Poster: Pillars of Sand: The current state of Datasets in the field of Network Intrusion Detection

Gints Engelen
imec-DistriNet
KU Leuven
Leuven, Belgium
gints.engelen@kuleuven.be

Robert Flood
University of Edinburgh
Edinburgh, UK
s1784464@ed.ac.uk

Lisa Liu
School of Engineering and IT
University of New South Wales
Canberra, Australia
l.liuthorrold@adfa.edu.au

Vera Rimmer
imec-DistriNet
KU Leuven
Leuven, Belgium
vera.rimmer@kuleuven.be

Henry Clausen
University of Edinburgh
Edinburgh, UK
henry.clausen@ed.ac.uk

David Aspinall
University of Edinburgh
London, UK

Wouter Joosen
imec-DistriNet
KU Leuven
Leuven, Belgium
wouter.joosen@kuleuven.be

Abstract—**Network Intrusion Detection Systems play a critical role in protecting network architectures from harm. In the past decade, Machine Learning has moved to the forefront of research in this field, with many approaches resulting in great performance on benchmark NIDS datasets. The relevance of these performance results is however directly tied to the quality of the benchmark datasets used for training, which have so far not been subjected to thorough analysis. As part of our work, we have performed a large-scale manual investigation of the most commonly used publicly available NIDS datasets, where we have uncovered numerous errors due to problems in data pre-processing, attack simulation and labelling. We also highlight the lack of variability in both benign and malicious traffic, which often renders the classification task trivial. To quantify this variability, we have devised an automated methodology that can be applied without requiring expert domain knowledge. Nevertheless, we believe it is vital for any NIDS benchmark datasets to undergo a thorough manual analysis before being widely adopted. As a follow-up of our previous work where we provided an improved version of the CICIDS 2017 dataset, we are also actively working on improving the CSE-CIC-IDS 2018 dataset, which we intend to release to the research community.**

Index Terms—**network intrusion detection, machine learning, benchmark dataset, data collection.**

1. Introduction

Network Intrusion Detection Systems (NIDS) are devices that are placed at strategic locations within a network infrastructure in order to protect it from internal and external threats. These threats range from attempts to gain unauthorised access to the network, to large-scale DDoS attacks that aim to disrupt the services of the network's hosts. With attackers becoming more sophisticated, new threats are emerging on a daily basis, and traditional rule-based Intrusion Detection Systems are at risk of being overwhelmed by the sheer number of zero-day attacks. This is why, over the past decade, NIDS research has gravitated towards Machine Learning (ML), which does not rely as much on manual updates in order to detect new attacks.

Research in this field has shown a lot of promise. High performance results on benchmark NIDS datasets [1]–[3] seem to indicate that using ML for NIDS is a solved problem. However, the obtained results are heavily dependent on the quality of the used datasets. While some research has highlighted the disparity between traffic found in these datasets and that of a real-world environment [4], the network traffic found in these datasets has generally not been subjected to a thorough manual investigation.

In earlier work, we have shown that the CICIDS 2017 dataset [5] suffered from a multitude of issues which made its use as a benchmark dataset questionable [6]; as part of that work we released an improved version of this dataset. Following up on that, we have performed a large-scale manual analysis of five modern and widely-used NIDS datasets, where we uncovered a wide range of issues pertaining to labelling, attack simulation, documentation, and network traffic realism.

As part of this work, we are working on improving the CSE-CIC-IDS 2018 dataset, specifically making sure that the ground truth of all labels is as accurate as possible. Using our fixed version of the CICFlowMeter tool [7] guarantees that this version of the dataset is also free from the kind of artefacts (*such as TCP Appendices*) that were present in the CICIDS 2017 dataset [6].

Finally, while we believe that any modern NIDS dataset will have to be subjected to intense scrutiny before seeing widespread adoption in the field, we also propose some automated techniques that could help give an overview of the overall quality of a NIDS dataset.

2. Background and Motivation

Machine Learning has already been applied with great success in other fields, notably due to its capability of learning a task while requiring minimal human supervision. Typical ML approaches do not directly operate on raw data; instead, the data first goes through a process called *feature extraction* before being fed into the model.

When it comes to network traffic, a common way to pre-process the data is by grouping packets together into so-called *flows*, identified by a 5-tuple of {*Src IP, Dst IP, Src Port, Dst Port, Protocol*}. Typical flow-level features are, for example, *total forward packets*, *bytes sent per second*, and *total flow duration*.

The vast majority of ML approaches require significant amounts of training data. In contrast to fields like Computer Vision and Natural Language Processing, where training data is abundantly present, collecting training data in the field of Network Intrusion Detection is more challenging. This is mainly due to the inherent privacy concerns that come with collecting network traffic.

Research in this domain has tried to alleviate the problem of a lack of datasets by creating synthetic datasets in a controlled network environment, which were then made publicly available. Given the high performance results obtained by ML-based approaches on these datasets, it is striking that adoption of ML for NIDS in industry has been comparatively slow. One of the problems lies in the fact that these synthetically generated datasets were immediately adopted in the field, with comparatively little research done analysing the quality and validity of these datasets.

Due to the large variety and variability of network traffic as well as a rapidly evolving threat landscape, Network Intrusion Detection in a real world setting is hard. Benchmark datasets serve to certify a model's capability of successfully operating in such a complex real world environment, and so the classification of traffic in a benchmark dataset should be equally difficult. Lastly, correct ground truth of all labels in a benchmark dataset is crucial in order to verify that the model successfully learned the classification task at hand.

3. Datasets and Methodology

In order to help improve the quality and utility of NIDS datasets, we perform a large-scale manual analysis of 5 widely-used datasets. As part of our selection criteria we strive to include datasets that have been cited numerous times (>100), and where authors published both the raw traffic files (PCAPs) as well as feature-extracted data (flows). We also included 2 IoT-based datasets - despite not meeting the citation criteria - in order to diversify the types of network architectures that we analyse.

The selected datasets are CSE-CIC-IDS 2018 [8], UNSW NB15 [9], TON-IoT [10] and IoT-23 [11]. Due to some updates in our methodology we also revisited CICIDS 2017 [5], despite already having analysed it once in our previous work [6].

3.1. Verifying label correctness

In this step we focus our efforts on the malicious labels. When analysing the validity of these labels, we verify that the underlying network traffic actually exhibits characteristics that would be indicative of malicious traffic of the type specified by the label. Additionally, we verify as much as possible that the attack has been correctly implemented and that it has an appreciable effect on the target victim.

One key aspect that should facilitate this task is the documentation of the dataset generation process published by the dataset authors. For all analysed datasets, however, we found that a significant amount of information was lacking when it comes to the way that most attacks were implemented. Instead, we had to rely on manual analysis of the raw network traffic in order to understand the nature of the attack, and verify correctness of its given label.

For CSE-CIC-IDS 2018, as part of this manual analysis we also took the opportunity to fully reverse-engineer and subsequently improve on the existing labelling logic.

3.2. Network traffic variety

As mentioned previously, the varied nature of both benign and malicious network traffic should make the classification task quite hard. Moreover, we can expect that a smart adversary will try to make his malicious traffic appear to be as similar to benign traffic as possible. For simpler attacks such as DoS and DDoS, we expect that the attack traffic shares some characteristics with large spikes of normal traffic. In summary, malicious traffic found in benchmark datasets should not be *too* dissimilar to benign traffic, as this would otherwise simplify the classification task considerably.

In order to measure this similarity, we employ 4 different ”metrics”:

- Overlap Region Volume: Calculates the amount of overlap between data points of 2 different classes
- Maximum Fischer's Discriminant Ratio: Gives an indication of how easily data from 2 classes can be separated
- 1-Nearest Neighbours Ratio: Assigns to each datapoint the class of its nearest neighbour. A very high number for a given class indicates that there is very little overlap with other classes, again pointing to a trivial classification task for the model.
- Few-Shot Learning Accuracy: Using a Siamese model, we feed it 2 samples at a time and have it learn whether they belong to the same class or not. We then observe how many samples it needs to train on before reaching an accuracy score of over 95%.

4. Results

4.1. Manual Analysis

As part of our manual analysis, we found numerous problems which can be grouped into five categories:

- Repetitive Attack traffic: Attack traffic that is repetitive to the extent that the classification problem is considerably simplified. While some attack classes exhibit this characteristic as part of their nature (e.g. Portscan traffic), we expect most attack classes to each contain a decent variety of attack traffic.
- Unclear Attack: Attacks where we are unable to identify the attack taking place or why the attack

Mistake Type	Majority Affected	Partially Affected	Does Not Apply
Repetitive	22	5	40
Unclear Attack	10	1	56
Ineffective Attack	5	1	61
Simulation Artifacts	16	1	50
Mislabelled Data	6	14	47
Total	41	16	10

TABLE 1. NUMBER OF ATTACK CLASSES SUFFERING FROM EACH PROBLEM

should be considered as malicious. For instance, downloading a file from a server.

- Ineffective Attack: The attack fails to have an effect on the target victim.
- Simulation Artifacts: Unintended side effects of the dataset generation process which lead the trained classifier to exhibit shortcut learning. This category includes artifacts from the feature extraction process.
- Mislabelled Data: The attack class contains benign traffic or attack traffic from another class.

A breakdown for the presence of these problems across all classes of the five datasets is given in table 1. We say that a class is *majorly* affected by a problem if it is present in more than 50% of the flows. If the number of flows is below that threshold, we say that the class is *partially* affected.

4.2. Automated Traffic Analysis

So far, results of our automated analysis help explain why classifiers trained on these dataset easily reach very good performance numbers: for almost every malicious class in our tested datasets, the Overlap Region Volume was below 0.1. This means that the distributions of this class and the benign traffic overlapped minimally. For many cases the ORV value was 0, meaning the overlap limits itself to at most one single datapoint. The Maximum Fischer's Discriminant and 1-Nearest Neighbours ratios further confirmed this.

Across virtually all classes, our Siamese network was capable of achieving over 95% accuracy after only having seen 200 samples. In some cases, such as the UNSW NB15's *Worms* attack, it achieved over 99% accuracy after seeing less than 50 samples.

5. Discussion and Future Work

Based on our findings, we can ascertain two things.

Firstly, for each dataset our manual analysis revealed numerous problems with the data labelling process, attack setup and/or feature extraction process. Secondly, our automated analysis found that the malicious traffic shared little similarities with underlying benign traffic, meaning that the classification task is far from challening.

We believe the above-mentioned issues are serious to the extent that these datasets can not be claimed to be representative of real-world traffic. As such, good performance results obtained by classifiers on these benchmark datasets can not be expected to generalise to a live production environment.

As part of our work, we are also working on an algorithm that would help detect mislabeled samples in a NIDS dataset, as manual analysis of each individual data point is infeasible. Finally, we will publish the fixed CSE-CIC-IDS 2018 dataset, as well as the extended documentation containing the details of all of our findings on our website [7].

6. Conclusion

In our work, we are performing a large-scale analysis of five popular NIDS datasets. We uncovered numerous problems in the dataset generation process, and additionally found that the malicious traffic found in these datasets is often easily distinguishable from benign traffic. As a result, existing benchmark datasets are not suitable to evaluate a classifier's capability to be succesfully deployed in a real-world environment. As part of our work we hope to pave the way for better NIDS datasets, and we stress again that we believe a dataset should be subjected to a thorough manual analysis by domain experts before being adopted by the wider research community.

References

[1] L. Leichtnam, E. Totel, N. Prigent, and L. Mé, "Sec2graph: Network attack detection based on novelty detection on graph structured data," in *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*. Springer, 2020, pp. 238–258.

[2] A. R. Gad, A. A. Nashat, and T. M. Barkat, "Intrusion detection system using machine learning for vehicular ad hoc networks based on ton-iot dataset," *IEEE Access*, vol. 9, pp. 142 206–142 217, 2021.

[3] J. Yoo, B. Min, S. Kim, D. Shin, and D. Shin, "Study on network intrusion detection method using discrete pre-processing method and convolution neural network," *IEEE Access*, vol. 9, pp. 142 348–142 361, 2021.

[4] H. Hindy, D. Brosset, E. Bayne, A. K. Seeam, C. Tachtatzis, R. Atkinson, and X. Bellekens, "A taxonomy of network threats and the effect of current datasets on intrusion detection systems," *IEEE Access*, vol. 8, pp. 104 650–104 675, 2020.

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

[6] G. Engelen, V. Rimmer, and W. Joosen, "Troubleshooting an intrusion detection dataset: the cicids2017 case study," in *2021 IEEE Security and Privacy Workshops (SPW)*. IEEE, 2021, pp. 7–12.

[7] "Extended documentation of the corrected CICFlowMeter tool, and the regenerated CICIDS 2017 dataset." https://downloads.distrinet-research.be/WTMC2021.

[8] "CSE-CIC-IDS 2018 Dataset," https://www.unb.ca/cic/datasets/ids-2018.html, accessed: 2022-04-30.

[9] N. Moustafa and J. Slay, "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *2015 military communications and information systems conference (MilCIS)*. IEEE, 2015, pp. 1–6.

[10] N. Moustafa, "A new distributed architecture for evaluating ai-based security systems at the edge: Network ton_iot datasets," *Sustainable Cities and Society*, vol. 72, p. 102994, 2021.

[11] "IoT-23 dataset," https://www.stratosphereips.org/datasets-iot23, accessed: 2022-04-30.

Pillars *of* Sand

The state of Datasets in Network Intrusion Detection

Gints Engelen[1], Robert Flood[2], Lisa Liu[3], Vera Rimmer[1], Henry Clausen[2], David Aspinall[2], Wouter Joosen[1]

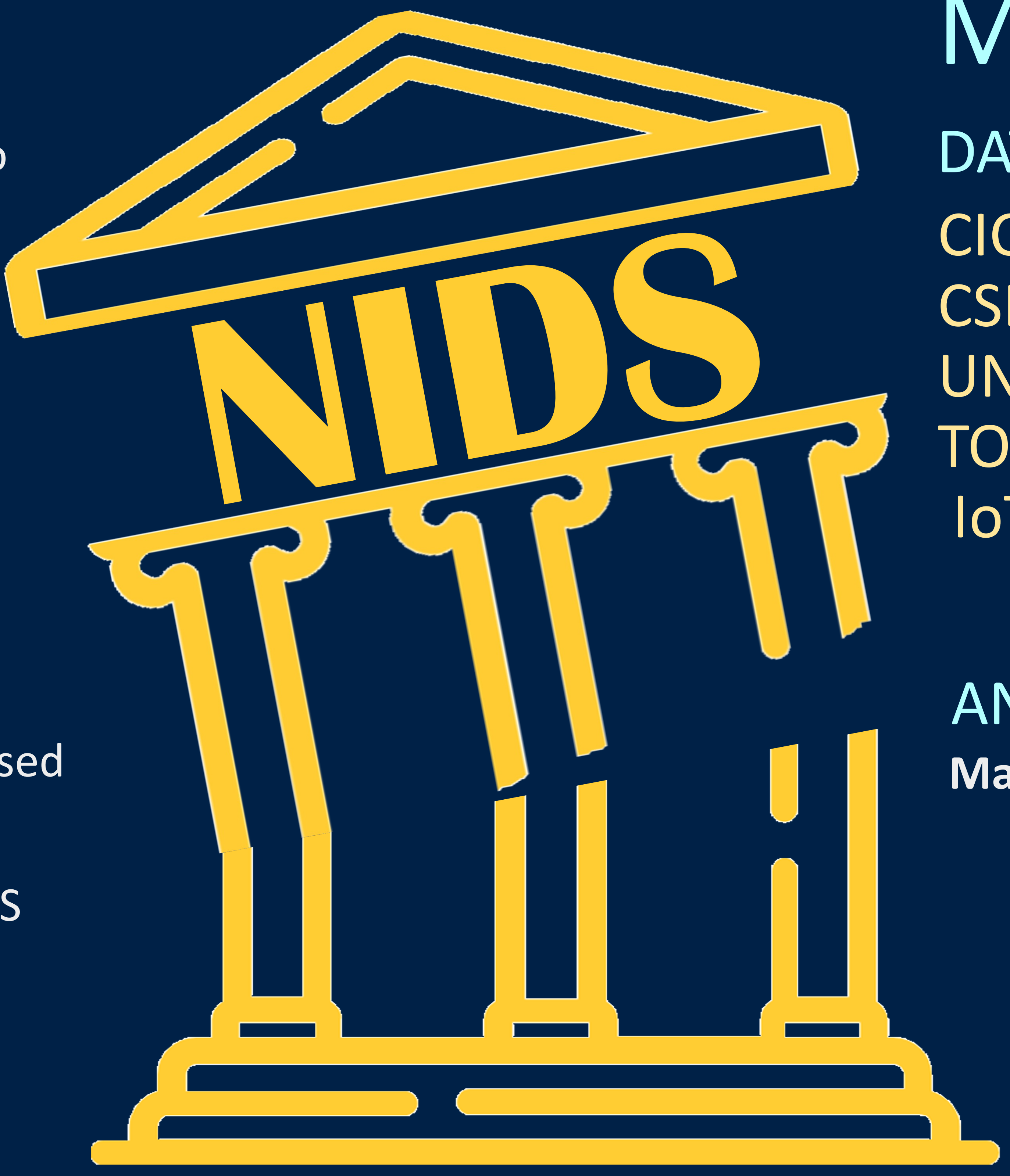
[1] imec-Distrinet KU Leuven, [2] University of Edinburgh, [3] School of Engineering and IT - University of New South Wales

INTRODUCTION

Research has applied Machine Learning to Network Intrusion Detection in the past decade, with great results obtained on benchmark datasets. However, whether these results generalize to a real-world setting depends on the quality of these datasets.

As part of our work, we contribute the following:

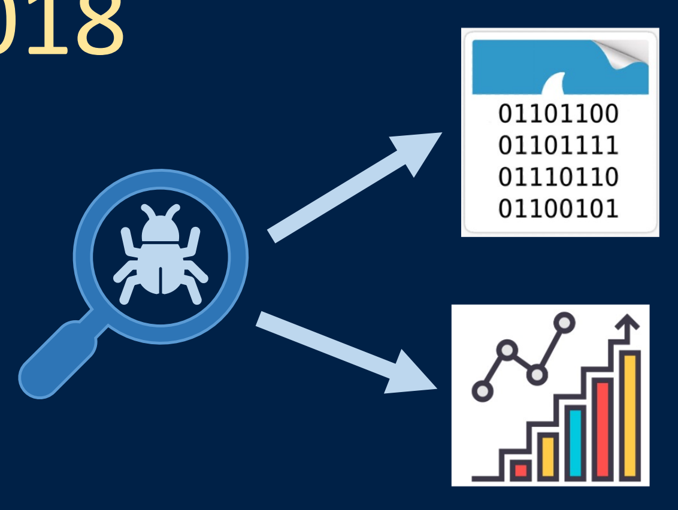
- A large-scale manual and automated analysis of 5 modern and commonly used NIDS datasets
- An improved version of the CSE-CIC-IDS 2018 dataset
- Identify key requirements for future NIDS datasets



METHODOLOGY

DATASETS

CICIDS 2017
CSE-CIC-IDS 2018
UNSW NB15
TON-IoT
IoT-23



ANALYSIS

Manual CSV and PCAP analysis

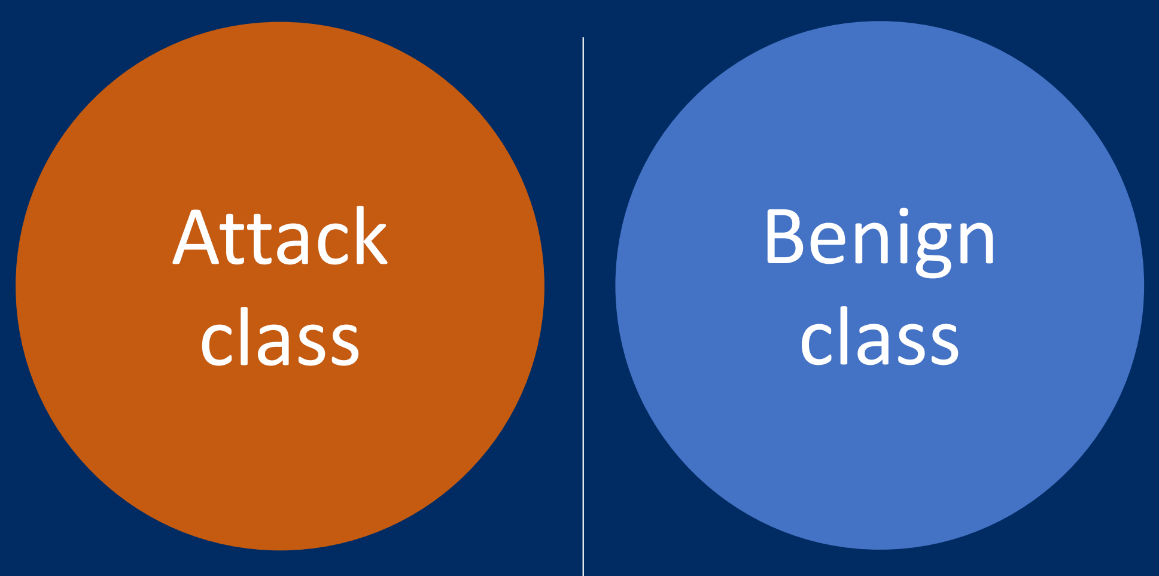
Automated statistical and ML-based analysis:

- Fischer's Discriminant Ratio
- Overlap Region Volume
- 1 Nearest Neighbours Ratio
- Few-shot Learning Accuracy

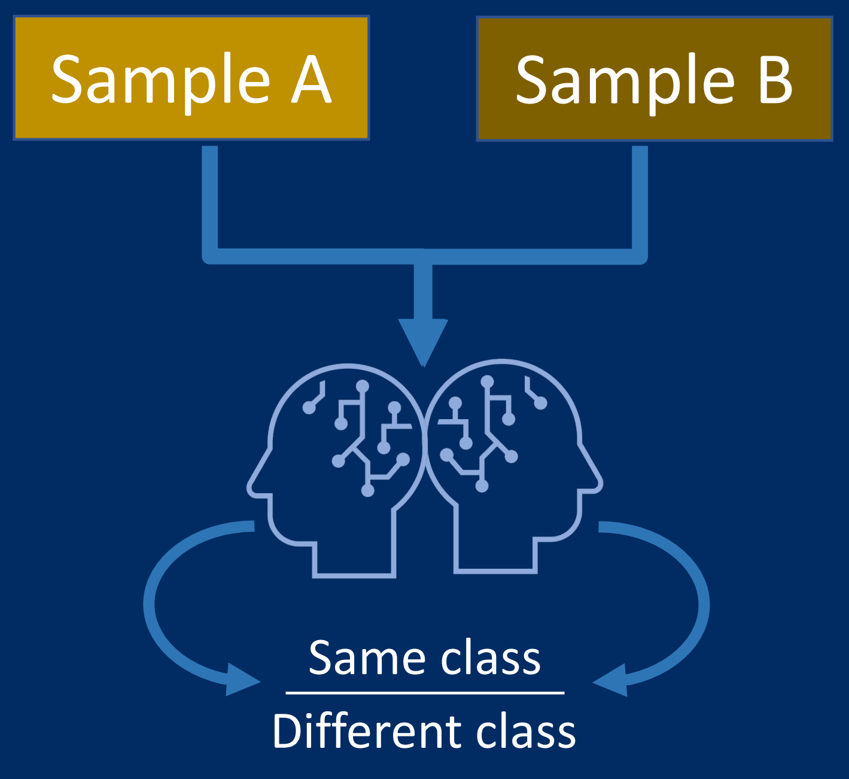
RESULTS

Our analysis uncovered several categories of problems that occurred systematically in these datasets. Below are the number of classes that are affected by each problem.

Problem type	Majorly affected	Partially affected	Does not apply
Repetitive	22	5	40
Unclear Attack	10	1	56
Ineffective Attack	5	1	61
Simulation Artefacts	16	1	50
Mislabelled Data	6	14	47
Total affected classes	41	16	10



We further found that there was little to no overlap between almost any given malicious class and the benign class, significantly trivialising the classification task.



In our few-shot learning approach, our Siamese Network managed to achieve 95% accuracy across virtually all classes after seeing not more than 200 samples.

CONCLUSION

Network Intrusion Detection in a real-life setting is supposed to be hard. Accordingly, benchmark datasets used for evaluation of NIDS approaches should offer an equally challenging classification task. We found however that these datasets suffer from numerous problems that render them unsuitable for NIDS benchmarking. Future datasets should improve on traffic realism and variety, provide better documentation, and ultimately undergo extensive manual analysis before being adopted by the research community.

WHAT NEXT?

- ➢ Develop automated approach to correct mislabelling of traffic flows.
- ➢ Investigate datasets that use real traffic, such as UGR-16 and MawiLAB.
- ➢ Collect our own traffic for realism comparison.

Project website:

DistriNet
KU LEUVEN