Analyzing the Implications of Packet Anonymization on Data Utility

Gopal Nambiar
Shreyas Madhav Ambattur Vijayanand
Ruthuvikas Ravikumar

December 8, 2022

Abstract

Over the past decade, increased cybersecurity attacks and threats to user privacy have resulted in the ubiquitous adoption of robust data sanitization and anonymization methodologies by organisations around the globe. However, this has also triggered an aura of dissatisfaction among security researchers and analysts who expect high-utility data that has not undergone a lossy anonymization process. There is hence, a dire need to establish a balanced trade-off strategy between protecting user security and ensuring that security researchers attain high-utility data. The first step towards achieving the satisfaction of both the parties involved is to analyse and compare the utility of the data before and after anonymization. This will provide us with an understanding of how different aspects of anonymization impact the utility of the data, helping us take appropriate measures to preserve utility. This project creates a tool (named SHRUG) to collect data through packet traces, anonymize the data, and restructure the data with anonymized packets to analyze the differences between the anonymized and unanonymized data. We use intrusion detection as the task upon which we judge the utility of the data received and focus primarily upon the IP Address Fields. Snort has been employed for this purpose. We have also built a intrusion detection system based on gradient boosting models to predict attacks on both the unanonymized and anonymized data. Additionally, We aim to conduct a comparative analysis of all the anonymization algorithms' impact on data utility from the intrusion detection perspective. We use metrics like accuracy, precision, recall, true positives, true negatives, false positives and false negatives (as a confusion matrix), helping us draw functional inferences that can be used to improve the sanitization process.Also, the codebase for this project is available in :
https://github.com/Ruthuvikas/Infosec-Project/tree/main

1 Introduction

The study of network traffic on real world networks has been quite popular for understanding patterns in online communication for several purposes ranging from improving quality of service to attack prevention. The ubiquitous presence of network communication and its usefulness for different tasks(packet analysis[1][10], network management[17], traffic control[5], network security[24] etc.) has attracted global attention and is a now a field of extensive research. The atomic rise of the domain has also been accompanied by increased attacks and attempts to infiltrate networks to access confidential information transmitted. Attackers may also learn unauthorized information by monitoring the network traffic without even attacking[3].This has sprouted reluctance in the minds of network organization to share unanonymized network information for research as it can be exploited if it falls into the wrong hands. Hence, the network traffic is anonymized to protect sensitive information before sharing. While anonymization does help mask/alter real world traces and enhance privacy, it is a lossy process that inherently affects the usefulness of network data for researchers who hope to derive quality information from it. The research community is not particularly happy about this degradation in the quality of network traces which has created the necessity to find a balance point that satisfies both parties such that both privacy and usefulness of the data are upheld. In this project we aim to study the effects of packet trace anonymization on the utility of the data to infer how different aspects of

anonymization impact the usefulness of the data, which in turn will help us take appropriate measures to preserve utility.

2 Related Literature

Trace anonymization of packet traces is a well-researched field and several packages/ approaches have been in existence and made open source for the security community to explore. The network log files that are used are usually in the form of netflow[26], pcap[28], nfdump[12], pccat, bgpdump[32] and iptable[27]. Some of the popular available techniques and tools include FLAIM[30], TCPdprive,Anontool[7],ip2anonip,Anonym[6] etc . Packages such as FLAIM comprise of a set of anonymization algorithms that can be applied on the packet data to obtain anonymized output. The prominent anonymization approaches are BlackMarker, Randomization, Prefix preserving, Truncation etc. For the purpose of network trace anonymization, researchers have incorporated these approaches to develop methodologies that protect data privacy. Prefix Preserving algorithms[33][2] have attained the spotlight as one of the most frequently implemented algorithms in real world applications due to the high relevance of IP addresses in compromising user information. One example of Prefix preserving algorithm is CryptoPAn[8] which employs pseudonyms to replace the IP addresses while ensuring that address hierarchical relationships are maintained for analytical purposes. IP addresses that possessed the same prefix in the initial trace will maintain this similarity in the anonymized output trace as well. This helped take one step further towards attaining a balance but it is essential to understand anonymization-utility tradeoff to solve the problem.This has motivated researchers to direct their efforts towards studying how different anonymization algorithms impact data usefulness[14] and the selection of the field to be anonymized affects utility of the data[22].

Intrusion Detection tasks[23] focus on protecting and recognizing anomalous activity[13] across a network in order to protect network resources from attackers trying to infiltrate the network[15]. Several softwares[25] have been built and researched in the past for this purpose[9]. Intrusion detection is a quite popular research domain in the field of machine learning[20][19]. Several deep learning algorithms[4][29]have been implemented by researchers in real world applications to detect and classify intrusions. Considering the importance and data utility involved for the same, we choose this task to judge our anonymization algorithms and their impact on our collected network traffic datasets. Another core idea fuelling this research direction is to run multiple anonymization algorithms to generate different views for the same network trace and compare the usefulness against a task over the network, such as attack detection. One of these views may preserve the data utility to the extent that it can be used by researchers effectively[21]. The field which is being anonymized may also play a vital role in how useful the data is. Different network related task and analytical processes make use of different fields in the packet traces. It is therefore important to understand whether the field plays a greater role in affecting anonymization when compared to the algorithm used. With this project, we hope to move one step closer towards understanding the effects of anonymization on utility and tipping the scales towards obtaining the perfect privacy-utility balance.

3 Project Description

In this project, we create SHRUG - a anonymization tool package designed to secure packet traces. The tool will receive packets in the form of pcap files, anonymize the appropriate fields to restrict sensitive information and return the modified pcap files. SHRUG will comprise of six anonymization algorithms that can be implemented for the purpose. The anonymization algorithms as part of our tool include BlackMarker anonymization, Randomization, Prefix Preserving anonymization(CrytoPan)[2], Truncation, Permutation and Reverse Truncation(12 and 20).

As utility of the data can be quite subjective based upon the purpose in which it is used for, we focus the task of intrusion detection to analyse the usefulness of our anonymized readings. We are going to use SNORT and an intrusion detection machine learning model for the purpose of attack recognition before and after anonymization of the packets and comparatively analyse how each anonymization algorithm affects the

accuracy of the intrusion predictions. General accuracy, true positive rates, false positive rates, true negative rates and false negative rates are used as the metric of measurement for the intrusion detection task, which has been portrayed in the form of confusion matrices through the machine learning model. We hope to use this information to provide more clarity on what anonymization algorithms have a lesser impact on data usefulness and how they affect data understanding for intrusion detection.

In addition to employing SNORT to identify intrusions with and without data anonymization, we created our own machine learning intrusion detection model which is powered by the LGBM[11] and XGBoost[16] algorithms. Both of these machine learning algorithms are powered by gradient boosting techniques that are built as a ensemble of decision trees. This will help us predict and classify a given network trace as Benign or as an Attack. We hoped to derive unequivocal conclusions on the effects of anonymization on intrusion detection by using the aforementioned techniques.

4 Datasets

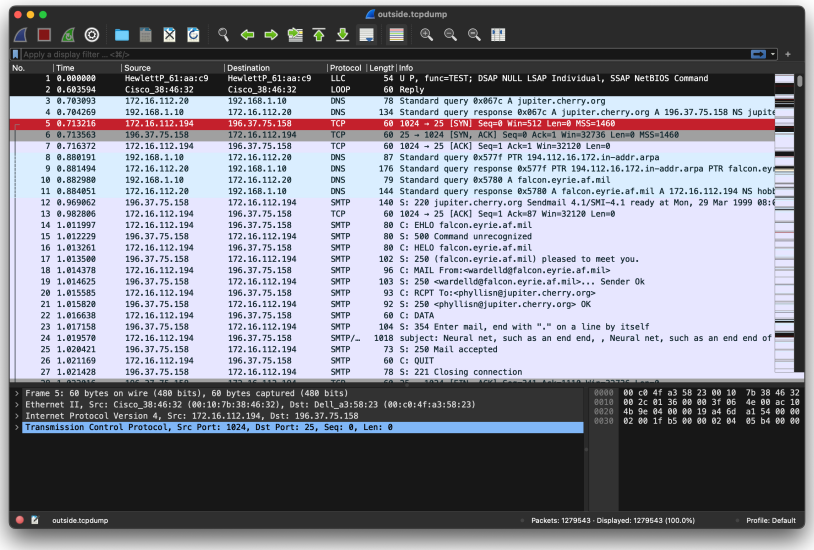
For this project we used two datasets, one was used for the SNORT analysis, and the other for the Machine learning model. To perform the SNORT analysis, we used the 1999 DARPA Intrusion Detection Evaluation Dataset[18]. For the DARPA dataset, the simulation network gathered data around-the-clock. The external tcpdump data files were examined using the tcpslice tool. Packet traces for the span of seven weeks were collected which comprised of attacks in between normal traffic. For the machine learning model we used the CICIDS2017 dataset[31]. The dataset comprises five days' worth of benign and up-to-date common attacks and closely reflects authentic, real-world data (PCAPs). The dataset includes a plethora of attacks including but not limited to DDoS, Infiltration and Brute Force SSH.

5 Anonymization Algorithm and Intrusion Detection Results

5.1 SNORT results

We configured SNORT with the community rules that are open to the public. See Figure 1 for DARPA dataset visualized on Wireshark. Our first step was to run the DARPA dataset through SNORT with the community rules enabled. All rules contributed by members of the open source community or Snort Integrators are referred to as community rules. All Snort users have free access to these rules, which are licensed by GPLv2(GNU General Public License version 2).

Figure 1: DARPA dataset



As seen in Figure 2, the results from running SNORT on the DARPA dataset identifies 2,478 different alerts that match multiple rules in the community ruleset. A sample of these alerts can be seen in Figure 3.

Figure 2: DARPA Output

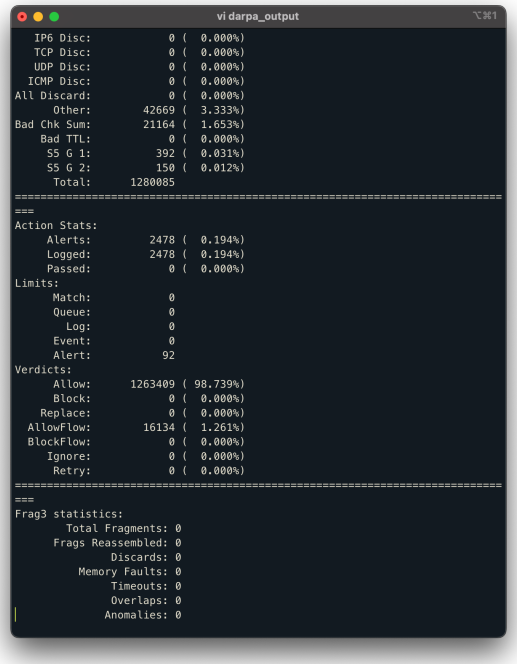


Figure 3: DARPA Alerts

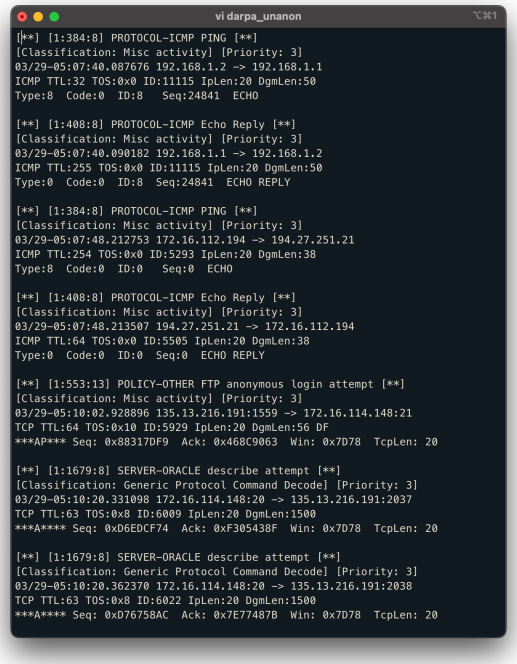
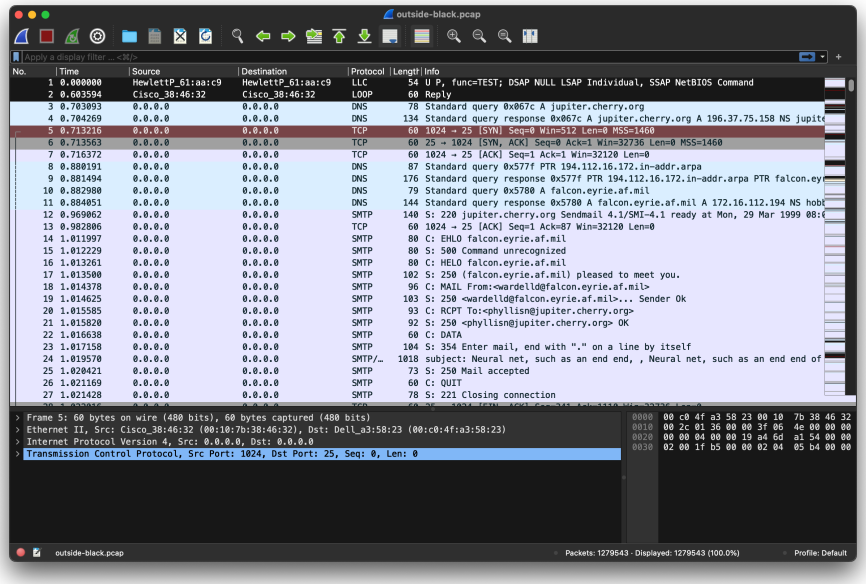


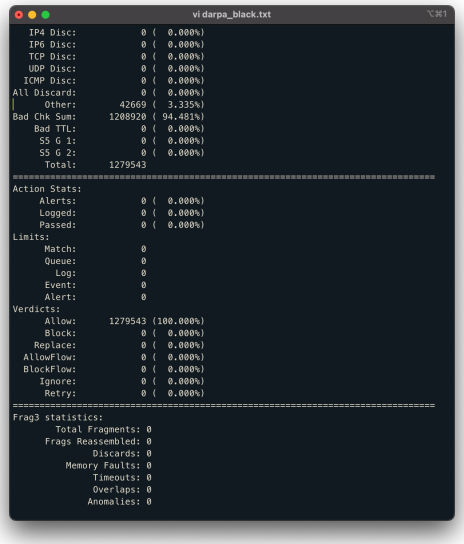
Figure 4: DARPA dataset after Blackmarker Anonymization



We aimed to identify whether anonymizing the given dataset would yield different results. In order to accomplish that, we ran our anonymization algorithms on the DARPA dataset before running it through

SNORT for a second time. As an example, this is what the dataset now looked like after applying the BlackMarker anonymization algorithm (See Figure 4). On running SNORT on the newly formed anonymized DARPA dataset, we see that none of the previously generated alerts show up any longer, indicating that the IP addresses were crucial in the discerning of the alerts. Since the BlackMarker anonymization algorithm prevents SNORT from reading the raw IP address, the events that were previously detected are no longer seen.

Figure 5: SNORT output after BlackMarker anonymization

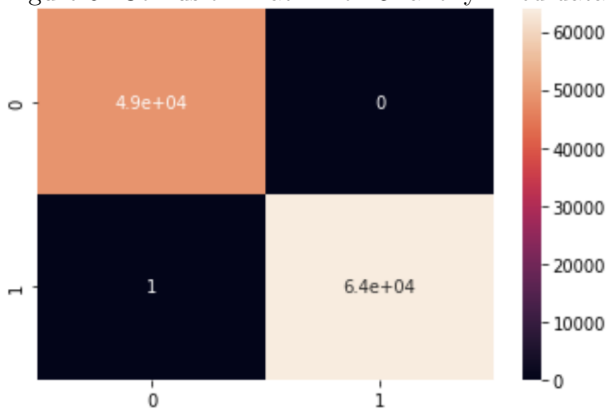


```
vi darpa_black.txt
   IP4 Disc:            0 (  0.000%)
   IP6 Disc:            0 (  0.000%)
   TCP Disc:            0 (  0.000%)
   UDP Disc:            0 (  0.000%)
  ICMP Disc:            0 (  0.000%)
All Discard:            0 (  0.000%)
     Other:        42669 (  3.335%)
Bad Chk Sum:      1208920 ( 94.461%)
     Bad TTL:            0 (  0.000%)
      S5 G 1:            0 (  0.000%)
      S5 G 2:            0 (  0.000%)
       Total:      1279543
================================================================================
Action Stats:
     Alerts:            0 (  0.000%)
     Logged:            0 (  0.000%)
     Passed:            0 (  0.000%)
Limits:
      Match:            0
      Queue:            0
        Log:            0
      Event:            0
      Alert:            0
Verdicts:
      Allow:      1279543 (100.000%)
      Block:            0 (  0.000%)
    Replace:            0 (  0.000%)
  AllowFlow:            0 (  0.000%)
  BlockFlow:            0 (  0.000%)
     Ignore:            0 (  0.000%)
      Retry:            0 (  0.000%)
================================================================================
Frag3 statistics:
        Total Fragments: 0
      Frags Reassembled: 0
              Discards: 0
          Memory Faults: 0
               Timeouts: 0
               Overlaps: 0
              Anomalies: 0
```

5.2 ML Model Results

- **Unanonymized data results**
  When the ML model is trained on the unanonymized dataset it gives the following results:

Figure 6: Confusion Matrix for Unanonymized data.



- **BlackMarker Algorithm**
  BlackMarker is an anonymization algorithm which replaces every information in the field by a fixed value. In our case BlackMarker algorithm was implemented and each field was replaced by 0. When BlackMarker was run through our intrusion detection ML model classifier, the following results were generated.

Figure 7: Confusion Matrix for BlackMarker.

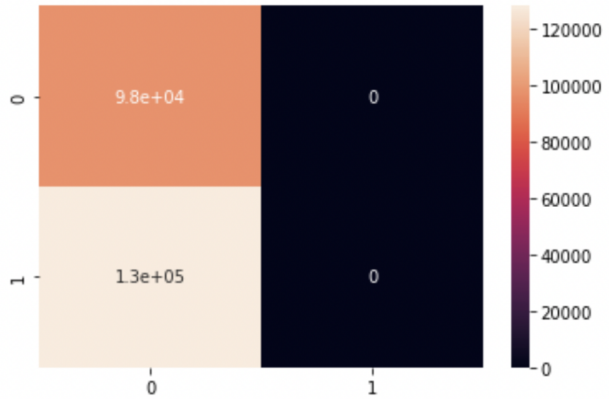


Figure 8: Classification Report for BlackMarker.

	precision	recall	f1-score	support
0	0.43	1.00	0.60	97718
1	0.00	0.00	0.00	128027
accuracy			0.43	225745
macro avg	0.22	0.50	0.30	225745
weighted avg	0.19	0.43	0.26	225745

From the results of the intrusion detection model on the original data and on the blackmarked data, we infer that blackmarker severely affects data utility due to its lossy nature and the ability to detect network attacks is negatively affected. (drop in intrusion detection accuracy from 99.9% on the original format test data to close to 20% on the BlackMarker test data)

- **Permutation Algorithm**
  This algorithm is mainly used for anonymization of the IP address and MAC address fields. In our case, the Permutation algorithm was implemented for the anonymization of the source and destination IP addresses. This algorithm was then run through our intrusion detection ML model classifier and the following results were generated.

Figure 9: Confusion Matrix for permutation.

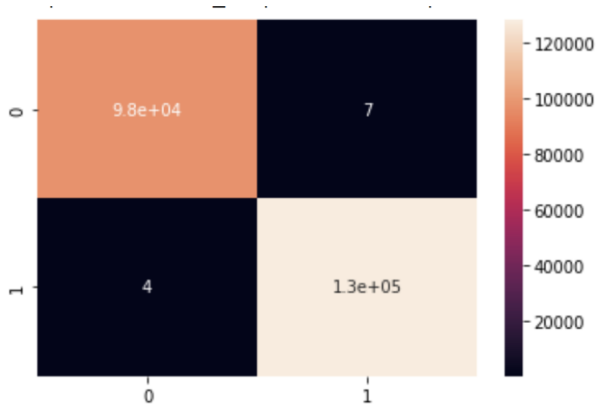
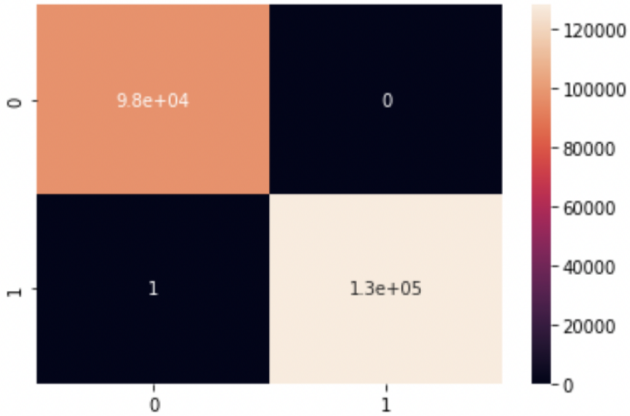


Figure 10: Classification Report for permutation.

	precision	recall	f1-score	support
0	1.00	1.00	1.00	97718
1	1.00	1.00	1.00	128027
accuracy			1.00	225745
macro avg	1.00	1.00	1.00	225745
weighted avg	1.00	1.00	1.00	225745

- **Truncation Algorithm**
  This algorithm is also mainly used on IP addresses. It replaces the n least significant bits of the IP address field with 0. In our case, this algorithm was implemented for n = 12 and n = 20 and ran through the intrusion detection ML model classifier and following results were generated.

Figure 11: Confusion Matrix for Truncation n = 12.



- **Reverse Truncation Algorithm**

This algorithm is similar to Truncation Algorithm. But instead the most significant field will be replaced with zeros. In our case, this algorithm was implemented for n = 12 and n = 20 and ran through the intrusion detection ML model classifier and following results were generated.

Figure 12: Confusion Matrix for Reverse Truncation n = 12.

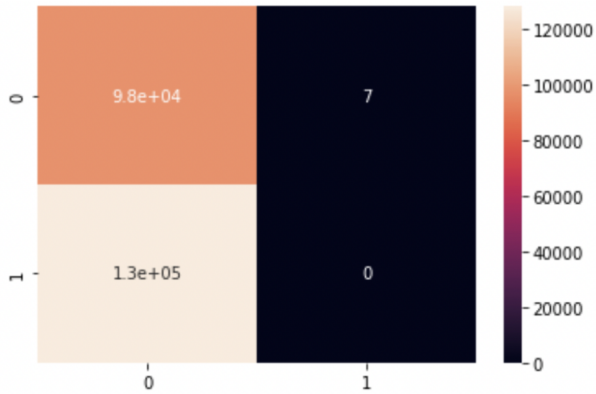


Figure 13: Classification Report for Reverse Truncation n = 12.

	precision	recall	f1-score	support
0	0.43	1.00	0.60	97718
1	0.00	0.00	0.00	128027
accuracy			0.43	225745
macro avg	0.22	0.50	0.30	225745
weighted avg	0.19	0.43	0.26	225745

Figure 14: Confusion Matrix for Reverse Truncation n = 20.

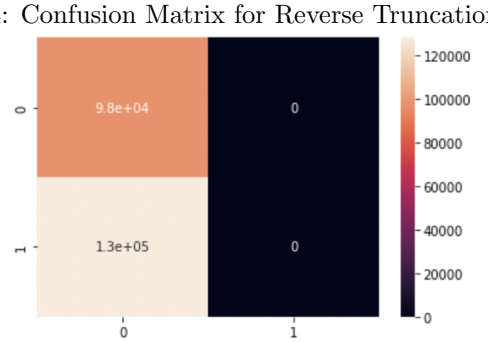


Figure 15: Classification Report for Reverse Truncation n = 20.

	precision	recall	f1-score	support
0	0.43	1.00	0.60	97718
1	0.00	0.00	0.00	128027
accuracy			0.43	225745
macro avg	0.22	0.50	0.30	225745
weighted avg	0.19	0.43	0.26	225745

- **Randomizer Algorithm**
  This algorithm will replace the information value of each field with random values. In our case, this algorithm was applied on the IP address field and ran through the intrusion detection ML model classifier and following results were generated.

Figure 16: Confusion Matrix for Randomizer.

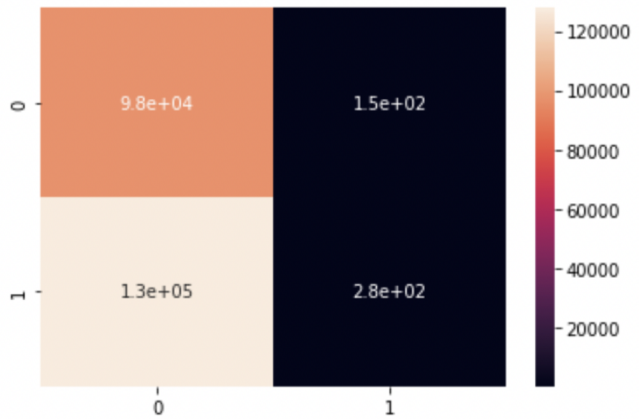


Figure 17: Classification Report for Randomizer.

	precision	recall	f1-score	support
0	0.43	1.00	0.60	97718
1	0.66	0.00	0.00	128027
accuracy			0.43	225745
macro avg	0.55	0.50	0.30	225745
weighted avg	0.56	0.43	0.26	225745

- **Prefix Preserving Algorithm**
  This algorithm is implented through Cryptopan[2], this algorithm preserves the prefix information of the field. In our case, we are using the Crytopan module that is already available and ran through the intrusion detection ML model classifier and following results were generated.

Figure 18: Confusion Matrix for Prefix Preserving.

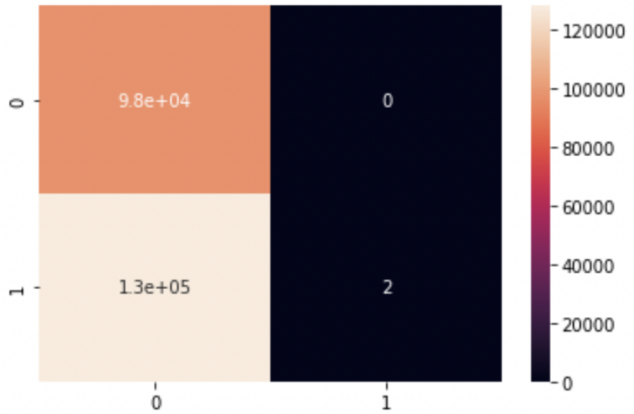


Figure 19: Classification Report for Prefix Preserving.

	precision	recall	f1-score	support
0	0.43	1.00	0.60	97718
1	1.00	0.00	0.00	128027
accuracy			0.43	225745
macro avg	0.72	0.50	0.30	225745
weighted avg	0.75	0.43	0.26	225745

6 Discussions and Conclusion

Based upon both our Machine learning Intrusion Detection results and SNORT analysis, we can get a general idea of how the anonymization of IP addresses affects the utility of the original data for intrusion detection. As expected, the worst performance in terms of utility is Black Marker Algorithm which is a highly lossy process (22% accuracy on our custom intrusion detection model for anonymized CICIDS '17 data and no alerts were generated when run through SNORT for anonymized DARPA). While the original data has the highest utility, the prefix preserving algorithm, permutation and Trunction Algorithm have appreciable performance for intrusion detection, followed by Randomizer, then Reverse Truncations and finally Black Marker. In future, this project can be extended towards exploring much more fields in the packets to assess each one's influence in affecting data utiltiy. Apart from our own machine learning model and SNORT, ZEEK(Bro) can also be used to analyse and evaluate the packets before/after anonymization.

References

[1] M. Bykova, S. Ostermann, and B. Tjaden. "Detecting network intrusions via a statistical analysis of network packet characteristics". In: *Proceedings of the 33rd Southeastern Symposium on System Theory (Cat. No.01EX460)*. 2001, pp. 309–314. DOI: 10.1109/SSST.2001.918537.

[2] Kai Cao et al. "POSTER: An Online Prefix-Preserving IP Address Anonymization Algorithm for Passive Measurement Systems". In: *Security and Privacy in Communication Networks*. Ed. by Bhavani Thuraisingham, XiaoFeng Wang, and Vinod Yegneswaran. Cham: Springer International Publishing, 2015, pp. 581–584. ISBN: 978-3-319-28865-9.

[3] Bogdan Copos et al. "Is Anybody Home? Inferring Activity From Smart Home Network Traffic". In: *2016 IEEE Security and Privacy Workshops (SPW)*. 2016, pp. 245–251. DOI: 10.1109/SPW.2016.48.

[4] Ayesha S. Dina and D. Manivannan. "Intrusion detection based on Machine Learning techniques in computer networks". In: *Internet of Things* 16 (2021), p. 100462. ISSN: 2542-6605. DOI: https://doi.org/10.1016/j.iot.2021.100462. URL: https://www.sciencedirect.com/science/article/pii/S2542660521001037.

[5] Zubair Md. Fadlullah et al. "State-of-the-Art Deep Learning: Evolving Machine Intelligence Toward Tomorrow's Intelligent Network Traffic Control Systems". In: *IEEE Communications Surveys Tutorials* 19.4 (2017), pp. 2432–2455. DOI: 10.1109/COMST.2017.2707140.

[6] Tanjila Farah and Ljiljana Trajković. "Anonym: A tool for anonymization of the Internet traffic". In: *2013 IEEE International Conference on Cybernetics (CYBCO)*. 2013, pp. 261–266. DOI: 10.1109/CYBConf.2013.6617434.

[7] Michalis Foukarakis et al. "Flexible and high-performance anonymization of NetFlow records using anontool". In: *2007 Third International Conference on Security and Privacy in Communications Networks and the Workshops - SecureComm 2007*. 2007, pp. 33–38. DOI: 10.1109/SECCOM.2007.4550304.

[8] "J. Fan, J. Xu, M. H. Ammar and S. B. Moon, "Prefix-preserving IP address anonymization: Measurement-based security evaluation and a new cryptography-based scheme", Comput. Netw., vol. 46, no. 2, pp. 253-272, 2004." In: ().

[9] Muhammad Asim Jamshed et al. "Kargus: a highly-scalable software-based intrusion detection system". In: *Proceedings of the 2012 ACM conference on Computer and communications security*. 2012, pp. 317–328.

[10] Keisuke Kato and Vitaly Klyuev. "Large-scale network packet analysis for intelligent DDoS attack detection development". In: *The 9th International Conference for Internet Technology and Secured Transactions (ICITST-2014)*. 2014, pp. 360–365. DOI: 10.1109/ICITST.2014.7038838.

[11] Ilhan Firat Kilincer, Fatih Ertam, and Abdulkadir Sengur. "A comprehensive intrusion detection framework using boosting algorithms". In: *Computers and Electrical Engineering* 100 (2022), p. 107869.

[12] Ralph Koning et al. "CoreFlow: Enriching Bro security events using network traffic monitoring data". In: *Future Generation Computer Systems* 79 (2018), pp. 235–242. ISSN: 0167-739X. DOI: https://doi.org/10.1016/j.future.2017.04.017. URL: https://www.sciencedirect.com/science/article/pii/S0167739X17305952.

[13] Sailesh Kumar. "Survey of current network intrusion detection techniques". In: *Washington Univ. in St. Louis* (2007), pp. 1–18.

[14] Kiran Lakkaraju and Adam Slagell. "Evaluating the Utility of Anonymized Network Traces for Intrusion Detection". In: *Proceedings of the 4th International Conference on Security and Privacy in Communication Netowrks*. SecureComm '08. Istanbul, Turkey: Association for Computing Machinery, 2008. ISBN: 9781605582412. DOI: 10.1145/1460877.1460899. URL: https://doi.org/10.1145/1460877.1460899.

[15] Aleksandar Lazarevic et al. "A comparative study of anomaly detection schemes in network intrusion detection". In: *Proceedings of the 2003 SIAM international conference on data mining*. SIAM. 2003, pp. 25–36.

[16] Thi-Thu-Huong Le, Yustus Eko Oktian, and Howon Kim. "XGBoost for imbalanced multiclass classification-based industrial internet of things intrusion detection systems". In: *Sustainability* 14.14 (2022), p. 8707.

[17] Allan Leinwand and Karen Fang. *Network Management: A Practical Perspective*. USA: Addison-Wesley Longman Publishing Co., Inc., 1993. ISBN: 0201527715.

[18] Richard Lippmann et al. "Results of the DARPA 1998 Offline Intrusion Detection Evaluation." In: Jan. 1999.

[19] Ziadoon Kamil Maseer et al. "Benchmarking of Machine Learning for Anomaly Based Intrusion Detection Systems in the CICIDS2017 Dataset". In: *IEEE Access* 9 (2021), pp. 22351–22370. DOI: 10.1109/ACCESS.2021.3056614.

[20] Preeti Mishra et al. "A Detailed Investigation and Analysis of Using Machine Learning Techniques for Intrusion Detection". In: *IEEE Communications Surveys Tutorials* 21.1 (2019), pp. 686–728. DOI: 10.1109/COMST.2018.2847722.

[21] Meisam Mohammady et al. "A Multi-View Approach to Preserve Privacy and Utility in Network Trace Anonymization". In: *ACM Trans. Priv. Secur.* 24.3 (Feb. 2021). ISSN: 2471-2566. DOI: 10.1145/3439732. URL: https://doi.org/10.1145/3439732.

[22] Meisam Mohammady et al. "Preserving Both Privacy and Utility in Network Trace Anonymization". In: *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. ACM, Jan. 2018. DOI: 10.1145/3243734.3243809. URL: https://doi.org/10.1145%2F3243734.3243809.

[23] B. Mukherjee, L.T. Heberlein, and K.N. Levitt. "Network intrusion detection". In: *IEEE Network* 8.3 (1994), pp. 26–41. DOI: 10.1109/65.283931.

[24] Mohan V. Pawar and J. Anuradha. "Network Security and Types of Attacks in Network". In: *Procedia Computer Science* 48 (2015). International Conference on Computer, Communication and Convergence (ICCC 2015), pp. 503–506. ISSN: 1877-0509. DOI: https://doi.org/10.1016/j.procs.2015.04.126. URL: https://www.sciencedirect.com/science/article/pii/S1877050915006353.

[25] Nicholas Puketza et al. "A software platform for testing intrusion detection systems". In: *IEEE software* 14.5 (1997), pp. 43–51.

[26] Mohanad Sarhan et al. "NetFlow Datasets for Machine Learning-Based Network Intrusion Detection Systems". In: *Big Data Technologies and Applications*. Ed. by Zeng Deze et al. Cham: Springer International Publishing, 2021, pp. 117–135. ISBN: 978-3-030-72802-1.

[27] Akilan Selvacoumar, Ahmad Ryad Soobhany, and Benjamin Jacob Reji. "P2PRC—A Peer-To-Peer Network Designed for Computation". In: *Proceedings of International Conference on Information Technology and Applications*. Ed. by Abrar Ullah et al. Singapore: Springer Nature Singapore, 2022, pp. 517–526. ISBN: 978-981-16-7618-5.

[28] Leslie F. Sikos. "Packet analysis for network forensics: A comprehensive survey". In: *Forensic Science International: Digital Investigation* 32 (2020), p. 200892. ISSN: 2666-2817. DOI: https://doi.org/10.1016/j.fsidi.2019.200892. URL: https://www.sciencedirect.com/science/article/pii/S1742287619302002.

[29] Geeta Singh and Neelu Khare. "A survey of intrusion detection from the perspective of intrusion datasets and machine learning techniques". In: *International Journal of Computers and Applications* 44.7 (2022), pp. 659–669. DOI: 10.1080/1206212X.2021.1885150. eprint: https://doi.org/10.1080/1206212X.2021.1885150. URL: https://doi.org/10.1080/1206212X.2021.1885150.

[30] Adam Slagell, Kiran Lakkaraju, and Katherine Luo. "FLAIM: A Multilevel Anonymization Framework for Computer and Network Logs". In: *Computing Research Repository - CORR* (Jan. 2006), pp. 63–77.

[31] External Data Source. *Intrusion Detection Evaluation Dataset (CICIDS2017) (07/03/2017 to 07/07/2017)*. en. 2018. DOI: 10.23721/100/1478780. URL: https://www.impactcybertrust.org/dataset_view?idDataset=917.

[32] Taku Wakui, Takao Kondo, and Fumio Teraoka. "GAMPAL: Anomaly Detection for Internet Backbone Traffic by Flow Prediction with LSTM-RNN". In: *Machine Learning for Networking*. Ed. by Selma Boumerdassi, Éric Renault, and Paul Mühlethaler. Cham: Springer International Publishing, 2020, pp. 196–211. ISBN: 978-3-030-45778-5.

[33] Jun Xu et al. "Prefix-preserving IP address anonymization: measurement-based security evaluation and a new cryptography-based scheme". In: *10th IEEE International Conference on Network Protocols, 2002. Proceedings.* 2002, pp. 280–289. DOI: 10.1109/ICNP.2002.1181415.