Natural Resources for Human Health
eISSN: 2583-1194

Original Research

Received 24 May 2026
Revised 26 June 2026
Accepted 10 July 2026

Natural Resources for Human Health 2026, Vol. 6 (12s): 240–259

KEYWORDS:
Explainable Agent AI · Cloud–Edge Computing · Adaptive Monitoring

Secure Cloud–Edge Agent AI Compromise Detection and Autonomous Recovery Using Spy Agent Verification

M.Gayathri[1] Dr. G. Srinaganya[2]

[1]PG and Research Department of Computer science, National College (Autonomous), Affiliated to Bharathidasan University, Trichy-62000, Tamilnadu, India
[2]PG and Research Department of Computer science, Assistant Professor, PG and Research Department of Computer Science, National College (Autonomous), Affiliated to Bharathidasan University ,Trichy-620001, Tamilnadu, India
Email id: gayathriphdnct@gmail.com, srinaganyanaac2014@mail.com

ABSTRACT: The fast growth in intelligent services in Cloud-Edge infrastructure has led to the growing reliance on Autonomous Agent AI for decisions, management, and execution of services. Yet, the constant running of the Agent AI leads to security challenges because any compromised agents might show unusual behavior patterns in terms of communications, execution, resource usage, and decision-making while performing the services. Traditional AI-based security approaches are mostly concerned with attack detection and adaptation of responses but not much in terms of safeguarding Agent AI from attack and restoring the legitimate operational knowledge of the Agent AI. The proposed research presents the development of the Secure and Self-Protective Explainable Agent AI framework that can be used to ensure security and resilience of Cloud–Edge systems. In particular, the framework uses secure entity registration and verification, behavioural learning, adaptive monitoring and Spy Agent, who monitors all processes continuously and detects compromises. As soon as the compromise is detected, the compromised Agent AI is isolated, and its legitimate knowledge base is saved. Then a knowledge migration process using transfer learning takes place, where a new Agent AI instance is created and hardened, and suspicious activity is tracked. Two algorithms are developed for the secure Cloud–Edge entity verification with adaptive monitoring and Spy Agent-based compromise detection with automatic Agent AI replacement. The Self-Protective Explainable Agent AI framework provides continuous security assessment, saving of legitimate knowledge during recovery, and maintaining operations of services with a hardened agent.

1 INTRODUCTION

Graph RAG is the technique that is suggested to be utilized in order to provide explainability in dataset discovery. In this case, the idea of graph-based retrieval-augmented generation is used in order to improve dataset discovery. In particular, the method is focused on improving the process of explainability in dataset discovery [1]. The idea of advanced persistent threat (APT) is related to a series of sophisticated activities that are likely to take place during various stages and by various actors. The modeling of the actions of an adversary based on different types of actors will help in the development of adversary emulation and cyber threat intelligence. It stresses the importance of focusing on relations between different attacks as opposed to independent attacks [2].

The ICS (Industrial Control Systems) faces vulnerabilities since there is the ability to find out and attack networked industrial devices. Using advanced traffic analysis makes it possible to detect any kind of communication that has some sort of exploit attempt pattern. In other words, traffic analysis is a good method for detecting harmful behavior. However, traffic analysis by

itself may not be enough for detecting malicious entities working under trust relationship [3].

A honeypot is a proactive form of security where attackers are lured and their actions are analyzed. Interactive and real responses can be created using the honeypot model based on LLMs. The difficulty here would be in consistently keeping up the responses while resisting the efforts made by attackers to figure out that the system being attacked is indeed a honeypot [4].

Multi-agent and distributed learning can contribute to better cyberattacks detection through engaging several elements in making security decisions. Multi-agent deep-learning technology is able to distribute the activities related to attack detection among interconnected security components. These types of architectures prove the potential of integration between distributed intelligence and machine learning in the area of cybersecurity. Nevertheless, detection of attacks in distributed systems does not imply finding the group of malicious agents collaborating in manipulating other participants [5].

Machine learning-based IDS can perform efficient attacks classifications, yet the decision-making process behind these models is often hard to understand. XAI methods like LIME and SHAP can help to find the feature values influencing the model's decision and thus provide helpful explanations for security specialists. However, it is not sufficient to solve the problem of malicious collaboration between several entities [6].

IoT networks are subject to continuously evolving and heterogeneous attack scenarios that necessitate adaptive attack detection mechanisms for ensuring their security. Combination of ensemble learning with Q-learning algorithms allows for developing adaptive attack detection, and the explanation of the produced predictions will be facilitated using XAI techniques. This technique will improve the adaptability of the intrusion detection system to varying attack scenarios, although the main goal is still attack classification and not coordinated manipulation of trust relations [7]. Distributed learning refers to a situation where heterogeneous IoT ecosystems result in distributed data that makes it hard to carry out centralized intrusion detection in an efficient manner. The edge-federated learning algorithm is one example of distributed learning that enables the training of models in different IoT environments without necessarily centralizing the local data. Distributed learning does not measure the trust and behavioral relationship of the entities involved [8].

Real-time dynamic response identification is considered in the case of highway structural health monitoring data. The technique is designed to analyze dynamic responses using the structural health monitoring data in real time [9].

In host-based intrusion detection, there is another security layer where the malicious actions against particular IoT devices and hosts can be monitored. There are machine learning and ensemble-based techniques to classify various kinds of intrusions and to attain good performance levels through accuracy, precision, recall, and F1 score measurements. This is how the efficiency of intelligent classification is shown in IoT security. Nonetheless, the classification of attacks cannot help to differentiate between the independent and collective ones [10].

Learning using transformer architecture has gained more relevance in recent years due to the capability of understanding relations among inputs. The inclusion of explainable artificial intelligence allows the identification of other factors contributing to the results of the prediction process. This technique helps make prediction automation techniques more transparent and understandable. The following is an illustration of how combining learning architecture with explainability can help in decision making [11].

Hybrid NLP methods can facilitate the analysis and forecasting of complex textual data using the combination of multiple computational methodologies. The use of explainable AI technologies allows achieving increased transparency in respect to the decisions made by predictive algorithms. This is how the concept of explainable learning can be applied to increase the interpretability of automated analysis and forecasting processes [12].

Multi-agent cooperative systems utilize information exchange between the participating agents in order to make distributed decisions. Cooperative localization shows how information from several autonomous agents can be used to enhance the performance of a system in a distributed way without full reliance on the centralized controller. Information exchange can be applied to distributed cybersecurity scenarios as well. But unreliable or malicious participants may affect collective decisions, which shows the necessity to assess the reliability of information provided by the individual agents [13].

Counterfactual temporal motifs generation is examined in the case of temporal graph neural networks. The technique is aimed at generation of counterfactual temporal motifs in order to study the behavior of temporal graph neural networks. This

technique can be used to reveal the characteristics of temporal graph neural networks [14]. Cyber threat intelligence is able to enhance security in Industrial IoT through identifying and classifying various forms of malicious behaviors. The use of machine learning and deep learning technologies can be utilized in conducting binary and multi-class intrusion detections, whereby security systems will be able to identify various patterns of attacks. This highlights the capability of intelligent systems in improving threat identification. Nonetheless, the classification of attacks does not imply that there are malicious agents that collaborate to affect other entities [15].

1.1 Literature Review

Noman et al. provided a thorough study on the log poisoning attacks that can be performed against IoT environments by analyzing the attack methods, evasion methods, detection methods, and mitigation approaches. This study proved how the manipulated log data can help to hide the malicious activity . The critical nature of the log poisoning attacks and the difficulties related to their detection were also addressed in this study. However, this study primarily concentrates on the analysis and mitigation of the attacks [16]. Yin et al. offered a review of the state of art of on-device recommenders through examining their main technologies, features and evolution. The paper reviewed the main issues related to the on-device recommender systems and addressed the issues arising during the process of on-device recommendation. This comprehensive review covers all aspects of the evolution and usage of on-device recommenders [17]. Bondok et al. studied Trojan attacks on federated learning-based electricity-theft detection in smart grids, where malicious participants alter their training data locally to avoid detection. In their study, three approaches named Redundancy, Med-Selection, and Combined-Selection were suggested in order to minimize the effect of the malicious participants in model combination. The experimental results have proven that the success rate of the Trojan attack could be as high as 90% and that the proposed defense mechanisms decrease the success rate significantly. However, the paper is focused on the problem of federated learning poisoning only [18]. Hassan et al. proposed a trust aware routing framework for MANETs using a combination of federated learning and multi objective optimization. This scheme takes into account both trust related factors and routing requirements to facilitate secure communication in highly dynamic and insecure environment of MANETs. Federated learning allows collaboration without requiring collection of sensitive network information in central locations. Nevertheless, this scheme is primarily focused on trust aware routing in MANETs and does not specifically deal with collusion of nodes to manipulate trust and reputation scores [19]. Rodis et al. made an extensive review on multimodal explainable artificial intelligence, which is concerned with methods that enhance the transparency and interpretability of sophisticated AI systems. The study analyzed approaches that facilitate multimodal data integration, and the capacity of such approaches to offer explanations for AI decision-making. In addition, challenges connected with multimodal data integration, interpretation, and evaluation have been considered. Nevertheless, the research focuses on XAI approaches in general and fails to tackle the problem of manipulation of trust and collusion detection [20]. Masud et al. studied the use of explainable artificial intelligence in resilient security applications within the IoT. In this context, it was pointed out how the use of XAI would help in improving the transparency, interpretability, and robustness of AI-based security schemes, as well as helping to detect security threats in IoT systems. The challenges of using XAI in terms of explainability, robustness, privacy, and resource-constrained environments have been also presented. Nevertheless, this research offers a wider security approach rather than developing a trust-based mechanism for collusion detection [21]. Farooq et al. designed an explainable artificial intelligence technique for building an interpretable anaemia prediction model by improving the transparency of machine learning-based medical predictions. The suggested methodology integrates the concepts of prediction models and explanation methods that help users comprehend how the prediction model makes decisions. The research proved the potential of explainability in increasing transparency and confidence in AI-based prediction systems. However, the use of this technique is only applicable in healthcare prediction and does not aid in malicious-agent detection research [22].

Kearney et al. explored transfer learning in relation to simulations as well as recorded data for predicting lateral pinch thumb tip forces. In the experiment, a model based on LSTM was trained on simulated data and fine-tuned with recorded data. This transfer learning model showed better predictive results than a model that was trained only on recorded data, thus proving the benefit of knowledge gained from simulation in cases where real-world data is scarce. Even though this paper shows the potential of transfer learning, it is only indirectly related to cybersecurity and collusion detection due to its biomedical nature [23]. Tourab et al. discussed the application of machine learning and explainable artificial intelligence within studies on the gut microbiome. This review has concentrated on the ways in which learning-based approaches could be used for obtaining insights from intricate biological data sets in an interpretable way. The increasing significance of explainability when it comes to the

analysis of relationships between microbiome properties and patient outcomes has been considered, along with existing challenges and possible lines of research combining ML and XAI. Nevertheless, the review is field-specific and does not discuss cybersecurity and trust management issues [24]. Ragab et al. developed ESPRESSO which is intended to enable search in the decentralized Web. The paper highlights the problems of enabling search in the decentralized Web environment and presents a structured way of doing so. The paper mainly aims at improving search abilities in the decentralized Web [25].

Mahir et al. presented a novel hydro-informatic modeling approach that integrates feed-forward neural networks, federated learning, and explainable artificial intelligence to predict floods. This approach provides a means of collaborative learning while maintaining data privacy. The use of explainability in this framework helps enhance the understanding of the outcome of the predictions made. This research shows the potential use of federated learning and XAI in achieving reliable predictions. The only downside of this research is that the focus of the application is flood prediction, and trust is not explored [26].

Siam et al. have introduced the concept of IP SafeGuard, an intelligent approach that makes use of AI to find malicious IP addresses in a networked environment. In this case, an intelligent technique is used to analyze network-based attributes in order to differentiate malicious from non-malicious IP addresses. Intelligent detection schemes are useful in enhancing the capability of security systems to recognize suspicious entities in networks and also decrease reliance on traditional rule-based schemes. IP-level classification, however, fails to capture the coordination between agents involved in collusion attacks [27].

Hmimou et al. introduced a multi-agent security architecture in which large language models are used for threat detection and correlation from disparate sources of security information. The purpose of this model is to solve the problems faced by individual security rule and signature-based systems in terms of offering context and correlation of attack events across multiple attack vectors. However, while the proposed model addresses security event correlations, it does not specifically address the issue of collusion detection through trust in social and multi-agent networks [28].

Melhem et al. came up with LENS, which is a lightweight and explainable framework for identifying APTs through LLMs at the edge for 6G networks. This framework makes use of a fine-tuned DistilBERT model in representing the network streams and contextual metadata into representations that are good for APT detection by embedding explainability in the detection framework. The accuracy and recall of the framework when evaluated on the CICAPT-IIoT dataset were found to be around 0.82, making it applicable in resource-constrained Raspberry Pi edge devices. However, this framework only aims at APT detection, without exploring collusion among the malicious parties [29]. Chaganti has introduced an AI-powered security system framework for the Internet of Things ecosystem through the incorporation of artificial intelligence-based intrusion detection, blockchain-based decentralized trust management, edge computing, optimization, and game theory concepts.

This framework is designed to address the challenges of scalability, real-time threat detection, data security management, and resource management in the IoT ecosystems. The fusion of AI, blockchain, and edge computing leads to a more extensive security framework that is able to accommodate distributed IoT ecosystems. But this framework does not specifically address the problem of detecting collusion between multiple agents [30].

2. RESEARCH GAP

The AI-based IoT and Edge security approaches are primarily concerned with threat detection, optimization, and mitigation. However, they are not sufficiently concerned about the compromise of the Agent AI itself and its recovery from such compromise. The main deficiency in this regard is that there is no single solution that provides for independent Agent AI verification, adaptive monitoring, compromise isolation, knowledge retention, knowledge migration through Transfer Learning approach, secure replacement of the compromised Agent AI, and monitoring of the replacement. The Secure Cloud-Edge Entity Verification and Spy Agent-Based Compromise Detection approach is a solution to the problem.

2.1 Overview of the Secure and Self-Protective Explainable Agent AI Framework

Secure and Self-protective Explainable Agent AI Framework ensures that the Explainable Agent AI working in Cloud-Edge environment is protected through security and recovery process. The architecture includes Cloud-Edge Entity Verification, Adaptive Monitoring, Independent Spy Agent, Agent AI Compromise Detection, Agent AI Isolation, Legitimate Knowledge Protection, Transfer-Learning-based Knowledge Migration, Replacement Explainable Agent AI, Hardening, and Suspicious Activity Logging. All these processes work as security lifecycle, starting from securing entity verification and adaptive monitoring all the way to independent observation of Agent AI, its compromise, migration, and replacement.

Natural Resources for Human Health
eISSN: 2583-1194

3. EXPLAINABLE AGENT AI SELF-PROTECTION FRAMEWORK WITH SPY AGENT-ENABLED AUTONOMOUS RECOVERY FOR CLOUD–EDGE ENVIRONMENTS

The architecture of the Secure and Self-Protective Explainable Agent AI Framework for Cloud-Edge environments is depicted in Figure 1. The framework ensures the establishment of a security flow in the following order – Secure Cloud-Edge Entity Verification, then Adaptive Monitoring and observation independently of the Spy Agent. The processed data of the observed agent's operation are evaluated in terms of the security status of the primary Explainable Agent AI by the Agent AI Compromise Detection component. In case the compromise is not detected, monitoring by Spy Agent continues permanently. In case of compromise, the isolation of the primary Agent AI is done, the genuine data of its operation is saved, and Transfer Learning Based Knowledge Migration is performed in support of a Replacement Explainable Agent AI. After that, security hardening and suspicious activity tracking are performed for the replacement, and the safe operation is continued.

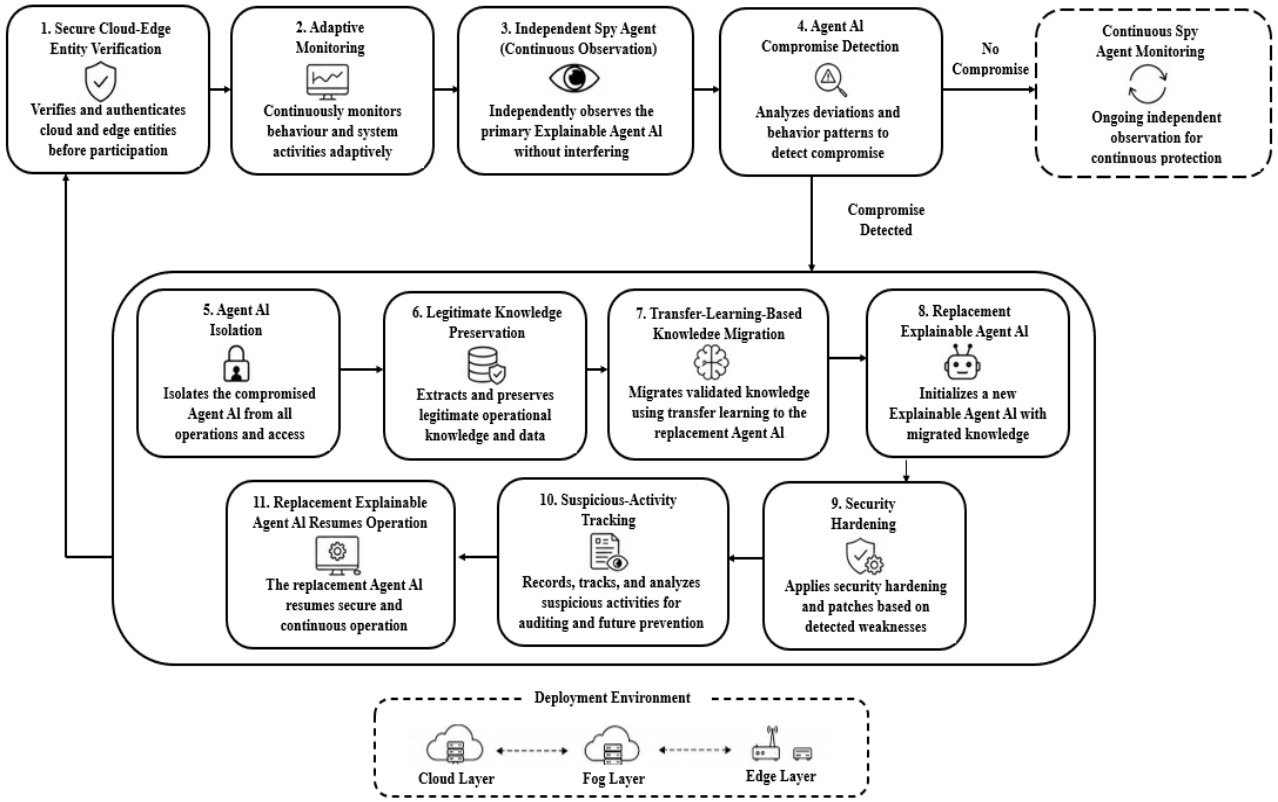

Fig. 1 Secure Autonomous Agent Replacement and Continuous Threat Monitoring Framework for Cloud–Edge Environments

The architecture starts with Secure Cloud-Edge Entity Verification, where participating Cloud and Edge entities are verified before entering the operating environment. Information about the relevant entity is analyzed to determine the trusted participation status. After verification, Adaptive Monitoring analyzes the operational status and sets the appropriate monitoring status based on the detected security status.

Next, an independent Spy Agent observes the main Explainable Agent AI independently of the normal decision making process. The operational information is used by Agent AI Compromise Detection module, which evaluates any deviations from the normal behavior and detects the security status of the main Agent AI.

In the case where there is no Agent AI compromise, the system proceeds with Continuous Spy Agent Monitoring phase, where the main Explainable Agent AI can operate under independent observation. If the Agent AI compromise was detected, the

Agent AI Isolation phase removes the compromised Agent AI from the operational activity.

The validated operational information is processed through Legitimate Knowledge Preservation, where the knowledge needed to continue operation is kept, while all other compromised information is removed. The knowledge is transferred using Transfer-Learning-Based Knowledge Migration to initialize Replacement Explainable Agent AI.

The replacement Agent AI then goes through the process of Security Hardening, where proper security control measures are fortified based on the identified vulnerability. Suspicious Activity Tracking logs security-related data as part of the ongoing security assessment.

These activities are followed by the Resume Operations of the Replacement Explainable Agent AI, and the framework goes back to the Verification phase. The Spy Agent independently monitors during further operations, and this completes the cycle of verification, monitoring, compromise, autonomous recovery, and protection.

3.1 Role of the Secure and Self-Protective Explainable Agent AI Framework

Secure and Self-Protective Explainable Agent AI is a method of providing security and resiliency for Agent AI running in Cloud–Edge architecture. It focuses on maintaining trustworthiness and capability to autonomously deal with security compromise. The Spy Agent runs independently of the main Explainable Agent AI, providing an ability to detect suspicious changes without relying solely on the monitored agent. After a compromise is detected, the Agent AI Isolation helps to minimize the consequences of this particular agent, and Legitimate Knowledge Preservation preserves the abilities that are needed for further functioning of the system. Transfer-Learning-Based Knowledge Migration allows using the validated knowledge by a Replacement Explainable Agent AI. The Security Hardening will make sure that the replacement can resist against the detected weakness, and Suspicious-Activity Tracking stores information related to security incidents. In this way, Secure and Self-Protective Explainable Agent AI represents a shift from security monitoring and detecting compromise to containing and recovering from the detected incident without reinitialization of the whole Cloud–Edge architecture.

Algorithm 1: Secure Cloud–Edge Entity Verification and Adaptive Monitoring

Input: Cloud and Edge registration request, configuration parameters, historical records, ratings, certificates, resource information, communication data, operational parameters

Output: Security status, reference profile, monitoring level

BEGIN

1. Initialize the Explainable Agent AI and security module.

2. Receive the Cloud and Edge registration request and collect configuration, resource, connectivity, reliability, service, and operational information.

3. Retrieve available ratings, feedback, certificates, and historical records, and perform background verification.

4. Determine whether the entity is NEW or KNOWN.

5. IF entity = NEW THEN

    Observe initial normal behaviour and construct Reference_Profile.

  ELSE

    Retrieve the previously established Reference_Profile.

  END IF

6. WHILE the entity remains active DO

    Collect current operational behaviour.

    Deviation ← Compare(Current_Behaviour, Reference_Profile).

    IF Deviation ≤ Accepted_Threshold THEN

```
        Monitoring_Level ← LIGHTWEIGHT
        Security_Status ← NORMAL
    ELSE
        Monitoring_Level ← HEAVY
        Analyse communication, data transfer, resources,
        connectivity, configuration, service, execution,
        prompts, comments, requests, and suspicious activities.
        Compare observations with the suspicious-data
        database and correlate them with historical records.
        Security_Status ←
            {NORMAL, SUSPICIOUS, POTENTIALLY_COMPROMISED}
        IF Security_Status = POTENTIALLY_COMPROMISED THEN
            Forward entity to compromise-handling procedure.
        END IF
    END IF
    Update security record and behavioural assessment.
    Continue the next monitoring cycle.
  END WHILE
RETURN Security_Status, Reference_Profile, Monitoring_Level
END
```

3.2 Adaptive Monitoring Level Selection

Upon authentication of the Cloud and Edge entity and gathering of the reference information, there is an evaluation of the present operational state compared with the corresponding reference state. This step aims at dynamically assessing the right level of monitoring depending on the deviation detected. Entities that are still within the acceptable range of behaviour are classified as Lightweight Monitoring while those deviating largely will be subjected to Heavy Monitoring in order to examine the communication, data transfers, resource utilization, connectivity, configuration, services, execution, and suspicious behavior.

$$MLt = \{LM, HM, Dt \leq Dth, Dt > Dth \quad (1)$$

M(t) in Eq. (1) refers to the level of monitoring that is allocated to the Cloud and Edge node at time t. D(t) indicates the observed difference in the behaviour of the system from the expected normal profile, whereas D_th indicates the defined acceptable difference threshold. LM stands for Lightweight Monitoring, and HM stands for Heavy Monitoring. If D(t) ≤ D_th, then the observed difference is within the acceptable limit, and hence, there will be lightweight monitoring for normal operation. On the other hand, if D(t) > D_th, then heavy monitoring takes place to conduct the security assessment of the system.

3.3 Heavy Monitoring-Based Suspicious Activity Analysis

Whenever the Heavy Monitoring feature is set to operate, the entity is closely scrutinized through various operational attributes. This analysis takes into account such aspects as communication, data transfer, resources used, connectivity, configuration, services, execution, suspicious activities, and abnormal requests. All these are compared with the suspicious data base in order to assess the security

status.

$$SAt = n1i = 1\sum nwisi, t \qquad (2)$$

In Eq. (2), SAL(t) stands for the total Suspicious Activity Level at time t. $S_i(t)$ is the suspiciousness value, which has been normalized on the basis of the i-th monitored operational characteristic, whereas w_i is the corresponding importance weight of the operational characteristic. N is the total number of operational characteristics under examination during Heavy Monitoring. These operational characteristics are communication, data transmission, resource utilization, connectivity, configuration, services, behavior, and suspicious activities. The Suspicious Activity Level is derived by multiplying the normalized suspiciousness value of an operational characteristic with its corresponding importance weight and adding all such values.

It is combined with the historical analysis and data suspiciousness assessment for determining whether the object is still Normal, Suspicious or Potentially Compromised.

Figure 2 illustrates the operational framework of Secure Cloud–Edge Entity Verification and Adaptive Monitoring, focusing on the connection between entity evaluation, reference data, operational state evaluation, and security monitoring. The process involves adaptive monitoring levels and comprehensive security evaluation in order to ensure ongoing security of Cloud–Edge entities. This framework serves as a foundation for maintaining security logs and forwarding suspicious entities for subsequent action.

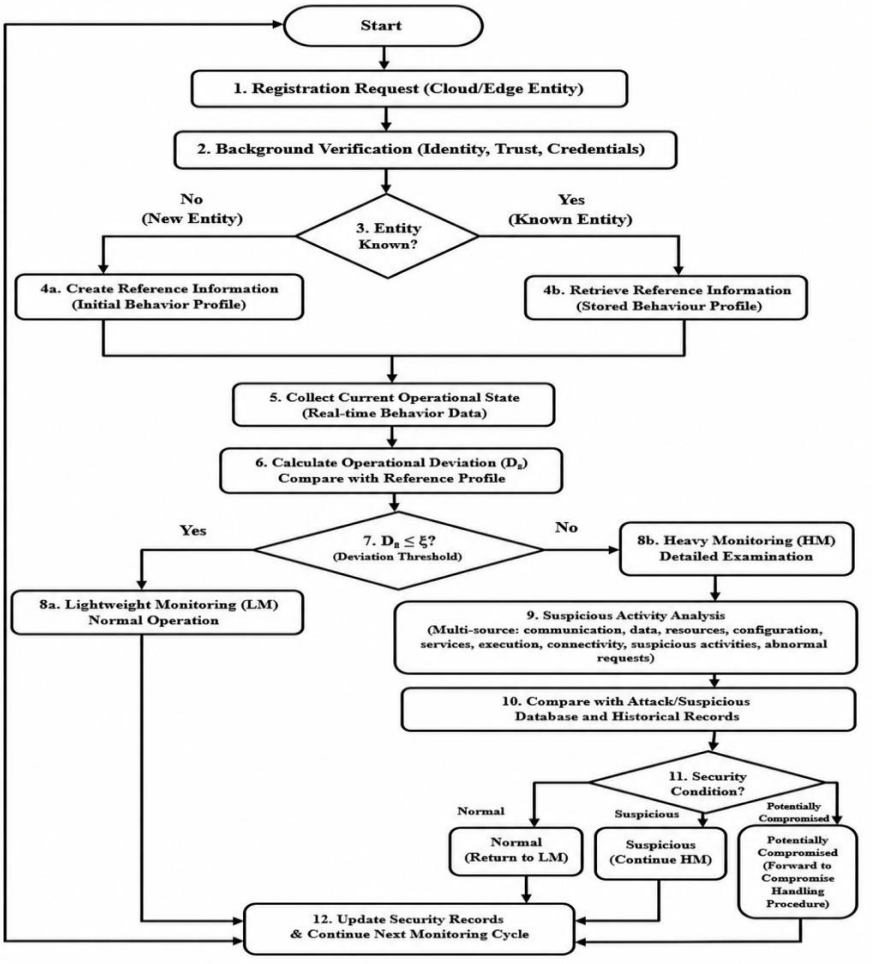

Fig. 2. Adaptive Security Monitoring and Threat Detection Workflow for Cloud and Edge Entities

The workflow is described as the constant security assessment of Cloud–Edge entities via verification, operational state assessment, adaptive monitoring, and security condition management. The workflow starts with a request for registration from a Cloud or Edge entity. Background verification assesses the identification, trustworthiness, and credentials before the entity proceeds to the monitoring stage.

The entity being verified is categorized into either a known or a new registered entity. In the case of a new registered entity, its first normal behavior is monitored to create the necessary reference. In the case of a known entity, its past reference data is collected.

After that, the system acquires the present status of the operations and compares it against the reference data. Depending on the output of the process, it is determined what level of monitoring is needed. If there are no deviations from the acceptable range of values, the system continues LM operation. Otherwise, HM starts to work.

In Heavy Monitoring, several parameters such as communication, data transfer, resource utilization, connectivity, configuration, service, performance, suspicious activities, and abnormal requests are evaluated. The collected observations are then compared with the suspicious data database and are correlated with past instances. This serves as an added criterion to distinguish a normal operation from an event related to security.

This process results in one of three possible outcomes; these are Normal, Suspicious, or Potentially Compromised. In case of a Normal outcome, the entity will return to the state of Lightweight Monitoring, but in case of a Suspicious outcome, Heavy Monitoring will be sustained in order to continue with investigation.

Algorithm 2: Spy Agent-Based Compromise Detection and Autonomous Agent AI Replacement

Input: Primary Explainable Agent AI, operational traces,

communication records, execution information, historical security records

Output: Hardened Replacement Explainable Agent AI, updated security status, suspicious-activity tracking record

```
BEGIN
  Initialize Primary Explainable Agent AI and its security functions
  Deploy independent Spy Agent outside the Primary Agent AI
  decision path
  Reference_Profile ← Establish trusted operational profile
                      of the Primary Agent AI
  WHILE Primary Agent AI is active DO
    Current_Traces ← Collect(
      Decision,
      Communication,
      Execution,
      Resource,
  Configuration,
      Service,
  Data_Handling
    )
    Deviation ← Compare(Current_Traces, Reference_Profile)
```

```
IF Deviation is insignificant THEN
    Continue normal operation
    Continue Spy Agent observation
ELSE
    Assessment ← Perform_Secondary_Verification(
        Current_Traces,
        Historical_Security_Records,
        Communication_Records,
        Execution_Information
    )
    IF Assessment = NORMAL_VARIATION THEN
        Resume normal operation
    ELSE IF Assessment = SUSPICIOUS_ACTIVITY THEN
        Increase monitoring intensity
        Retain suspicious activity for further analysis
    ELSE IF Assessment = AGENT_AI_COMPROMISE THEN
        Isolate Primary Agent AI
        Prevent sensitive operations and communication
        Legitimate_Knowledge ← Preserve(
            Learned_Patterns,
            Cloud_Information,
            Edge_Information,
            Resource_Information,
            Service_Information,
            Historical_Information,
            Security_Knowledge,
            Decision_Making_Knowledge
        )
        Exclude compromised or suspicious information
        Replacement_Agent ← Initialize
            Replacement Explainable Agent AI
        Migrated_Knowledge ←
            Transfer-Learning-Based Knowledge Migration(
                Legitimate_Knowledge
```

```
        )
      Configure Replacement_Agent using Migrated_Knowledge
      Apply Security Hardening according to
      the detected compromise
      Update security rules
      Strengthen behavioural monitoring
      Tracking_Record ← Initialize
        Suspicious-Activity Tracking
      Configure Tracking_Record with:
        Source
        Timestamp
        Communication_Path
        Associated_Cloud_Entity
        Associated_Edge_Entity
      Validate Replacement_Agent
      Deploy hardened Replacement_Agent
      Resume Cloud–Edge operations
      Activate Spy Agent observation of
      Replacement_Agent
      IF further suspicious activity is detected THEN
        Activate Suspicious-Activity Tracking
        Record source and communication information
        Report suspicious activity
      END IF
    END IF
  END IF
  Continue secure operation and Spy Agent monitoring
END WHILE
RETURN Replacement_Agent,
    Updated_Security_Status,
    Tracking_Record
END
```

3.4 Legitimate Knowledge Selection

After verifying the Agent AI Compromise, the infected Agent AI will be isolated, and its knowledge will be analyzed before recovery takes place. This is done in order to separate the information that can still be used to run the system from the one

involved in the breach. This knowledge comprises learned information, cloud and edge information, resource and service information, operational information, security information, and decision-making knowledge.

The legitimate knowledge set is defined as:

$$KL = \{ki \in KT \mid C(ki) = 0\} \quad (3)$$

K_L in equation (3) refers to the knowledge base of legitimate knowledge, K_T is the total knowledge that was determined using the affected Agent AI, k_i stands for individual knowledge and C(k_i) stands for the indicator that determines whether the knowledge is involved in the compromised activity or not. The value of C(k_i) being 0 shows that the knowledge has been kept for the purpose of recovery, while the value of C(k_i) = 1 implies that the knowledge is not included in the knowledge base.

Figure 3 depicts the flow chart of Spy Agent-Based Compromise Detection and Autonomous Agent AI Replacement Process. This flow chart encompasses the independent observation of Spy Agents as well as operational trace analysis for detecting any notable deviation in the Primary Explainable Agent AI. Following security analysis, the process differentiates between the three conditions: normal variability, suspicious behavior, and Agent AI Compromise, whereby the later triggers knowledge preservation, transfer learning based migration, replacement and security hardening of the agent AI.

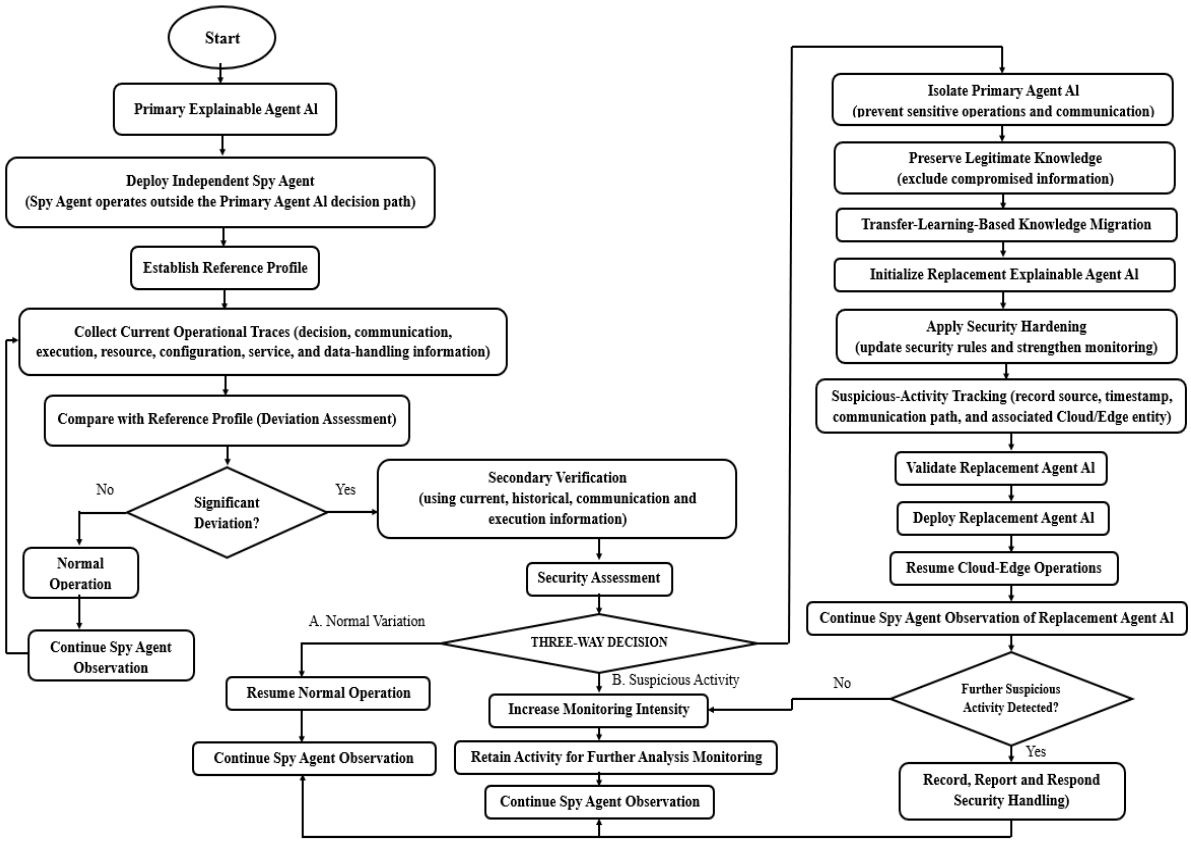

Fig. 3. Workflow of Spy Agent-Based Compromise Detection and Autonomous Agent AI Replacement

4. DATASET DESCRIPTION

In the experiment, the evaluation is done through three public data sets acquired from Kaggle, which will serve as examples of the two different needs for data in the system, namely operational-state learning under normal Cloud-Edge operations and detecting cybersecurity attacks. In particular, Cloud and Edge data sets will be utilized for operational-state learning and generating reference information for monitoring purposes, while the cybersecurity data set will serve separately for attack

pattern comparisons in Heavy Monitoring.

4.1 Cloud Operational-State Dataset

Multi-Tenant Cloud Dataset is used to represent the operational status of the Cloud services. This dataset contains the information regarding the Cloud resources and services, which may be used to characterize the availability, utilization, communications, and service status. Appropriate Cloud attributes are chosen depending on their ability to represent Cloud operational status.

The selected data is used to define the normal Cloud operational state and create the corresponding reference data. The reference data is then used to conduct Lightweight Monitoring to find out if the current state of the Cloud still operates within the allowed operational boundaries. It means that the dataset is used for normal operational-state training and not attack data training. Kaggle dataset link**:**

https://www.kaggle.com/datasets/ziya07/multi-tenant-cloud-dataset

4.2 Edge Operational-State Dataset

The Multi-Tier IoT Resource Allocation Dataset is used to depict the operational and resource status of Edge devices. The data set is comprised of information related to different computing layers, offering resource and performance attributes related to Edge computing.

In this case, the only data that are selected from the original dataset are those that are related to the Edge environment. The rest of the computing layers present in the original dataset are not introduced in the Cloud-Edge model. The selected Edge data are then used for learning the normal conditions and generating the Edge reference data.

Kaggle dataset link:

https://www.kaggle.com/datasets/ziya07/multi-tier-iot-resource-allocation-dataset

4.3 Cybersecurity Attack Dataset

Edge-IIoTset Cyber Security Dataset is employed alone in performing analysis for cyber security purposes. It comprises normal and attack instances gathered in an IIoT setup comprising Cloud, Fog, Edge, and IoT technologies and includes labeled security instances pertaining to various types of cyber attacks.

In this case, the data set is not used to create the Cloud and Edge reference data. Instead, the security observations are taken into account when there is heavy monitoring to perform comparisons for attack patterns. In this way, suspicious observations can be analyzed in relation to security patterns, which helps determine whether the state of affairs is Normal, Suspicious, or Potentially Compromised.

Kaggle dataset link:

https://www.kaggle.com/datasets/mohamedamineferrag/edgeiiotset-cyber-security-dataset-of-iot-iiot

4.4 Dataset Utilization in the System

The three datasets serve distinct purposes within the experimental process. The Multi-Tenant Cloud Dataset supplies the data that is needed for Cloud operational-state learning, whereas the Multi-Tier IoT Resource Allocation Dataset serves as the source of the data needed for Edge operational-state learning. This data is utilized to define the reference states of Cloud and Edge entities.

The Edge-IIoTset is employed individually in attack pattern comparison and cybersecurity analysis. In cases where the present Cloud or Edge state is far from its reference state, the system goes into Heavy Monitoring. The observed suspicious state is then compared to the security patterns in Edge-IIoTset.

The Table 1 describes the datasets chosen for the experimental study and specifies their roles in the system. The Multi-Tenant Cloud Dataset offers information about Cloud resources, whereas the Multi-Tier IoT Resource Allocation Dataset offers information about the performance in relation to the Edge environment. The Edge-IIoTset dataset serves for the purposes of attack analysis and comprises both normal and attack data.

Table 1 Comparison of datasets used in the experimental evaluation

Dataset	Computing Environment	Primary Purpose	Information Utilized	Role in the System
Multi-Tenant Cloud Dataset	Cloud	Normal Cloud operational-state learning	Resource and service-related operational information	Cloud reference-state construction
Multi-Tier IoT Resource Allocation Dataset	Edge	Normal Edge operational-state learning	Resource, connectivity, and execution-related information	Edge reference-state construction
Edge-IIoTset Cyber Security Dataset	IoT with Cloud–Edge components	Cybersecurity attack detection	Normal and attack-related security observations	Attack-pattern comparison and security analysis

5. EXPERIMENTAL SETUP

The experimental assessment of the Cloud–Edge security architecture with adaptive monitoring and Agent AI compromise handling with autonomous Agent AI is performed within the test Cloud–Edge computing environment. The assessment considers learning in the operational state, adaptive monitoring, cybersecurity analysis, Agent AI compromise verification, and Agent AI secure replacement. The assessment uses three datasets: Multi-Tenant Cloud Dataset for learning in the Cloud operational state; Multi-Tier IoT Resource Allocation Dataset and Edge-IIoTset for attacks' pattern and cybersecurity analysis.

Cloud and Edge datasets are processed separately to create the corresponding reference information. In the Cloud dataset, records are taken to represent the normal behavior of the Cloud system, whereas the Edge dataset includes those records that correspond to the resource and behavioral state of the Edge systems. Altogether, 500 records are taken from each dataset, making the total number of records 1500. The datasets are split separately in such a way that 350 records become training records, 75 are validation records, and 75 are testing records. Training records are used to generate reference information, whereas the validation records are used for the selection of parameters.

At the monitoring stage, Explainable Agent AI gets an observation about the state of the system from the Cloud or Edge entity and compares it with the respective reference data. In case the deviation stays inside the acceptable threshold, then the operation mode will be Lightweight Monitoring for that entity. If any deviation is detected as suspicious, then Heavy Monitoring will start. Such an evaluation takes into account the resource utilization, connectivity, service availability, execution data, communication data, and characteristics of the data.

Classification of the security assessment into Normal, Suspicious, and Potential Compromise takes place. In case of normal classification, the agent continues to be monitored under the Lightweight Monitoring regime. In the case of suspicious classification, the agent stays under the regime of Heavy Monitoring. Once potential compromise is verified, the second process of security begins. The verification is carried out by an independent Spy Agent.

After the detection of compromise, the valid operational knowledge necessary to sustain the service is retained, but any knowledge linked to the detected compromise is prevented from being migrated. A new Explainable Agent AI system is created and knowledge is migrated using the Transfer-Learning-Based Knowledge Migration. This new Agent AI system is configured with enhanced security features and capability for monitoring suspicious activities. After validating the knowledge, the new agent assumes operation of the cloud-edge services under the constant surveillance of Spy Agent.

Table 2 Experimental Configuration

Parameter	**Configuration**
Cloud Dataset	Multi-Tenant Cloud Dataset
Edge Dataset	Multi-Tier IoT Resource Allocation Dataset
Cybersecurity Dataset	Edge-IIoTset

Records per Dataset	500
Total Records	1500
Training Records	350 per dataset
Validation Records	75 per dataset
Testing Records	75 per dataset
Data Split	70:15:15 independently for each dataset
Cloud Data Utilization	Normal Cloud operational-state learning
Edge Data Utilization	Normal Edge resource-state learning
Security Data Utilization	Attack-pattern and cybersecurity evaluation
Monitoring Levels	Lightweight Monitoring and Heavy Monitoring
Security Assessment	Normal, Suspicious, Potentially Compromised
Compromise Verification	Independent Spy Agent
Compromise Response	Agent AI isolation
Knowledge Preservation	Validated legitimate operational knowledge
Knowledge Migration	Transfer-Learning-Based Knowledge Migration
Replacement Mechanism	Secure Replacement Explainable Agent AI
Security Enhancement	Replacement-specific security hardening
Activity Tracking	Source, timestamp, communication path, affected entity, associated activity
Post-Replacement Monitoring	Continuous Spy Agent observation
Comparison Condition	Same datasets, partitions, preprocessing, and testing conditions
Evaluation Metrics	Accuracy, Precision, Recall, F1-score, and FPR

6. RESULTS AND DISCUSSION

Three different datasets were used for the experiment, which included learning about the operational state in the Cloud environment, learning about the state of the resource in the Edge environment, and cybersecurity analytics. All datasets consisted of 500 records each, giving a total of 1,500 records altogether. These were then further divided into 350 training records, 75 validation records, and 75 testing records. Below are the outcomes of the security classification testing with 75 records.

6.1 Security Classification Performance

Confusion matrix contained 36 True Positives (TP), 36 True Negatives (TN), 2 False Positives (FP), and 1 False Negatives (FN). This means that out of 75 samples used for testing, 72 were classified correctly.

Table 3 results have shown that the security technique has succeeded in identifying many of the observations related to security. 96.00% Accuracy has been achieved, which means that most of the testing observations were classified in a correct manner. The value of 97.30% Recall is crucial in the case, since it shows that most of the positive cases have been identified by the classifier. Specificity and Precision both have achieved the value of 94.74%, and 96.00% F1-score shows the balance between these two. FNR was only 2.70%.

Table 3 Security Classification Results

Metric	Result
True Positive (TP)	36
True Negative (TN)	36
False Positive (FP)	2
False Negative (FN)	1
Accuracy (%)	96
Precision (%)	94.74
Recall (%)	97.3
F1-score (%)	96
Specificity (%)	94.74
False Positive Rate (FPR) (%)	5.26
False Negative Rate (FNR) (%)	2.7
Response Time (ms)	6.8

6.2 Confusion Matrix Analysis

The confusion matrix generated from the security testing sample is illustrated in Fig. 4 below. The system can distinguish 36 positives and 36 negatives accurately. There are only two negatives that are mistakenly recognized as positives, and one positive is mistakenly identified as negative.

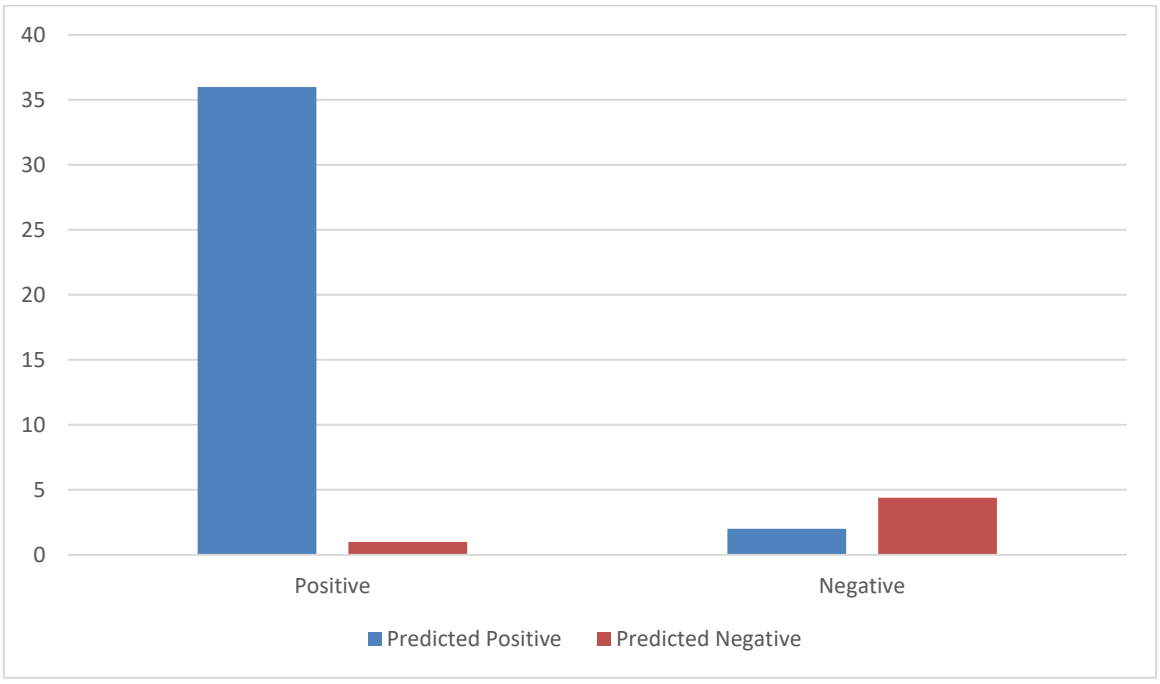

Fig. 4 Comparison of Prediction Outcomes for Positive and Negative Cases

6.3 Performance Comparison

Comparison with Table 4 indicates that the Secure Cloud-Edge Entity Verification and Spy Agent-Based Compromise Detection framework exhibits greater Detection Accuracy. Detection Accuracy rises from 94.78% to 96.00%, thus making an improvement of 1.22 percentage points. FPR remains very similar, varying just by 0.04 percentage points. Response Time increases by 0.425 milliseconds. This small increment is explained by the extra security actions taken by the security framework.

Table 4 Performance Comparison

Metric	AI-Driven IoT Security with KKT Optimization	Secure Cloud-Edge Entity Verification and Spy Agent-Based Compromise Detection	Difference
Detection Accuracy (%)	94.78	96.00	+1.22
FPR (%)	5.22	5.26	+0.04
Response Time (ms)	6.375	6.800	+0.425

6.4 Response Time Comparison

Response Time comparison is depicted in Fig. 5. The value measured by the KKT approach is 6.375 ms while the one for Secure Cloud-Edge Entity Verification and Spy Agent-Based Compromise Detection approach is 6.800 ms. The difference is 0.425 ms.

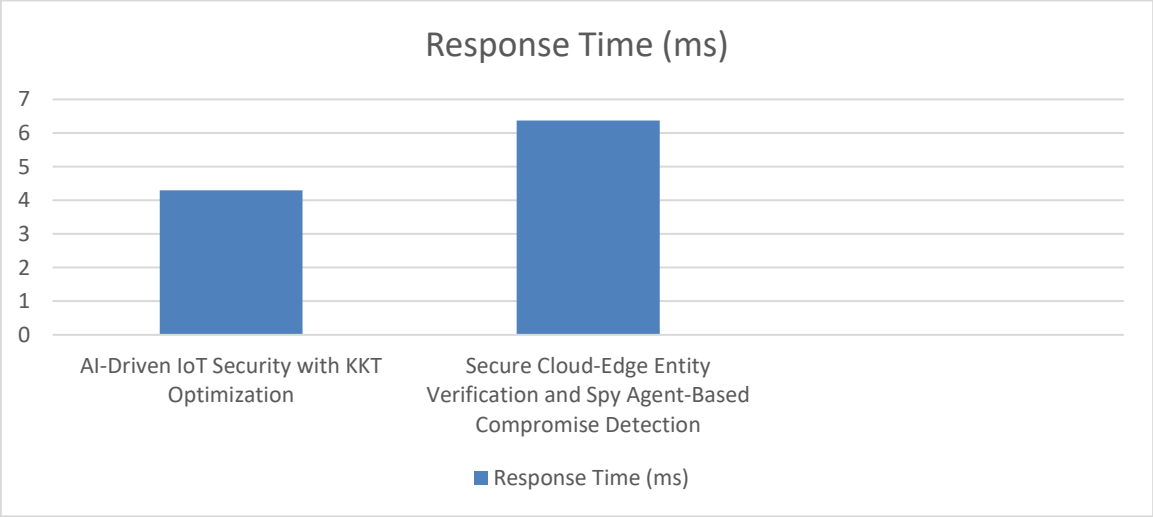

Fig. 5 Comparison of Response Time Between AI-Driven IoT Security with KKT Optimization and the Proposed Approach

6.5 Reproducibility

The design of the experiment allows the whole assessment procedure to be conducted in the same manner repeatedly. There are three sets of data which contain 500 observations in each set. Each of those datasets was separately split into 350 training observations, 75 validation observations, and 75 testing observations.

Table 5 Reproducibility Configuration

Parameter	Configuration
Multi-Tenant Cloud Dataset	500 records
Multi-Tier IoT Resource Allocation Dataset	500 records
Edge-IIoTset	500 records
Total records	1,500
Training records	350 per dataset
Validation records	75 per dataset
Testing records	75 per dataset

Partition ratio	70:15:15
Security testing subset	75 records
TP	36
TN	36
FP	2
FN	1
Response Time	6.800 ms

To ensure reproducibility, the same procedure for data partitioning and data preprocessing, training and validation process, test procedure, threshold parameters, and evaluation metrics must be maintained while performing the same experiment. It will guarantee that any variations in the performance will be due to the security mechanism being tested and not because of experimental variations.

7. CONCLUSION

The Secure Cloud-Edge Entity Verification and Spy Agent-Based Compromise Detection framework has been formulated with the purpose of enhancing the security of Agent AI deployed within Cloud-Edge infrastructure. It has been formulated for tackling suspicious activities using normal operation state learning, adaptive monitoring, cybersecurity analysis, Spy Agent independent verification, isolation of Agent AI, Transfer Learning-Based Knowledge Migration, replacement, security hardening, and post-replacement monitoring.

The experimental analysis employed 1,500 data points, which were made up of 500 data points in the Multi-Tenant Cloud Dataset, Multi-Tier IoT Resource Allocation Dataset, and Edge-IIoTset. Each of the datasets was split individually into training (70%), validation (15%), and testing (15%) data points. In terms of the 75 data points security test set, the mechanism produced 36 TP, 36 TN, 2 FP, and 1 FN, giving an Accuracy of 96.00%, Precision of 94.74%, Recall of 97.30%, F1-score of 96.00%, and Specificity of 94.74%.

As per the AI-Driven IoT Security with KKT Optimization technique, there is a Detection Accuracy of 94.78%, an FPR of 5.22% and a Response Time of 6.375 ms. The Secure Cloud-Edge Entity Verification and Spy Agent-Based Compromise Detection system yielded a Detection Accuracy of 96.00%, an FPR of 5.26%, and a Response Time of 6.800 ms. Hence, there was an increase in Detection Accuracy by 1.22%. The difference in Response Time was 0.425 ms.

The significant strength of the Secure Cloud-Edge Entity Verification and Spy Agent-Based Compromise Detection system is that it makes security processing extend itself not only to detection but also to autonomous recovery. The Self-Protective Explainable Agent AI framework detects compromise, verifies the compromised Agent AI using a Spy Agent, isolates the compromised Agent AI, retains the correct knowledge, executes Transfer-Learning-Based Knowledge Migration, replaces the compromised Agent AI with the secured one, and finally implements security hardening.

Hence, the outcomes show that the Secure Cloud-Edge Entity Verification and Spy Agent-Based Compromise Detection scheme has been effective in security classification along with automated compromise management and Agent AI recovery. This approach is ideal for the cloud-edge environment with Agent AI systems that need to be constantly secure from any compromise attacks.

7.1 Limitations and Future Work

The Secure Cloud-Edge Entity Verification and Spy Agent-Based Compromise Detection mechanism is limited by certain factors that could be improved in future work. First, the experimental evaluation of the mechanism uses a limited sample of 1,500 records; hence, the evaluation does not necessarily reflect the real-world environment of Cloud-Edge systems. Besides, the use of benchmark datasets does not capture all possible dynamic resource conditions, communication, and new attacks that affect the Agent AI environment. Moreover, in current experiments, only the security classification and compromise detection performance have been tested, but the computational overhead, communication overhead, and energy consumption due to continuous monitoring using Spy Agents need to be evaluated. Finally, the Transfer-Learning-Based Knowledge Migration and

replacement process needs long-term evaluation to test how well the replacement of compromised agents perform after repeated attacks. Future research will thus consider evaluating the mechanism using large and varied datasets as well as real-time Cloud-Edge testbeds. New experimental setups will also be carried out to test unknown and evolving attack behaviours to improve adaptive compromise detection. Moreover, future work will concentrate on reducing Spy Agent overhead and coordinating multiple Spy Agents.

Author Contributions

Funding

Data Availability Statement

The datasets used in this study are publicly available benchmark datasets. The Multi-Tenant Cloud Dataset, Multi-Tier IoT Resource Allocation Dataset, and Edge-IIoTset were used for Cloud operational-state learning, Edge resource-state learning, and cybersecurity analysis, respectively. The datasets were processed and partitioned into training, validation, and testing subsets as described in the experimental setup. The data used for the experiments are available from their respective public repositories, and no privately collected or personally identifiable data were used in this study.

Conflicts of Interest

Acknowledgement

REFERENCES

[1] Diamantini, C., Mele, A., Mircoli, A. *et al.* A Graph RAG Approach to Enhance Explainability in Dataset Discovery. *Data Sci. Eng.* **11**, 30–52 (2026). https://doi.org/10.1007/s41019-025-00313-x

[2] D. Gernhardt, S. Groš and D. Ćutić, "Modeling Advanced Persistent Threats Through Organizational Roles for Adversary Emulation and Cyber Threat Intelligence," in *IEEE Access*, vol. 14, pp. 111636-111665, 2026.

[3] S. Reza Ghazinour Naeini, A. Shameli-Sendi and M. Jabbarifar, "Protecting Industrial Control Systems From Shodan Exploitation Through Advanced Traffic Analysis," in *IEEE Access*, vol. 13, pp. 45985-45997, 2025.

[4] G. Yang, Z. Sun and Y. Wang, "ShellBox: Adversarially Enhanced LLM-Interactive Honeypot Framework," in *IEEE Access*, vol. 13, pp. 143618-143630, 2025.

[5] M. Rajaee and K. Mazlumi, "Multi-Agent Distributed Deep Learning Algorithm to Detect Cyber-Attacks in Distance Relays," in *IEEE Access*, vol. 11, pp. 10842-10849, 2023.

[6] D. Gaspar, P. Silva and C. Silva, "Explainable AI for Intrusion Detection Systems: LIME and SHAP Applicability on Multi-Layer Perceptron," in *IEEE Access*, vol. 12, pp. 30164-30175, 2024.

[7] P. Turaka and S. K. Panigrahy, "Dynamic Attack Detection in IoT Networks: An Ensemble Learning Approach With Q-Learning and Explainable AI," in *IEEE Access*, vol. 12, pp. 161925-161940, 2024.

[8] S. S. Mahadik, P. M. Pawar and R. Muthalagu, "Edge-Federated Learning-Based Intelligent Intrusion Detection System for Heterogeneous Internet of Things," in *IEEE Access*, vol. 12, pp. 81736-81757, 2024.

[9] Qi, Z., Su, X., Wang, Y. *et al.* Real-Time Dynamic Response Identification for Highway Structural Health Monitoring Data. *Data Sci. Eng.* **11**, 248–259 (2026). https://doi.org/10.1007/s41019-025-00336-4

[10] M. K. Nallakaruppan, S. R. K. Somayaji, S. Fuladi, F. Benedetto, S. K. Ulaganathan and G. Yenduri, "Enhancing Security of Host-Based Intrusion Detection Systems for the Internet of Things," in *IEEE Access*, vol. 12, pp. 31788-31797, 2024.

[11] O. I. Aboulola, E. A. Alabdulqader, A. A. AlArfaj, S. Alsubai and T. -H. Kim, "An Automated Approach for Predicting Road Traffic Accident Severity Using Transformer Learning and Explainable AI Technique," in IEEE Access, vol. 12, pp. 61062-61072, 2024.

[12] T. Turan and E. U. Küçüksılle, "Summarization, Prediction, and Analysis of Turkish Constitutional Court Decisions With Explainable Artificial Intelligence and a Hybrid Natural Language Processing Method," in IEEE Access, vol. 13, pp. 59766-59779, 2025.

[13] S. Shahkar, "Cooperative Localization of Multi-Agent Autonomous Aerial Vehicle (AAV) Networks in Intelligent Transportation Systems," in *IEEE Open Journal of Intelligent Transportation Systems*, vol. 6, pp. 49-66, 2025.

[14] Zhao, Y., Xu, Y., Liu, N. *et al.* Generating Counterfactual Temporal Motifs: Unraveling the Mysteries of Temporal Graph Neural Networks. *Data Sci. Eng.* **11**, 199–212 (2026). https://doi.org/10.1007/s41019-025-00321-x

[15] S. Sadhwani, U. K. Modi, R. Muthalagu and P. M. Pawar, "SmartSentry: Cyber Threat Intelligence in Industrial IoT," in *IEEE Access*, vol. 12, pp. 34720-34740, 2024.

[16] H. A. Noman, O. M. F. Abu-Sharkh and S. A. Noman, "Log Poisoning Attacks in IoT: Methodologies, Evasion, Detection, Mitigation, and Criticality Analysis," in *IEEE Access*, vol. 12, pp. 118295-118314, 2024.

[17] Yin, H., Qu, L., Chen, T. *et al.* On-Device Recommender Systems: A Comprehensive Survey. *Data Sci. Eng.* **10**, 591–620 (2025). https://doi.org/10.1007/s41019-025-00308-8

[18] H. Bondok *et al.*, "A Trojan Attack Against Smart Grid Federated Learning and Countermeasures," in *IEEE Access*, vol. 12, pp. 191828-191846, 2024.

[19] S. M. Hassan, M. Murtadha Mohamad, F. Binti Muchtar and F. Bin Yusuf Patel Dawoodi, "Enhancing MANET Security Through Federated Learning and Multiobjective Optimization: A Trust-Aware Routing Framework," in *IEEE Access*, vol. 12, pp. 181149-181178, 2024.

[20] N. Rodis, C. Sardianos, P. Radoglou-Grammatikis, P. Sarigiannidis, I. Varlamis and G. T. Papadopoulos, "Multimodal Explainable Artificial Intelligence: A Comprehensive Review of Methodological Advances and Future Research Directions," in IEEE Access, vol. 12, pp. 159794-159820, 2024.

[21] M. T. Masud, M. Keshk, N. Moustafa, I. Linkov and D. K. Emge, "Explainable Artificial Intelligence for Resilient Security Applications in the Internet of Things," in IEEE Open Journal of the Communications Society, vol. 6, pp. 2877-2906, 2025.

[22] M. Sajid Farooq et al., "Developing a Transparent Anaemia Prediction Model Empowered With Explainable Artificial Intelligence," in IEEE Access, vol. 13, pp. 1307-1318, 2025, doi: 10.1109/ACCESS.2024.3522080.

[23] K. M. Kearney, T. Ordonez Diaz, J. B. Harley and J. A. Nichols, "Transfer Learning With Simulated and Recorded Data Improves Predictions of Lateral Pinch Thumb-Tip Forces," in *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 34, pp. 3224-3231, 2026.

[24] H. Tourab et al., "The Use of Machine Learning and Explainable Artificial Intelligence in Gut Microbiome Research: A Scoping Review," in IEEE Journal of Biomedical and Health Informatics, vol. 30, no. 1, pp. 717-731, Jan. 2026, doi: 10.1109/JBHI.2025.3593198.

[25] Ragab, M., Savateev, Y., Oliver, H. *et al.* ESPRESSO: A Framework to Empower Search on the Decentralized Web. *Data Sci. Eng.* **9**, 431–448 (2024). https://doi.org/10.1007/s41019-024-00263-w

[26] S. H. Mahir et al., "Advanced Hydro-Informatic Modeling Through Feedforward Neural Network, Federated Learning, and Explainable AI for Enhancing Flood Prediction," in IEEE Open Journal of the Computer Society, vol. 6, pp. 726-738, 2025, doi: 10.1109/OJCS.2025.3556424.

[27] A Siam, M. Alazab, A. Awajan, M. R. Hasan, A. Obeidat and N. Faruqui, "IP SafeGuard–An AI-Driven Malicious IP Detection Framework," in *IEEE Access*, vol. 13, pp. 90249-90261, 2025.

[28] Y. Hmimou, M. Tabaa, A. Khiat and Z. Hidila, "A Multi-Agent System for Cybersecurity Threat Detection and Correlation Using Large Language Models," in *IEEE Access*, vol. 13, pp. 150199-150215, 2025.

[29] S. Bani Melhem, M. Golec, A. Alwarafy and Y. Khamayseh, "LENS: Lightweight and Explainable LLM-Based APT Detection at the Edge for 6G Security," in *IEEE Access*, vol. 13, pp. 172402-172415, 2025.

[30] K. C. Chaganti, "A Scalable, Lightweight AI-Driven Security Framework for IoT Ecosystems: Optimization and Game Theory Approaches," in *IEEE Access*, vol. 13, pp. 72235-72247, 2025.