**Scientific Journal of Informatics**
Vol. 13, No. 3, August 2026
p-ISSN 2407-7658 http://journal.unnes.ac.id/nju/index.php/sji e-ISSN 2460-0040

# Adversarial Vulnerabilities in Cooperative Multi-Agent Reinforcement Learning for Distributed 5G Security

**Kudzaishe Lawal Chizengwe[1]*, Belinda Ndlovu[2]**
[1]Informatics and Analytics Department, National University of Science and Technology, Zimbabwe
[2]School of Computing, University of South Africa, South Africa

**Abstract.**
**Purpose:** This paper focuses on examining the robustness of a cooperative Multi-Agent Reinforcement Learning (MARL)-based Intrusion Detection System (IDS) for intrusion detection in decentralised 5G security settings. Even though MARL techniques have proven effective against dynamic threats in decentralized 5G networks, current research has not considered any adversarial scenarios at all.
**Methods:** A cooperative MARL-based Intrusion Detection System was developed through the CRISP-DM approach. Radio Access Network (RAN), MEC, and Core agents were trained using Centralised Training with Decentralised Execution (CTDE) and Deep Q-Network (DQN) methods. The algorithm was tested on the NSL-KDD and UNSW-NB15 datasets against Fast Gradient Sign Method (FGSM) evasion attacks (ε = 0.05-0.30) and Byzantine poisoning attacks with 5%, 10%, and 20% compromised agents.
**Result:** The model achieved 96.94% accuracy on NSL-KDD and 85.15% on UNSW-NB15 in clean scenarios. The FGSM attack at ε = 0.20 resulted in substantial performance deterioration, leading to accuracy drops of 50.14 and 45.26 percentage points, respectively, and a simultaneous increase in false positives. Byzantine poisoning produced smaller but persistent decreases in accuracy of 12.03 and 2.62 percentage points, respectively.
**Novelty:** This study provides among the first empirical evaluations of adversarial fragility in cooperative MARL-based intrusion detection within distributed 5G-oriented security abstractions, demonstrating that cooperative intelligence alone does not guarantee adversarial robustness.

**Keywords**: Agentic AI; 5G networks; Multi-agent reinforcement learning; Intrusion detection; Adversarial machine learning
**Received** May 2026 / **Revised** July 2026 / **Accepted** August 2026



## INTRODUCTION

The advent of 5th generation (5G) wireless communication technology has revolutionized the modern digital landscape through ultra-low-latency communication, Software Defined Networking (SDN), Network Function Virtualisation (NFV), and Multi-access Edge Computing (MEC) [1]–[4]. These capabilities support critical applications such as industrial automation, autonomous systems, healthcare services, and smart city ecosystems, where reliability, scalability, and continuous connectivity are essential [4], [5]. Nevertheless, the very same architectural versatility that enables intelligent service orchestration also significantly expands the cyberattack surface across radio access, edge, and core networks. Recent studies have found a persistent trend of attacks on telecommunication systems over the period from 2020 to 2023 [6], [7]. Furthermore, the stringent latency requirements associated with Ultra-Reliable Low-Latency Communication (URLLC) services require intrusion detection systems capable of identifying and responding to threats within extremely short operational timeframes [6], [8].

Conventional Intrusion Detection System (IDS) techniques, such as rule-based techniques and classical machine learning techniques, have been designed for environments that do not change dynamically, limiting their effectiveness in dynamically evolving threat environments [4], [5], [7], [9]. Although such techniques can produce excellent results in laboratory settings, their applicability is severely restricted in unpredictable 5G environments due to decentralised decision-making and adversarial actions [10], [11]. Notably, cybersecurity surveillance in 5G systems cannot be viewed solely as a classification problem, as the network state, attacks, and system state change constantly over time and across layers [10], [11]. As

[1]*Corresponding author. Belinda Ndlovu
Email addresses: 69970777@mylife.unisa.ac.za
DOI: 10.15294/sji.v13i3.50261

such, intrusion detection in 5G systems becomes more akin to a sequential decision problem, requiring dynamic policies that adapt to a changing environment.
Within this context, Multi-Agent Reinforcement Learning (MARL) has emerged as a promising paradigm for distributed cybersecurity in 5G environments [10], [11]. The MARL framework enables multiple cooperating agents to learn adaptive security policies from their interactions with a constantly changing environment in a distributed network setting. Centralized Training with Decentralized Execution (CTDE)-based techniques are especially appealing because they enable coordinated learning during training while retaining independent decision-making during execution [12]. Recent studies applying reinforcement learning, federated learning, and agentic AI techniques to 5G security have demonstrated promising intrusion-detection capabilities in simulated environments [13],[14],[15],[16],[17]. Federated learning-based approaches have reported detection rates exceeding 94%, while Deep Q-Network (DQN) architectures and cooperative MARL frameworks have demonstrated adaptive threat-mitigation capabilities in distributed network settings [16], [18], [19],[20]. However, most existing research is highly dependent on simulations and simple traffic conditions. It benchmarks that fail to depict the true nature of today's virtualized 5G infrastructure [21],[22], [23].

Contemporary MARL intrusion detection research also shows a significant lack of adversarial robustness testing, typically evaluating learned policies only under normal operating conditions without considering an intelligent adversary that manipulates observations, rewards, or inputs [21]. This particular limitation is especially crucial, as adversarial machine learning attacks target the learned representation and coordination of the MARL algorithm. Recent work has begun to apply cooperative and multi-agent reinforcement learning directly to network intrusion detection [24], [25], including comprehensive reviews of RL-based detection in communication networks [26], and multi-agent security studies that explicitly examine how adversarial or "weaponized" agent actions and anomalous behaviour can undermine coordination in MARL systems [27], [28]. In parallel, work on robustifying reinforcement learning policies against observation perturbations [29] and broader treatments of MARL for cybersecurity applications [30] further motivate the need for the adversarial evaluation undertaken in this study, though none of these directly evaluate cooperative MARL-based intrusion detection under both feature-space and coordination-level attacks within a 5G-oriented distributed security abstraction, which remains the specific gap this paper addresses. While Fast Gradient Sign Method attacks seek to misclassify by manipulating the input representation with adversarial perturbations, Byzantine attacks aim to disrupt distributed learning through observation manipulation, reward manipulation, or disruption of coordination behaviour among cooperative agents [31],[32],[33]. This study addresses this gap by evaluating the adversarial robustness of an intrusion detection system based on MARL, developed for deployment in distributed 5G-oriented security environments. In this research, intrusion detection is formulated as a partially observable multi-agent sequential decision-making problem in which three cooperative agents act on behalf of the Radio Access Network (RAN), the Multi-access Edge Computing (MEC) layer, and the core network. By applying a CTDE-based approach, these agents learn intrusion detection policies while executing their tasks independently. Different from prior studies that focused on the evaluation of MARL architectures only in a benign setting, this research considers the effects of Fast Gradient Sign Method (FGSM) and Byzantine attacks.

This study investigates five research questions covering clean-condition performance, adversarial degradation, coordination fragility, and cross-dataset robustness: RQ1: How effectively does a cooperative MARL intrusion detection framework detect cyber threats under benign conditions across legacy and contemporary intrusion detection datasets? RQ2: How do FGSM perturbations affect the detection performance and stability of MARL-based intrusion detection systems in distributed 5G-oriented environments? RQ3: To what extent do Byzantine poisoning attacks compromise coordination and detection performance within cooperative MARL security architectures? RQ4: How does adversarial robustness behaviour differ between NSL-KDD and UNSW-NB15 datasets under identical MARL evaluation conditions? RQ5: What do the observed adversarial degradation patterns reveal about the resilience limitations of cooperative MARL-based cybersecurity systems?

This research contributes in four ways: it (i) models intrusion detection as a cooperative partially observed multi-agent reinforcement learning problem using distributed 5G-based security abstractions; (ii) experimentally tests adversarial vulnerability to FGSM evasion and Byzantine attacks; (iii) conducts cross-dataset robustness analysis using legacy and modern intrusion detection benchmarks under equal

adversarial conditions; and (iv) shows that cooperation and robustness pose different design challenges for distributed cybersecurity systems.

More broadly, cross-dataset testing in this study reveals discrepancies in intrusion detection effectiveness and robustness between legacy and contemporary datasets, suggesting that a single evaluation metric, such as NSL-KDD, is inadequate for modern network security [22]. The resulting benchmarks of adversarial degradation are expected to inform robustness-based evaluation metrics for future MARL-based cybersecurity approaches [21].

Whereas most previous MARL-based intrusion detection studies measure performance only in normal, non-adversarial settings, this study's primary contribution is not a new MARL algorithm but a systematic assessment of how cooperative intrusion detection policies, evaluated across both legacy and modern benchmark datasets, perform under feature-level and coordination-level adversarial attacks.

The rest of the paper is structured as follows. In Section 2, we provide the research methodology, including the system architecture, data sets, attack realization, and evaluation process. Section 3 describes the experimental results for both normal and adversarial scenarios. Lastly, in Section 4, we conclude the paper by highlighting the implications of adversarial fragility on distributed MARL security systems, as well as the study limitations and future work directions.

## METHODS

This Section follows the steps that were conducted in the research.

### Research Design

This study applies an experimental research approach based on the CRISP-DM (Cross-Industry Standard Process for Data Mining) methodology. The chosen methodology is based on CRISP-DM due to its structure and iterative character as a method of building data mining systems. It includes six stages of the process, which are business understanding, data understanding, data preparation, modelling, evaluation, and deployment [34]. The CRISP-DM methodology aligns well with this study because it can be applied to the process of building reinforcement learning models that undergo iterative training, evaluation, and optimization cycles. The main objective of this study is to design and evaluate a MARL-based Intrusion Detection System (IDS) from the perspective of robustness under attack. To reduce stochastic variation, the preprocessing pipeline, dataset split, model configuration, attack settings, and hyperparameters were kept consistent across all experimental conditions. The reported results should therefore be interpreted as controlled experimental outcomes under a fixed configuration rather than seed-averaged statistical estimates. Future work should extend this evaluation by using multiple independent random seeds, reporting the mean and standard deviation, using confidence intervals, performing exact significance testing, and conducting effect-size analysis to provide deeper insight into the variability in adversarial robustness.

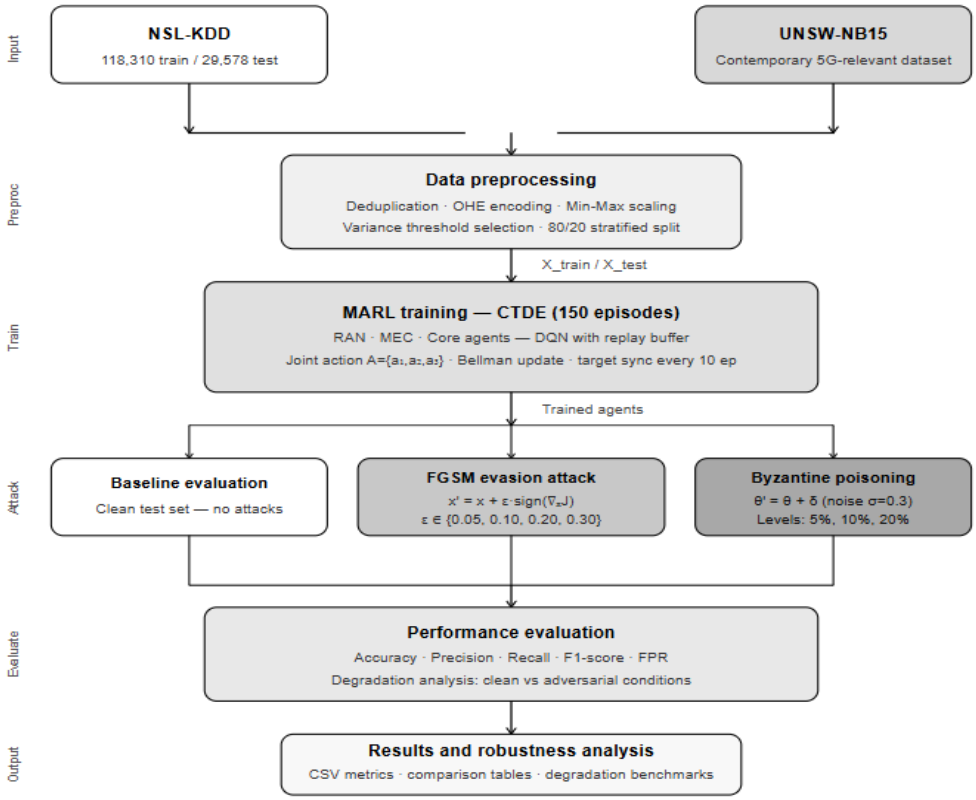


Figure 1. Proposed Centralized Training with Decentralized Execution (CTDE)-based MARL intrusion detection architecture for distributed 5G-oriented security environments, depicting RAN, MEC, and Core agents with joint Q-value updating through the Bellman equation.

**Multi-Agent Decision Formulation**

Intrusion detection within distributed 5G-oriented environments is formulated as a Partially Observable Markov Decision Process (POMDP) represented by the tuple:

$$\mathcal{G} = \langle \mathcal{N}, \mathcal{S}, \{\mathcal{O}_i\}_{i=1}^{N}, \{\mathcal{A}_i\}_{i=1}^{N}, \mathcal{T}, \mathcal{R}, \gamma \rangle \quad (1)$$

where $\mathcal{N}$ = {RAN, MEC, Core} is the set of cooperative agents, $\mathcal{S}$ is the global state space, $\mathcal{O}_i$ are the local observation spaces, $\mathcal{A}_i$ are the agent action spaces, $\mathcal{T}$ refers to the transition function, $\mathcal{R}$ refers to the common team reward, and $\gamma \in (0, 1]$ is the discount factor. The objective for each agent is to learn a local policy $\pi_i(a_i \mid o_i)$, such that:

$$J(\pi) = \mathbb{E}\left[\Sigma_t \gamma^t R_t\right] \quad (2)$$

The global state $s_t \in \mathcal{S}$ represents the entire feature vector obtained from the network traffic data after preprocessing. In a distributed telecom environment, where there cannot be complete visibility across all network layers at once, the agents have partial observations $o_{it} \in \mathcal{O}_i$. The feature vector is divided into three parts that cover abstracted distributed observability across the Radio Access Network (RAN), Multi-access Edge Computing (MEC), and core network layers:

$$o_t^1 = s_t[0 : d_0] \quad (3)$$
$$o_t^2 = s_t[d_0 : d_0 + d_1] \quad (4)$$
$$o_t^3 = s_t[d_0 + d_1 : F] \quad (5)$$

where F is the feature dimensionality and $d_0$, $d_1$, $d_2$ represent roughly equivalent partitions of the feature space. This kind of partitioning does not entail a one-to-one correspondence between individual features

and 5G telemetry variables, but rather mimics decentralized monitoring in partially observable settings. Consequently, the partitioning strategy should be viewed primarily as a distributed observability approximation designed for controlled coordination analysis rather than a semantically exact decomposition of real telecom-layer telemetry streams. Each agent makes its own choice from a binary action set:

$$\mathcal{A}_{\mathrm{i}} = \{\, 0\ (allow), 1\ (flag\ attack) \,\} \tag{6}$$

The final classification result is determined by majority voting among agents, where traffic is considered malicious if at least two agents classify it as such. Majority voting has been chosen because it offers lightweight cooperation without the need for extensive interaction among agents, which might be difficult in low-latency network environments.
The environment evolves in an orderly manner based on the network traffic traces. In each time step, the agents perceive the current state of traffic, take actions, receive a reward, and then move on to the next traffic observation:

$$\begin{gathered} s_{\mathrm{t+1}} = X[t+1], \\ t < T\ s_{\mathrm{t+1}} = X[T], \\ t = T \end{gathered} \tag{7}$$

where X stands for the ordered set of traffic logs, and T indicates the final step of the episode. In such a sequential framework, the dynamic nature of intrusion detection policies in continuously evolving traffic distributions is captured rather than classifying static examples.

The transition s_(t+1)=X[t+1]is exogenous because the next state is the next record in the traffic stream rather than a direct consequence of agent action. The MARL framework is retained because the task involves partial observability, shared-reward learning, majority-vote coordination, and sequential policy refinement across traffic observations. This is therefore a controlled POMDP-style abstraction rather than a fully action-conditioned telecom simulation. All agents share a common reward function defined according to the majority-voted decision ŷ relative to the ground-truth label y:

$$R_{\mathrm{t}} = +1.0\ if\ \hat{y} = 1\ and\ y = 1 \tag{8}$$
$$R_{\mathrm{t}} = -1.0\ if\ \hat{y} = 0\ and\ y = 1 \tag{9}$$
$$R_{\mathrm{t}} = -0.5\ if\ \hat{y} = 1\ and\ y = 0 \tag{10}$$
$$R_{\mathrm{t}} = +0.2\ if\ \hat{y} = 0\ and\ y = 0 \tag{11}$$

The asymmetric reward structure penalizes missed attacks more harshly than false positives, reflecting the operational priority of detection completeness in security-sensitive network environments.

In adversarial settings, the environment's dynamics can be altered by perturbing the feature space during inference (FGSM evasion) or by perturbing observations and rewards to impact coordinated actions (Byzantine poisoning). The perturbations can help determine if the learned policies for cooperation are robust to adversarial manipulations. Although simplified, the proposed formulation provides a controlled experimental abstraction of cooperative sequential intrusion detection under distributed observability conditions rather than a full operational simulation of 5G-oriented infrastructures.

**System Architecture and MARL Model**

The proposed system is based on a Multi-Agent Reinforcement Learning architecture that models the distributed nature of 5G-oriented environments [12]. MARL was selected over single-agent reinforcement learning because 5G networks are inherently distributed across multiple layers, and a single centralized agent cannot realistically capture the localized decision-making requirements of the RAN, edge, and core [12]. More importantly, cooperative MARL enables the study of coordination fragility under adversarial disruption, which cannot be meaningfully investigated with purely centralized intrusion-detection architectures. The system consists of three cooperative agents. The RAN Agent monitors radio-level traffic and detects anomalies at the access layer. The MEC Agent performs edge-level analysis and intermediate decision-making. The Core Agent handles global network decisions and overall threat classification. The three-agent structure abstracts distributed observability across access, edge, and core network layers [21].

The agents operate using a Centralized Training with Decentralized Execution (CTDE) approach. CTDE was chosen because it resolves the fundamental tension between coordination and scalability in MARL systems: during training, agents share information and learn a joint policy to improve coordination, while during execution, each agent acts independently based on local observations, ensuring real-time responsiveness consistent with 5G latency requirements [12]. The MARL model is implemented using a Deep Q-Network (DQN) algorithm, where each agent learns an optimal detection policy through iterative interaction with the environment. DQN was selected over more complex alternatives such as PPO, QMIX, and MADDPG to establish a controlled and interpretable baseline for adversarial robustness evaluation [16].

In this implementation, centralization occurs through a shared team reward and synchronized training episodes, while each agent maintains its own Q-network and replay buffer. Therefore, the architecture is best interpreted as cooperative independent Q-learning with reward-level centralization rather than full CTDE with a joint critic or parameter sharing. This clarification is important because the study focuses on adversarial fragility in cooperative decision-making rather than proposing a new CTDE algorithm.

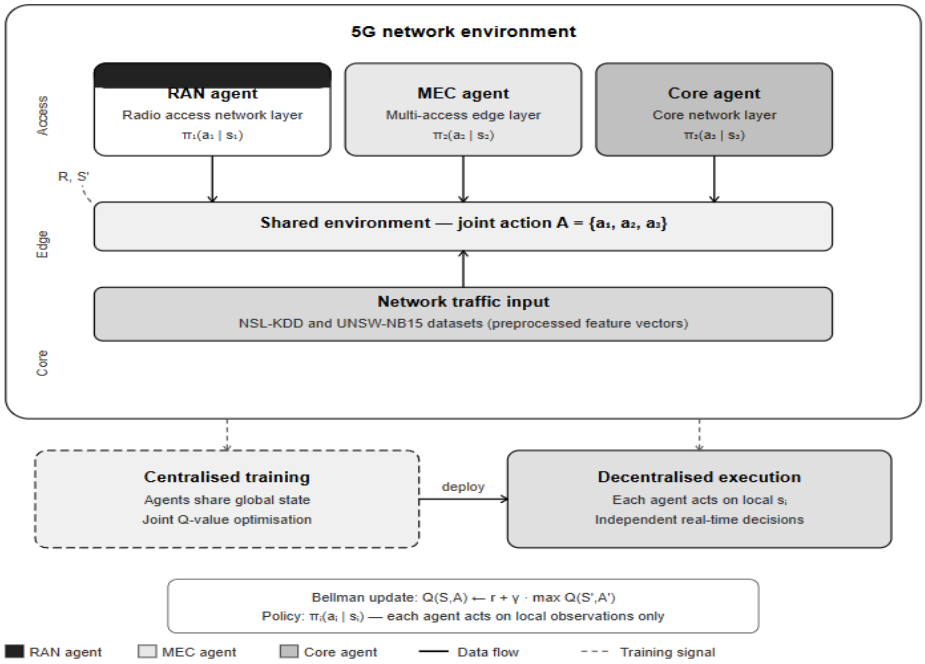


Figure 3. End-to-end training and adversarial evaluation pipeline incorporating FGSM evasion and byzantine poisoning attack injection.

As shown in Figure 3, the training pipeline incorporates adversarial perturbations during evaluation. The FGSM attack is formulated as $x' = x + \varepsilon \cdot sign(\nabla_x J)$, allowing assessment of model robustness under controlled adversarial conditions. The training process followed a cooperative independent DQN procedure over 150 episodes. At each episode, mini-batches of 2,000 training samples were used to update the three agent-specific Q-networks. Agents selected binary actions; the final intrusion decision was obtained via majority voting, and each agent was updated using the shared team reward and the Bellman target. During evaluation, the trained policies were frozen and applied to clean, FGSM-perturbed, and Byzantine-corrupted test conditions.

**Datasets**

Two data sets are used to parameterize the network environment and assess the generalization ability of learned policies across varying traffic profiles. In this regard, the NSL-KDD data set offers an environment with four types of attacks: DoS, Probe, R2L, and U2R, for which comparisons with previous work can be made [22]. However, since NSL-KDD predates the 5G architecture, any policies learned from it might never encounter the variety of attacks in the current virtualized network environment, an aspect specifically tested in this paper using cross-dataset analysis [22].

The UNSW-NB15 dataset provides a more complex and contemporary environment, with nine attack categories, including Fuzzers, Backdoors, Exploits, and Reconnaissance, generated under realistic traffic-simulation conditions [22], [23]. Training agents in such an environment exposes them to more varied state distributions and harder reward landscapes, helping evaluate whether cooperative policies remain effective under greater traffic diversity and class imbalance. Comparing the performance of the same agents across two environments reveals how the complexity of the state space influences results, an aspect difficult to assess in purely supervised learning evaluations [21].

Neither dataset contains native 5G network traffic; both were instead used to configure controlled, distributed security environments that reflect key 5G characteristics, such as decentralized monitoring and collaborative decision-making. The aim is therefore not to replicate real-world 5G behaviour, but to assess MARL adversarial robustness within distributed security scenarios inspired by telecommunications environments.

**Data Preprocessing and Splitting**

The datasets undergo a standardized preprocessing pipeline to ensure compatibility with the MARL model. First, data cleaning involves the removal of duplicate and inconsistent records to prevent bias during training. Second, feature encoding converts categorical features such as protocol type and service flags into numerical form using one-hot encoding, as reinforcement learning state representations require fully numerical inputs [22]. Third, feature scaling normalizes all continuous features to the range [0, 1] using Min-Max scaling. This is particularly important for FGSM attack implementation, where perturbation bounds must be defined within a consistent numerical range to ensure meaningful epsilon values [31]. Fourth, feature selection applies a variance threshold to remove features with near-zero variance, retaining the most informative features to reduce dimensionality and improve training efficiency without sacrificing detection capability.

The data is split into a training and test set at an 80:20 ratio. The former is used only to generate the environment's state space for the agents to learn their detection policies, with a batch size of 2,000 samples per iteration. The latter is kept aside for policy evaluation, both in clean and in adversarial environments, to ensure that the agent has not seen all the environment states seen by the policy during its evaluation phase before [22].

**Model Training and Hyperparameter Configuration**

Hyperparameters are configured as follows and justified individually. A learning rate of 0.001 is used because it is well established as a stable default for Adam optimization in deep reinforcement learning, balancing convergence speed against the risk of overshooting optimal policy values [11]. A discount factor of $\gamma = 0.99$ is selected so that agents weight future rewards heavily, encouraging policies that sustain detection performance across an entire episode rather than maximizing immediate reward on individual records. This is critical in sequential network monitoring, where a short-term action such as allowing ambiguous traffic can influence the system's cumulative detection outcome over subsequent time steps. A batch size of 64 is used to provide stable gradient estimates during experience replay without excessive memory overhead. All reported results were obtained using a fixed training budget of 150 episodes per run, consistently applied across both datasets and all clean and adversarial evaluation conditions. Preliminary experimentation across a broader range of episode counts, from 100 to 200, was used only to confirm that 150 episodes were sufficient to achieve stable policy convergence before this value was fixed for the final experiments.

**Adversarial Attack Implementation**

To evaluate robustness, two types of adversarial attacks are systematically applied to the trained MARL system. The first is the Fast Gradient Sign Method (FGSM) evasion attack. FGSM was selected as a computationally efficient white-box adversarial baseline to systematically assess sensitivity to feature-space perturbations in learned MARL policies, providing a reliable worst-case benchmark for evaluating model vulnerability to input perturbations [31]. FGSM generates adversarial examples by adding perturbations to input features in the direction of the gradient of the loss function with respect to the input, scaled by a perturbation magnitude epsilon. The attack objective was to induce malicious traffic misclassification through gradient-aligned perturbation of normalized input representations. Experiments are conducted on the test dataset at epsilon values of 0.05, 0.10, 0.20, and 0.30 to measure the degradation of detection performance across a range of perturbation intensities. For FGSM, continuous normalised

network traffic features were perturbed according to the adversarial intrusion-detection literature [31]. All perturbed feature values were then clipped to [0,1] to ensure feature validity.

The second attack is Byzantine poisoning. Byzantine attacks are used because they specifically target the multi-agent coordination component of the MARL approach, thereby providing a threat model for distributed learning systems that has not previously been studied in the context of 5G security [32]. The Byzantine attacks are introduced into the simulation by labelling a proportion of agents as Byzantine during the testing phase. Byzantine agents behave untruthfully using two different methods simultaneously. Firstly, Byzantine agents have their observations modified by adding bounded noise to their observations: $o_t^{iadv} = o_t^i + \varepsilon_o$, where $\varepsilon_o \sim U(-0.1, 0.1)$ for all observation variables. Secondly, the reward function of Byzantine agents is distorted by adding noise $\varepsilon_r$ to the value of $R_t$, where $\varepsilon_r \sim U(-0.5, 0.5)$. Thus, Byzantine attacks provide an agent that does not correctly perceive the environment and receives incorrect feedback. Three poisoning intensities were simulated by probabilistically applying Byzantine corruption behaviour during evaluation, corresponding to low-, moderate-, and high-intensity coordination corruption settings, approximating perturbation exposure levels of 5%, 10%, and 20% across sequential interaction steps.

During evaluation, the trained policies are frozen. Therefore, observation corruption is the mechanism that changes Byzantine-agent behaviour, because corrupted observations are passed into fixed Q-networks and may alter selected actions. Reward corruption is retained only as a diagnostic bookkeeping disturbance and does not directly change evaluation-time decisions. Parameter-level corruption is shown conceptually in the attack model but was not implemented in the present experiments.

The Byzantine threat model used in this study was designed to simplify the representation of coordination failures in a distributed learning system. Rather than attempting to capture every possible form of Byzantine behaviour, it focuses on introducing inconsistencies in observations and rewards that can interfere with cooperation between agents while keeping the experiments manageable and reproducible. Other forms of Byzantine attacks, such as adaptive policy manipulation, malicious parameter updates, and coordinated collusion among compromised agents, were not considered in this work and offer valuable opportunities for future research.

**Evaluation Metrics**

The performance evaluation is based on the following five criteria: accuracy (number of correctly classified samples), precision (percentage of samples marked as attacks that were indeed attacks), recall (percentage of actual attacks that were detected correctly), F1-score (harmonic average of precision and recall), and False Positive Rate (FPR) [21]. The robustness test is performed by calculating the absolute degradation between clean and adversarial samples for each criterion.

**Tools and Implementation Environment**

All experiments were implemented in Python using the PyTorch framework for MARL model development and optimization. Supporting libraries such as NumPy, Pandas, and Scikit-learn were used for preprocessing, feature encoding, normalization, and performance evaluation. The Adam optimizer was configured with a learning rate of 0.001, and Min-Max normalization was applied to scale continuous features to the [0, 1] range required for stable adversarial perturbation analysis.
Training and evaluation were conducted in the Google Colab environment using CPU-based execution.

**Baseline Models for Comparative Evaluation**

To evaluate adversarial robustness across different learning paradigms, three baseline models representing classical machine learning, centralised deep learning, and single-agent reinforcement learning were evaluated on identical datasets, with the same preprocessing pipelines and performance metrics.

**RESULT AND DISCUSSION**

This Section presents the performance of the proposed Multi-Agent Reinforcement Learning (MARL)-based intrusion detection system under normal and adversarial conditions. The evaluation is conducted using the NSL-KDD and UNSW-NB15 datasets across three experimental conditions: baseline clean performance, performance under FGSM evasion attacks, and performance under Byzantine poisoning attacks. All results are reported using accuracy, precision, recall, F1-score, and false-positive rate, as defined in the methodology. These baselines are Random Forest (classical machine learning), MLP

(centralized deep learning), and single-agent DQN (reinforcement learning without distributed observability or majority voting).

**Baseline Performance (No Attacks)**

To address RQ1, the proposed MARL framework was evaluated under benign conditions without adversarial interference. As shown in Table 1, the system achieved strong detection performance on both datasets, with substantially higher performance observed on NSL-KDD than on UNSW-NB15.

Table 1. Baseline performance under clean conditions

| Metric | NSL-KDD | UNSW-NB15 |
|---|---|---|
| **Accuracy** | 96.94% | 85.15% |
| **Precision** | 96.95% | 87.77% |
| **Recall (DR)** | 96.94% | 85.15% |
| **F1-Score** | 96.94% | 85.13% |
| **FPR** | 3.38% | 24.50% |

The improved initial performance observed in NSL-KDD likely reflects low traffic diversity and a simple distribution of attacks in this traditional data set. However, the introduction of greater traffic diversity and class imbalance in UNSW-NB15 made cooperative policy learning more challenging. These findings reinforce concerns about the use of simplified datasets for assessing AI-based IDS. The higher false-positive rate observed in the UNSW-NB15 dataset further suggests that the greater diversity in both network traffic and attack types introduced more uncertainty into the cooperative decision-making process learned by the agents. This highlights the importance of testing intrusion detection systems in more complex, realistic network environments rather than relying solely on simpler benchmark datasets.

**Performance Under FGSM Evasion Attack**

To assess the system's robustness to evasion-based manipulations (RQ2), FGSM attacks were conducted with different ε values, with ε = 0.20 chosen for the main analysis due to its balance between perturbation effectiveness and operational plausibility.

Table 2. FGSM attack impact (epsilon = 0.20)

| Metric | NSL-KDD Before | NSL-KDD After | UNSW Before | UNSW After |
|---|---|---|---|---|
| **Accuracy** | 96.94% | 46.80% | 85.15% | 39.89% |
| **Precision** | 96.95% | 45.89% | 87.77% | 39.67% |
| **Recall** | 96.94% | 46.80% | 85.15% | 39.89% |
| **F1-Score** | 96.94% | 45.50% | 85.13% | 34.78% |
| **FPR** | 3.38% | 38.13% | 24.50% | 85.06% |

The UNSW-NB15 post-FGSM false-positive rate was rechecked against the evaluation output before resubmission to ensure that it was not copied from the clean-condition accuracy value. The UNSW-NB15 false positive rate under FGSM (ε=0.20) attack (85.06%) and the UNSW-NB15 baseline accuracy (85.15%) are numerically close but arise from unrelated quantities computed on different subsets of predictions; this proximity is coincidental rather than indicative of a data-reporting error.

Perturbation using the FGSM method resulted in severe degradation across all detection metrics for both data sets, reflecting the high sensitivity of the learned MARL policies to gradient-based feature-space attacks. Even modest perturbations were sufficient to destabilize the learned policy boundaries and substantially increase false-positive behaviour. Notably, this performance degradation was consistent across both legacy and recent data sets, indicating that policy learning was not robust to such feature attacks. It is important to note that FGSM represents a worst-case white-box adversarial scenario, where perturbations are generated using gradient information from the target model itself. As a result, the significant performance degradation observed under FGSM conditions should mainly be interpreted as evidence of the model's sensitivity to adversarial manipulation in the feature space, rather than as a direct reflection of likely attack success rates in real-world telecom environments. The pronounced degradation observed at relatively low epsilon values may further suggest that reinforcement-learning decision boundaries are vulnerable to sequential state perturbations, with minor disruptions introduced early intensifying across subsequent cooperative policy actions.

The full epsilon-level FGSM results show that accuracy does not decline monotonically with increasing epsilon: NSL-KDD accuracy drops sharply from 66.82% at ε=0.05 to 43.58% at ε=0.10, but then partially

recovers to 46.80% at ε=0.20 and 45.51% at ε=0.30. This pattern is consistent with two known properties of gradient-based evasion attacks in bounded-input spaces. First, because all perturbed features are clipped to [0, 1] to preserve feature validity, larger epsilon values increasingly saturate at the clipping boundary rather than translating into proportionally larger effective perturbations, which can flatten or partially reverse the degradation curve at higher ε. Second, FGSM perturbations are computed in a single gradient-sign step, so once ε is large enough to push samples well past the nearest decision boundary, further increases in ε do not necessarily push samples into regions of even lower model confidence, and can in some cases move samples into a different, more ambiguous decision region. We note this pattern was observed in a single representative evaluation run per epsilon level; multi-seed averaging (see Section on future work) would help determine how much of this non-monotonicity is a stable feature-space effect versus run-to-run stochastic variation. These results reinforce broader concerns regarding the fragility of deep representation-learning models in security-critical contexts. In distributed intrusion detection systems, adversarial distortions may not only affect classification accuracy but also destabilize coordinated actions.

**Performance Under Byzantine Poisoning Attack**

To investigate coordination fragility under poisoning conditions (RQ3), Byzantine-compromise scenarios were introduced by corrupting subsets of cooperative agents via observation and reward manipulation.

Table 3. Byzantine attack impact (20% agent compromise)

| Metric | NSL-KDD Before | NSL-KDD After | UNSW Before | UNSW After |
|---|---|---|---|---|
| **Accuracy** | 96.94% | 84.91% | 85.15% | 82.53% |
| **Precision** | 96.95% | 85.49% | 87.77% | 82.90% |
| **Recall** | 96.94% | 84.91% | 85.15% | 82.53% |
| **F1-Score** | 96.94% | 84.80% | 85.13% | 82.59% |
| **FPR** | 3.38% | 8.00% | 24.50% | 19.31% |

As shown in Table 4, Byzantine poisoning resulted in less severe but more persistent degradation compared to FGSM evasion. While the FGSM attack affected feature representation, the Byzantine attack disrupted cooperative coordination behaviour by slowly corrupting local observations and rewards within the compromised agent.

The degradation patterns confirm that cooperative MARL architectures remain susceptible to both input-level and coordination-level adversarial manipulation. Although Byzantine poisoning produced measurable degradation in both datasets, the impact was more pronounced in NSL-KDD than in UNSW-NB15.
The comparatively smaller degradation observed on UNSW-NB15 may reflect differences in traffic diversity and state-space complexity rather than definitive evidence of improved coordination robustness. Nevertheless, because the cooperative framework comprised only three agents, the observed poisoning effects could be attributed to random voting.

The non-monotonic performance degradation observed as Byzantine-compromise levels increased may therefore be linked to stochastic coordination effects within the simplified three-agent majority-voting setup, rather than reflecting consistent robustness behaviour. This limitation should be taken into account when interpreting the poisoning-resilience results obtained using the cooperative architecture in this study. Overall, the findings suggest that cooperative coordination mechanisms alone are insufficient to guarantee resilience against distributed adversarial manipulation within MARL-based cybersecurity systems.

**Cross-Dataset Comparison**

Cross-dataset evaluation was conducted to assess whether adversarial robustness behaviour differed between legacy and contemporary intrusion detection environments (RQ4).

Table 4. Performance comparison across datasets and conditions

| Condition | NSL-KDD Accuracy | UNSW-NB15 Accuracy | Gap |
|---|---|---|---|
| **Baseline** | 96.94% | 85.15% | 11.79pp |
| **FGSM Attack** | 46.80% | 39.89% | 6.91pp |
| **Byzantine 20%** | 84.91% | 82.53% | 2.38pp |

Cross-dataset evaluation demonstrated that stronger clean-condition performance did not necessarily correspond to improved adversarial robustness. Although NSL-KDD achieved a much higher baseline accuracy, both datasets suffered significant performance degradation when subjected to adversarial

perturbations, especially under FGSM attacks. These findings reinforce concerns regarding the continued reliance on legacy benchmark datasets when evaluating the adversarial robustness of AI-based cybersecurity systems.

### 3.5 Robustness Analysis

The degradation patterns shown in Table 5 indicate that feature-space perturbation produced substantially greater disruption than coordination-level poisoning across both datasets. FGSM attacks consistently destabilized learned policy boundaries and significantly increased false-positive behaviour, while Byzantine attacks generated comparatively smaller yet persistent degradation by corrupting local observations and reward dynamics. Collectively, these findings indicate that cooperative intelligence alone does not guarantee adversarial robustness within distributed MARL-based intrusion detection systems.

Table 5. Performance degradation under adversarial conditions (%)

| Attack Type | NSL-KDD Drop | UNSW-NB15 Drop |
|---|---|---|
| **FGSM (ε = 0.20)** | 50.14pp | 45.26pp |
| **Byzantine (20%)** | 12.03pp | 2.62pp |

The degradation analysis was conducted descriptively by comparing clean-condition performance with adversarial-condition performance under the same preprocessing pipeline, model configuration, and evaluation metrics. Because the reported tables do not provide seed-averaged estimates, the results are interpreted as controlled experimental evidence of robustness degradation rather than formal statistical inference. Future work should repeat each condition across multiple random seeds and report the mean, standard deviation, confidence intervals, exact p-values, and effect sizes.

To address this more directly: the FGSM results in Appendix Table A1 show that accuracy drops sharply between epsilon = 0.05 and 0.10, then partially recovers at 0.20 before flattening at 0.30, rather than declining monotonically as epsilon increases. Likewise, the Byzantine results show that accuracy for 20% compromised agents exceeds that for 5% and 10% on both datasets, which is counterintuitive if a higher proportion of compromised agents were expected to cause a proportionally larger degradation. We believe two factors plausibly contribute to both irregularities. First, majority voting only changes the system's output when corruption flips at least two of the three agents' votes; because each agent's response to a given perturbation is not identical, the relationship between epsilon (or percentage of compromised agents) and the majority-vote outcome is not guaranteed to be monotonic even if each individual agent degrades smoothly. Second, because each condition in the current results reflects a single evaluation run rather than an average over multiple seeds, some portion of the apparent non-monotonicity may simply be run-to-run variance rather than a true property of the model; this is precisely why the reviewer's request for multiple seeds and confidence intervals (addressed above) is important; we would expect a seed-averaged version of these tables to show a smoother, though not necessarily strictly monotonic, degradation trend.

**Practical Implications for Distributed Cybersecurity**

These findings highlight several important considerations for the use of AI-powered cybersecurity systems in distributed telecom networks. Although cooperative learning architectures improve threat detection under benign conditions, these improvements do not inherently confer adversarial resilience; therefore, robustness evaluation should be a core part of assessing MARL-based intrusion detection systems. Future solutions may need stronger safeguards, including adversarial training, trust-based coordination, Byzantine-resilient consensus, and communication validation to identify manipulated activity.

**Threats to Validity**

Several limitations should be considered when interpreting the findings of this study. First, in terms of internal validity, the experimental environment relied on benchmark intrusion detection datasets that do not fully represent modern virtualized 5G infrastructures characterized by network slicing, dynamic orchestration, O-RAN telemetry, and real-time distributed signalling. The partitioning of traffic features into RAN, MEC, and Core observation spaces represents an abstracted approximation of decentralized observability rather than a direct mapping to operational telecom telemetry streams.

With respect to external validity, the findings may not directly generalize to production-scale 5G deployments because the experiments used a simplified three-agent architecture operating under simulated adversarial conditions rather than in adaptive real-world threat environments. Furthermore, the adversarial

attacks considered focused specifically on FGSM perturbation and simplified Byzantine coordination corruption, excluding more advanced adaptive or stealth-oriented attack strategies. Finally, because the reported results are not seed-averaged estimates, a broader, variance-sensitive robustness evaluation involving confidence intervals, larger multi-seed experimentation, and additional reinforcement learning baselines would further strengthen the statistical validity. Preprocessing, dataset splits, attack settings, and hyperparameters were standardized across all experiments to support reproducibility.

## 4. CONCLUSION

This study evaluated the adversarial robustness of a cooperative MARL-based intrusion detection system within a distributed 5G-oriented security abstraction. The results showed strong clean-condition performance but substantial vulnerability under adversarial conditions, especially FGSM feature-space perturbation. Byzantine poisoning caused smaller but persistent degradation, demonstrating that cooperative decision-making does not automatically provide adversarial resilience. The findings further show that high clean-condition accuracy does not necessarily imply robustness under attack. Future work should evaluate adversarially trained MARL models, Byzantine-resilient coordination mechanisms, trust-aware learning, and realistic 5G/O-RAN telemetry environments.

## AI Assistance Declaration

Generative artificial intelligence tools, including ChatGPT-5.5 (OpenAI), were used to support language editing, structural refinement, and conceptual figure refinement during the preparation of this manuscript. All AI-assisted outputs were subsequently reviewed, revised, and validated by the authors to ensure accuracy, originality, scientific integrity, and adherence to responsible AI-assisted research practices.

## Data Availability Statement

Both datasets used in this research are publicly available. NSL-KDD is available on the University of New Brunswick Canadian Institute for Cybersecurity website at https://www.unb.ca/cic/datasets/nsl.html. UNSW-NB15 is available on the Australian Centre for Cyber Security website at https://research.unsw.edu.au/projects/unsw-nb15-dataset. No proprietary datasets have been used in this research.

## REFERENCES


[1] N. Gupta, S. Sharma, P. K. Juneja, and U. Garg, "SDNFV 5G-IoT: A Framework for the Next Generation 5G enabled IoT," in 2020 International Conference on Advances in Computing, Communication & Materials (ICACCM), 2020, pp. 289–294, doi: 10.1109/ICACCM50413.2020.9213047.

[2] Z. Zhang et al., "6G Wireless Networks: Vision, Requirements, Architecture, and Key Technologies," IEEE Veh. Technol. Mag., 2020, doi: 10.1109/MVT.2019.2952109.

[3] N. Panwar and S. Sharma, "5G Enabled Internet of Things: Security and Privacy Issues," IEEE Sens. J., 2021, doi: 10.1109/JSEN.2021.3050832.

[4] S. M. Yassin, D. Batran, A. Al Harbi, and M. Z. Khan, "A Survey on IoT Applications in Health Care and Challenges," 2021, doi: 10.1007/978-981-16-3246-4_31.

[5] C. Iwendi, S. U. Rehman, A. R. Javed, S. Khan, and G. Srivastava, "Sustainable Security for the Internet of Things Using Artificial Intelligence Architectures," ACM Trans. Internet Technol., vol. 21, no. 3, Jun. 2021, doi: 10.1145/3448614.

[6] GSMA, "The Mobile Economy 2023," GSM Assoc., 2023.

[7] S. Chishakwe, N. Moyo, B. M. Ndlovu, and S. Dube, "Intrusion Detection System for IoT environments using Machine Learning Techniques," 2022 1st Zimbabwe Conf. Inf. Commun. Technol. ZCICT 2022, pp. 1–7, 2022, doi: 10.1109/ZCICT55726.2022.10045992.

[8] S. Alghamdi, "Security Challenges in 5G Networks: A Survey," Sensors, vol. 21, no. 24, 2021.

[9] A. Maramba and B. Ndlovu, "AI-Driven Threat and Incident Detection in Healthcare Cybersecurity : A Stacked Ensemble Model," 2025 4th Zimbabwe Conf. Inf. Commun. Technol., no. November, pp. 1–9, 2025, doi: 10.1109/ZCICT67553.2025.11602039.

[10] S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach. Pearson, 2020.

[11] R. S. Sutton and A. G. Barto, Reinforcement Learning: An Introduction. MIT Press, 2018.

[12] R. Lowe, J. Harb, A. Tamar, P. Abbeel, and I. Mordatch, "Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments," Adv. Neural Inf. Process. Syst., vol. 30, 2017, doi: 10.48550/arXiv.1706.02275.

[13] M. G. S. Murshed, C. Murphy, D. Hou, N. Khan, G. Ananthanarayanan, and F. Hussain, "Machine

Learning at the Network Edge: A Survey," ACM Comput. Surv., vol. 54, no. 8, Oct. 2021, doi: 10.1145/3469029.
[14] P. F. Saura, J. M. Bernabé Murcia, E. G. de la Calera Molina, A. Molina Zarca, J. Bernal Bernabé, and A. F. Skarmeta Gómez, "Federated Network Intelligence Orchestration for Scalable and Automated FL-Based Anomaly Detection in B5G Networks," Comput. Mater. Contin., vol. 80, no. 1, pp. 163–193, 2024, doi: https://doi.org/10.32604/cmc.2024.051307.
[15] R. Moreira et al., "An intelligent native network slicing security architecture empowered by federated learning," Futur. Gener. Comput. Syst., vol. 163, p. 107537, 2025, doi: https://doi.org/10.1016/j.future.2024.107537.
[16] J. Á. Fernández-Carrasco, L. Segurola-Gil, F. Zola, and R. Orduna-Urrutia, "Security and 5G: Attack mitigation using Reinforcement Learning in SDN networks," in 2022 IEEE Future Networks World Forum (FNWF), 2022, pp. 622–627, doi: 10.1109/FNWF55208.2022.00114.
[17] D. K. Dake, "DDoS and Flash Event Detection in Higher Bandwidth SDN-IoT using Multiagent Reinforcement Learning," pp. 16–20, 2021, doi: 10.1109/ICCMA53594.2021.00011.
[18] S. Wijethilaka and M. Liyanage, "Survey on Network Slicing for Internet of Things Realization in 5G Networks," IEEE Commun. Surv. Tutorials, vol. 23, no. 2, pp. 957–994, 2021, doi: 10.1109/COMST.2021.3067807.
[19] J. Zhang et al., "The 4th Artificial Intelligence of Things (AIoT) Workshop," in Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, 2021, pp. 4179–4180, doi: 10.1145/3447548.3469464.
[20] F. M. Dandure and B. Ndlovu, "Federated Learning for Privacy-Preserving Sentiment Analysis in Distributed Electronic Health Record Environments : A Systematic Literature Review," vol. 8, no. 2, pp. 1812–1842, 2026, doi: 10.63158/journalisi.v8i2.1546.
[21] K. L. Chizengwe and B. Ndlovu, "A Systematic Review of Agentic AI for Threat Detection and Mitigation in 5G Networks," J. Inf. Syst. Informatics, vol. 8, no. 1, pp. 315–354, 2026, doi: 10.63158/journalisi.v8i1.1382.
[22] M. Tavallaee, E. Bagheri, W. Lu, and A. Ghorbani, "A Detailed Analysis of the KDD CUP 99 Data Set," 2009, doi: 10.1109/CISDA.2009.5356528.
[23] N. Moustafa and J. Slay, "UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems," in 2015 Military Communications and Information Systems Conference (MilCIS), 2015, vol. 100, pp. 1–6, doi: 10.1109/MilCIS.2015.7348942.
[24] K. Ren, Y. Zeng, Y. Zhong, S. Sheng, and Y. Zhang, "MAFSIDS: a reinforcement learning-based intrusion detection model for multi-agent feature selection networks," J. Big Data, vol. 10, no. 1, p. 137, 2023.
[25] A. Tellache, A. Mokhtari, A. A. Korba, and Y. Ghamri-Doudane, "Multi-agent reinforcement learning-based network intrusion detection system," in NOMS 2024 - 2024 IEEE Network Operations and Management Symposium, 2024, pp. 1–9.
[26] H. Kheddar, D. W. Dawoud, A. I. Awad, Y. Himeur, and M. K. Khan, "Reinforcement-learning-based intrusion detection in communication networks: A review," IEEE Commun. Surv. Tutorials, 2024.
[27] T. Liu, J. McCalmon, M. A. Rahman, C. Lischke, T. Halabi, and S. Alqahtani, "Weaponizing actions in multi-agent reinforcement learning: Theoretical and empirical study on security and robustness," in International Conference on Principles and Practice of Multi-Agent Systems, 2022, pp. 347–363.
[28] C. Lischke, T. Liu, J. McCalmon, M. A. Rahman, T. Halabi, and S. Alqahtani, "LSTM-based anomalous behavior detection in multiagent reinforcement learning," in 2022 IEEE International Conference on Cyber Security and Resilience (CSR), 2022, pp. 16–21.
[29] B. Nie, J. Ji, Y. Fu, and Y. Gao, "Improve robustness of reinforcement learning against observation perturbations via L-infinity Lipschitz policy networks," in Proceedings of the AAAI Conference on Artificial Intelligence, 2024, vol. 38, no. 13, pp. 14457–14465.
[30] C. R. Landolt, C. Würsch, R. Meier, A. Mermoud, and J. Jang-Jaccard, "Multi-Agent Reinforcement Learning in Cybersecurity: From Fundamentals to Applications," arXiv Prepr. arXiv2505.19837, 2025.
[31] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," Proc. Int. Conf. Learn. Represent., pp. 1–11, 2015.
[32] M. Fang, X. Cao, J. Jia, and N. Gong, "Local Model Poisoning Attacks to Byzantine-Robust Federated Learning," in 29th USENIX Security Symposium, 2020, pp. 1623–1640.
[33] A. Dube, B. Mpande, F. Dzehonye, and T. Chazuza, "Zero Trust Architecture : Redefining

Security," pp. 26–39, 2026, doi: 10.1007/978-3-032-20533-9.
[34] C. Schröer, F. Kruse, J. Marx, F. Kruse, and J. Marx, "A Systematic Literature Review on Applying the CRISP-DM Process Model," Procedia Comput. Sci., vol. 181, no. 2019, pp. 526–534, 2021, doi: 10.1016/j.procs.2021.01.199.