A Cloud-Based AI-Powered Threat Deception Platform

S Aakash[1], S Ahamed Asarudeen[2], R A Arun Kumar[3], S Kirthik Sarvash[4], Mrs. P. Elakkiya[5]
[1, 2, 3, 4]Dept of Computer Science and Engineering
[5]Assistant Professor, Dept of Civil Engineering
[1, 2, 3, 4, 5] Anjalai Ammal Mahalingam Engineering College, Kovilvenni, Tamil Nadu, India

Abstract- *Modern web applications are increasingly targeted by automated bots and sophisticated attackers using advanced exploitation techniques such as injection attacks, credential stuffing, and reconnaissance-based probing. Traditional intrusion detection systems primarily focus on detection and blocking, often failing to extract actionable intelligence from adversarial interactions. This paper presents a cloud-based, AI-powered threat deception platform that actively engages attackers through realistic honeypot interfaces and tarpit mechanisms while simultaneously analyzing behavioral and payload-level data. The proposed system integrates rule-based attack signature detection with an XGBoost-based behavioral machine learning model to identify malicious activity with high accuracy. Severity assessment is performed using CVSS 3.1 scoring, and detected threats are mapped to OWASP Top 10 categories and relevant CVE references. The platform is fully deployed on cloud infrastructure using Firebase Hosting, a Flask-based backend, and Azure Blob Storage for scalable logging. Experimental evaluation demonstrates effective detection of multiple attack vectors including XSS, SQL injection, command injection, and automated bot behavior, while maintaining low operational cost. The results indicate that the proposed system not only detects threats but also converts attacks into valuable security intelligence.*

Keywords: Honeypot, Threat Deception, Tarpit, Machine Learning, XGBoost, Cloud Security, CVSS, OWASP.

I. INTRODUCTION

With the rapid adoption of cloud-based web applications, cyberattacks have become more frequent, automated, and complex[1]. Attackers increasingly employ bots [2] and exploit kits to identify vulnerabilities such as injection flaws, authentication weaknesses, and misconfigurations.

Conventional security mechanisms such as firewalls and also a signature-based intrusion detection systems are reactive in nature and provide limited insight into attacker behaviour. Threat deception has emerged as an effective defensive strategy that intentionally exposes controlled and isolated fake systems to attract attackers [3] by allowing adversaries to interact with deceptive environments, defenders can observe attack techniques, tools, and behavioral patterns development life cycle. When combined with tarpit mechanisms, which deliberately slow down attacker interactions, deception systems can significantly increase attacker cost and exposure time. This paper proposes a cloud-based AI-powered threat deception platform that combines deception, tarpit techniques, and machine learning-based behavioral analysis [4]. Unlike traditional systems that merely block attacks, the proposed platform actively engages attackers, delays them, and extracts intelligence that can be used for further security enhancement.

II. RELATED WORKS

Honeypot-based security mechanisms have been widely studied as proactive defense tools for intrusion detection and attack analysis [5]. Early honeypots focused on low interaction systems that captured basic attack data, while modern high-interaction honeypots allow deeper attacker engagement. A comprehensive comparative study by Zou et al. [6] demonstrated that high interaction honeypots captured 76.12% of total attack packets and attracted 70.61% of unique attacker IPs, compared to only 23.88% and 29.39% respectively for low-interaction variants. The study concluded that high-interaction systems provide substantially richer threat intelligence but require significantly higher maintenance overhead and pose greater security risks if compromised. Several researchers have explored medium interaction honeypots as a balanced approach [7,8]. These systems emulate specific vulnerable services with sufficient realism to attract targeted attacks while maintaining operational simplicity. However, many existing honeypot solutions, regardless of interaction level, lack scalability and advanced analytics capabilities [9,10]. The challenge of maintaining realistic deception while preventing honeypot fingerprinting remains a significant research concern [11].

Machine learning techniques have been applied to intrusion detection to overcome the limitations of static rule-based systems [12,13]. Algorithms such as Random Forest, Support Vector Machines, and Gradient Boosting mechanisms have demonstrated improved detection accuracy for network and application-layer attacks. XGBoost, in particular, has gained attention due to its robustness, efficiency, and ability to handle complex feature interactions [14]. A comprehensive comparison by SOW and Adda [15] evaluated Random Forest, XGBoost, and Deep Neural Networks (DNN) on the NSL-KDD dataset, employing SMOTE for class imbalance handling and Optuna for hyperparameter optimization. Their results showed Random Forest achieving 99.80% accuracy, slightly outperforming XGBoost (99.79%) and significantly exceeding DNN (98.66%). However, they noted that XGBoost demonstrated superior interpretability through feature importance analysis and faster training times on imbalanced datasets. Despite these advancements, limited research has explored the integration of deception systems with behavioral machine learning models [16] in a fully cloud-based environment. The proposed system addresses this gap by combining rule-based payload detection, ML-based behavioral analysis, and standardized severity scoring within a unified architecture [17].

Behavioral analysis has emerged as a complementary approach to signature-based detection, particularly for identifying automated attacks and distinguishing bots from human users [18]. Unlike payload inspection that examines what data is sent, behavioral analysis focuses on how interactions occur, including timing patterns, request sequences, and interaction dynamics [18,19].

Cloud infrastructure has transformed security system deployment by providing elastic scalability, distributed processing capabilities, and cost-effective resource allocation [20], [21]. Several researchers have explored cloud-based honeypot deployments to overcome the scalability limitations of traditional on-premises solutions [22], [23].

Despite significant advances in both honeypot technologies and machine learning-based intrusion detection, several critical gaps persist in current research: First, limited work has explored the integration of rule-based signature detection with behavioral machine learning models in honeypot environments. Most systems employ either pattern matching or ML classification, but not both in a complementary architecture [24]. Second, existing honeypot systems rarely incorporate standardized security metrics such as CVSS scoring and OWASP categorization for detected threats [9], [10]. This limits their utility for security operations that require risk-based prioritization and compliance reporting.

III. PROPOSED SYSTEM ARCHITECTURE

The overall architecture of the proposed Cloud-Based AI-Powered Threat Deception Platform is illustrated in Fig. 1. The system is designed using a modular cloud native approach to enable scalability, real-time threat analysis, and centralized monitoring. The architecture consists of five major components: User/Attacker Interface, Deception Frontend, Backend Threat Detection Engine, Data Logging Layer, and Administrative Dashboard.

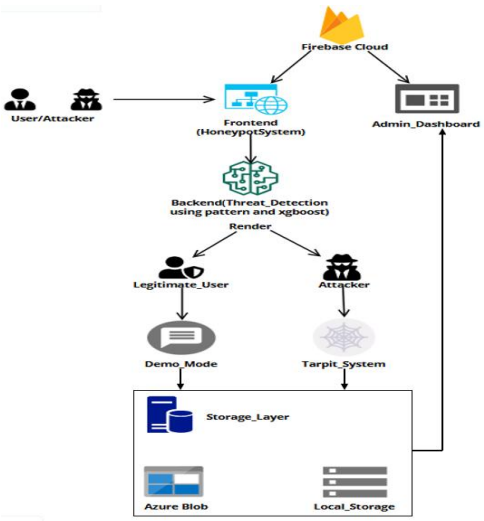

Fig. 1. System Architecture

A. User / Attacker Interface

The system is exposed to external users and potential attackers through the internet. Attackers interact with the platform believing it to be a legitimate web application. All interactions are intentionally directed toward the deception environment, ensuring that real production systems remain isolated and protected.

B. Deception Frontend (Honeypot System)

The frontend acts as a honeypot system and is hosted on Firebase Cloud Hosting. It presents a realistic web interface that mimics an authentic application. This layer captures both payload-level data and behavioral interaction data such as form inputs, submission timing, and interaction patterns. The frontend serves as the primary attack surface, attracting malicious users while remaining completely decoupled from real services.

C. Backend Threat Detection Engine

The backend component is responsible for analyzing incoming data from the frontend and is implemented using a

Flask-based REST API hosted on a cloud environment. The threat detection engine operates in two stages:

1. Pattern-Based Detection, which identifies known attack signatures such as XSS, SQL injection, command injection, and path traversal using predefined rules.
2. Machine Learning-Based Detection, which employs an XGBoost classifier to analyze behavioral features and distinguish between legitimate users and automated or malicious attackers. For this proposed system, 10,000 labeled samples are used including 70% of human behaviour and 30% of bot-based behaviour.

The backend combines results from both detection mechanisms to determine the attack severity and threat confidence.

D. Data Logging and Storage Layer

All detected events and interaction logs are securely stored in a centralized data logging layer. This layer uses a hybrid storage approach:

- Local Storage (Render) for temporary and fast-access logs.
- Azure Blob Storage for long-term, scalable, and persistent storage of attack data.

The stored logs include session identifiers, timestamps, detected vulnerabilities, severity levels, and mapped security references. This design ensures data reliability and availability for further analysis.

E. Administrative Dashboard

The administrative dashboard is hosted on Firebase Cloud and provides real-time visualization of security events. It retrieves aggregated attack statistics from the backend and presents insights such as total attacks, severity distribution, detected vulnerability types, and historical trends. This component enables security administrators to monitor threats, analyze attacker behavior, and evaluate system performance through an intuitive interface.

IV. RESULTS AND DISCUSSION

A. Experimental Setup

The proposed threat deception platform was deployed in a cloud environment using Firebase for frontend hosting and a Flask-based backend hosted on Render. Azure Blob Storage was utilized for persistent log storage. The honeypot web application was exposed to the internet to simulate a realistic attack surface. Various attack scenarios were manually and automatically generated to evaluate the effectiveness of the system. The dataset used for machine learning classification consisted of both benign user interactions and malicious payloads, including SQL Injection, Cross Site Scripting (XSS), command injection, and path traversal attempts. Behavioral features such as request frequency, payload length, special character density, and interaction timing were extracted and used for training and testing the XGBoost classifier.

B. Detection Performance

The system demonstrated effective detection of malicious activities using the hybrid detection approach. Rule-based filtering successfully identified known attack signatures, while the AI-based classifier handled unknown or obfuscated attack patterns. The XGBoost model achieved high classification accuracy in distinguishing between legitimate and malicious users; the accuracy is evaluated as 98.7% for this system. The integration of behavioral analysis reduced false positives compared to traditional signature-only detection methods. The model showed strong performance in identifying automated attack behavior such as brute-force attempts and scripted payload injections.

- Accuracy – 98.7%
- Precision – 99.3%
- F1-Scoring – 99.0%
- False Positive Rate – 0.7%

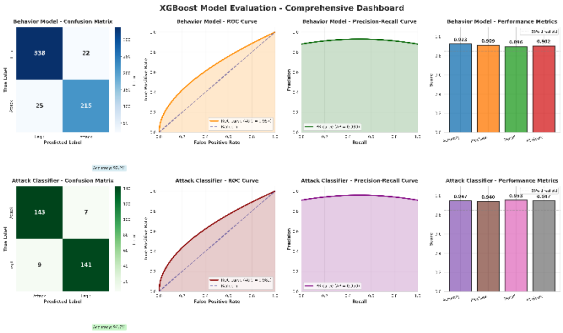

Fig. 2. XgBoost Model Evaluation

C. Logging and Visualization Analysis

All detected attacks were logged with detailed metadata, including timestamp, IP address, attack type, and severity level. The centralized logging system ensured data consistency and availability for analysis. The administrative dashboard provided real-time visualization of attack trends, enabling security administrators to quickly identify dominant

attack vectors and severity distributions. The results showed that injection-based attacks constituted the majority of malicious attempts, followed by scripting based attacks. The dashboard enabled effective monitoring without requiring direct backend access.

Fig. 3. Admin Dashboard

V. CONCLUSION

This paper presented a cloud-based AI-powered threat deception platform designed to proactively detect, analyze, and engage malicious users. By integrating honeypot-based deception with machine learning–driven threat classification, the system effectively identified both known and unknown attack patterns while safely isolating attackers from real resources. The use of cloud infrastructure ensured scalability, centralized logging, and real-time visualization through an administrative dashboard. Experimental results demonstrate that the proposed approach significantly improves threat intelligence collection while achieving lower false positive rates than conventional detection methods. Overall, the system represents a practical and efficient solution for modern cybersecurity challenges.

REFERENCES

[1] B. R. Bhimireddy, A. Nimmagadda, H. Kurapati, L. R. Gogula, K. Maheswari, and M. Sivakumar, "Web security and web application security: Attacks and prevention," in Proc. 9th Int. Conf. Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, Mar. 17–19, 2023.

[2] T. Arai, Y. Okabe, Y. Matsumoto, and K. Kawamura, "Detection of bots in CAPTCHA as a cloud service utilizing machine learning," in Proc. Int. Conf. Inf. Netw. (ICOIN), Barcelona, Spain, Jan. 2020.

[3] Darren Malvern, Bilal and Simon, "A practical honeypot-based threat intelligence framework for cyber defence in the cloud," in Proc. 19th Int. Conf. Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT), Pafos, Cyprus, Jun. 2024.

[4] Gurpreet Singh Walia and P. Deepalakshmi, "Protecting Cloud Computing Environments from Cyber Threats with AI-Powered Machine Learning Systems" in 2025 Global Conference in Emerging Technology (GINOTECH) PUNE, India 09-11 May 2025.

[5] V. Devi, A. Aiswarya, S. Deepa, P. S. Nanthini, and M. S. Sweatha, "Analysis of cyber defense mechanisms using honeypots in cloud environment," in Proc. Int. Conf. Intelligent Data Communication Technologies and Internet of Things (IDCIoT), Bengaluru, India, Jan. 2023.

[6] J. Zou, Z. Sun, C. Ku, X. Li, and A. Dahbura, "WiP: Developing high-interaction honeypots to capture and analyze region-specific bot behaviors," in Proc. Symp. Science of Security (HoTSoS), Baltimore, MD, USA, 2024.

[7] Zain Shamsi, Daniel Zhang, Daehyun Kyoung, Alex Liu, "Measuring and Clustering Network Attackers using Medium-Interaction Honeypots" in IEEE European Symposium on Security and Privacy Workshops (EuroS&PW) Genoa, Italy 06-10 June 2022.

[8] A. Theocharidou, K. Mitev, and T. K. Dimitrakos, "Hunting high or low: Evaluating the effectiveness of high-interaction and low-interaction honeypots," in Proc. Socio-Tech. Aspects Security (STAST 2022), Copenhagen, Denmark, Sep. 2022.

[9] W. Fan, Z. Du, D. Fernández, and V. A. Villagra, "Enabling an anatomic view to investigate honeypot systems: A survey," IEEE Syst. J., vol. 12, no. 4, pp. 3906–3919, Dec. 2018.

[10] I. Mokube and M. Adams, "Honeypots: Concepts, approaches, and challenges," in Proc. 45th Annu. Southeast Regional Conf., Winston-Salem, NC, USA, 2007.

[11] S. Morishita et al., "Detect me if you… oh wait. An internet-wide view of self-revealing honeypots," in Proc. IFIP/IEEE Symp. Integrated Network Service Manage. (IM), Arlington, VA, USA, 2019.

[12] A. Vasireddy, M. R. Senapati, N. Diwedi, P. C. Dash, and V. Sindhu, "Intrusion detection and prevention system to analyse and prevent malware using machine learning," in Proc. 2nd Int. Conf. Edge Computing and Applications (ICECAA), Namakkal, India, Jul. 2023.

[13] H. Li and J. Xu, "Application of artificial intelligence technology in honeypot technology," in Proc. IEEE Int. Conf. Cybernetics and Intelligent Systems (CIS) and IEEE Int. Conf. Robotics, Automation and Mechatronics (RAM), Chengdu, China, Dec. 2022.

[14] S. Madhavan, M. Brindha, A. Krishna, and M. Thilipan, "An efficient predictive analysis model of customer purchase behavior using Random Forest and XGBoost algorithm," in Proc. 4th Int. Conf. Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, Nov. 2020.

[15] T. H. SOW and M. Adda, "Enhancing IDS performance through a comparative analysis of Random Forest, XGBoost, and deep neural networks," Comp. Commun., vol. 231, article 107824, Sep. 2025.
[16] D. Thamizhselvi, Aswin Kumar K and Ajith Raj N, "Unauthorized Access Detection System Using Machine Learning with Honeypot Integration" in 2024 International Conference on Smart Technologies for Sustainable Development Goals (ICSTSDG) Chennai - 600077, Tamil Nadu, India 06-08 November 2024.
[17] Dharshan J Y, Arjun R Amarnath, Gadamsetty Siva Adithya, Anitha V and Kumaran U, "SmartHoneypot: Honeypot System for Lightweight Cyber Threat Detection and Behavior Analysis" in 2025 International Conference on Electrical, Electronics, and Computer Science with Advance Power Technologies - A Future Trends (ICE2CPT) Jamshedpur, India 29-31 October 2025.
[18] Z. Chu, S. Gianvecchio, A. Koehl, H. Wang, and S. Jajodia, "Blog or block: Detecting blog bots through behavioral biometrics," Comput. Netw., vol. 57, no. 3, pp. 634–646, Feb. 2013.
[19] Y. Boshmaf, D. Logothetis, G. Siganos, et al., "Integro: Leveraging victim prediction for robust fake account detection in OSNs," in Proc. Network and Distributed System Security Symp. (NDSS), San Diego, CA, USA, 2015.
[20] A. L. 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, 2016.
[21] Amazon Web Services, "Overview of security in the cloud," AWS White Paper, 2021.
[22] J. Wang, S. Zhang, and Y. Li, "A highly interactive honeypot-based approach to network threat management," Future Internet, vol. 15, no. 4, article 127, Mar. 2023.
[23] S. Gupta, P. Mishra, and D. Gupta, "Cloud-based honeypot deployment: Challenges and opportunities," in Proc. Int. Conf. Advances Comput., Commun. Inform. (ICACCI), Bangalore, India, 2018.
[24] M. Conti, Q. Qiu, A. P. Mathur, and J. Zhao, "Survey of intrusion detection systems: techniques, datasets and challenges," Cybersecurity, vol. 2, article 20, Jul. 2019.