Spectrum of Engineering Sciences
ISSN (e) 3007-3138 (p) 3007-312X Volume 4, Issue 5, 2026

EXPLAINABLE AI-BASED INTRUSION DETECTION SYSTEM FOR CLOUD-IOT SECURITY

Maqsood Ahmed Dero[*1], Dr Mairaj Nabi[2], Dr Marina[3], Dr Rahila Tallal[4], Arsalan Rajper[5], Mah Saba Maheen[6]

[*1,2,3,5,6]*Department of Information Technology, Shaheed Benazir Bhutto University, Shaheed Benazirabad, Sindh, Pakistan*
[4]*Department of Law, Shaheed Zulfiqar Ali Bhutto University of Law, Karachi, Sindh, Pakistan*

[*1]maqsood.ahmed0717@gmail.com, [2]mairajbhatti@sbbusba.edu.pk, [3]marina@sbbusba.edu.pk, [4]rahila.tallal@gmail.com, [6]sabamaheen@sbbusba.edu.pk

DOI: https://doi.org/10.5281/zenodo.20376204

Keywords
Explainable Artificial Intelligence, Intrusion Detection System, Cloud-IoT Security, SHAP, Random Forest, Network Security, Privacy

Article History
Received: 28 March 2026
Accepted: 07 May 2026
Published: 25 May 2026

Corresponding Author: *
Maqsood Ahmed Dero

Abstract
Cloud computing and the Internet of Things (IoT) are coming together quickly , creating a confusing security situation with diverse technologies needing integration along with overloaded networks, shared environments and ongoing threats such as DoS (Denial of Service), Distributed DoS (DDoS), man in the middle and data breaches . [1]. The current level of reliance on existing IDSs within the Cloud-IoT domain is on black box machine learning techniques of various types. Many IDSs using this type of technology can detect intrusions with great accuracy but they offer no insight into how they arrive at their conclusions. This is a major limitation when considering the importance of analysts understanding and having confidence in the recommendations made by the automated application of these systems. [2]. An XAI-IDS is proposed in this paper, which is an XAI-based Intrusion Detection System (IDS) that uses a Random Forest (RF) ensemble classifier with SHAP to provide high accuracy and human-readable explanations at both the feature and instance level while detecting intrusions in Cloud-IoT Security. Both UNSW-NB15 and N-BaIoT are used as test datasets, which take into consideration four of the main components of Cloud-IoT Security: protection of data/information, networks, access control and privacy. [3]. The experimental findings reflect an overall detection accuracy rate of 98.3%, a macro F1-score of 97.8%, and a false positive rate of only 0.9%. In addition, using SHAP analysis, the most important features for each attack class can be identified, which allows security analysts to positively audit their decisions, optimize their policies, and respond properly to cloud-IoT threats. The proposed framework is a deployable, transparent, and multi-component security framework. It overcomes the issue identified in previous research where most solutions focused on only one component of a multi-layered security solution. [4].

I. INTRODUCTION

A. Background

The Internet of Things, or the IoT, describes the physical connection of any tangible thing capable of having an electronic sensor, actuator, or communicating device included or embedded into it, to allow for remote access to that object for gathering and exchanging data without the

intervention of humans. [1]. Due to the high capacity for storing data, the elasticity in processing large volumes of data, and the ability to provide Infrastructure as a Service, Platform as a Service, and Software as a Service through cloud computing, combining IoT and cloud solutions have created tremendous innovative opportunities in many industries including, but not limited to, smart health care, smart cities, industrial automation, and transportation [2]. At the same time, the convergence of Cloud-IOT has created an increased attack footprint and opened the door to a wide range of threats such as exposing data, intrusion into networks, exploiting protocols and browsers, flaws related to multi-tenancy, and unauthorized access [3].

One of the main issues with deploying cloud IoT technology is the protection of devices operating in unattended, resource- constrained, easy to eavesdrop, and vulnerable to physical compromise environments. Therefore, it is essential to have security measures in place for cloud IoT systems to reduce these risks and ensure data protection. [5]. The IoT devices gather and analyze data without dependence on people, as there is no need; thus, the development of new forms of security that are not just based on asymmetric keys or being IP (Internet Protocol)-centric must occur. Since the four main components of cloud/IOT security (network, data, access, and privacy) each need their own unique forms of security, and the current academic literature has focused on only one of the four components and not all four and how they relate to each other. [4].

B. Problem Statement

While based on machine learning intrusion detection systems (IDSs) are used in modern cloud-IoT scenarios, the majority of them rely on opaque black-box models with high detection rates. They raise a fundamental issue regarding trust: security analysts do not know the reason for the alert, cannot verify the logic leading to a classification, and do not know whether the model is correctly classifying real intrusions or simply making false positive classifications based upon spurious correlations in the training data [6]. . The absence of interpretability leads to a direct conflict between audit requirements and responsible AI principles, especially within regulated health care settings and IoT in industrial environments. In addition, previous survey research indicates that the majority of proposed security solutions for cloud-IoT address only one of the four components of security and that no complete approach exists that combines an increase in explainability along with security through prevention of all four components. [4].

C. Research Motivation and Objectives

Thanks to Explainable AI (XAI) techniques, which utilize SHAP values derived from cooperative game theory, we can formally analyze how much a single input feature contributes to a model's prediction [8]. By combining SHAP explanations and a high-quality Intrusion Detection System (IDS), we are able to create a security operation that satisfies the various stakeholders' need for trust in Cloud-Internet of Things (Cloud-IoT) security operational capabilities. This gives us an effective and robust security operation that can provide both trust and efficiency. There were three unifying motivators behind this paper. The three needs are: (1) explainable decisions made by humans (as opposed to by machine), (2) accurate multi-attacks found in cloud-IoT environments and (3) a framework to unify all four types of security within the cloud-IoT domain. The objectives of this research are: (i) to create and implement an IDS architecture that meets all four areas of the cloud-IoT security domain while providing intrusion detection capabilities, (ii) to develop SHAP explanations that provide both feature-level and instance-level interpretability; (iii) to assess performance against two publicly available IoT/cloud security data sets; (iv) to provide measurable improvements to previous black-box methods both in terms of overall accuracy and amount of explanation detail.

D. Research Contributions

- An integrated XAI-IDS framework for dealing with the multiple forms of threats associated with network and data security, access

control and privacy in the Cloud-IoT environment [4].

- State-of-the-art detection accuracy and a way to assign assaults to particular instances are obtained by integrating SHAP values with Random Forest ensemble categorization.
- Empirical evaluation using two benchmark datasets (UNSW-NB15 and N-BaIoT), shows 98.3% accuracy, 97.8% macro F1 and 0.9% false-positive rate.
- Providing a taxonomy that will help analysts refine their policies by identifying features of interest correlated to specific types of attacks on Cloud-IoT environments through SHAP-ordered feature imports.
- Conducting critical comparative analyses of the approach and other seven other systems and showing that they are superior in detecting and preventing several types of attacks on Cloud-IoT environments [4].

E. Paper Organization

The review of research in the areas of cloud-IoT security, IDS, and XAI are presented in Section II. The architecture of an XAI-IDS and the methodology for the study are outlined in Section III; whereas Section IV relates to implementation of the proposed architectures. Section V includes the results and a discussion of the results pertaining to the study conducted.
Lastly, Section VI of the report provides a summary with a discussion of future work objectives.

II. Literature Review and Related Work

. Cloud-IoT Security Landscape

There are three main functions that characterize the relationship between IoT and Cloud-based systems: communication, computing, and storage [3]. IoT devices present a set of distinctive security challenges compared to traditional cloud computing methods due to their limited computational power and resources, which may prevent the implementation of complex cryptographic mechanisms for protection [5]. According to scholarly literature, four major security dimensions associated with the IoT and cloud integration environment are Access Control, Network Security, Data Security, and Privacy [4]. Access Control enables a minimized risk of security breach through limiting access to sensitive data sets via the identification and authentication of authorized users only [9].Data Security ensures that unauthorized users cannot expand access to sensitive information by utilizing encryption technologies, thereby providing Data Confidentiality, Data Integrity and Data Availability [10]. Network Security refers to the policies and procedures (e.g., firewalls, intrusion detection systems, etc.) which prohibit unauthorized access to the network resources and systems by third parties. [11]. Privacy refers to controlling the collection, utilization and sharing of personal information [12]. Recent studies have shown that the environment for IoT and Cloud-based security systems present unique challenges when designing cloud-based systems.
According to Al-Garadi et al. [14], ensemble machine learning methods provide superior performance relative to single classifiers when compared across a variety of attack types. In their comprehensive analysis of vulnerability classes affecting IoT devices, Abomhara and Køien [15] identified the following classes: Eavesdropping, Physical Access, Replay Attacks and Side-Channel Attacks. Each of these areas are directly applicable to the design of IDS-Intrusion Detection Systems for
Cloud-IoT integration.B.

Existing Intrusion Detection Approaches

The authors of [6] used both classic and machine learning techniques to perform a detailed review of NIDS for IoT devices; they provided a full taxonomy of the various detection techniques that are classified by machine learning paradigm.
According to [16], Sugi and Ratna found that while their LSTM and KNN algorithms were successful in detecting attacks in simulation scenarios, they did not include real- life validation or testing. [17] demonstrates the use of the CorrAUC wrapper- based feature selection algorithm, which includes the use of Shannon Entropy and Bijective Soft Sets integrated with machine learning algorithms, produced accuracy

levels of 96% or better; however, there was no element of explainability associated with these results. The Cluster Based Algorithm (CBA) and Key Match Algorithm (KMA) were proposed by [18] as intrusion detection algorithms to detect routing attacks; the CBA produced a positive detection rate for intrusion detection between 76 and 96 percent, and there were two examples of adversaries in the report. An automated knowledge driven intrusion detection system, known as Kalis, was developed by [19] that provided real time detection and required no changes to existing IoT systems to operate. The Kalis system's accuracy will not improve over time because it has no machine learning component for learning. The signature-based intrusion detection system for cloud-IoT developed by [20] limited intrusions through the use of temporal and geographical user profiles. However, due to the nature of signature-based systems, they have inherent limitations in regards to zero day or zero hour attacks..

C. Explainable AI in Security

Lundberg and Lee's groundbreaking work on SHAP (SHapley Additive exPlanations), which is a unified approach to interpretability in AI that uses cooperative game theory allows you to assign each feature of an AI model a contribution amount for predicting the outcome of an event [8]. SHAP values are suitable for many applications, including security; its theoretical properties, such as accuracy on a local scale (local accuracy), missing values (missingness), and consistency across different models (consistency) can all be demonstrated as being reliable. When using SHAP in the broader security domain, it has been used for numerous applications, including classifying network traffic for identifying features that stand out in an attack and providing security analysts with the ability to provide effective context when determining whether or not to detect. Currently, there is no research in the academic literature that focuses on applying XAI specifically to cloud/internet of things intrusion detection systems, including the security model's four component security model. Toth & Chowdhury (2021)'s research integrated honeynets with cloud networks to boost the rate at which intrusion detection systems could identify intruders [21], but provided no explanation of the results. Similarly, Alam (2020)'s study on how to create a mobile ad hoc network (MANET) that provides secure access to 5G devices on the Internet Of Things via the cloud [22] did not consider the interpretability gap. This paper aims to fill this gap by combining the benefits of using accurate high accuracy ensemble classifiers with SHAP to explain predicted outcomes of models used in cloud / internet of things literature.

Summary Comparison of Related Works

Table I. Comparison of Related Works on Cloud-IoT Intrusion Detection

Author	Method	Dataset	Security Components	Accuracy	XAI
Chaabouni et al. [6]	ML/DL Survey (NIDS)	Multiple	Network	Variable	None
Sugi & Ratna [16]	LSTM, KNN	Simulation	Network	~94%	None
Shafiq et al. [17]	CorrAUC + ML	IoT Traffic	Network	>96%	None
Choudhary & Kesswani [18]	KMA, CBA	Simulation	Network	76–96%	None
Midi et al. [19]	Kalis (Knowledge-driven)	Real IoT	Data Security	N/A	None

Rebbah et al. [20]	Signature-based IDS	Cloud-IoT	Data Security	Moderate	None
Alshehri et al. [23]	DSA-Block (Blockchain)	Cloud-IoT	Access Control	High	None
Proposed XAI-IDS	**RF + SHAP**	**UNSW-NB15,N-BaIoT**	**All Four Components**	**98.3%**	**SHAP**

E. Research Gap

The study performed by Gimba et al. [4] clearly indicates that the majority of current cloud-IoT security solutions have only one security component, and they suggest that in the future, machine learning and blockchain could be good directions to pursue. According to Table I, previous research does not support the idea of creating a unified intrusion detection system that encompasses all four cloud-iot security components along with a . Therefore, to fill this void, the authors of this paper offer an XAI-IDS framework made up of Random Forest classification, SHAP-based explanations, and a multi-component threat model.

A. Research Design

This study employs an experimental quantification methodology. An explainable artificial intelligence intrusion detection system (XAI-IDS) was formulated, taught and observed while being contained within two public domain benchmark databases. Data comparing the XAI-IDS performance against six baseline approaches confirms that the XAI-IDS has improved performance by statistical means. The qualitative evaluation of explanation quality uses a statistical method (i.e. the distribution of SHAP values), which were evaluated and confirmed as qualified by experienced data security practitioners.

B. Proposed XAI-IDS Architecture

Fig. 1 — XAI-IDS Pipeline: Data loading → preprocessing → RF Classification → SHAP Explanation Engine → Alert & Reporting

The five-stage pipeline structure of the XAI-IDS framework shown in Fig 1 consists of the following stages; Stage 1 represents continuous collection of network traffic, system logs, and device telemetry from the cloud-IoT infrastructure (Data Ingestion). At Stage 2, the Data Preprocessing and Feature Extraction techniques of normalisation, one-hot-encoding and feature selection create a structured feature matrix. Stage 3 produces a multi-class intrusion detection using Random Forest Classification and outputs class probability vectors per traffic instance. Stage 4 calculates SHAP values for the results of Stage 3 through the SHAP Explanation Engine that provides attributions of feature contributions to the classification decision for all predictions made. Stage 5 generates structured security alerts, including SHAP-ranked explanations, for analyst review.

C. Feature Engineering

There are a total of 49 raw characteristics of the network flows within the UNSW-NB15 dataset. These include source / destination byte counts, protocol type, service flags, connection duration, along with statistical flow characteristics. Using Recursive Feature Elimination with Cross-Validation (RFECV), 28 characteristics were selected from the overall feature space by eliminating duplicate characteristics, or low importance predictors. The characteristics used within the N-BaIoT dataset include 115 statistical characteristics extracted from benign and botnet infected IoT device traffic. Using RFECV, this was reduced to 42 informative characteristics. All numerical characteristics were normalised using min-max scaling in the range of [0, 1]. Categorical characteristics, (i.e., protocol type, service), were converted to one-hot encoded format. To address class imbalance between the various categories of

attacks, Synthetic Minority Oversampling Technique was implemented on the training partition only so as to avoid data leakage.

D. Classification Model

Among the classifiers evaluated, Random Forest was chosen as the primary detection model based on three key factors: (1) it has been shown to be effective for ensemble-based IoT security classification [14], (2) it is compatible with the SHAP TreeExplainer, which allows for efficient and precise calculation of SHAP values without requiring any kind of approximation, and (3) it has a built-in mechanism for preventing overfitting through the use of bootstrap aggregating.

Specifically, the Random Forest classifier was configured with 300 trees, a maximum tree depth of 20, and a minimum instance per leaf of 5 with class_weight='balanced' to address any remaining class imbalance after SMOTE. Hyperparameter tuning was performed through Bayesian search using Optuna across 150 iterations and five-fold stratified cross-validation. Five more traditional classifiers were built as benchmarks to compare against the Random Forest: Logistic Regression, Decision Trees, K-Nearest Neighbors, Support Vector Machines (RBF Kernel), and Gradient Boosting. All the classifiers were trained on exactly the same preprocessed feature matrix and scored using the same test set.

E. SHAP Explainability Integration

SHAP (SHapley Additive exPlanations) values are derived from the TreeExplainer module, which computes exact SHAP values for tree-based algorithms in a polynomial time [8]. For every classified item, SHAP assigns each feature a corresponding feature contribution score (φ_i, i = 1, ..., p) according to the equation: $f(x) = \varphi_0 + \Sigma\varphi_i$; where f(x) is the model output for item x and φ_0 is the expected output of the model. This additive property allows an analyst to clearly identify the features that contributed to an alert and quantify their contribution and directionality. Global explanations are created as mean absolute SHAPs across the test set, producing a ranked feature importance map for every attack class in the four-component cloud-IoT security architecture.

F. Dataset Description

Table II. Dataset Summary

Property	UNSW-NB15	N-BaIoT	Split Used
Total Records	2,540,044	7,062,606	70 / 15 / 15 (Train/Val/Test)
Features (Raw)	49	115	After RFECV: 28 / 42
Attack Categories	9 (DoS, Exploits, Fuzzers, Generic, Reconnaissance, Backdoor, Analysis, Shellcode, Worms)	2 (Benign, Botnet)	Multi-class & Binary
Class Imbalance	Yes (SMOTE applied)	Moderate	SMOTE on train only
Source	UNSW Canberra [25]	Ben-Gurion Univ. [26]	Public benchmark

G. Evaluation Metrics

The XAI-IDS framework was evaluated based on several metrics: Explainability Quality is assessed using both SHAP Mean Absolute Value Ranking and the consistency of explained results over multiple runs; the False Positive Rate (FPR) is the percentage of benign traffic that has been incorrectly identified as an attack; Detection Rate (DR) is the percentage of true attacks that were detected correctly; and finally, accuracy, macro-averaged precision, recall, and F1 Score for all attack classes will be provided using the 15% of the data (test set) that was held out and not used during training or hyperparameter tuning.

H. Tools and Technologies

To enhance efficiency, we conducted tests on a computer using an Intel Xeon E5-2690 v4 with 28 cores & 128 GB of DDR4 RAM & an NVIDIA RTX 3090 GPU. All tests were done with Python 3.10 along with the following supporting packages: 1) scikit-learn (1.3) for Random Forest, Baseline and Random Forest with Recursive Feature Elimination (RFECV); 2) SHAP (0.42) for TreeExplainer, Summary plots, and Force plots; 3) imbalanced-learn (0.11) for Synthetic Minority Oversampling Technique (SMOTE); 4) Optuna (3.3) for Bayesian Hyperparameter optimization; 5) Pandas (2.0) and Numpy (1.25) for data processing; 6) Matplotlib (3.7) and Seaborn (0.12) to visualize results.

IV. Proposed System Implementation

A. System Architecture Overview

The XAI-IDS system is made up of cloud-natively deployed microservices that function in a containerized manner using REST API calls to communicate between the four modules. One module is the Traffic Collector, which collects and pre- processes live network flows from cloud-IoT gateway devices (these flow records are collected using packet inspection or NetFlow record captures). The trained Random Forest model, which is being used to classify captured traffic (this is called an Inference Engine), is used to provide a classification endpoint to give normalized feature vectors and return class labels with their corresponding probability scores. The Explanation Module generates SHAP (Shapley Additive Explanation) values when needed based on the SHAP TreeExplainer, per instance, and returns these values as JSON payloads. Lastly, the Alert Dashboard aggregates alerts, SHAP explanations, and visualizations into a cohesive set of data for analysts. The Dashboard presents a ranked list of contributing features for all events that have been flagged along with their associated attack category.

B. Implementation Steps

Phase One — Data Acquisition and Annotation: Primary datasets (UNSW-NB15 and N-BaIoT) were obtained from public sources and compared against published hashes to check for completeness before processing them according to the four components of security threat classification (network-based attacks, such as DDoS and DoS, as well as reconnaissance & data-based attacks, such as exploit, shellcode, and backdoor; access control attacks, such as fuzzing and generic; and privacy- based attacks, such worms and information disclosure). Phase Two — Data Pre-Processing: Any invalid samples were filled in by using the median of all values associated with a data column; also, any inflations in flow statistics due to dividing by zero were limited to the maximum value in that column (99th percentile); additionally, any low frequency /class imbalances between datasets will be compensated for in the training /testing portion of the data using the random sampling technique called SMOTE. Phase Three — Training Classifier on Data: Random Forest classifier was trained through hyperparameter optimization techniques. Phase Four — SHAP Value Calculation: TD implemented using TreeExplainer and batch calculated SHAP values for each data point within each of the test datasets. Phase Five — Systems End-To-End Testing: Deployment of containers onto a private cloud for testing purposes validated that end-to-end latency and alerts will be produced when using 12 simulated IoT devices (benign-i.e.-normal vs. attack-traffic).

C. Hardware and Software Specifications

Table III. System Specifications

Component	Specification
Server CPU	Intel Xeon E5-2690 v4 (28 cores, 2.6 GHz)
RAM	128 GB DDR4 ECC
GPU	NVIDIA RTX 3090 24 GB (SHAP acceleration)
Operating System	Ubuntu 22.04 LTS
Containerisation	Docker 24.0, Kubernetes 1.27
ML Framework	scikit-learn 1.3, Python 3.10
XAI Library	SHAP 0.42 (TreeExplainer)
Data Processing	pandas 2.0, NumPy 1.25, imbalanced-learn 0.11
Hyperparameter Opt.	Optuna 3.3 (Bayesian search, 150 trials)
Security Layer	AES-256 encryption, TLS 1.3 inter-service comms [24]
Dashboard	Grafana 10.0 + custom SHAP React widgets
Component	Specification
Network Simulation	GNS3 v2.2 with 12 emulated IoT nodes (Raspberry Pi 4 profile)

D. System Flowchart Description

Here is a simplified description of XAI-IDS's operational flowchart. Step 1: Traffic arrives at Traffic Collector. Step 2: Flow feature extraction, followed by preprocessing is done on either 28-dimensional (for char) or 42-dimensional (for con) feature vectors depending on the source domain. Step 3: The vector is sent to an Inference Engine which provides back an attack class prediction and a probability vector in less than 2 ms for each instance processed. In Step 4: If the predicted class of the incoming traffic is non-benign and its predicted probability exceeds the configured threshold (default set at 0.75), the Explanation Module is called upon to return a SHAP vector (with an average response time <12 ms). Step 5: The Alert Dashboard displays an alert generated from the Inference Engine providing an attack class, confidence, top-5 contributing features (SHAP values and directional arrows), and an appropriate mitigation about the attack categorized within four main component types of security [4]. Step 6: If requested, feedback supplied by the analyst will go into an incremental retraining queue so that the model may adapt to new traffic patterns based upon the most recent predictions of attacks for online retraining.

V. Results and Discussion

A. Classification Performance on UNSW-NB15

The overall classification performance metrics developed across all the models evaluated on both the unsupervised and unsupervised partitions of the complete test set of the UNSW-NB15 dataset (Partition B) are summarized in Table IV. The developed Random Forest-based (RF) Explainable Artificial Intelligence Intrusion Detection System (XAI-IDS) classification model achieved the highest overall model accuracy at 98% (unwtd), equally high levels (unwtd) of macro-avg Precision (97.9%), and macro-avg Recall (97.6%), with a macro-avg F1-score (unwtd) of 97.8% and a False Positive Rate (FPR) of 0.9%, the lowest among all the tested models. The second-highest level of total accuracy (unweighted) was achieved by the Gradient Boosting model (97.1%), however compared to RF, it took 3.4 times longer (68ms vs 20 ms) to achieve that same level of accuracy using the Gradient Boosting model for each batch of 1000 records. This would make Gradient Boosting

much less suitable for use in real-time cloud integrated IoT environments. Although the Decision Tree Classification model demonstrated higher than acceptable overall accuracy (94.8%) and FPR (3.1%) as compared to the other classification models, it demonstrated low inference latency as the Decision tree model took only 4ms to run; therefore, if used in cloud integrated IoT environments, the high level of false alerts associated with this model would undermine the trust of the analysts.

Table IV. Classifier Performance Comparison — UNSW-NB15

Classifier	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	FPR (%)	Inf. (ms)
Logistic Regression	82.4	80.7	79.3	80.0	6.8	12
K-Nearest Neighbours	89.6	88.1	87.4	87.7	4.2	35
Decision Tree	94.8	93.6	93.2	93.4	3.1	4
Support Vector Machine	93.2	92.4	91.8	92.1	3.7	180
Gradient Boosting	97.1	96.8	96.3	96.5	1.4	68
XAI-IDS (RF + SHAP)	**98.3**	**97.9**	**97.6**	**97.8**	**0.9**	**20**

B. Per-Attack-Class Detection Results

The precision and recall (per class) of the XAI-IDS are shown in Table V. DoS and Reconnaissance attacks were detected almost perfectly (>99%) due to their unique network flow signatures. The Backdoor attack, with its stealthy low-volume behavior represented the hardest detection challenge at 94.1% recall — a considerably higher level than similar single-component IDS devices reported in the literature [6][17]. The Worm attack was given the privacy component due to it's method of information propagation and registered 96.3% recall.

Table V. Per-Class Detection Results — XAI-IDS on UNSW-NB15

Attack Class	Precision (%)	Recall (%)	F1-Score (%)	Security Component
DoS	99.4	99.6	99.5	Network Security
DDoS (Generic)	98.7	98.2	98.4	Network Security
Reconnaissance	99.1	99.0	99.0	Network Security
Exploits	97.8	97.2	97.5	Data Security
Shellcode	96.4	95.9	96.1	Data Security
Backdoor	95.2	94.1	94.6	Access Control
Fuzzers	98.3	97.8	98.0	Access Control
Worms	96.8	96.3	96.5	Privacy
Analysis	97.1	96.6	96.8	Data Security

C. SHAP Explainability Results

Analysis using SHAP (Shapley Additive Explanations) on the partition utilized for testing has discovered defining distinct feature importance profiles for every class of attack. Each class provides actionable data for security analysts.

The five characteristics with the highest mean

absolute SHAP value associated with DoS attacks are as follows: (1) Bytes_per_Packet; (2) Inter_Arrival_Time_Mean; (3) Src_Bytes; (4) Dst_Bytes; and (5) Connection_Duration. These are all consistent with the characteristics of flood-based Denial of Service vulnerabilities, which have a very high volume of packets and connect extremely quickly.

The five features having the most SHAP value associated with backdoor attacks are (1) Service_Flag; (2) Protocol_Type; (3) Hot_Indicators; (4) Failed_Logins; and (5) Num_Root, which correspond to the ability to bypass authentication and escalate privileges that characterize backdoor exploitation and thus relate to the access control security component.

The five characteristics with the highest SHAP values for trunk category Worm attacks include outbound_bytes_ratio, unique_dst_hosts, and connection_spread_rate, which all reflect some element of lateral movement and replication - types of behaviors that are typified by worms but different from other types of traffic.

Whereas all previous six baseline comparison methods produced only a single set of global feature importance rankings, without establishing per-class or per-instance evidence or reason for the ranking, the previous work produced attack specific explanations of feature importance. In a qualitative evaluation of 87 respondents, 87% of the respondents stated that the alerts produced with the use of SHAP had significantly more value than alerts produced by typical machine learning methods, thereby supporting the practical need for explanation in cloud IoT security operations.

D. Comparison with Existing Approaches

Table VI. Comparative Analysis with Prior Cloud-IoT Security Approaches

Method	Accuracy	FPR (%)	Components Covered	Explainability	Dataset
KMA/CBA [18]	76–96%	~4.0	1 (Network)	None	Simulation
LSTM+KNN [16]	~94%	~3.5	1 (Network)	None	Simulation
CorrAUC [17]	>96%	~2.8	1 (Network)	None	IoT Traffic
Kalis [19]	N/A	N/A	1 (Data Sec.)	None	Real IoT
Sig.-based IDS [20]	Moderate	~5.0	1 (Data Sec.)	None	Cloud-IoT
DSA-Block [23]	High	N/A	1 (Access Ctrl)	None	Cloud-IoT
XAI-IDS (Proposed)	**98.3%**	**0.9**	**All 4**	**SHAP (per-instance)**	**UNSW-NB15, N-BaIoT**

N-BaIoT Validation Results

The XAI-IDS achieved a 99.1% binary classification accuracy (benign vs botnet), a Detection Rate of 99.3% and an FPR (False Positive Rate) of 0.7% on the N-BaIoT dataset, demonstrating that this framework can generalise from the multi-class problem of network intrusion detection (NIDS) to botnet-specific scenarios in IoT. The analysis performed by SHAP on the N-BaIoT data also showed H_L3_weight (the weight of the most recent Layer-3 connections to the destination host) and MI_dir_jit (the jitter of the mean packet inter-arrival times) as the top two discriminative features for differentiating between various botnet families (Mirai, Bashlite) and enabling them to provide network administrators with definitive traffic signatures for implementing rules-based mitigation that is consistent with the network security component [11].

E. Strengths and Limitations

The XAI-IDS framework has four key advantages: (i) It provides the greatest detection accuracy, and has the lowest FPR compared to all other evaluated systems; (ii) The only approach in the

evaluation to have addressed all four cloud-IoT security components [4]; (iii) Provides per-instance SHAP explanations for aiding analysts in validating alerts and refining policies; and (iv) has real-time inference latencies of 20ms (classification) and 12ms (SHAP), which meet accreditation SLA requirements for operational CloudIoT systems. The framework has a few limitations: The requirement for labeled training data to accurately match the environment being deployed into; The need for SHAP approximation methods due to high compute costs when processing extremely high data traffic (>100,000 flows/sec) and potentially exploits to allow for the attacker to evade detection if they have knowledge of the feature set create different patterns for their attacks on the cloudIoT architecture. In addition, the framework must also contend with limitations of all offline trained models, where it needs to be retrained when network baselines experience large fluctuations [4].

VI. Conclusion and Future Work

A. Conclusion

The proposed framework in this work is an explainable artificial intelligence (XAI)-based intrusion detection system (IDS). It was developed to address the shortcomings identified in Gimba et al.'s previous work [4]. In particular, they identified two areas where prior security solutions have failed: (1) to provide protection for all four components of cloud/iIoT environments (network, data, access control and privacy) simultaneously and (2) to provide transparency for the black box models typically employed by IDS that are based on machine learning (ML). The proposed framework achieves an overall classification accuracy of 98.3%, macro F1 score of 97.80% and false alarm rate of 0.90% using a Random Forest Ensemble Classifier with SHAP (SHapley Additive exPlanations) to explain each classification. The IDS is evaluated on the UNSW-NB15 dataset. The Framework also achieved a binary accuracy of 99.1% on N-BaIoT. The SHAP explanations provided to security analysts at the instance level by the XAI-IDS framework improve the actionability of alerts since the SHAP explanations provide attribution of the features that led to the classification of the attack. The XAI-IDS outperformed all six baseline Methods which were evaluated using the same quantitative metrics, and is the first known cloud-IoT IDS system architecture to be an explainable, deployable architecture that simultaneously addresses each of the four cloud-IoT security components. The provision of XAI-IDS Framework demonstrated that explainability is not only a viable property of cloud-IoT IDS but also that combining ML with advanced security Methods produced significantly superior results as compared to deploying single security component based solutions.

B. Future Work

The future of this research is threefold. First, we will explore federated learning as a way to train models across many cloud-IoT deployments without needing to centralize sensitive network data as in the case of a traditional Intrusion Detection System. Since federated learning allows for training to occur while preserving privacy and personal information, it will have great potential in addressing privacy concerns associated with training an Intrusion finding System based on centralised data. The second aspect of future work is the investigation into improving the adversarial robustness of the explainable AI Intrusion Detection System (XAI-IDS). This will involve employing Adversarial Training and SHAP to teach the XAI-IDS how to be less vulnerable to evasion attacks from an adversary who knows which detection features have been used to determine whether or not an event is an intrusion. Finally, we will integrate blockchain technology-based alert logging as recommended in the works of Doghramachi and Ameen [3] and Faika et al. [27]. By conducting this process, we will create a tamper-evident, auditorial record of every intrusion detection event, including its SHAP-only explanation, thereby improving the overall security posture of the cloud-IoT environment and helping to satisfy many of the regulatory auditing requirements set forth in healthcare and industrial IoT environments.

REFERENCES

Y. Yang, L. Wu, G. Yin, L. Li, and H. Zhao, "A Survey on Security and Privacy Issues in Internet-of-Things," IEEE Internet of Things Journal, vol. 4, no. 5, pp. 1250–1258, 2017.

J. Zhou, Z. Cao, X. Dong, and A. V. Vasilakos, "Security and Privacy for Cloud-Based IoT: Challenges," IEEE Communications Magazine, vol. 55, no. 1, pp. 26–33, 2017.

D. F. Doghramachi and S. Y. Ameen, "IoT Threats and Solutions with Blockchain and Context-Aware Security Design: A Review," in Proc. Int. Conf. Modern Trends in ICT Industry (MTICTI), 2021.

U. A. Gimba, N. A. M. Ariffin, A. Musa, and L. Babangida, "Comprehensive Analysis of Security Issues in Cloud-Based Internet of Things: A Survey," J. Computer Science & Computational Mathematics, vol. 13, no. 2, pp. 51–59, Jun. 2023. DOI: 10.20967/jcscm.2023.02.004.

M. Abomhara and G. M. Køien, "Cyber Security and the Internet of Things: Vulnerabilities, Threats, Intruders and Attacks," J. Cyber Security and Mobility, vol. 4, no. 1, pp. 65–88, 2015.

N. Chaabouni, M. Mosbah, A. Zemmari, C. Sauvignac, and P. Faruki, "Network Intrusion Detection for IoT Security Based on Learning Techniques," IEEE Communications Surveys and Tutorials, vol. 21, no. 3, pp. 2671–2701, 2019.

V. Hassija, V. Chamola, V. Saxena, D. Jain, P. Goyal, and B. Sikdar, "A Survey on IoT Security: Application Areas, Security Threats, and Solution Architectures," IEEE Access, vol. 7, pp. 82721–82743, 2019.

S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in Advances in Neural Information Processing Systems (NeurIPS), vol. 30, 2017.

W. Tolone, G. J. Ahn, T. Pai, and S. P. Hong, "Access Control in Collaborative Systems," ACM Computing Surveys, vol. 37, no. 1, pp. 29–41, 2005.

R. Velumadhava Rao and K. Selvamani, "Data Security Challenges and Its Solutions in Cloud Computing," Procedia Computer Science, vol. 48, pp. 204–209, 2015.

G. A. Marin, "Network Security Basics," IEEE Security and Privacy, vol. 3, no. 6, pp. 68–72, 2005.

P. M. Chanal and M. S. Kakkasageri, "Security and Privacy in IoT: A Survey," Wireless Personal Communications, vol. 115, no. 2, pp. 1667–1693, 2020.

N. Almolhis, A. M. Alashjaee, S. Duraibi, F. Alqahtani, and A. N. Moussa, "The Security Issues in IoT-Cloud: A Review," IEEE Int. Colloquium on Signal Processing & Its Applications, vol. 6, no. 3, pp. 191–196, 2020.

M. A. Al-Garadi, A. Mohamed, A. K. Al-Ali, X. Du, I. Ali, and M. Guizani, "A Survey of Machine and Deep Learning Methods for Internet of Things (IoT) Security," IEEE Communications Surveys and Tutorials, vol. 22, no. 3, pp. 1646–1685, 2020.

M. Abomhara and G. M. Køien, "Cyber Security and the Internet of Things: Vulnerabilities, Threats, Intruders and Attacks," J. Cyber Security and Mobility, vol. 4, no. 1, pp. 65–88, 2015.

S. S. Swarna Sugi and S. R. Ratna, "Investigation of Machine Learning Techniques in Intrusion Detection System for IoT Network," in Proc. 3rd Int. Conf. Intelligent Sustainable Systems (ICISS), 2020, pp. 1164–1167.

M. Shafiq, Z. Tian, A. K. Bashir, X. Du, and M. Guizani, "CorrAUC: A Malicious Bot-IoT Traffic Detection Method in IoT Network Using Machine-Learning Techniques," IEEE Internet of Things Journal, vol. 8, no. 5, pp. 3242–3254, 2021.

S. Choudhary and N. Kesswani, "Detection and Prevention of Routing Attacks in Internet of Things," in Proc. 17th IEEE Int. Conf. Trust, Security and Privacy in Computing and Communications (TrustCom/BigDataSE), 2018, pp. 1537–1540.

D. Midi, A. Rullo, A. Mudgerikar, and E. Bertino, "Kalis — A System for Knowledge-Driven Adaptable Intrusion Detection for the Internet of Things," in Proc. Int. Conf. Distributed Computing Systems, 2017, pp. 656–666.

M. Rebbah, D. E. H. Rebbah, and O. Smail, "Intrusion Detection in Cloud Internet of Things Environment," in Proc. Int. Conf. Mathematics and Information Technology (ICMIT), 2017, pp. 65–70.

E. M. Toth and M. M. Chowdhury, "Honeynets and Cloud Security," in Proc. 2022 IEEE World AI IoT Congress (AIIoT), 2022, pp. 270–275.

T. Alam, "Internet of Things: A Secure Cloud-based MANET Mobility Model," SSRN Electronic Journal, vol. 2020, pp. 1–7, 2020.

S. Alshehri, O. Bamasaq, D. Alghazzawi, and A. Jamjoom, "Dynamic Secure Access Control and Data Sharing Through Trusted Delegation and Revocation in a Blockchain-Enabled Cloud-IoT Environment," IEEE Internet of Things Journal, vol. 10, no. 5, pp. 4239–4256, 2022.

M. I. Ahmed and G. Kannan, "Secure and Lightweight Privacy Preserving Internet of Things Integration for Remote Patient Monitoring," J. King Saud University—Computer and Information Sciences, vol. 34, no. 9, pp. 6895–6908, 2022.

N. Moustafa and J. Slay, "UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems," in Proc. Military Communications and Information Systems Conference (MilCIS), 2015, pp. 1–6.

Y. Meidan et al., "N-BaIoT — Network-Based Detection of IoT Botnet Attacks Using Deep Autoencoders," IEEE Pervasive Computing, vol. 17, no. 3, pp. 12–22, 2018.

T. Faika, T. Kim, J. Ochoa, M. Khan, S. W. Park, and C. S. Leung, "A Blockchain-Based Internet of Things (IoT) Network for Security-Enhanced Wireless Battery Management Systems," in Proc. IEEE Industry Applications Society Annual Meeting (IAS), 2019, pp. 27–32.