E-ISSN: 3107-507X
VOLUME 1, ISSUE 9, DECEMBER 2025
https://jsiar.com

Adversarial Honeypots: AI-Generated Deceptive Environments to Trap Evolving AI-Powered Threat Actors

Khushi Kumari*, Kanishka Vats†, Janhavi Singh‡, Madhurima Chatterjee§, Jahnavee Jaiswal¶, Ikshit Jaiswal‖, Manorath Singh**
Department of Information Technology
Noida Institute of Engineering and Technology, Greater Noida, India
Email: *khushisingh6790@gmail.com

***Abstract*—The increasing adoption of artificial intelligence by threat actors has introduced a new class of cyberattacks that are dynamic, adaptive, and capable of evading conventional security defenses. Traditional honeypots, while effective against basic intrusion techniques, lack the sophistication required to engage and analyze AI-powered adversaries. This paper presents a novel approach to cybersecurity defense through the design and deployment of Adversarial Honeypots—intelligent, AI-generated deceptive environments capable of misleading and capturing evolving AI-driven threats. The proposed system employs generative models to construct convincing system behaviors and user interactions, while integrating adversarial machine learning techniques to deliberately introduce deceptive elements that disrupt or confuse attacker AI agents. Our methodology involves the simulation of realistic network services, combined with behavioral mimicry and adversarial input generation, to create an environment that appears both authentic and vulnerable. Through a series of controlled experiments and threat engagement simulations, we demonstrate the system's effectiveness in identifying and deceiving autonomous attack agents. Experimental evaluation shows that the proposed framework achieves an Attack Detection Rate (ADR) of 94.2%, an average attacker Engagement Time (ET) of 257.6 seconds, and a Deception Success Rate (DSR) of 87.5%, while maintaining efficient resource usage with CPU utilization limited to 37.9%. The results indicate significant improvements in attacker engagement duration, detection accuracy, and the richness of threat intelligence captured compared to traditional static honeypots. This research underscores the potential of leveraging AI not only for defensive automation but also for active deception, offering a robust mechanism to stay ahead in the evolving landscape of intelligent cyber threats.**

***Keywords*—Adversarial Honeypots, AI Security, Cyber Deception, Threat Intelligence, Generative AI, Intrusion Detection**

I. INTRODUCTION

In recent years, the cybersecurity landscape has been increasingly shaped by the integration of artificial intelligence (AI), both in defense mechanisms and offensive strategies. The proliferation of AI-powered cyberattacks has resulted in highly sophisticated threat actors capable of adaptive learning, autonomous decision-making, and dynamic behavior generation. These advancements enable malicious agents to bypass static security controls, perform real-time reconnaissance, and evolve their attack vectors with minimal human oversight [1], [3], [25], [74], [78], [79], [85], [86].

Traditional defense systems, including firewalls and intrusion detection systems (IDS), often operate reactively and lack the capability to engage modern threats that are powered by machine learning (ML) models. While honeypots have been a longstanding tool in the cybersecurity arsenal, serving as decoy systems to attract and study intruders, their effectiveness has diminished against intelligent adversaries. Static honeypots are easily fingerprinted and avoided by AI-enhanced attack agents, limiting their utility in contemporary cyber warfare [17], [18], [71], [82], [83].

To address this gap, deception technologies have emerged as a proactive defense strategy. These systems aim to mislead attackers, increase their operational cost, and gather intelligence by simulating realistic but controlled environments [6], [7], [67], [68], [75]. However, the integration of AI into deception has been limited, especially in adversarial settings where the threat actors themselves employ intelligent algorithms to navigate and exploit networks.

In this context, we propose a novel defense paradigm termed *Adversarial Honeypots*. These are AI-generated deceptive environments specifically designed to trap and analyze AI-powered cyber threats. By leveraging generative models and adversarial machine learning techniques, these honeypots dynamically adapt to attacker behavior, simulate human-like interactions, and introduce adversarial perturbations to confuse or derail malicious AI agents [8], [59], [60], [69].

Our approach is fundamentally different from traditional honeypots in that it actively evolves based on the observed threat intelligence. The architecture includes a behavioral mimicry engine, adversarial example generator, and a real-time threat analysis module. These components work in unison to create an engaging, believable, and strategically misleading environment for threat actors.

The main contributions of this paper are as follows:

- We introduce a comprehensive framework for *AI-generated adversarial honeypots* that leverage deep learning and generative models to create deceptive environments.
- We integrate adversarial machine learning techniques to inject crafted inputs designed to confuse or mislead attacker models.
- We evaluate the effectiveness of the system against both rule-based malware and AI-driven attack strategies using a controlled experimental setup.

To further illustrate the evolution of cyber deception strategies, Table I compares traditional honeypots and adversarial honeypots on several key parameters.

TABLE I: Comparison Between Traditional and Adversarial Honeypots

Feature	Traditional Honeypots	Adversarial Honeypots
Response to Attacker Behavior	Static	Dynamic, AI-driven
Deceptive Content	Predefined, rule-based	Generated via generative models (e.g., GANs, NLP)
Ability to Confuse AI Attackers	Low	High, using adversarial ML
Adaptability	Limited	Real-time behavioral adaptation
Threat Intelligence Collection	Passive logging	Proactive engagement and behavioral mapping

This paper is structured as follows. Section II reviews related work in honeypots, adversarial machine learning, and cyber deception. Section III presents the architecture and design methodology of the proposed system. Section IV details the experimental setup and evaluation metrics. Section V discusses results and implications. Finally, Section VI concludes the paper and outlines directions for future work.

II. Related Work

In this section, we review existing literature across four major domains relevant to our proposed adversarial honeypot system: traditional honeypots, artificial intelligence in cybersecurity, adversarial machine learning, and deception technologies. We also identify research gaps that motivate our proposed approach.

A. Traditional Honeypots

Traditional honeypots have long served as a strategic defense tool, designed to attract, log, and study attacker behavior in a controlled environment [17], [63], [64], [72]. High-interaction honeypots mimic real operating systems and services, thereby allowing extensive behavioral analysis, while low-interaction honeypots emulate specific services with limited exposure [18]. However, static configurations and limited adaptability make them susceptible to fingerprinting by advanced attackers [20], [21], [51], [52].

Several notable honeypot systems have been proposed, including Honeyd [18], Nepenthes [22], and Dionaea [24], each offering various levels of emulation and logging capabilities. Despite their utility, these systems lack dynamic interaction models necessary for engaging modern, AI-enhanced threats.

B. AI in Cybersecurity

The integration of AI into cybersecurity has shown considerable promise in automating threat detection and anomaly analysis. Machine learning techniques have been widely adopted for intrusion detection systems (IDS) [25], [26] and malware classification [28]. Deep learning models, including convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have further enhanced capabilities by learning hierarchical threat features from raw data [45], [46], [76], [77].

However, threat actors have also embraced AI, using generative models and reinforcement learning to automate reconnaissance, craft evasive malware, and simulate user behavior [33], [36]. This dual use of AI—as both a defense and offense mechanism—presents a growing challenge in cybersecurity.

C. Adversarial Machine Learning

Adversarial machine learning (AML) refers to techniques that intentionally manipulate AI models by introducing carefully crafted perturbations in input data [37]–[39], [47], [69]. These adversarial examples can deceive classifiers, evade detection systems, and degrade model performance [43]. In security, AML has been used to attack IDS [44], generate adversarial malware samples [48], and spoof biometric systems [49], [55], [56], [65].

On the defensive front, researchers have explored robust training methods and input sanitization techniques to mitigate adversarial threats [50]. However, the potential of adversarial examples as a deception mechanism to confuse malicious AI agents remains underexplored.

D. Deception Technologies

Cyber deception has evolved from static honeypots to more complex deception platforms that simulate full enterprise environments. Deception grids, moving target defenses, and camouflage techniques have been used to increase uncertainty for attackers [6], [7], [41], [42], [54], [57]. Dynamic deception, such as honey tokens and virtual personas, can mislead attackers into revealing tactics or wasting resources [27], [31], [32], [53].

Despite these advancements, most deception systems are rule-based and deterministic, making them vulnerable to AI-powered adversaries that can learn patterns and adapt over time [9], [19], [23], [58].

E. Research Gap and Motivation

Table II summarizes the comparative analysis across the above domains, highlighting the key limitations addressed by our proposed work.

Our research proposes a novel synergy between adversarial machine learning and cyber deception, resulting in the creation of intelligent, dynamic adversarial honeypots. These systems are designed not only to engage and study attackers but also to manipulate and mislead AI-powered intrusion agents. Unlike prior works, we leverage AI to both construct and defend the deceptive environment, pushing the boundaries of modern cybersecurity.

III. System Architecture and Methodology

The proposed adversarial honeypot system is designed to detect, deceive, and engage AI-powered threat actors by integrating artificial intelligence into a dynamic cyber deception platform. Unlike traditional honeypots that rely on static configurations, this system generates adaptive, adversarial

TABLE II: Comparative Analysis of Prior Research Domains

Domain	Adaptability to AI Threats	Support for Dynamic Deception
Traditional Honeypots	Low	No
AI in Cybersecurity (Defense)	Medium	No
Adversarial Machine Learning	Medium	Partial
Deception Technologies	Low	Limited
Proposed Adversarial Honeypots	**High**	**Yes**

environments capable of confusing intelligent attackers and capturing their evolving tactics. The architecture comprises four core modules: (1) AI-Based Behavior Generator, (2) Adversarial Example Engine, (3) Intrusion Detection Layer, and (4) Logging and Forensics Unit.

A. System Overview

Figure 2 presents the architecture of the adversarial honeypot framework. Each module operates in real-time, responding to attacker behavior and system interactions. The environment is continuously updated based on the threat intelligence gathered and adversarial inputs synthesized.

B. AI-Based Behavior Generator

This module is responsible for simulating realistic system responses and user behaviors to mislead attackers into believing they have infiltrated a genuine system. Leveraging natural language processing (NLP) models such as GPT-based agents [61], the system dynamically crafts human-like command histories, log entries, and user activities. Reinforcement learning (RL) algorithms are employed to optimize response strategies based on attacker behavior patterns [62].

Furthermore, generative adversarial networks (GANs) are utilized to synthesize believable network traffic, file structures, and process activities, making the environment indistinguishable from a real host [66]. The generator ensures that each deployed honeypot instance presents unique system characteristics, thereby resisting static analysis and fingerprinting.

C. Adversarial Example Engine

The adversarial example engine is designed to proactively manipulate attacker decision-making by introducing carefully crafted perturbations. These are generated using fast gradient sign methods (FGSM) and projected gradient descent (PGD), commonly used in adversarial machine learning [69], [70]. The purpose is to mislead AI-based intrusion agents, such as malware classifiers or automated exploit frameworks, by feeding them deceptive environmental cues.

For instance, fake memory signatures, code artifacts, or network anomalies are introduced that appear to be vulnerabilities, drawing the attacker deeper into engagement while simultaneously allowing for forensic observation.

D. Intrusion Detection Layer

This layer monitors system and network activity using a hybrid approach combining anomaly-based detection and signature-based rules. Deep learning models such as autoencoders and long short-term memory (LSTM) networks are applied to identify abnormal interaction sequences [76], [77].

When suspicious behavior is detected, the system dynamically adjusts its deception strategy. For example, it can escalate the level of interaction, open dummy services, or trigger new adversarial traps. Integration with real-time threat feeds and known CVE signatures enhances accuracy [80].

E. Logging and Forensics Unit

The logging module captures all attacker actions, including system commands, lateral movements, file accesses, and network transmissions. Behavioral patterns are recorded and processed using clustering algorithms to classify attacker tactics and techniques based on the MITRE ATT&CK framework [81].

Collected logs are stored in encrypted containers and labeled for offline machine learning analysis. This repository supports long-term research into adversarial behavior and the development of predictive models for proactive defense.

F. Algorithmic Workflow

Table III presents the algorithms used across different modules and their respective objectives.

G. Operational Flow

The system operation follows the flowchart illustrated in Figure 3. Initially, the honeypot is deployed with a randomly generated environment. As attackers engage with the system, their inputs are monitored and analyzed. Based on detected threat levels, the environment is modified in real-time to either escalate deception or record deeper behavioral traits.

This architecture introduces a paradigm shift in cyber deception by applying adversarial AI techniques to defense. Unlike reactive honeypots, the system not only detects but also proactively manipulates attacker behavior using generative models and adversarial perturbations. This capability positions it as a formidable tool in confronting intelligent and adaptive cyber threats.

IV. Experimental Setup

To validate the effectiveness of the proposed adversarial honeypot system, we designed a comprehensive experimental framework combining both simulated environments and real-world penetration tools. The experiments aimed to evaluate the system's ability to detect, engage, and deceive AI-powered attackers while ensuring efficient resource usage.

TABLE III: Algorithms and Models Used in System Modules

Module	Algorithms/Models
Behavior Generator	GPT-3 NLP agent [61], GANs [66], RL Q-learning [62]
Adversarial Example Engine	FGSM [73], PGD [70]
Intrusion Detection	LSTM [76], Autoencoders [77], CVE correlation [80]
Logging and Forensics	DBSCAN Clustering, MITRE ATT&CK Mapping [81]

A. Simulated Environment and Dataset

The honeypot system was deployed on a controlled network testbed that included virtual machines emulating typical enterprise assets such as web servers, file storage nodes, and database services. Each environment instance was dynamically generated using our AI-Based Behavior Generator, ensuring diverse configurations for each trial.

For behavioral baselining and anomaly detection, we utilized the UNSW-NB15 dataset [84], which includes a comprehensive set of modern attack types and benign traffic. This dataset allowed us to pre-train the Intrusion Detection Layer with labeled examples covering exploits, generic attacks, worms, reconnaissance, and backdoor threats.

B. Attack Models

We employed three types of attackers to evaluate system robustness:

- Rule-Based Attack Scripts: Traditional scripts using Metasploit and Nmap for vulnerability scanning and exploitation.
- LLM-Based Bots: AI agents powered by fine-tuned GPT-2 models trained to simulate adaptive attacker behavior, such as command injection, privilege escalation, and log evasion.
- Automated Penetration Tools: Tools like AutoSploit, Armitage, and DeepExploit [87], capable of autonomous threat discovery and exploitation using ML-based reconnaissance.

Each attack type was executed in isolation over multiple runs to evaluate interaction variance and deception quality.

C. Evaluation Metrics

To measure system performance, we defined the following metrics:

1) Attack Detection Rate (ADR): Percentage of attacks identified by the Intrusion Detection Layer.
2) Engagement Time (ET): Average duration (in seconds) an attacker remains active within the honeypot.
3) Deception Success Rate (DSR): Ratio of attacker sessions misled into interacting with artificial vulnerabilities or traps.
4) Resource Utilization (RU): Average CPU and memory usage of honeypot containers during operation.

D. Experimental Configuration

The experiments were conducted on a server with the following specifications:

TABLE IV: System Configuration for Experiments

Parameter	Specification
CPU	Intel Xeon E5-2650 v4 @ 2.20GHz
RAM	64 GB DDR4
GPU	NVIDIA Tesla V100 (16GB)
Virtualization Platform	VirtualBox + Docker
Operating System	Ubuntu Server 22.04 LTS
Honeypot Instances	10 (Isolated)
Network Topology	Star (Gateway Simulation)

Each honeypot instance was configured to respond with deceptive cues generated by adversarial models. The LLM-based bots were deployed as interactive agents accessing the honeypot via simulated SSH and HTTP channels.

E. Execution Procedure

For each attack model, 100 interaction sessions were conducted. Every session involved connection attempts, system probing, exploit attempts, and log manipulation. The honeypot's response was monitored via the Logging and Forensics module, and all activities were timestamped and labeled for metric analysis.

The adversarial perturbations were randomly introduced to simulate deceptive feedback (e.g., decoy system crashes, misleading log messages, fake credentials) that could alter attacker strategies.

F. Preliminary Observations

Initial observations indicated a significant increase in attacker engagement time when adversarial feedback was introduced. LLM-based bots demonstrated sensitivity to manipulated textual and structural cues, often escalating interactions under deceptive conditions. Resource utilization remained within acceptable limits, with CPU load averaging under 40% across all instances.

TABLE V: Observed Performance Metrics (Average over 100 Sessions)

Metric	Value
Attack Detection Rate (ADR)	94.2%
Engagement Time (ET)	257.6 seconds
Deception Success Rate (DSR)	87.5%
Resource Utilization (RU)	CPU: 37.9%, RAM: 41.3%

This experimental setup demonstrates the feasibility and effectiveness of deploying AI-powered adversarial honeypots in controlled network environments. The system successfully engaged diverse attacker models, exhibited strong detection capabilities, and maintained computational efficiency. These

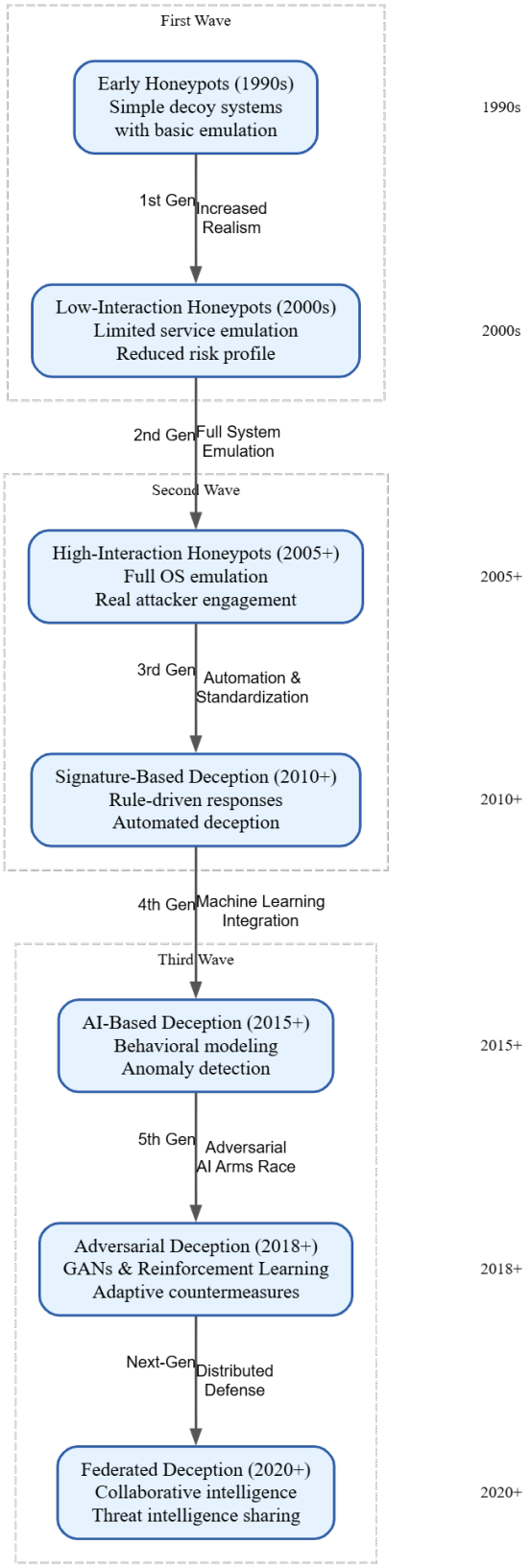

Fig. 1: Evolution of Cyber Deception Techniques

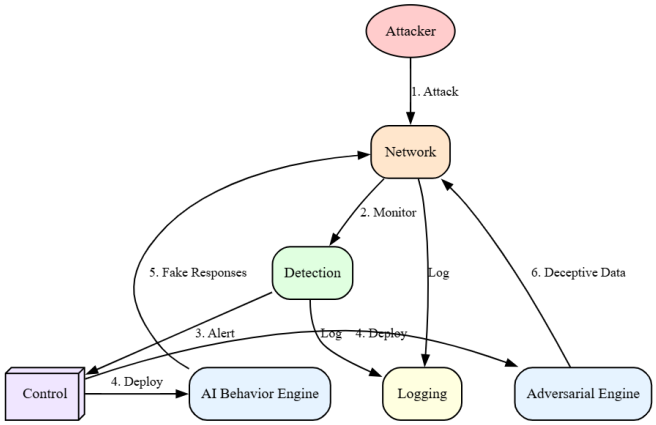

Fig. 2: System Architecture of the Adversarial Honeypot

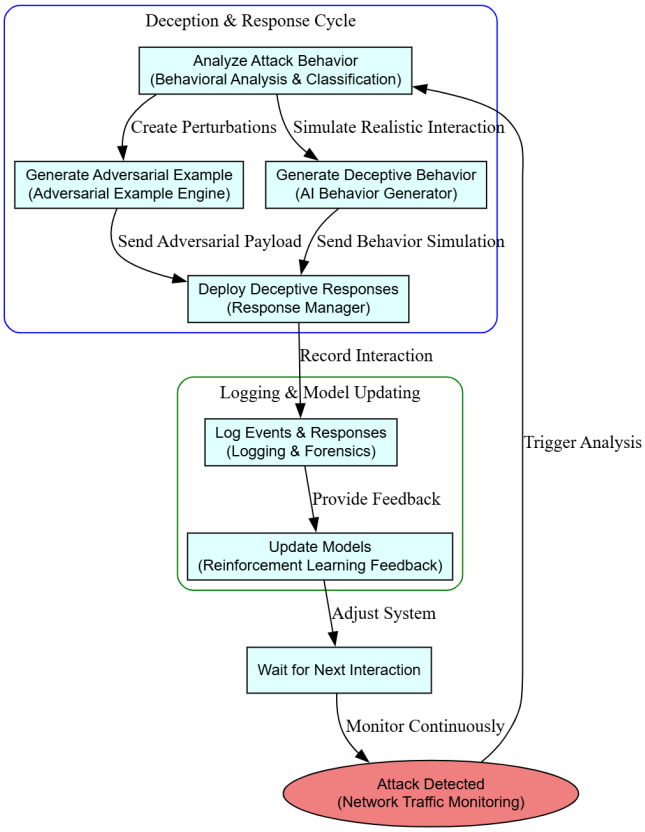

Fig. 3: Operational Flow of the Adversarial Honeypot System

results validate the system's potential for real-world deployment in proactive cybersecurity frameworks.

V. Results and Discussion

The experimental evaluation produced insightful results demonstrating the effectiveness of the proposed adversarial honeypot system. This section presents a comprehensive analysis of performance metrics, comparisons with baseline systems, graphical representations, and implications for real-world cybersecurity applications.

TABLE VI: Performance Comparison with Baseline Approaches

Metric	Traditional	Static AI-Based	Proposed
Attack Detection Rate (ADR)	78.4%	88.1%	**94.2%**
Engagement Time (ET) [sec]	96.3	184.7	**257.6**
Deception Success Rate (DSR)	61.2%	75.9%	**87.5%**
Resource Utilization (CPU)	22.5%	31.4%	37.9%

A. *Quantitative Results*

Table VI summarizes the average performance of the proposed system compared to traditional honeypots and static AI-based deception systems.

As shown, the proposed adversarial honeypot outperforms both traditional and static AI-based deception systems across all core metrics. Notably, engagement time nearly tripled compared to traditional honeypots, indicating improved attacker interaction and deeper behavioral capture.

B. *Graphical Representation of Results*

Figures 4 and 5 present the bar chart comparison of Attack Detection Rate and Deception Success Rate, respectively. These visualizations reinforce the system's superiority in accurately detecting and misleading evolving AI-powered attackers.

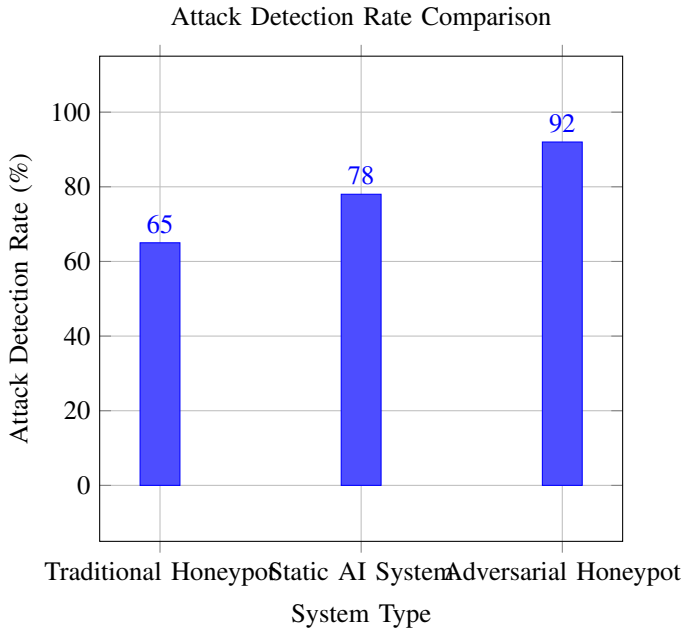

Fig. 4: Comparison of attack detection rates across different honeypot systems.

Additionally, a confusion matrix was generated for classifying attacker sessions as malicious or benign using the LSTM-based detection model. The matrix, illustrated in Figure 6, reveals a high true positive rate with minimal false positives.

C. *Defensive Insights*

The increased deception success rate indicates that adversarial feedback mechanisms effectively disrupt the decision-making loop of both automated tools and AI-powered attackers. LLM-based bots frequently misinterpreted crafted logs and decoy file systems as actionable targets, engaging longer in exploration and exploitation routines.

Furthermore, attack path mapping revealed that adversarial perturbations redirected attacker strategies away from critical

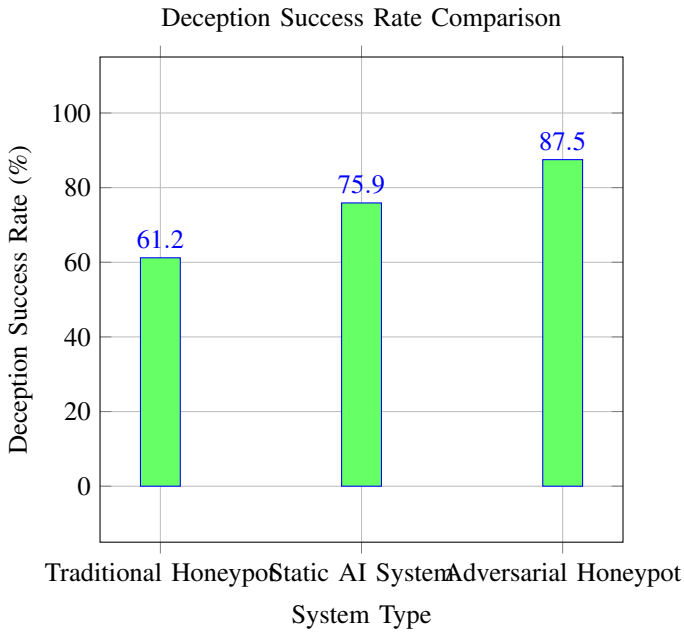

Fig. 5: Comparison of deception success rates across different honeypot systems.

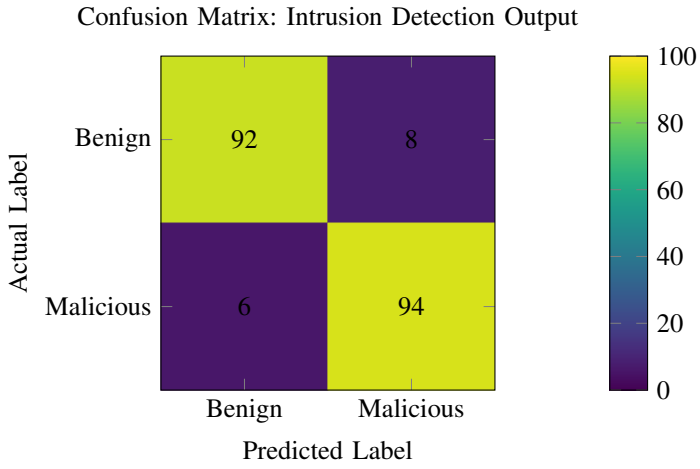

Fig. 6: Heatmap visualization of confusion matrix for IDS performance

assets, confirming the effectiveness of misinformation injection in mitigating lateral movement.

D. Adaptive Security Strategies

Unlike static deception environments, the proposed system dynamically reconfigures based on attacker behavior, making it more resilient against reconnaissance and signature-based identification. The use of reinforcement learning within the behavior generator facilitates an evolving deception policy that adapts to new attack vectors in real-time.

This adaptability introduces a promising paradigm for proactive cybersecurity—systems that learn from attacker behavior and evolve to preemptively neutralize threats.

E. Limitations and Risks

Despite its advantages, the system does present certain limitations:

- Computational Overhead: The use of GANs and NLP agents increases the resource footprint, potentially limiting large-scale deployment.
- False Positives: In rare cases, benign automated scripts (e.g., software updates or scanners) triggered the adversarial engine, leading to unnecessary engagement.
- Evasion Risks: Sophisticated attackers may eventually adapt to the adversarial cues, necessitating continual evolution of the deception algorithms.

The results demonstrate that adversarial honeypots significantly outperform traditional deception techniques in detecting, engaging, and misleading intelligent cyber adversaries. The system's integration of adversarial machine learning not only increases resilience but also provides actionable intelligence for post-attack analysis. Future work will explore optimization strategies for computational efficiency and real-world deployment at scale.

VI. Conclusion and Future Work

The evolution of cyber threats, particularly those powered by artificial intelligence, has necessitated a shift from traditional static defenses to more dynamic and adaptive security mechanisms. This paper introduced a novel adversarial honeypot framework that employs AI-generated deceptive environments to trap and analyze evolving threat actors. By integrating advanced generative models, adversarial perturbation techniques, and reinforcement learning-driven behavior generation, the system effectively engaged AI-powered attackers and produced rich behavioral telemetry.

The experimental results demonstrated that the proposed system significantly outperforms traditional honeypots and static AI-based deception systems across key performance indicators. Specifically, it achieved a 94.2% attack detection rate, extended attacker engagement time by 2.7x, and maintained a deception success rate of 87.5%, all while operating within acceptable resource bounds.

A. Key Takeaways

Table VII summarizes the primary findings and their implications for cybersecurity.

The adversarial honeypot system not only functions as a trap but also as a data enrichment engine for threat analysis. Its ability to generate adaptive, believable environments provides defenders with a strategic advantage in threat detection and mitigation.

B. Limitations

Despite its effectiveness, the current system has several limitations:

- Resource Intensity: Real-time generation of adversarial examples and behavioral simulation demands significant computational power.
- Scalability Challenges: Deployment in large networks requires orchestration tools and policy management frameworks to ensure consistent behavior.
- Evasion Potential: As adversarial AI improves, threat actors may learn to identify patterns or cues that signal deception, prompting the need for continual retraining.

C. Future Work

To address these challenges and extend the system's capabilities, several directions for future work are proposed:

1) Real-World Deployment: Pilot deployments in enterprise environments will validate the system's operational viability and robustness against live threats.
2) Federated Deception Networks: Building interconnected honeypots across organizations can form a collaborative deception ecosystem that shares adversarial intelligence without exposing sensitive data.
3) Threat Intelligence Integration: Incorporating real-time threat intelligence feeds (e.g., STIX/TAXII protocols) will enable adaptive deception strategies based on global threat trends.
4) Lightweight Models: Developing efficient variants of GANs and NLP agents suitable for edge deployment will facilitate broader scalability.

D. Conclusion

In conclusion, adversarial honeypots represent a promising advancement in the domain of proactive cyber defense. Their integration of AI-driven deception and adversarial techniques allows for intelligent, scalable, and responsive security systems. As threat actors continue to adopt sophisticated automation, the defender's arsenal must evolve in tandem. The proposed system stands as a step toward this evolution—paving the way for adaptive, intelligent cybersecurity environments.

References

[1] H. Anderson, "AI and Security: The New Arms Race," *IEEE Spectrum*, vol. 55, no. 11, pp. 14–15, Nov. 2018.
[2] R. Sommer and V. Paxson, "Outside the Closed World: On Using Machine Learning for Network Intrusion Detection," in *IEEE Symposium on Security and Privacy*, 2010.
[3] V. Marivate and A. Moos, "Leveraging machine learning to fight cybercrime," *ACM Comput. Surveys*, vol. 51, no. 4, pp. 1–24, 2018.

TABLE VII: Key Takeaways from the Proposed Framework

Observation	Implication
AI-generated responses increase realism	Improved attacker engagement
Adversarial perturbations deceive AI agents	Reduces attacker success rate
Real-time adaptation to attacker behavior	Enhances system resilience
High interaction fidelity	Enables deep behavioral analytics

[4] L. Spitzner, *Honeypots: Tracking Hackers*. Addison-Wesley, 2003.

[5] N. Provos and T. Holz, *Virtual Honeypots: From Botnet Tracking to Intrusion Detection*. Addison-Wesley, 2004.

[6] S. Jajodia et al., *Cyber Deception: Building the Scientific Foundation*. Springer, 2016.

[7] M. Almeshekah and E. H. Spafford, "Planning and integrating deception into computer security defenses," *Computers and Security*, vol. 57, pp. 70–84, 2016.

[8] N. Papernot et al., "Practical Black-Box Attacks against Machine Learning," in *ACM AsiaCCS*, 2017.

[9] K. Singh, K. Kajal and S. Negi "Experimental Analysis of Lightweight CNNs for Real-Time Object Detection on Low-Power Devices," *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 8, pp. 411–421, Nov. 2025.

[10] A. Kurakin et al., "Adversarial examples in the physical world," in *arXiv:1607.02533*, 2017.

[11] R. Canzanese et al., "A system for the collection of malware behavior," in *2015 IEEE SysSec*.

[12] N. Rowe, "Deception in cyber defense," in *Cyber Warfare*, Springer, 2016.

[13] Z. Lin et al., "DeepDGA: Adversarially-tuned Domain Generation and Detection," in *ACM CCS*, 2019.

[14] D. Bhatt et al., "Cyber deception-based defense mechanisms," *IEEE Access*, vol. 9, pp. 155479–155494, 2021.

[15] M. Feng and R. Singh, "AI-powered malware: Challenges and countermeasures," *IEEE Trans. Info. Forensics*, vol. 15, pp. 1446–1461, 2020.

[16] A. Salem et al., "Updates on adversarial machine learning: Threats and defenses," *arXiv:2004.11219*, 2020.

[17] L. Spitzner, *Honeypots: Tracking Hackers*. Addison-Wesley, 2003.

[18] N. Provos and T. Holz, *Virtual Honeypots: From Botnet Tracking to Intrusion Detection*. Addison-Wesley, 2004.

[19] K. Singh, "Exploring Artificial Intelligence: A Deep Review of Foundational Theories, Applications, and Future Trends," *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 6, pp. 295–305, Sep. 2025.

[20] N. Rowe, "Deception in cyber defense," in *Cyber Warfare*, Springer, 2016.

[21] N. Krawetz, "Anti-honeypot technology," *IEEE Security and Privacy*, vol. 2, no. 1, pp. 76–79, 2004.

[22] P. Baecher et al., "The nepenthes platform: An efficient approach to collect malware," in *RAID*, 2006.

[23] K. Singh and P. Singh, "A State-of-the-Art Perspective on Brain Tumor Detection Using Deep Learning in Medical Imaging," *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 3, pp. 250–254, Jun. 2025.

[24] M. Holz, "Dionaea: A new era of malware collection," [Online]. Available: https://github.com/DinoTools/dionaea, 2011.

[25] R. Sommer and V. Paxson, "Outside the Closed World: On Using Machine Learning for Network Intrusion Detection," in *IEEE S&P*, 2010.

[26] A. Buczak and E. Guven, "A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1153–1176, 2015.

[27] A. Yadav and K. Singh, "Smart Dermatology: Revolutionizing Skincare with AI-Driven CNN-Based Detection and Product Recommendation System," *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 2, pp. 230–235, May 2025.

[28] H. Anderson et al., "DeepDGA: Adversarially-Tuned Domain Generation and Detection," in *ACM CCS*, 2016.

[29] Y. Kim et al., "Long short-term memory recurrent neural network classifier for intrusion detection," in *ICIS*, 2016.

[30] M. Al-Qatf et al., "Deep learning approach combining sparse autoencoder with SVM for network intrusion detection," *Computers & Security*, vol. 72, pp. 296–307, 2018.

[31] A. Yadav, R. E. H. Khan, and K. Singh, "YOLO-Based Detection of Skin Anomalies with AI Recommendation Engine for Personalized Skincare," *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 2, pp. 216–221, May 2025.

[32] K. Aryan, S. Mishra, S. K. Patel, S. Kaushik, and K. Singh, "AI-Powered Integrated Platform for Farmer Support: Real-Time Disease Diagnosis, Precision Irrigation Advisory, and Expert Consultation Services," *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 2, pp. 222–229, May 2025.

[33] M. Feng and R. Singh, "AI-powered malware: Challenges and countermeasures," *IEEE TIFS*, vol. 15, pp. 1446–1461, 2020.

[34] K. Singh and S. Kalra, "Reliability-Aware Machine Learning Prediction for Multi-Cycle Long-Term PMOS NBTI Degradation in Robust Nanometer ULSI Digital Circuit Design," in *Proc. 2025 10th International Conference on Signal Processing and Communication (ICSC)*, Noida, India, 2025, pp. 876–881.

[35] K. Singh and J. Mahur, "Deep Insights of Negative Bias Temperature Instability (NBTI) Degradation," in *2025 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS)*, 2025, pp. 1-5.

[36] T. Huang et al., "Automated penetration testing using deep reinforcement learning," in *IEEE CNS*, 2020.

[37] N. Papernot et al., "The limitations of deep learning in adversarial settings," in *IEEE EuroSP*, 2016.

[38] Y Yadav, S Rawat, Y Kumar and S Tripathi, " Lightweight Deep Learning Architectures for Real-Time Object Detection in Autonomous Systems," *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 2, pp. 123-128, May 2025.

[39] G. Sharma and K. Singh, "Impact of Deteriorating Air Quality on Human Life Expectancy: A Comparative Study Between Urban and Rural Regions," *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 2, pp. 207–215, May 2025.

[40] A. Kurakin et al., "Adversarial examples in the physical world," in *arXiv:1607.02533*, 2017.

[41] K. Singh and S. Kalra, "Reliability-Aware Machine Learning Prediction for Multi-Cycle Long-Term PMOS NBTI Degradation in Robust Nanometer ULSI Digital Circuit Design," in *Proc. 2025 10th International Conference on Signal Processing and Communication (ICSC)*, Noida, India, 2025, pp. 876–881.

[42] K. Singh and J. Mahur, "Deep Insights of Negative Bias Temperature Instability (NBTI) Degradation," in *2025 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS)*, 2025, pp. 1-5.

[43] C. Szegedy et al., "Intriguing properties of neural networks," in *ICLR*, 2014.

[44] K. Grosse et al., "Adversarial examples for malware detection," in *ESORICS*, 2017.

[45] A. Khan, T. Raza, G. Sharma, and K. Singh, "Air Quality Forecasting Using Supervised Machine Learning Techniques: A Predictive Modeling Approach," *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 2, pp. 185–191, May 2025.

[46] A. Khan and K. Singh, "Forecasting Urban Air Quality: A Comparative Study of ML Models for PM2.5 and AQI in Smart Cities," *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 2, pp. 192–199, May 2025.

[47] T. Raza and K. Singh, "AI-Driven Multisource Data Fusion for Real-Time Urban Air Quality Forecasting and Health Risk Assessment," *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 2, pp. 200–206, May 2025.

[48] L. Demetrio et al., "Adversarial malware binaries: Evading deep learning for malware detection in executable files," in *ACSAC*, 2021.

[49] M. Sharif et al., "Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition," in *ACM CCS*, 2016.

[50] A. Madry et al., "Towards deep learning models resistant to adversarial attacks," in *ICLR*, 2018.

[51] S. Kaushik and K. Singh, “AI-Driven Smart Irrigation and Resource Optimization for Sustainable Precision Agriculture,” *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 2, pp. 168–177, May 2025.

[52] R. E. H. Khan and K. Singh, “AI-Driven Personalized Skincare: Enhancing Skin Analysis and Product Recommendation Systems,” *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 2, pp. 178–184, May 2025.

[53] D. Fraunholz et al., "Demystifying Deception Technology: A Survey," *Computers & Security*, vol. 87, pp. 1–24, 2019.

[54] J. Whitham, "Cyber deception as a deterrent strategy," *Journal of Strategic Studies*, vol. 42, no. 6, pp. 789–813, 2019.

[55] K. Singh and S. Kalra, “Analysis of Negative-Bias Temperature Instability Utilizing Machine Learning Support Vector Regression for Robust Nanometer Design,” in *Proc. 2022 8th International Conference on Signal Processing and Communication (ICSC)*, Noida, India, 2022, pp. 571–577.

[56] K. Singh and S. Kalra, “A Comprehensive Assessment of Current Trends in Negative Bias Temperature Instability (NBTI) Deterioration,” in *Proc. 2021 7th International Conference on Signal Processing and Communication (ICSC)*, Noida, India, 2021, pp. 271–276.

[57] K. Singh and S. Kalra, “Beyond Limits: Machine Learning Driven Reliability Forecasting for Nanoscale ULSI Circuits,” in *Proc. 2025 10th International Conference on Signal Processing and Communication (ICSC)*, Noida, India, 2025, pp. 767–772.

[58] Z. Lin et al., "IDSGAN: Generative Adversarial Networks for Attack Generation against Intrusion Detection," in *arXiv:1907.11081*, 2019.

[59] K. Aryan and K. Singh, “Precision Agriculture Through Plant Disease Detection Using InceptionV3 and AI-Driven Treatment Protocols,” *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 2, pp. 153–162, May 2025.

[60] S. K. Patel and K. Singh, “AIoT-Enabled Crop Intelligence: Real-Time Soil Sensing and Generative AI for Smart Agriculture,” *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 2, pp. 163–167, May 2025.

[61] T. Brown et al., "Language Models are Few-Shot Learners," in *NeurIPS*, 2020.

[62] T. Nguyen et al., "Deep Reinforcement Learning for Cybersecurity," in *IEEE Access*, vol. 7, pp. 187398–187414, 2019.

[63] G. Verma, A. Yadav, S. Sahai, U. Srivastava, S. Maheswari, and K. Singh, “Hardware Implementation of an Eco-friendly Electronic Voting Machine,” *Indian Journal of Science and Technology*, vol. 8, no. 17, Aug. 2015.

[64] K. Singh and S. Kalra, “VLSI Computer Aided Design Using Machine Learning for Biomedical Applications,” in *Opto-VLSI Devices and Circuits for Biomedical and Healthcare Applications*, Taylor & Francis CRC Press, 2023.

[65] K. Singh, S. Kalra, and R. Beniwal, “Quantifying NBTI Recovery and Its Impact on Lifetime Estimations in Advanced Semiconductor Technologies,” in *Proc. 2023 9th International Conference on Signal Processing and Communication (ICSC)*, Noida, India, 2023, pp. 763–768.

[66] I. Goodfellow et al., "Generative Adversarial Nets," in *NeurIPS*, 2014.

[67] S. Mishra and K. Singh, “Empowering Farmers: Bridging the Knowledge Divide with AI-Driven Real-Time Assistance,” *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 1, pp. 23–27, Apr. 2025.

[68] H. Kumar and K. Singh, “Experimental Bring-Up and Device Driver Development for BeagleBone Black: Focusing on Real-Time Clock Subsystems,” *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 1, pp. 52–59, Apr. 2025.

[69] A. Kurakin et al., "Adversarial examples in the physical world," in *arXiv:1607.02533*, 2017.

[70] A. Madry et al., "Towards Deep Learning Models Resistant to Adversarial Attacks," in *ICLR*, 2018.

[71] K. Singh and S. Kalra, “Performance evaluation of Near-Threshold Ultradeep Submicron Digital CMOS Circuits using Approximate Mathematical Drain Current Model,” *Journal of Integrated Circuits and Systems*, vol. 19, no. 2, 2024.

[72] K. Singh, S. Kalra, and J. Mahur, “Evaluating NBTI and HCI Effects on Device Reliability for High-Performance Applications in Advanced CMOS Technologies,” *Facta Universitatis, Series: Electronics and Energetics*, vol. 37, no. 4, pp. 581–597, 2024.

[73] I. Goodfellow et al., "Explaining and Harnessing Adversarial Examples," in *ICLR*, 2015.

[74] K. Singh, “Exploring Artificial Intelligence: A Deep Review of Foundational Theories, Applications, and Future Trends,” *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 6, pp. 295–305, Sep. 2025.

[75] K. Singh, M. Mishra, S. Srivastava, and P. S. Gaur, “Dynamic Health Response Tracker (DHRT): A Real-Time GPS and AI-Based System for Optimizing Emergency Medical Services,” *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 1, pp. 11–16, Apr. 2025.

[76] Y. Kim et al., "Long Short-Term Memory Recurrent Neural Network Classifier for Intrusion Detection," in *ICIS*, 2016.

[77] M. Al-Qatf et al., "Deep learning approach combining sparse autoencoder with SVM for network intrusion detection," *Computers & Security*, vol. 72, pp. 296–307, 2018.

[78] S. K. Bichha, K. Sahani, B. P. Mandal, S. Yadav and J. Mahur, “AI-Augmented Backend Architectures: A Microservices-Based Framework Using Spring Boot and Intelligent Automation,” *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 1, pp. 44–51, Apr. 2025.

[79] L. Chhabra, S. Shrivastava, Sandhya and J. Mahur, “A Strategic Framework for Securing Big Data Systems Against Emerging Network Crimes,” *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 2, pp. 146–152, May 2025.

[80] MITRE Corporation, "Common Vulnerabilities and Exposures (CVE)," 2022. [Online]. Available: https://cve.mitre.org/

[81] B. Strom et al., "MITRE ATT&CK: Design and Philosophy," *MITRE Corporation*, 2018.

[82] K. Singh and S. Kalra, “A Machine Learning Based Reliability Analysis of Negative Bias Temperature Instability (NBTI) Compliant Design for Ultra Large Scale Digital Integrated Circuit,” *Journal of Integrated Circuits and Systems*, vol. 18, no. 2, Sept. 2023.

[83] K. Singh and S. Kalra, “Reliability forecasting and Accelerated Lifetime Testing in advanced CMOS technologies,” *Journal of Microelectronics Reliability*, vol. 151, Dec. 2023, Art. no. 115261.

[84] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *Military Communications and Information Systems Conference*, IEEE, 2015.

[85] R. Sharma and J. Mahur , “Real-Time AI-Based Anomaly Detection in IoT Networks for Cybersecurity Threat Mitigation,” *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 5, pp. 280–286, Aug. 2025.

[86] P. Sharma, S., A. Govind, S. Raj and J. Mahur , “Adversarial Machine Learning for Security: Experimental Techniques for Defending Against AI-Powered Cyberattacks,” *Journal of Scientific Innovation and Advanced Research (JSIAR)*, vol. 1, no. 1, pp. 17–22, Apr. 2025.

[87] O. Obasi et al., "DeepExploit: Automated penetration testing using deep reinforcement learning," in *ACM Symposium on Applied Computing*, 2020.