IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Cyber Battlefield: Adaptive Offensive and Defensive AI using Reinforcement Learning

[1]Suraj Salunke, [2]Rahul Thube, [3]Shrikesh Gaikwad, [4]Rushikesh Kale

BE Scholars,

[1]Department of Artificial Intelligence & Data Science,

[1]Marathwada Mitra Madal's College of Engineering, Pune, India

Abstract: Cyberattacks are on the rise, driven by the rapid growth of automated tools and increasingly sophisticated attack strategies. Traditional security systems often struggle to keep up with these constantly evolving threats. To address this challenge, our research introduces a Reinforcement Learning-based Cyber Battlefield that realistically simulates both attack and defense scenarios using autonomous AI agents.

In this system, there are two main players: an Offensive AI (Red Team) and a Defensive AI (Blue Team). The Red Team's job is to find weaknesses in network services and attempt to exploit them. Meanwhile, the Blue Team monitors the system, looks for suspicious activity, and responds with defensive measures in real time. This setup creates an ongoing competition, encouraging both agents to adapt and improve through a system of rewards and penalties.

As the simulations progress, the attacking agent learns which tactics are most successful, while the defending agent gets better at spotting and responding to threats. We built this framework using Python, Django, Metasploit, reinforcement learning strategies, and advanced anomaly detection algorithms to create a training environment that closely mirrors real-world cybersecurity challenges.Our experiments showed that, over multiple simulation rounds, the system improved its ability to detect attacks and respond quickly. The platform also adapted to new attack patterns as the training continued. Ultimately, this project offers a scalable, controlled environment for cybersecurity experimentation, autonomous defense research, and AI-driven analysis of emerging threats.

Index Terms **- Cybersecurity, Reinforcement Learning, Offensive AI, Defensive AI, Metasploit, Autonomous Agents.**

I. INTRODUCTION

With the increasing dependence on digital systems, cybersecurity has become one of the most important concerns in modern computing environments.Organizations now contend with a wide range of cyberattacks—like phishing, malware, unauthorized access, denial-of-service, and the exploitation of software vulnerabilities. Traditional defenses still depend heavily on static, rule-based methods, which often can't keep pace with clever or previously unknown threats.

To tackle these evolving risks, researchers are turning to smarter solutions powered by Artificial Intelligence (AI) and Machine Learning (ML). Reinforcement Learning stands out because it enables systems to learn from experience, getting better at decision-making over time through a process of rewards and penalties.

In our project, we introduce a simulated Cyber Battlefield where two AI agents go head-to-head. The Red Team AI acts as the attacker: it tries to uncover weaknesses, scan for open services, and launch exploit attempts. In contrast, the Blue Team AI plays defense—monitoring system logs, spotting unusual behavior, and taking action to neutralize threats.

Unlike traditional setups where offense and defense operate separately, this platform lets both sides evolve together as they interact continuously. We've also incorporated tools like Metasploit for realistic exploitation testing, alongside intelligent monitoring modules for analyzing defensive strategies. By bringing offensive and defensive learning into a single, interactive framework, this system aims to mirror real-world cyber battles in a safe, controlled environment.

II. SYSTEM ARCHITECTURE

The Cyber Battlefield framework is built with a layered, modular architecture that allows different cybersecurity operations to function independently while still working together seamlessly. This design supports everything from launching cyberattacks and monitoring traffic to analyzing threats, applying reinforcement learning, and visualizing results in real time—all within a controlled environment. Rather than depending on a single central component, the system spreads tasks across multiple modules, making it easier to scale, maintain, and upgrade as needed.

Our goal with this architecture was to create realistic cyber warfare simulations, where offensive and defensive agents interact continuously. Each layer plays a distinct role in the simulation, starting with reconnaissance and vulnerability discovery, and extending all the way to intelligent response and policy adaptation. The modular approach also makes it straightforward for researchers to add new security tools or AI models without having to overhaul the entire system.

A. **Host and Target Environment**

At the core of the simulation is the Host and Target Environment—the digital battlefield where both AI agents operate. This layer includes virtual machines, web apps, APIs, databases, and simulated network services, all designed to mimic real-world enterprise networks. We intentionally configure a variety of vulnerable services here, so the Red Team AI can conduct reconnaissance and exploitation attempts safely.

To increase realism, target systems are spread across multiple simulated subnets, creating a complex network topology. Exposed services—like HTTP servers, SSH ports, databases, and application interfaces—allow for authentic attack and defense scenarios. The environment constantly generates network traffic and system logs, which the monitoring modules analyze to detect and respond to threats.

A crucial feature of this layer is its strong isolation and sandboxing. All attack and defense actions are kept strictly within this experimental environment, ensuring that no harmful activity escapes into the real world. Thanks to virtualization, researchers can run repeated simulations with different setups and security configurations, making it a flexible and safe tool for cybersecurity experimentation.

B. **Analysis and Detection Layer**

The Analysis and Detection Layer serves as the platform's watchful eye, gathering detailed information about system behavior, network traffic, and potential vulnerabilities. This module acts as the main observation point, supplying critical data to both the attacker and defender AI agents.

The reconnaissance component kicks things off by scanning for open ports, active services, operating systems, and weaknesses in the target environment. With tools like Nmap and custom Python scripts, this process is automated and efficient. The findings are organized into structured datasets, making them ready for machine learning analysis later on.

Beyond just finding vulnerabilities, the layer keeps a close eye on all network activity. It tracks system logs, failed login attempts, unusual network packets, suspicious requests, and abnormal traffic patterns in real time. The goal is to spot signs of compromise before a major attack can take place.

For better accuracy, the system blends signature-based monitoring (which quickly catches known threats) with anomaly-based analysis (which looks for odd behavior that could signal new or evolving attacks).

C. **Offensive AI Module**

The Offensive AI module is the Red Team's brain within the Cyber Battlefield. Its main goal is to find weaknesses, map out exploitation paths, and launch attacks on the target systems. Unlike traditional attacks that follow fixed scripts, this AI uses Reinforcement Learning to get smarter over time, dynamically improving its strategies with each simulation.

The attack cycle starts with reconnaissance, as the AI gathers intel on active services and exposed vulnerabilities. Using this data, it considers different attack methods—like SQL Injection, brute-force attempts, payload delivery, and privilege escalation. Every choice the agent makes is shaped by its past successes and failures, which are stored in its learning model.

Reinforcement learning drives its evolution: successful attacks earn rewards and push the agent towards similar tactics in the future, while failures or blocked attempts result in penalties, prompting it to try new approaches. The Red Team AI is also designed to adapt—when the Blue Team catches on to a particular attack, the offensive agent switches things up, seeking stealthier, harder-to-detect strategies. This adaptability leads to more realistic, challenging cyber battles.

The module also keeps thorough exploit logs, recording timestamps, targeted services, payload details, success rates, and system responses. These records are invaluable for tracking attack efficiency and understanding how the AI learns adversarial behaviors.

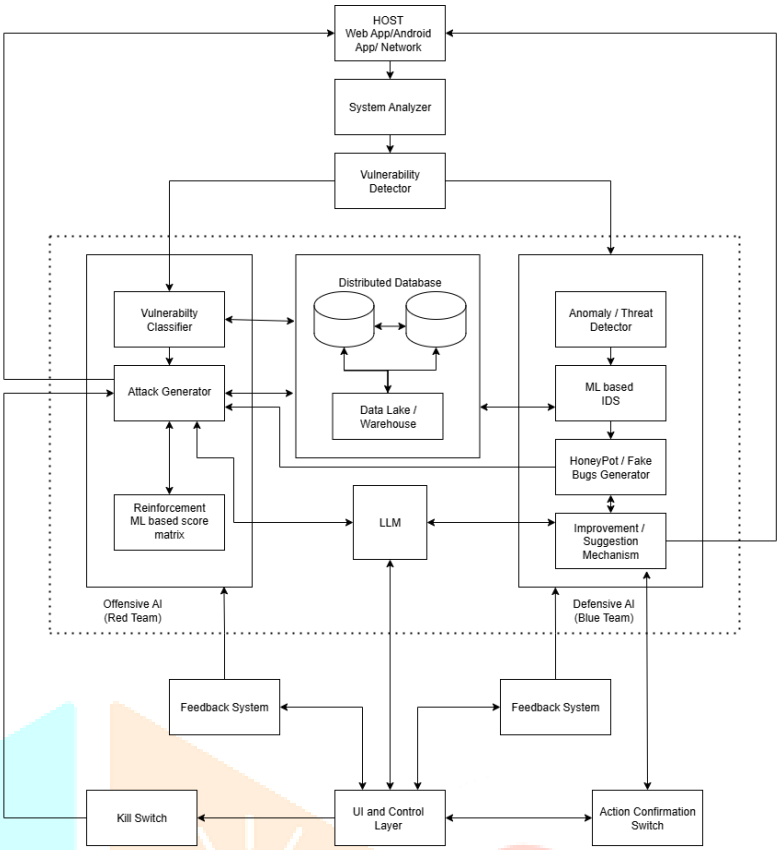

Fig. 1. Internal Workflow of Attack Detection and Monitoring Process

D. **Defensive AI Module**

The Defensive AI module serves as the Blue Team, taking charge of monitoring, detecting, preventing, and mitigating malicious activities. It keeps a constant watch over network traffic and system behaviors, actively looking for attack attempts generated by the Red Team AI.

This module harnesses machine learning and anomaly detection to classify suspicious activities by severity. It pays special attention to warning signs like traffic spikes, repeated failed logins, unauthorized scans, malformed requests, and known exploit signatures. When a threat is identified, the defensive engine chooses a response—this could mean blocking IP addresses, applying firewall rules, limiting suspicious traffic, alerting administrators, or even redirecting attackers to honeypots to waste their resources.

Reinforcement Learning is integrated into the defensive strategy as well. Each successful defense increases the reward score, encouraging the model to repeat effective tactics. If a response fails, the score drops, prompting the AI to try alternative measures. Over time, this helps the Blue Team AI learn which defenses work best against different attack patterns.

A key challenge for the defensive module is minimizing false positives, as too much blocking can disrupt legitimate users. The system continuously tunes its policies to strike a balance between strong security and accessibility, refining its decisions through repeated simulations.

E. **Central Intelligence and Storage Layer**

At the heart of the platform is the Central Intelligence Layer, which acts as the system's coordinator. It manages communication between modules, stores historical data, and supports both AI agents in their decision-making.

A centralized data repository logs everything—scans, exploit attempts, anomalies, defensive actions, and reinforcement learning rewards. This serves as the system's episodic memory, helping agents refine their strategies based on past outcomes.

The intelligence layer also aggregates and preprocesses logs, turning raw data from different components into structured formats ready for machine learning. Sometimes, lightweight language processing is used to summarize security events, making it easier for researchers and administrators to interpret results. After each simulation, policy updates are managed here: as the Red and Blue Teams finish their adversarial rounds, the latest strategies are stored and shared back with the agents for future learning.

The storage setup is designed for scalability, so it can efficiently handle larger datasets and more complex simulations without performance issues.

F. Control and Monitoring Layer

The Control and Monitoring Layer gives human operators a clear view of everything happening during simulations. It features a dashboard, reporting tools, administrative controls, and built-in safety mechanisms.Through the dashboard, users see real-time details like the number of scans, detected vulnerabilities, active alerts, blocked IPs, ongoing attacks, and simulation stats. Graphs and activity feeds make it easy to track the behavior of both AI agents as they train and evolve.

Admins can tweak simulation settings—like target IP ranges, the number of attack rounds, response thresholds, and defensive policies—directly from the control panel. The system also keeps a history of past simulations, so users can review and compare performance across experiments.To ensure everything runs ethically and safely, the platform includes protective features like emergency shutdown controls and strict network isolation. These safeguards guarantee that all simulated attacks stay within the intended research environment and don't spill over into external systems.

III.METHODOLOGY

The Cyber Battlefield platform uses a dynamic, competitive learning model where offensive and defensive AI agents continuously interact within a simulated environment. Rather than sticking to rigid rules, both AIs learn and adapt their strategies over time by analyzing the results of each simulation round. This approach creates a more realistic cybersecurity simulation, reflecting the ever-changing landscape of real-world cyber threats, which often outpace traditional, static security models.

The system's workflow is broken down into key stages: reconnaissance (gathering information), attack execution, anomaly monitoring (watching for unusual behavior), response generation, feedback collection, and policy updates. In each simulation cycle, the AI agents gather new data from their environment and refine their strategies using reinforcement learning—essentially learning by trial and error, guided by rewards for success and penalties for failure. As the agents repeatedly challenge each other, they both evolve: attackers develop more complex techniques, while defenders become increasingly adaptive.

This methodology does more than just simulate cyberattacks; it's also designed to study how autonomous systems learn and adapt in a competitive setting. The continuous feedback loop allows the platform to observe both how attackers change tactics after being blocked and how defenders adjust to new threats.

A. Attack Cycle

The attack cycle describes the Red Team AI's workflow during offensive operations. It starts with reconnaissance, as the agent collects details about the target—such as active ports, running services, exposed APIs, operating systems, and potential vulnerabilities. This phase is critical, since the quality of information gathered directly impacts the success of later attacks.

Automated scanning tools and custom scripts help the agent enumerate (list) system details across the target environment. Discovered ports and services are checked against vulnerability databases to pinpoint possible attack vectors (the routes or methods used for attacks). Once vulnerabilities are found, the AI evaluates multiple strategies—like SQL Injection, brute-force logins, remote code execution, privilege escalation, session hijacking, or denial-of-service—choosing the best fit based on its learning model.

Every action and the corresponding response from the target system are logged. Reinforcement learning drives the AI's decision-making: a successful attack earns a reward, while failed or blocked attempts incur penalties. Over many simulations, the Red Team learns which attack patterns are most effective in different scenarios.

Crucially, the offensive AI adapts its behavior. If an attack fails, it doesn't mindlessly repeat the same method. Instead, it tweaks its timing, payloads, or targets to try and bypass defenses—just like real-world hackers. The attack cycle continues until set objectives are reached, the attack is blocked, or the simulation ends. All activity is recorded to further refine the AI's strategies and to support future research.

B. Defense Cycle

The defense cycle lays out how the Blue Team AI detects, analyzes, and counters malicious activity from the Red Team. This process runs nonstop throughout each simulation, with the main goal of protecting the system's integrity while minimizing the potential damage from cyber threats. It all begins with continuous traffic monitoring and log analysis: the system watches network packets, authentication logs, API requests, process activity, and system events in real time. Monitoring modules capture this data, which is then analyzed for unusual patterns or suspicious behavior.

Machine learning and anomaly detection techniques help classify these activities by their risk level. The system looks for warning signs like repeated failed logins, strange request patterns, unauthorized scans, exploit signatures, or suspicious outbound traffic—comparing current activity against historical baselines to improve accuracy. When a threat is spotted, the defensive engine picks the best mitigation strategy based on the attack's severity. Responses might include blocking an IP address, throttling traffic, sending out security alerts, isolating vulnerable services, or luring attackers into honeypots (decoy systems to trap malicious actors).

Reinforcement learning helps the Blue Team get better over time. If a defense strategy works, it earns a reward; if an attack slips through, the score drops. This feedback loop lets the defensive AI gradually refine its responses for optimal effectiveness.One major challenge is reducing false positives—mistakenly flagging safe activity as dangerous. Excessive blocking can disrupt regular operations, so the Blue Team works to balance strong security with system stability, constantly fine-tuning its decision-making.

The defense and attack cycles run side-by-side, ensuring real-time interaction between threat generation and mitigation.

C. Co-Evolution Process

The co-evolution process sets this Cyber Battlefield apart from traditional simulations. Instead of keeping attack and defense separate, both AI agents evolve together through ongoing adversarial interaction.

The simulation breaks down into multiple rounds, or episodes. In each episode, the Red Team launches attacks while the Blue Team works to detect and stop them. At the end of every round, both agents review what happened and update their learning policies.With each new episode, both agents get smarter. The Red Team hones stealthier and more effective attack strategies, while the Blue Team sharpens its detection and mitigation skills. This back-and-forth creates an artificial “arms race”—mirroring what happens in real-world cybersecurity, where attackers and defenders constantly try to outsmart each other.

The co-evolution method enables researchers to observe long-term learning patterns, such as how quickly defenses adapt to new threats or how attackers switch tactics after failures. These insights are invaluable for understanding the dynamics of adaptive cyber warfare.Another key benefit is improved system robustness. Since the agents are always adapting to changing scenarios, they learn to generalize their tactics rather than memorizing fixed patterns. This means they’re better prepared to handle new, unseen threats.

All policy updates from this process are stored in the platform’s central intelligence layer, preserving learning history for future research and experimentation.

In summary, the co-evolution methodology transforms the Cyber Battlefield into a smart, ever-evolving research platform that models realistic adversarial behavior and drives advanced cybersecurity innovation.

IV. Experimental Results and Visualisations

The following figures represent the output of the Cyber Battlefield platform during the experimental rounds. These visualizations demonstrate the real-time interaction between the Red and Blue AI agents.

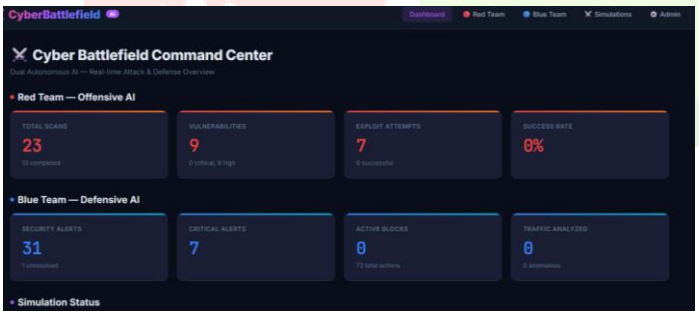

Fig. 2. CyberBattlefield Dashboard - Security Activities

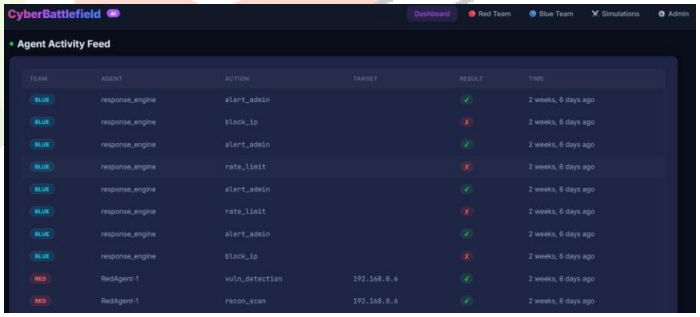

Fig. 3. Activity Tracking of Offensive and Defensive Agents

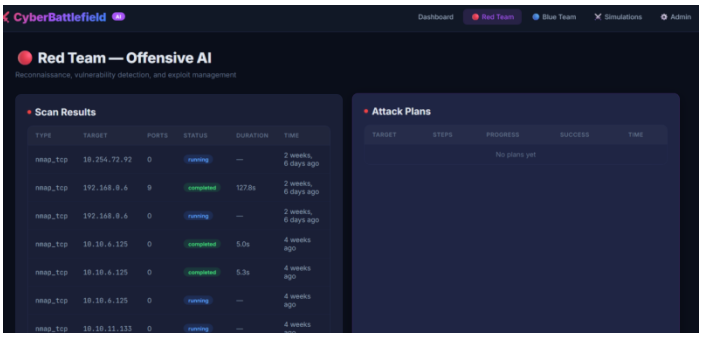

Fig. 4. Network Results Generated by the RedTeam Module

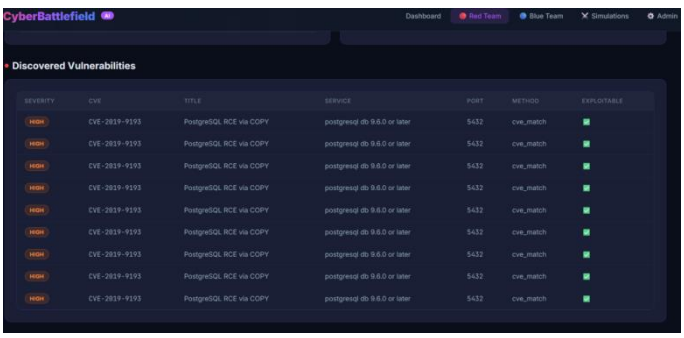

Fig. 5. Vulnerabilities Collected During Security Analysis.

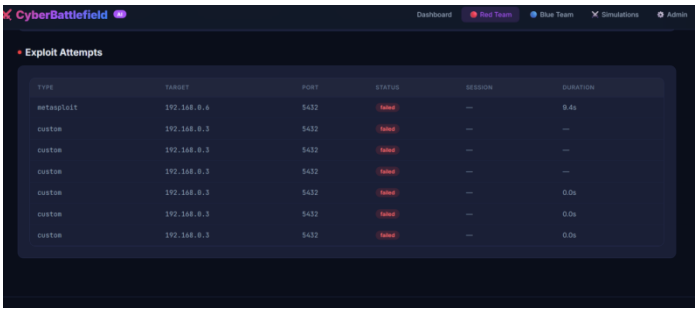

Fig. 6. Recorded Exploitation Attempts

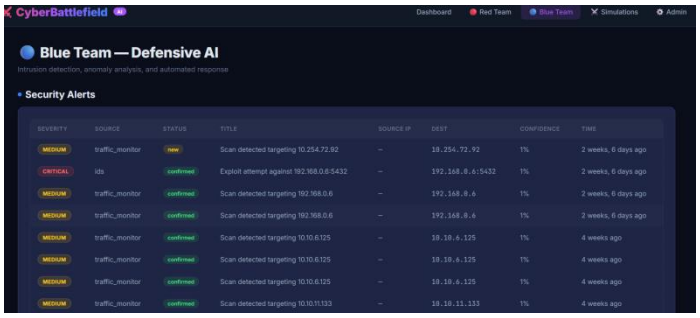

Fig. 7. Threat Notifications Generated by the Defensive System.

V. LITERATURE REVIEW

Recent advancements in artificial intelligence and cybersecurity have fundamentally changed how modern security systems are designed, analyzed, and defended. Researchers from various fields have explored using machine learning to improve threat detection, automate defenses, and strengthen digital infrastructure against ever-evolving attacks. However, these same technologies also introduce new risks, as intelligent systems can be misused to automate sophisticated cyberattacks.

Ahi and Valizadeh [1] examined the influence of Large Language Models (LLMs) in cybersecurity. They found that LLMs can help analysts understand threats, automate report writing, and improve monitoring. However, they also warned that these models can be exploited to generate convincing phishing emails, malicious scripts, or social engineering attacks. Their study highlights the need for strong ethical restrictions and safety features when integrating generative AI into security environments.

Liguori et al. [3] explored how generative AI models could be used offensively. Using natural language prompts, these models can generate exploitation code and attack scripts—raising serious concerns about unchecked AI use in cybersecurity. Their work stresses the importance of building safeguards into future AI-driven security tools to prevent abuse, while still supporting legitimate research and defense.

On the defensive front, Fu [2] introduced a machine learning-based intrusion detection framework that combines Deep Neural Networks with ensemble learning (using multiple models together). This approach showed higher accuracy in detecting malicious traffic than traditional systems and reduced classification errors, making it more reliable for complex networks.

El Rai and Darseesh [4] contributed by applying Graph Attention Networks—AI that focuses on relationships between devices—for cyber-physical system security. By analyzing connections instead of treating each device separately, their approach was better at uncovering complex attack patterns, especially in environments like drone networks and distributed systems.

Mishra et al. [6] tackled transparency in AI-powered security. They combined reinforcement learning with Explainable AI (XAI), making automated security decisions more understandable and trustworthy for human analysts. This is crucial, as analysts often struggle to interpret why AI systems make certain choices.

Other researchers have focused on resilience and insider threats. Vitekar et al. [7] developed an intelligent framework to monitor and identify suspicious actions by internal users in cyber-physical systems. While the system improved detection, it had challenges handling advanced replay attacks and adaptive insider threats.

Jawhar et al. [8,9] explored AI's role in cyber insurance and adaptive training. Their studies showed that predictive models can help organizations assess cyber risk and make better investment decisions. They also found that AI-driven, personalized training results in better cybersecurity preparedness than traditional, one-size-fits-all programs.

Patel et al. [11] looked at generative AI in cloud security, showing that automating tasks like alert analysis and threat prioritization can speed up response times and improve efficiency in large cloud environments.

Overall, these studies highlight how AI is transforming both the offensive and defensive sides of cybersecurity. Machine learning models are improving threat detection, automating analysis, and strengthening defenses. Yet there's growing concern about the misuse of generative AI for malicious purposes.

Despite these advances, there's a notable gap in research around adversarial co-evolution—where attackers and defenders both learn and adapt together over time. Most existing systems focus on either offense or defense, not both in combination. Few platforms provide a unified environment where AI agents can continuously compete and evolve.The Cyber Battlefield framework aims to bridge this gap by creating a reinforcement learning-driven environment where offensive and defensive agents learn together through direct competition. This approach enables researchers to study adaptive cyber warfare in a controlled setting, laying the groundwork for future research in autonomous cybersecurity systems.

VI. ACKNOWLEDGMENT

The authors would like to thank Project and Research Guide Mrs. Madhavi Indalkar and the Department of AI&DS at MMCOE for providing the resources for this project.

REFERENCES

[1] K. Ahi and S. Valizadeh, "Large Language Models (LLMs) and Generative AI in Cybersecurity and Privacy: A Survey of Dual-Use Risks,AI-Generated Malware, Explainability, and Defensive Strategies," *Proc. IEEE SVCC*, 2025.
[2] R. Fu, "Design and Implementation of Network Intrusion Detection System Based on Machine Learning," *Proc. IEEE ICISCN*, 2025.
[3] P. Liguori, R. Natella, and D. Cotroneo, "Generative AI in Cybersecurity: Generating Offensive Code from Natural Language," *Proc. IEEE DSN-S*, 2025.
[4] M. C. El Rai and M. Darseesh, "Dual-Branch Graph Attention Network for Cyber-Physical Intrusion Detection System for UAVs," *Proc. IEEE CCNCPS*, 2025.
[5] S. Kumar and A. K. Keshri, "Cyber Security against Cyber Warfare and Cyber Terrorism," *Proc. IEEE ISAC3*, 2025.
[6] M. Mishra, R. R. Pradhan, and K. Agrawalla, "AI for Cybersecurity Threat Detection: A Machine-Enabled Computing Perspective," *Proc. IEEE ASSIC*, 2025.
[7] A. B. Vitekar, P. B. Kadam, and S. R. Patil, "Development of an Intelligent and Resilient Security Box to Defend Against Insider Attacks in Cyber-Physical Systems," *Proc. IEEE ICSCSA*, 2025.
[8] S. Jawhar, R. Kumar, and V. Patel, "Enhancing Cyber Resilience with AI-Powered Cyber Insurance Risk Assessment," *Proc. IEEE CCWC*, 2024.
[9] S. Jawhar, R. Kumar, and V. Patel, "AI-Driven Customized Cyber Security Training and Awareness," *Proc. IEEE ICAIC*, 2024.
[10] D. Hayman, T. Lopez, and J. Nguyen, "Intelligence-Driven Threat Actor Analysis: BlackBasta and Affiliates," *Proc. IEEE Cyber-RCI*, 2024.
[11] A. Patel, N. Mehta, and S. Verma, "Generative AI for Automated Security Operations in Cloud Computing," *Proc. IEEE ICAIC*, 2025.