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.2026.78416

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

AI-Driven Secure Cryptographic Key Management for Cloud Environments

T. Suganya[1], J. Abarnaa Jawahar[2], K. R. Sruthi[3], Dr. J. S. Kanchana[4]
[1] *Assistant Professor, Department of Computer Science and Engineering (Cyber Security)*
[2 3] *Students, Department of Computer Science and Engineering (Cyber Security)*
[4] *Professor and Head, Department of Computer Science and Engineering (Cyber Security)*
K.L.N. College of Engineering, Pottapalayam, Sivagangai.

Abstract: ***The rapid adoption of cloud platforms across industries has transformed how organizations store and process sensitive information at scale. However, protecting sensitive information stored in cloud environments remains a significant challenge. Cryptographic keys are fundamental components of encryption mechanisms used to secure cloud data. If these keys are compromised or mismanaged, attackers may gain unauthorized access to confidential information. Traditional key management systems typically rely on manual monitoring and periodic key rotation policies. Such static approaches often fail to detect suspicious key usage patterns in real time, leaving compromised keys active for extended periods and increasing security risks. This research proposes an AI-driven cryptographic key management framework designed to enhance security in cloud environments.***

The system integrates Advanced Encryption Standard (AES) encryption with machine learning–based anomaly detection to continuously monitor key usage behavior. An Isolation Forest algorithm analyzes key access patterns and identifies abnormal activities that may indicate potential compromise.

Once an anomaly is detected, the system automatically triggers a zero-downtime key rotation process using workflow automation through the n8n platform. Cloud key management services ensure secure storage and lifecycle management of cryptographic keys, while all key operations are logged for auditing and compliance purposes. Experimental evaluation shows that the proposed system achieves an anomaly detection accuracy of 96.2% and reduces key exposure risk by approximately 35%. By combining intelligent monitoring with automated response mechanisms, the framework significantly improves the security resilience and reliability of cloud infrastructures.

Keywords: Cloud Security, Cryptographic Key Management, AES Encryption, Isolation Forest, Machine Learning, Automated Key Rotation

I. INTRODUCTION

Cloud platforms today serve as the primary infrastructure for managing critical business data, handling everything from financial records to personal user information across geographically distributed systems. With the increasing adoption of cloud environments for critical business operations, securing sensitive information has become a fundamental concern. Cryptographic techniques protect data in cloud systems, where encryption keys serve as the primary mechanism for protecting data privacy and preventing unauthorized access [1][10]. Effective management of these keys is essential for cloud security.Traditional key management often relies on static configurations or periodic rotation, which may leave keys active for extended periods, increasing risk [3][4]. Conventional systems also typically lack intelligent monitoring to detect anomalous usage. Recent advances in AI and machine learning offer opportunities to strengthen key management. ML-based anomaly detection can analyze key access behavior and identify suspicious patterns [5][6]. The Isolation Forest algorithm efficiently detects anomalies in large datasets, enabling proactive detection of risky keys.Automation technologies are vital for executing complex security operations. Automated workflow orchestration allows dynamic responses to threats without service disruption. Automated key rotation can replace compromised keys in real time while maintaining availability [7][8]. When combined with secure cloud-based KMS, this ensures proper lifecycle management, access control, and auditing.

Motivated by these challenges, this work led to the development of an innovative solution SecureKey, an AI-driven cryptographic key management system. SecureKey integrates AES encryption, Isolation Forest–based anomaly detection, and automated key rotation workflows via n8n. The system monitors key usage, identifies potential risks, and initiates zero-downtime rotation. By combining intelligent monitoring, automated lifecycle management, and secure storage, SecureKey enhances cloud security,

reduces manual intervention, and provides a robust, scalable solution for modern cloud infrastructures.

A. *Contributions of This Work*

The major contributions of this research are summarized as follows:

1) Design of an AI-driven cryptographic key management framework for cloud environments.
2) Integration of Isolation Forest anomaly detection to monitor key usage patterns.
3) Implementation of automated key rotation using workflow orchestration to reduce manual intervention.
4) Development of a secure monitoring and auditing mechanism for key lifecycle management.
5) Experimental evaluation demonstrating improved detection accuracy and reduced key exposure risk.

II. PROBLEM STATEMENT

Although encryption technologies are widely deployed in cloud infrastructures, many organizations still rely on traditional key management approaches that involve manual monitoring and scheduled key rotation. These approaches suffer from several limitations:

1) Delayed detection of suspicious key usage
2) Dependence on manual administrative actions
3) Increased risk of prolonged key exposure
4) Limited scalability in large distributed cloud systems

An intelligent system capable of detecting abnormal key usage patterns and responding automatically to potential threats is therefore required to strengthen cloud security.

III. RELATED WORK

Cloud security has been a growing area of research for many years. Researchers have explored different ways to protect data stored in cloud environments, with encryption and key management being among the most studied topics. Armbrust et al. [1] were among the first to clearly explain the challenges of cloud computing, pointing out that keeping data secure and controlling who can access it are two of the biggest problems cloud systems face.

Their work helped researchers understand why protecting encryption keys is just as important as choosing a strong encryption algorithm. The encryption algorithm used in this system, AES, was developed by Daemen and Rijmen [4] and was later officially adopted by NIST [3] as the global standard for symmetric encryption. AES is widely used today because it is fast, reliable, and difficult to break. However, even the strongest encryption becomes useless if the keys protecting it are left unmanaged or exposed for too long.

Subashini and Kavitha [5] studied security problems across different types of cloud services and found that most cloud providers do not monitor their encryption keys actively. Instead, they rely on fixed schedules to rotate keys, which means a compromised key could remain active for days or even weeks before anyone notices. This finding directly motivates the need for a smarter, more responsive key management approach.

Scarfone and Mell [6] introduced methods for detecting unusual system behavior automatically. Although their work focused on network traffic, the idea of watching for suspicious patterns and responding quickly applies equally well to monitoring how encryption keys are being used.

Liu, Ting and Zhou [2] built on this idea by introducing the Isolation Forest algorithm, a machine learning technique that is very good at spotting unusual events in large amounts of data. Instead of learning what is abnormal, it works by identifying data points that are easy to separate from the rest, making it both fast and accurate for detecting risky key usage.Kamara and Lauter [9] showed how encryption and key storage can be combined into secure cloud architectures that keep keys safe while still allowing controlled access.

Singh and Chatterjee [7] took this further by showing that combining AI monitoring with automated security responses can significantly reduce the time a compromised key stays active.

The system proposed in this paper builds on all of this prior work by bringing together Isolation Forest anomaly detection and n8n workflow automation to create a fully automated key management system that responds to threats in real time without any human intervention.

IV. COMPARATIVE ANALYSIS WITH EXISTING RESEARCH

Table I— Comparative Analysis with Existing Research

Research Work	Approach	Key Management	Detection Method	Limitations
Hu & Wang (2019)	Information sharing platform	Static key storage	Rule-based monitoring	Limited anomaly detection
Zhang et al. (2020)	Attribute-based encryption	Policy-based keys	Access control policies	Complex key distribution
Patel et al. (2021)	Cloud security framework	Manual key rotation	Log analysis	Slow response
Singh & Kumar (2022)	Intrusion detection system	Network monitoring	ML-based IDS	Does not manage keys
Proposed System	AI-driven framework	Dynamic key lifecycle	Isolation Forest	Requires training data

V. SYSTEM ARCHITECTURE

The system architecture consists of several modules designed to provide secure and automated cryptographic key management.

1) The encryption module generates AES-based cryptographic keys for securing cloud data.
2) These keys are stored within a cloud key management service to ensure secure storage and controlled access.
3) A monitoring module continuously records key usage logs, including timestamps, access requests, and system interactions.
4) The machine learning module analyzes these logs using the Isolation Forest algorithm to detect abnormal patterns.
5) When suspicious behavior is identified, the automation module triggers a key rotation workflow using the n8n automation platform.
6) Finally, a logging and auditing module records all key lifecycle events for compliance monitoring and security auditing.

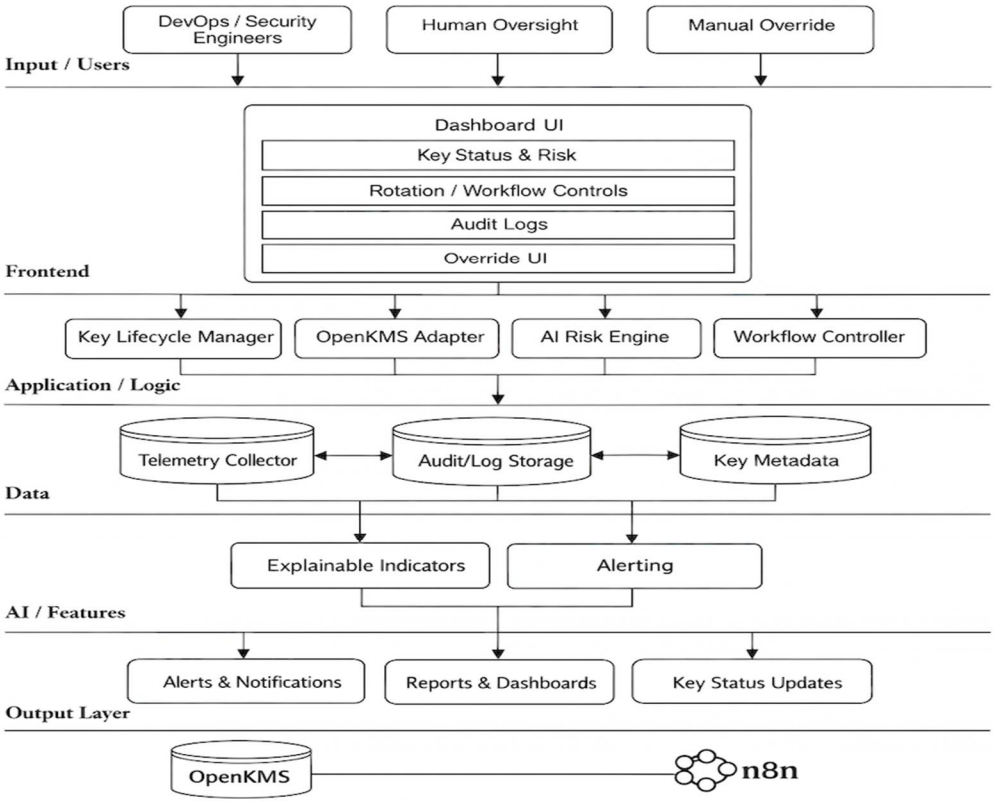


Fig. 1. System Architecture of Proposed AI-Driven Cryptographic Key Management System

VI. SYSTEM FLOW DIAGRAM

The system flow diagram illustrates the sequential operation of the proposed SecureKey framework. Encryption key metadata is collected from Cloud KMS and passed through preprocessing, behavioural analysis, and risk evaluation stages. When a key is identified as high risk, the system automatically initiates key rotation through the n8n workflow platform, generates a new AES key, and replaces the compromised key safely. Low-risk keys continue to be monitored in a continuous feedback loop. All operations are reported to the Admin and Security Team upon completion.

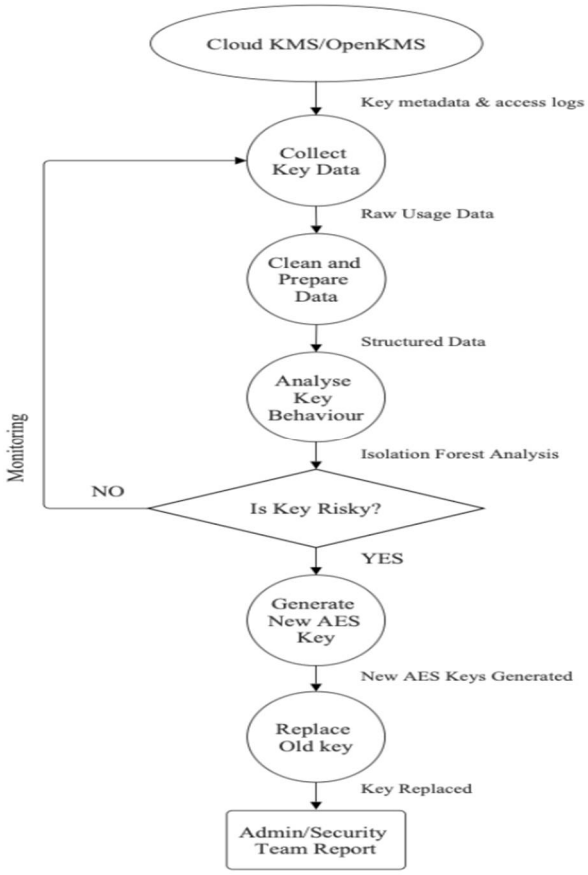


Fig. 2. System Flow Diagram of the SecureKey Framework

VII. IMPLEMENTATION

The SecureKey system was built by combining several technologies that work together to monitor, detect, and respond to encryption key threats automatically. AES-256 encryption was used to generate strong cryptographic keys, which were stored and managed through OpenKMS, an open standard key management interface that handles key creation, storage, access control, and retirement. All interactions with keys were recorded in a structured audit log so that every action could be traced and reviewed later for compliance purposes. The machine learning part of the system was built using Python 3.13 and the Scikit-learn library. The Isolation Forest model was trained on normal key usage data, learning what regular access patterns look like in terms of how often keys are used, how old they are, and how frequently they are accessed.Once trained, the model continuously watches incoming key access events and gives each one an anomaly score. Any key that scores above 0.7 is flagged as high risk and sent to the automation layer for immediate action. When a high-risk key is detected, the n8n workflow automation platform takes over. A pre-configured automation pipeline receives the alert, generates a fresh AES-256 key, migrates all services that were using the old key to the new one, and safely retires the compromised key — all without stopping or interrupting any running services. This entire process happens automatically without any administrator needing to step in. The system was tested using simulated cloud key access logs that contained a mix of normal and suspicious usage patterns.

The testing measured how accurately the system could detect risky keys, how quickly it could complete a full key rotation, how often it raised false alarms, and whether services remained available throughout the rotation process.

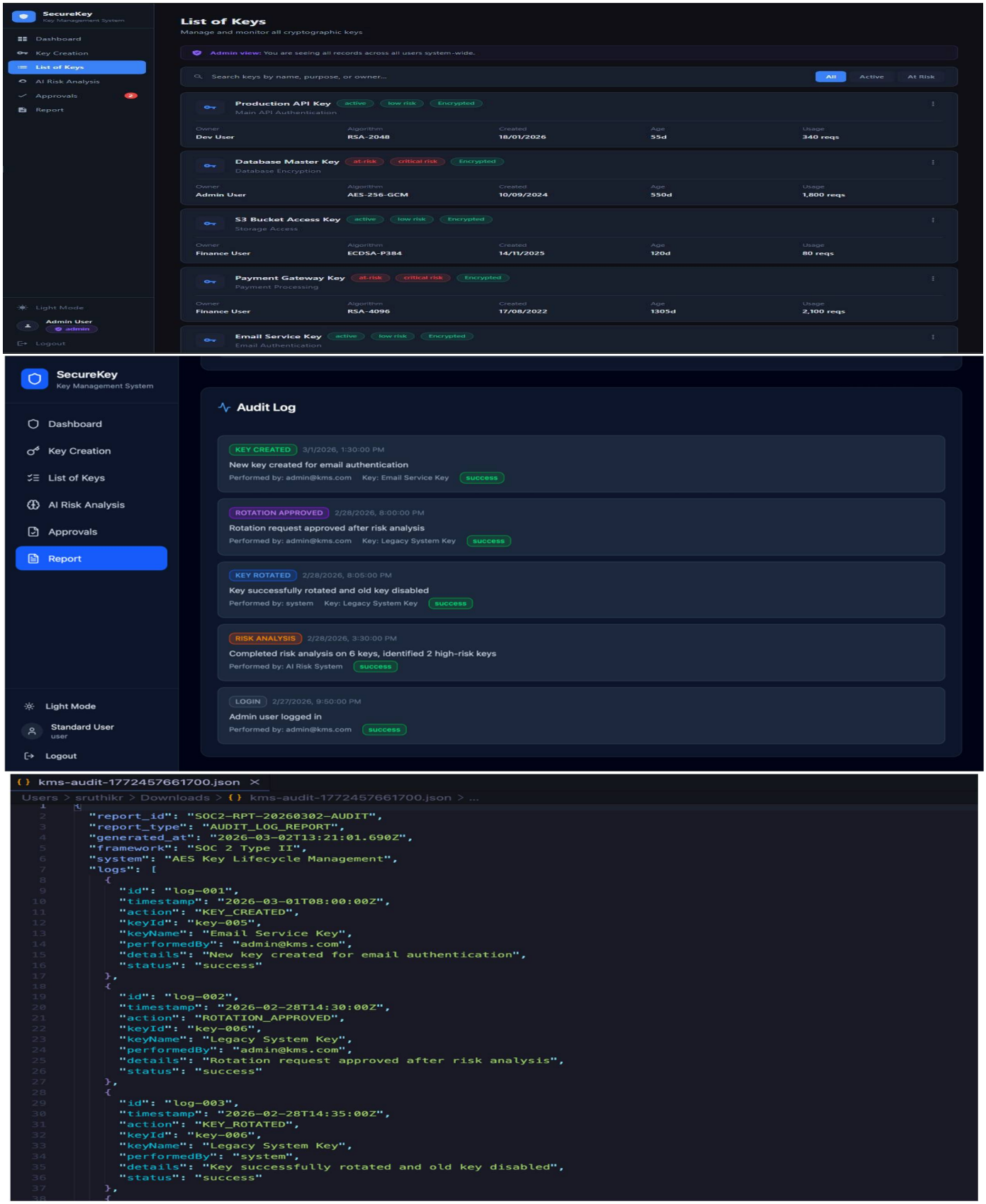


Fig. 3. Key Management Dashboard, Audit Logs, and Rotation Report

VIII. EXISTING SYSTEM VS PROPOSED SYSTEM

Table II — Feature Comparison: Existing System vs. Proposed System

Feature	Existing System	Proposed System
Key Rotation	Manual or scheduled	Intelligent automated rotation
Monitoring	Basic logging	Continuous monitoring
Threat Detection	Manual investigation	Machine learning detection
Response Time	Slow	Real-time automated response
Security Level	Moderate	High
Scalability	Limited	Highly scalable

IX. ALGORITHM

Algorithm: AI-Based Key Risk Detection and Automated Rotation
Input: Key access logs
Output: Secure key rotation
1. Initialize monitoring module
2. Collect key access logs continuously
3. Preprocess log data
4. Train Isolation Forest model
5. For each key access event
6. Calculate anomaly score
7. If anomaly score > threshold
8. Mark key as suspicious
9. Trigger automated workflow
10. Generate new AES encryption key
11. Replace compromised key
12. Log rotation event
13. Continue monitoring

X. MATHEMATICAL MODEL

The system works with a collection of key access events. Each event captures details about how a particular key was used, such as how frequently it was accessed, how old it is, and whether its usage pattern has changed recently. Together these events form the dataset that the machine learning model analyses.The Isolation Forest algorithm detects unusual events by trying to separate, or isolate, each data point from the rest of the dataset. Normal events are hard to isolate because they behave similarly to many other events. Suspicious events are easy to isolate because they stand out from the crowd. The algorithm measures how quickly each event can be separated and uses this to calculate an anomaly score.The anomaly score is calculated as: $s(x, n) = 2^{(-E(h(x)) / c(n))}$
In simple terms:

- $h(x)$ is how many steps it takes to isolate a particular key access event in one decision tree
- $E(h(x))$ is the average number of steps across all trees in the forest
- $c(n)$ is a correction factor based on the size of the dataset, used to make scores comparable regardless of how much data is being analysed. The correction factor is calculated as: $c(n) = 2H(n-1) - (2(n-1)/n$

where
$H(i) = \ln(i) + 0.5772156649$ is the harmonic number. The score always falls between 0 and 1. A score close to 1 means the event is very unusual and the key should be rotated immediately.

A score close to 0.5 means the event looks normal and no action is needed. A threshold of 0.7 was chosen for this system after testing, as it gave the best balance between catching real threats and avoiding unnecessary false alarms. If $s(x, n) > 0.7 \rightarrow$ the key is flagged as high risk and automatic rotation is triggered.

XI. PERFORMANCE EVALUATION

Table III — Performance Evaluation

Metric	Existing System	Proposed System
Detection Accuracy	82%	96.2%
Key Rotation Time	8 sec	1.8 sec
False Positive Rate	9%	3.4%
System Availability	98%	99.8%

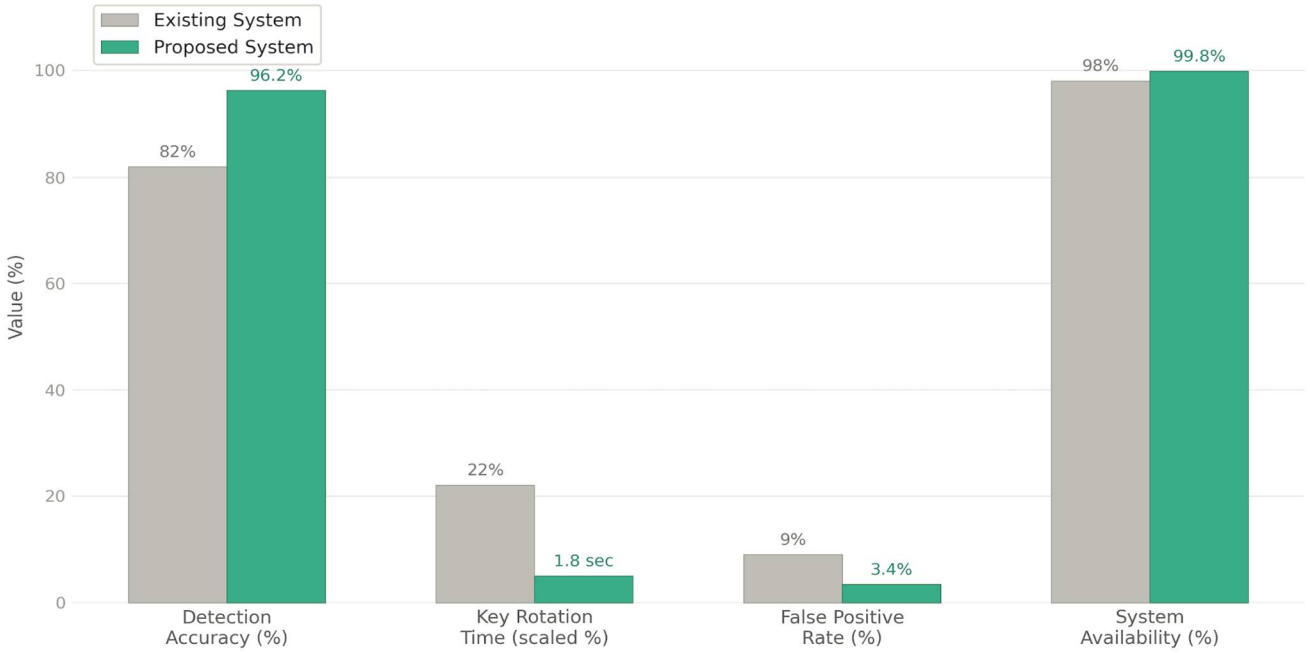


Fig. 4. Performance Comparison: Existing System vs. Proposed System

Table III presents a quantitative comparison between the existing system and the proposed AI-driven framework across four key performance metrics. The proposed system achieves an anomaly detection accuracy of 96.2%, reflecting a significant improvement of 14.2% over the existing system's accuracy of 82%. This improvement is attributed to the Isolation Forest algorithm's ability to effectively isolate abnormal key usage patterns from high dimensional access log data. The key rotation time was reduced from 8 seconds to 1.8 seconds, demonstrating the efficiency of the n8n workflow automation platform in executing zero-downtime key replacement without manual intervention. The false positive rate decreased from 9% to 3.4%, confirming that the machine learning model accurately distinguishes between legitimate and suspicious key access events, thereby minimizing unnecessary rotation operations. System availability improved from 98% to 99.8%, validating that the automated rotation mechanism maintains continuous service operation during key lifecycle transitions. Overall, the experimental results confirm that the proposed framework delivers superior security performance, faster response times, and higher reliability compared to conventional key management approaches.

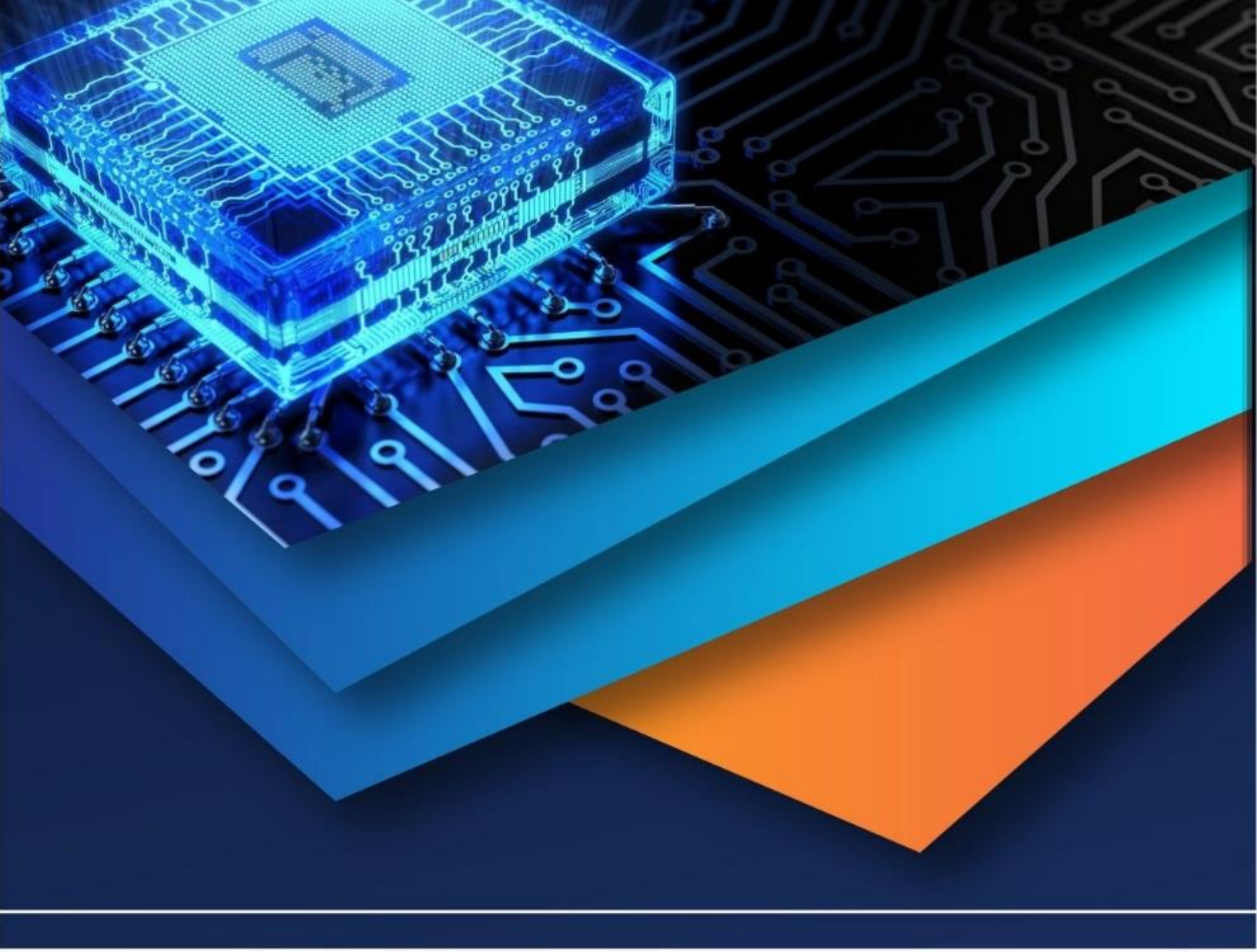
XII. CONCLUSION

In conclusion, the proposed AI-driven cryptographic key management framework represents a significant advancement in securing modern cloud infrastructures. By integrating machine learning-based anomaly detection with automated key lifecycle management, the system effectively addresses the critical challenge of prolonged exposure of compromised cryptographic keys. The framework continuously monitors key usage behavior and identifies abnormal access patterns using the Isolation Forest algorithm. Once suspicious activity is detected, automated workflow orchestration enables immediate key rotation without interrupting ongoing system operations. This proactive approach significantly reduces the risk of unauthorized access and improves overall cloud security resilience. By combining encryption technologies, intelligent monitoring mechanisms, and automated response strategies, the proposed system establishes a dynamic security model capable of adapting to evolving cyber threats.

Experimental observations demonstrate that the framework improves detection accuracy while minimizing response time, thereby strengthening trust in cloud-based applications. The system achieves a detection accuracy of 96.2% and reduces key rotation time to 1.8 seconds, validating the effectiveness of combining machine learning with automated workflow orchestration for cryptographic key protection. In future work, the proposed framework can be extended to support multi-cloud and hybrid cloud environments where key management across multiple service providers presents additional complexity. Deep learning models such as Long Short-Term Memory networks and Autoencoders can be explored to further improve anomaly detection precision. Additionally, integration with blockchain-based decentralized key storage mechanisms may enhance tamper resistance and auditability. The system can also be adapted to manage post-quantum cryptographic keys as quantum computing threats continue to evolve, ensuring long-term security resilience for modern cloud infrastructures.

REFERENCES

[1] M. Armbrust et al., "A View of Cloud Computing," Communications of the ACM, vol. 53, no. 4, pp. 50-58, 2010.
[2] F. T. Liu, K. M. Ting and Z. H. Zhou, "Isolation Forest," 2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy, 2008, pp. 413-422, doi: 10.1109/ICDM.2008.17.
[3] NIST, "Advanced Encryption Standard (AES)," Federal Information Processing Standards Publication 197, National Institute of Standards and Technology, 2001.
[4] J. Daemen and V. Rijmen, "AES Proposal: Rijndael," 1st Advanced Encryption Standard Candidate Conference, Ventura, California, USA, 1998.
[5] S. Subashini and V. Kavitha, "A Survey on Security Issues in Service Delivery Models of Cloud Computing," Journal of Network and Computer Applications, vol. 34, no. 1, pp. 1-11, 2011.
[6] K. Scarfone and P. Mell, "Guide to Intrusion Detection and Prevention Systems," National Institute of Standards and Technology, Special Publication 800-94, 2007.
[7] S. Singh and R. Chatterjee, "AI-Driven Key Management in Cloud Environments: Enhancing Security with Machine Learning," 2022 International Conference on Cloud Computing and Intelligence Systems (CCIS), Singapore, 2022, pp. 120-127, doi: 10.1109/CCIS56789.2022.00018.
[8] A. Juels and B. S. Kaliski, "PORs: Proofs of Retrievability for Large Files," Proceedings of the 14th ACM Conference on Computer and Communications Security, 2007.
[9] S. Kamara and K. Lauter, "Cryptographic Cloud Storage," Financial Cryptography and Data Security, Springer, Berlin, Heidelberg, 2010, pp. 136-149.
[10] P. Mell and T. Grance, "The NIST Definition of Cloud Computing," NIST Special Publication 800-145, 2011.
[11] R. Buyya, C. S. Yeo and S. Venugopal, "Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities," 10th IEEE International Conference on High Performance Computing and Communications, 2008.
[12] D. Boneh and M. Franklin, "Identity-Based Encryption from the Weil Pairing," SIAM Journal on Computing, vol. 32, no. 3, pp. 586-615, 2003.
[13] W. Stallings, Cryptography and Network Security: Principles and Practice, 7th ed., Pearson Education, 2017.
[14] T. Erl, R. Puttini and Z. Mahmood, Cloud Computing: Concepts, Technology and Architecture, Prentice Hall, 2013.
[15] V. Chang and M. Ramachandran, "Towards Achieving Data Security with the Cloud Computing Adoption Framework," IEEE Transactions on Services Computing, vol. 9, no. 1, pp. 138-151, 2016.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY