iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** III **Month of publication:** March 2026

DOI: https://doi.org/10.22214/ijraset.2026.78511

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue III Mar 2026- Available at www.ijraset.com

Fortifying Cyber Defense: Honeypot-Driven Strategies for DDoS Detection and Prevention

Vivekkumar Chauhan[1], Isha Ben Pulintara[2], Mohammad Tasawwur Khan[3], Usmani Abu Rabey[4], Jagruti More[5]
Theem College of Engineering

Abstract: ***In today's interconnected digital ecosystem, cyber threats, especially sophisticated Distributed Denial of Service (DDoS) attacks, are evolving at a pace that our current security tools simply cannot match. These traditional systems are fundamentally reactive; they rely on known signatures and static rules, making them ineffective against new, zero-day attack vectors and stealthy application-layer threats that mimic legitimate traffic. This reactive posture is a critical vulnerability, as it means we only begin to act after our systems are already under siege, turning our efforts into mere damage control rather than prevention. To counter this, our project proposes a radical shift to a proactive, intelligence-driven defence centered around a high-interaction honeypot. This isn't just a simple decoy; it is a fully functional, sandboxed environment designed to be an irresistible target, luring attackers to reveal their complete playbook in a safe, monitored setting. Once an attacker engages with this honeypot, we can capture an incredibly rich stream of data, including their IP addresses, the specific malware they deploy, and most importantly, their Tactics, Techniques, and Procedures (TTPs). This live threat intelligence will then be fed into an automated system, creating a powerful real-time feedback loop where our defences, like firewalls and traffic filters, learn from every assault and instantly update themselves to block similar threats across our entire live network. The ultimate goal here transcends simple attack prevention; by continuously analysing this data, we move beyond being just proactive to becoming truly predictive. We can start to identify emerging attack trends and forecast our adversaries' next moves, allowing us to build defences for threats before they even materialize, thereby ensuring maximum uptime and solidifying our reputation as a secure and resilient organization.***

Keywords: 1) Distributed Denial of Service (DDoS) 2) High-interaction 3) Honeypot 4) Cybersecurity 5) Threat intelligence 6) Real-time Mitigation Predictive security 7) Network security

I. INTRODUCTION

In the era of hyper-connectivity, organizations across every sector face an ever-growing risk from cyber threats, with Distributed Denial of Service (DDoS) attacks emerging as one of the most disruptive and damaging forms of assault. These attacks have evolved from simple volumetric floods into complex, multi-vector campaigns that can cripple critical infrastructure, disrupt business continuity, and inflict significant financial and reputational losses. Traditional defense mechanisms such as signature-based intrusion detection systems, static firewall rules, and conventional mitigation appliances are no longer sufficient. Their reactive nature leaves organizations vulnerable to novel, zero-day exploits and stealthy application-layer attacks that blend seamlessly with legitimate user traffic. As adversaries grow more sophisticated, relying solely on reactive defense translates into inevitable downtime and costly recovery efforts. To overcome these limitations, this project introduces a proactive and intelligence-driven defense framework powered by high-interaction honeypots. Unlike low-level decoys, high interaction honeypots provide a realistic and fully functional environment that actively engages attackers, compelling them to expose their methods and strategies. By capturing detailed information such as attacker IP addresses, malicious payloads, and behavioral patterns, the honeypot transforms into a dynamic source of live threat intelligence. This intelligence feeds into an automated feedback loop that continuously updates security controls—such as firewalls, intrusion prevention systems, and traffic filters—in real time. Beyond immediate mitigation, the proposed system aspires to achieve predictive capabilities. By analyzing trends and attacker tactics over time, it becomes possible to anticipate future attack strategies and preemptively reinforce defenses. This transition from reactive to predictive security not only minimizes downtime but also ensures long-term resilience. Through this innovative honeypot-driven approach, the project aims to establish a robust cybersecurity framework that safeguards digital assets while fostering trust and reliability in today's interconnected digital ecosystem.

II. PROCEDURE FOR PAPER SUBMISSION

It begins with Phase 1: Planning, during which the Project Specification task is completed in early January. This phase plays a crucial role in defining the project's objectives, scope, and technical requirements, serving as the foundation for all upcoming stages. Following this, Phase 2: Development focuses on building the project's core infrastructure. The Core System Design and Database Setup activities take place through mid to late January, highlighting the technical backbone of the system. These tasks involve designing the system's architecture, creating robust databases, and preparing the framework necessary to support later modules. In Phase 3: Detection Modules, which runs through February, the emphasis shifts to enhancing the project's security and monitoring capabilities. Two major components — Threat Detection and Alert System — are developed during this stage. These modules ensure that potential risks or breaches can be identified and reported promptly, strengthening the system's reliability and resilience. Phase 4: Analytics, scheduled for March, introduces the Dashboard Development process. This phase aims to integrate analytical and visualization tools that help interpret real-time data, monitor performance metrics, and present insights in a user-friendly manner. The dashboard becomes a key tool for continuous assessment and decision-making throughout the project's lifecycle. Finally, Phase 5: Testing & Launch covers the final stretch of the project in April. The Testing Phase focuses on evaluating system performance, checking for bugs or vulnerabilities, and verifying that all features operate seamlessly under different conditions. This ensures quality assurance and readiness for deployment. The project concludes with the Final Deployment milestone on April 10, 2026, marking the official launch of the completed system.

III. EXPERIMENTAL SETUP

Hardware Configuration The experimental setup consists of three virtual machines deployed on a vercel. The honeypot server is configured with 4 virtual CPUs, 8 GB of RAM, and 100 GB of storage, running Ubuntu Server 22.04 LTS. The detection server has 8 virtual CPUs, 16 GB of RAM, and 500 GB of storage, operating on CentOS 8 Stream to handle intensive packet analysis tasks. The management workstation is equipped with 4 virtual CPUs, 8 GB of RAM, and 256 GB of storage.

Software Implementation The honeypot layer consists of Dionaea and Cowrie, which are used for capturing and emulating different attack vectors. Dionaea focuses on multi-protocol vulnerabilities, while Cowrie mimics SSH and Telnet services to log brute-force and command-based intrusions. Additionally, a custom Python-based HTTP honeypot is developed to monitor and analyze Layer 7 web attacks. The detection system uses Scapy for packet manipulation and custom detection scripts to identify abnormal traffic patterns and Distributed Denial of Service (DDoS) indicators. TCPDump and Wireshark are employed for packet capture and in-depth analysis. The web-based management interface is developed using the Flask framework, integrated with Socket.IO for Realtime communication and Chart.js for visualizing attack statistics. All captured data, including metadata and logs, are stored in a PostgreSQL database to ensure scalability and efficient querying.

Testing Methodology Testing is conducted through simulated DDoS and intrusion attacks generated from a Kali Linux machine. Tools such as hping3 are used for SYN flood attacks, GoldenEye for HTTP-based floods, and Slowloris for application-layer slow requests. To simulate legitimate user activity, Apache JMeter is used to generate baseline network traffic. Later, multi-vector DDoS attacks combining different layers are tested to evaluate the robustness of the system. Performance metrics are analyzed in terms of detection efficiency, system stability, and resource utilization under varying loads.

IV. METHODOLOGY

The proposed project follows a proactive and intelligence-driven methodology to detect and mitigate Distributed Denial of Service (DDoS) attacks using a high-interaction honeypot–based defense framework. Initially, a controlled network environment is designed consisting of a target web server, attacker simulation module, firewall, honeypot server, and monitoring dashboard. The system is developed in a local or virtualized setup to safely simulate attack scenarios without affecting real-world infrastructure. A traffic generation module is used to simulate DDoS attacks by sending a large number of requests to the target server, allowing the system to observe server performance under heavy load and define a threshold for abnormal traffic. A continuous monitoring module analyzes incoming traffic based on request rate, response delay, CPU usage, and IP frequency to detect suspicious behavior. When the traffic exceeds the predefined threshold, the system identifies it as a potential attack and redirects the suspicious requests to a high-interaction honeypot instead of directly blocking them. The honeypot behaves like a real server and records detailed information such as attacker IP address, request type, payload, and attack pattern, which is used as live threat intelligence. This information is then passed to an automated mitigation module that dynamically updates firewall rules, blocks malicious IP addresses, and filters abnormal traffic while allowing legitimate users to access the server. All activities are logged and displayed on a real-time dashboard showing traffic status, detected attacks, honeypot activity, and server performance.

The collected attack data is further analyzed to identify patterns and predict future threats, enabling the system to strengthen defenses in advance. Finally, the system is tested under normal traffic, moderate attack, and high-volume DDoS conditions to evaluate detection speed, mitigation efficiency, and overall stability, demonstrating the effectiveness of the proposed honeypot-driven proactive cybersecurity framework.

V. MOTIVATION

The increasing dependence on internet-based services has made systems more vulnerable to cyber threats, especially Distributed Denial of Service (DDoS) attacks that can disrupt services and cause major losses. Traditional security methods are mostly reactive and unable to handle modern, complex attacks effectively. Therefore, this project is motivated by the need for a proactive and intelligent security system using a high-interaction honeypot to detect, analyze, and mitigate attacks in real time. The goal is to improve network security, reduce downtime, and build a predictive defense mechanism that can protect digital infrastructure from advanced cyber threats.

VI. SYSTEM ARCHITECTURE

The proposed system architecture is developed to provide a proactive and intelligent defense against Distributed Denial of Service (DDoS) attacks by using a honeypot-based security framework organized into multiple layers, where each layer performs a specific role in monitoring, detection, analysis, and protection. In the Honeypot Deployment Layer, it is more or less dashboard a high-interaction honeypot is placed inside the network to attract suspicious traffic so that attacker information such as IP address, request frequency, and payload behavior can be recorded. The Network Analysis Intelligence Layer continuously observes incoming packets and compares them with predefined limits to identify unusual patterns. The Threat Mitigation and Response Layer then takes automatic action by blocking harmful IP addresses, updating firewall rules, and redirecting unwanted traffic to the honeypot, ensuring that the original server continues to function normally. The proposed system architecture is developed to provide a proactive and intelligent defense against Distributed Denial of Service (DDoS) attacks by using a honeypot-based security framework organized into multiple layers, where each layer performs a specific role in monitoring, detection, analysis, and protection. The Interface Layer enables the administrator to control and observe the system through a dashboard, while the Presentation Layer displays traffic details, alerts, and system performance in a clear and understandable form. In the Honeypot Deployment Layer, it is more or less dashboard a high-interaction honeypot is placed inside the network to attract suspicious traffic so that attacker information such as IP address, request frequency, and payload behavior can be recorded. The Network Analysis Intelligence Layer continuously observes incoming packets and compares them with predefined limits to identify unusual pattern. The Threat Mitigation and Response Layer then takes automatic action by blocking harmful IP addresses, updating firewall rules, and redirecting unwanted traffic to the honeypot, ensuring that the original server continues to function normally. The proposed system architecture is developed to provide a proactive and intelligent defense against Distributed Denial of Service (DDoS) attacks by using a honeypot-based security framework organized into multiple layers, where each layer performs a specific role in monitoring, detection, analysis, and protection.continously observes incoming packets and compares them with predefined limits to identify unusual patterns. When abnormal traffic is detected, the DDoS Detection and Traffic Analysis Layer confirms whether the activity is malicious or legitimate. All events are stored in the LoggIn the Honeypot Deployment Layer, it is more or less dashboard a high-interaction honeypot is placed inside the network to attract suspicious traffic so that attacker information such as IP address, request frequency, and payload behavior can be recorded. The Network Analysis Intelligence Layer and Forensic Layer, which keeps detailed records for future analysis, report generation, and attack prediction. The Threat Mitigation and Response Layer then takes automatic action by blocking harmful IP addresses, updating firewall rules, and redirecting unwanted traffic to the honeypot, ensuring that the original server continues to function normally. This multi-layered architecture supports real-time monitoring, accurate detection, automatic response, and improved security, making the system reliable for handling modern and advanced cyber threats without interrupting normal services. The Interface Layer enables the administrator to control and observe the system through a dashboard, while the Presentation Layer displays traffic details, alerts, and system performance in a clear and understandable form. In the Honeypot Deployment Layer, it is more or less dashboard a high-interaction honeypot is placed inside the network to attract suspicious traffic so that attacker information such as IP address, request frequency, and payload behavior can be recorded. The Network Analysis Intelligence Layer continuously observes incoming packets and compares them with predefined limits to identify unusual patterns. When abnormal traffic is detected, the DDoS Detection and Traffic Analysis Layer confirms whether the activity is malicious or legitimate. All events are stored in the Logging, Reporting, and Forensic Layer, which keeps detailed records for future analysis, report generation, and attack prediction.

The Threat Mitigation and Response Layer then takes automatic action by blocking harmful IP addresses, updating firewall rules, and redirecting unwanted traffic to the honeypot, ensuring that the original server continues to function normally. This multi-layered architecture supports real-time monitoring, accurate detection, automatic response, and improved security, making the system reliable for handling modern and advanced cyber threats without interrupting normal services.

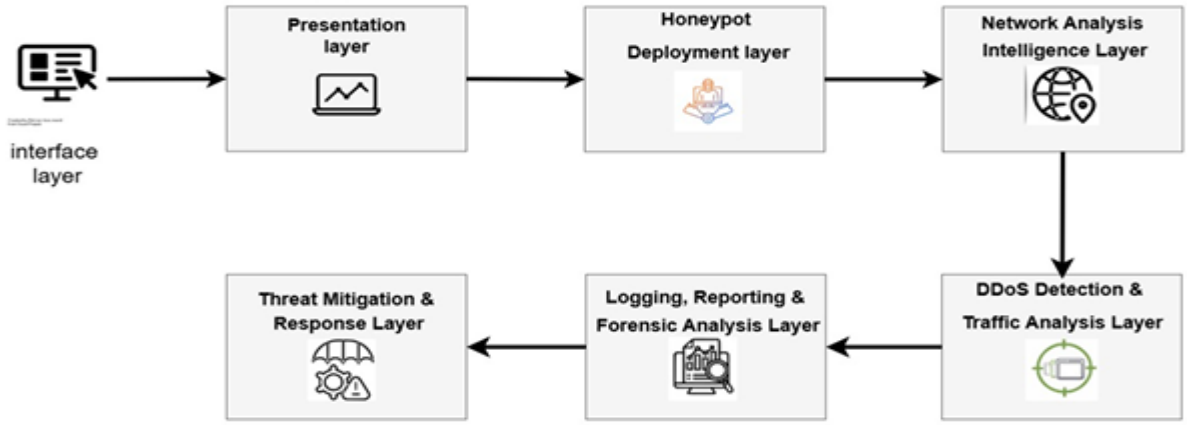


Figure – System Architecture

VII. RESULTS AND DISCUSSIONS

The experimental results demonstrate the effectiveness of the proposed honeypot-based framework in monitoring network activity and identifying abnormal traffic patterns. During the testing stage, several attack scenarios were generated to examine how the system behaves under different network conditions. These scenarios included simulated Distributed Denial of Service (DDoS) traffic along with normal user requests in order to evaluate detection accuracy. The monitoring module continuously analysed incoming traffic and was able to recognize suspicious behaviour based on predefined thresholds and traffic patterns.

When abnormal traffic was detected, the system redirected the malicious requests to the honeypot environment instead of allowing them to reach the primary server. This approach allowed the framework to capture valuable attack information such as source IP addresses, request frequency, payload data, and behavioural patterns of the attacker. All collected data were stored as logs and further processed by the analytics module to understand the nature of the attack. The dashboard also provided a visual representation of detected attacks, severity levels, and system activity in real time.

Figure I - Attack Events Dashboard This figure shows the attack event monitoring interface of the Honey Shield platform. It allows the administrator to simulate and observe different types of attacks such as HTTP flood in a controlled environment. The dashboard records important information including source IP, attack type, severity level, and request count. This helps in analyzing how the system reacts to malicious traffic. The generated events are further used to test the detection and mitigation capabilities of the framework.

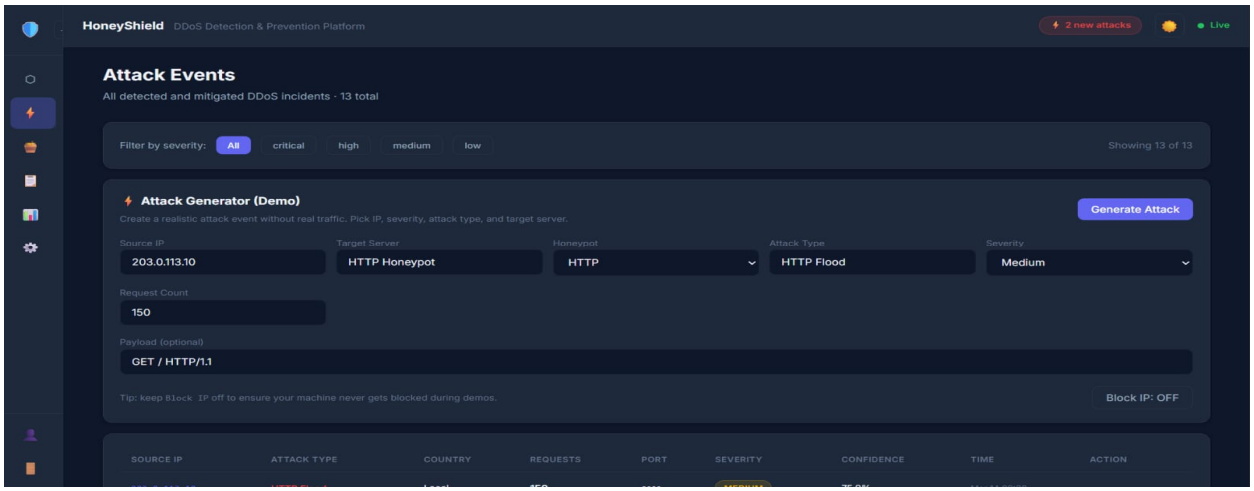


Figure-II Machine Learning Analytics and Intelligence This figure illustrates the analytics module of the Honey Shield system, which uses machine learning techniques to evaluate network traffic and detect abnormal behaviour. The system processes collected traffic data and identifies patterns that differ from normal network activity. The dashboard presents key metrics such as the total number of analysed events, detected anomalies, and the overall anomaly rate. In addition, the confidence score indicates how reliably the model classifies suspicious traffic as potential threats. Graphical visualizations such as attack distribution and severity breakdown help in understanding the frequency and impact of different attack types. These insights assist administrators in monitoring security conditions and making informed decisions for threat mitigation.

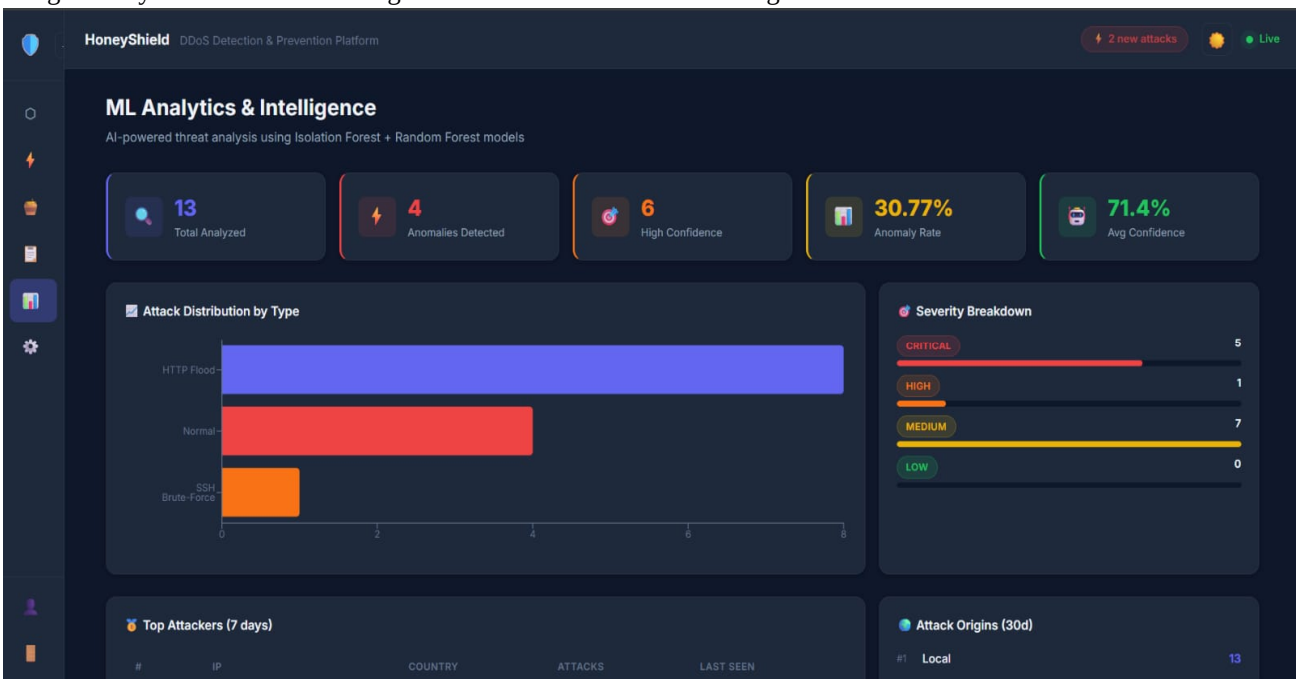


Figure III - Honeypot Systems Monitoring This figure illustrates the status of different honeypot services deployed in the system. Multiple honeypots such as SSH, HTTP, FTP, Telnet, and SMTP are configured to imitate real network services. These decoy services attract attackers and record their interaction attempts. The dashboard displays metrics like total hits and unique source IPs for each honeypot. This information helps in studying attacker behavior and identifying commonly targeted services.

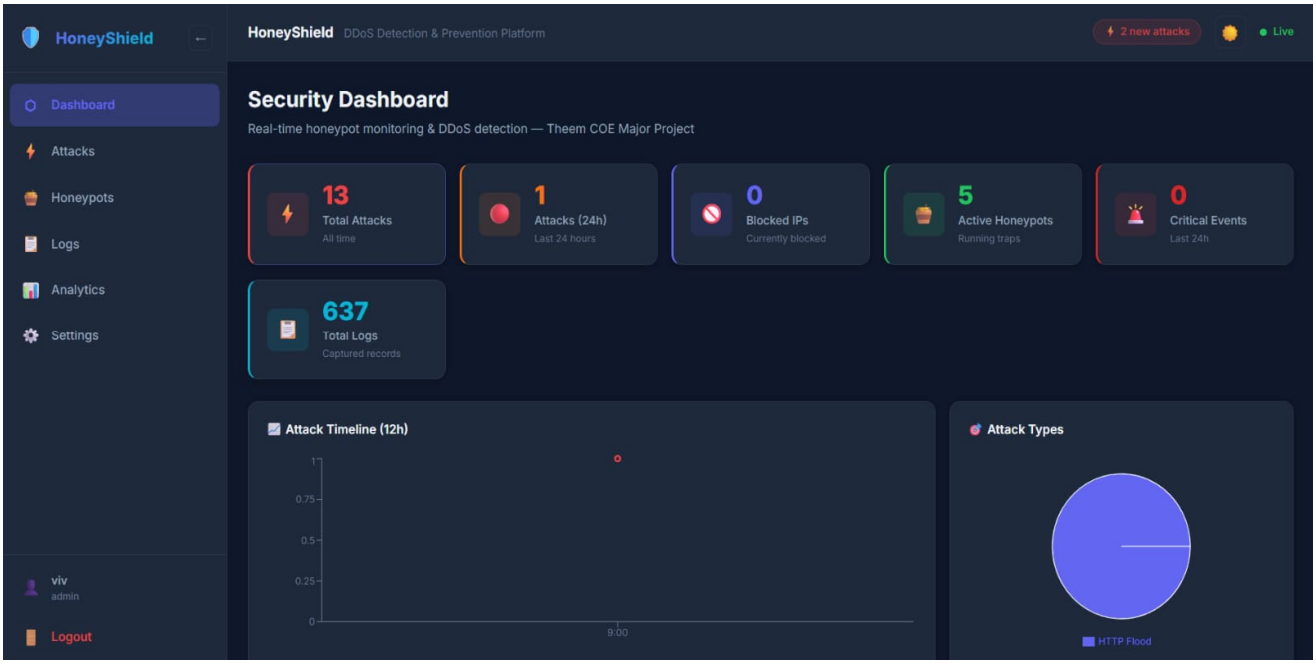
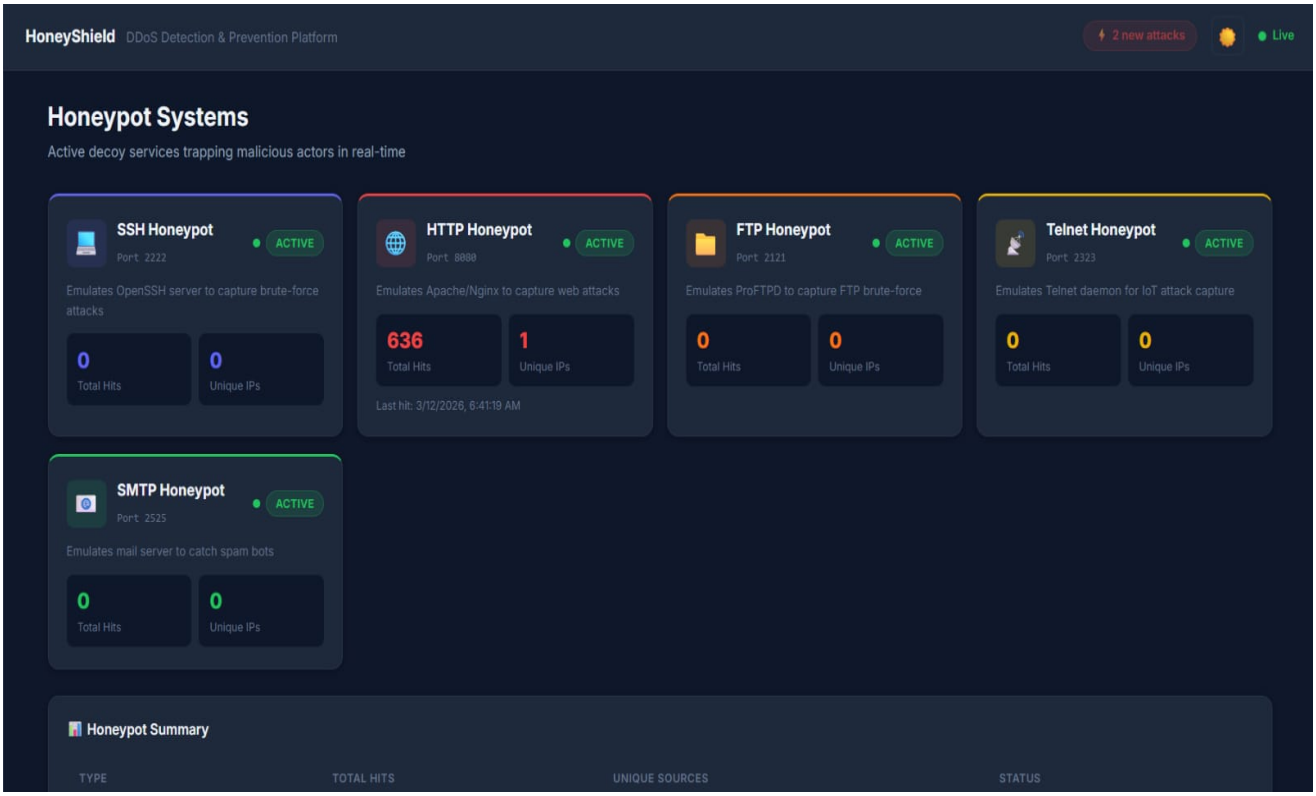


Figure IV- Security Monitoring Dashboard This figure displays the main security dashboard used for real-time monitoring of the network environment. It summarizes key metrics including total detected attacks, active honeypots, captured logs, and recent attack activity. Graphical visualizations provide insights into attack timelines and distribution of attack types. The dashboard enables administrators to quickly assess the current security status of the system. This centralized monitoring improves response time during potential cyber threats.



VIII. CONCLUSION

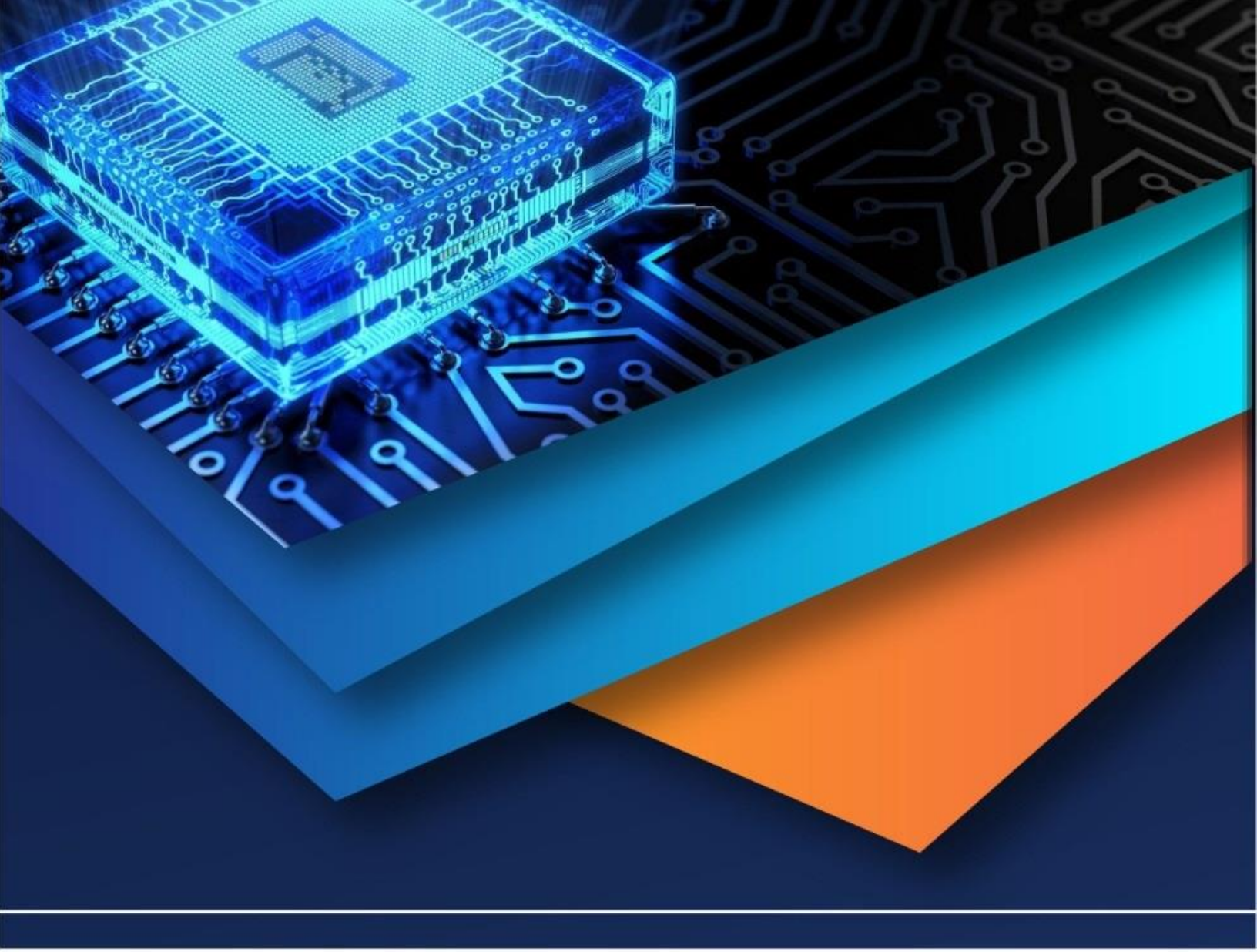
The project on DDoS detection and mitigation using Splunk, Logstash, a real server, and a fake server presents a practical framework for improving network security and handling distributed denial-of-service attacks. By combining multiple tools within a single architecture, the system enables continuous monitoring of network traffic, efficient analysis of log data, and faster response to abnormal activities.Logstash plays a key role in collecting and processing logs from various sources such as firewalls, web servers, and routers. It converts the raw data into a structured format and forwards it to Splunk for further analysis. Splunk acts as the central platform for monitoring and visualization, where dashboards and correlation searches help identify unusual traffic patterns. For example, a rapid increase in request volume or repeated connections from the same IP address can indicate a potential attack, allowing early detection and quick action. This server imitates a real service to attract malicious traffic and record attacker behavior. By diverting suspicious activity toward the honeypot, it becomes possible to study attack methods and gather useful forensic information without affecting the real production server. At the same time, the real server continues to provide services under protected conditions.Another important aspect of the system is automated mitigation. Once suspicious traffic is detected, alerts and scripts can automatically update firewall rules, block harmful IP addresses, or trigger other defensive actions through APIs. This automation reduces the delay between detection and response and helps maintain service availability during an attack.The architecture is also designed to be scalable and adaptable. Additional log sources, new detection rules, or extra servers can be incorporated without major structural changes. This flexibility allows the system to evolve with future technologies, including machine learning-based anomaly detection or external threat intelligence integration.Overall, the project demonstrates an effective approach to detecting and mitigating DDoS attacks through data analysis and proactive monitoring. The combination of log processing, analytical visualization, deception techniques, and automated response mechanisms improves network visibility and reduces the impact of malicious traffic. The results highlight that integrating monitoring tools with intelligent defense strategies can significantly strengthen an organization’s ability to handle modern DDoS threats.

IX. ACKNOWLEDGEMENT

First and foremost, we thank God Almighty for blessing us immensely and empowering us at times of difficulty like a beacon of light. Without His divine intervention, we wouldn't have accomplished this project without any hindrance. We are also grateful to the Management of Theem College of Engineering for their kind support. Moreover, we thank our beloved Principal Dr. Riyazoddin Siddiqui, our Director, Dr. N.K. Rana for their constant encouragement and valuable advice throughout the course. We are profoundly indebted to Prof. Raees Ahmad, Head of the Department of Computer Engineering, and Prof. Jagruti More, Project Coordinator, for helping us technically and giving valuable advice and suggestions from time to time. They are always our source of inspiration. Also, we would like to take this opportunity to express our profound thanks to our guide Prof. Jagruti More, Assistant Professor, Computer Engineering, for his valuable advice and wholehearted cooperation without which this project would not have seen the light of day. We express our sincere gratitude to all Teaching/Non-Teaching staff members of the Computer Engineering department for their co-operation and support during this project.

REFERENCES

[1] Weiler, N. Honeypots for Distributed Denial of Service Attacks.
[2] Bellaïche, M., & Grégoire, J.-C. Avoiding DDoS with Active Management of BacklogQueues.
[3] Thileeban, A., &Nallathamby, D. J. Use of Honeypots for Mitigating DoS Attacks Targeted on IoT Networks.
[4] Das, V. V. Honeypot Scheme for Distributed Denial-of-Service Attack.
[5] Nawrocki, M., Kristoff, J., &Hiesgen, R. SoK: A Data-Driven View on Methods toDetect Reflective Amplification DDoS Attacks Using Honeypots.
[6] Sembiring, I. Implementation of Honeypot to Detect and Prevent Distributed Denialof Service Attack.
[7] Sardana, A., & Joshi, R. C. Honeypot Based Routing to Mitigate DDoS Attacks onServers at ISP Level.
[8] Shi, L., Li, Y., & Liu, T. Dynamic Distributed Honeypot Based on Blockchain.
[9] Bose, A. K., Arnob, M. F. M., & Safran, M. An Enhanced LSTM Approach for Detecting IoT-Based DDoS Attacks Using Honeypot Data.
[10] Oula, M. A., & Hamza, H. D. Detection and Mitigation of DDoS Attacks UsingEnsemble Learning and Honeypots in a Novel SDN-UAV Network Architecture.
[11] Morić, Z., Dakić, V., &Regvart, D. Advancing Cybersecurity with Honeypots and Deception Strategies.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY