iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IV **Month of publication:** April 2026

DOI: https://doi.org/10.22214/ijraset.2026.81247

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 IV Apr 2026- Available at www.ijraset.com

Pi-Pot:Portable Wireless SSH Honeypot using Raspberry Pi

Shinimol Y, Jagan U D, Soorya Kiran B, Tania Francis, Daniel Isaac E
Dept.of Computer Scienceand Engineering Rajadhani Institute of Engineering and Technology Trivandrum, India

Abstract*—Network security is a top priority for organizations to protect against cyberattacks. One effective method for monitoring and analyzing malicious activity is through the use of honeypots— deceptive systems designed to attract attackers, log their actions, and gather valuable threat intelligence. This project focuses on building a wireless and portable honeypot using a Raspberry Pi, specifically designed to detect and analyze SSH brute-force attacks. By simulating a vulnerable SSH service using the Cowrie honeypot, the system captures login attempts and attacker behavior in real time. The collected data can then be analyzed to understand attack patterns and improve security measures. Honeypots like this one generate focused, manageable logs that provide meaningful insights without overwhelming data. This lightweight and cost-effective setup offers an accessible way to study intrusion attempts and strengthen network defenses using deception-based techniques.
Keywords*—security, honeypots, Raspberry Pi, SSH brute force attack, Cowrie, deception-based techniques.

I. INTRODUCTION

In today's hyper-connected digital world, cybersecurity is a critical concern as the number and sophistication of cyberattacks continue to grow with increased reliance on digital platforms and networks. To address this, proactive defense mechanisms like honeypots are used to both protect systems and study attacker behavior. A honeypot is a decoy system designed to appear vulnerable, attracting attackers and allowing security professionals to analyze their tactics, tools, and methods. However, traditional honeypots are often costly, resource-intensive, and limited in flexibility due to their fixed deployment. This project introduces a portable, wireless honeypot using a Raspberry Pi, a compact and affordable device capable of running lightweight cybersecurity applications. It simulates an SSH service using Cowrie, an open-source honeypot that records login attempts, credentials, commands, and session activity for analysis. The system's portability and low power consumption allow deployment across various environments such as homes, offices, and public networks, enabling the collection of diverse attack data. It logs key information like IP addresses, timestamps, and attacker actions, while also providing real-time Telegram alerts and a web-based dashboard for visualizing attack patterns. Future enhancements include integrating machine learning, supporting additional services like HTTP and FTP, and improving data security. Overall, this solution offers a cost-effective, scalable, and accessible approach to studying cyber threats and strengthening cybersecurity awareness and preparedness.

II. HARDWARE REQUIREMENTS

Pi-pot required 2 main components.

A. Raspberry Pi (4 GB Recommended)

The central processing unit of the system, chosen for its small size, powerful ARM Cortex-A72 processor, sufficient RAM for running multiple processes, built-in Wi-Fi for wireless connectivity, and energy efficiency suitable for portable operation.

B. MicroSD Card (32GB or higher)

Storage medium for installing Raspberry Pi OS, honeypot software, logging attacker activities, and other system data. Higher capacity ensures ample space for logs and software updates.

III. SOFTWARE REQUIREMENTS

A. Raspberry Pi Os

The official Debian-based operating system optimized specifically for Raspberry Pi hardware, providing a stable, secure, and efficient platform for running applications and services. It includes essential tools, libraries, and drivers tailored for the Raspberry Pi, ensuring smooth hardware integration, reliable performance, and ease of use for both development and deployment of projects.

B. Cowrie Honeypt

Acting as the "trap," Cowrie is a medium-interaction honeypot. It doesn't just block attackers; it emulates a fake UNIX filesystem. It is configured to listen on a non-standard port while pretending to be a real SSH server, tricking attackers into revealing their tools and intentions.

C. Filebeats

Filebeat serves as the system's "courier." It sits quietly in the background, constantly monitoring the cowrie.json log file. The moment a new line of data is written (such as a login attempt), Filebeat instantly harvests that data and securely transports it to Elasticsearch for processing.

D. Elasticsearch

This is the "brain" of your data pipeline. It is a highly scalable search and analytics engine that stores the raw logs sent by Filebeat. It indexes the data, allowing you to search through thousands of attack records in milliseconds to find specific IP addresses or malicious commands.

E. Kibana Web Interface

Kibana is the window into your data. It provides the web interface where you can transform raw text logs into visual intelligence. You use it to build "heat maps" of attacker locations, pie charts of common passwords, and time-series graphs that show when your network is most under attack.

F. Telegram Bot

Enables seamless integration with the Telegram messaging platform to send real-time attack notifications and alerts directly to administrators' mobile devices. This ensures immediate awareness of suspicious activities, allowing for quick monitoring and response, while supporting automated message delivery for events such as login attempts and potential security breaches.

IV. SYSTEM ARCHITECTURE

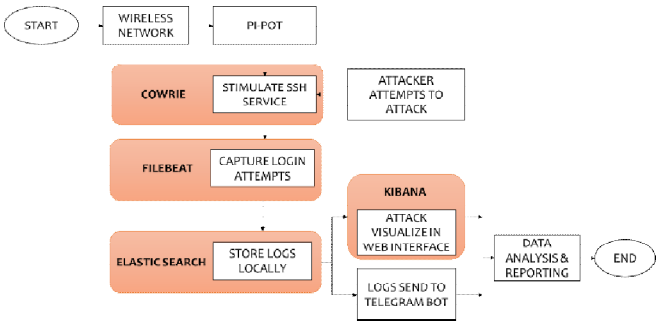


Figure 1 shows the workflow diagram of the Pi-pot system.

A. System Workflow

1) Start: The process is initiated to deploy and run the honeypot system.

2) Wireless Network: The Pi-Pot connects to a wireless network, establishing its presence and making the system accessible for remote attacks.

3) Cowrie (Simulate SSH Service): The system runs the Cowrie honeypot to simulate a vulnerable SSH service, making it appear as a real and attractive target for unauthorized users attempting to gain access.

4) Attacker Attempts to Attack: External attackers or automated bots scan the network, detect the exposed service, and attempt to breach the system using brute-force login attempts or other exploit techniques.

5) Filebeat (Capture Login Attempts):Every interaction, including login attempts and commands executed by attackers, is immediately intercepted and captured by Filebeat, which functions as a real-time log collector and forwarder.

6) Elasticsearch (Store Logs Locally): The captured data is then forwarded to Elasticsearch, where logs are efficiently indexed and stored locally, enabling fast retrieval, searching, and structured storage of large volumes of attack data.

7) Kibana (Attack Visualization in Web Interface):The stored logs are processed and visualized using Kibana, providing an interactive web-based dashboard that displays attack patterns, source locations, frequency of attacks, and other useful insights.

8) Logs Sent to Telegram Bot:Important log data and high-priority alerts are automatically sent in real-time to the network administrator via a Telegram bot, ensuring immediate awareness and quick response to ongoing attack activities.

9) Data Analysis & Reporting:The collected data is thoroughly analyzed to identify attack trends, common vulnerabilities, and behavioral patterns, which are then used to generate detailed threat intelligence reports for improving security measures.
10) End: The workflow completes after data analysis and reporting, while the system continues to operate in a continuous monitoring state, remaining ready to detect and respond to future attack cycles.

B. System Implementation

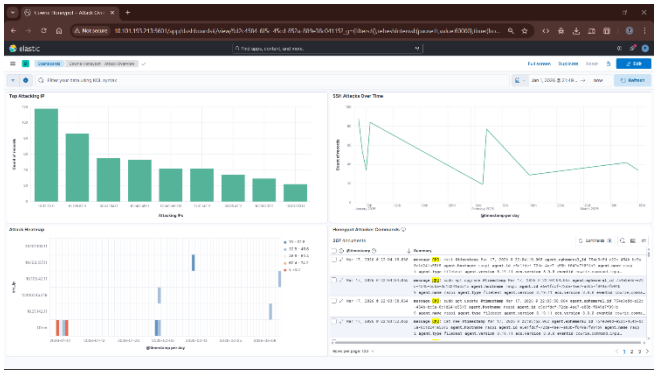
The implementation of the portable wireless SSH honeypot begins with hardware assembly using a Raspberry Pi, microSD card, and power bank to establish a stable foundation. The latest 64-bit Raspberry Pi OS is installed and configured with essential settings such as filesystem expansion, time zone, system updates, and SSH access for remote management. The device is then connected to a wireless network, and iptables is configured to redirect traffic, allowing the honeypot to capture automated attack attempts while keeping management secure. Next, the required software environment is set up by installing Python 3 and cloning the Cowrie repository along with its dependencies. The Cowrie honeypot is then configured by customizing its settings, including hostname, fake filesystem, and weak credentials to attract attackers, while enabling JSON logging for further analysis.

Following this, the ELK stack is integrated by installing Elasticsearch and configuring Filebeat to collect and forward Cowrie log data for efficient indexing and search. Kibana is then set up to visualize this data through dashboards that display attack patterns, geolocation, and frequency. A Telegram bot is implemented using a Python script to send real-time alerts for critical events such as successful login attempts. The system is tested by simulating attacks to ensure proper data capture, visualization, and alert delivery. Finally, the system is secured by disabling unnecessary services and managing log storage, after which it is deployed in various environments to collect diverse threat intelligence while being centrally monitored through the dashboard.

V. FINDINGS AND ANALYSIS

The results from the deployment of the Pi-Pot system confirm its effectiveness as a reliable and highly responsive threat intelligence tool. Once active on the network, the Cowrie software successfully simulated a vulnerable SSH service, attracting automated bots and malicious users, and capturing unauthorized login attempts with high precision, including timestamps, IP addresses, and credentials. The portability of the Raspberry Pi enabled the system to operate in various environments and detect local network threats. A key highlight was the seamless integration of the ELK stack, where Filebeat collected real-time data, Elasticsearch stored and indexed it, and Kibana transformed it into clear visual dashboards showing attack patterns, geolocations, and commonly used passwords, making it easier to identify trends such as peak attack times. The Telegram Bot alerting system further enhanced functionality by sending instant notifications for successful logins or suspicious activities, providing continuous situational awareness without manual log monitoring. Additionally, analysis of the captured data revealed that attackers heavily rely on automated scripts targeting weak credentials and known vulnerabilities. Overall, the project demonstrated that effective cybersecurity monitoring can be achieved using a low-cost, portable Raspberry Pi, successfully meeting its objectives as a flexible and powerful solution for gathering actionable threat intelligence and supporting cybersecurity education.

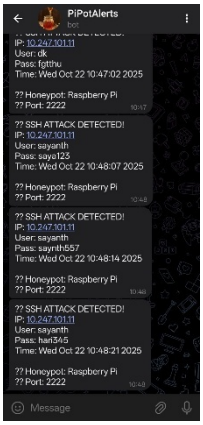
A. Kibana Web Interface



The provided dashboard illustrates the analytical capabilities of the Kibana interface in processing and visualizing raw security data captured by the Pi-Pot.

The upper-left bar chart, "Attacking IPs," identifies the most persistent sources of unauthorized trafficemerging as the most frequent attacker, followed closely by other internal network addresses. This visualization allows administrators to instantly recognize which specific nodes are responsible for the highest volume of brute-force attempts. Complementing this is the timeseries line graph in the upper-right corner, which tracks the "Count of records per 12 hours." This graph shows a significant spike in malicious activity. This data is critical for identifying peak attack windows and understanding the temporal patterns of automated botnet behavior. The lower section of the dashboard features an Attack Heatmap, which shows exactly when specific IP addresses tried to attack the system. This chart uses different colors to highlight "bursts" of activity; for instance, darker or redder areas indicate a high number of hacking attempts in a short period. By looking at this timeline, researchers can easily tell the difference between a one-time random scan and a long-term, targeted attack campaign. The bottom-right conner shows the timestamp, commands run by the attacker and so on. Together, these visuals turn thousands of messy log files into clear, useful information. They show that the system does more than just trap attackers, it organizes the data so that administrators can quickly investigate threats and identify the most dangerous or persistent hackers on the network. This makes it much easier to see patterns and strengthen network defenses against real-world threats.

B. Telegram Bot



PiPotAlerts Telegram bot in action, delivering real-time notifications for SSH brute-force login attempts detected by your Raspberry Pi honeypot. Each alert message provides comprehensive details about the detected attack, including the source IP address, attempted username, password, timestamp, and the port targeted. The format allows administrators to quickly assess the identity and frequency of attack attempts, facilitating swift responses to ongoing threats. This instant alerting capability ensures situational awareness, enabling rapid investigation and mitigation actions from any location with Telegram access. The bot greatly improves operational efficiency by consolidating critical security events into an easily accessible and timestamped chat stream, making it an effective solution for monitoring and responding to cyber threats in real time.

VI. ADVANTAGES

1) Extreme Portability: The system is highly portable and compact, allowing it to be easily carried and deployed in a wide range of environments such as cafes, offices, homes, and public networks. This flexibility enables the study of location-specific threats and variations in attack behavior across different network conditions.

2) Low Cost: The solution is built using affordable, off-the-shelf components like the Raspberry Pi and microSD card, making it a cost-effective option. This accessibility allows students, researchers, and small businesses to implement and experiment with cybersecurity tools without requiring expensive infrastructure.

3) Simple and Rapid Deployment: The system is designed for quick setup and ease of use, allowing it to be configured and activated on a new network within minutes. Minimal technical complexity ensures that even users with basic knowledge can deploy the honeypot efficiently.

4) Energy Efficient: The Raspberry Pi consumes very low power compared to traditional servers, making it highly energy efficient. This allows the system to run continuously for extended periods using a small power bank, making it suitable for long-term and remote deployments. It also reduces overall operational costs and minimizes heat generation during continuous operation.

VII. FUTURE SCOPE

The system's capabilities can be extended into a *multi-protocol honeypot* by emulating additional vulnerable services such as Telnet, FTP, and RDP, allowing it to detect and analyze a wider range of attack vectors and gather more diverse threat intelligence while supporting research on multi-protocol intrusion techniques. For improved responsiveness, an *automated alerting system* can be implemented to send notifications via multiple channels such as email and SMS whenever new attacks are detected, ensuring administrators remain informed in real time regardless of their location or preferred communication method. Additionally, integrating *machine learning algorithms* with the collected logs can help detect unusual or highly coordinated attack patterns that may evade traditional analysis, enabling dynamic adaptation to emerging threats and enhancing overall security.

Further improvements include implementing *encrypted storage* to protect captured credentials, logs, and attacker data, ensuring privacy, compliance with security best practices, and safeguarding against unauthorized access or breaches. The system can also adopt *deception-driven security* by incorporating fake data, systems, and decoys to mislead attackers and delay intrusion attempts. Moreover, expanding the honeypot to support *IoT and industrial environments* will allow monitoring of connected devices and industrial control systems, addressing the growing attack surface in modern interconnected infrastructures.

VIII. CONCLUSION

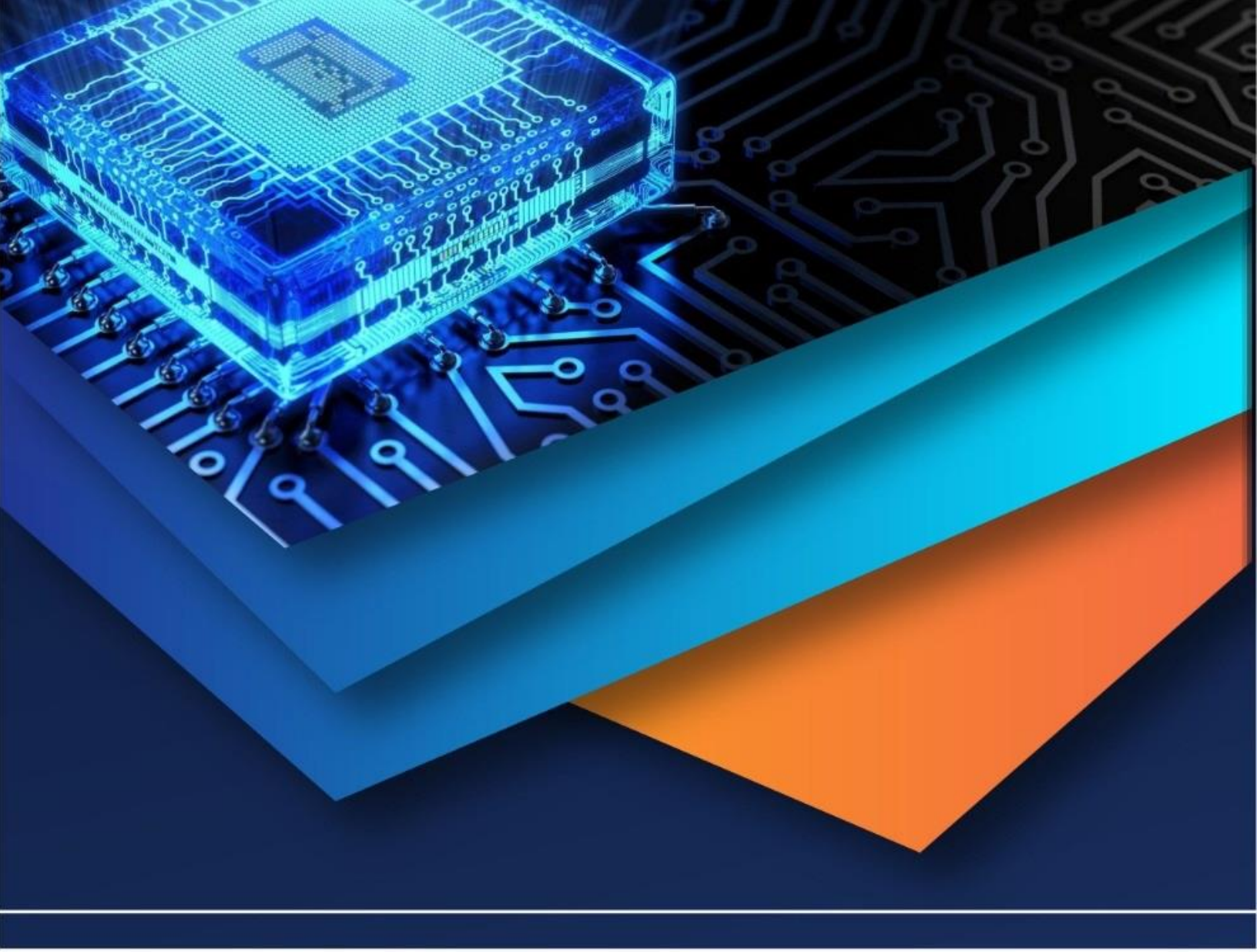
In conclusion, the Pi-Pot system represents a significant achievement in creating a portable, efficient, and accessible tool for real-world threat intelligence. By combining the Raspberry Pi with a powerful suite of open-source software, this project successfully bridges the gap between theoretical security and practical defense. While traditional honeypots are often expensive and stationary, the Pi-Pot offers a "drop-and-go" solution that is both mobile and powerful, making advanced cybersecurity research possible in any environment from corporate offices to public cafes.The core strength of the system is its seamless data pipeline, which manages information through Filebeat, Elasticsearch and Kibana. Filebeat is a lightweight "courier" monitors the Cowrie honeypot in real time. It ensures that every login attempt and command is instantly harvested, preventing data loss even if an attacker tries to hide their tracks. Elasticsearch serving as a highspeed digital archive, Elasticsearch indexes thousands of attack logs. This allows administrators to search through weeks of data in milliseconds to identify specific IP addresses or malicious patterns. Kibana is a tool transforms raw, confusing logs into clear, visual intelligence. Through Kibana, we can view geographic heatmaps of attacker locations and charts of the most common passwords, turning complex data into a clear story of network threats.The integration of a Telegram Bot provides an active layer of defense by sending instant push notifications to a smartphone. This ensures administrators are alerted to high-priority events, like a "successful" fake login, the moment they occur. Combined with its low cost and power consumption, the Pi-Pot is an ideal, budget-friendly solution for students and small businesses seeking professional-grade security. More than just a trap, the system serves as a vital research platform, providing a safe environment to observe automated botnets and malicious payloads. By making these advanced tools accessible, the project supports a proactive approach to cybersecurity. Its modular design also allows for future growth, such as adding machine learning to detect sophisticated attacks, making it a significant advancement in mobile network defense.

IX. ACKNOWLEDGMENT

The authors would like to thank [Shinimol Y, Asst. Professor] for their support. The authors also acknowledge the use of open-source resources, including the Cowrie honeypot, ELK stack like Filebeat, Elasticsearch, Kibana and a Telegram bot.

REFERENCES

[1] https://github.com/cowrie/cowrie Cowrie SSH/Telnet honeypot software for simulating vulnerable services.
[2] https://securityboat.github.io/Pentesting/Physical/Raspberry%20Pi/honeypot-usingraspberry-pi-and-opencanary./ Raspberry Pi honeypot project overview and technical steps.
[3] https://www.jetir.org/papers/JETIR2304C48.pdf Honey-Pi: A Honeypot installed on Raspberry-Pi, benchmarking and applications.
[4] https://www.raspberrypi.com/documentation/computers/os.html Official documentation for Raspberry Pi OS setup and features.
[5] https://isc.sans.edu/diary/30468 Community and modern honeypot platform improvements.
[6] HoneyPi: Raspberry Pi Honeypot Implementation: https://trustfoundry.net/2017/08/22/honeypi-easy-honeypot-raspberry-pi/
[7] Honeypot using Raspberry Pi & Opencanary: https://securityboat.github.io/Pentesting/Physical/Raspberry%20Pi/honeypot-usingraspberry-pi-and-opencanary./
[8] Telegram Bot API: https://core.telegram.org/bots/api
[9] Elastic Stack (ELK) Overview: https://www.elastic.co/what-is/elk-stack
[10] Elasticsearch: https://www.elastic.co/docs/reference/elasticsearch
[11] Kibana: https://www.elastic.co/docs/reference/kibana
[12] Filebeat: https://www.elastic.co/docs/reference/beats/filebeat.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY