performance on cic ids 2017 dataset. In *Journal of Physics: Conference Series*, volume 1192, page 012018. IOP Publishing, 2019.

[24] B. Z. H. Zhao, A. Agrawal, C. Coburn, H. J. Asghar, R. Bhaskar, M. A. Kaafar, D. Webb, and P. Dickinson. On the (in)feasibility of attribute inference attacks on machine learning models. In *2021 IEEE European Symposium on Security and Privacy*, pages 232–251, 2021.

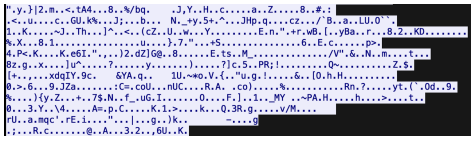
[25] A. Ziller, D. Usynin, R. Braren, M. Makowski, D. Rueckert, and G. Kaissis. Medical imaging deep learning with differential privacy. *Scientific Reports*, 11(13524), 2021.

6 Appendix

Domain	Dataset	Model Type	Reasonable Budget	References
Finance	Adult [7]	Logistic regression	$\varepsilon \geq 0.1$	[5]
Finance	Adult [7]	Decision tree	$\varepsilon \geq 1$	[11]
Healthcare	eICU [16]	Deep learning	$\varepsilon \geq 2.88$	[3]
Healthcare	Pneumonia [14]	Deep learning	$\varepsilon \geq 2.69$	[25]
Network traffic	PCAP data [18]	Logistic regression (payload)	$\varepsilon \geq 0.45$	This paper
Network traffic	PCAP data [18]	Random forest (payload)	$\varepsilon \geq 0.01$	This paper

Table 1: Comparison of privacy budgets (smallest values of ε achieving acceptable utility) between our network traffic domain and other domains in other works. For a fair comparison, we adjusted the size of our training dataset to match the size of datasets from these other domains; therefore, the reasonable budgets shown for the network traffic domain here differ from those in Figure 2.

(a) An example of a plain network packet: a data unit transmitted over a computer network without any additional headers, encapsulation, or encryption.



(b) An example of an encrypted network packet: a packet that has been encoded or scrambled to protect content from unauthorized access or interception.

Figure 1: An illustration of the two types of packets: encrypted and un-encrypted (or plain).

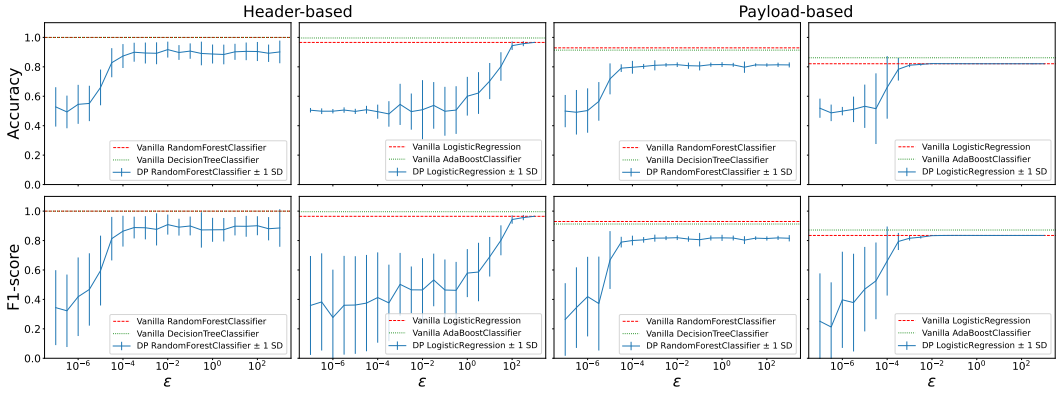


Figure 2: Comparison of model performance between vanilla classifiers and classifiers trained with ε-DP guarantees across a range of ε privacy budget values. Both the accuracy and the F1-score are shown. The first two columns show model results from the header-based approach, while the last two pertain to the payload-based approach. The solid lines represent the average metric value over 30 seeds and the error bars represent one standard deviation in each direction. Dotted lines indicate the non-DP models' performance.