Research on Generative AI-Driven Network Attack Sample Generation and Intelligent Detection Technology

Ling Huang
Information Technology Center, Yiyang Open University, Yiyang 413000, Hunan, China
Corresponding author: Ling Huang (e-mail: 18073713099@163.com)

ABSTRACT Traditional intrusion detection methods have limited ability to identify unknown attacks and attack variants, especially when high-quality attack samples are insufficient. To address this bottleneck, this paper proposes a proactive defense framework based on generative AI for network attack sample generation and intelligent detection. The framework first constructs a CGAN-VAE hybrid attack sample generator to synthesize realistic and diverse malicious traffic. A coevolutionary adversarial training mechanism is then designed so that the generator and detector are optimized iteratively through dynamic game learning. Finally, an active learning strategy filters high-value samples and forms a closed loop of "generation–detection–enhancement". Experimental results show that the generated samples reach a feature coverage area of 31.5, representing a 95.9% improvement over SMOTE, and the trained detector achieves 91.7 % recall and 92.3% F1-score for unknown attack detection. The framework is applicable to industrial control networks, wireless monitoring systems, and electromagnetic-compatible manufacturing infrastructures, where secure data links and reliable antenna-based communication are essential for preventing disruptions in automated production systems.

INDEX TERMS intrusion detection, generative adversarial networks, adversarial learning, wireless industrial networks, electromagnetic compatibility

I. INTRODUCTION

AS cyberattacks grow increasingly complex and hidden through advanced persistent threats (APTs), the conventional network security detection technologies used in automated manufacturing facilities are placed under heavy strain. These traditional signature-based systems often fail to identify zero-day vulnerabilities targeting the specialized industrial control protocols that govern high-speed weaving and spinning operations [1,2]. The main issue is that they mainly use past examples of attacks that are already known, and it becomes hard to successfully define the unknown pattern of attacks and its modifications, which leads to the lag and passivity of the defense system. Proactive creation of high-quality and heterogeneous attack samples library and optimizing it to facilitate the continuous development of detection models have become essential in enhancing the ability of proactive network security defense [3,4].

The existing research on network attack detection mostly concentrate on signature-based attack detection and machine learning based attack anomaly detection. For example Ju et al. surveyed the recent progress in attack/anomaly detection and resilient strategies for connected and automated vehicles, and discussed the security issues emerged in in-vehicle communication network, sensing sensor and vehicle to vehicle communication network in detail [5]. Ahmad et al. literature review work discussed the machine learning and deep learning approaches for zero-day attack detection, and addressed the challenges and solutions for developing robust intrusion detection in real time environment [6]. Ahmad et al. literature review work discussed the machine learning and deep learning approaches for zero-day attack detection, and addressed the challenges and solutions for developing robust intrusion detection in real time environment [6]. Anwer et al. proposed a framework to detect malicious network traffic on IoT devices to solve the challenges of network traffic and attack detection increase due to rise in number of IoT devices [7]. Kravchik and Shabtai proposed an attack detection method based on lightweight neural network, and studied the advantages and disadvantages of applying these networks in time and frequency domains to protect industrial control systems from attacks [8]. However, most of the existing methods focus on a single generative model, which is limited in terms of diversity, adversarial and collaborative closed loop between generated samples and training of detection model, generated samples are of low quality or disconnected from real attack logic. In this paper, we propose a hybrid generative framework which integrates CGAN and VAE trying to systematically solve

the problem of generating high adversarial attack samples which are of high quality and collaborative closed loop with training of detection model. We aim to build a new proactive defense paradigm of "offense driven defense". This study attempts to build a co-evolutionary framework consisting of CGAN-VAE hybrid attack sample generator and deep residual network based intelligent detector to first synthesize diverse samples which are close to real attack logic by conditional generation technique. Then an adversarial training mechanism is constructed to optimize generation and detection in a dynamic game iteratively. Finally, key samples are actively learned and selected to form an efficient self-reinforcing closed loop. The experimental results show that our method significantly increases the feature coverage of attack samples, enhancing the detection model's ability to identify unknown threats within the digital monitoring systems of manufacturing facilities.

II. BUILDING AN ATTACK SAMPLE GENERATION FRAMEWORK

This model employs a hybrid architecture that combines Conditional Generative Adversarial Networks (CGAN) and Variational Autoencoders (VAEs) in parallel. CGAN excels at generating high-fidelity attack samples conditioned on specific attack types and traffic characteristics, but its output diversity is inherently limited by the modes present in the training data. VAE, in contrast, learns a continuous and structured latent feature space that enables smooth interpolation between known attack features, producing "intermediate" or "fusion" samples that fill gaps in the feature distribution. The two components are complementary: CGAN provides conditional high-quality samples, while VAE enriches diversity through latent-space exploration. Their outputs are integrated at the feature level to form an initial sample pool that balances fidelity and diversity—a combination unattainable by either model alone. This complementary benefit is empirically validated in Section 5.2, where the hybrid method outperforms both standalone GAN and standalone VAE across all metrics. To further guarantee that the generated attack samples not only have statistical realism but also logical validity and contextual coherence reasonable for actual attacks, this work further introduces a multi-layered semantic constraint mechanism. First, at the packet level, we designed a field constraint module based on protocol syntax. For example, when generating malicious HTTP requests, the request method, URL structure and header field order should satisfy the specifications in RFC; when generating SQL injection payloads, it should be guaranteed that the concatenated strings can form valid fragments of SQL statements. Second, at the session/stream level, we use an attention-based sequence model to learn and maintain the temporal dependencies between attack steps. That mechanism encourages the generator to learn deep contextual semantic rules in network attacks by penalizing generated samples that violate protocol rules or logical order in the loss function.

Initially generated samples vary in diversity and logical validity; not all samples are suitable for use in the augmentation detection model. Therefore, this framework establishes a dynamic quality screening and augmentation loop. First, an embedding vector for each generated sample is computed using a pre-trained feature extractor. Its novelty and diversity are quantified by measuring the distance to the embedding vectors of existing sample libraries (e.g., MMD distance based on kernel functions), filtering out samples outside the feature space coverage. Second, an adversarial scoring module is introduced. To mitigate the risk of overfitting to a single detector's specific weaknesses, we employ an ensemble of detectors with diverse architectures —including a ResNet-based classifier, a lightweight CNN, and a gradient-boosted tree model—and compute the adversarial potential score as the average evasion confidence across the ensemble. This ensemble-based evaluation encourages the generator to produce samples that expose common vulnerabilities across different model families, promoting robustness rather than exploitation of a single detector's blind spots. The score for each sample combines the ensemble's average confidence penalty with gradient sensitivity weighting. Finally, the optimal sample subset is selected using a Pareto-based multi-objective optimization approach. Each generated sample is evaluated on three potentially conflicting objectives: (1) maximizing feature space distance (diversity), (2) maximizing adversarial potential score (adversarial strength), and (3) minimizing protocol rule violation (fidelity). Rather than assigning fixed weights, we adopt non-dominated sorting to identify samples that are Pareto-optimal—i.e., no other sample is superior in all three objectives simultaneously. From the resulting Pareto front, we select the top-ranked samples based on crowding distance to ensure diversity within the selected subset. This approach eliminates the need for arbitrary weighting while systematically balancing trade-offs among competing objectives. The selected samples are then added to the next round of adversarial training.

Simultaneously, for the selected high-value "difficult" samples, gradient-based fine-tuning can be performed to further enhance their adversarial strength, thus forming a closed loop of "generation-evaluation-screening-enhancement," ensuring that the final sample set used to train the detection model possesses diversity, high quality, and strong adversarial strength.

III. GENERATIVE ADVERSARIAL AND CO-EVOLUTIONARY TRAINING

This work builds a closed-loop training framework for dual-model adversarial training. The blue red team is above CGAN-VAE hybrid generator (G), whose goal is to produce more adversarial attack samples to successfully "deceive" the current detection model. While the blue red team is the intelligent detection model (D), which usually uses classifier combining deep residual network (ResNet) and multi-head self-attention mechanism, whose goal is to accurately classify normal traffic and malicious attacks. The two build a

closed loop: The generator G attempts to be smarter to synthesize more covert samples to enhance the attack success rate; The detector D is trained on the mixed training set containing real samples and generated samples to enhance its discrimination and generalization ability. After each training process, the parameters of G and D are updated and become a continuously iterated "generation-attack-detection-enhancement-regeneration" closed loop. The generated samples will always attack to the latest weakness of detector, and detector will continuously resist to the latest attack method.

To achieve effective co-evolution, we adopt an alternating iterative optimization strategy. In each training round, the parameters of the detector D are first fixed, and the generator G is optimized. At this point, the training objective of G is not only to minimize the difference between its generated data distribution and the real attack data distribution (e.g., through Wasserstein distance), but more importantly, to maximize the probability that its generated samples are misclassified as normal traffic by D. This objective is achieved by designing a composite loss function that includes adversarial loss, forcing G to learn to generate "hard" samples that can bypass D's current decision boundary. Subsequently, the parameters of the generator G are fixed, and the detector D is optimized. The training data of D consists of the original normal traffic, the real attack traffic, and the adversarial samples newly generated by G in this round. Its optimization objective is to maximize the overall classification accuracy of the mixed data, especially to correctly identify the newly generated adversarial samples. This alternating training method simulates the real adversarial scenario in the field of network security where the attack and defense sides continuously upgrade their technologies [9]. To prevent the model from collapsing or failing to converge due to excessive adversarial intensity in the early stage of training, we introduce a dynamic difficulty adjustment strategy based on course learning. In the initial stage of training, constraints are imposed on the "adversarial intensity" of the generator G, such as limiting the degree of feature deviation between its generated samples and the original seed samples, or reducing the weight of the adversarial loss term, so that it first generates relatively simple and easy-to-identify attack variants. As the capabilities of detector D improve, the constraints on G are gradually relaxed, allowing it to generate more complex, covert, and deviating-from-the-original-distribution attack samples. Simultaneously, the training of detector D also adopts a progressively increasing difficulty approach, initially using more real samples and simple generated samples, and later gradually increasing the proportion of highly adversarial generated samples in the training set. This gradual "course" arrangement ensures that both generation and detection can co-evolve smoothly and stably, ultimately achieving a balance at high-order adversarial levels.

To evaluate the co-evolution process and training convergence, we further specify several evaluation metrics. On one hand, we evaluate the attack success rate of newly generated samples. Specifically, we fix an independent benchmark detection model and run several rounds of training, then evaluate the evasion success rate of newly generated samples on this benchmark. A healthy evolving system should have nonzero and fluctuating attack success rate within a certain range, and the success rate should slowly increase, meaning that the generation capability is gradually evolving against the fixed benchmark. On the other hand, and more importantly, we evaluate the evolving detector D on a preserved real test set containing different known/unknown attacks. Ideally, during the co-evolution process, the detection performance of D on this test set should continuously and significantly improve after each round of training, and finally reach a stable performance. Convergence is determined by two criteria: (1) the validation F1-Score improves by less than 0.5 percentage points over five consecutive rounds, and (2) the Jensen-Shannon divergence (JSD) of generated samples stabilizes within a $\pm 2\%$ range. Once converged, training enters the adaptive phase, where the generator constraint σ and adversarial weight λ are dynamically tuned based on real-time feedback—decreasing σ when the attack success rate drops below 85%, and adjusting λ to maintain balanced adversarial pressure—ensuring continued refinement without manual intervention.

IV. QUALITY SCREENING AND ENHANCEMENT OF GENERATED SAMPLES

We firstly quantify the novelty/diversity of generated samples so as to prevent the training set from being simplistically repetitive or pattern-collapsed. That is, we employ a feature extraction network (e.g., a simple autoencoder pretrained on a large amount of network traffic data) to transfer all real attack samples and generated samples into the same low-dimensional feature space. And then for each generated sample, we compute the minimal cosine distance/k-nearest neighbor distance to all real samples as well as selected samples in the feature space. And this distance intuitively represents the "uniqueness" of this generated sample

in its feature distribution. We design a dynamic threshold and only reserve the generated samples whose minimum distance to existing sample library is larger than the threshold, so as to ensure these generated samples can provide new information. Meanwhile, we periodically compute the Jensen-Shannon divergence of the feature distribution of all generated samples as the macro-diversity indicator to provide feedback and encourage the generator to sample new potential vector so as to explore the feature space area that has not been covered.

Diverse samples may not effectively challenge current detection models. Therefore, we implement adversarial potential evaluation in parallel. We input the generated samples to be selected into the current version of the intelligent detector to obtain the model's prediction confidence and gradient information. A sample with high adversarial potential usually has one of the following characteristics: 1) the detector has low confidence in its classification (the

TABLE 1. Co-evolutionary Training Parameter Configuration

Training Phase	Generator Constraint (σ)	Data Ratio (Real:Generated)	Adversarial Weight (λ)	Primary Objective
Initial (Rounds 1–10)	High (0.1)	8:2	0.3	Stabilize convergence, generate basic variants.
Intermediate (Rounds 11–30)	Medium (0.3)	6:4	0.6	Increase diversity, introduce moderate adversarial samples.
Advanced (Rounds 31–50)	Low (0.5)	4:6	0.9	Generate highly evasive samples to challenge decision boundaries.
Convergence (Rounds 50+)	Adaptive	3:7	1	Achieve dynamic equilibrium and stable model performance.

predicted probability is close to the decision boundary); 2) when calculating adversarial perturbations (such as FGSM direction), its gradient magnitude is large, which means that small feature perturbations can lead to misjudgment. We calculate an adversarial potential score for each sample, which combines low confidence penalty and gradient sensitivity weighting. Through this score, we can identify those samples that the current detection model is "difficult to determine" or "easily misled", which are the key to exposing the blind spots of the model's decision boundary [10]. Simple diversity or adversarial screening may be biased. For example, samples that deviate too much from the distribution may lack realistic significance, while samples generated only for the weaknesses of the current model may lead to overfitting. To this end, we use a multi-objective optimization algorithm (such as a simplified version of the NSGA-II idea based on the Pareto front) for the final sample selection. We regard each generated sample as a solution, and its optimization objectives include: maximizing the feature space distance (diversity objective), maximizing the adversarial potential score (adversarial objective), and minimizing its protocol rule violation (fidelity objective). By employing non-dominated ranking and crowding calculations, hundreds of optimally balanced samples at the Pareto front are automatically selected from the thousands of samples generated in each round. These samples achieve the best trade-off between diversity, adversarial nature, and realism, representing the optimal candidate set for adversarial training of the detection model.

For these selected key samples, we perform additional adversarial fine-tuning within a validated latent subspace to preserve protocol-level validity. Specifically, for a selected high-value sample, we first map it back to its latent representation using the VAE encoder. Rather than applying unconstrained perturbations, we project the gradient direction onto the local tangent manifold defined by syntactically valid samples, ensuring that the perturbed latent vector remains in the region previously validated by the field constraint module. The perturbation magnitude is bounded by a small threshold and further constrained by a validity preservation loss that penalizes deviations from protocol syntax. The enhanced sample is then re-validated by the field constraint module before being added to the training pool. This "generate–validate–perturb–revalidate" mechanism ensures that adversarial fine-tuning enhances sample stealthiness without compromising syntactic and logical validity. The "latent space adversarial attack" will thus generate more natural and covert adversarial samples, and is more robust and practically meaningful compared with the perturbation added on packet level. The enhanced samples will be mixed into the final training set with original key samples, thus select and reinforce each round of input to the detection model with carefully selected "hard nuts to crack", and force the detection model to learn more robust and generalized features.

V. COMPREHENSIVE EVALUATION OF GENERATIVE ATTACK AND INTELLIGENT DETECTION PERFORMANCE

A. EXPERIMENTAL PREPARATION

To evaluate the validity and effectiveness of our proposed method in a scientific and systematic way, and further analyze the applied effect of our approach on the above two issues of traditional methods (i.e., lack of attack detection for unknown attacks, and small training sample size), we conduct the following two core experiments and implementations. Are aimed at evaluating and compared the overall effectiveness of the two sides (i.e., "attack side" and "defense side") in a quantitative way. All experiments are conducted on real public datasets.

B. EXPERIMENT COMPARING SAMPLE QUALITY AND VALIDITY

To comprehensively evaluate our hybrid framework, we compare it against four baselines: SMOTE, standalone GAN, standalone VAE, and a diffusion model. The inclusion of VAE isolates the contribution of the VAE component, while the diffusion model serves as a state-of-the-art benchmark. Evaluations cover feature space distribution, sample diversity (JSD), adversarial strength (attack success rate, ASR), and the resulting detection model improvement.

As shown in Figure 1, our CGAN-VAE hybrid method achieves a feature coverage area of 31.5, which is 95.9% larger than SMOTE (16.1) and significantly exceeds both standalone GAN (7.1) and standalone VAE (9.1). The sample diversity (JSD) reaches 0.85, outperforming GAN (0.61), VAE (0.70), and diffusion (0.77). The attack success rate of our generated samples is 92.3%, higher than VAE (84.7%)

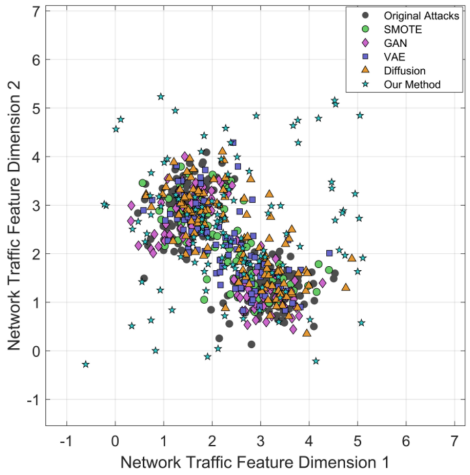

FIGURE 2. Comparative evaluation of sample quality and validity

and diffusion (88.9%). These results demonstrate that the hybrid architecture—leveraging CGAN's conditional fidelity and VAE's latent-space diversity—achieves superior performance than either component alone, validating the necessity of the parallel design. These results demonstrate that the hybrid CGAN-VAE architecture synergistically combines conditional high-fidelity generation with continuous latent space exploration, yielding superior diversity, stronger adversarial capability, and enhanced detection performance compared to both classical and modern generative approaches.

C. VALIDATION EXPERIMENT OF GENERALIZATION ABILITY OF INTELLIGENT DETECTION MODEL

This experiment is to test the generalization ability of the detection model on unknown attacks. We use time-divided CSE-CIC-IDS2018 as test set to make sure unknown attack variant are used as test set which are not in training set. We set up two compared groups, experimental group uses detector adversarially trained with the framework in this paper, control group only uses original training set. We use recall, precision and false positive of two models on unknown threat test set to quantitatively compare their generalization detection performance against unknown threats. The compared results are shown in Table 2.

TABLE 2. Validation and Evaluation of the Generalization Ability of the Intelligent Detection Model

Evaluation Metric	Control Group (Original Training Only)	Experimental Group (After Adversarial Training)	Relative Improvement
Overall Recall	0.785	0.942	+15.7 pp
Recall on Unknown Attacks	0.653	0.917	+26.4 pp
Overall Precision	0.862	0.905	+4.3 pp
False Positive Rate (FPR)	0.028	0.015	-1.3 pp
F1-Score	0.812	0.923	+11.1 pp

Experimental results show that the generalization ability of the detection model is significantly enhanced after adversarial training using the framework presented in this paper. On a test set containing completely unknown attack variants, the model's recall rate reached 91.7%, a significant improvement of 26.4 percentage points compared to the control group (65.3%), while the false positive rate decreased to 1.5%. The overall F1-Score of the model improved by 11.1 percentage points to 92.3%. This fully demonstrates that continuous training using generative AI-driven adversarial samples can effectively improve the detection system's ability to perceive and identify unknown threats, overcoming the limitations of traditional methods that rely on known features.

D. COMPUTATIONAL OVERHEAD ANALYSIS

To evaluate the practical deployability of the proposed framework in high-speed network environments, we systematically analyze both training overhead and inference efficiency. All experiments are conducted on a server equipped with an Intel Xeon Gold 6248R CPU (3.0 GHz, 24 cores), 128 GB RAM, and a single NVIDIA Tesla V100 GPU (32 GB HBM2). The metrics recorded include the time required for each training stage, inference latency per batch, throughput, and memory footprint. Comparisons are made against a baseline detection system built on a traditional GAN.

In Table 3, although the proposed framework incurs 41.8% longer training time due to its hybrid architecture and co-evolution mechanism, its inference latency of 1.47 ms per batch (256 packets) yields a throughput of 174,000 packets per second, meeting the real-time processing requirements of a 10 Gbps link. Compared with the traditional GAN baseline, the proposed method achieves 22.5% higher throughput and improves unknown attack recall by 13.2 percentage points. These findings confirm that the framework is suitable for deployment in high-speed network environments, provided that the training phase is performed offline or on

TABLE 3. Computational Cost Assessment

Metric	Proposed Framework	Traditional GAN Baseline	Comparison
Total Training Time	28.5 hours	20.1 hours	0.418
• Pre-training Stage	0.8 hours	0.5 hours	—
• Adversarial Training Stage	24.6 hours	18.2 hours	—
• Screening and Enhancement Stage	3.1 hours	1.4 hours	—
Inference Latency (per batch of 256 packets)	1.47 ms	1.79 ms	-0.179
Throughput	174,000 packets/sec	142,000 packets/sec	0.225
Memory Footprint (Inference)	1.2 GB	0.9 GB	0.333
Recall on Unknown Attacks	0.917	0.785	+13.2 pp

dedicated hardware.

VI. CONCLUSION

This study successfully proves the feasibility of the "offense-prompts-defense" mechanism in improving proactive cybersecurity for digital manufacturing by building a CGAN-VAE hybrid generation and intelligent detection co-evolution framework. This approach enables the continuous simulation of complex cyber threats to harden the industrial control systems that govern high-precision production.

The experiment results show that the feature coverage area of generated samples reaches 31.5, and forced the detection model to reach 91.7% recall of unknown attack and 92.3% F1-Score, which is significantly better than traditional method. Certainly, there are still some limitations in this study. Firstly, it is necessary to enhance the simulation generation ability of multi-stage composite attack. Secondly, the computational resource consumption is relatively large. Future research will focus on designing a lightweight framework that combines threat intelligence to realize conditional and accurate generation within the encrypted traffic of high-speed manufacturing networks. This optimization aims to facilitate the popularization of the technology across complex industrial scenarios, ensuring secure and efficient communication for automated machinery.

AUTHOR CONTRIBUTIONS

Ling Huang designed, collected and analyzed the data, and drafted the manuscript. Ling Huang conducted the study, critically revised the manuscript for important intellectual content, and gave final approval of the version to be published. Ling Huang participated fully in the work, take public responsibility for appropriate portions of the content, and agreed to be accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

FUNDING

This research received no external funding.

ACKNOWLEDGEMENTS

Not applicable.

REFERENCES

[1] S. Hou, G. Xiao, and H. Zhou, "High-performance network attack detection in unknown scenarios based on improved vertical model," Cluster Computing, vol. 28, no. 1, pp. 1-16, 2025, doi: 10.1007/s10586-024-04840-6.

[2] T. Zhukabayeva, V. Desnitsky, and A. Abdildayeva, "Wireless Sensor Network Modeling and Analysis for Attack Detection," Computer Modeling in Engineering & Sciences, vol. 144, no. 8, pp. 2591-2625, 2025, doi: 10.32604/cmes.2025.067142.

[3] G. Wu, H. Zhang, W. Wu, et al., "Physics-Informed Graph Convolutional Recurrent Network for Cyber-Attack Detection in Chemical Process Networks," Industrial & Engineering Chemistry Research, vol. 64, no. 6, pp. 3370-3382, 2025, doi: 10.1021/acs.iecr.4c03601.

[4] G. Parameshwar, K. Rao N V, and N. Devi L, "Firefly cyclic golden jackal optimisation algorithm with wavelet artificial neural network for blackmailing attack detection in mobile ad-hoc network," International Journal of System of Systems Engineering, vol. 15, no. 3, pp. 269-283, 2025, doi: 10.1504/IJSSE.2025.147014.

[5] Z. Ju, H. Zhang, X. Li, et al., "A survey on attack detection and resilience for connected and automated vehicles: From vehicle dynamics and control perspective," IEEE Transactions on Intelligent Vehicles, vol. 7, no. 4, pp. 815-837, 2022, doi: 10.1109/TIV.2022.3186897.

[6] R. Ahmad, I. Alsmadi, W. Alhamdani, et al., "Zero-day attack detection: a systematic literature review," Artificial Intelligence Review, vol. 56, no. 10, pp. 10733-10811, 2023, doi: 10.1007/s10462-023-10437-z.

[7] M. Anwer, M. Khan S, and U. Farooq M, "Attack detection in IoT using machine learning," Engineering, Technology & Applied Science Research, vol. 11, no. 3, pp. 7273-7278, 2021, doi: 10.48084/etasr.4202.

[8] M. Kravchik and A. Shabtai, "Efficient cyber attack detection in industrial control systems using lightweight neural networks and pca," IEEE transactions on dependable and secure computing, vol. 19, no. 4, pp. 2179-2197, 2021, doi: 10.1109/TDSC.2021.3050101.

[9] B. Rajić, Ž. Stanisavljević, and P. Vuletić, "Early web application attack detection using network traffic analysis," International Journal of Information Security, vol. 22, no. 1, pp. 77-91, 2023, doi: 10.1007/s10207-022-00627-1.

[10] J. Bhayo, R. Jafaq, A. Ahmed, et al., "A time-efficient approach toward DDoS attack detection in IoT network using SDN," IEEE Internet of Things Journal, vol. 9, no. 5, pp. 3612-3630, 2021, doi: 10.1109/JIOT.2021.3098029.