International Journal of Science and Research Archive

eISSN: 2582-8185

Cross Ref DOI: 10.30574/ijsra

Journal homepage: https://ijsra.net/

(RESEARCH ARTICLE)

Check for updates

Framework for Cloud Data Security Using Agentic AI

Naseer R, Srujan K M *, Deepthi A S, Divyashree C H and Goutham M

Department of Computer Science & Engineering, Bapuji Institute of Engineering and Technology, Davanagere, Karnataka, India.

International Journal of Science and Research Archive, 2025, 15(01), 1730-1735

Publication history: Received on 18 February 2025; revised on 27 April 2025; accepted on 30 April 2025

Article DOI: https://doi.org/10.30574/ijsra.2025.15.1.1255

Abstract

Cloud platforms are ever more vulnerable to advanced cyber threats, which demand intelligent and self-reliant security systems. We introduce CloudShield, a prototype Agentic AI system for mimicking live cloud data protection in Azure platforms. The system mimics round-the-clock log collection in Azure-type protocols, employs the Isolation Forest algorithm to identify outliers, and responds automatically to attacks such as brute-force attacks and malware. Logs are locally stored in a SQLite database, encrypted for secure storage, and can be deployed entirely self-contained without dependencies. CloudShield includes an interactive real-time dashboard for threat visualization and system analysis. Its agent-based, modular architecture supports scalability, automation, and high detection rates—making it a strong experimental model for current cloud security research.

Keywords: Cloud Security; Agentic AI; Anomaly Detection; Isolation Forest; Azure Logs; Automated Response

1. Introduction

Cloud computing transformed storage of data and provision of services but in doing so introduced myriad security threats with its dynamic, distributed, and highly scalable environment. Legacy security controls do not usually identify advanced or new threats in real-time, making cloud infrastructures vulnerable to brute-force attacks, malware, and privilege escalation. To solve these challenges, this paper introduces CloudShield—a light, modular, and smart cloud security system driven by Agentic AI. The platform simulates real-time Azure cloud log monitoring, uses the Isolation Forest algorithm for anomaly detection, and auto-responses to high-risk behavior. With local deployment independent of external cloud services, CloudShield provides self-contained operation and increased data privacy. Logs are stored encrypted locally within a SQLite database, and there is an interactive dashboard providing real-time threat insights and analytics. The fundamental tenets of Agentic AI applied herein are echoed across various applications, such as precision agriculture [7][8] and biometric security [9], thereby testifying to the versatility of AI-based systems. By combining machine learning with autonomous decision-making agents, CloudShield provides a proactive, scalable solution optimized for academic and experimental cloud environments.

2. Related work

Recent cloud security trends have continued to focus on the use of artificial intelligence and automation in reacting to evolving threat environments. Zhou et al. [1] developed an LLM-based proactive cloud protection system focusing on real-time response and prediction of threats. This is a conceptual foundation for CloudShield's utilization of Agentic AI in enabling autonomous decision-making. Aref et al. [2] explored human-AI collaboration via cognitive hierarchy-based reinforcement learning for enhanced cloud security operations. Their system responds dynamically to multi-level reasoning of agents, complementing our project deployment of modular AI agents with decision awareness for adaptive threat management. Yan et al. [3] developed the Intelligent Security Service Framework (ISSF) for cloud-native that

* Corresponding author: Srujan K M

dictates a multi-layered security framework identifying and combating automatically. This influences CloudShield's architecture directly, specifically response module coordination and anomaly detection module coordination. Gupta et al. [4] proposed MAIDS, a method to identify malicious agents using behavioral knowledge in cloud infrastructures. Their entity profiling and detection of malicious intent correlate with CloudShield's IP-based scoring and behavioral threat assessment. Both Merritt [5] and Huang [6] discussed the growing application of Agentic AI in identity management and the coordination of edge-cloud security. Their contributions regarding autonomous agents of trust, access, and networking informed CloudShield's agent-based architecture for remediation of safely processed logs. These studies constitute the cornerstone of our framework. CloudShield raises these ideas to the next level with an entirely local, standalone, and expandable AI-powered defense mechanism supported by Azure cloud infrastructures, real-time anomaly score, and remediation automation — third-party cloud dependency-free.

3. Methodology

The proposed structure, CloudShield, is a modular, agentic system to be used in simulated real-time anomaly detection and incident response on Azure-based clouds. It operates with Agentic AI, simulating logs, extracting features, scoring anomalies, automating response, and reporting. As illustrated in Fig. 1, CloudShield is composed of five primary components: Log Collector, Feature Extractor, Anomaly Detection Module (Isolation Forest), Automated Response Manager, and a Visualization Dashboard. Simulated Azure activity logs corresponding to real-world usage patterns are generated and processed usingAPI-based modules.

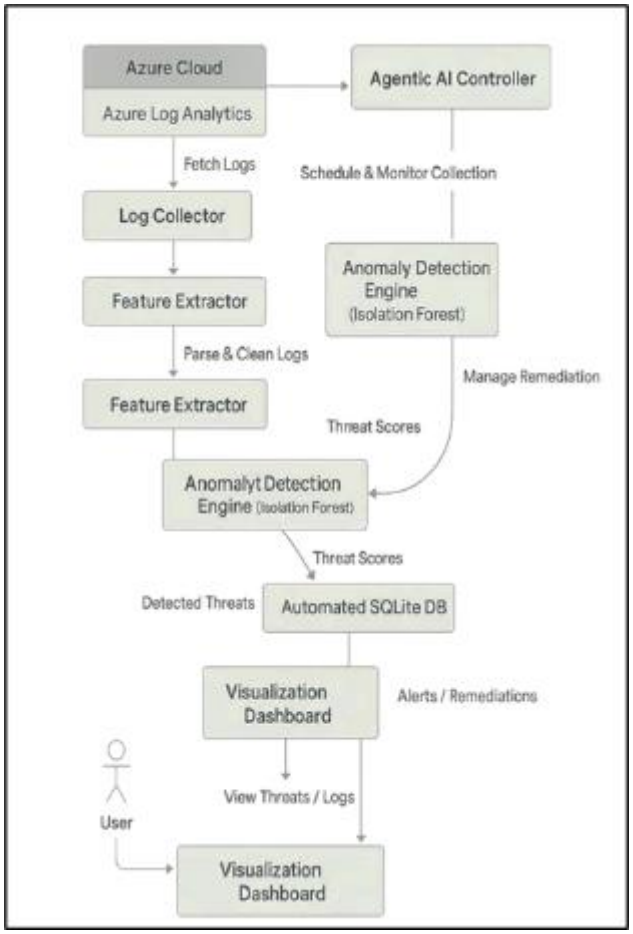

Figure 1 CloudShield Methodology overview

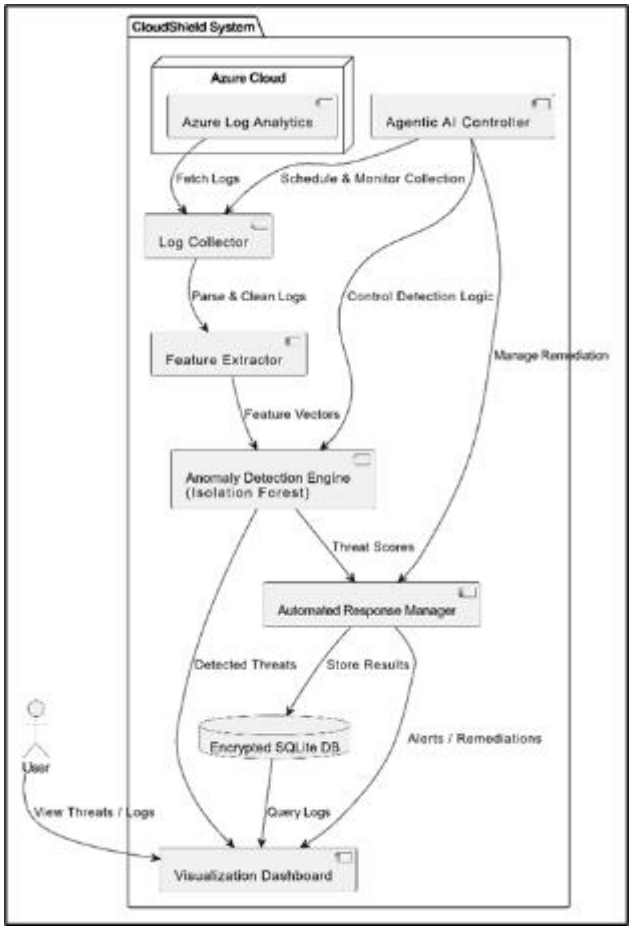

Figure 2 Data flow and component interaction diagram

The logs are cleaned and checked for behavioral deviations such as unusual access patterns and login peaks. Anomaly Detection has an event-scoring function performed using Isolation Forest, grading events by severity. For the worst anomalies at greatest risk, pre-configured action such as IP quarantine or access revocation is initiated by the Agentic AI Controller. All responses and events are logged in an encrypted SQLite database, as displayed in Fig. 2.

The dashboard offers real-time monitoring and report export capability for analysis. The agent-based, modular architecture renders CloudShield highly scalable, autonomous, and best suited for academic simulation of smart cloud threat management.

3.1. Proposed algorithm

3.1.1. Design Considerations

The system operates locally and is built entirely using Python-based microservices. Simulated Azure cloud logs are generated to mimic real-world events from sources such as Active Directory, Key Vault, and Virtual Machines. These logs are processed in real time by a FastAPI backend. Each log is encrypted and locally stored in a SQLite database. Pre-trained and optimized Isolation Forest model is utilized to identify behavioral anomalies in high-dimensional log data. Autonomous log analysis, anomaly detection, and threat response are managed by an Agentic AI controller. Visualization of results is achieved through a React+Recharts based dashboard.

3.2. Description of the Proposed Algorithm

The system consists of the following main steps:

3.2.1. Step 1: Logging Simulation and Feature Extraction:

Predefined templates are used to simulate synthetic Azure logs based on actual event patterns. Each log entry is broken down into a formatted list of attributes such as IP address, username, resource ID, event type, timestamp, and access result. The attributes are vectorized and normalized before being fed to machine learning inference.

3.2.2. Step 2: Anomaly Detection using Isolation Forest:

Anomalies are detected by the average number of partitions required to isolate cases, which is achieved by enriching the Isolation Forest model with logs. The anomaly score is given by:

$$\text{Anomaly Score} = 2^{-\frac{E(h(x))}{c(n)}}$$

where E(h(x)) is the mean path length for instance x and c(n) is the normalization factor. Events with a score greater than a dynamic threshold (e.g., 0.7) are marked as possible threats.

3.2.3. Step 3: Threat Classification and Scoring:

Anomalies are labeled into threat types such as Malware, Brute Force, Unauthorized Access, or Data Exfiltration based on source context and behavior patterns. Each is labeled with a severity level: Low, Medium, High, or Critical.

3.2.4. Step 4: Autonomous Response Triggering:

Agentic AI triggers response actions as suitable based on severity:

- Critical/High threats: Quarantine IPs, revoke access.
- Medium threats: Alert admins or isolate resources.
- Low threats: Log the event for monitoring.
- These replies are saved securely in the database.

3.2.5. Step 5: Report Generation and Visualization:

All system replies and threat incidents are logged timestamps. The dashboard shows real-time status and facilitates exporting detailed reports like:

- Total number of threats by severity and type.
- Actions taken and Timeline of Key Events.

- Pseudo code
  - Step 1: Begin log monitoring service and log collector agent.
  - Step 2: Retrieve Azure cloud logs using authenticated API credentials.
  - Step 3: Preprocess all log entries and extract features (e.g., IP address, event type, access result, timestamp).
  - Step 4: Convert extracted features into numerical form acceptable as ML input.
  - Step 5: Feed feature vectors to the trained Isolation Forest model.
  - Step 6: Calculate the anomaly score for every log event.
  - Step 7: Assign Anomaly Level and Severity Level to each and every one of the identified anomalies.
  - Step 8: Call Agentic AI Controller to select autonomous response option (e.g., IP blocking, admin alert).
  - Step 9: Document all procedures and risks in the encrypted SQLite database.
  - Step 10: Refresh the real-time dashboard with the fresh data.
  - Step 11: Generate a system report and optionally save threat logs.
  - Step 12: Continuously keep adding logs and watches (Step 2).
  - Step 13: Summarize and alert the system administrator.

4. Results

The CloudShield platform was designed and implemented utilizing emulated Azure cloud activity logs to validate the system's performance in real-time cloud-based anomaly detection and response. Python implementation of the backend facilitated secure gathering of logs, preprocessing, anomaly detection using Isolation Forest, and automatic initiation of response. Logs and threat-driven information were stored securely in an encrypted SQLite database for lean yet effective data management. For the frontend, there was a creation of a responsive and interactive dashboard using TypeScript and React, ensuring seamless user interaction, real-time monitoring, visualization of anomalies, and report creation. To display the functionality of the platform, each module was tested via controlled simulations and its results were presented using system snapshots.

Fig. 3. signals an intuitive interface with immediate access to received log data, anomaly detection overview, and current response status. Anomalies can be accessed and monitored by users through several modules for viewing system operations and analytics. Fig. 4. displays the timeline of the processed events colored by Severity Level (High, Medium, Low). This visualization enables the security analysts to clearly see and prioritize anomalies that have to be investigated.

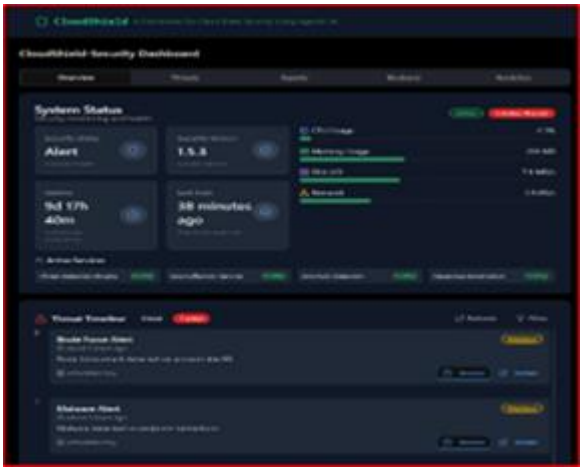

Figure 3 Overview Section of CloudShield Dashboard

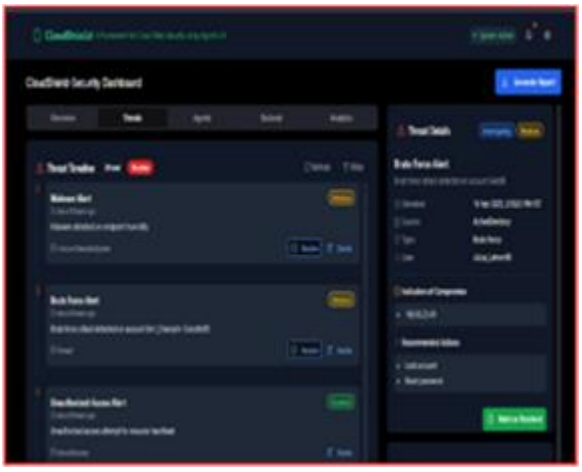

Figure 4 Threat Timeline Section of CloudShield Dashboard

Fig. 5. displays Isolation Forest model-computed anomaly scores. Plots each log event according to its computed Anomaly Level, so users can see normal vs. suspect activity visually. Offers unambiguous presentation of pattern aberrations and facilitates decision-making. Fig. 6. collates threat logs, anomaly scores, system response, and risk metadata into an exportable report. Ideal for auditing, compliance review, and reviewing history for incidents.

Figure 5 Anomaly Score Distribution of Threats

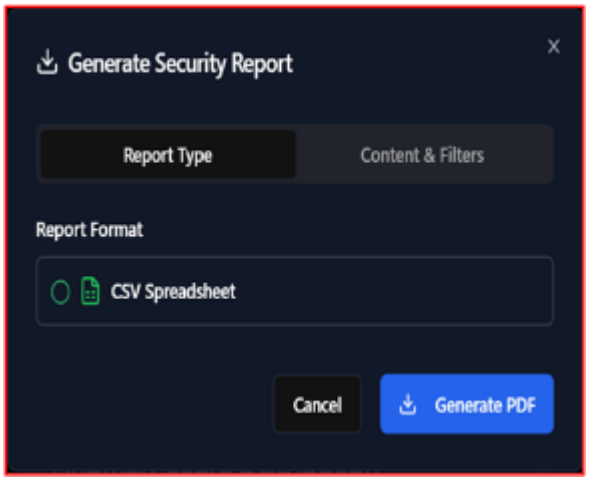

Figure 6 Generation of Report in CSV Format of Threats

4.1. Comparison

To compare the performance and intelligence of the system we proposed, we compared some of the most common machine learning algorithms for anomaly detection, such as One-Class SVM, K-Means Clustering, DBSCAN, and Isolation Forest. As can be seen in Fig. 7, the Isolation Forest algorithm performed significantly better than the others, with an F1-score of 0.94, and high accuracy and a low false-positive rate. These measurements indicate its aptness for handling large-scale, high-dimensional cloud log data and therefore make it the optimal choice for CloudShield's anomaly detection engine.

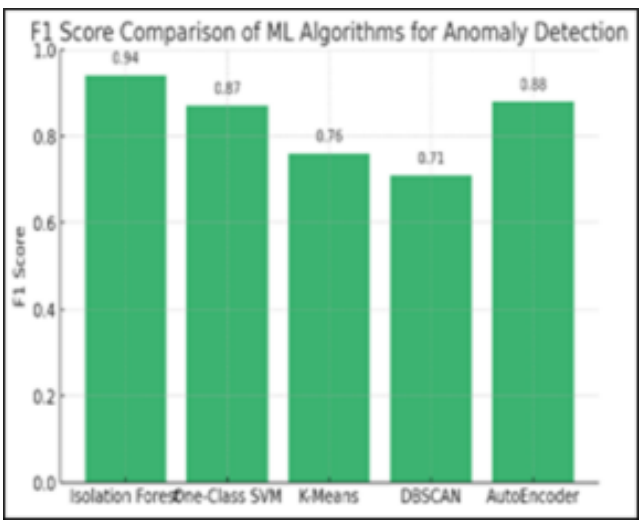

Figure 7 F1 Score Comparison of ML Algorithms

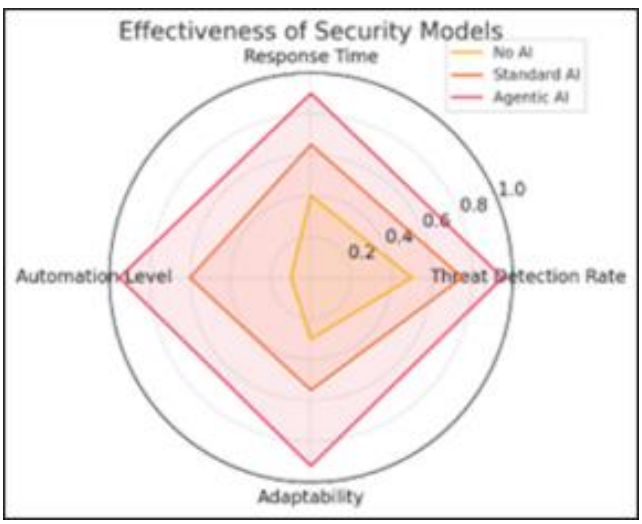

Figure 8 Effectiveness of Security Models

Aside from algorithm-level comparison, we assessed the system's intelligence and operational autonomy by comparing three security models, No AI, Traditional AI, Agentic AI. The models were compared based on four main parameters: Threat Detection Rate, Response Time, Automation Level, Adaptability. As illustrated in Fig. 8, Agentic AI model outperformed the other models in all four metrics. It demonstrated abilities to detect, analyze, and respond automatically to threats in real-time without intervening, which helps reduce operational overhead.

This comparison supports the architectural and algorithmic decisions taken in CloudShield's development, as it supports its robustness as a scalable, intelligent, and autonomous security infrastructure for Azure cloud environments.

5. Conclusion and future work

CloudShield effectively addresses the increasing demand for intelligent, autonomous cloud security through simulated Azure log monitoring, Isolation Forest-based anomaly detection, and Agentic AI-driven threat response. Its modular, lightweight design eliminates third-party dependencies, making it highly deployable. With encrypted local storage, an interactive dashboard, and autonomous decision-making, CloudShield establishes a dynamic, self-sufficient defense model—capable of proactively detecting, analyzing, and responding to evolving cloud-based threats in real-world environments.

As CloudShield is still a simulation-based proof-of-concept, future work will be toward integrating live Azure services such as Azure Monitor and Defender to enable live log ingestion. Extensions include graph-based threat correlation, reinforcement learning-based adaptive response, and multi-agent coordination for scalability. CloudShield also sees integration with enterprise SIEM tools and moving to a cloud-native database, enabling robust, real-world deployment for smart, autonomous cloud data security.

Compliance with ethical standards

Disclosure of conflict of interest

No conflict of interest to be disclosed.

References

[1] Y. Zhou, G. Cheng, K. Du, and Z. Chen, "Toward Intelligent and Secure Cloud: Large Language Model Empowered Proactive Defense," arXiv:2412.21051, 2024.

[2] Z. Aref, S. Wei, and N. B. Mandayam, "Human AI Collaboration in Cloud Security: Cognitive Hierarchy - Driven Deep Reinforcement Learning," arXiv:2502.16054, 2025.

[3] Y. Yan, K. Huang, and M. Siegel, "ISSF: The Intelligent Security Service Framework for Cloud Native Operation," arXiv:2403.01507, 2024.

[4] Kishu Gupta, Deepika Saxena, Rishabh Gupta, and Ashutosh Kumar Singh, "MAIDS: Malicious Agent Identification-based Data Security Model for Cloud Environments", arXiv:2412.14490, 2024.

[5] D. Merritt, "Agentic AI: Transforming Cloud Networking and Edge Intelligence in 2025," Aviatrix Blog, 2025.

[6] K. Huang, "Agentic AI Identity Management Approach," Cloud Security Alliance, 2025.

[7] R. Naseer, N. S. Sahana, K. K. Sangeetha, S. S. Bedre, and C. S. Sneha, “AI-Based Drone for Crop Disease Detection in Precision Agriculture,” *Int. J. Eng. Res. Technol.*, vol. 13, no. 4, 2024.

[8] R. Naseer, M. K. Sneha, Udayalaxmi, U. N. Gowda, and M. D. Vismaya, “AI-Based Drone for Pesticide Recommendation and Spraying for Precision Agriculture,” *Int. J. Eng. Res. Technol.*, vol. 13, no. 4, 2024.

[9] N. Rajasab and M. Rafi, “A deep learning approach for biometric security in video surveillance system using gait,” *Int. J. Safety Secur. Eng.*, vol. 12, no. 4, pp. 491–499, 2022.