Integrated Intrusion Detection System with Automated Micro-Service Recovery in Java Based Environment

1. Mohammad Jaweed Yaser, PG Scholar, Department of Information Technology , Shadan College of Engineering and Technology , Hyderabad , Telangana, India - 500086. Email :- mdjaweedyaser@gmail.com

2. Md. Ateeq Ur Rahman, Professor, Department of Computer Science and Engineering, Shadan College of Engineering and Technology, Hyderabad, Telangana, India - 500086 Email:- mail_to_ateeq@yahoo.com

3. Shaik Shakeer Basha, Professor, Department of Information Technology, Shadan College of Engineering and Technology, Hyderabad, Telangana, India - 500086 Email:- shakir.qa@gmail.com

Abstract: The IDS with Automated Micro-service Recovery is a promising solution which protects the Java based web environment and provides continuous service availability. The system uses the KNN algorithm to analyze the network data and identify any anomalies by comparing test inputs against known normal patterns, unknown patterns and attack patterns. If something is detected as being "abnormal", the system will dynamically reach out to other microservice routes with matching scores to ensure that there are no disruptions. The architecture is based on MySQL as the database management system, and Tomcat7 as the web server platform, which enables an easy integration and deployment at the enterprise level java. IDS model gains a high accuracy rate both in normal situations and in attack situations, while automatic service recovery has self-healing capabilities to increase the reliability of the system and reduce downtime. Real time monitoring and risk assessment enables administrators to take preemptive measures against network threats. Experimental testing has shown that the system can maintain a very high detection rate and provide intelligent rerouting, thus providing a scalable framework for resilient networked applications.

"Keywords: *Intrusion Detection System, K-Nearest Neighbors, Anomaly Detection, Java Web Application, Microservice Recovery, Self-Healing, Risk Scoring, Network Security."*

1. INTRODUCTION

With the emergence of web-based services and microservice architectures, the design, deployment, and scaling of today's systems have come into a vastly new paradigm. Java based web environments are the preferred choice of enterprise solutions for being platform independent, robust and well supported by an ecosystem. But, with the introduction of microservices, there's a lot of complexity in managing them, and applications are open to numerous network-level security threats like invasion, lateral attacks, and service-level anomalies. As microservices are always communicating over the networks, one small security bug can quickly spread to the dependent services, leading to larger damage and impacting critical operations [1].

The dynamic behavior of microservices poses a challenge to traditional network security techniques like as firewalls, rule-based monitoring and signature-guided intrusion detection. These methods are predominantly reactive and make use of pre-determined attack signatures and fixed criteria limiting their application in the face of zero-day threats and moving vectors. With the addition of software-defined networking, cloud-native platforms and container orchestration, visibility and

control of traffic become ever more complex and complex making real-time anomaly detection a must, not an option [2][3].

Anomaly-based intrusion detection has emerged as a popular adaptive method for detecting anomalies in normal network activity. Anomaly detection techniques expose traffic patterns and service interactions to bring to light subtle and undetected dangers. In parallel, current distributed systems focus on observability, robustness and automatic failure recovery. A cloud-based monitoring infrastructure and architectures based on telemetry offer real-time visibility of service health, performance and security events in a variety of environments [4][5].

Recent breakthroughs in artificial intelligence and graph-based learning have greatly improved the anomaly detection capabilities in microservice ecosystems. Technologies based on graph attention networks, temporal modeling and functional connectivity analysis are able to identify and localize anomalies at a fine-grained level, even within complex service relationships [6][7][8]. These smart approaches offer proactive threat detection while reducing false-positive instances and increasing system awareness. Besides, AI-enabled mitigation technologies respond automatically to identified risks, minimizing human delay in response and intervention [9][10].

The major purpose is to create a secure and robust Java web environment by way of continuous network monitoring, accurate anomaly detection and automated service recovery. The solution aims to deliver continuous service availability, reduce the time to security response, detect cascaded failures and enhance overall reliability, while defending applications built using microservice architectures from today's network threats.

2. RELATED WORK

Cloud-native and microservices-based systems have become more complex and a lot of research has been conducted in the area of intelligent security methods to identify anomalies and ensure service reliability. Recent research highlights that perimeter security is not enough to protect distributed microservices, where the attack is likely to be subtle behavioural changes rather than a clear signature. Therefore, AI-driven anomaly detection has become an important way to enhance the security of cloud-native applications by learning the usual behavior of services continually and spotting malicious or aberrant activity in real time [11].

There are several pieces of work that have a goal of describing the nature of anomalies in microservice ecosystems. The extensive research shows that there are many data sources - such as logs, metrics, traces - that collectively measure the system health. The ability to correlate multiple sources can help increase the accuracy of detection and context of the assaults and faults within the anomaly detection systems. Study of this area has shed light on the ability to scale and adapt monitoring for service interactions that vary in their dynamic nature through the use of statistical and ML techniques across dispersed systems [12].

The availability of representative datasets has also played an important influence in the success of anomaly detection studies. Microservice API logs and metrics from open datasets are good benchmarks for assessing detection methods. These datasets are normal and abnormal operating patterns, and can be used to try out supervised and unsupervised learning techniques and also for repeatability in security research [13]. These tools are especially beneficial for verifying intrusion

detection approaches in scenarios that closely mirror real-world deployments.

In practice, no labeled attack data exists, which has resulted in growing interest in unsupervised anomaly detection. In this place, the research is focused on clustering, density estimation and reconstruction based methods for anomaly detection, without the knowledge of attack signatures. These techniques work very well when developing microservices' behaviors that evolve frequently due to scalability, upgrades, and dynamic routing [14]. Unsupervised models – flexible and can be useful for identifying unusual threats and performance anomalies.

Though detection continues to be important, current efforts point towards the use of root cause analysis and common security architectures. In the integrated techniques, dependency analysis is used in conjunction with anomaly detection for identification of failure or attacks in dependent systems. Rather than single alarms for problem response, these frameworks deliver actionable information to shorten the time to diagnosis and/or decrease the operational overhead in a large scale microservice installation [15].

Security research has also been conducted on microservices, from a more general software engineering viewpoint. Security conversations and evolution of vulnerabilities reflect on typical vulnerabilities for the microservice communication, configuration and deployment processes. These can be used to discover common security problems and to create more robust detection and mitigation methods [16]. Systematic mapping research is helpful to identify and categorize the current state of knowledge of microservice security, thereby merging the methodologies, barriers and open issues for further research in the area [17].

More security solutions for microservices are starting to feature resilience and recovery. Studies on disaster recovery processes, Kubernetes orchestration, and cloud security procedures validate the importance of automated response solutions. These techniques help to isolate affected services quickly, reroute them and restore them to normal use thereby reducing downtimes and making them available [18].

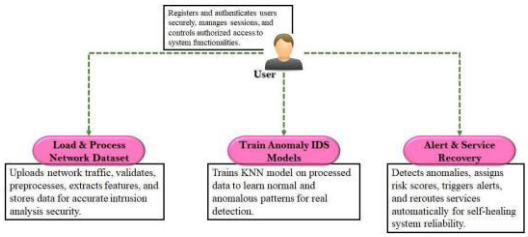
The self-healing system idea has been gaining traction, with the addition of AI-based anomaly detection. Such systems automatically react to recognized anomalies by initiating corrective measures, like restarting services, rerouting traffic, or reallocating resources. Empirical studies have shown that self-healing methods enhance system resilience and reduce the need for manual intervention in addressing security threats [19].

Finally, DL-based cybersecurity risk assessment is extended by frameworks of the possible impact and prioritization of the response to the anomaly detection. It is these techniques that are used to assess the risk in microservice architectures, so that appropriate decisions are made, and the right defensive measures are implemented in complex cloud environments [20].

New studies have demonstrated the application of smart systems in security and automation. Slamani et al. [26] stress on human-centric industrial robots for adaptive automation and system efficiency. Kaushik et al. [27] present a discussion on how blockchain can be combined with artificial intelligence to better secure IoT. These, in combination, give rise to secure, intelligent and durable distributed computing environments.

3. MATERIALS AND METHODS

The suggested system combines an intelligent IDS and automatic microservice recovery to improve security and ensure uninterrupted operations in Java based online environments. It uses KNN algorithm for analyzing network traffic , detecting anomalies , classifying events as normal or malicious by comparing with known patterns . With the system, risk scores are calculated by monitoring network data in real-time, enabling proactive mitigation of possible risks. When a defect or attack occurs in the main service path, the system immediately uses alternate microservices to provide the service without interruption and to ensure the system is self-healing. Efficient data storage is achieved with the help of MySQL while the Tomcat7 with its scalable deployment capabilities secures stable web operations. Uploading data-sets, training an IDS model, and visualizing real-time anomaly detection and service rerouting, is done with an intuitive user interface, which is solid for enterprise-level applications [21][22][23].



“Fig.1 Proposed Architecture of the Integrated IDS System”

The workflow for a user-centric IDS is shown in Fig. 1. Secure user authentication and session management gets started. It then branches off into three fundamental components: loading and preparing network data for feature extraction, training a KNN model to identify normal and abnormal traffic patterns, and automatically triggering alarms and service recovery to ensure the stability of the self-healing system.

i) Load & Process Network Dataset:

Load & Process Network Dataset is important network dataset module in the system, where the users can load the network traffic data to the system for analysis. After uploading the system checks and validates the data on a large scale to make sure that all data is filled properly, are correct and are in the right format; which decreases the scope of errors during further analysis. Some preprocessing methods are used to clean the data, reduce noise, handle missing values and normalize relevant characteristics in order to make sure that the dataset is consistent and adequate for anomaly detection. The module also extracts some key characteristics from the raw network data and converts it to a structured format, which can be efficiently processed by the IDS. The data is then securely stored in the database, ensuring data integrity and easy access for real-time monitoring and analysis. The module's processing and organization of network data improves the performance and reliability of the IDS, making it more capable of detecting anomalous patterns, and the provision of proactive security management within the java based Web environment.

ii) Train Anomaly IDS Models:

The Train Anomaly IDS Models module creates an intelligent model for detecting and identifying normal and abnormal network behavior. The system leverages on this processed and validated network data to identify pattern and correlation in the traffic data, and captures the features of the routine operations and potential anomalies. By training the model to identify these subtle shifts, it can enhance its real-time threat detection capabilities, improving its ability to detect malicious activity or system failures. This module also fine-tunes the model parameters to boost classification accuracy and also

minimize false-positive false-negative rates, for dependable network event analysis. The model leverages past and current data to gain knowledge and gain predictive capabilities to make decisions proactively. Once trained, it can accurately classify network events as they come in, and rate the risk of each event so that automated responses can be made. This guarantees that the operations of microservices are durable, safe and can recover from abnormal scenarios.

iii) Alert & Service Recovery:

The Alert & Service Recovery module continuously evaluates network events as they happen and identifies anomalies in the incoming data that might be a security threat or cause issues within the network. Each event recorded is given a risk score that measures how severe and likely an attack or failure is. In case of anomaly with high risk, the other microservices are automatically activated to reroute the traffic, thereby avoiding any disruptions in the operations and lack of service continuity. This self-healing ability means that essential services can remain operational during an attack and if the system fails. Meanwhile, alerts are generated and sent to administrators, providing real-time information on hazards detected and making it possible to have some human monitoring when necessary. The module is intended to be designed in a way that will make it proactive in detecting anomaly and automatically recover from it in order to increase the resilience, dependability and security of the online Java environment. It ensures that the system remains operational and secure during failures, provides strong and resilient network management, and minimises downtime and safeguards sensitive data.

iv) Algorithm:

KNN is one of the most popular supervised ML algorithms which is used for classification and regression by measuring the similarity between the data points. The approach involves calculating the distances between a test instance and every training example, usually via some distance measure such as Euclidean distance or Manhattan distance, and determining the closest neighbors. K is a constant number of closest neighbours . The majority label of these neighbours is then used to determine the class of the test instance. KNN has been effectively applied in the context of network security to detect anomalies based on comparison with known normal and malicious patterns in the incoming network traffic [24]. This allows the system to give a risk score to every network event, accurately identify potential risks and support proactive risk mitigation. As KNN is simple and can be used to perform real-time analysis for continuous monitoring, automatic decision making, and maintaining service stability in online Java based applications, it is suitable for dynamic contexts [25].

4. RESULTS AND DISCUSSIONS

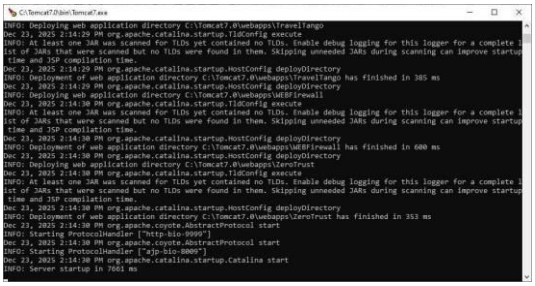


Fig.2 Tomcat7 Server

In above fig 2, you can see that java web server is started. Open browser and type: http://localhost:8080/MicroserviceIDS/index.jsp and press enter key to get the following page.

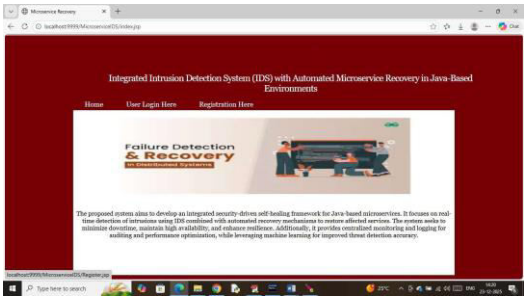


Fig.3 Home Page

Click on the 'Registration Here' link in above fig 3 to get below page.

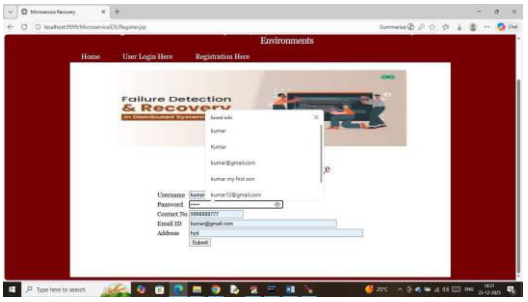


Fig.4 Registration

Here, user enters sign up details and then clicks on the button to reach the below page: In above fig 4 user is entering sign up details and after that click button to reach below page

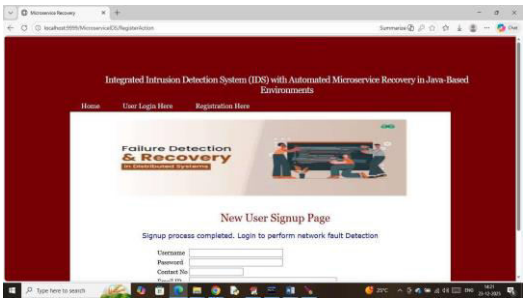


Fig.5 Registration Process Completed

In above fig 5, user sign up task accomplished and now click on 'User Login' link to arrive at below page.

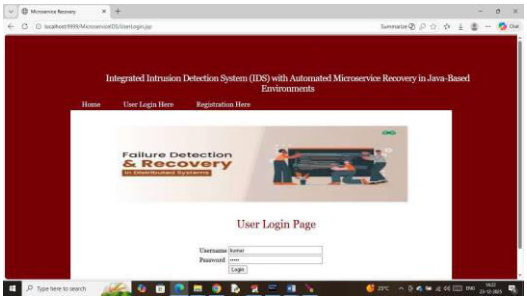


Fig.6 User Login

After login it will display below page in fig 6 above.

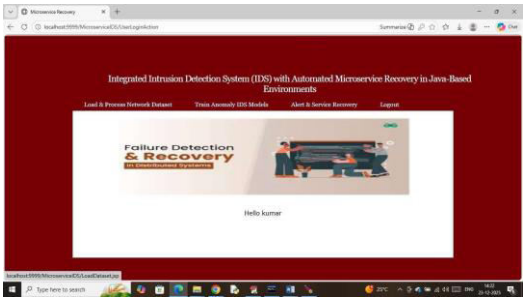


Fig.7 Main Page

Click on 'Load & Process Network Data' link in the fig 7 above and you will be directed to the following page:

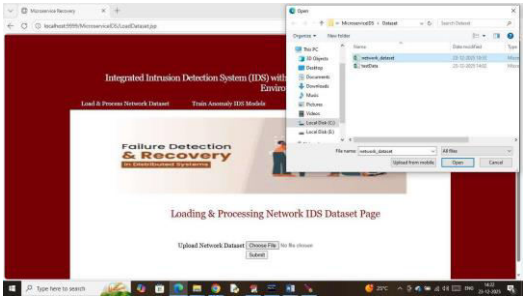


Fig.8 Upload Network Dataset

After uploading a dataset as shown in above fig 8, click buttons to get below page.

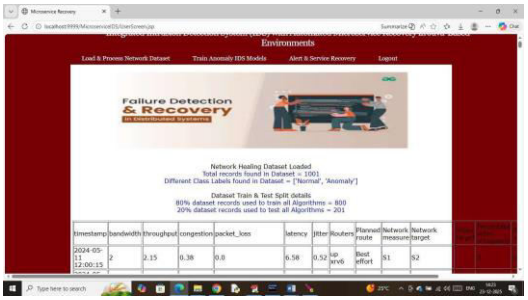


Fig.9 Dataset Loaded - Result

Now click on the 'Train Anomaly IDS Model' link in Fig 9 above and you will get the following page; the model will be trained.

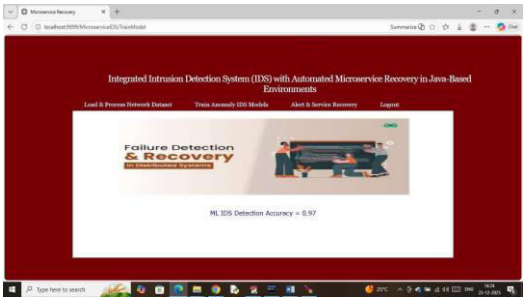


Fig.10 ML IDS Detection Accuracy

In above fig 10, java based KNN IDS model got a accuracy rate of 97% now click on 'Alert & Service Recover' link to view below page.

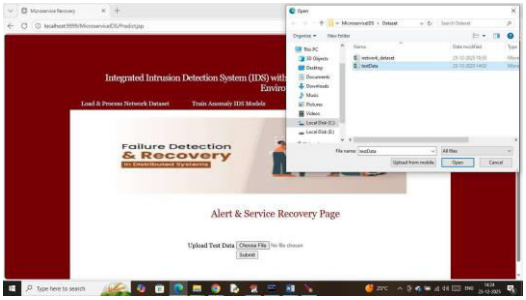


Fig.11 Upload Test Data

Pick and upload test data in fig 11 above and click on any button to get below page Fig 12.

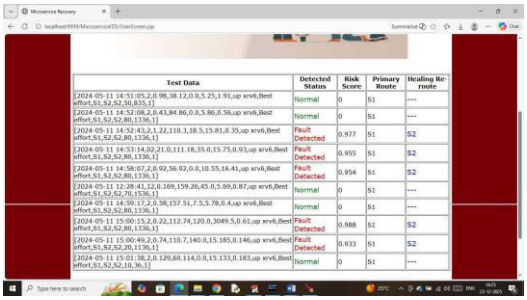


Fig.12 Test Data Result – Normal or Fault Detected

In above fig 12 network test data can be seen in 1st column and in 2nd column detected status and risk score will be seen, If detected as 'Normal' then it will continue with 'Primary Route Service' and if detected as 'Fault' in 'Primary Route' it will take 'Alternate route' as 'S2' to show as 'Self Healing Concept'.

5. CONCLUSION

The Integrated IDS with Automated Micro-service Recovery is a highly effective solution to enhance security and reliability in online environment based on Java. The KNN algorithm is employed to efficiently detect abnormal network behavior by comparing the data received with the data that is known as a normal case or attack case. A high detection accuracy of 97% is achieved, which guarantees correct classification of the events within a network with fewer false positives. The automatic service recovery method is enabled at all times, as it dynamically switchs to the alternative microservices when abnormalities are found in the primary service route, as a self-healing capability. Combined, MySQL for database management and Tomcat7 for web deployment offer a powerful and scalable platform for enterprise applications. Real-time monitoring and risk scoring and analysis enable administrators to get actionable intelligence about the health of their network to aid in quick threat response. Overall, the system successfully combines

anomaly detection and smart service management functionality to provide a robust and secure architecture that ensures operational stability, reduces downtime, and improves the security of the Java-based Web services.

For future research, more sophisticated ML and DL models can be explored, such as RF, SVM, or LSTM, to further enhance the effectiveness of anomaly detection and to better handle the changing nature of attacks. We could incorporate into them real-time threat information feeds and automated threat prioritization to provide more proactive defensive systems. With distributed microservices and cloud based deployments, the scalability and fault tolerance would also be enhanced. Predictive analytics and behavioral profiling can assist in uncovering advanced and zero-day threats. Furthermore, a graphical dashboard for real-time visualization of network status, anomalies and service recovery actions will make the system more user-friendly for administrators and intelligent and adaptive, providing complete network security management.

REFERENCES

[1] Mohottige, T. I., Polyvyanyy, A., Buyya, R., Fidge, C., & Barros, A. (2024). Microservices-based Software Systems Reengineering: State-of-the-Art and Future Directions. arXiv preprint arXiv:2407.13915.

[2] Rahim, J. A., Nordin, R., & Amodu, O. A. (2024). Open-Source Software Defined Networking Controllers: State-of-the-Art, Challenges and Solutions for Future Network Providers. Computers, Materials & Continua, 80(1).

[3] Abdelfattah, A. S., Cerny, T., Yero Salazar, J., Li, X., Taibi, D., & Song, E. (2024). Assessing evolution of microservices using static analysis. Applied Sciences, 14(22), 10725.

[4] Choudhury, S., & Liibaan, L. (2025). Cloud-based Observability in Distributed Systems: Designing a scalable observability architecture in the cloud using Azure, OpenTelemetry and. NET.

[5] Seroukhov, S. (2024). MICROSERVICES DESIGN PATTERNS WITH JAVA. Bpb Publications.

[6] Akmeemana, L., Attanayake, C., Faiz, H., & Wickramanayake, S. (2025). GAL-MAD: Towards explainable anomaly detection in microservice applications using graph attention networks. arXiv.

[7] Zhang, Q., Lyu, N., Liu, L., Wang, Y., Cheng, Z., & Hua, C. (2025). Graph neural AI with temporal dynamics for comprehensive anomaly detection in microservices. arXiv.

[8] Winchester, G., Parisis, G., & Berthouze, L. (2025). FC-ADL: Efficient microservice anomaly detection and localisation through functional connectivity. arXiv.

[9] Singh, S. (2025). Integrating AI-Based anomaly detection with MPK-isolated microservices for proactive security in optical networks. Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol, 11(2), 1693–1704.

[10] Ramamoorthi, V. (2024). Anomaly detection and automated mitigation for microservices security with AI. Applied Research in Artificial Intelligence and Cloud Computing, 7(6), 211–222.

[11] Enhancing cloud-native application security using AI-driven microservice anomaly detection. (2025). Journal of Artificial Intelligence for Data-Driven Discovery, 9, 32–41.

[12] Nobre, J., Solteiro Pires, E. J., & Reis, A. (2023). Anomaly detection in microservice-based systems. Applied Sciences, 13(13), 7891.

[13] LO2: Microservice API anomaly dataset of logs and metrics. (2025). arXiv.

[14] Unsupervised anomaly detection in microservices. (2025). Upubscience.

[15] Meyer, O., Johnson, E., & Brown, J. (2025). A unified framework for anomaly detection and root cause analysis in microservice systems. Computer Life.

[16] Automated identification of security discussions in microservices systems. (2021). Journal of Systems and Software, 181, 111046.

[17] Securing microservices and microservice architectures: A systematic mapping study. (2021). Computer Science Review, 41.

[18] Disaster recovery and application security in microservices: exploring Kubernetes and cloud solutions. (2024). Journal of Artificial Intelligence and Big Data.

[19] Self-healing microservices and AI-based anomaly detection system. (2025). International Research Journal of Modernization in Engineering Technology and Science.

[20] Towards deep learning enabled cybersecurity risk assessment for microservice architectures. (2025). Cluster Computing.

[21] NetMoniAI: An agentic AI framework for network security & monitoring. (2025). arXiv.

[22] Anomaly Detection in Microservice-Based Systems. (2023). MDPI Applied Sciences article.

[23] AI-driven intrusion detection and prevention systems to safeguard 6G networks. (2025). Scientific Reports.

[24] Clustered federated learning architecture for network anomaly detection in large scale IoT. (2023). arXiv.

[25] Anomal-E: A self-supervised network intrusion detection system based on graph neural networks. (2022). arXiv.

[26] Slamani, M., Louhichi, B., Arslane, M., & Alawad, M. A. (2026). Smart industrial robots in the transition from automation to human-centric sustainable systems: a comprehensive review of technologies, challenges, and future directions. The International Journal of Advanced Manufacturing Technology, 1–28.

[27] Kaushik, D., Gulia, P., Gill, N. S., Yahya, M., Shukla, P. K., & Shreyas, J. (2026). Synergizing blockchain and AI to fortify IoT security: a comprehensive review. Artificial Intelligence Review, 59(2), 41.