# Attack originality detector using Machine Learning

Charlotte Millan[1,2,*], Lotfi Chaari[2,*], Amel Aissaoui[2] and Eric Gilbert[1]

[1]CS Group - Embedded Industrial cybersecurity systems, firstname.surname@cs-soprasteria.com
[2]Toulouse INP, IRIT, University of Toulouse, firstname.surname@toulouse-inp.fr

**Abstract**

Security Information and Event Management (SIEM) systems are critical tools in modern cybersecurity, designed to aggregate, analyze, and correlate security data for real-time threat detection. While effective against known and over threats, SIEM systems often struggle to detect subtle variations in attack patterns. These variations, though minor, can signal the evolution of an attack or serve as precursors to more severe attacks, such as advanced persistent threats (APTs). Addressing this gap requires innovative detection methods capable of identifying such "original" attacks. In this paper, we propose a model combining multiple weak classifiers, each employing a different classification technique to highlight features indicative of original attacks. A voting system is then applied to the results, classifying as "original" attack segments that fail to achieve consensus. Our method was tested and validated using DoS and Probes attack groups in the NSL-KDD dataset, a benchmark for intrusion detection systems.

**Keywords**

Artificial Intelligence, Machine Learning, Cybersecurity, classification, Voting system

## 1. Introduction

Today, malicious actors only need a functioning attack scenario to achieve their objectives, and the more interconnected objects are, the greater the attack surface (number of exploitable entry points and vulnerability sources). To cope with the ever-increasing number and complexity of attacks on their information systems, business and public authorities are turning to platforms for monitoring their network operations: Security Information and Event Management (SIEM) software solutions. They must be capable of continuously collecting and aggregating large quantities of data indicating possible security flaws, known as alerts. Captured at different points on the network and emitted by heterogeneous sources (antivirus, firewalls, network probes, authentication servers, etc.), these alerts have three main characteristics: velocity, volume, and variety.

One of the essential functions of SIEMs is to correlate the security alerts generated by the same intrusion attempt, by consolidating alerts and contextualizing them. Clear information can therefore be provided to Security Operation Center (SOC) analysts who are responsible for decision making and often overwhelmed by the exploitation of *0-days* vulnerabilities and the number of false positives. The use of data science in cybersecurity can help to correlate events, identify recurring patterns, and detect abnormal behaviors to improve SIEM performances [1].

In [2], the authors propose a systematic approach to convert scores from models into probabilities ready for use in cybersecurity models. To do this, they consider each decision model as a classifier. Thanks to their multiple-score calibration approach, the scores from their three systems can be combined and converted into a single probability measure that is more relevant for decision-making. They conclude that Platt's logistic regression performs acceptably over a wide range of experiments. They also point out that multi-score calibration and score extension to higher dimensions, both increase performance in most cases.

The study in [3] uses anomaly scores to detect Advanced Persistent Threats (APTs). To predict the attack

*Joint National Conference on Cybersecurity (ITASEC & SERICS 2025), February 03-8, 2025, Bologna, IT*
*Corresponding author.
charlotte.millan@cs-soprasteria.com (C. Millan); lotfi.chaari@toulouse-inp.fr (L. Chaari); amel.aissaoui@toulouse-inp.fr (A. Aissaoui); eric.gilbert@cs-soprasteria.com (E. Gilbert)
http://lotfi-chaari.net (L. Chaari)
0000-0002-3590-0370 (L. Chaari)

stage, the authors proposed an approach for detecting APT occurrences and systematically mapping the corresponding events to stages in the execution chain, using anomaly scoring and Machine Learning (ML) based on a Bayesian network model. Although his method works with raw data, it is more effective when integrated with SIEM. To cope with Distributed Denial of Services (DDoS) attacks, the authors in [4] propose a method using various supervised ML methods for classification and prediction. Using Random Forest models with an accuracy of 99 %, a $K$-Nearest-Neighbors (KNN) method with 96 % and a Logistic Regression method with 85 %, they obtained good results for DoS attack detection where experiments have been conducted on the NSL-KDD dataset. As regards unsupervised learning, the authors in [5] emphasis on the fact that such methods learn what is normal for a dataset and are therefore able to find variations in unclassified sets such yet unknown attack. To detect anomalies, they use a combination of six unsupervised ML models. They conclude that unsupervised ML methods are efficient for generalizing and detecting unknown patterns but may fail to control false positive rates. The authors therefore recommend using hybrid models that combine supervised and unsupervised learning.

Based on the current state of the art, it becomes clear that existing methods mainly target well-known attacks, leaving a critical gap in detecting more nuanced, evolving threats. SIEM systems often struggle to identify subtle variations in attack patterns that may indicate the evolution of an attack or act as early warnings for more severe threats. To address this limitation, our work focuses on detecting these variations, characterizing what we define as "original attacks". We believe that detecting these original attacks will enhance threat detection capabilities by uncovering previously unnoticed attack. Accordingly, we propose a novel method to classify attacks originality using a hierarchical classification model made up of three weak classifiers and a voting system to point out consensual classifications of attacks based on recorded features.

In this work, we propose a method to classify attacks originality using a hierarchical classification model made up of three weak classifiers and a voting system to point out consensual classifications of attacks based on recorded features. We considered that's original attack will be a sign of evolution of a well-known attack, or the first of an advanced attack, and detecting them will be improved the SIEM. We start working on this subject to find low signals, premises of APT attacks. This paper is organized as follows. We first present the used data and the proposed method in Section 2. Experimental results is then presented in Section 3. Finally, conclusions and future work are drawn in Section 4.

## 2. Materials and Methods

Detecting attack originality is a complex challenge, as originality cannot be easily defined or analytically modeled. To address this, we adopt a data-driven approach, leveraging ML method, to learn from data in an unsupervised learning framework. We first describe the used NSL-KDD dataset, and then go through the proposed model.

### 2.1. Materials

The data used in this study are extracted from the NSL-KDD database [6]. Each entry in this dataset is made up of 41 features, 38 of which are in numeric format. The other three are relate to the type of used protocol, the type of connection to the server and the state of the connection at the capture time. Figures 1 and 2 illustrate two segments from the NSL-KDD dataset, where, for instance, the second and third features stand for the protocol and service, respectively. The last field, frame in red, indicates the label associated with each entry: attack type or normal segment.

```
(0,udp,private,SF,105,146,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,
0.00,1.00,0.00,0.00,255,254,1.00,0.01,0.00,0.00,0.00,0.00,0.00,0.00,.normal)
```

**Figure 1:** Entry of NSL-KDD dataset, normal segment.

```
(0,icmp,ecr_i,SF,508,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,31,31,0.00,0.00,0.00,
0.00,1.00,0.00,0.00,255,255,1.00,0.00,1.00,0.00,0.00,0.00,0.00,0.00,.smurf.)
```

**Figure 2:** Entry of NSL-KDD dataset, smurf attack segment.

It is worth noting that the NSL-KDD dataset is the most widely used for intrusion detection studies, as presented by the authors of: [5],[7], [8].
The methodology we have used begins with the presentation of the relevant datasets: in the case of this study, we need a dataset containing normal attacks and flows, and with the distribution of known attacks.

## 2.2. Methods

As illustrated in Figure 2, our method processes attack segments that are precategorized according to their classes and analyzing them to classify each as either an original or a known (normal) attack. Figure 3 details the proposed pipeline to analyze data segments and detect attack originality.

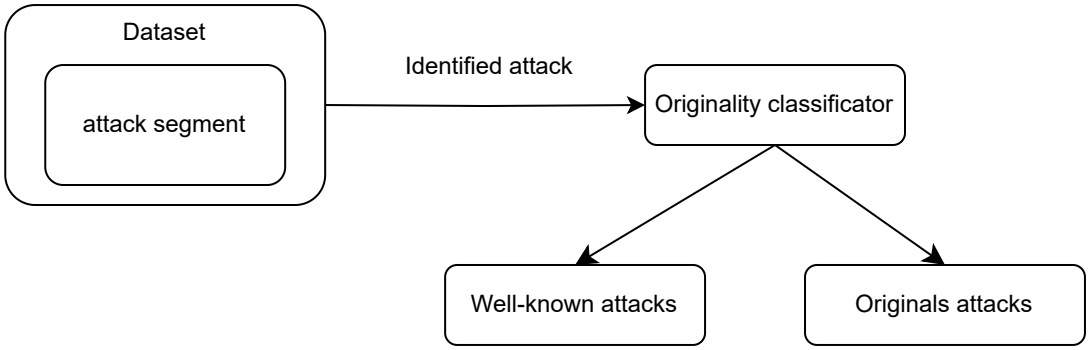


**Figure 3:** Proposed data analysis pipeline.

As illustrated in Figure 3, detected attacks are analyzed to decide whether they are original or not, compared to detected attacks of the same type.

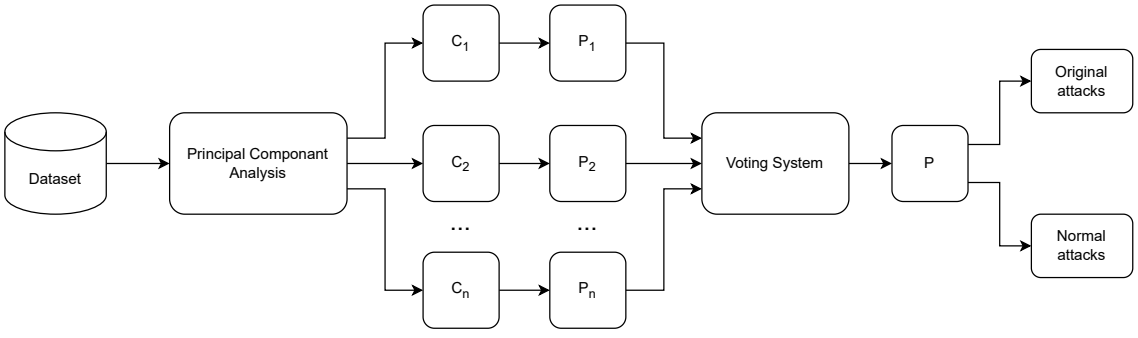


**Figure 4:** Flowchart of the proposed ML method for anomaly originality classification.

The proposed methodology is illustrated in Figure 4, where a Principal Component Analysis (PCA) module is used as a dimension reduction technique in order to project the dimensions of the dataset into a lower dimension space. Here, we choose two components, as it was enough for our system. The classification problem is addressed by using $N$ weak classifiers and a voting system. Each classifier $C_i$ generates a binary output $P_i$, defined as follows:

$$P_i = \begin{cases} 1, & \text{if normal} \\ 0, & \text{if original.} \end{cases} \tag{1}$$

Each model returns 1 if the segment is normal (usual attack), 0 if is not (original attack), and the voting system calculates the following weighted decision score:

$$P = \sum_{i=1}^{N} \omega_i P_i \tag{2}$$

where $N$ is the number of weak classifiers and $(\omega_i)_{1 \leq i \leq N}$ are the weights associated to each weak classifier. In our case, the same weight is associated to all classifiers ($\forall i \in \{1, \ldots, N\}, \omega_i = 1/N$).
It is worth noting that a consensus is reached if and only if $P \in \{0, 1\}$. The decision of the voting system is set to True (normal attack) if $P \in \{0, 1\}$. For all the remaining cases, the attack is classified as original (presents significant deviation from the general behavior of the same class of attack). In our study, we used three weak classifiers ($C_1$, $C_2$, $C_3$):

1. **K-means ($C_1$)**: classifies data based on their similarity. This unsupervised method does not consider the type of attack, only the similarity of the segments between each other's. The number of clusters has been set to the number of attack types in the dataset. As K-means is not supposed to detect outliers, we use the distance between centroids of the clusters and the segment itself. A threshold is adjusted to determine whether a segment may be considered belonging to one of the clusters or not. Each attack's type has a maximal distance between the point and the center, attacks beyond are considered as original.
2. **$K$-nearest-neighbors ($C_2$)** : KNN classifies data according to the distance between neighbors. This unsupervised learning algorithm is generally chosen for its simplicity. The number of neighbors $K$ is set to the number of attack types: to detect original attacks, we took an interest for the segment misclassified by the algorithm. To define outlier, we use decision boundary: we define a minimum distance between the points and the boundary, and all points below this distance are then considered original.
3. **One Class SVM ($C_3$)** : 1C-SVM is an unsupervised classifier that detects outliers using a given kernel (Gaussian in our case). It returns 0 if the segment is outlier, 1 if it is normal.

For one entry segment, and once the weak classifiers are learned, their outputs ($P_1$, $P_2$ and $P_3$) are used as input for the voting system. These inputs are binary integers which take 1 when the attack is normal, and 0 if original. The voting system classifies a segment as normal attack (non-original) if and only if all weak classifiers agree.

## 3. Experimental results

In this section, we present the experimentation and the results of the proposed method on the widely used NSL-KDD dataset [6, 9].

### 3.1. Protocol

In the following, we will focus on two groups of attacks: Denial of Services (DoS) and Probes which are the most presents in the dataset. The other groups of attacks represent 2% of the dataset, which is too unbalanced to be used without bias. Then, we have divided the dataset in two parts: 80% for training and 20% for test.
The dimension reduction step (PCA) has been fit on the training data where different numbers of dimensions have been tested. By cross validation, it turned out that 2 is the most suitable number of dimensions. The output of the voting system is then analyzed for all segments according to the evaluation strategy in Section 3.2. Inspections of some non-consensual attacks are also performed by a

cyber analyst in Section 3.2.4. It is worth noting that our system does not focus on attacks classification, but on detecting their originality. As attacks withing the same class are supposed to be relatively homogeneous, attacks detected as being original are those which do not match any consensus between the used classifiers ($C_1$, $C_2$ and $C_3$).

## 3.2. Validation of the proposed model

In the following, results on DoS and Probes attack are detailed. Each type of attack has different sub-types for which originality detection results are provided for weak classifiers and the voting system. As regards metrics, it is not useful in our case to report precision or F1 score results since our model does not focus on attack detection, but on their originality. Moreover, no originality annotation is provided in the used dataset. Annotations only indicate the attack type for each segment. To validate our results, we report them, then we analyzes them with the manual validation of a data analyst, as made in the SOC today.

### 3.2.1. Originality detection for Denial of Services attacks

DoS attacks are the most represented in the NSL-KDD dataset, with 45909 entries divided into five attack's type : back, neptune, pod, smurf and teardrop. Table 1 reports the results obtained by the different weak classifiers, as well as the proposed voting system.

**Table 1**
Overview of DoS Attack, with the number of attack in the dataset, number of attack found by each model, number of attack reaching the consensus and number of original attacks and the percentage of original attacks by attack types

| Attack type | Back | Neptune | Pod | Smurf | Teardrop |
|---|---|---|---|---|---|
| Dataset | 956 | 41214 | 201 | 2646 | 892 |
| K-means | 2 | 40674 | 126 | 2646 | 600 |
| KNN | 956 | 41214 | 200 | 2646 | 892 |
| 1C-SVM | 816 | 41037 | 119 | 2602 | 844 |
| Consensus | 2 | 40588 | 78 | 2602 | 582 |
| Original | 954 | 626 | 123 | 44 | 310 |
| % originals | 99,79 | 1,52 | 61,19 | 1,66 | 34,75 |

At first sight, we suspected that K-means method might be inefficient, as it detected too few back attacks. This suspicion was alimented by the distribution of data in each cluster, shown in figure 5. Since the distribution of segments within clusters did not align with attack types, we initially thought this mismatch affects the detection of back attacks. But the result obtained" by the anomalies' detection models show up a majority of anomalies in back's attacks, made us conclude the problem is on the constitution of the attack types, and not in the model.

We can see in figure 5 that neptune attacks are present in clusters 0, 1 and 4, while teardrop attacks are in cluster 3 together with smurf and pod ones. Back attacks are in cluster 2. The *K-means* model here is efficient for classified attacks that are predominantly present in dataset, in contrast to weakly represented ones.

As shown in Table 1, we can see that the voting system works well for neptune attacks, which is expected because they are massively represented in the database. Back attacks, as are not classified by *K-means*, are the most poorly detected by the voting system, as they only achieve consensus in 1% of cases. Pod and Teardrop, which are better defined, still have a high number of original attacks, with an average of 50% of segment detected as original attacks.

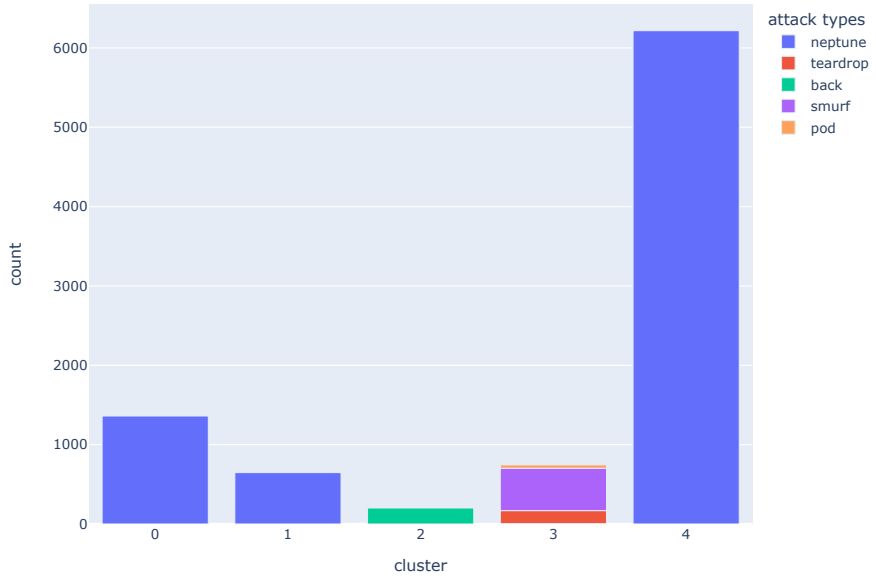


**Figure 5:** Attack distribution by cluster in *K-means* model, DoS attacks.

The voting system allows us to take advantage of the three models, and outlines attacks for which no consensus is reached by the voting system.

### 3.2.2. Originality detection for Probe attacks

Probe attacks are less numerous than DoS ones, with only four attack's types (ipsweep, nmap, portsweep and satan) listed in 11656 attacks. We can see the distribution of the data on Table 2.

**Table 2**
Overview of Probe Attack, with the number of attack in the dataset, number of attack found by each model, number of attack reaching the consensus and number of original attacks.

| Attack types | Ipsweep | Nmap | Portsweep | Satan |
|---|---|---|---|---|
| Dataset | 3599 | 1493 | 2931 | 3633 |
| K-means | 3599 | 1201 | 2917 | 3546 |
| KNN | 3599 | 1493 | 2931 | 3633 |
| 1C-SVM | 3569 | 1452 | 2909 | 3589 |
| Consensus | 3569 | 1197 | 2902 | 3512 |
| Original | 30 | 296 | 29 | 121 |
| % of originals | 0,83 | 19,83 | 0,99 | 3,33 |

When we analyze results in table 2, *K-means* model correctly identifies more than 80% of attacks, and 100% of the ipsweep attacks, based on the number of attacks found by type, on the number of attacks presents in the dataset. Both models demonstrate good performance, achieving a classification accuracy, with a good classification rate between 96% and 100%. Around 5% of the attacks demonstrate the consensus, and the segments show sufficient variation to be considered as originals.

### 3.2.3. Comparison with state of the art methods

In order to validate the proposed model, two state of the art methods among those often adopted by

experts are used: *Local Outlier factor* (LoF) and *Isolation Forest* (IF). These methods are usually used as anomaly detectors, and in our case, they are applied to attack data to outline anomalies that could be considered as a form of originality. These methods are adopted by expert thanks to their efficiency in anomalies detection, as shown by the authors, in [10] and [11], who demonstrate respectively the efficiency of IF for anomalies detection and cyber-attacks prediction. In [12] and [13], authors used LOF for anomalies detection and network intrusion detection, respectively.

For the DoS attacks, we obtained the distribution shown in table 3

**Table 3**
Overview of DoS Attack, with the number of attack in the dataset, number of original attack found by each model and percentage of original attacks.

| Attack type | Back | Neptune | Pod | Smurf | Teardrop |
|---|---|---|---|---|---|
| Dataset | 956 | 41214 | 201 | 2646 | 892 |
| LoF | 160 | 1841 | 33 | 182 | 80 |
| %Original LoF | 16,74 | 4,47 | 16,42 | 6,88 | 8,97 |
| IF | 948 | 1348 | 134 | 1738 | 423 |
| %Original IF | 99,16 | 3,27 | 66,67 | 65,68 | 47,42 |

We see that LoF and IF are not consensual: as LoF seems to be more precise than IF. However, if we obtained better results than IF for all the attack types, the original attacks detected for back, pod and teardrop by LoF are less than ours. But our three classifiers share similar results, which confirm the efficiency of the model. However, back attacks seem to be less prone to original attacks. As our system detect a majority of original attacks (99%), it is interesting to see that IF models still detects 99,16% of original attacks, making think that back attacks are difficult to classifies.
We use the same method for Probes attacks, and the results are presented on table 4

**Table 4**
Overview of Probes Attack, with the number of attack in the dataset, number of original attack found by each model and percentage of original attacks

| Attack type | ipsweep | nmap | portsweep | satan |
|---|---|---|---|---|
| Dataset | 3599 | 1493 | 2931 | 3633 |
| LoF | 128 | 29 | 222 | 204 |
| %Original LoF | 3,56 | 1,94 | 7,57 | 5,62 |
| IF | 12 | 259 | 203 | 592 |
| %Original IF | 3,11 | 17,35 | 6,93 | 16,30 |

The state of the art methods seems more efficient for Probes attacks but exception made for nmap attacks, our methods seem more precise.
After analysis, we have 80% of segments in common and we choose to examine manually the 20% to confirm if they are originals, and the results are presented in the section bellow.

### 3.2.4. Validation by a cyber analyst

We investigate whether attacks classified as original by our model and normal by the competing methods are indeed original from the expert point of view. We therefore define this three-steps procedure:

1. We first analyze features of normal attacks. Then we make statistics to evaluate nominal values of those features.
2. We compare features of each attack segment to the previously identified nominal values.
3. We use the tuple notions and duplicated to analyze, as we know the value range for each descriptor. We considered an attack as original only if its features are unique.

To check our results, we analyze segment, without knowing their classification by our system, and then, we compare if the manual analysis and the output of the system matches (original only detected by our model, original detected by both model, and original only detected by state-of-the-art model).

1. For 45 original attack only detected by our model, we actually confirm after analysis 32 originals attacks,
2. For 55 original attack detected by our model and the state-of-the-art models, we confirm after analysis 44 originals attacks,
3. For 100 original attack detected by state-of-the-art model, we confirm after analysis 88 originals attack

To illustrate our analysis, we choose segment number 81 557, a neptune attack, as example: our attention was drawn by three variations on the descriptors:

1. Duration: The length of time duration of the connection is about 2 seconds, when a duration for a normal neptune attacks is 0 seconds.
2. Flag: The status of the connection is SF (Normal establishment and termination), when normal neptune attacks have a flag S0 (connection attempt seen but no reply).
3. Src_bytes: The number of data bytes transferred from source to destination in single connection is 10714 bytes when a normal neptune attacks is 0.

For probes attacks, we can mainly cite variation on the service (Destination network service used) descriptor, like supdup, which appears on the segment number 41 594 (attack nmap), when normal nmap attacks never use it.
We also examine segments detected by the state of the art methods as original, in contrast to our model. The traditional method classifies it as original, when it is just a false positive. For segments number 253 and 125876 which do not present any variation of the patterns, it is inexplicable why they have been classified so. Finally, segment number 125694 is classified as original, when is rejected (Flag REJ) : it value are different from the normal ones, because the attack never succeed.

This analysis, carried out on a representative number of samples, confirms that our system is performing well, as 76% of attacks detected were effectively original : our system can complement existing methods, contributing in the long term to reducing the number of alerts raised by SOCs requiring human analysis.

## 4. Conclusion and future work

In this paper, we proposed a method for attack originality detection. The proposed method is based on the collaboration of different weak classifiers with a voting system which is responsible for detecting non-consensual attacks identified as original. The obtained results are promising, making evidence of situations where classical techniques fail to detect attack originality. Conclusions have also confirmed by a CS Group cyber analyst. Results confirm that although competing techniques may detect more abnormal attacks, they fail to detect others that have been confirmed as original by the expert, confirming the specificity of our method, and the complement it offer to state-of-the-art method.
This method is a first step to improve SIEM capabilities of early stage APT detection. Future work will focus on adding state-of-the-art classifiers to strengthen our voting system, and then investigating deep learning methods for attack originality detection.

## Acknowledgments

The authors would like to thank KOUAMEN Josee Alvine, analyst of the BU Cyber of CS group, for the results validation, and the Data & Process Intelligence Team of CS-Group for their precious help, particularly MENIODEM DELIOTA Keline.

## Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

## References


[1] D. S. Berman, A. L. Buczak, J. S. Chavis, C. L. Corbett, A Survey of Deep Learning Methods for Cyber Security, https://www.mdpi.com/2078-2489/10/4/122, 2019.

[2] W. A. Yousef, I. Traoré, W. Briguglio, Classifier Calibration: With Application to Threat Scores in Cybersecurity, IEEE Transactions on Dependable and Secure Computing 20 (2023) 1994–2010. doi:10.1109/TDSC.2022.3170011.

[3] O. L. Soh, Advanced Persistent Threat Detection Using Anomaly Score Calibration and Multi-class Classification, Master's thesis, University of Victoria, 2023.

[4] S. Haribalaji, P. Ranjana, Distributed Denial of Service (DDOS) Attack Detection Using Classification Algorithm, in: 2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS), 2024, pp. 1–6. doi:10.1109/ADICS58448.2024.10533510.

[5] J. Meira, R. Andrade, I. Praça, J. Carneiro, G. Marreiros, Comparative Results with Unsupervised Techniques in Cyber Attack Novelty Detection, in: P. Novais, J. J. Jung, G. Villarrubia González, A. Fernández-Caballero, E. Navarro, P. González, D. Carneiro, A. Pinto, A. T. Campbell, D. Durães (Eds.), Ambient Intelligence – Software and Applications –, 9th International Symposium on Ambient Intelligence, Springer International Publishing, Cham, 2019, pp. 103–112. doi:10.1007/978-3-030-01746-0_12.

[6] M. Tavallaee, E. Bagheri, W. Lu, A. A. Ghorbani, A detailed analysis of the KDD CUP 99 data set, in: 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, 2009, pp. 1–6. doi:10.1109/CISDA.2009.5356528.

[7] I. Trrad, Applying Deep Learning Techniques for Network Traffic Classification: A Comparison Study on the NSL-KDD Dataset, 2024. doi:10.21203/rs.3.rs-3869444/v1.

[8] A. D. Vibhute, C. H. Patil, A. V. Mane, K. V. Kale, Towards Detection of Network Anomalies using Machine Learning Algorithms on the NSL-KDD Benchmark Datasets, Procedia Computer Science 233 (2024) 960–969. doi:10.1016/j.procs.2024.03.285.

[9] D. Protic, Review of KDD Cup '99, NSL-KDD and Kyoto 2006+ datasets, Vojnotehnicki glasnik 66 (2018) 580–596. doi:10.5937/vojtehg66-16670.

[10] R. Sharma, M. Grover, Enhancing Cybersecurity with Machine Learning: Evaluating the Efficacy of Isolation Forests and Autoencoders in Anomaly Detection, in: 2024 7th International Conference on Circuit Power and Computing Technologies (ICCPCT), volume 1, 2024, pp. 1017–1021. doi:10.1109/ICCPCT61902.2024.10673338.

[11] R. C. Ripan, M. M. Islam, H. Alqahtani, I. H. Sarker, Effectively predicting cyber-attacks through isolation forest learning-based outlier detection, SECURITY AND PRIVACY 5 (2022) e212. doi:10.1002/spy2.212.

[12] N. J. Johannesen, M. L. Kolhe, M. Goodwin, Vertical Approach Anomaly Detection Using Local Outlier Factor, in: H. Haes Alhelou, N. Hatziargyriou, Z. Y. Dong (Eds.), Power Systems Cybersecurity: Methods, Concepts, and Best Practices, Springer International Publishing, Cham, 2023, pp. 297–310. doi:10.1007/978-3-031-20360-2_12.

[13] A. R. Vasudevan, S. Selvakumar, Local outlier factor and stronger one class classifier based hierarchical model for detection of attacks in network intrusion detection dataset, Frontiers of Computer Science 10 (2016) 755–766. doi:10.1007/s11704-015-5116-8.