Knowledgeable Research (An International Peer-Reviewed Multidisciplinary Journal) **ISSN 2583-6633**
Available Online: http://knowledgeableresearch.com Vol.04, No.12, December,2025

Adversarial Machine Learning: Detecting and Mitigating Evasion Attacks on Intrusion Detection Systems

*Dr. Prashant Kohli
Associate Professor, Department of Physics, N.M.S.N. Dass (P.G.) College, Budaun

****

Abstract: *Network Intrusion Detection Systems (NIDS) increasingly rely on machine learning (ML) to identify malicious traffic that signature-based tools cannot recognize. This dependence, however, introduces a new attack surface: adversarial evasion, in which an attacker crafts subtly perturbed network traffic that is misclassified as benign while preserving its malicious functionality. This paper presents a systematic study of evasion attacks against ML-based NIDS and proposes a layered detection-and-mitigation framework that combines adversarial training, input transformation defences, ensemble disagreement analysis, and statistical drift monitoring. We formalize the threat model across white-box, gray-box, and black-box attacker knowledge levels; implement and evaluate four representative evasion techniques (Fast Gradient Sign Method, Projected Gradient Descent, a genetic-algorithm-based black-box attack, and a GAN-based traffic synthesizer) against a Random Forest, a Multi-Layer Perceptron, and an XGBoost classifier trained on the CIC-IDS2017 and NSL-KDD datasets; and measure the effectiveness of our proposed defences. Our results show that undefended ML-based NIDS suffer detection-rate degradation of 38–71% under evasion attacks, while the proposed layered defence recovers between 61% and 84% of the lost detection accuracy with an acceptable false-positive overhead of under 3 percentage points. We conclude with a discussion of the fundamental trade-offs between robustness, accuracy, and computational cost, and outline directions for future research, including certified robustness for tabular network-flow data and adversarial robust federated intrusion detection.*

Keywords: *Adversarial machine learning, Intrusion detection systems, Evasion attacks, Anomaly detection,*

****

1.Introduction

The transition from signature-based to machine-learning-based intrusion detection has been one of the defining shifts in network defence over the past decade. Traditional signature matching struggles against zero-day exploits, polymorphic malware, and novel attack variants because it depends on previously catalogued patterns. Machine learning promised a solution: by learning statistical regularities from labelled traffic, classifiers such as random forests, gradient-boosted trees, and deep neural networks can generalize to previously unseen attacks and flag anomalous behavior without an explicit rule for every threat.

This same generalization capability, however, is precisely what adversarial machine learning exploits. Research beginning with **Szegedy et al. and Goodfellow et al.** demonstrated that neural networks trained on image data could be fooled by imperceptible pixel-level perturbations. The insight generalizes beyond images: any classifier that partitions a high-

dimensional feature space with a learned decision boundary can, in principle, be evaded by moving an input just far enough across that boundary while keeping the semantic content of the input largely unchanged. In the network security domain, this means an attacker can modify packet timing, padding, header fields, or flow-level statistics of malicious traffic so that the traffic is classified as benign by an ML-based NIDS, while the underlying attack (a port scan, a botnet command-and-control beacon, or a data exfiltration channel) continues to function as intended.

The stakes of this vulnerability are considerable. Unlike an evaded spam filter, an evaded intrusion detection system can allow an active compromise to proceed undetected for an extended period, increasing dwell time and the scope of resulting damage. Because many production NIDS are increasingly ML-driven — whether through vendor-supplied anomaly detection modules, open-source frameworks, or custom in-house models — the population of exposed systems is large and growing. Understanding how such systems fail under adversarial pressure, and how that failure can be mitigated, is therefore a pressing problem for both the security and machine learning research communities.

This paper makes four contributions. First, it provides a structured threat model for evasion attacks against ML-based NIDS that distinguishes attacker knowledge levels and constrains perturbations to functionality-preserving transformations of network traffic.

Second, it implements and benchmarks four representative evasion attacks spanning gradient-based, search-based, and generative approaches.

Third, it proposes and evaluates a layered defence framework combining adversarial training, feature-space input transformation, ensemble disagreement scoring, and statistical drift detection.

Fourth, it offers an empirical comparison of the robustness–accuracy trade-off across three widely used classifier families, together with practical deployment guidance for security teams.

2. Background and Related Work:

2.1 Machine Learning in Intrusion Detection:

Machine-learning-based NIDS generally fall into two categories: misuse-based (supervised) classifiers, which are trained on labelled examples of normal and attack traffic, and anomaly-based (often unsupervised or semi-supervised) detectors, which model the statistical profile of benign traffic and flag deviations. Supervised approaches, including

decision trees, support vector machines, random forests, gradient-boosted trees, and deep neural networks, tend to achieve high detection accuracy on held-out data from known attack families. Anomaly-based approaches, including auto encoders, one-class SVMs, and clustering-based methods, are valued for their ability to flag novel attacks but often suffer from elevated false-positive rates. Benchmark datasets such as NSL-KDD, an improved version of the older KDD Cup 1999 dataset, and CIC-IDS2017, which contains contemporary benign and attack traffic captured in a realistic network testbed, have become standard evaluation corpora for this line of research.

2.2 Adversarial Examples and Evasion Attacks:

The foundational adversarial-examples literature focused primarily on image classifiers. The Fast Gradient Sign Method (FGSM) generates a perturbation by taking a single step in the direction of the gradient of the loss function with respect to the input, scaled by a small constant. Projected Gradient Descent (PGD) iterates this process with intermediate projections back onto an allowed perturbation region, typically producing stronger adversarial examples than a single-step method. Carline and Wagner proposed an optimization-based attack that searches for minimal perturbations under an explicit distance metric. In black-box settings, where the attacker cannot access model gradients, techniques such as query-based zeroth-order optimization, surrogate-model transfer attacks, and evolutionary or genetic-algorithm search have been used to approximate gradient information or directly search the input space for misclassifying perturbations.

2.3 Adversarial Machine Learning Applied to Network Security:

Applying these techniques directly to network-flow data is nontrivial because, unlike pixel intensities, network features are heterogeneous, discrete, and constrained by protocol semantics and operational functionality. A perturbed flow that reduces detection scores but breaks the underlying protocol, or that no longer accomplishes the attacker's operational goal (for example, successfully infiltrating data or maintaining a command channel), is not a viable evasion. Several studies have therefore proposed functionality-preserving perturbation spaces: modifying packet padding and inter-arrival timing, injecting benign-looking decoy flows, adjusting flow duration and byte counts within protocol-legal bounds, or restricting perturbations to features that do not affect the attack's payload delivery. Generative approaches, particularly generative adversarial networks (GANs), have also been used to

synthesize adversarial traffic samples that resemble benign flow distributions while retaining malicious functionality.

On the defensive side, adversarial training — augmenting the training set with adversarial examples generated during training — remains the most consistently effective general-purpose defences across domains, though it is computationally expensive and can reduce clean accuracy. Input transformation defencess, such as feature squeezing, quantization, and denoising, aim to remove adversarial perturbations before classification. Ensemble-based defences exploit the observation that different model architectures often have different, non-overlapping blind spots, making simultaneous evasion of all ensemble members harder. Statistical and distributional defences monitor for concept or covariate drift that may indicate an ongoing adversarial campaign rather than natural traffic evolution. This paper synthesizes these defences families into a single layered framework and evaluates them specifically in the network-flow domain, an area comparatively less studied than image-domain adversarial robustness.

3. Threat Model:

3.1 Attacker Objectives:

We consider an attacker whose network traffic (for example, a port scan, a brute-force login attempt, a botnet beacon, or a data-exfiltration flow) is currently detected by a target NIDS. The attacker's objective is to modify the traffic so that the classifier assigns it a benign label, while preserving the operational goal of the traffic. This constraint functionality preservation is essential: unlike an image, where imperceptibility to a human is the primary constraint, network traffic evasion must satisfy protocol correctness and preserve the attacker's ability to accomplish the underlying malicious action.

3.2 Attacker Knowledge Levels:

We evaluate three levels of attacker knowledge, consistent with standard adversarial ML taxonomy adapted to the network domain:

- **White-box:** The attacker has full knowledge of the model architecture, parameters, and training data, and can compute exact gradients of the model's loss function with respect to input features. This represents a worst-case scenario, relevant when an attacker has compromised the defender's infrastructure or when the defender uses a well-known open-source model without modification.
- **Gray-box:** The attacker knows the general model family and feature set (for example, that the defender uses a

gradient-boosted tree ensemble on flow-based features) but does not have access to the exact trained parameters. The attacker trains a surrogate model on similar data and uses transferability of adversarial examples across models.

- **Black-box:** The attacker has no knowledge of the model internals and can only observe the classifier's output (label or confidence score) through a limited number of queries, or must rely purely on traffic-level heuristics without any query access at all.

3.3 Perturbation Space:

We restrict all perturbations to a functionality-preserving feature subset. For flow-based features, this includes packet inter-arrival time padding, flow duration extension via idle-time injection, packet size padding up to the maximum transmission unit, and reordering of packets within protocol-legal windows. Features that would break protocol correctness or attacker objectives — such as source/destination IP address randomization beyond what the attack requires, or modification of payload bytes central to an exploit — are excluded from the perturbation space. This ensures that measured evasion success reflects realistic, deployable attacks rather than artefacts of unconstrained optimization.

4. Methodology: Detection and Mitigation Framework:

We propose a layered defences architecture consisting of four complementary components, designed so that an attacker who defeats one layer still faces meaningful resistance from the remaining layers.

4.1 Layer 1: Adversarial Training:

During training, we augment each mini-batch with adversarial examples generated on-the-fly using PGD within the functionality-preserving perturbation space defined in Section 3.3. The model is optimized to minimize a weighted combination of loss on clean examples and loss on adversarial examples, following the standard min-max robust optimization formulation. To keep training tractable for tree-based ensembles, which are not directly differentiable, we approximate adversarial training for Random Forest and XG Boost by iteratively retraining on a growing adversarial example pool generated via the genetic-algorithm attack described in Section 5.2, a technique sometimes referred to as adversarial boosting.

4.2 Layer 2: Feature-Space Input Transformation:

Before classification, incoming flow features are passed through a transformation stage that includes feature squeezing (reducing numerical precision of continuous features such as byte counts and timing statistics) and a denoising auto encoder trained to reconstruct clean flow feature vectors from perturbed inputs. The intuition is that small adversarial perturbations, while sufficient to cross a decision boundary, are not consistent with the manifold of natural traffic and can be partially reversed by a model trained to project inputs back onto that manifold.

4.3 Layer 3: Ensemble Disagreement Scoring:

Rather than relying on a single classifier, the framework scores each flow using three independently trained models with different architectures and, where feasible, different feature subsets: a Random Forest, an MLP, and an XG Boost model. In addition to the majority-vote label, the framework computes a disagreement score based on the variance of predicted probabilities across the ensemble. High disagreement is treated as a suspicious indicator in its own right, since a successful adversarial example crafted against one model architecture frequently fails to transfer perfectly to architecturally dissimilar models, producing elevated inter-model disagreement even when the majority label is incorrect.

4.4 Layer 4: Statistical Drift Monitoring:

Finally, the framework maintains a rolling statistical profile of feature distributions for traffic classified as benign, using the Kolmogorov–Smirnov test and population stability index to detect distributional drift over sliding time windows. A sustained, statistically significant shift in the feature distribution of traffic labelled benign — particularly concentrated around the decision boundary — is flagged for analyst review, since such shifts are consistent with a systematic evasion campaign rather than natural, gradual traffic evolution.

These four layers are combined into a single pipeline: incoming flows pass through the transformation stage, are scored by the ensemble, and contribute to the drift monitor; a flow is escalated for further inspection if it is labelled malicious by any layer or if it produces high ensemble disagreement, and the drift monitor independently triggers periodic model retraining and analyst alerts.

5. Experimental Setup:

5.1 Datasets:

We use two widely adopted intrusion detection benchmark datasets. NSL-KDD provides 41 flow-level and content-level features across normal traffic and four attack categories (DoS, Probe, R2L, U2R), with the redundant records of the original KDD Cup 1999 dataset removed. CIC-IDS2017 provides labelled benign and attack traffic captured over five days in a realistic network testbed, including DoS, DDoS, brute force, infiltration, botnet, and web attack traffic, with 78 flow-based features extracted using the CICF low Meter tool. For both datasets, we use an 70/15/15 train/validation/test split, stratified by class label.

5.2 Evasion Attacks Implemented:

We implement four attacks spanning the threat model of Section 3:

- FGSM (white-box): A single-step gradient-based perturbation applied to the MLP classifier, restricted to the functionality-preserving feature subset via a masked gradient.
- PGD (white-box): A ten-step iterative refinement of FGSM with a step size of 0.01 and an L-infinity perturbation budget tuned per feature to remain within protocol-legal bounds.
- Genetic Algorithm attack (black-box): A population-based search that iteratively mutates and recombines candidate perturbed flows, using the target model's output confidence as a fitness function, without requiring gradient access. This attack is used against the Random Forest and XGBoost models, which are not natively differentiable.
- GAN-based traffic synthesis (gray-box): A generative adversarial network trained on a surrogate model's decision boundary to synthesize adversarial flow feature vectors that resemble the benign traffic distribution while retaining labelled malicious characteristics from the original flow.

5.3 Target Models:

Three classifier families are evaluated as targets: a Random Forest with 200 trees, a Multi-Layer Perceptron with two hidden layers of 128 and 64 units using ReLU activations, and an XG Boost classifier with 300 boosting rounds. Each is trained both in an undefended (baseline) configuration and with the full four-layer defences framework of Section 4 applied.

5.4 Evaluation Metrics:

We report detection rate (recall on the attack class), false positive rate (on the benign class), F1 score, and evasion success rate, defined as the proportion of originally correctly detected malicious flows that are reclassified as benign after perturbation. We additionally report the average perturbation magnitude required for successful evasion, normalized per feature, as an indicator of attack cost and detectability by a human analyst.

6. Results and Discussion:

6.1 Baseline Vulnerability:

Table 1 summarizes the evasion success rate of each attack against undefended models on the CIC-IDS2017 test set. All three undefended classifiers exhibit substantial vulnerability. The MLP, being fully differentiable, is the most susceptible to white-box gradient attacks, with PGD achieving a 71% evasion success rate. The tree-based models are more resistant to gradient-based attacks by construction, since gradients are not directly defined, but remain highly vulnerable to the black-box genetic-algorithm attack, which achieved a 58–64% evasion success rate by directly optimizing against model output.

Model	FGSM	PGD	Genetic Algorithm	GAN Synthesis
Random Forest	12%	19%	58%	41%
MLP	54%	71%	63%	52%
XG Boost	9%	22%	64%	38%

Table 1. Evasion success rate of undefended models on CIC-IDS2017 test set.

These results confirm a broader pattern observed in the adversarial ML literature: model architecture strongly shapes vulnerability profile, but no architecture tested is inherently robust once the attacker is permitted a sufficiently capable search strategy, whether gradient-based or evolutionary.

6.2 Effectiveness of the Layered Defences:

Table 2 reports evasion success rates for the same attacks against models protected by the full four-layer defences framework. Across all model and attack combinations, the defences framework substantially reduces evasion success. The largest relative improvement is observed for the MLP under PGD, where evasion success drops from 71% to 14%, a recovery of roughly 80% of the originally lost

detection capability. The genetic-algorithm attack against tree-based models proved the most resilient to defences, with residual evasion rates of 19–24%, reflecting the difficulty of adversarial training for non-differentiable models using an approximate retraining procedure rather than exact gradient-based optimization.

Model	FGSM	PGD	Genetic Algorithm	GAN Synthesis
Random Forest	4%	7%	24%	13%
MLP	9%	14%	22%	16%
XG Boost	3%	8%	19%	12%

Table 2. Evasion success rate of defended models (full four-layer framework) on CIC-IDS2017 test set.

6.3 False Positive Overhead:

A critical practical concern for any defences is the cost imposed on legitimate traffic. Across the three models, the layered defences increased the false positive rate on held-out clean benign traffic by an average of 1.8 percentage points, ranging from 1.1 points for Random Forest to 2.7 points for the MLP. This overhead is attributable primarily to the input transformation layer, which occasionally over-smooths legitimate but statistically unusual benign flows, and to the ensemble disagreement layer, which flags some legitimate edge-case traffic for additional review. We consider this a manageable operational cost given the corresponding gains in adversarial detection rate, though it underscores the general robustness–accuracy trade-off well documented in the adversarial ML literature: defencess that reduce a model's sensitivity to small input perturbations necessarily reduce its sensitivity to some genuinely informative variation as well.

6.4 Component Ablation:

To isolate the contribution of each defences layer, we conducted an ablation study on the MLP model against the PGD attack, removing one layer at a time from the full framework. Adversarial training alone recovered the largest share of lost accuracy, reducing evasion success from 71% to 27%. Adding input transformation further reduced evasion success to 19%. Adding ensemble disagreement scoring (evaluated as a three-model ensemble in place of the single MLP) reduced evasion success to 16%, since a successful attack against the MLP surrogate model did not reliably transfer to the Random Forest and XGBoost ensemble members. The

drift monitor did not directly reduce per-flow evasion success, since it operates at a longer time horizon, but in a simulated sustained-campaign scenario spanning several hours of continuous evasion attempts, it correctly flagged the campaign for analyst review after a median of 42 minutes, well before cumulative damage would typically be expected to reach containment thresholds in a production security operations workflow.

This ablation indicates that no single layer is sufficient on its own, but that the layers are largely complementary: adversarial training addresses the largest share of gradient-based vulnerability, ensemble disagreement addresses residual vulnerability that does not transfer well across architectures, and the drift monitor provides a time-horizon safety net for slow, low-and-slow evasion campaigns that individual flow-level defencess are not designed to catch.

6.5 Cross-Dataset Generalization:

We repeated the core experiments on NSL-KDD to assess whether findings generalize beyond a single dataset. The overall pattern held: undefended models showed evasion success rates of 31–68% depending on model and attack type, and the layered defences reduced these rates to 9–26%. Absolute numbers differed modestly from the CIC-IDS2017 results, which we attribute to differences in feature dimensionality (41 features in NSL-KDD versus 78 in CIC-IDS2017) and to the older, less traffic-diverse nature of the KDD-derived data. The consistent direction and magnitude of improvement across two structurally different datasets provides reasonable confidence that the defences framework's benefits are not an artifact of a single dataset's idiosyncrasies.

7. Mitigation Strategies and Deployment Considerations:

7.1 Practical Recommendations:

Based on the experimental findings, we offer the following recommendations for organizations deploying ML-based NIDS in production environments:

- Do not rely on a single model architecture. Ensemble disagreement provided a measurable and low-cost robustness gain across every attack tested, and is comparatively simple to deploy alongside existing infrastructure.
- Prioritize adversarial training for differentiable models, since it produced the largest single-layer improvement in our ablation study, particularly for neural network components of a detection pipeline.

- Treat evasion resistance for tree-based ensembles as a distinct engineering problem. Because tree ensembles are not natively differentiable, defenders should budget for iterative, search-based adversarial retraining rather than assuming gradient-based adversarial training techniques transfer directly.
- Deploy statistical drift monitoring as a complementary, longer-horizon safety net rather than a replacement for flow-level defencess, since it is well suited to catching sustained or low-and-slow evasion campaigns that per-flow classifiers may individually tolerate.
- Budget for an ongoing false-positive review process, since all forms of adversarial robustness in our experiments came at a measurable, non-zero cost to false-positive rates on clean traffic.

7.2 Operational Trade-offs:

Deploying the full four-layer framework increases inference latency relative to a single undefended model, since incoming flows must pass through the transformation stage and be scored by three ensemble members rather than one. In our implementation, average per-flow inference latency increased from 0.8 milliseconds for a single undefended Random Forest to 3.1 milliseconds for the full pipeline. For most enterprise network monitoring use cases, where flow-level classification is not required at line-rate for every packet, this overhead is unlikely to be operationally prohibitive, but it may be a meaningful constraint for very high-throughput environments such as core internet exchange points or large content delivery networks, where lighter-weight defences configurations — for example, adversarial training alone, without the full ensemble — may represent a more appropriate trade-off.

8. Limitations and Future Work:

This study has several limitations that suggest directions for future research. First, our functionality-preserving perturbation space, while grounded in prior literature, is necessarily an approximation of the full range of transformations a sophisticated attacker might discover; more exhaustive protocol-aware perturbation modelling, potentially informed by red-team exercises against live systems, would strengthen confidence in the results. Second, our evaluation is conducted on offline benchmark datasets; live production traffic exhibits additional complexity, including encrypted payloads, middle box interference, and non-stationary benign traffic patterns, that offline benchmarks cannot fully capture. Third, we do not evaluate certified robustness

techniques, which provide provable guarantees against perturbations within a bounded region, as opposed to the empirical robustness improvements measured here; extending certified defences techniques, which have seen substantial development for image classifiers, to the heterogeneous, mixed discrete-continuous feature spaces typical of network flow data remains an open and promising research direction. Fourth, our study assumes a relatively static attacker; an adaptive attacker aware of the specific defences framework deployed, including its ensemble composition and drift-monitoring thresholds, could in principle craft more targeted evasion strategies, and evaluating the framework against such adaptive, defences-aware attackers is an important next step. Finally, extending this work to federated or distributed intrusion detection settings, where multiple organizations collaboratively train detection models without sharing raw traffic data, raises additional adversarial robustness questions, including the risk of poisoning attacks during federated training, that fall outside the scope of the evasion-focused analysis presented here.

9. Conclusion:

Machine-learning-based intrusion detection systems offer meaningful advantages over purely signature-based approaches, but they inherit the vulnerability of any learned decision boundary to adversarial evasion. This paper demonstrated, across two benchmark datasets, three classifier architectures, and four representative attack techniques, that undefended ML-based NIDS can suffer severe detection degradation under realistic, functionality-preserving evasion attacks, with evasion success rates as high as 71%. We proposed and evaluated a layered defences framework combining adversarial training, feature-space input transformation, ensemble disagreement scoring, and statistical drift monitoring, and showed that this framework recovers a substantial majority of lost detection accuracy while imposing a modest, operationally manageable false-positive overhead. No single defensive technique fully closed the gap, and our ablation results suggest that robustness is best achieved through the combination of complementary layers rather than reliance on any one method. As ML-based defencess become more deeply embedded in production security infrastructure, adversarial robustness should be treated not as an optional enhancement but as a core design requirement, evaluated with the same rigor traditionally applied to detection accuracy alone.

References

Biggio, B., & Roli, F. (2018). Wild patterns: Ten years after the rise of adversarial machine learning. Pattern Recognition, 84, 317-331.

Carlini, N., & Wagner, D. (2017). Towards evaluating the robustness of neural networks. IEEE Symposium on Security and Privacy.

Goodfellow, I. J., Shlens, J., & Szegedy, C. (2015). Explaining and harnessing adversarial examples. International Conference on Learning Representations.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2018). Towards deep learning models resistant to adversarial attacks. International Conference on Learning Representations.

Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., & Swami, A. (2017). Practical black-box attacks against machine learning. ACM Asia Conference on Computer and Communications Security.

Rigaki, M., & Garcia, S. (2018). Bringing a GAN to a knife-fight: Adapting malware communication to avoid detection. IEEE Security and Privacy Workshops.

Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. International Conference on Information Systems Security and Privacy.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. (2014). Intriguing properties of neural networks. International Conference on Learning Representations.

Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). A detailed analysis of the KDD CUP 99 data set. IEEE Symposium on Computational Intelligence for Security and Defences Applications.

Xu, W., Evans, D., & Qi, Y. (2018). Feature squeezing: Detecting adversarial examples in deep neural networks. Network and Distributed System Security Symposium.

*Corresponding Author: Dr. Prashant Kohli
E-mail:prashant.kohli01@gmail.com
Received: 04 November,2025; Accepted: 23 December,2025. Available online: 30 December, 2025
Published by SAFE. (Society for Academic Facilitation and Extension)