Detection of Data Poisoning Attacks On NSL-KDD Using Isolation Forest And Random Forest

Aqsa Ghaffar
Department of Information and Communication Engineering
Islamia University of Bahawalpur
Bahawalpur, Pakistan
aqsaghaffar908@gmail.com

Sidra Ghaffar
Department of Information and Communication Engineering
Islamia University of Bahawalpur
Bahawalpur, Pakistan
sidraghaffar774@gmail.com

Sana Tariq
Department of Information and Communication Engineering
Islamia University of Bahawalpur
Bahawalpur, Pakistan
sana.tariq@iub.edu.pk

Muhammad Aoun
Department of Information and Communication Engineering
Islamia University of Bahawalpur
Bahawalpur, Pakistan
aoun.muhammad@iub.edu.pk

***Abstract*—Data poisoning attacks are a serious threat to machine learning models used in cybersecurity. In this paper, we performed three types of data poisoning attacks — Label Flipping, Targeted Poisoning, and Clean Label Boundary Attack — on the NSL-KDD cybersecurity dataset at two poisoning rates of 20% and 50%. We used the Isolation Forest algorithm to detect and remove poisoned samples and then retrained a Random Forest classifier to recover model accuracy. Our results show that at 20% poisoning, Clean Label Boundary Attack caused the biggest accuracy drop of 5.92%, reducing accuracy from 91.87% to 85.95%. At 50% poisoning, Label Flipping caused the most severe degradation with accuracy dropping by 19.00% from 91.33% to 72.33%. Isolation Forest provided meaningful but incomplete recovery across all attacks, performing best against Targeted Poisoning at 50% where accuracy was restored to 91.36%. These findings show that cybersecurity datasets are vulnerable to data poisoning attacks and stronger defense methods are needed to fully protect intrusion detection systems.**

***Index Terms*—Data poisoning, NSL-KDD, Isolation Forest, Random Forest, Intrusion Detection, Label Flipping, Cybersecurity**

I. Introduction

Data poisoning attacks are one of the most serious threats to machine learning systems [1], AI [2]. These attacks work by injecting fake or manipulated data into the training dataset. As a result, the model learns incorrect patterns and starts making wrong predictions. Today, machine learning is widely used in cybersecurity to detect network attacks. Intrusion detection systems (IDS) are trained on datasets like NSL-KDD to identify harmful network traffic, including probe, DoS, U2R, and R2L attacks. But if the training data are corrupted, the whole system can fail. As a result of data poisoning attacks, the integrity, reliability, and accuracy of the model are degraded.

There are several forms of data poisoning attacks including clean-label attacks in which labels remain the same but input data is altered slightly, as well as backdoor attacks and data injecting attacks, and label flipping attacks, where the label of the actual data is altered. Even low-level stealthy attacks like label flipping have been shown to decrease the accuracy of the model. For instance, Anthropic found that just 250 fake documents are enough to poison a model. Surprisingly, it does not matter how big or small the model is. A very large model and a tiny model can both be fooled by the same small amount of bad data [3]. Furthermore, a recent article demonstrates that in earlier years, it was considered that data poisoning mainly occurred at training-time. However, real incidents show it can strike anywhere in the lifecycle, and it does not always come from outside hackers. Internal adversaries with data access are also as dangerous as attackers adding malicious data in model to reduce its efficiency [4].

A recent study shows that machine learning models are also dealing with data poisoning attacks. In [5], a researcher named Choeun from Aalborg University did a study that shows how data is poisoned. He used a dataset called CIFAR-10 which is used for image classification. He found that both visible and invisible triggers can mess up ML programs and they take wrong decisions. His findings show that visible triggers are easy to use and very effective.

Additionally, in [6] Kure et al. wrote their work on detecting and preventing AI models from data poisoning attacks. They used CIFAR-10 dataset and performed label poisoning attack by changing the label of images. In their experiment they flipped the label of dog with cat; as a result of data poisoning attack the accuracy of model degraded by 27% proving that AI models are vulnerable to adversarial data attacks. Even unintentional errors can reduce the accuracy of model resulting in wrong predictions which we cannot afford in sensitive sectors.

Verde et al. [7] explore the effects of data poisoning attacks on the reliability of machine learning models, and how noise can be used to undermine performance. They examined the impacts that noise in the data may have on machine learning programs. They took a sample of voice

DOI: https://doi.org/10.67535/tsp.000003.002
Published under the ICACNC 2026 Proceedings

recordings and introduced noise to it. They discovered that even an insignificant amount of noise can cause the machine learning program to be less accurate. In [8], Paudice et al. discuss the issue of label flipping attacks, a variant of data poisoning where attackers modify the labels of training data to deceive machine learning models. They present an effective algorithm to perform optimal label flipping attacks, and also propose a defense mechanism which identifies and rectifies mislabeled data points, thus reducing the impact of the attack. Nevertheless, the efficiency of the approach against more advanced poisoning schemes and its scaling in large-scale applications are not fully discussed.

Data poisoning attacks have been identified as a serious threat across multiple critical domains. In healthcare, E.G. Acuña Acuña [9] demonstrated that corrupted training data in medical AI systems can lead to life-threatening misdiagnoses, highlighting the urgent need for robust defense mechanisms.

NSL-KDD is a cybersecurity dataset with 41 features which make it both complex and high dimensional. Poisoned samples are not easy to detect because some poisoned traffic looks similar to normal traffic. This is a perfect and challenging dataset to test data poisoning detection. In [10], authors highlighted the challenges of datasets. They used two datasets KDD99 and UNSW-NB15 which are cybersecurity datasets. The dataset had an imbalance problem. They solved it by using SMOTE, Robust Scaler and Quantile Transformer techniques. Most research on data poisoning has focused on image datasets and text models. Less attention has been given to cybersecurity datasets like NSL-KDD, which are the backbone of intrusion detection systems. On top of that, many proposed defenses only work when the defender already knows what kind of attack took place — an assumption that rarely holds in reality. This paper takes a different approach. We test three types of poisoning attacks on NSL-KDD and defend against them using Isolation Forest, which does not require any knowledge of the attack in advance.

The main contributions of this paper are as follows:

- We applied three types of data poisoning attacks — Label Flipping, Targeted Poisoning, and Clean Label Boundary Attack — on the NSL-KDD dataset at two poisoning rates of 20% and 50%, evaluating their impact using Accuracy, F1-Score, Precision, and Recall.
- We used Isolation Forest algorithm to detect and remove poisoned samples from training data and used Random Forest classifier to recover model accuracy.
- At 20% poisoning, Clean Label Boundary Attack caused the largest drop from 91.87% to 85.95%, Label Flipping dropped from 91.31% to 86.93%, and Targeted Poisoning from 91.47% to 88.31%. At 50% poisoning, Label Flipping caused the most severe drop from 91.33% to 72.33%, Clean Label Boundary from 91.40% to 79.48%, and Targeted Poisoning from 83.85% to 80.48%. Recovery accuracies at 20% were 88.50%, 90.40%, and 88.89% and at 50% were 81.17%, 91.36%, and 85.50% for Label Flipping, Targeted Poisoning, and Clean Label Boundary respectively.

II. Related Work

Data poisoning attacks have been studied by many researchers in the past and they have investigated the effect of poisoning attacks on ML models trained on cybersecurity datasets.

A. Detection Strategies for Poisoning Attacks

In [11], Topological Data Analysis (TDA) was used with three algorithms to detect poisoned samples in network intrusion datasets before training. It was tested on UNSW-NB15 and CICIDS2017 datasets and was computationally expensive, limiting its practical use. In contrast, our paper uses Isolation Forest on NSL-KDD dataset which is simpler and more efficient.

In [6], Kure et al. wrote their work on detecting and preventing AI models from data poisoning attacks. They used CIFAR-10 dataset and Insurance claim dataset. Their framework was more complex, causing challenges for real world implementation.

In [12], Mari et al. developed a machine learning intrusion detection system using NSL-KDD dataset and tested it against adversarial traffic generated by GAN. They improved performance using adversarial training but did not propose a clear recovery method after poisoning.

In [13], they used feature selection to reduce computational complexity and enhance model accuracy. A total of 111,386 instances were trained and 37,129 instances were tested from the NSL-KDD dataset. Their paper neglects the concept of data poisoning. All eight algorithms used were supervised and require labeled data to work. They only built a detection system and did not propose any method for recovery after poisoning.

In [14], they analyze NSL-KDD for intrusion detection and used CRISP-DM methodology. NSL-KDD has imbalanced classes but they did not highlight this problem. They also focused on detection but did not give any idea of recovery. Alsaban and Fkih [15] proposed a stacking ensemble framework combining KNN, RF, DT, and LR classifiers to detect data poisoning attacks across four benchmark datasets. In [16]–[18] different detection strategies are studied.

B. Defense Strategies Against Data Poisoning Attacks

In [19], they proposed a defense strategy. They checked parameter updates from each participant. If a participant sent unusual updates they marked them as malicious participants. Their findings show that even a minor presence of malicious participants will decrease the classification accuracy. This defense system is only limited to insider attacks and cannot detect external data poisoning attacks like label flipping on dataset.

In [9], Acuña Acuña also proposed a multi-layer defense mechanism for data poisoning attack.

In [20], this paper proposes a defense method against multi-label flipping attacks in federated learning. In [21], this paper proposed FLORAL, an adversarial defense strategy, and used SVM as base algorithm. Their bi-level optimization requires

prior knowledge which is not possible in real world problems. In [22], [23] defense strategies are discussed.

Recent studies show that even poisoning as little as 0.001% of training data can degrade model accuracy by up to 30%. In [24] they reviewed all major data poisoning research from 2018–2025 and analyzed its impact in healthcare, finance, autonomous systems and AI. Defense strategies involve adversarial training, knowledge graph verification and data sanitization but none of them fully solve the problem.

However, none of these defense methods were specifically tested on NSL-KDD cybersecurity dataset using unsupervised detection, which this paper addresses.

III. Methodology

NSL-KDD was selected as it is a widely used cybersecurity benchmark containing real-world network traffic from four attack categories: DoS, Probe, User, and Lease. Features were encoded, scaled to [0,1] using Min-Max scaling, and labels simplified to normal (0) and attack (1). Official train-test splits were used unchanged. Figure 1 provides a pictorial representation of how we went about the entire process.

Isolation Forest was selected for its unsupervised nature and efficiency on high-dimensional data, requiring no labeled samples or prior knowledge of attack type.

A. Baseline Model

The number of trees used in the Random Forest is 100 on clean data to get a baseline accuracy. This baseline was used as a reference point to measure the extent of the damage caused by each attack on the model. The overall accuracy of the baseline was 91.31%, testing the model with clean data to check the model's performance.

B. Data Poisoning Attacks

Three attacks were applied at 20% and 50% poisoning rates. Label Flipping randomly changes correct labels to wrong ones. Targeted Poisoning swaps normal labels to attack and attack labels to normal simultaneously, corrupting the model in both directions. The Clean Label Boundary Attack adds small Gaussian noise to feature values without changing any labels, pushing samples toward the decision boundary. This makes poisoned samples look legitimate and hard to detect.

C. Anomaly Detection

After each attack, Isolation Forest was applied to find and remove suspicious samples. It works by isolating each sample using random trees; samples that are isolated quickly are considered anomalies. Flagged samples were removed before retraining.

D. Recovery and Evaluation

The model was retrained on the cleaned data and tested on the same test set. We recorded four metrics for every experiment: clean accuracy, poisoned accuracy, recovered accuracy, and weighted F1-Score to account for class imbalance.

Fig. 1. Experimental Workflow for Evaluating Data Poisoning Attacks and Recovery in an Intrusion Detection System

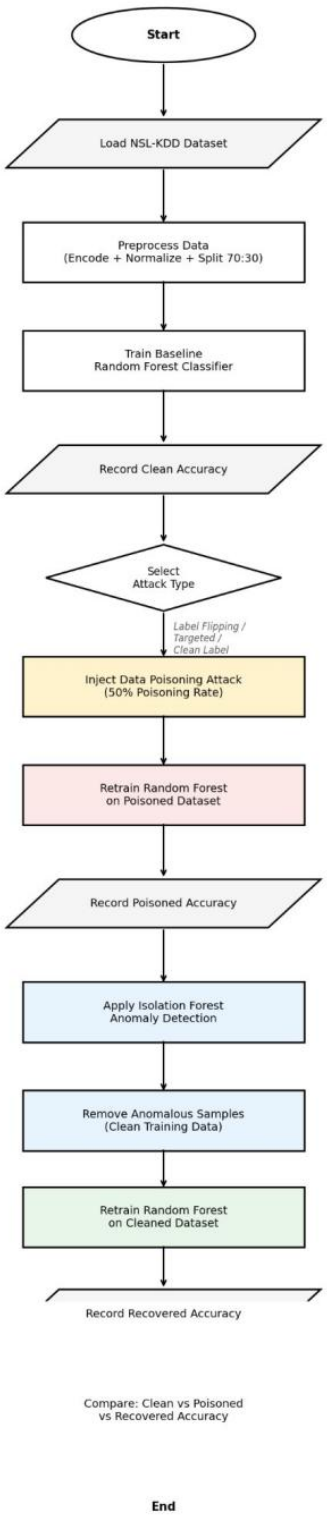

Fig. 1. Experimental Pipeline

E. Experimental Setup

All experiments were run on Google Colab in Python 3.x. The Random Forest classifier and Isolation Forest were both built using scikit-learn. Data was loaded and manipulated using Pandas and NumPy, and all visualizations were created using Matplotlib. The Random Forest was set up with the following parameters: 100 trees with Gini impurity for node splitting. A fixed random seed of 42 was used so that the results are fully reproducible. The contamination parameter was fixed at 0.1 as a conservative estimate, since defenders cannot know the true poisoning extent in advance. Testing at 20% and 50% poisoning rates simulates real-world scenarios where the attacker exceeds this assumption.

IV. Results and Discussion

A. Baseline Accuracy

Prior to initiating any attack, the Random Forest model was trained on clean data and achieved a baseline accuracy of approximately 91.31%, with an F1-Score of 0.9131, Precision of 0.9135, and Recall of 0.9131. This confirms the model works well on clean NSL-KDD data and serves as the reference point for measuring the impact of each poisoning attack. Slight variations in clean baseline accuracy (91.31%–91.87%) are due to differences in the random training subsets modified by each attack. All experiments used a fixed seed of 42 and 70/30 split for reproducibility.

B. Results at 20% Poisoning Rate

1) Label Flipping Attack (20%): The accuracy of Label Flipping was reduced from 91.31% to 86.93% at 20% poisoning rate, which is a decrease of 4.38 percentage points. The F1-Score, Precision and Recall of the poisoned model were 0.8690, 0.8695 and 0.8693 respectively. This was a case where 20% of the training labels were corrupted, but the performance of the model was still good because of the use of 100 trees in Random Forest that made the corrupted samples' damage spread in the model. After the Isolation Forest cleaning, accuracy was attained at 88.50% with an F1-Score of 0.8847, Precision of 0.8853 and Recall of 0.8850, which resulted in an improvement of 1.57 percentage points, but the accuracy did not match the clean baseline again.

2) Targeted Poisoning Attack (20%): The accuracy of Targeted Poisoning at 20% decreased 3.16 percentage points from 91.47% to 88.31%. The F1-Score of poisoned model was 0.8828, the Precision of poisoned model was 0.8835 and the Recall of poisoned model was 0.8831. Although this attack is small (20%), it is still damaging as it corrupts labels in both directions. The accuracy after cleaning using Isolation Forest is 90.40%, with F1-Score 0.9038, Precision 0.9043 and Recall 0.9040, representing a 2.09 percentage point improvement and almost a 100% restoration to the clean baseline accuracy.

3) Clean Label Boundary Attack (20%): Clean Label Boundary Attack results in the highest drop with an accuracy drop from 91.87% to 85.95% (-5.92%). Poisoned model: F1-Score 0.8592, Precision 0.8598, Recall 0.8595. The attack is a very stealthy attack as values of the features are not changed by the attack, only noise is added. Remarkably, the accuracy rose to 88.89% (F1-Score 0.8886, Precision 0.8892 and Recall 0.8889) after applying the Isolation Forest cleaning algorithm, which improved by 2.94 percentage points. Its lower recovery gain confirms its stealthy nature, as unchanged labels make poisoned samples hard to detect.

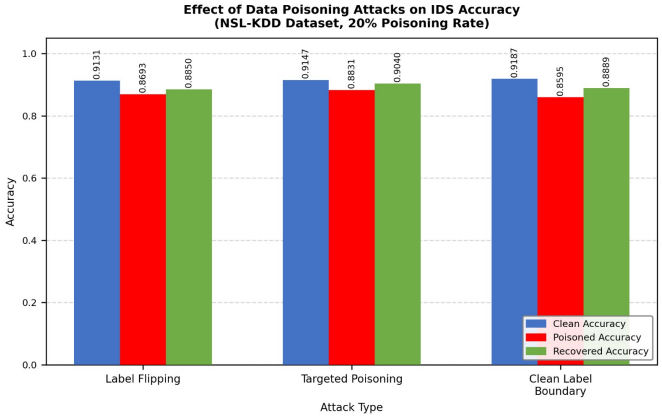

Fig. 2. Effect of Data Poisoning Attacks on IDS Accuracy at 20% Poisoning Rate

C. Results at 50% Poisoning Rate

1) Label Flipping Attack (50%): The most severe degradation among all the studies in the overall study was due to Label Flipping with accuracy performance being degraded from 91.33% to 72.33% at 50% poisoning. The F1-Score of the model poisoned is 0.7230, the Precision of the poisoned model is 0.7236 and the Recall of the poisoned model is 0.7233. 50% of training labels were corrupted and the model's ability to classify the traffic as a normal one or an attack was lost. The result of the Isolation Forest cleaning gave a slight improvement of 8.84 percentage points in accuracy to 81.17% and an F1-Score of 0.8114, Precision of 0.8120 and Recall of 0.8117, which was not near the clean baseline.

2) Targeted Poisoning Attack (50%): Targeted Poisoning at 50% led to an accuracy drop from 83.85% to 80.48%. For the poisoned model the F1-Score, Precision and Recall were 0.8045, 0.8051 and 0.8048, respectively. Although the accuracy loss appears small, it is a bidirectional corruption thereby making it a very dangerous attack. The result showed that, after removing the samples by using the Isolation Forest algorithm, the accuracy increased to 91.36% while the F1-Score, Precision and Recall of the result was 0.9133, 0.9138 and 0.9136 respectively, which is better than the poisoned scenario, whose result was baseline 83.85%, indicating that the algorithm of Isolation Forest is quite effective in detecting and removing the samples in this attack.

3) Clean Label Boundary Attack (50%): The accuracy of Clean Label Boundary Attack at 50% went down by 11.92 percentage points from 91.40% to 79.48%. Poisoned model got F1-Score as 0.7945, Precision as 0.7951 and Recall as 0.7948 as shown in figure 3. This is a significant drop as compared

to an increase of 20% in poisoned samples that leads to more distortion of the decision boundary. After the Isolation Forest cleaning, the accuracy of the model restored to 85.50% with F1-Score of 0.8547, Precision of 0.8553 and Recall of 0.8550 with an improvement of 6.02 percentage points. The lower recovery gain of 6.02 points compared to Label Flipping (8.84 points) confirms its stealthy nature, as unchanged labels make poisoned samples statistically similar to clean data and hard to detect.

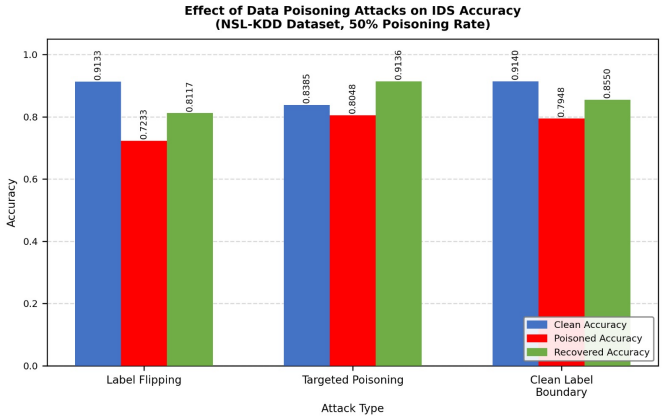

Fig. 3. Effect of Data Poisoning Attacks on IDS Accuracy at 50% Poisoning Rate

D. Comparative Analysis Across Poisoning Rates

Table I presents a full summary of all results across both poisoning rates.

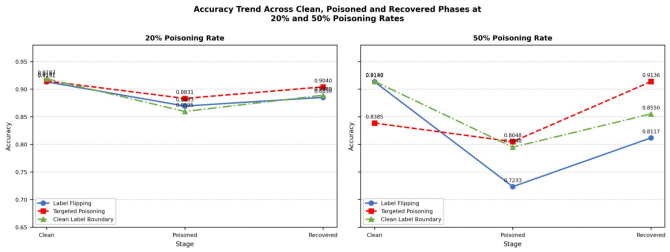

Fig. 4. Accuracy Degradation by Attack Type and Poisoning Rate

E. Summary of Experiment

Four key observations can be drawn from the experimental results. First, the damage caused by Label Flipping grows with the poisoning rate, and Isolation Forest provides only partial recovery at high rates, restoring accuracy to 81.17% against a clean baseline of 91.33% at 50% poisoning. Second, although Targeted Poisoning at 50% reduced accuracy from 83.85% to 80.48%, Isolation Forest recovered it to 91.36%, actually exceeding the poisoned baseline. This is because bidirectional label corruption creates statistically unusual samples that Isolation Forest is designed to detect. Third, Clean Label Boundary Attack was the hardest to defend against, as it modifies only feature values while keeping labels intact, making poisoned

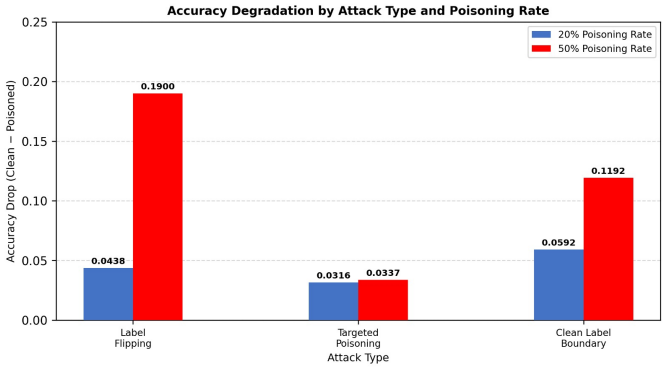

Fig. 5. Isolation Forest Recovery Gain by Attack Type and Poisoning Rate

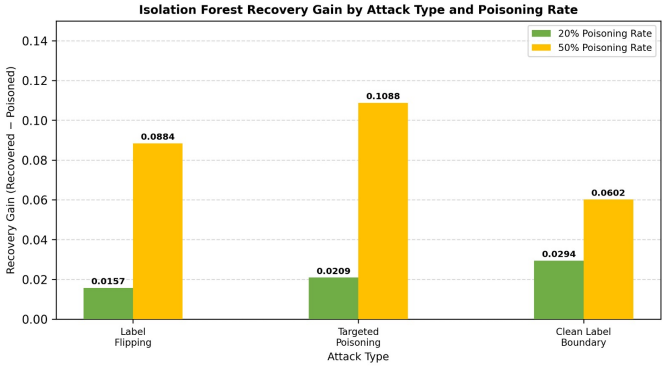

Fig. 6. Accuracy Trend Across Clean, Poisoned and Recovered Phases at Both Poisoning Rates

samples difficult to distinguish from clean data. Fourth, Isolation Forest is effective against simple low-rate attacks but loses effectiveness as attack complexity and poisoning rate increase.

V. Conclusion

This research paper aims to investigate the impact of three data poisoning attacks Label Flipping, Targeted Poisoning, and Clean Label Boundary Attack on a Random Forest based intrusion detection system trained on the NSL-KDD dataset at poisoning rates of 20% and 50%. Isolation Forest was used to detect and remove poisoned samples before retraining the model to measure recovery. NSL-KDD was selected because it is a widely used benchmark in cybersecurity research containing real network traffic with four attack categories. It eliminates redundant records from KDD Cup 1999, making results reliable and comparable with previous work. The baseline model achieved 91.31% accuracy, F1-Score of 0.9131, Precision of 0.9135, and Recall of 0.9131. At 20% poisoning, Clean Label Boundary Attack caused the biggest accuracy drop of 5.92% from 91.87% to 85.95%. At 50% poisoning, Label Flipping caused the most severe degradation dropping accuracy by 19.00% from 91.33% to 72.33%. Targeted Poisoning at 50% showed the best recovery, with Isolation Forest restoring accuracy to 91.36%, even exceeding the poisoned baseline. Clean Label Boundary Attack is the most stealthy

TABLE I
FULL SUMMARY OF RESULTS: ACCURACY, F1-SCORE, PRECISION, AND RECALL

Rate	Attack	Cl Acc	Po Acc	Re Acc	Cl F1	Po F1	Re F1	Cl Prec	Po Prec	Re Prec	Cl Rec	Po Rec	Re Rec
20%	Label Flipping	0.9131	0.8693	0.8850	0.9131	0.8690	0.8847	0.9135	0.8695	0.8853	0.9131	0.8693	0.8850
20%	Targeted Poisoning	0.9147	0.8831	0.9040	0.9147	0.8828	0.9038	0.9150	0.8835	0.9043	0.9147	0.8831	0.9040
20%	Clean Label Boundary	0.9187	0.8595	0.8889	0.9187	0.8592	0.8886	0.9190	0.8598	0.8892	0.9187	0.8595	0.8889
50%	Label Flipping	0.9133	0.7233	0.8117	0.9133	0.7230	0.8114	0.9136	0.7236	0.8120	0.9133	0.7233	0.8117
50%	Targeted Poisoning	0.8385	0.8048	0.9136	0.8385	0.8045	0.9133	0.8388	0.8051	0.9138	0.8385	0.8048	0.9136
50%	Clean Label Boundary	0.9140	0.7948	0.8550	0.9140	0.7945	0.8547	0.9143	0.7951	0.8553	0.9140	0.7948	0.8550

attack since it never modifies labels, making it the hardest to defend against using unsupervised methods.

There are a few limitations to this study. Isolation Forest failed to achieve high-accuracy models at high levels of poisoning, and the unbalanced class distribution in NSL-KDD had a negative effect on the baseline performance. In future work, we plan to test our approach on CICIDS-2017 and UNSW-NB15, and explore stronger defense methods and combinations of multiple detection strategies to improve recovery against advanced poisoning attacks.

REFERENCES

[1] X. Wang, F. Wang, Y. Hong, and X. J. Ban, "Transferability in data poisoning attacks on spatiotemporal traffic forecasting models," *Transportation Research Part C: Emerging Technologies*, vol. 183, p. 105501, 2026.

[2] F. Abtahi, F. Seoane, I. Pau, and M. Vega-Barbas, "Data poisoning vulnerabilities across health care artificial intelligence architectures," *Journal of Medical Internet Research*, vol. 28, p. e87969, 2026.

[3] A. Souly, J. Rando, E. Chapman, X. Davies, B. Hasircioglu, E. Shereen, C. Mougan, V. Mavroudis, E. Jones, C. Hicks, N. Carlini, Y. Gal, and R. Kirk, "A small number of samples can poison LLMs of any size," *arXiv preprint arXiv:2510.07192*, 2025. [Online]. Available: https://arxiv.org/abs/2510.07192

[4] Lakera Team, "Introduction to data poisoning: A 2026 perspective," Lakera Blog, 2026. [Online]. Available: https://www.lakera.ai/blog/training-data-poisoning

[5] C. Song, "Data poisoning attacks in machine learning: Risks and defenses," 2025.

[6] H. I. Kure, P. Sarkar, A. B. Ndanusa, and A. O. Nwajana, "Detecting and preventing data poisoning attacks on ai models," in *2025 Photonics & Electromagnetics Research Symposium (PIERS-Spring)*. IEEE, 2025, pp. 01–12.

[7] L. Verde, F. Marulli, and S. Marrone, "Exploring the impact of data poisoning attacks on machine learning model reliability," *Procedia Computer Science*, vol. 192, pp. 2624–2632, 2021.

[8] A. Paudice, L. Muñoz-González, and E. C. Lupu, "Label sanitization against label flipping poisoning attacks," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2018, pp. 5–15.

[9] E. G. A. Acuña, "Healthcare cybersecurity: Data poisoning in the age of ai," *Journal of Comprehensive Business Administration Research*, 2024.

[10] A. Rehman, T. Saba, M. M. Jamjoom, S. Al-Otaibi, and M. I. Khan, "Advances in machine learning for explainable intrusion detection using imbalance datasets in cybersecurity with harris hawks optimization," *Computers, Materials & Continua*, vol. 86, no. 1, pp. 1–15, 2026.

[11] G. F. Monkam, M. J. De Lucia, and N. D. Bastian, "A topological data analysis approach for detecting data poisoning attacks against machine learning based network intrusion detection systems," *Computers & Security*, vol. 144, p. 103929, 2024.

[12] A.-G. Mari, D. Zinca, and V. Dobrota, "Development of a machine-learning intrusion detection system and testing of its performance using a generative adversarial network," *Sensors*, vol. 23, no. 3, p. 1315, 2023.

[13] I. Abrar, Z. Ayub, F. Masoodi, and A. M. Bamhdi, "A machine learning approach for intrusion detection system on NSL-KDD dataset," in *Proceedings of the International Conference on Smart Electronics and Communication (ICOSEC)*. IEEE, 2020, pp. 919–924.

[14] Y. Yuliana, D. H. Supriyadi, M. R. Fahlevi, and M. R. Arisagas, "Analysis of nsl-kdd for the implementation of machine learning in network intrusion detection system," *Journal of Informatics Information System Software Engineering and Applications (INISTA)*, vol. 6, no. 2, pp. 80–89, 2024.

[15] Y. B. Alsaban and F. Fkih, "A stacking ensemble framework for robust detection of data poisoning attacks," *WSEAS Transactions on Signal Processing*, vol. 22, pp. 1–11, 2026.

[16] V. Raghavan, T. Mazzuchi, and S. Sarkani, "An improved real time detection of data poisoning attacks in deep learning vision systems," *Discover Artificial Intelligence*, vol. 2, no. 1, p. 18, 2022.

[17] S. Perry, S. Perry, Y. Jiang, and C. Walter, "Detecting data poisoning attacks in image datasets using a vision-language hybrid pipeline," in *Proceedings of the 59th Hawaii International Conference on System Sciences (HICSS)*, 2026, pp. 1–10.

[18] A. Takiddin, M. Ismail, R. Atat, K. R. Davis, and E. Serpedin, "Robust graph autoencoder-based detection of false data injection attacks against data poisoning in smart grids," *IEEE Transactions on Artificial Intelligence*, vol. 5, no. 3, pp. 1287–1301, 2024.

[19] A. Khraisat, A. Alazab, M. Alazab, T. Jan, S. Singh, and M. A. Uddin, "Securing federated learning: a defense strategy against targeted data poisoning attack," *Discover Internet of Things*, vol. 5, no. 1, p. 16, 2025.

[20] W. Ma, Q. Zhao, and W. Tian, "A defense method against multi-label poisoning attacks in federated learning," *Scientific Reports*, vol. 15, no. 1, p. 26197, 2025.

[21] M. I. Bal, V. Cevher, and M. Muehlebach, "Adversarial training for defense against label poisoning attacks," 2025. [Online]. Available: https://arxiv.org/abs/2502.17121

[22] T. Clement, C. Gbaja, and H. A. Onayemi, "Adversarial machine learning: Defense mechanisms against poisoning attacks in cybersecurity models," *International Journal of Engineering and Computer Science*, 2025. [Online]. Available: https://www.ijecs.in/index.php/ijecs/article/view/5156

[23] Y.-C. Lai, J.-Y. Lin, Y.-D. Lin, R.-H. Hwang, P.-C. Lin, H.-K. Wu, and C.-K. Chen, "Two-phase defense against poisoning attacks on federated learning-based intrusion detection," *Computers & Security*, vol. 129, p. 103205, 2023.

[24] F. Hartle III, S. Mancini, and E. Kerry, "Data poisoning 2018–2025: A systematic review of risks, impacts, and mitigation challenges," *Issues in Information Systems*, vol. 25, no. 4, pp. 433–442, 2025.