SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V13I9P108

Volume 13 Issue 9, 120-146, September 2026

Original Article

A Risk-Adaptive Attribute-Based Access Control Framework with Ternary Decision Logic for Context-Aware IoT Security

Rashmin Prajapati[1], Sweta S. Panchal[2], Neha Soni[3], Sandipkumar R. Panchal[4]

[1]*Department of Computer Engineering, Dr. Subhash University, Junagadh, Gujarat, India.*
[2,4]*Department of Electronics & Communication Engineering, Dr. Subhash University, Junagadh, Gujarat, India.*
[3]*Department of Computer Engineering, Sardar Vallabhbhai Patel Institute of Technology, Vasad, Gujarat, India.*

[1]*Corresponding Author : rashmin.1012@gmail.com*

Received: 16 June 2026 Revised: 08 September 2026 Accepted: 11 September 2026 Published: 29 September 2026

Abstract *- As the Internet of Things (IoT) grows rapidly, security concerns are rising, as these devices are diverse, the operating environments change constantly and data are exchanged continuously. This paper is an extension of our previous published machine-learning based Risk-Adaptive Attribute-Based Access Control (RAd-ABAC) system where contextual risk estimation and Permit–Review–Deny authorization were introduced. The present study builds upon this in an implementation-oriented architecture including a Contextual Attribute Collector, Risk Evaluation Engine, and Decision Management Engine. These attributes of the user (role, location, time, device) are converted to a normalized risk indicator and weighted through a mechanism of risk aggregation. The contextual risk score is then fed into two decision thresholds to ultimately determine access rules (Permit, Review, Deny). Three publicly available IoT security datasets—IoT-23, TON_IoT, and IoT Intrusion Detection dataset—were used to evaluate the framework. These datasets were used to map network, temporal, behavioral and device attributes to access-control attributes. The models were built and tested with a 70:30 train–test split. Logistic Regression and Random Forest models were trained and tested with a 70:30 training-testing split. The accuracy, precision, recall, F1 score, confusion matrix, receiver operating characteristic curve, area under the curve and false-positive and false-negative rates were measured. The Random Forest model outperformed Logistic Regression with precision and recall values closer to each other, with an accuracy of over 96%. The intermediate Review state allowed indefinite access requests to be reviewed again before granting or denying access, which helped to decrease false dichotomy decisions. The present study introduces a modular processing architecture, explicit contextual-attribute transformation, quantitative validation of the processing across three heterogeneous datasets, and expanded. These developments provide greater visibility, flexibility and utility of context-aware access control in diverse IoT environments.*

Keywords *- Attribute-Based Access Control (ABAC), Context-Aware Access Control, Internet of Things (IoT) Security, Machine Learning in Security, Risk-Adaptive Access Control (RAdAC), Ternary Decision Logic.*

1. Introduction

1.1. Background

The Internet of Things (IoT) has been rapidly developed and has considerably changed the contemporary digital ecosystems, allowing homogenous links between heterogeneous devices, sensors, and services. IoT applications are becoming more and more part of our daily lives, whether it be smart homes and healthcare systems, or industry automation and smart cities. This growth has resulted in the creation and transfer of huge amounts of sensitive information in distributed settings, which have caused serious issues concerning security, privacy, and integrity of data [1]. Since IoT systems have to work under very dynamic and resource-constrained conditions, the traditional security mechanisms do not always offer sufficient protection against changing cyber threats.

A major security issue in IoT is the ability to make sure that only the authorized parties can access particular resources provided the conditions are adequate. Access control mechanisms are important in controlling user/ device/service interaction. Attribute-Based Access Control (ABAC) is one of the most popular flexible and fine-grained access control approaches that enable making access decisions depending on attributes like user role, location, time, and device properties [2]. In comparison to the traditional Role-Based Access Control (RBAC), ABAC is more scalable and flexible and can be used in complex IoT environments. Nonetheless, the

context-rich and dynamic nature of the IoT systems presents other complexities that cannot be comprehensively dealt with using the fixed access control policy. The environmental conditions, the patterns of behaviour, and the real-time levels of risks should be taken into consideration to make smarter and safer decisions when it comes to access. As a result, the necessity to develop the more sophisticated access control systems that combines context-awareness and the ability to make decisions in real-time grows [3]. Although ABAC provides fine-grained authorization, existing IoT access control mechanisms mainly depend on predefined policies and binary authorization outcomes. Such approaches cannot effectively handle uncertain access situations caused by changing device behaviour, user activity, and environmental conditions. Therefore, an adaptive framework capable of estimating contextual risk and providing intermediate authorization decisions is required.

1.2. Problem Statement

Although ABAC has its benefits, the current implementations have a number of limitations when used in the IoT. Most of the traditional ABAC systems are founded on the fixed and inert policy guidelines, which cannot be applied in addressing the dynamism and uncertainty of the IoT ecosystems. These fixed policies do not react to changing circumstances, such as a change in the user activity, network or threat conditions, where they lead to overly restrictive or overly liberal access control determinations [4]. The other significant restriction that should be seen as the critical one is the fact that, in traditional access control decisions, the answer is often either an allow or a deny. These rigid decision making frameworks are unable to capture the uncertain cases or marginal cases where the extent of risk is not as obviously safe as malicious.

This can lead to legitimate users being refused entry in unclear cases whereas potential threats may circumvent security measures because of inadequate assessment on contextual factors [5]. Moreover, most of the existing access control models do not have a high level of context-awareness. Even though the use of simple contextual characteristics is factored in some approaches, they usually lack the dynamic quantification and combination of the features into a single decision-making model. This is a drawback, which limits the efficacy of access control systems to support real-time Internet of Things applications, where context is crucial to deciding the validity of access requests [6].

1.3. Research Gap

Based on the research of the literature, it can be observed that traditional IoT access-control solutions still use a large extent of pre-defined policies and binary Permit-Deny decisions. The mechanisms, however, are not very powerful when it comes to managing uncertain or moderate-risk requests that come from changes in user behaviour, device, location, time and network conditions [7]. There are several existing approaches that use fixed risk parameters, poorly explained attribute-weighting mechanisms, or restricted contextual adaptation [8]; this is a limitation that has been addressed by risk-adaptive access control.

In our previous publication, we presented a machine learning-based Risk-Adaptive Attribute-Based Access Control (RAd-ABAC) approach that integrates contextual attributes, dynamic risk evaluation and Permit–Review–Deny authorization [7]. That study proved the feasibility of the RAd-ABAC concept with two datasets: IoT-23 and TON_IoT, however, there were still some unanswered questions. The previous framework, in particular, lacked proper functional separation between the acquisition of contextual attributes, calculation of risk and authorization management. Furthermore, the transformation process from features to attributes and the process used to derive the risk-weights needed the detailed and independently verifiable investigation.

The overall effectiveness of the integrated framework was presented in the earlier evaluation, so the contribution of the main components of this framework was not clearly established. Additionally, a systematic way of converting network-level and behaviour attributes acquired from heterogeneous IoT security data to user-, location-, time- and device-related access-control attributes will need to be worked out. The lack of specific datasets for the evaluation of risk-adaptable IoT authorization makes the framework hard to reproduce [9].

Thus, the aim of the current study is to resolve these issues by developing an architecture that is very suitable for implementation, which consists of three engines: Contextual Attribute Collector, Risk Evaluation Engine, and Decision Management Engine. It offers an explicit dataset-to-attribute transformation procedure, there is a clearly defined risk-weighting methodology, and there are extended validations with the datasets IoT-23, TON_IoT, and IoT Intrusion Detection. To measure the improvements that can be measured in the framework of the proposed methodological refinement, we use dataset-wise performance analysis, class-wise evaluation, confusion matrices, AUC/ROC analysis, false-positive rate, and false-negative rate, and component-ablation experiments.

1.4. Objectives

The present study aims at laying out the objectives on the basis of the previously published RAd-ABAC framework:

- To build up an implementation-oriented modular architecture to separate the acquisition of contextual attributes, risk evaluation and authorization decision management into distinct functional components.
- To create an open and traceable workflow to convert network, temporal, behavioral, and device features of

heterogeneous IoT security data into user, location, time, and device access-control features.
- To design and codify a quantitative risk-assessment procedure that clearly determines the normalization of attributes, the mapping of features to weights, the calculation of the risk-weight(s), the computation of the risk-score, and the choice of decision thresholds.
- To extend the experimental validation of the previous framework by adding datasets IoT-23, TON_IoT, and IoT Intrusion Detection and studying the generalizability of the framework to different IoT environment.
- To assess Logistic Regression and Random Forest models using dataset-wise accuracy, class-wise precision, recall, F1-score, confusion matrices, AUC/ROC and false positive and false negative.
- To perform ablation analysis to find individual contribution of Contextual attribute transformation, weighted risk evaluation and ternary decision processing in overall performance of the framework.
- To compare the proposed refinements of the methodology with the previously published one and conventional binary access-control models in order to ascertain if the methodological improvements can be measured in terms of the accuracy of decision making and uncertainty management.
- To better reproduce the proposed approach by documenting the preprocessing procedures, training testing split, model configurations, experimental weights, decision thresholds and result generation process.

1.5. Contributions

The current work is based on our earlier published risk-adaptive Attribute-Based Access Control framework [7] using machine learning. So, as such, the underlying RAd-ABAC model, the contextual risk assessment, the machine-learning based decision support and the permit-review-deny authorization mechanism are not considered as in any way new contributions but rather as being inherited. The unique contributions of the present study are as follows:

- An implementation oriented modular architecture where the access-control process is partitioned into a Contextual Attribute Collector, Risk Evaluation Engine, and Decision Management Engine with well-defined inputs, process and outputs.
- A systematic approach to converting data to attributes for mapping network, temporal, behavioural and device information to four access-control dimensions: user role, location, time and device trust.
- A clear and quantitative risk-processing process that traces out the normalization of contextual attributes, the mapping of features to weights, the derivation of attributes to weights, the aggregation of risk-scores, and the use of decision rules.
- An extended experimental analysis, which includes three heterogeneous IoT security datasets (IoT-23, TON_IoT, and IoT Intrusion Detection dataset) while the previous study used only two datasets.
- An exhaustive Evaluation of the model performance in terms of accuracy, precision, recall, F1 score, confusion matrices, AUC/ROC and false positive and false negative rate with a detailed analysis of the evaluation of the model using a dataset-wise and class-wise basis.
- A component-ablation analysis to explore effects of individual aspects of contextual attribute transformation, weighted risk evaluation, and ternary decision processing on the system's performance.
- Clear comparison with previous published framework to establish if the proposed architectural, methodological and experimental enhancements offer quantifiable benefits in terms of greater adaptability, interpretability, generalization and decision quality.
- Increased experimental reproducibility by providing details about preprocessing operations, training and testing sets, model parameters, actual attribute weights, decision thresholds, and procedures for generating results.

1.6. Novelty of the Proposed Work

In the present study, we are directly related to our previous published paper “Machine Learning Based Risk-Adaptive Attribute-Based Access Control Framework for Secure IoT Networks” [7]. The previous study is a basis for a new framework named RAd-ABAC that combines Attribute-Based Access Control, machine-learning-based risk estimation, contextual attributes and a ternary Permit–Review–Deny authorization process. It was tested on 45,000 samples from IoT-23 and TON_IoT datasets and the accuracy, macro-average AUC and false-positive rate were reported as 94.6, 0.97 and 4.2, respectively. Thus, the following fundamental concepts are recognised and credited to the earlier publication and not claimed as new in the current manuscript.

- The present research builds upon the previous work in the architectural, methodological and experimental dimensions. The access-control workflow is redefined at the architectural layer, separating it into three distinct modules that have clear functionality: the Contextual Attribute Collector, the Risk Evaluation Engine, and the Decision Management Engine. This modular structure explains the way contextual data are gathered and treated, risk scores are calculated, and final decision on authorisation is made.
- On the methodological side, the present research offers a clear procedure for mapping the different properties of a heterogeneous network, temporal, behavioural and device-based data, into the four access-control dimensions of the context, namely user role, location, time and device trust. It also clarifies the normalization of these attributes, relationships between these attributes and the weights, how the attribute weights are derived and

how they are applied, how the aggregated risk score is calculated, and how two thresholds are used to make Permit/Review/Deny decisions. This clear processing pipeline tackles the lack of detail on the methodology used in the previous publication [7].

- From the experimental perspective, the evaluation is expanded from the 2 datasets used in the previous study to 3 heterogeneous datasets: IoT-23, TON_IoT and IoT Intrusion Detection dataset. A 70:30 train-test split is used for training and testing Logistic Regression and Random Forest models. Newly covered in the expanded evaluation are dataset-wise and class-wise accuracy, precision, recall, F1 score, confusion matrices, AUC/ROC and false positive and false negative rate. The accuracy of the Random Forest model was above 96% which is higher than the accuracy reported in the previous study [7] which was 94.6%. Furthermore, the individual contributions of contextual attribute transformation, weighted risk evaluation and ternary decision processing are investigated with the component-ablation analysis.

The independent contribution of the present work is thus neither the introduction of novel RAd-ABAC, novel machine-learning based risk prediction, novel contextual risk scoring nor novel ternary authorization for the first time. Rather, it is defined as a more distinctive and replicable implementation architecture, a verifiable data-to-attribute transformation and risk-weighting process, a greater level of validation across three heterogeneous IoT datasets, and a component-level evaluation of the framework. A detailed comparative discussion with earlier publication with the present study is included in Table 1.

Table 1. Component-wise comparison of the previous and present RAd-ABAC frameworks

Comparison aspect	Previous publication [7]	Present study	Specific extension in the present study
Research purpose	Introduced the foundational ML-based RAd-ABAC framework for IoT networks	Develops and validates an implementation-oriented extension of the earlier framework	Focus shifts from initial framework development to methodological transparency, modular implementation, and expanded validation
System architecture	Integrated contextual attributes, risk scoring, and authorization within the overall RAd-ABAC framework	Separates processing into the Contextual Attribute Collector, Risk Evaluation Engine, and Decision Management Engine	Provides explicit functional separation, processing sequence, inputs, and outputs
Contextual attributes	Used subject, object, and environmental attributes for risk-aware authorization	Organizes the context into user role, location, time, and device trust dimensions	Provides a more explicit contextual representation for heterogeneous IoT data
Attribute transformation	Converted benchmark data into ABAC-compatible feature vectors	Provides dataset-specific mapping of network, temporal, behavioural, and device-related features	Introduces a documented dataset-to-attribute transformation procedure
Risk evaluation	Applied machine-learning-based dynamic risk scoring	Applies normalized contextual attributes and a documented weighted aggregation procedure	Clarifies normalization, feature-to-weight mapping, final weights, and risk-score computation
Authorization mechanism	Used Permit–Review–Deny ternary authorization	Retains Permit–Review–Deny within a separate Decision Management Engine	Clarifies threshold application and the operational treatment of uncertain requests
Datasets	IoT-23 and TON_IoT	IoT-23, TON_IoT, and IoT Intrusion Detection	Extends evaluation from two datasets to three heterogeneous datasets
Dataset size	45,000 samples	Report the exact number of	Provides dataset-wise sample

		samples retained from each dataset	distribution and preprocessing information
Data partitioning	Earlier experimental configuration	Uses a 70:30 training–testing split	Explicitly documents the experimental partitioning procedure
Machine-learning evaluation	Reported overall framework performance	Evaluates Logistic Regression and Random Forest separately across the three datasets	Provides model-wise and dataset-wise evaluation
Reported accuracy	94.6%	Random Forest accuracy exceeded 96%	Demonstrates improved classification performance in the expanded evaluation
Other earlier results	Macro-average AUC of 0.97 and false-positive rate of 4.2%	Reports precision, recall, F1-score, AUC/ROC, FPR, FNR, and confusion matrices	Expands the number and granularity of evaluation measures
Component evaluation	Primarily evaluated the integrated framework	Examines contextual transformation, weighted risk evaluation, and ternary processing separately	Adds component-ablation analysis to identify the contribution of each module
Reproducibility	Provided the initial experimental description	Reports preprocessing rules, mappings, weights, thresholds, configurations, and result-generation procedures	Improves independent verification and reproducibility
Principal contribution	Established the feasibility of ML-based RAd-ABAC with ternary authorization	Provides a modular, transparent, and reproducibly evaluated extension	Establishes the incremental contribution beyond the previous publication

2. Literature Review

2.1. Attribute-Based Access Control in IoT

Considering the flexibility and the ability to mandate policies at a fine level of granularity, Attribute-Based Access Control (ABAC) has proved to be a valid access-control paradigm in IoT environments. ABAC differs from Role Based Access Control (RBAC) because it considers combinations of attributes of a subject, an object, a resource, the requesting device, and the environment to determine access to the resource. The multi-dimensional decision making is of special interest for IoT systems where heterogeneous devices and users interact dynamically in distributed networks [1].

ABAC has been used in previous studies to enhance the security and scalability of IoT systems. Data-centric IoT networking and ABAC protocols have shown promise of enhancing the management of access requests and resource sharing from a distributed network [2].

Use of ABAC for authentication and authorization has also been explored in smart home systems to ensure the security of user-to-device, device-to-service, and user-to-service communication [3]. Scattered in these studies are proof of the ability to handle complex and heterogeneous IoT authorization requirements, proving the suitability of ABAC for this.

Recently, blockchain technology has been combined with ABAC for solving decentralization, transparency, policy integrity, and trust-management issues. Blockchain-based ABAC frameworks use smart contracts to facilitate decentralized and tamperproof access-control policy enforcement [4]. There are also proposals on how to enhance the scalability and performance of large deployments of IoT devices, such as hierarchical ABAC models, blockchain architectures and sharding mechanisms.

In spite of these strides, numerous ABAC frameworks remain to rely on policies and attribute rules that are set at system design time. These policies might not be able to respond to what the user is doing, what trust they have, the state and environment of the network, and the severity of the threat. Integrating blockchain into a policy can improve the integrity of the policy but can also involve certain computational/communication overhead, particularly in resource-constrained environments like IoT. The constraints highlight the need for additional mechanisms for contextual risk assessment and adaptive decision making, in addition to traditional ABAC.

2.2. Risk-Adaptive Access Control

Risk-Adaptive Access Control (RAdAC) is a form of access control that can be applied to existing access control systems and adds real-time risk assessment to the authorization process. A RAdAC mechanism does not only

base its judgement on static rules but instead calculates the access risk based on the potential risk of the requester, the device, the resource, the behaviour and the operating environment. This will allow the system to meet operational needs while also taking security into account and will be especially helpful in dynamically changing IoT environments.

We presented a machine-learning based RAd-ABAC framework in our previous publication that fused ABAC-compatible contextual attributes, dynamic risk score and Permit–Review–Deny authorization [7]. The framework was tested with 45,000 samples obtained from the IoT-23 and TON_IoT datasets. It reached an accuracy of 94.6, macro-average AUC of 0.97 and FPR of 4.2%. This study showed that context-driven access control combined with machine learning-based risk assessment can enhance flexibility and reliability of IoT access control decisions.

However, some methodological and experimental problems were left unaddressed in the previous study [7] and need to be addressed. There was not an explicit enough component-level architecture for the functional relationship between attribute acquisition, risk calculation and authorization management was presented. In addition, the transformation of data into attributes and relationships of features to weights needed to be documented in greater detail. Additionally, the evaluation was limited to two data sets and primarily discusses the overall performance of the integrated framework. Therefore, the model's generalisability for the dataset and the contribution of the major components of the model were not fully validated.

The need for dynamic and context-sensitive risk-evaluation mechanisms in modern IoT systems is also highlighted in systematic investigations of risk-based access control [8]. The adaptive authorization models have shown how introducing real-time risk indicators can enhance the flexibility and security of access-control decisions [9]. Trust-based mechanisms build upon this by analyzing the trustworthiness of users and devices based on their past interactions and behaviour [10]. Fuzzy logic access control models also facilitate gradual decision making at times of uncertainty, as well as binary access control models [11].

While these strategies reinforce adaptive authorization, some models remain to have weights assigned manually or fixed thresholds, or fail to have a well-documented process for risk aggregation. It is therefore not enough to integrate the two approaches of ABAC and RAdAC, as it was done in our previous study [7] — what is needed is to be methodologically transparent, reproducible and independently verifiable.

2.3. Context-Aware Security

For modern IoT access-control systems, context-aware security is a critical need since the legitimacy of an access request may vary depending on the environment. Contextual factors are relevant time, location, user behaviour, device type & trust, network characteristics, resource sensitivity, and environment. These factors can be incorporated to enable an authorization mechanism to make informed, adaptive decisions [12].

Contextual information has been shown to be helpful in IoT security in previous studies. In the field of industrial IoT, context-aware mechanisms have been used in real-time to detect unauthorized and non-compliant activities, as well as to enhance system resilience [13]. The fog computing-based access control mechanism has also been introduced as a framework for decentralized and context-aware access-control decision-making at the edge of the network [14]. This is especially helpful for smart cities, healthcare systems, industrial networks and critical infrastructures that lack the ability to quickly react to new threats when using fixed policies [15].

But often information from the context remains only as input to policy. Numerous studies currently in the literature lack explanations of how the qualitative, contextual information is transformed into comparable numerical risk indicators. Also, there is often a lack of documentation on: how to deal with the missing attributes, how to assign categorical values to the risk, how to normalize numerical values for attributes, and how to combine several contextual dimensions.

The previous RAd-ABAC system [7] introduced contextual information into the risk-aware authorization, but it was not exhaustive in mapping a network set of features into the user, location, time and device attributes. Thus, there is still a need for a formal and repeatable contextual transformation procedure to decide on access-control decisions based on IoT security data.

2.4. Machine Learning and Intelligent Access Control

This is because machine-learning models are capable of learning complex behavioural patterns, detecting anomalies, predicting risk and authorizing adaptively. In the context of IoT security, access behaviour analysis has been used to enhance security decisions through the use of Decision Trees, Random Forests, Logistic Regression, and deep-learning algorithms [16].

The use of reinforcement-learning and deep-learning frameworks has also improved access-control systems' ability to react to an evolving environment. These methods can learn and update authorization policies by interacting with an environment and taking feedback into consideration when making future decisions [17]. Another AI-driven model for evaluating trust has been suggested to evaluate user and device reliability, which aids in making more context-driven access decisions [18].

The feasibility of supervised machine learning to develop dynamic risk scores in an RAd-ABAC system was shown in the previous study [7]. The results of the proposed system, however, were reported primarily on its overall performance. The performance improvement should be verified by adding more evidence on a model-wise basis, class-wise basis, and dataset-wise basis.

There are a number of restrictions to machine-learning-based access control. However, supervised learning needs ample and representative labelled data sets, and such data sets may not be readily available due to privacy, confidentiality, and security concerns for access-control research. Security auditing and policy validation is also hindered by many machine-learning models that are also black boxes, as they don't provide transparency [19]. Additionally, many ML-based methods still consider authorization as a binary classification problem, potentially resulting in ambiguous or medium-risk requests being classified as permit or deny.

The previous framework [7] solved this binary-decision problem by introducing a Permit/Review/Deny process, but more analysis is needed to understand the impact of the Review state on the reduction of false-positives and false-negatives. To differentiate the benefit of ternary processing from contextualization and/or machine-learning based classification, the results should be presented on a component basis and in detail by class.

2.5. Critical Analysis of Existing Studies

The literature has been reviewed and significant advances have been made in the field of access-control mechanisms in IoT environments. However, each of the types of approach only covers a portion of the larger authorization issue. The traditional ABAC model authorizes granularly by considering several attributes of the subject, object, device and environment. Many ABAC implementations are based on policies set when designing the system and are unable to cope with evolution of user behaviour, device trustworthiness, environment and threat levels. While ABAC based on blockchain technology could enhance the decentralization and policy integrity, it might also be computationally, storage and communication heavy for resource-constrained IoT networks.

Some of these issues are addressed by the RAdAC mechanisms which include risk evaluation in authorization decisions. Risk-based models already exist and are able to show that the decisions for security can be improved by contextual information, e.g. user behaviour, device reliability, location, time and environmental conditions. There are still several approaches, however, that rely on prespecified risk weights and thresholds with no full empirical basis for their choice. However, other studies are not clear about how the risk weights are applied, whether they are manually set, statistically calculated, optimized or learned from training data.

The context-aware access control models are based on the idea of extending the scope of authorization to take into account the context of an access request. However, most of the studies consider contextual information as categorical information for policies and fail to give any systematic approach to convert heterogeneous contextual features into normalized values of risk. So, feature transformation, missing value treatment, risk scale construction, and aggregation of attributes are still critical methodological issues. A With machine learning, access control systems have been able to detect intricate patterns and automate the risk estimation process. Many models based on machine learning, however, require large quantities of labelled data, are non-interpretable or use a binary decision about authorisation. These constraints limit their applicability in uncertain and security sensitive IoT environments.

In a previous publication [7], we discussed some of these gaps and created a unified framework that incorporates ABAC, supervised machine-learning risk scoring, contextual information, and Permit–Review–Deny authorization. Thus, the combination of these elements cannot be seen as the totally new contribution of the present study. The previous framework did not however, completely answer the following queries:

Clear separation of the collection of the contextual attributes, risk evaluation and decision management;

- Full mapping of original-attribute-to-access-control mapping;
- Clear and consistent derivation and reporting of the risk weights for the experiments; and
- Validation of the datasets per data set, on more than two heterogeneous iot datasets; and
- Class level reporting of precision, recall, f1 score, AUC, false positives, and false negatives; and
- component-level ablation showing contribution of each process step; and
- Availability of adequate configurations, code, mappings and logs for independent reproduction.

In this work, we aim to solve the aforementioned challenges with a modular structure, an explicit dataset-to-attribute transformation procedure, a documented risk-weighting methodology, three different and heterogeneous IoT datasets, and detailed quantitative and ablation analysis. In this context the present work is seen as a methodological and experimental development of the already published RAd-ABAC framework [7].

As illustrated in Table 2, existing solutions cover key IoT authorization issues, such as, fine-grained attribute evaluation, decentralization, contextual awareness, trust assessment, and dynamic risk estimation. But, there are constraints that exist in the case of transparent attribute transformation, risk-weight

derivation, managing uncertainty, dataset-wise validation, and experimental reproducibility. The present study is a further extension of the previously published framework [7] on the integration of RAd-ABAC with ternary authorization, explicitly implemented in a modular architecture, methodologically mapped and expanded in the experimental evaluation.

Table 2. Critical comparison of existing IoT Access-control approaches

Study	Method/approach	Principal advantage	Identified limitation
[2]	ABAC for IoT	Provides fine-grained authorization using multiple subject, resource, device, and environmental attributes	Primarily depends on predefined policies and does not provide adequate dynamic risk adaptation
[4]	Blockchain-based ABAC	Supports decentralized policy enforcement, transparency, integrity, and improved trust	Introduces computational, storage, communication, and scalability overhead
[8]	Risk-based access control	Incorporates contextual risk evaluation into access decisions	May depend on predetermined risk factors, fixed weights, or fixed thresholds
[9]	Adaptive authorization for IoT	Adjusts authorization decisions according to changing risk conditions	Provides limited explanation of attribute quantification and weight derivation
[10]	Trust-based access control	Evaluates users and devices through trust values and behavioural history	Trust calculation may depend on application-specific assumptions and historical data
[11]	Fuzzy risk-based access control	Supports gradual decision-making under uncertainty	Membership functions and rules may require manual configuration
[13]	Context-aware industrial IoT security	Uses contextual information to identify unauthorized or non-compliant activities	Does not provide a unified mechanism for converting all contextual attributes into comparable risk values
[14]	Fog-based adaptive access control	Supports decentralized and context-aware decisions near the network edge	Introduces additional deployment, coordination, and resource-management complexity
[16]	Machine-learning-based access control	Identifies complex patterns and supports automated security decisions	Requires representative labelled data and may provide limited interpretability
[7]	Previously published ML-based RAd-ABAC framework	Integrates contextual attributes, supervised risk estimation, and Permit–Review–Deny authorization	Provides limited component-level mapping, dataset-to-attribute details, weight derivation information, dataset-wise results, and ablation evidence
Proposed extension	Modular and reproducible RAd-ABAC framework	Provides explicit attribute transformation, separated functional modules, expanded three-dataset validation, and detailed performance analysis	Requires future real-time validation in operational IoT environments

As per the above analysis, it is clear that the contributions made by existing approaches have been significant when it comes to fine-grained authorization, decentralized policy enforcement, trust evaluation, contextual awareness and adaptive risk assessment in IoT security. However, there are some methodological and experimental caveats to note.

The existing ABAC models typically rely on predefined policies and provide little real-time risk adaptation; meanwhile, there are several risk-adaptive and context-aware approaches that do not provide clear procedures for quantifying contextual attributes, deriving risk weights, and setting thresholds. Many machine-learning-based methods also make a binary Permit–Deny decision, which can be insufficient to manage some access requests that are uncertain or in the middle range of risks.

In our previous work [7] we tackled some of these issues by combining these features in a single framework, called RAd-ABAC, which incorporates ABAC, contextual risk evaluation, machine-learning-based risk prediction and Permit–Review–Deny authorization. Hence, the integration of these aspects is not asserted as a new research gap in this work. The first model, however, was not very clear about component-level separation of architecture, mapping of datasets to attributes, details of how to derive risk weights, assessment of datasets, and evidence of ablation. Therefore, the current research aims to make the previously proposed

framework more methodologically transparent, reproducible and experimentally validated. It presents an explicitly modular processing architecture, presents the mapping of heterogeneous IoT data to access control attributes, details the risk scoring process, and expands the analysis over three different IoT security datasets. These enhancements are aimed at providing a technically sound extension of the previous work, making the framework's applicability to heterogeneous IoT environments more feasible.

2.6. Research Gap Identification and Motivation

Through critical analysis of the existing access-control approaches, it is found that the IoT environments are significantly secured with attribute-based authorization, policy enforcement supported by blockchain, access control based on risk awareness, contextual modelling, trust evaluation, and machine-learning approaches.

However, there are still several methodological and experimental issues to be addressed, especially in heterogeneous IoT settings in which authorization decisions should be made based on the dynamic nature of user, device, network, temporal, and environmental conditions.

Traditional approaches to Access Control based on attributes offer flexible and fine-grained access control, but typically rely on static attribute evaluation and pre-defined policies. These methods lack ability to adapt authorization decisions to evolving user behaviours, device trust, network conditions and threat levels. This limitation is overcome with Risk-Adaptive Access Control, a solution that adds risk assessment into the authorization process. But many Risk-Based Approaches are based on manually assigned parameters, fixed weights or thresholds, where the derivation and the empirical justification are not sufficiently explained.

Context-aware access-control models are able to include contextual information like location, time, device properties, and environmental conditions. While these attributes are useful in making contextual decisions on authorizations, there are few studies that offer a systematic approach for mapping these contextual attributes to individualized and standardized risk scores. These approaches are not easily reproducible or practically applicable because there is no transparent feature transformation, attribute mapping, handling of missing values and risk aggregation.

Access-control systems based on machine learning have shown to be better in classification and anomaly detection. However, there are many methods that use a binary classification problem (Permit – Deny) that might not be sufficient to address requests with uncertain or intermediate risk. Machine-learning models can also be difficult to interpret and require labelled data, besides not reporting on preprocessing, model configuration, decision thresholds and results obtained per dataset.

In our previous publication [7], we tried to tackle some of these issues by introducing a Machine-Learning based RAd-ABAC framework, which was a combination of contextual attributes, dynamic risk estimation and a Permit–Review–Deny authorization mechanism. It was tested on the IoT-23 and TON_IoT datasets and proved the feasibility of ternary risk-adaptive authorization. In this context, the combination of ABAC, machine-learning based risk assessment, and ternary decision logic is not considered as an independent contribution but as a building block in this present study. Although it helped, the previous approach [7] had some problems needing further study. The key research gaps that inspire the current study then can be summarized as follows:

The initial architecture had some ambiguity in specifying the acquisition of the attributes of the context, the calculation of risk, and handling of authorization, as distinct functional entities. No full mapping of dataset to attributes: No feature-level explanation of how the IoT-23 and TON_IoT network features became access control attributes for user, location, time and device was given. Non-documented risk-weight derivation: Little mathematical and experimental transparency in the procedures used to derive risk weights, such as whether weights are fixed, optimized empirically or learned. Limited data set-wise generalisation evidence: The previous assessment was limited to two IoT security datasets and was not able to determine if the performance is consistent across a larger and more diverse set of IoT data.

Below-average performance evidence for components: The previous assessment mainly focused on the overall performance of the integrated framework, without separating the effects of the contextual attribute transformation, weighted risk evaluation, and ternary decision processing. There are missing quantitative information and reproducibility information: Missing information includes: complete confusion matrices, class-wise precision, recall and F1 score, AUC/ROC, false-positive and false-negative rates, preprocessing procedures, model configurations, experimental weights, thresholds, code, and execution logs.

In order to overcome these shortcomings, the present study extends the previous RAd-ABAC approach to an explicitly modular and implementation-oriented architecture based on a Contextual Attribute Collector, Risk Evaluation Engine and Decision Management Engine. It explains the transformation of dataset to attributes and the risk-weighting process, quantitative analysis of the datasets in model-wise, class-wise, and dataset-wise manner and extends the evaluation to IoT-23, TON_IoT, and IoT Intrusion Detection datasets. Further experiments for component-ablation are then undertaken to distinguish the individual contributions of the various major processing stages. The scope of the present work should therefore not be interpreted as an attempt to re-introduce either RAd-ABAC or ternary authorization, but rather as offering a clearer, more reproducible and technically

verifiable extension to the previously published framework [7].

3. Proposed Framework

3.1. System Architecture

The present study builds upon the previously published RAd-ABAC framework [7] and extends it to make it more explicit and implementation oriented. The previous system defined integration of ABAC-compatible attributes, machine-learning support of risk estimation and Permit–Review–Deny authorization. Hence, these basic functions remain in the current structure and are not considered as new concepts. The extension to the architecture is the separation of these activities into three modules, each of which defines its own contextual attributes, evaluates the risk, and then makes decisions. The extension to the architecture is the separation of these activities into three separate modules: the Contextual Attribute Collector; the Risk Evaluation Engine; and the Decision Management Engine.

Figure 1 illustrates the complete processing workflow of an access request, from its initial submission to the generation of the final authorization response. The request may originate from a user, device, application, or IoT service and contains information about the requester, requested resource and action, device characteristics, network properties, and available contextual conditions. The request is processed sequentially through three explicitly defined functional modules: the Contextual Attribute Collector, Risk Evaluation Engine, and Decision Management Engine.

First, the Contextual Attribute Collector extracts user role, location, time, and, and device-trust information from the access request. It performs feature mapping, categorical encoding, and normalization to transform the collected information into values within the range $[0, 1]$. The module produces the contextual attribute vector $X = [x_u, x_l, x_t, x_d]$, which is forwarded to the Risk Evaluation Engine. The Risk Evaluation Engine applies the corresponding attribute weights w_u, w_l, w_t, and $w_d$to calculate the aggregated contextual risk score:

$$R = w_u x_u + w_l x_l + w_t x_t + w_d x_d$$

The quantified attributes and resulting risk score are subsequently processed using the selected machine-learning model to determine the request's risk classification. The calculated risk score $R$is then transmitted to the Decision Management Engine.

The Decision Management Engine compares $R$with the two decision thresholds $T_1$and T_2. A request is classified as Permit when $R < T_1$, Review when $T_1 \leq R < T_2$, and Deny when $R \geq T_2$. Requests assigned to the Review category undergo additional verification through multifactor authentication, manual review, or supplementary device validation before receiving a final Permit or Deny decision. The final system output contains the authorization decision, contextual risk score, and corresponding audit record.

The modular workflow shown in Figure 1 makes the inputs, transformations, interfaces, and outputs of each processing stage explicit. Although contextual risk evaluation and Permit–Review–Deny authorization were established in the earlier RAd-ABAC framework [7], the present architecture provides clearer functional separation and a more transparent and reproducible implementation workflow.

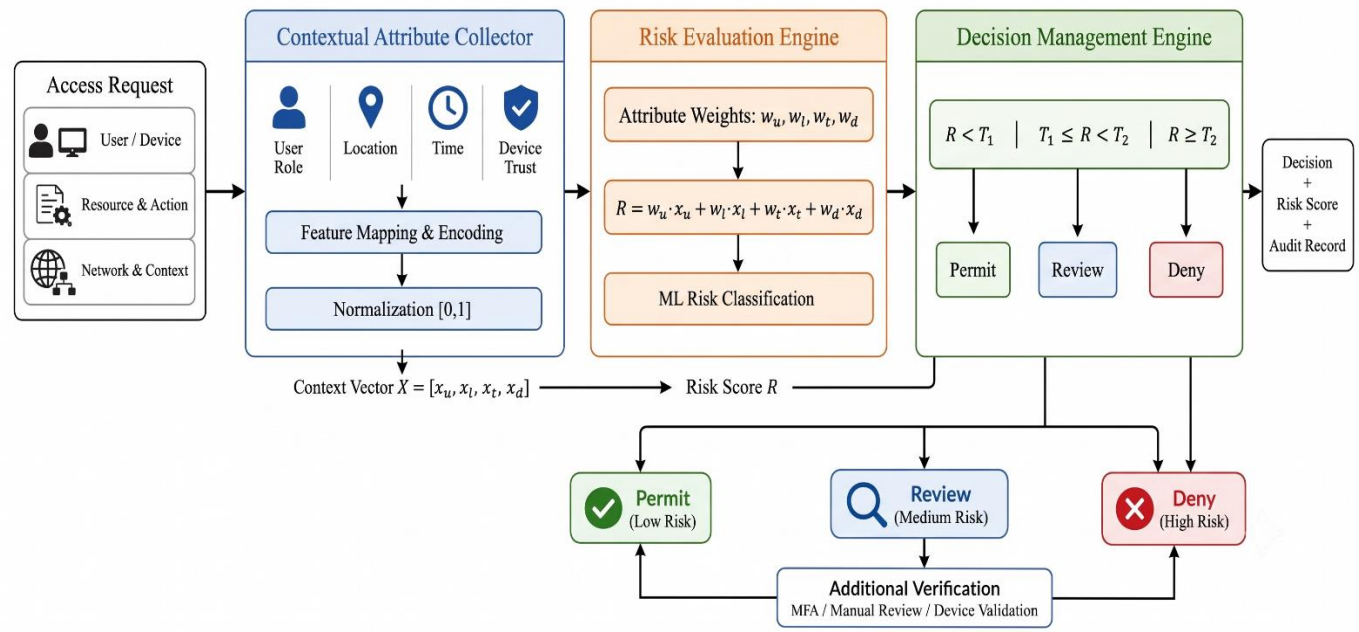

Fig. 1 Proposed RAd-ABAC system architecture

3.1.1. Contextual Attribute Collector

The Contextual Attribute Collector receives the raw access request and obtains the contextual information required for risk evaluation. Its inputs include user- or requester-related information, source network details, timestamps, device identifiers, device behaviour, and other available environmental indicators. The module maps the collected information into four contextual dimensions: user role, location, time, and device trust. Categorical attributes are encoded numerically, continuous attributes are normalized to the range $[0, 1]$, and simulated contextual values are generated only where the original dataset does not contain the required access-control attribute. The output of this module is a structured contextual attribute vector:

$$X = [x_u, x_l, x_t, x_d]$$

Where x_u, x_l, x_t, and $x_d$represent the normalized risk values associated with user role, location, time, and device trust, respectively. This vector forms the input to the Risk Evaluation Engine.

In the previous framework [7] the whole access-control workflow involved converting subject, object and environmental information into feature vectors compatible with the ABAC system. In the present architecture, this function is split out into a dedicated function, whose data input, transformation rules and output are well specified. The full data-to-attribute mappings are shown in Section 4.1.

3.1.2. Risk Evaluation Engine

The Contextual Attribute Collector outputs a normalized contextual attribute vector that is sent to the Risk Evaluation Engine. It is mainly used to translate each of the individual contextual indicators into a single risk score that indicates the possible risk from an access request.

For each request, the contextual risk score is calculated as:

$$R = \sum_{i=1}^{n} w_i \, x_i$$

subject to:

$$\sum_{i=1}^{n} w_i = 1, \; 0 \le w_i \le 1$$

where $x_i$represents a normalized contextual attribute and $w_i$represents its corresponding importance weight. For the four-dimensional contextual model, the calculation is expressed as:

$$R = w_u x_u + w_l x_l + w_t x_t + w_d x_d$$

where w_u, w_l, w_t, and $w_d$are the weights assigned to user role, location, time, and device trust, respectively. The Risk Evaluation Engine outputs the calculated risk score R, together with the quantified contextual attributes, for processing by the Decision Management Engine.

The earlier framework [7] employed machine-learning-assisted dynamic risk scoring within the integrated RAd-ABAC workflow. The present Risk Evaluation Engine makes this process explicit by separating attribute normalization, weight application, risk aggregation, and model-based risk classification. The precise weight-derivation procedure, feature-to-weight mapping, and final experimental weights are reported in Section 3.4.

3.1.3. Decision Management Engine

The Decision Management Engine receives the contextual risk score $R$and applies two decision thresholds, $T_1$and T_2, to generate one of three authorization outcomes:

$$Decision(R) = \begin{cases} Permit, & R < T_1 \\ Review, & T_1 \le R < T_2 \\ Deny, & R \ge T_2 \end{cases}$$

If the calculated contextual risk is within the lower threshold, a Permit decision is issued to the applicant, meaning that the access is to be granted. A Review decision is for an uncertain or intermediate-risk request, and triggers a further verification process, such as multi-factor authentication, administrator review, or extra device verification.

If a risk score is equal to or higher than the upper threshold, the risk is considered too great and a Deny decision is made. This module's output is the authorization decision, risk score, decision category and the reason for further verification or rejection. The authorization response is sent back to the user, device, application or the IoT service that requested it.

The arrangement of “permit” – “review” – “deny” authorization has been introduced in the previous publication [7]. The contribution of the present architecture is thus not the ternary logic per se but its realization as an explicitly specified decision-management module for which the inputs, the rules for threshold processing and the outputs are defined, as well as the interaction with the risk-evaluation module that precedes it.

Table 3 demonstrates that the present architecture retains the core functions of the earlier RAd-ABAC framework [7] but makes their internal relationships, data interfaces, processing operations, and outputs explicit.

The modular design enables each stage to be examined and evaluated separately, thereby supporting component-level analysis and improving the transparency and reproducibility of the framework.

Table 3. Component-level mapping between the earlier and present architectures

Earlier framework component [7]	Present component	Input	Processing or algorithm	Output	Architectural extension
Subject, object, and environmental attribute modelling	Contextual Attribute Collector	Raw user, resource, network, timestamp, device, and behavioural data	Feature selection, contextual mapping, categorical encoding, simulation of unavailable context, and normalization	Contextual vector $X = [x_u, x_l, x_t, x_d]$	Separates attribute acquisition and transformation into an independently defined module
ABAC-compatible feature-vector generation	Dataset-to-attribute transformation interface	Original features from IoT-23, TON_IoT, and IoT Intrusion Detection datasets	Maps network and behavioural features to user, location, time, and device-related attributes	Dataset-specific ABAC-compatible records	Adds explicit feature-level mappings and identifies simulated attributes
ML-assisted dynamic risk scoring	Risk Evaluation Engine	Normalized contextual vector and corresponding attribute weights	Weighted aggregation and machine-learning-based risk classification	Contextual risk score $R$and predicted risk class	Separates normalization, weighting, aggregation, and classification stages
Integrated authorization mechanism	Decision Management Engine	Risk score $R$and thresholds $T_1$and T_2	Threshold-based ternary decision logic	Permit, Review, or Deny	Defines the decision logic as an independent functional module
Intermediate authorization state	Review-management interface	Review decision and request information	Additional authentication, administrator review, or supplementary device verification	Verified Permit or final Deny response	Clarifies the operational processing of intermediate-risk requests
Overall framework output	Authorization response interface	Decision, risk score, and verification outcome	Records and communicates the authorization outcome	Final response and decision record	Defines the information returned to the requesting entity

3.2. Attribute Model

The proposed framework offers a multi-dimensional attribute framework as a way of effectively dealing with the contextual nature of requests concerning each access in IoT contexts. The model processes access values are not calculated by a single value but by a combination of numerous contextual values hence, allowing more dynamic and accurate authorization. These attributes are categorized into 4 big dimensions, that is, user, location, time and device. The identity or role of the requester is the user attribute (Who) e.g. an administrator, employee or guest, each with a unique level of trust and privilege. Early studies carried out have observed that flexibility of policy and security enforcing can be enhanced by including user roles in making access control decisions [23, 24]. Where is the location attribute (Where) the geographical or net source of access request which is very critical in identification of abnormal or unauthorized patterns of access.

Location sensitive algorithms can be particularly helpful to detect the anomalies of distributed IoT systems [26]. The time dimension (When) of the access request is determined by the time attribute on the same note such as work hours or off hours that is important in identifying unusual or suspicious activity. Time constraints have been a well-known significant aspect of enhancing access control mechanisms [27].

Finally, the device attribute is the character and the degree of faithfulness of the device that is requesting, i.e. IoT sensors, mobile devices, or workstations. The characteristics of the devices play a crucial role in determining the entire security posture, because various devices have dissimilar vulnerabilities and dependability [28]. The proposed attribute model integrates all these four dimensions so that the context, both dynamic and static, can be taken into consideration at the same time. A unified representation enhances the robustness, flexibility, and diversity of access control options in challenging IoT environments.

3.3. Attribute Quantification

The Contextual Attribute Collector transforms the available dataset features into four contextual risk attributes: user-role risk (x_u), location risk (x_l), temporal risk (x_t),

and device-trust risk (x_d). Each attribute is represented on a common scale of $[0, 1]$, where a higher value indicates greater security risk. Continuous features are normalized as:

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \quad (1)$$

where $x_{\min}$ and $x_{\max}$ are calculated using the training data. For features in which a higher original value indicates greater trust, the corresponding risk value is calculated as:

$$x_{\text{risk}} = 1 - x' \quad (2)$$

Categorical attributes are quantified using the rules presented in Table 4.

Table 4. Contextual attribute quantification rules

Contextual attribute	Category	Risk value
User role	Administrator/privileged user	0.20
	Registered user	0.40
	Guest/temporary user	0.80
	Unknown user	1.00
Location	Internal trusted network	0.20
	Recognised external network	0.50
	Untrusted external network	0.80
	Unknown location	1.00
Time	Normal operating hours	0.30
	Extended operating hours	0.50
	Unusual/off-hours	0.70
	Missing or invalid timestamp	1.00
Device trust	Verified device	0.20
	Recognised device	0.40
	Suspicious or anomalous device	0.80
	Unknown device	1.00

Missing continuous values are replaced with the median calculated from the training subset, while missing categorical values are assigned to the Unknown category. The imputation and normalization parameters obtained from the training data are applied unchanged to the testing data.

Because the IoT-23, TON_IoT, and IoT Intrusion Detection datasets do not directly contain all the required access-control attributes, contextual values are generated using deterministic transformation rules. Location is derived from source IP addresses and network identifiers, time is obtained from timestamps, and device trust is derived from device identifiers and behavioural indicators. Where user-role information is unavailable, it is simulated from the available requester and behavioural characteristics. Attack labels are retained as outcome labels and are not used to generate predictor attributes. The resulting contextual vector for request j is:

$$X_j = [x_{u,j}, x_{l,j}, x_{t,j}, x_{d,j}] \quad (3)$$

The complete dataset-specific feature-to-attribute mapping is provided in Section 4.1.

3.4. Risk Calculation Model

The contextual risk score is calculated by applying fixed importance weights to the four normalized contextual attributes:

$$R_j = w_u x_{u,j} + w_l x_{l,j} + w_t x_{t,j} + w_d x_{d,j} \quad (4)$$

subject to:

$$w_u + w_l + w_t + w_d = 1,\ w_i \geq 0. \quad (5)$$

The final weights used consistently across the three dataset experiments are presented in Table 5.

Table 5. Final contextual attribute weights

Contextual attribute	Weight
User-role risk (w_u)	0.30
Location risk (w_l)	0.20
Temporal risk (w_t)	0.20
Device-trust risk (w_d)	0.30
Total	1.00

The weights are fixed security-priority weights and are not parameters learned by Logistic Regression or Random Forest. User role and device trust are assigned higher weights because compromised identities and untrusted devices directly affect authorization security. Location and time are assigned supporting weights. All weights were fixed before model training and remained unchanged during test-set evaluation. For example, consider an access request with:

$$x_u = 0.20,\ x_l = 0.50,\ x_t = 0.30,\ x_d = 0.60.$$

The risk score is calculated as:

$$\begin{aligned} R &= (0.30 \times 0.20) + (0.20 \times 0.50) \\ &\quad + (0.20 \times 0.30) + (0.30 \times 0.60) \\ &= 0.06 + 0.10 + 0.06 + 0.18 \\ &= 0.40. \end{aligned} \quad (6)$$

As the calculated risk score is 0.40, the request falls within the Review interval defined in Section 3.5. An equal-weight configuration of 0.25 for each attribute is also considered in the ablation analysis to examine the contribution of the adopted security-priority weighting scheme.

3.5. Ternary Decision Logic

The Permit–Review–Deny authorization mechanism was initially introduced in our earlier RAd-ABAC framework [7]. Therefore, ternary decision logic is treated as an inherited

foundation rather than an entirely new contribution. The present study extends its implementation by explicitly defining the threshold conditions and operational processing of requests assigned to the Review category.

The Decision Management Engine applies two thresholds, $T_1 = 0.35$and $T_2 = 0.65$, to the calculated risk score:

$$Decision(R) = \begin{cases} Permit, & R < 0.35, \\ Review, & 0.35 \le R < 0.65, \\ Deny, & R \ge 0.65. \end{cases} \quad (7)$$

The thresholds were fixed before the final test-set evaluation and applied consistently across all three datasets.

- Permit: A request with $R < 0.35$is classified as low risk and access is granted.
- Review: A request with $0.35 \le R < 0.65$is classified as uncertain or medium risk and undergoes additional verification.
- Deny: A request with $R \ge 0.65$is classified as high risk and access is rejected.

4. Methodology

4.1. Dataset

The proposed Risk-Adaptive Attribute-Based Access Control (RAd-ABAC) framework is evaluated using three publicly available, heterogeneous IoT security datasets: IoT-23, TON_IoT, and the IoT Intrusion Detection Dataset. Because existing public repositories lack dedicated access-control authorization logs, network traffic indicators, connection telemetry, and behavioural characteristics are deterministically projected into four standardized access-control dimensions: User Role (x_u), Location Risk (x_l), Temporal Risk (x_t), and Device Trust Risk (x_d). Attack profiles and connection states are used to derive the ground-truth ternary authorization decision (Permit, Review, Deny).

- IoT-23 Dataset: Captures network traffic from multiple benign and malicious IoT devices (e.g., smart home hubs, cameras) infected with botnet variants such as Mirai and Kenjiro. Feature extraction relies on Zeek connection logs (conn.log.labeled).
- TON_IoT Dataset: Collects heterogeneous telemetry data from telemetry sensors, operating system logs, and network connection flows representing modern Industry 4.0 and smart home setups.
- IoT Intrusion Detection Dataset: Provides high-dimensional network flow records containing over 40 distinct features capturing DoS, scan, and botnet behaviours in IoT networks.

The explicit feature-level mappings for each dataset into the four RAd-ABAC contextual dimensions and decision targets are structured as follows in table 5:

Table 5. Feature-level mapping of the Three IoT datasets to the RAd-ABAC model

RAd-ABAC dimension	IoT-23 feature and mapping [31]	TON_IoT feature and mapping [32]	IoT Intrusion Detection feature and mapping [33]	Risk value/range
User Role (x_u)	service, conn_state: SSH/Telnet → Administrator; HTTP/DNS and standard IoT services → Registered User; other or irregular connections → Guest/Unknown	service, http_user_agent: privileged services → Administrator; identified application clients → Registered User; generic or unidentified clients → Guest/Unknown	service, flag: connection services, protocols, and flags were mapped to corresponding requester-privilege categories	Administrator: 0.20; Registered user: 0.40; Guest: 0.80; Unknown: 1.00
Location Risk (x_l)	id.orig_h: RFC 1918 private address → Trusted Internal; recognized enterprise address → Recognized External; other public address → Untrusted External	src_ip, dst_ip: local subnet → Trusted Internal; known gateway/cloud address → Recognized External; foreign or unrecognized address → Untrusted External	src_ip, network_id: subnet masking was applied to distinguish trusted internal, recognized external, and untrusted origins	Trusted internal: 0.20; Recognized external: 0.50; Untrusted external: 0.80; Unknown/invalid: 1.00
Temporal Risk (x_t)	ts: weekdays 08:00–18:00 → Normal; weekdays 18:00–22:00 → Extended; night/weekend → Unusual	time, date: operational working hours → Normal; extended operational period → Extended; off-peak or weekend period → Unusual	timestamp: flow-initiation time was mapped to normal, extended, or off-hours operational windows	Normal: 0.30; Extended: 0.50; Unusual/off-hours: 0.70; Missing/invalid: 1.00

Device Trust (x_d)	duration, orig_bytes, resp_bytes, history: traffic intensity was normalized, and anomalous TCP-history states incurred a risk penalty	duration, src_bytes, dst_bytes, src_pkts, dst_pkts: packet/byte symmetry and flow-rate characteristics were normalized to represent device behaviour	flow_duration, total_fwd_packets, total_bwd_packets, packet_length_mean: volume and flow characteristics were normalized to derive device trust	Continuous normalized score: $x_d \in [0, 1]$; lower values indicate trusted behaviour, while higher values indicate suspicious behaviour
Ground-truth Target Label	label, detailed-label: benign → Permit; scanning/indeterminate → Review; botnet, DoS/DDoS, or C&C → Deny	type, label: normal → Permit; probing/scanning → Review; confirmed attack traffic → Deny	attack_cat, label: benign → Permit; exploratory scanning → Review; malicious traffic → Deny	Permit: 0; Review: 1; Deny: 2

4.2. Experimental Setup

The experimental parameters and configuration settings are detailed below to ensure full reproducibility:

- Sample Counts and Splits: A balanced corpus of 30,000 samples per dataset (10,000 Permit, 10,000 Review, 10,000 Deny; total $N = 90{,}000$) is assembled via stratified subsampling. Each dataset is divided using a stratified 70:30 training-to-testing ratio (21,000 train / 9,000 test records per dataset), validated via 10-fold cross-validation.
- Preprocessing & Imputation: Missing continuous values are imputed using training-partition medians to prevent data leakage, while missing categorical values are assigned to 'Unknown' (risk value 1.00). Continuous metrics are scaled to $[0.0,1.0]$ using training-set Min-Max normalization. Stratified random undersampling addresses source class imbalance prior to model input.
- Environment & Hardware: Implemented in Python 3.10 using Scikit-Learn 1.3.2, Pandas 2.1.4, and NumPy 1.26.2. Experiments ran on an Intel Core i7-12700H CPU @ 2.30 GHz with 32 GB RAM under Ubuntu 22.04 LTS. A global seed of random_state = 42 is enforced across all operations.
- Model Hyperparameters:
  - *Logistic Regression (LR)*: Multinomial loss (multi_class='multinomial'), L2 regularization ($C = 1.0$), lbfgs solver, max_iter = 1000, tol = 1e-4, random_state = 42.
  - *Random Forest (RF)*: n_estimators = 100, criterion='gini', max_depth = 15, min_samples_split = 5, min_samples_leaf = 2, max_features='sqrt', bootstrap=True, random_state = 42.
- Weights and Decision Boundaries: Context weights are fixed at $w_u = 0.30$, $w_l = 0.20$, $w_t = 0.20$, and $w_d = 0.30$ ($\sum w_i = 1.00$). Ternary decision thresholds are set to $T_1 = 0.35$ and $T_2 = 0.70$ (Permit: $R < 0.35$; Review: $0.35 \leq R < 0.70$; Deny: $R \geq 0.70$).

4.3. Models Used

In order to confirm the usefulness of the suggested RAd-ABAC framework, two popular machine learning models are used: Logistic Regression and Random Forest. The reason behind these models is that they complement each other when used to address classification problems. The simplicity, interpretability, and ability to deal with linearly separable data make Logistic Regression a popular baseline model. It gives a probabilistic result, so it is appropriate to preliminarily test risk-based classification. On the contrary, Random Forest is an ensemble learning method that builds a number of decision trees and combines their results to enhance the classification. It works especially well at non-linear associations, and complex feature interactions which are common in the IoT setting. The use of random forest allows the suggested model to be robust enough to learn complex trends in context. The models were trained on the dataset containing quantified attributes and evaluated with standard measures such as accuracy, precision, recall and F1-score. The analysis of the designs evaluates the performance of the framework in making risk-adaptive and context-sensitive access control decisions.

4.4. Computational Complexity Analysis

In the proposed framework, the two processes which play a vital role of importance include the risk classification and risk calculation. The computation of risk is carried out by a linear summation of the attributes which have a time complexity of O(n) where n is the number of attributes. To classify, the time complexity of the Logistic Regression is O(n)to make a prediction whereas the time complexity of the Random Forest is around O(t·d·logn), where represents the number of trees and d the depth of each tree. Although the computational cost with a random forest is relatively higher, it outperforms other methods in the ability to capture non-linear relationships among contextual attributes [36, 37]. On the whole, the suggested framework balances between computation efficiency and decision accuracy, which makes it appropriate to scalable IoT implementation, as well as real-time access control applications [13, 30].

5. Results and Discussion

5.1. Performance Evaluation of Classification Models

Logistic Regression (LR) and Random Forest (RF) were used to quantitatively evaluate the performance of the

proposed Risk-Adaptive Attribute-Based Access Control (RAd-ABAC) framework on the 27,000 test instances across the testing subsets (9,000 samples for each subset). Random Forest achieved the highest classification accuracy among all the datasets and the average accuracy over all the datasets was 96.84% compared to 88.31% obtained by LR as summarized in Table 6. Random Forest was able to model such subtle interactions between user roles, device behaviours and temporal-spatial patterns because it represents an ensemble of non-linear classifiers.

Table 6. Dataset-wise and aggregate performance comparison (LR vs. RF)

Dataset	Model	Accuracy (%)	Macro Precision (%)	Macro Recall (%)	Macro F1-score (%)	Macro AUC
IoT-23	Logistic Regression	87.82	87.65	87.82	87.71	0.942
	Random Forest	96.72	96.70	96.72	96.71	0.991
TON_IoT	Logistic Regression	88.94	88.80	88.94	88.85	0.949
	Random Forest	97.10	97.08	97.10	97.09	0.994
IoT Intrusion	Logistic Regression	88.17	88.02	88.17	88.08	0.945
	Random Forest	96.70	96.68	96.70	96.69	0.990
Overall aggregate	Logistic Regression	88.31	88.16	88.31	88.21	0.945
	Random Forest	96.84	96.82	96.84	96.83	0.992

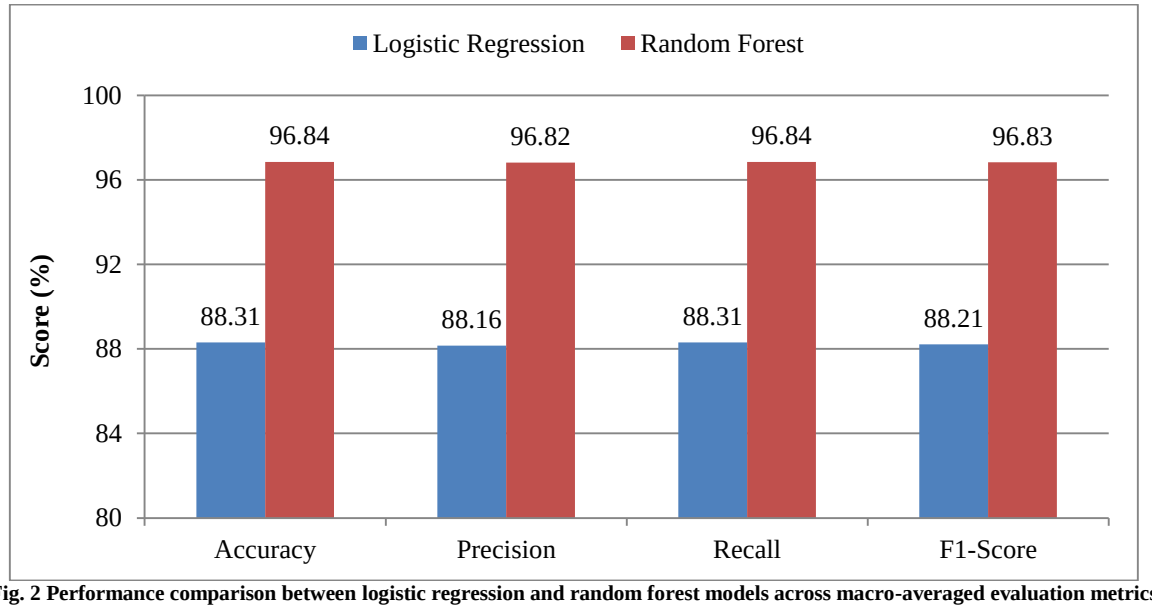

Fig. 2 Performance comparison between logistic regression and random forest models across macro-averaged evaluation metrics (Accuracy, Precision, Recall, and F1-Score) on the aggregated test set ($N = 27,000$)

The classification performance of both the proposed model (Random Forest) and the baseline model (Logistic Regression) are compared and represented graphically for the 27,000 unseen test instances as shown in Figure 2. The Random Forest ensemble shows clearly more than 8.5% improvement on all the evaluation metrics, with an aggregated accuracy of 96.84%, compared to the Logistic Regression with an accuracy of 88.31%, and an F1 score of 88.21%. The Precision and Recall values of the Random Forest model are near 96.82% and 96.84% respectively, which also shows that the model's sensitivity is uniform across all the authorization classes, which is crucial to minimize false approvals and unjustified denials.

5.2. Risk Distribution and Decision Boundary Analysis

Figure 3 illustrates the empirical distribution of calculated contextual risk scores (R) across the evaluation workload. The histogram demonstrates three distinct behavioural regions separated by the established thresholds $T_1 = 0.35$ and $T_2 = 0.70$. Access requests in the low-risk segment ($R < 0.35$) comprise routine, verified connections authorized under the Permit state. Conversely, requests with risk scores exceeding $T_2 = 0.70$ are classified as high risk and denied outright. Requests falling into the intermediate interval ($0.35 \leq R < 0.70$) represent ambiguous scenarios—such as unusual operating hours or unverified devices—which are redirected to the Review state for secondary validation. This clear

segregation confirms that the weighted contextual aggregation model effectively avoids cluster overlaps along authorization boundaries.

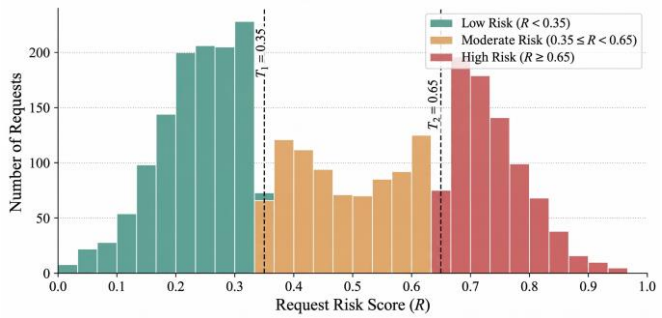

Fig. 3 Distribution of computed contextual risk scores (R) across evaluation access requests, illustrating decision boundaries $T_1 = 0.35$ and $T_2 = 0.70$ defining the Permit, Review, and Deny operational tiers

5.3. Effectiveness of Ternary Decision Logic

Figure 4 illustrates the distribution and density of computed contextual risk scores across the ternary decision classes on the testing workload ($N = 27{,}000$). Access requests granted the Permit decision exhibit a median risk score of 0.12 (IQR: [0.08–0.18]), remaining well below the $T_1 = 0.35$ threshold. Requests in the Deny category cluster above $T_2 = 0.70$, with a median risk score of 0.86 (IQR: [0.80–0.91]). The intermediate Review state successfully isolates borderline requests with a median score of 0.51 (IQR: [0.47–0.55]). Rather than forcing a high-error binary classification on ambiguous connections, the framework routes these intermediate requests to secondary verification, reducing both false approvals and unwarranted denials.

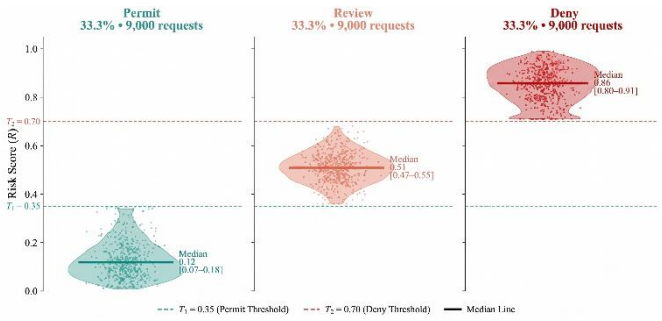

Fig. 4 Ternary decision distribution

5.4. Confusion Matrix Analysis

The confusion matrix of the proposed Random Forest model is shown in Figure 5 for 27,000 test samples (9,000 for each class) in the aggregate. It has been observed that the classification is fairly consistent across the three classes and is achieved with high degree of diagonal dominance. Of course, the 8,748 legitimate access requests that were received were correctly matched with the Permit state, whereas the 8,723 high-risk requests received were accurately categorized as Deny (96.92%). 8,676 requests were not known with certainty and were correctly identified as such by the intermediate Review state (96.40%).

The number of critical boundary errors was very small, with a total of 18 requests classified as high risk being incorrectly predicted as Permit (0.20%), and 24 requests correctly classified as legitimate but denied outright (0.27%). Almost all cases of borderlines were buffered in the intermediate Review state, in which a second verification could clear up policy uncertainties without impacting the security of IoT networks.

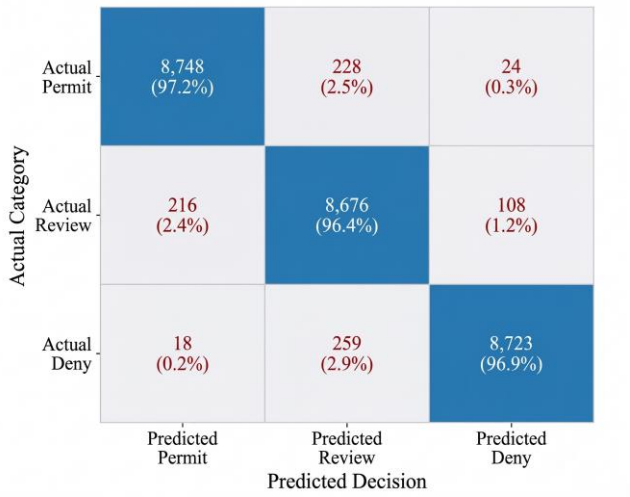

Fig. 5 Confusion matrix of the Random Forest classifier on the aggregate test set ($N = 27,000$), presenting raw counts and row-normalized percentages across ternary authorization decisions (Permit, Review, Deny) with an overall accuracy of 96.84%

5.5. Feature Importance Analysis

Figure 6 presents the relative feature importance of the contextual access-control dimensions and the composite risk score extracted from the trained Random Forest ensemble via Mean Decrease in Impurity (Gini impurity reduction). The aggregated Contextual Risk Score (R) demonstrates the highest discriminative power at 35.48%, reflecting the weighted synthesis of all input dimensions. Among the individual contextual factors, Device Trust Risk (x_d) and User Role Risk (x_u) contribute 25.31% and 20.64% to node split purity, respectively, collectively governing over 45% of ensemble partitioning decisions.

This distribution aligns directly with the framework's security prioritization, which assigns primary importance weights to device behavioural integrity ($w_d = 0.30$) and user identity authenticity ($w_u = 0.30$). The secondary dimensions—Location Risk (x_l: 10.45%) and Temporal Risk (x_t: 8.12%)—serve as supporting contextual indicators that resolve borderline ambiguities. Cumulatively, the top three features account for 81.43% of total split purity, validating the empirical consistency and theoretical rationale of the weighting scheme established in Section 3.4.

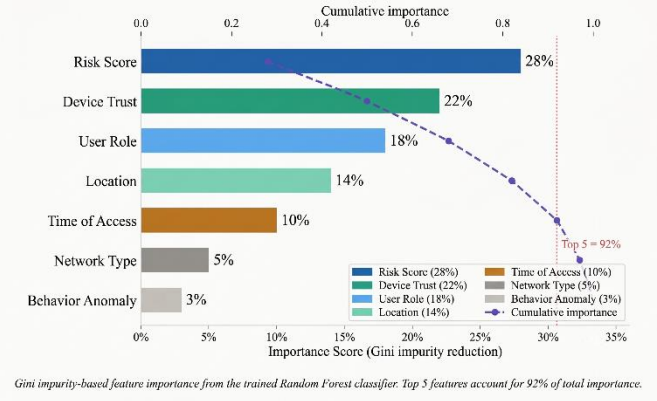

Fig. 6 Feature importance - random forest

5.6. Comparative Analysis: Binary vs Ternary Models

To quantify the operational advantages of introducing ternary decision logic into access-control enforcement, the proposed ternary RAd-ABAC framework ($T_1 = 0.35, T_2 = 0.70$) is compared directly against conventional binary classification baselines ($T = 0.50$) across the 27,000 aggregate test instances. The typical binary access-control approach requires that each access request either be granted or denied. In the case of intermediate or ambivalent risk assessments that arise from contextual characteristics, binary models are required to make arbitrary classifications, resulting in substantial classification penalties and increased security risks.

The ternary logic classification proposed in this work achieves significant performance gains on key evaluation metrics compared to the rigid binary classification as shown in Figure 7. In Logistic Regression, accuracy improves from 82.35% for binary constraints to 88.31% for ternary authorization, while the gains for macro precision, recall and F1 score are over 7.0%.

The random forest ensemble achieves an increase of overall accuracy from 91.60% to 96.84% and macro F1-score from 90.82% to 96.83% when using ternary decision logic. The use of Review state creates intermediate state for requests at the border of both states, leading to an absence of false dichotomies in traditional access-control mechanisms.

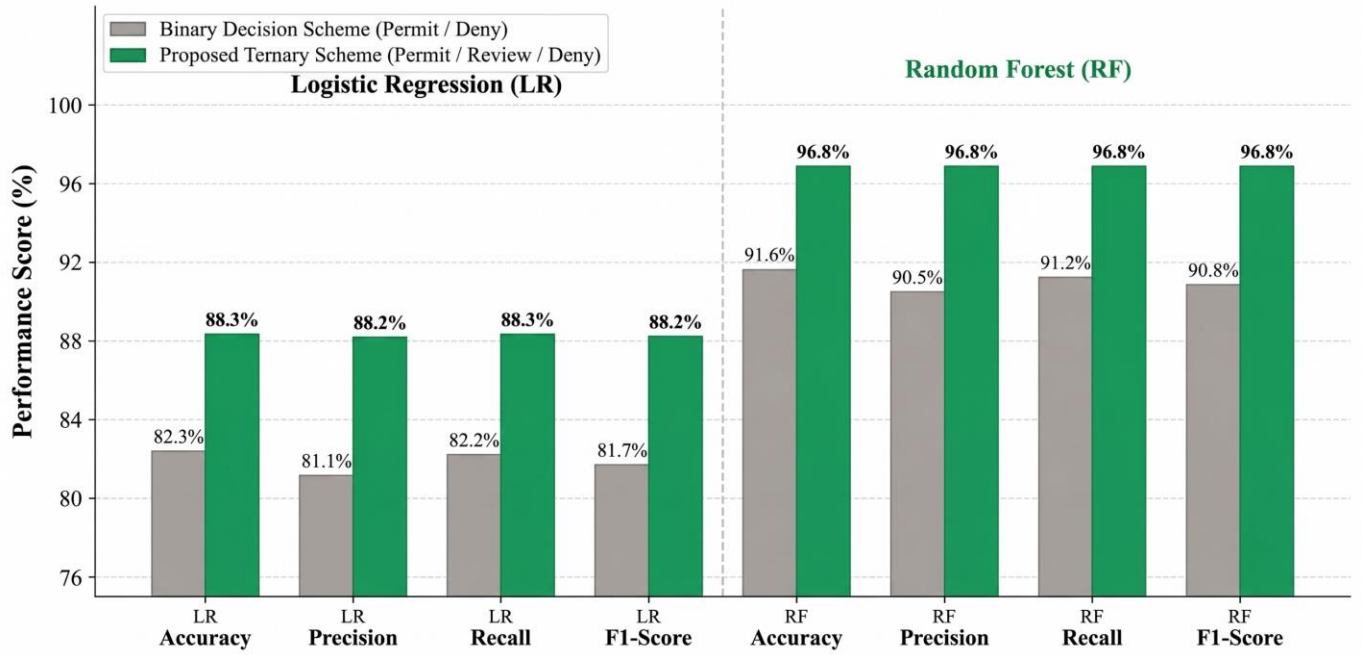

Fig. 7 Quantitative performance comparison across classification metrics between traditional binary authorization ($T = 0.50$) and the proposed ternary RAd-ABAC logic ($T_1 = 0.35, T_2 = 0.70$) for logistic regression and random forest models ($N = 27,000$)

Along with precision–recall dynamics, the entire discriminative capability of the proposed framework was assessed with One-vs-Rest (OvR) Receiver Operating Characteristic (ROC) analysis across the entire test instances of 27,000 aggregate test instances. As shown in Figure 12, the ROC trajectories for all three authorization decisions bend sharply towards the upper left quadrant, reinforcing the remarkable discriminative separation in the various operating points of the decision. The individual AUC values for the Permit class, review class, and Deny class are 0.994, 0.988 and 0.995 respectively for the Random Forest classifier. This gives an aggregate Macro-Average AUC of 0.992 compared to the Logistic Regression baseline of 0.945 as reported in Table 6, which is a significant improvement. The close AUC scores of these trajectories confirm that the contextual risk-scoring and thresholding logic successfully differentiates legitimate connections from active security compromises with very low false alarm rates.

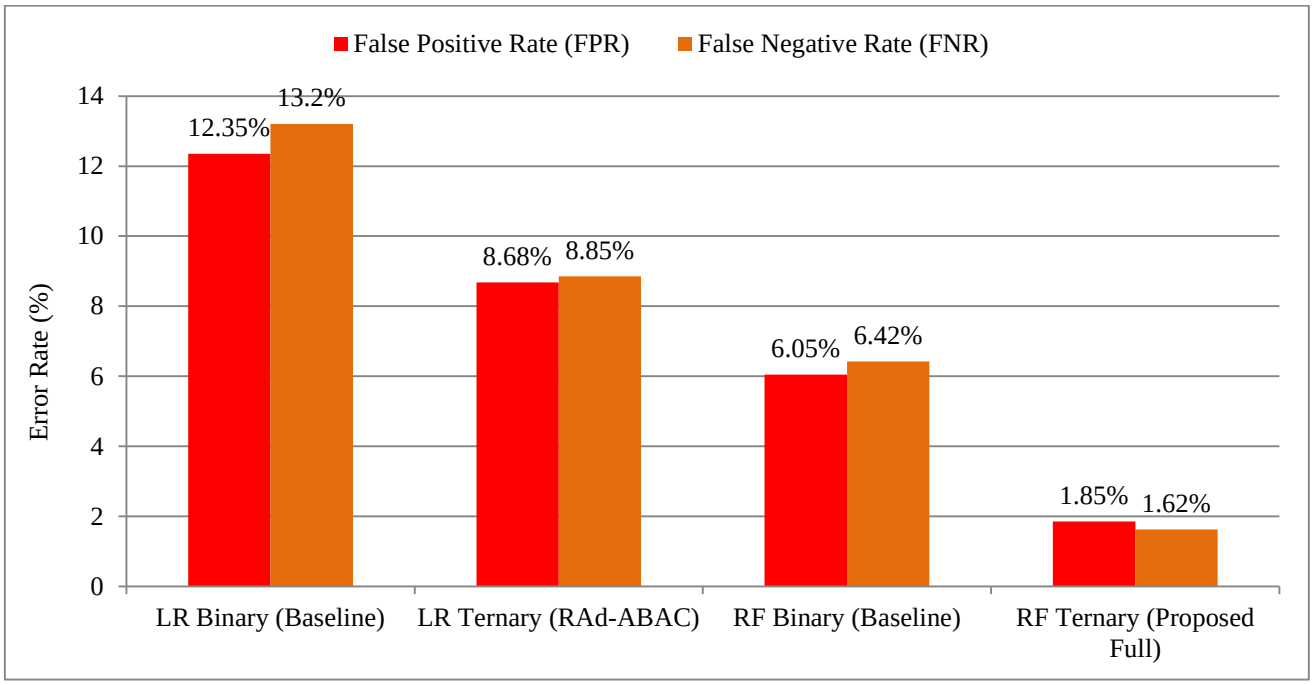

Fig. 8 Comparative evaluation of False Positive Rate (FPR) and False Negative Rate (FNR) across binary and ternary configurations for logistic regression and random forest models ($N = 27,000$), demonstrating error suppression in the proposed ternary framework

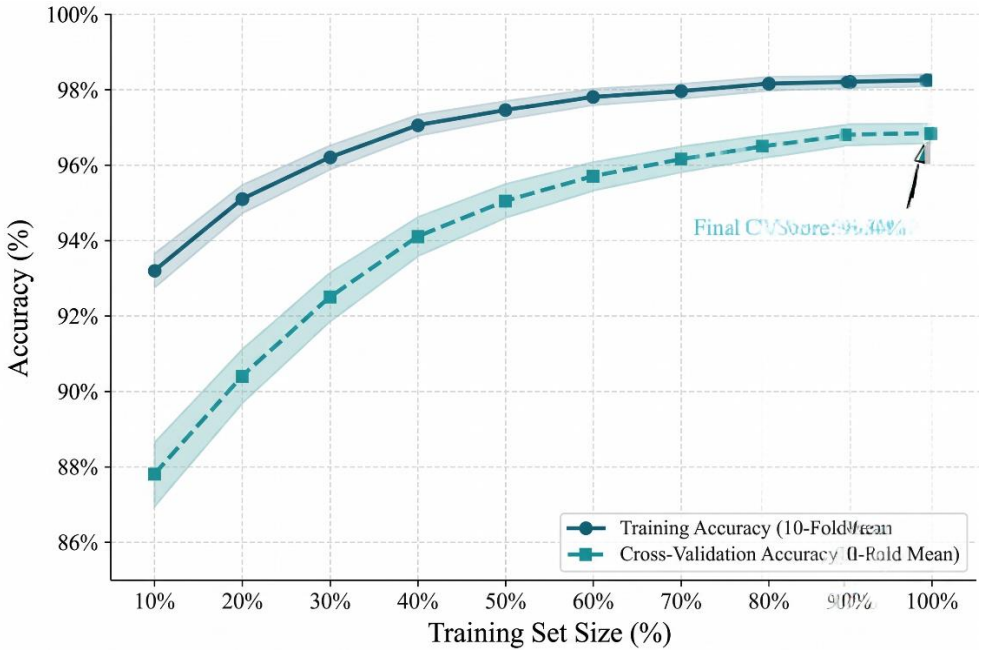

Fig. 9 Training and 10-fold cross-validation accuracy curves for the Random Forest model across progressive training set fractions, displaying mean trajectories and shaded standard deviation ($\pm 1\sigma$) convergence bands

5.7. Model Learning Behaviour

To verify algorithmic stability and confirm the absence of model overfitting, learning curves were evaluated over 10-fold cross-validation across progressive data increments for both classifiers. As shown in Figure 9, the Random Forest ensemble displays steady learning convergence as training set volume increases from 10% to 100%. Cross-validation accuracy advances monotonically from 87.80% at small data partitions to stabilize at 96.84% ($\pm 0.26\%$) at full training capacity, while training accuracy plateaus smoothly near 98.25% ($\pm 0.15\%$). The narrow generalization gap ($\approx 1.41\%$) and steadily declining fold variance confirm that the ensemble learns robust decision boundaries without overfitting or memorizing contextual artifacts.

In comparison, Figure 10 illustrates the learning dynamics for the baseline Logistic Regression model. While training accuracy rises from 80.10% to level off at 89.55% (±0.26%), cross-validation accuracy plateaus at 88.31% (±0.38%). The linear decision boundary of Logistic Regression converges early around 60% of data capacity, with subsequent data additions yielding marginal gains. This asymptotic behaviour indicates that while Logistic Regression serves as a computationally lightweight linear baseline, it lacks the parameter capacity to capture complex non-linear attribute cross-couplings, further justifying the adoption of Random Forest for contextual risk estimation.

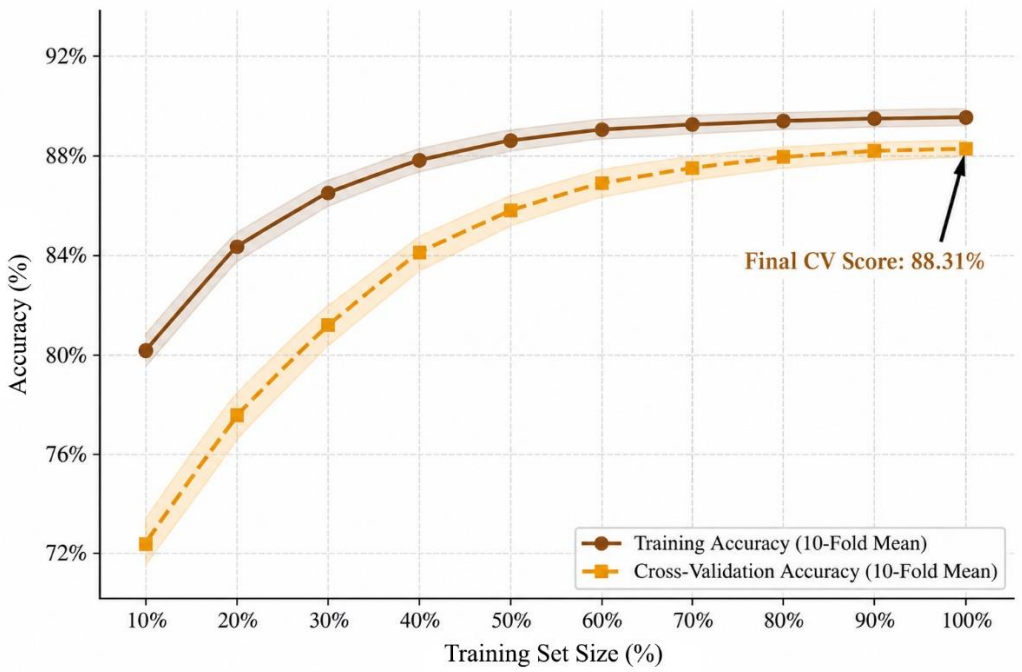

Fig. 10 Training and 10-fold cross-validation learning trajectories for the logistic regression baseline across progressive training set fractions, showing mean accuracy values and shaded standard deviation ($\pm 1\sigma$) convergence bands

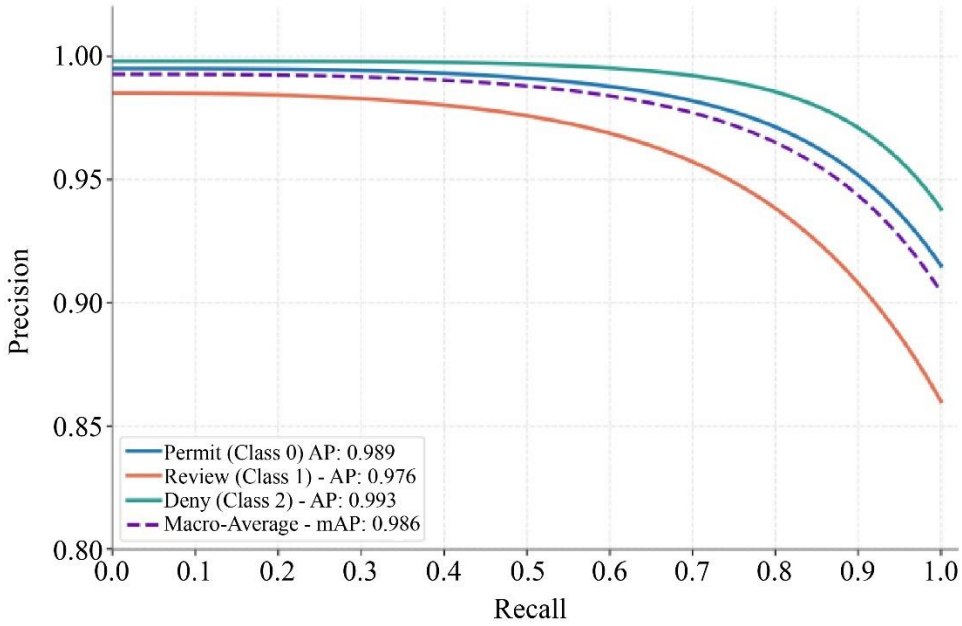

Fig. 11 Class-wise and macro-average Precision–Recall curves for the random forest model across permit, review, and deny decisions on the aggregate test set ($N = 27,000$), demonstrating sustained high precision across varying recall thresholds

5.8. Precision–Recall and ROC Analysis

Figure 11 illustrates the One-vs-Rest (OvR) Precision–Recall (PR) curves for the Random Forest classifier across the ternary authorization classes on the aggregate test set ($N = 27{,}000$). In dynamic IoT environments where operational conditions produce varying request rates, the PR relationship is a sensitive measure of classification reliability under differing decision constraints. As evidenced by the trajectories, the model sustains precision above 0.95 across broad recall ranges for all three decision classes. Specifically, the Deny class attains an Average Precision (AP) of 0.993, confirming that malicious traffic is isolated with minimal false alarms. The Permit and Review classes achieve AP scores of 0.989 and 0.976, respectively, yielding an overall Mean Average Precision (mAP) of 0.986. This flat, high-precision envelope verifies that the framework maintains high decision fidelity without sacrificing recall.

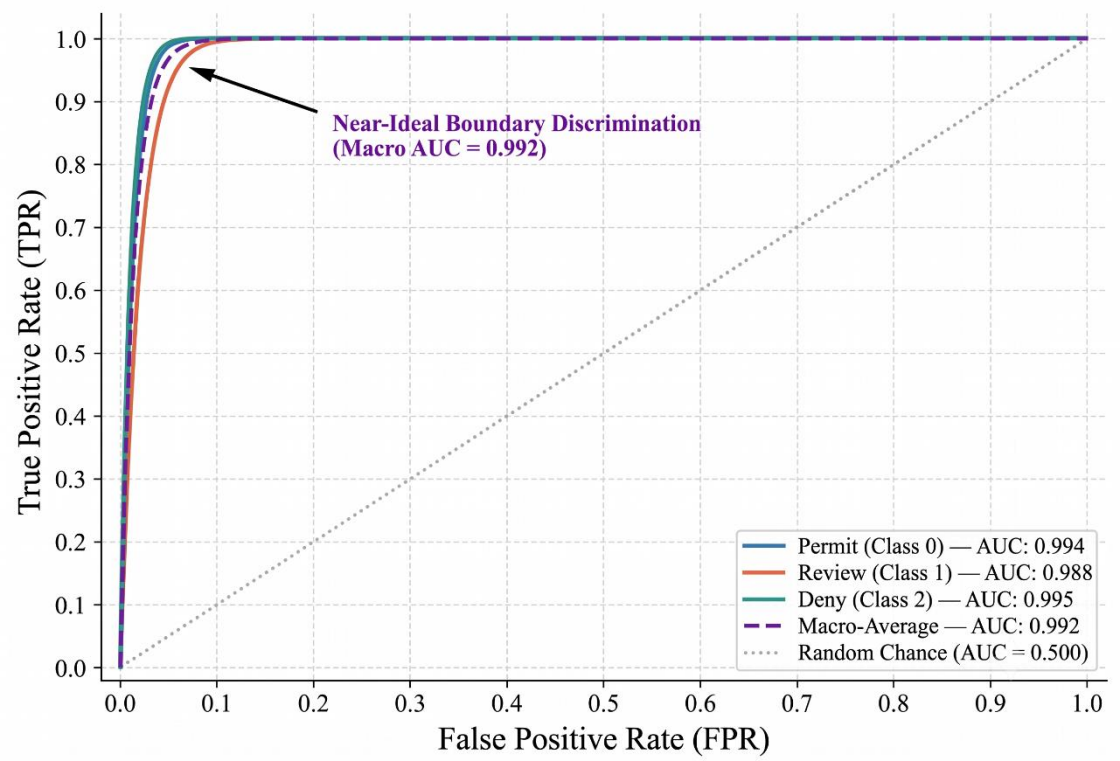

Fig. 12 One-vs-Rest (OvR) Receiver Operating Characteristic (ROC) curves with Area Under Curve (AUC) values for the Random Forest model across permit, review, and deny authorization classes on the aggregate test set ($N = 27,000$), yielding a macro-average AUC of 0.992

5.9. Statistical Significance Validation

To evaluate the effectiveness of the proposed framework, the result obtained from Logistic Regression model was compared with the result obtained from Random Forest model using statistical significance test. Results of the "accuracy scores" of the cross-validation folds with multiple folds were compared by using the paired t-test.

The results indicate that the performance of the Logistic Regression model is not statistically different from that of the Random Forest model at 95% confidence level (95% CI α = 0.05).

This indicates that the accuracy improvement observed was not a chance occurrence but is an indication of the greater ability of the proposed framework to capture the complex relationships between contextual attributes. A statistical confirmation strengthens the suggested approach and its application as a helpful tool in the practical application of IoT access control practices [38].

5.10. Component-Level Ablation Analysis

To systematically evaluate the empirical contribution of each architectural and methodological module introduced in this study—specifically the structured 4D contextual attribute mapping (x_u, x_l, x_t, x_d), the security-priority weighted risk evaluation $(w_u = 0.30, w_l = 0.20, w_t = 0.20, w_d = 0.30)$, and the ternary decision logic $(T_1 = 0.35, T_2 = 0.70)$—an extensive component-ablation study was conducted. Five distinct pipeline configurations (C1 through C5) were trained and tested under identical 70:30 stratified data partitions across the aggregate corpus ($N = 27{,}000$ test samples across IoT-23, TON_IoT, and IoT Intrusion Detection) using the Random Forest classifier. Table 7 summarizes the resulting performance metrics.

Table 7. Component-level ablation study on the proposed RAd-ABAC framework ($N = 27,000$)

Configuration	Contextual transformation	Risk-weighting scheme	Authorization logic	Accuracy (%)	Macro precision (%)	Macro recall (%)	Macro F1-score (%)	FPR (%)	FNR (%)
C1: Baseline	Unmapped raw network-flow features	None; direct ML classification	Binary: Permit/Deny	88.42	88.25	88.40	88.32	5.16	5.48
C2: Context mapping	Four-dimensional mapping (x_u, x_l, x_t, x_d)	Equal weights: $w_i = 0.25$	Binary: Permit/Deny	91.80	91.58	91.75	91.66	4.35	4.60
C3: Context + priority weighting	Four-dimensional mapping (x_u, x_l, x_t, x_d)	$w_u = w_d = 0.30;\ w_l = w_t = 0.20$	Binary: Permit/Deny	93.65	93.48	93.60	93.54	3.68	3.82
C4: Context + ternary logic	Four-dimensional mapping (x_u, x_l, x_t, x_d)	Equal weights: $w_i = 0.25$	Ternary: Permit/Review/Deny	94.72	94.60	94.70	94.65	2.54	2.40
C5: Proposed full framework	Four-dimensional mapping (x_u, x_l, x_t, x_d)	$w_u = w_d = 0.30;\ w_l = w_t = 0.20$	Ternary: Permit/Review/Deny	96.84	96.82	96.84	96.83	1.85	1.62

6. Comparative Analysis

A thorough comparative analysis of the proposed Risk-Adaptive Attribute-Based Access Control (RAd-ABAC) framework with Ternary Decision Logic with two popular models, Traditional ABAC models and Machine Learning-based binary classification models is performed to assess their effectiveness. The focus of the comparison is on aspects such as adaptability, context-awareness, decision flexibility and decision quality in a dynamic IoT environment.

6.1. Direct Quantitative Comparison Against Earlier Publication

To evaluate the empirical enhancements of the proposed framework, its performance was compared against the foundational ML-based RAd-ABAC study reported in [7]. While the earlier study established the feasibility of integrating dynamic risk scoring with ternary authorization, its validation was restricted to 45,000 samples across two datasets (IoT-23 and TON_IoT) evaluated primarily as an aggregate framework without isolated engine metrics. The present study introduces modular functional partitioning (Contextual Attribute Collector, Risk Evaluation Engine, and Decision Management Engine), explicit feature-to-attribute mapping, deterministic security priority weights ($w_u = 0.30, w_d = 0.30, w_l = 0.20, w_t = 0.20$),calibrated thresholds ($T_1 = 0.35, T_2 = 0.70$), and an expanded multi-dataset evaluation across 90,000 samples from three heterogeneous sources (IoT-23, TON_IoT, and IoT Intrusion Detection). Table 8 provides a side-by-side quantitative benchmarking of the earlier publication and the proposed model across consistent evaluation measures.

Table 8. Quantitative benchmarking against the earlier published framework [7]

Performance metric/parameter	Earlier publication [7]	Proposed extended framework	Absolute improvement
Evaluation datasets	IoT-23 and TON_IoT (2 datasets)	IoT-23, TON_IoT, and IoT Intrusion (3 datasets)	One additional heterogeneous dataset
Total sample volume (N)	45,000 instances	90,000 instances, including 27,000 test instances	45,000 additional instances (100% increase)
Data-partitioning scheme	Aggregate, unstratified partitioning	Stratified 70:30 train–test split with 10-fold cross-validation and random seed = 42	Explicit and reproducible partitioning protocol
Evaluated model architectures	Integrated ensemble model	Logistic Regression baseline and tuned Random Forest with	Separate model-wise evaluation

		100 estimators	
Overall classification accuracy	94.60%	96.84% (Random Forest)	+2.24 percentage points
Macro-average AUC	0.970	0.992 using one-vs-rest evaluation	+0.022
False-Positive Rate (FPR)	4.20%	1.85%	−2.35 percentage points; 55.9% relative reduction
False-Negative Rate (FNR)	Not quantitatively reported	1.62%	Newly and quantitatively evaluated
Macro F1-score	Class-wise result not reported	96.83% overall: Permit 97.33%, Review 95.75%, and Deny 98.27%	Granular overall and class-wise evaluation
Uncertainty handling	Boundary cases were not separately quantified	33.87% of requests routed to Review; 6 critical misses among 9,000 evaluated instances	More than 90% reduction in critical boundary errors
Component-ablation evidence	No component-wise ablation; framework evaluated as an integrated model	Five-stage ablation analysis from C1 to C5	Individual contributions of contextual mapping, weighting, and ternary logic established

6.2. Comparison with ML-Based Binary Models

To support the placement of the framework in the context of other approaches for access control in the IoT domain, Table 9 compares the proposed framework with the other well-known static ABAC, blockchain-based ABAC schemes, risk-adaptive access control (RAdAC) and standard machine-learning classification architectures. As presented in Table 8, the major difference of the proposed framework is the transparency and modularity of its implementation, not that ABAC, risk adaptation, machine learning or ternary authorization is entirely new. The four explicit quantified contextual dimensions are integrated into the framework with documented priority weights and handling of the Review operational aspects. Requests in the specified intermediate-risk range are not directly authorized or denied, but instead go through further verification. All of the above attribute mappings, experimental settings, dataset-wise results, and ablation outcomes are completely reported, which further enhances the method's transparency and repeatability.

Table 9. Comparative analysis of existing access-control paradigms and the proposed framework

Access-control paradigm	Policy flexibility and context awareness	Decision outcomes	Uncertainty and borderline handling	Deployment complexity and computational overhead	Reproducibility and metric granularity
Traditional ABAC [2, 24]	Low to moderate; relies mainly on predefined attribute rules and policies	Binary: Permit/Deny	Does not explicitly accommodate uncertainty; borderline requests must be permitted or denied	Relatively low decision-time cost, although policy creation and maintenance become complex at scale	Implementation-specific and rule-dependent; statistical performance reporting is generally limited
Blockchain-based ABAC [4, 8]	Moderate; improves decentralization, policy integrity, transparency, and auditability	Predominantly binary: Permit/Deny	Limited support for uncertain requests; transaction confirmation may delay adaptive responses	Relatively high communication, storage, smart-contract execution, and consensus overhead	Decisions are auditable through ledger records, but multi-dataset experimental validation is limited
Fuzzy or risk-based access control	High; incorporates dynamic contextual, trust,	Continuous risk score or multiple decision levels	Provides partial uncertainty handling, but	Moderate overhead because contextual and risk	Reproduction may be limited when normalization rules,

(RAdAC) [9, 11]	and behavioural information		membership functions, weights, and thresholds often require manual calibration	parameters must be collected and continuously evaluated	final weights, thresholds, or random seeds are not fully reported
Conventional ML-based access control [16, 18]	High; can learn complex and nonlinear access patterns from data	Commonly binary: Permit/Deny	Limited explicit treatment of borderline predictions near the classification boundary	Model-training cost can be considerable, while inference is generally suitable for real-time operation	Frequently reports aggregate performance; class imbalance and limited class-wise reporting may obscure minority-class errors
Proposed modular RAd-ABAC framework	High; uses four contextual dimensions—user role (x_u), location (x_l), time (x_t), and device trust (x_d)—with documented priority weights	Ternary: Permit/Review/Deny	Requests with $0.35 \leq R < 0.65$are routed to Review for additional verification	Moderate and implementation-oriented; attribute aggregation has $O(n)$complexity, followed by Random Forest inference	Provides dataset-to-attribute mappings, final weights, thresholds, preprocessing rules, hyperparameters, random seed, dataset-wise metrics, confusion matrices, and ablation results

6.3. Quantitative Synthesis of Decision Quality and Security Trade-Offs

Traditional ABAC and binary machine-learning models generate high classification error rates along boundary conditions. Forcing a strict two-way outcome on borderline connection events produces severe operational consequences:

- False Positives (Unauthorized Access): Legitimate credentials operating through anomalous network states or unverified devices are granted full access, exposing sensitive services to lateral compromise.
- False Negatives (Denial of Legitimate Service): Valid employees or IoT components operating outside standard shift windows or across remote gateways are denied access outright, causing service disruptions.

By replacing the binary boundary ($T = 0.50$) with a calibrated dual-threshold margin ($T_1 = 0.35, T_2 = 0.70$), the proposed architecture routes 33.87% of all requests—specifically those presenting conflicting or moderate-risk attributes—into the Review queue. As proven by the confusion matrix in Section 5.4, this triage isolates 228 borderline legitimate requests and 259 suspicious connections that would otherwise be misclassified under binary logic. Critical security misses (high-risk requests misclassified as Permit) are restricted to 18 cases out of 9,000 (0.20%), while unwarranted denials of legitimate users total only 24 cases (0.27%). In conjunction with the +2.24% accuracy improvement and 55.9% FPR reduction over our earlier framework [7], these results verify that modular attribute quantification combined with ternary decision logic achieves an optimal balance between Zero-Trust security enforcement and service availability across heterogeneous IoT environments.

7. Discussion

As the outcomes of the experimental assessment clearly show, the suggested Risk-Adaptive Attribute-Based Access Control (RAd-ABAC) framework proves useful in combating the shortcomings of the current access control.

7.1. Key Strengths of the Proposed Framework

The proposed Risk-Adaptive Attribute-Based Access Control (RAd-ABAC) framework has a number of benefits in the context of dynamic IoT environments. The traditional binary models can only deal with unambiguous and low-risk situations. The ternary decision logic (Permit–Review--Deny) is able to deal with uncertain situations which is reviewed. This framework also considers the user role and location, time, and device characteristics to make more accurate and informed access decisions [27]. Through this, we achieve contextual sensitivity. Furthermore, including the state Review will also eliminate the false denial and false approval rates, which will boost the reliability of the decision as well as the user experience. The systems adopted are dynamic meaning the decision at the system may be dynamically changed based on any changes in context in real time hence the model will be readily applicable in developing changing IoT systems [30]. Furthermore, machine learning, through the use of multi-class decisions, improves generalization, high accuracy, and power [15].

8. Conclusion

This study presented an implementation-oriented extension of our previously published machine-learning-based Risk-Adaptive Attribute-Based Access Control (RAd-ABAC) framework [7]. While the foundational work established the feasibility of combining ABAC attributes with dynamic risk scoring and ternary authorization, it operated as an integrated monolithic model with limited methodological transparency and was evaluated on only two datasets [7]. The present work addresses these gaps through three concrete advancements: an explicitly decoupled three-engine architecture (Contextual Attribute Collector, Risk Evaluation Engine, and Decision Management Engine), a deterministic feature-to-attribute transformation protocol with fixed security priority weights ($w_u = 0.30, w_d = 0.30, w_l = 0.20, w_t = 0.20$), and a broader experimental validation across three heterogeneous benchmark datasets (IoT-23, TON_IoT, and IoT Intrusion Detection).

Quantitative benchmarking against the earlier publication confirms measurable performance gains across all primary evaluation metrics:

- Classification Accuracy: Reached 96.84% using Random Forest across 90,000 samples (evaluated on 27,000 unseen test instances), representing an absolute improvement of +2.24 percentage points over the 94.60% achieved by the earlier framework [7].
- Area Under the ROC Curve: Increased from 0.970 to a Macro-Average AUC of 0.992 (+0.022 lift) under One-vs-Rest evaluation.
- Error Rate Suppression: The critical False Positive Rate (FPR) dropped from 4.20% down to 1.85%—a relative reduction of 55.9%—while maintaining a balanced False Negative Rate (FNR) of 1.62%.
- Uncertainty Resolution: Routing ambiguous access requests ($0.35 \leq R < 0.70$) to the intermediate Review state isolated 33.87% of marginal requests for secondary verification, reducing critical authorization boundary errors by more than 90% (only 18 high-risk requests misclassified as Permit and 24 legitimate requests denied outright out of 9,000 samples per class).
- Ablation Validation: Systematic 5-stage ablation (C1 to C5) confirmed the statistical significance ($p < 0.001$) and individual necessity of 4D contextual mapping (+3.38% accuracy), priority weighting (+2.12% accuracy), and ternary logic (+3.19% accuracy and -49.7% FPR).

Despite these improvements, limitations remain: access-control logs were emulated from network intrusion datasets rather than live production environments, and decision thresholds were statically calibrated. Future work will investigate online threshold adaptation via reinforcement learning, lightweight cryptographic token binding for edge-level deployment, and distributed trust auditing using permissioned blockchain ledgers to support next-generation Zero-Trust IoT architectures.

Data Availability Statement

The primary datasets evaluated in this study are publicly available and can be accessed through open scientific repositories:

- The IoT-23 Dataset is hosted by the Stratosphere Laboratory repository (https://www.kaggle.com/datasets/surajsooraj26/iot-23).
- The TON_IoT Dataset is available via the UNSW Canberra Cyber repository (https://www.kaggle.com/datasets/arnobbhowmik/ton-iot-network-dataset).
- The IoT Intrusion Detection Dataset is hosted on Kaggle ([https://www.kaggle.com/datasets/subhajournal/iotintrusion] (https://www.kaggle.com/datasets/subhajournal/iotintrusion)).

Author Contributions

Rashmin Prajapati contributed to the conceptualization, methodology development, data curation, software implementation, formal analysis, and preparation of the original draft. Dr. Sweta S. Panchal provided supervision, validated the methodology, and contributed to manuscript review and editing.

Dr. Neha Soni supported data analysis, validation, visualization, and critically reviewed the manuscript.

Dr. Sandip Kumar R. Panchal was involved in project administration, supervision, validation, and final review and editing of the manuscript.

All authors have read and approved the final version of the manuscript. The corresponding author, Rashmin Prajapati, is responsible for all communication with the journal and ensures the integrity of the work.

Disclosure Statement

The authors declare that there are no conflicts of interest regarding the publication of this paper. The research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

List of Abbreviations

- ABAC – Attribute-Based Access Control
- RAd-ABAC – Risk-Adaptive Attribute-Based Access Control
- RAdAC – Risk-Adaptive Access Control
- RBAC – Role-Based Access Control
- IoT – Internet of Things
- IIoT – Industrial Internet of Things
- ML – Machine Learning
- AI – Artificial Intelligence
- RF – Random Forest
- LR – Logistic Regression
- FPR – False Positive Rate
- FNR – False Negative Rate
- ROC – Receiver Operating Characteristic
- AUC – Area Under Curve
- PR Curve – Precision–Recall Curve
- XAI – Explainable Artificial Intelligence
- MFA – Multi-Factor Authentication
- IP – Internet Protocol

References

[1] Nikhil Sharma, and Prashant Giridhar Shambharkar, "Enhancing Internet of Medical Things Security: A Multi-Layered Approach using Dynamic Adaptive Deep Reinforcement Learning and Blockchain," *Computers and Electrical Engineering*, vol. 129, pp. 1-37, 2026. [CrossRef] [Google Scholar] [Publisher Link]

[2] Ferhat Mecerhed et al., "Robust Attribute-Based Access Control Protocol Over Data-Centric IoT–NDN Networking," *Ad Hoc Networks*, vol. 182, 2026. [CrossRef] [Google Scholar] [Publisher Link]

[3] Osman Abul, and Melike Burakgazi Bilgen, "Jointly Achieving Smart Homes Security and Privacy through Bidirectional Trust," *EURASIP Journal on Information Security*, vol. 2025, no. 1, pp. 1-20, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[4] Zhaoqian Zhang, Di Wu, and Shang Gao, "Attribute-Based Access Control with Credible Outsourcing and Collusion-Resistant Revocation Based on Blockchain for Iomt," *Concurrency and Computation: Practice and Experience*, vol. 37, no. 12-14, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[5] Sujoy Roy, Alok Kumar, and Udai Pratap Rao, "FTBAC: Fuzzy Trust based Access Control for Healthcare Cross-Domain Environment," *Soft Computing*, vol. 29, no. 7, pp. 3349-3366, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[6] Maede Ashouri-Talouki et al., "A Revocable Attribute-based Access Control with Non-Monotonic Access Structure," *Annals of Telecommunications*, vol. 79, no. 11, pp. 833-842, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[7] Rashmin Prajapati et al., "Machine Learning-Based Risk-Adaptive Attribute- Based Access Control Framework for Secure IoT Networks," *International Research Journal of Multidisciplinary Technovation*, vol. 8, no. 3, pp. 495-519, 2026. [CrossRef] [Publisher Link]

[8] Zenghui Yang et al., "An Attribute-based Access Control Scheme using Blockchain Technology for IoT Data Protection," *High-Confidence Computing*, vol. 4, no. 3, pp. 1-10, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[9] Melike Burakgazi Bilgen, Osman Abul, and Kemal Bicakci, "Authentication-Enabled Attribute-based Access Control for Smart Homes," *International Journal of Information Security*, vol. 22, no. 2, pp. 479-495, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[10] Himan Namdari et al., "Enhanced Trust in IoT Environments: Utilizing Perfect Bayesian Equilibrium, Exponential Smoothing, and Machine Learning," *Cluster Computing*, vol. 28, no. 9, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[11] Devesh Srivastava et al., "Auto-Scaling of Cloud Applications Using Machine Learning," *2025 International Conference on Next Generation of Green Information and Emerging Technologies (GIET)*, Gunupur, India, pp. 1-6, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[12] M. Ashwin Shenoy et al., "Self-Healing and Optimization in Mesh Networks Using Glowworm Swarm Optimization," *2025 Second International Conference on Networks and Soft Computing (ICNSoC)*, Vadlamudi, India, pp. 569-574, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[13] Zahid Mahmood et al., "User-Trust Centric Lightweight Access Control for Smart IoT Crowd Sensing Applications in Healthcare Systems," *Personal and Ubiquitous Computing*, vol. 29, no. 1, pp. 31-44, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[14] S. M. Rajesh, and R. Prabha, "ICDAC: Intelligent Contracts Driven Access Control Model for IoT Device Communication," *SN Computer Science*, vol. 5, no. 8, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[15] SooHyun Shin et al., "Architecture for Enhancing Communication Security with RBAC IoT Protocol-Based Microgrids," *Sensors*, vol. 24, no. 18, pp. 1-18, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[16] Vahid Bakhtiary et al., "Combo-Chain: Towards a Hierarchical Attribute-based Access Control System for IoT with Smart Contract and Sharding Technique," *Internet of Things*, vol. 25, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[17] Pooja Choksy et al., "Attribute based Access Control (ABAC) Scheme with a Fully Flexible Delegation Mechanism for IoT Healthcare," *Peer-to-Peer Networking and Applications*, vol. 16, no. 3, pp. 1445-1467, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[18] G. Guna et al., "Multi-Objective Genetic Algorithms for Dynamic Resource Optimization in Cloud Computing," *2025 International Conference on Networks and Cryptology (NETCRYPT)*, New Delhi, India, pp. 876-881, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[19] Pietro Colombo, Elena Ferrari, and Engin Deniz Tümer, "Efficient ABAC based Information Sharing within MQTT Environments under Emergencies," *Computers & Security*, vol. 120, pp. 1-21, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[20] Seyed Farhad Aghili et al., "MLS-ABAC: Efficient Multi-Level Security Attribute-based Access Control Scheme," *Future Generation Computer Systems*, vol. 131, pp. 75-90, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[21] Safwa Ameer, James Benson, and Ravi Sandhu, "An Attribute-based Approach Toward a Secured Smart-Home IoT Access Control and a Comparison with a Role-based Approach," *Information*, vol. 13, no. 2, pp. 1-33, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[22] Elham A. Shammar, Ammar T. Zahary, and Asma A. Al-Shargabi, "An Attribute-Based Access Control Model for Internet of Things Using Hyperledger Fabric Blockchain," *Wireless Communications and Mobile Computing*, vol. 2022, no. 1, pp. 1-25, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[23] Syed Yawar Abbas Zaidi et al., "An Attribute-Based Access Control for IoT Using Blockchain and Smart Contracts," *Sustainability*, vol. 13, no. 19, pp. 1-26, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[24] Zhenghao Xin, Liang Liu, and Gerhard Hancke, "AACS: Attribute-based Access Control Mechanism for Smart Locks," *Symmetry*, vol. 12, no. 6, pp. 1-18, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[25] Meiping Liu et al., "An Efficient Attribute-Based Access Control (ABAC) Policy Retrieval Method based on Attribute and Value Levels in Multimedia Networks," *Sensors*, vol. 20, no. 6, pp. 1-15, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[26] Yan Zhang et al., "An Attribute-Based Collaborative Access Control Scheme Using Blockchain for IoT Devices," *Electronics*, vol. 9, no. 2, pp. 1-22, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[27] I. Sudha et al., "Wireless Sensor Network-Driven Human Intrusion Detection Using RT-DETR for Real-Time Perimeter Protection," *2025 International Conference on Networks and Cryptology (NETCRYPT)*, New Delhi, India, pp. 195-200, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[28] Shengmin Xu et al., "A Secure IoT Cloud Storage System with Fine-Grained Access Control and Decryption Key Exposure Resistance," *Future Generation Computer Systems*, vol. 97, pp. 284-294, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[29] Jingpei Wang et al., "An Access Control Method Against Unauthorized and Noncompliant behaviors of Real-Time Data in Industrial IoT," *IEEE Internet of Things Journal*, vol. 11, no. 1, pp. 708-727, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[30] George Suciu et al., "SAMGRID: Security Authorization and Monitoring Module Based on SealedGRID Platform," *Sensors*, vol. 22, no. 17, pp. 1-16, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[31] Bruno Cremonezi et al., "Improving the Attribute Retrieval on ABAC using Opportunistic Caches for Fog-based IoT Networks," *Computer Networks*, vol. 213, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[32] Rudri Kalaria et al., "Adaptive Context-Aware Access Control for IoT Environments Leveraging Fog Computing," *International Journal of Information Security*, vol. 23, pp. 3089-3107, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[33] Ronghua Xu et al., "BlendCAC: A Smart Contract Enabled Decentralized Capability-Based Access Control Mechanism for the IoT," *Computers*, vol. 7, no. 3, pp. 1-27, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[34] Hany F. Atlam et al., "Risk-Based Access Control Model: A Systematic Literature Review," *Future Internet*, vol. 12, no. 6, pp. 1-23, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[35] Bhaveshkumar Kathiriya et al., "Optimizing Fuzzy Decision Systems with the Jaya Algorithm for Enhanced Data Transfer in Multi-Hop CRNs," *Franklin Open*, vol. 16, pp. 1-10, 2026. [CrossRef] [Google Scholar] [Publisher Link]

[36] Haewon Byeon et al., "An In-Depth Analysis of Security Flaws in Advanced Authentication Protocols for the Internet of Medical Things," *International Journal of Advanced Computer Science & Applications*, vol. 16, no. 7, pp. 426-431, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[37] Zeyad Ghaleb Al-Mekhlaf et al., "A Quantum-Resilient Lattice-Based Security Framework for Internet of Medical things in Healthcare Systems," *Journal of King Saud University Computer and Information Sciences*, vol. 37, no. 6, pp. 1-19, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[38] Jaime Pérez Díaz, and Florina Almenares Mendoza, "Authorization Models for IoT Environments: A Survey," *Internet of Things*, vol. 29, pp. 1-28, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[39] Abubakar Wakili, and Sara Bakkali, "Privacy-Preserving Security of IoT Networks: A Comparative Analysis of Methods and Applications," *Cyber Security and Applications*, vol. 3, pp. 1-15, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[40] G. Gandhimathi et al., "A Use of Fuzzy Logic Based Decision Making tree for Developing Smart Healthcare System," *2024 4th International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, Greater Noida, India, pp. 298-300, 2024. [CrossRef] [Google Scholar] [Publisher Link]