Adaptive Fuzzy Logic Risk- Based Access Control Model for Smart Contract Execution on Block Chain Systems

Omondi, Alan Odhiambo[1], Erick Oteyo Obare[1], Samuel Oonge[1]
[1]*Department of Information Technology, School of Computing and Informatics, Maseno University, Kisumu, Kenya*
Corresponding Author: alanodhiambo7@gmail.com

***Abstract*: Smart contracts contribute to the automation and efficiency of various processes, reducing the need for intermediaries in order to execute agreements on the blockchain platforms. Classical traditional access control models, Discretionary Access Control (DAC), Mandatory Access Control (MAC), and Role-Based Access Control (RBAC) represent conventional access control paradigms which have played a fundamental role in the management of resource access in many organizations and systems. These approaches employ predefined policies and conventions to govern and enforce access permissions. The access models of smart contracts deployed in Blockchain systems exhibit limited adaptability to dynamic changes in the system environment. Conventional crypto, the main access control mechanism encounters a challenge in effectively mitigating the security risks related to identity management authentication across the processes of consensus, initiation, and execution of smart contracts, specifically within Blockchain systems. The shortcoming of classical access control models, which were established in previous times lie in their lack of adaptability and responsiveness in detecting abnormal and malevolent behaviors through the process of observing and tracking user actions during the entirety of their access session. The situation at hand necessitates the implementation of adaptive access control models. The aim of this study was to create a risk-based access control model that is both dynamic and adaptive. To achieve this, the study designed an adaptive model, developed the adaptive risk-based access control model, then tested, implemented and evaluated the developed model. The study adopted a Mixed-Method research design which included Experimental research design that was used to design and develop the model while Action research design was used to test and evaluate the model anchored on within the PiECE framework. A fuzzy logic inference principle was employed to develop, while expert judgment technique was used to evaluate the model through an evaluation metric criterion using regression model analysis. To handle uncertain data ranges, encompassing categories such as severe, high, moderate, and low user risk estimating strategy based on fuzzy logic was employed. Data collection methods utilized the Ai data mining technique route thereafter involved cleaning, pre-processing and annotation of the sample size data sets. The cleaned data was split into training, testing and validating data sets which then empirically, the MATLAB toolkit was used in the development and testing phase of the proposed architecture for execution stages of smart contracts in blockchain platform. Ethical concerns were highlighted based on the pilot model's efficacy. This thesis presented a comprehensive evaluation of a dynamic risk-based access control model that integrates fuzzy logic, expert judgment, and blockchain-enabled smart contract monitoring. It detailed the design and implementation of a fuzzy inference system to address the limitations of static access models as the developed model incorporated expert-defined thresholds for risk estimation. The study's validation was through expert interviews that demonstrated the model's effectiveness via statistical and AI-based methods. Simulation using Simulink and MATLAB was employed to further validate the adaptive risk assessment mechanism. Integration of the model with eXACML ensured policy compliance. Comparative analysis confirmed the model's superiority in providing secure, adaptive access control for blockchain smart contract ecosystem. The attributes of the developed adaptive fuzzy logic access control model make contribution that can be utilized in future design of intelligent access systems that dynamically adjust the capabilities of users' based on their behaviors throughout access sessions to enhance further smart contracts' inherent secured nature.**

***Keywords*: Access control models, Defuzzification, Expert judgment mechanism, Fuzzy logic. Fuzzification, Logical Inference, Fuzzy operators, Fuzzy set, Membership function (MF), Security risk mechanism.**

1. Introduction

This paper develops a dynamic access control model that integrates fuzzy logic and expert judgment to estimate security risks associated with smart contract execution in blockchain systems. The model addresses the limitations of traditional static access control frameworks by dynamically assessing access decisions based on contextual risk factors: user context, resource sensitivity, action severity, and historical risk behavior. A Mamdani fuzzy inference system (FIS) was employed, and 81 fuzzy rules were validated by expert input. Simulation via MATLAB and Simulink demonstrated the model's robustness. Integration with the XACML policy framework and anomaly detection mechanisms enhances the model's real-time adaptability and security assurance. The results validate the model's applicability across domains and suggest future enhancements using AI-based hybrid models.

A. Background of the Study

Smart contract applications demand dynamic access control systems capable of real-time decision-making.

Existing models rely on static policies, limiting adaptability in fluctuating environments. This study develops a fuzzy logic-based access control framework that evaluates risk using contextual inputs, enhancing smart contract security in blockchain ecosystems. The dynamic and decentralized nature of blockchain ecosystems, particularly in smart contract operations, requires robust and adaptable access control mechanisms. Traditional Role-Based Access Control (RBAC) models lack contextual responsiveness and fail to adequately assess risk in real time. To address these limitations, we introduce a fuzzy logic-based model that dynamically evaluates access decisions using multi-dimensional input variables. This paper outlines the model's architecture, development process, and empirical validation through simulations and expert review. The dynamic and decentralized nature of blockchain ecosystems, particularly in smart contract operations, requires robust and adaptable access control mechanisms. Traditional Role-Based Access Control (RBAC) models lack contextual responsiveness and fail to adequately assess risk in real time. To address these limitations, we introduce a fuzzy logic-based model that dynamically evaluates access decisions using multi-dimensional input variables. This paper outlines the model's architecture, development process, and empirical validation through simulations and expert review. This paper proposes an adaptive fuzzy logic-based access control framework for secure smart contract execution within blockchain systems. By incorporating fuzzy rule sets, contextual inputs (user behavior, action criticality, resource sensitivity, and risk history), and integrating Simulink-based monitoring, this model provides a dynamic and responsive mechanism for determining access decisions. Statistical evaluation, anomaly detection, and expert validation confirm its superiority over traditional RBAC models in dynamic environments.

B. Blockchain

A blockchain refers to a decentralized and distributed digital ledger, documenting transactions across numerous computers in a secure and transparent manner (Bankykoom et al.,2018). It comprises a series of blocks, each containing transaction records. Central attributes of blockchain systems encompass decentralization, immutability, transparency, and security. Unlike traditional centralized systems, blockchain is decentralized to operate on a peer-to-peer network of computers (nodes). Each node on the network has a copy of the entire blockchain, and there is no central authority controlling the system (Arslan et al.,2020). This decentralization helps enhance security and resilience.

C. Smart Contracts

Some applications of blockchains, like Ethereum, support smart contracts. Smart contracts are agreements with terms directly coded into them, which automatically execute and enforce these terms when specific conditions are fulfilled (Buterin et al., 2018). Operating on a blockchain, smart contracts facilitate trustless and decentralized automation of processes, eliminating the necessity for intermediaries. Smart contracts are written in programming languages specifically designed for the blockchain platform they run on. For example, Code Execution, Ethereum uses Solidity. The code of a smart contract is deployed to the blockchain. Smart contracts operate on a decentralized blockchain network. The code and execution are distributed across multiple nodes, making the process resistant to censorship or interference from a single party. They automatically execute when predefined conditions specified in the code are met (Bankykoom et al., 2018).

D. Access Control Models

Access control models form an integral part of smart contracts' computing resources, which serve to manage and monitor access within a system. The access control model fall into three primary categories: classical access control models, dynamic access models, and object-based access models. Classical models, such as MAC, and RBAC, rely on predefined rules, while dynamic models like DAC, AAC, and UBAC consider dynamic factors for access decisions. Object-Based Access Control (OBAC) focuses on individual objects, allowing fine-grained control but necessitating complex implementation (Dolgui et al.,2020). Despite their strengths, classical models like MAC, RBAC, and ABAC have limitations, such as lack of centralized control or fine-grained access. Dynamic models are more intricate to manage, involving numerous attributes and policies. Current access models struggle with security concerns in blockchain systems, especially during smart contract execution, as they lack flexibility and sensitivity to abnormal actions (Aitzhan et al., 2016).

E. Fuzzy Logic Technique with Expert Judgement Mechanism

The fundamental principle underlying fuzzy logic involves the application of a predefined set of rules (if-else statements) in parallel to interpret certain values within the input vector and subsequently assign values to the output vector (Porwal et al., 2015). Fuzzy Logic is based on fuzzy sets in that, unlike classical sets, their membership is not a true-false‘ but not-quite-true-or-false‘ answer (Mathworks, 2021). The classical adaptive access control model relies on Boolean or crisp sets, where membership in a set is determined by a characteristic function that assigns a value of either 1 (true) or 0 (false) to each individual in the universal set X. A Fuzzy Membership Function (FMF) is a curve defining how each point in the input space is mapped to a membership value between 0 and 1. The choice of FMF depends on the application domain, considering factors like simplicity, convenience, speed, and efficiency. FMFs can be based on functions such as piecewise linear, Gaussian distribution, sigmoid curve, quadratic and cubic polynomial curves. Gaussian and sigmoidal functions, which are S-shaped and open to the right, are suitable for modelling access control in adaptive risk-based models. They are proven to be appropriate for linguistic variables and are supported in the MATLAB Fuzzy Logic Toolbox (Mathworks, 2021).

Expert judgment is a potent tool in risk analysis, offering diverse solutions and decisions across various domains, including psychology, criminal justice, financial forecasting, political science, and decision analysis, where expert judgment stands as the primary source of valuable information. When practical data is insufficient to describe the probability and impact of a specific incident, expert judgment becomes a valuable approach, providing a subjective evaluation based on the expert's experience and insights gained through careful group focus interviews. Estimating the probability of an incident in a risk analysis, especially for rare and extreme events, is a challenging task, given the inherent uncertainty (Walters et al., 2021). Expert judgment involves expressing inferential opinions derived from knowledge and experience (Yin et al., 2016). It is frequently employed to assess uncertain parameters in a probabilistic manner and evaluate various components of a model

While smart contracts offer numerous advantages, they also face several challenges that need to be addressed for broader adoption and improved functionality. Security is a critical challenge associated with executing smart contracts on a blockchain platform. Vulnerabilities and bugs in the smart contract's code could lead to exploits hence the need to conduct code audits, formal verification, and rigorous testing which are essential to mitigate these risks. The immutability of smart contracts, while a strength in terms of trust, becomes a challenge if there are bugs or vulnerabilities in the deployed code (Aitzhan et al., 2016). Once deployed, fixing such issues is difficult, and it requires careful consideration during the development phase (Ruddick et al.,2018). Many blockchain networks, especially those with high transaction volumes like Ethereum, face scalability challenges. As the number of transactions and smart contracts increases, the network may experience congestion and slower transaction processing times. Smart contracts on one blockchain may not be directly compatible or interoperable with those on another blockchain. This lack of standardization can hinder collaboration and limit the potential for integrated applications across different platforms (Huang et al.,2020).

F. Problem Statement

Smart contract utilization has brought unlimited benefits, but at the same time raises several security issues. This is because current access control models with rigid, inflexible and static structure with predefined rules that always give the same result in different situations fail to provide the required level or degree of security for such execution rendering system. The main gap lies in the lack of dynamism and adaptability within existing smart contract blockchain access models, which predominantly use cryptography as their classical access control mechanism. These traditional approaches are insufficient for detecting malicious actions or protecting system resources once access is granted. Classical access control approaches do not provide a way to detect malicious actions and protect system resources after granting the access. Consequently, if an abnormal action is detected, user privileges cannot be appropriately reduced, nor can the access session be effectively terminated. The risk estimation module used in dynamic access control model has no flexibility to adjusts a user's permission adaptively depending on user's behaviour in active access sessions, such that if an abnormal action is discovered, user privileges will be reduced to some degree or the access session will be terminated.

G. General Objective and Specific Objective

The objective of this study was to create an adaptive risk-based access control model for smart contract execution on block chain using fuzzy logic technique with expert judgment mechanism.

Specific objective of the study was to

1) Design an adaptive risk-based access control model.
2) Develop the adaptive risk-based access control model.
3) Test the adaptive risk-based access control model.
4) Evaluate the adaptability and dynamism of the created model.

H. Research Questions

What are the underlying design challenges of the existing adaptive models?

Will the developed model be adaptive and dynamic for risk-based access control that uses real-time and contextual information to determine the access decision?

Will the tested adaptive risk-based access control model provide dynamism on execution of smart contracts?

Will the evaluation of the adaptive risk-based access control model address the existing gap in execution of smart contract on block chain?

I. Scope of the Study

The focus of the study was to create an adaptive and dynamic risk-based access control model that uses real-time and contextual information to determine the access decision in freight and logistics business and legal environment.

2. Literature Review

A. Theoretical Framework: Exploratory Review on Risk-Based Estimation Paradigms

1) Game Theory

The game theory modelling process involves the decision-maker interacting with players, understanding their strategic decisions, and observing their preferences and responses. A game theory comprises four essential components: players, strategies, payoffs, and information. Players are the decision-makers within the game, and strategies represent the plans they employ in response to the moves of other players. Selecting suitable tactics is critical for players. Payoffs denote the rewards players receive in the game, influenced by both their actions and those of other players (Binmore et al.,2015). However, a critique of the game theory paradigm notes that risk analysis is based on user benefits rather than probability. Additionally, game theory is recommended in situations where

practical data is lacking, and it becomes complex, especially with more than two players. The use of mixed strategies can lead to random outcomes, making it less adaptable in smart contract identification management.

2) Decision tree

The decision tree is a widely used methodology in various machine learning operations, functioning as a decision support tool that generates decisions based on a set of rules organized in a tree structure. Constructing a decision tree model involves the partitioning of data into training and validation sets. Training data are employed to derive the necessary rules for the tree, while validation data are utilized to assess the tree and implement required modifications. Represented as a flow diagram, the decision tree features nodes, represented by rectangles, each describing the probability and impact of a risk. These rectangles are interconnected by arrows, with each arrow leading to another box indicating the percentage probability.

Despite its advantages, the decision tree model comes with certain limitations. One such limitation is its scalability, where an increase in the scale of the tree can make the resulting model challenging to interpret, requiring additional data for rule validation. Additionally, the decision tree model is based on expectations, making it difficult to plan for all contingencies that may arise from a decision (Santos et al., 2019).

3) Subjective Risk Assessment

Subjective risk assessment is employed to examine potential damages associated with a specific scenario. It can be defined as the systematic investigation of potential security breaches to a system and the resulting losses, utilizing a combination of available information about the situation and informed judgment regarding unknown information. The purpose of subjective risk assessment is to recognize the context of risk and establish acceptable risk values for each situation, achieved through comparisons with similar risks in analogous scenarios. While an effective risk assessment offers numerous benefits, such as providing a well-founded basis for preventing or minimizing the impact of risk, it is a subjective process influenced by experience and is only valid at a specific point in time, limiting its adaptability in validating user identity in smart contract executions (Cazzola et al.,2018).

4) Fuzzy Logic Inference Data System Technique

A fuzzy logic inference data system is a computational approach that simulates human thinking by describing the world in imprecise terms. Unlike computers that operate only on precise evaluations, the human brain can engage in reasoning with uncertainties and judgments. The fuzzy logic system is a precise problem-solving approach capable of working with both numerical data and linguistic knowledge simultaneously. It simplifies the management of complex systems without the need for a mathematical description (Atlam et al.,2021).

The computation process using the fuzzy logic system comprises three main phases:

Fuzzification – This phase converts crisp or classical variables of input and output into fuzzy variables to process and produce the desired output. Most variables are initially crisp, and fuzzification is used to handle imprecise information.

Fuzzy Inference Process – This phase involves building IF-THEN fuzzy rules to describe relationships between different inputs and output. Linguistic variables are used to represent conditions and outputs, creating rules that guide the fuzzy logic system in processing input data.

Defuzzification – This phase converts the fuzzy output back to a crisp output since the final result needs to be a precise variable.

Fuzzy sets, incorporated in a black-box approach, excel at handling complex mathematical equations and formulas, making them applicable in various computing modelling applications. Fuzzy logic, conceptualized by Zadeh (1995) provides a convenient way to map an input space to an output space, offering advantages such as modelling imprecise multivariate data and nonlinear functions of arbitrary complexity, based on natural language.

The general concept behind fuzzy logic involves applying a set of pre-defined rules (if-else statements) in parallel to interpret values in the input vector and assign values to the output vector. Fuzzy logic is based on fuzzy sets, where membership is not a simple true-false answer but a not-quite-true-or-false response within the unit interval [0,1]. Fuzzy logic has flexibility, robustness, and ease of understanding due to its basis in natural language. However, it requires domain experts to create accurate rules and involves more tests and simulations, which can be time-consuming, especially with an increasing number of rules (Bai et al.,2016).

B. Conceptual Framework

A conceptual framework is a diagrammatic representation of how variables interact. It provides a clear concept of the areas in which meaningful relationship are likely to exist (Zadeh et al.,2015). Smart contracts have expanded to include multiple applications and services. It is a dynamic and distributed system which creates several issues that need be taken into accounts when building an access control model that has the element of dynamism and adaptability. Figure 2.1 presents the conceptual framework of a smart contact with the adaptiveness and dynamism that is executed on blockchain.

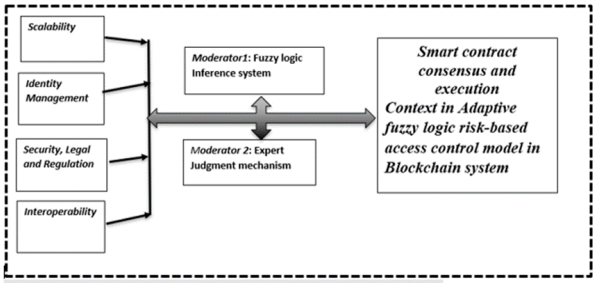


Fig. 1. Conceptual framework for access control in smart contracts (Odhiambo et al., 2024)

Figure 1 shows scalability, interoperability, security, legal

and regulations, and identity management as the independent variable in this study. Smart contract consensus and execution context as the depended variable. Expert judgment with fuzzy logic inference parameters are infused to contextualise the variables from initialization to execution of smart contract in a blockchain platform.

The study investigated these variables on how they relate and interact to determine a secure access and execution of smart contract to provide adaptability and dynamism on blockchain platform as documented herein:

1) Interoperability

Ensuring interoperability is essential for the execution of smart contracts on a blockchain platform, facilitating cross-platform execution, streamlining inter-chain communication, integrating external data sources via systems like oracles, and allowing execution on external applications running on legacy systems. Another crucial aspect influencing an access control model is the formulation of access policies. These policies need to be designed to accommodate multiple users and organizations. While each organization can establish its unique policies, there is a simultaneous need to adhere to the policies set by other organizations.

2) Scalability

The smart contract system encompasses billions of devices, generating an extensive volume of data that necessitates substantial processing capabilities. Designing an access model for smart contracts integrated within the blockchain must account for the expanding network size. Scalability, defined as the system's ability to manage growing workloads and accommodate expansion without compromising performance, is a critical consideration (Sharma et al.,2019). To address scalability challenges, various intervening and constraining variables, such as layer solutions, the implementation of sharding techniques involving multiple subsets of data execution portfolios, off-chain computation of data, and optimized consensus dynamic interactions within the smart contract environment, can alter the dynamics of access requirements between users. These adjustments enable the access policies to adapt to diverse situations and changing conditions while making access decisions within the system (Sharma et al.,2019).

3) Identity Management

Given that a smart contract is self-executing based on the terms outlined in coded agreements, the authentication of the digital identities of involved parties can be achieved through public-private key encryption, biometric authentication, or other digital signature methods. These variables play a crucial role in managing access and delegating authority attributes. In specific access scenarios, there is a need for Ai agents to operate on behalf of users for defined periods. Therefore, an access model should consider the delegation of authority to enhance usability and flexibility. Context awareness, defined by the Cambridge dictionary as the situation within which something happens, is an essential factor when constructing an access control model. Incorporating context awareness enables user interactions, making it imperative to consider real-time contextual information when making access decisions (Bancor et al., 2018).

4) Security, Legal and Regulations

Security is a crucial element in a self-executing contract, as once modified and deployed, the contract becomes binding and cannot be altered within the blockchain system. This inflexibility means that vulnerabilities could potentially be maliciously exploited by users once granted access. Granting user access alone is insufficient for the execution of smart contracts; an adaptive access model should be auditable. Therefore, it is essential to collect and store necessary evidence of various access operations. Legal and regulatory considerations become variable factors when publishing a new node on a smart contract and are still evolving within the established legal framework (Aitzhan et al., 2016). This evolving trend emphasizes the need for users to be acquainted with the legal jurisdiction environment in which the smart contract execution takes place especially for our case on freight and logistic business environment. An access control model for smart contracts, catering to billions of users with diverse security, legal, and regulatory awareness, must provide suitable interfaces to meet the varied needs of users.

5) Smart Contract Consensus and Execution Context

In a blockchain system that ensures secure access and the execution of smart contracts, it is crucial to establish a validation process through an algorithm such as proof of work or proof of stake. This consensus algorithm ensures agreement among all nodes on the network regarding the nature and state of the contract to be executed. Another essential component is the execution context of the smart contract, where the actual processing of contract terms, such as the transfer of digital assets and the update of records, occurs within lines of code (Gupta et al.,2018).

The blockchain system is open to everyone, allowing anyone to validate and audit transactions. Individuals utilize blockchain technologies to create various applications of their choosing. This type of database exists across different computer systems, forming a peer-to-peer network, eliminating the presence of a single, centralized database or server. However, this decentralized structure tends to increase network broadcasting (Wang et al.,2020). To address this, the selection of primary nodes becomes crucial, initializing the use of digital signatures with public key cryptography.

In the transmission of desired transactions through nodes, a P2P network is involved. Through a recognized algorithm, the node network validates identity and user status. Subsequently, a new block is added to the existing blockchain, containing a hash, verified proof of valid transactions with a timestamp, and the hash of the previous block. This prevents the block from being altered or a block being inserted between two existing blocks (Cheng et al.,2016). Smart contracts executed based on certain conditions can be written into the platform, applicable only to permissioned blockchains with a high level of trust. After solving the proof of work (PoW) puzzle, the block is

broadcast to other nodes, detecting vulnerabilities and virtually preventing attacks from intruding. Examples of PoW include Bitcoin, Kovan testnet, and Ethereum. The main goal was to develop a less computational but adaptive risk-based access control model than PoW with better dynamic and robust access security guarantees.

C. Summary of Literature

Smart contracts have captured the attention of experts, specialists, and researchers in both academia and industry due to their potential to revolutionize daily life activities (Ruddick et al., 2016). While their utilization brings numerous benefits, it also introduces various security challenges. The current access control models, characterized by rigid and static structures with predefined rules yielding consistent results across different situations, fall short in providing the necessary security for such execution-rendering systems. In response, this study presents an adaptive and dynamic risk-based access control model. The proposed model leverages real-time and contextual information from smart contracts associated with access requests to autonomously determine access decisions.

User attributes collected during access requests, data sensitivity, action severity, and user risk history serve as inputs to estimate the risk value for each access request in the proposed model. To enhance abnormality detection capabilities, smart contracts monitor user activities throughout the access session, detecting and preventing malicious attacks from authorized users. Recognizing the pivotal role of selecting an optimal risk estimation technique in building a risk-based model, the study discussed common risk estimation taxonomies used in related models. This section of the paper encompassed the theoretical framework, a critique of literature-research gaps related to the adaptability of existing models. A conceptual framework was developed as it sought to explore the design and creation of the adaptive model from an architectural perspective.

The limitations of Role-Based (RBAC), Attribute-Based (ABAC), and traditional cryptographic controls underscore the need for adaptive solutions. Fuzzy logic enables linguistic risk assessments, offering nuanced decisions. Expert judgment compensates for unavailable datasets, especially during cold start scenarios, a recognized challenge in adaptive RBAC systems. This paper proposes an adaptive fuzzy logic-based access control framework for secure smart contract execution within blockchain systems. By incorporating fuzzy rule sets, contextual inputs (user behavior, action criticality, resource sensitivity, and risk history), and integrating Simulink-based monitoring, this model provides a dynamic and responsive mechanism for determining access decisions. Statistical evaluation, anomaly detection, and expert validation confirm its superiority over traditional RBAC models in dynamic environments

3. Research Methodology

A. Research Design

The research adopted a Mixed-Method research design which included Experimental research design that was used to design and develop the model while Action research design was used to test and evaluate the model. Therefore, the study considered a Mixed-methods research design that integrated qualitative and quantitative data collection and analysis techniques. In Experimental research design the study conducted experiments where different risk levels were simulated, and the response of the adaptive model observed. Quasi-experimental designs were inter-looped in-order to manipulate variables and measured of outcomes. The performance of the model was experimented against traditional access models. Action research design which was employed in the context of testing the model involved iterative cycles of design, testing, evaluation, and refinement based on feedback from stakeholders. Table 3.1 shows the input risk assessment factors that provided action feedback.

1) The Experimental Model Set-Up

Experimental research design was used to design and develop the model by manipulating variables in a controlled environment. By employing experimental research design, conducted within a controlled environment, provided empirical evidence of access models' effectiveness, usability, and performance under varying risk conditions. This approach was used to identify strengths, weaknesses, and areas for improvement. Experimental setup procedures involved the design and configuration of a test environment that simulates real-world scenarios and conditions relevant to the evaluation objectives. We used the risk assessment factors identified in

Table 1
Risk assessment factors, access control policies, and adaptation mechanisms tested and evaluated in controlled environment

Category	Description	Evaluation Details
Variable Manipulation	Adjustments to core system parameters.	- Adjusted risk thresholds to test model sensitivity. - Modified access control policies based on assessed risk. - Tested various adaptation strategies (e.g., dynamic policy tuning).
Controlled Environment	Experimental isolation to ensure valid results.	- Controlled for extraneous variables. - Simulated network environment with defined risk factors, user behaviors, and configurations.
Random Assignment	Bias minimization through random grouping.	- Participants (e.g., system administrators) randomly assigned to groups using either the adaptive model or traditional access control.
Outcome Measurement	Evaluation of system effectiveness and usability.	- Security Effectiveness: Number of incidents or unauthorized attempts. - User Satisfaction: Feedback from users and administrators. - System Performance: Metrics like response time and resource utilization.
Statistical Analysis	Quantitative assessment of results.	- Statistical methods: t-tests, ANOVA, and regression analysis. - Compared outcomes across groups to identify significant differences.

(Odhiambo et al., 2024)

table 1 for testing access control policies and adaptation mechanisms to be tested and evaluated.

The experimental setup involved the iterative execution in three phases;

Pilot Experiments:

Pilot experiments provided preliminary information. Pilot experiments were conducted to gain insights into the behavior of various elements and components. It also allowed for a systematic and participatory investigation, ensuring that the adaptive model was refined and validated through multiple cycles of experimentation and stakeholder evaluation.

Exploratory Experiments was employed to investigate response patterns to parameter variations or interventions within the adaptive risk-based access control model. It aimed to generate hypotheses for subsequent formal testing in confirmatory experiments.

Confirmatory Experiments was conducted to rigorously test and validate the sample data sets confirming the hypotheses established prior to the initiation of all experiments.

The Evaluation of stakeholders, was implemented immediately after the completion of the Pilot and Exploratory Experiments. Path 4b allowed a return to the regular PiECEs cycle, ensuring the resumption of Exploratory Experiments interrupted by path 2b.

The exploratory experimentation was recursively carried out until all issues were resolved as illustrated in figure 3.1. In developing all the various system modules exploratory experiments involved:

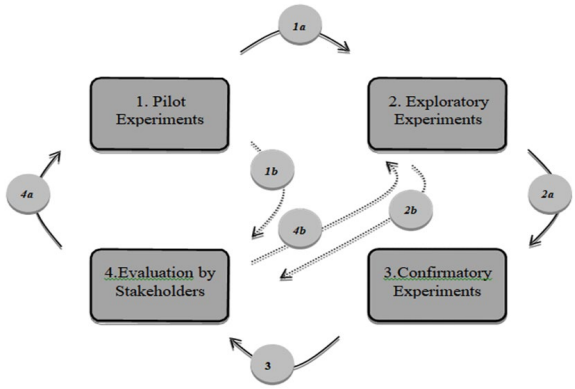


Fig. 2. The phases of PiECEs experiments model (Franklin et al., 2012)

The design of computer programs (code that implements with the fuzzy logic set modules) using fuzzy sets in the MATLAB toolkit;

System Development:

User (Evaluation by Stakeholders-freight business enteprises) Testing;

Model evaluation to adjust various variables in the system program code.

B. Data Collection and Preparation Methods

In order to design, develop and test the model the study collected data ensuring that data collection procedures were consistent, reliable, and aligned with study objectives. Quantitative data was collected through experiments and simulations, while qualitative data was gathered through focus groups (for expert judgment), and observations (output data).

Table 2
Data source and mining techniques assigned to projected mined data sets

DataMining Technique	Example Methods	Assigned Datasets	Data Sources
Classification (AI)	Decision Trees, SVM, Naive Bayes, KNN	- Authentication logs - Security incident reports - User/resource profiles - Historical access data	- GitHub Repos - Academic Databases
Clustering (Machine Learning)	K-Means, Hierarchical Clustering, DBSCAN	- User behavior patterns - Access frequency - Usage profiles	- GitHub Repos - Academic Databases
Regression Analysis (ML)	Linear, Polynomial, Logistic Regression	- Predictive trends - Access load patterns - Transaction volumes	- Blockchain Explorers (Etherscan, Blockchain.com) - CoinMarketCap - Cloud Service Logs
Anomaly Detection (ML)	Isolation Forest, One-Class SVM	- Outlier detection in access logs - Fraud detection - Intrusion attempts	- Security Logs - Network Monitoring Tools - Public Blockchain Data
Deep Learning	ANN, CNN, RNN	- Sequential access behavior - Complex transaction patterns - Sentiment & regulatory trend analysis	- The Graph - Social Media APIs (Twitter, Reddit) - Regulatory Data (SEC, FATF)

(Odhiambo et al., 2024)

Table 3

Category	Sources	Data Types
i) User Behavior Data	- GitHub blockchain project repositories - Academic databases	- User profiles and activity logs - Behavioral biometrics (e.g., typing, mouse movements) - Geolocation - Login patterns and access times
ii) Security Incident Data	- GitHub security logs - Academic studies on blockchain security	- Historical breaches and fraud attempts - Known malicious addresses - Attack types (phishing, malware, DoS) - Response times and outcomes
iii) Blockchain Transaction Data	- Etherscan (Ethereum) - Blockchain.com (Bitcoin, etc.) - The Graph (smart contract queries)	- Transaction details (sender, receiver, amount, timestamp) - Smart contract logs - Transaction frequency and volume - Success/failure patterns
iv) Network Data	- Network tools (Wireshark, Splunk) - Cloud platforms (AWS, GCP, Azure)	- Latency and bandwidth - Server load and performance - System uptime/downtime
v) External & Legal Data	- CoinMarketCap, CoinGecko (market data) - Twitter API, Reddit API (sentiment) - SEC, FATF (regulatory data)	- Market trends and price changes - Regulatory changes and compliance - News and sentiment from social/media platforms

The establishment of an effective security system, which dynamically adapts access permissions based on evolving risks and user behaviors, relied significantly on the meticulous collection and preparation of data. Some of the methods employed in collection of data from diverse sources, included the establishment of data pipelines, integration with Security Information and Event Management (SIEM) systems, and the utilization of log aggregators. The procedures entailed;

Coding: The process of assigning codes to specific variables or elements within the data to facilitate organized analysis and interpretation. For the coding process, sub-codes were derived from research objectives, questions, the research context, theoretical constructs, and the conceptual framework. The coding scheme remained flexible to accommodate emerging sub-codes from input data, and operational definitions were updated accordingly. The codebook provided guidance to the first-order coding, employing a descriptive coding technique. To address construct validity, multiple data sources, including smart contracts, documentation, harsh blocks, and board reports, were utilized to ensure convergence findings. Reliability was maintained through programmatically retrieving and storing analyzed transactional stages locally, maintaining a qualitative codebook of codes, and developing matrices from labelled data blocks.

Observation of Computed Output: Actively observing and analyzing the computed output generated by the adaptive risk-based access control model.

MATLab Simulations: Leveraging MATLab simulations to model and simulate various scenarios within the adaptive model for experimental analysis.

1) Data Sources

Data mining techniques were transacted through Artificial intelligence (Ai) route to retrieve seven (7) data sets which included Authentication logs, Security incident reports, User profiles and attributes, Resource attributes, Historical access data, Security event logs and Threat intelligence feeds. This involved the use of methodologies and algorithms to extract meaningful patterns, trends, and insights from large datasets using the common data mining techniques which included:

These techniques were applied individually or in combination, depending on the nature of the dataset and the specific objectives of the data mining task. These techniques were employed to uncover hidden patterns, relationships, and knowledge from structured, semi-structured, and unstructured data for the model. The target population for the study consisted of about 350 key data server repositories/sources.

The data to be collected as presented in table 3.2 included:

Authentication logs, Security incident reports, User profiles and attributes, Resource attributes, Historical access data, Security event logs and Threat intelligence feeds using data mining technique that linked up with the variables as was presented in the conceptual framework.

The table 4 shows a caption of such repositories as captured with the presumed output for each data set. Data sets were clustered into scalability, interoperability, security and identity management sources out from which the relevant input data type were collected from the sample size.

2) The Cleaning Pre-Processing and Annotation of the Sample Size Data Sets

During this phase, meticulous data preparation became paramount. The data cleaning process involved actively addressing various tasks to ensure the quality and integrity of the data. This encompassed tasks such as handling missing data through imputation or removal, eliminating duplicates to maintain data consistency, and conducting necessary data transformations. Transformations included annotating date/time conversions and applying one-hot encoding for categorical variables. The annotation of sampled data sets significantly contributed to risk assessment, incorporating factors such as user access frequency, behavior patterns, and resource sensitivity scores.

Table 4
Relevant input data types from the mined data sets

Variable Type	Input Data Types	Associated Aspects
Independent Variables	- Resource attributes - Historical access data - User profiles & attributes - Authentication logs - Incident reports - Threat intelligence feeds	- Scalability - Interoperability - Identity Management - Security and Regulation
Intervening Variables	- Smart contract vulnerabilities - Contract design/code - Third-party services - Transaction history - Data encryption - Transaction security - Incident response - Regulatory compliance - Risk tolerance	- Access Control - Blockchain Network Security

(Odhiambo et al., 2024)

Furthermore, data labels were assigned, and risk levels defined as low, moderate, or high. Historical data were semi-automated labelled accordingly, facilitating model training. To uphold privacy and adhere to data protection regulations, sensitive data, such as user personally identifiable information, were anonymized or encrypted. Numerical features underwent scaling or transformation to ensure uniformity across features, ensuring compatibility with the adaptive risk-based model. In order to ensure high-quality data for accurate risk assessment within the adaptive risk-based access control model, attention was devoted to addressing issues such as missing values, duplicates, and inconsistencies.

3) The Splitting of the Data Sets

The datasets underwent division into training sets (75%), validation sets (15%), and test sets (10%) to facilitate comprehensive model development, evaluation, and validation. If an imbalance was detected within the dataset, techniques such as oversampling or under sampling was applied to ensure a balanced representation of risk levels throughout the validation and test phases.

The selection of the appropriate Membership Functions (MFs) relied on the available dataset. The comparison of results between training data and real data, along with the calculation

of error values using Mean Average Percentage Error (MAPE), guided the selection process. MF techniques, including trapezoidal, Gaussian, triangular, sigmoidal, and bell-shaped waveforms, were at the study's disposal howbeit, trapezoidal MF, efficient in representing expert knowledge and streamlining the calculation process, was used to depict input and singleton centroid output fuzzy set in the created adaptive fuzzy logic risk-based access control model.

The testing phase involved 15% of the sample datasets defined criteria for how output risk changed concerning input risk factors. This was achieved through fuzzy rules acting as the knowledge base of the fuzzy logic system, utilizing IF-THEN statements to describe actions or outputs based on specific input combinations. The accuracy and efficiency of fuzzy rules was ensured by considering different risk factors and their combined behavior in producing output risk through machine learning algorithm.

In the validation of the remaining 10% of datasets, security experts contributed by providing appropriate fuzzy rules based on their knowledge and experience. During testing, defuzzification was employed to convert fuzzy variables into crisp variables. This process involved using defuzzification methods such as mean of maximum, center of area (centroid), modified center of area, height method, center of sum, and center of maximum. These tests aimed to ensure data accuracy and evaluate the performance of the adaptive risk-based access control model.

4) Data Manipulation Using the MATLAB Fuzzy Logic Toolkit

The MATLAB Fuzzy Logic Toolkit was applied to model freight &logistics business knowledge for secure identification management and the execution of smart contracts within the model. Fuzzification of inputs involved determining the degree to which they belong to appropriate fuzzy sets through membership functions, converting classical logic into fuzzy linguistic variables. In this stage, risk factors were transformed into linguistic variables, making them easily understandable. Three fuzzy sets, namely Low, Moderate, and High, represented action severity, user context, risk history and resource sensitivity fuzzy sets. For the output, five fuzzy sets—Negligible, Low, Moderate, High, and Unacceptable High—were employed.

By employing the fuzzy logic technique, subjectivity was reduced to an acceptable level. Quantitative input data allowed subjectivity to shift to the rule creation process, providing better control. While subjectivity cannot be entirely eliminated, expert judgment became a significant information source in decision-making operations, especially in risk analysis during the smart contract execution phase. Correct numerical data describing incident frequencies and their impact are often unavailable in most risk-based models (Ruddick et al.,2018). In cases where quantifying risk value using classical approaches is complicated, expert judgment offered a correct risk value for a specific scenario, particularly when appropriate experts are involved.

Expert judgment was sought through focus-group discussions with individuals possessing deep knowledge and expertise in system security to test the risk estimation process. In summary, the five stages were employed to implement the fuzzy logic system with expert judgment for estimating security risks in the adaptive risk-based access control model using the MATLAB Fuzzy Logic Toolkit.

5) Processing of Data Sets Using the MATLAB

a. Fuzzy Logic Design Technique with Logical Operations

This phase focused on applying logical operations to the fuzzified data to support nuanced access control decisions. The model introduced the User Permission Index (UPI), categorized as follows:

- *Full Access*: UPI ≤ 0.2
- *Limited Access with Monitoring*: 0.2 < UPI ≤ 0.7
- *No Access*: 0.8 < UPI ≤ 1.0

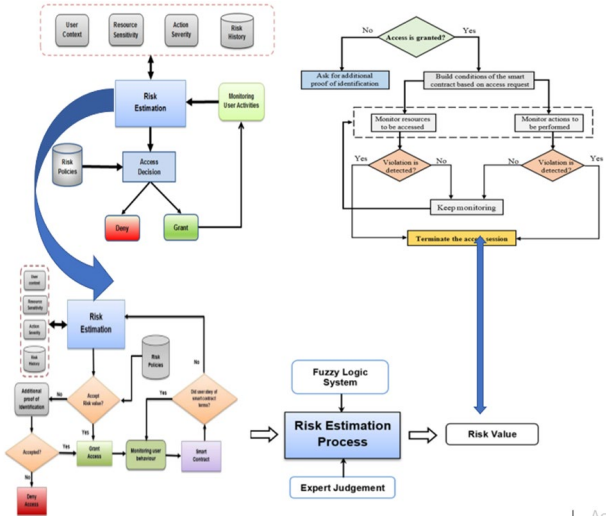


Fig. 3. Adaptive risk-based access control architecture model-Full module

This classification allowed values to belong to more than one fuzzy set simultaneously, enabling multi-level membership. For instance, a UPI of 0.2 may belong to both the "Full Access" and "Limited Access" sets, while a value of 0.4 could be considered part of both the "Limited Access" and "No Access" categories. This fuzzy logic approach supports more flexible and realistic decision-making compared to traditional binary logic.

Predefined if-else rules were employed to interpret these fuzzified values and derive corresponding output risk scores. This logical framework allowed the model to dynamically assess and adapt access permissions based on real-time input data, thereby enhancing the precision and adaptability of access control in blockchain-based environments.

6) Empirical Framework for the Developed Model

Figure 3 illustrates the proposed architectural design of the adaptive risk-based access control model for blockchain-based smart contract systems. This model actively monitors user behavior during access sessions. In this architecture, the blockchain functions as a software connector, featuring a

complex modular structure. It supports various configurations and integrates multiple variables previously discussed, which are crucial for identifying user attributes both before and after access is granted.

This study devised and integrated four distinct system modules to construct the adaptive risk-based access control model. The development process involved continuous collaboration between developers and end users to ensure the system aligned with predefined requirements and specifications. This collaborative approach allowed developers to enhance their understanding of the system's technical components and overall feasibility.

The blockchain modules were programmed using the latest Waspmote Integrated Development Environment (IDE). The implementation involved uploading and executing code via the MATLAB toolkit, supported by the web-based Waspmote IDE provided by Libelium.

The Fuzzy Freight and Logistics Business Smart Contract Knowledge System was developed in MATLAB to simulate various scenarios in the smart contract execution phase. Regular updates and maintenance are planned to maintain the system's effectiveness in addressing evolving security risks. Exploratory data analysis and data visualization techniques were employed to uncover patterns and insights crucial for informed decision-making.

The modeling approach adopted combines machine learning algorithms and rule-based systems, selected based on the complexity of the operating environment and the availability of relevant data. The adaptive risk-based model is continuously trained on historical datasets labeled with corresponding risk levels. The data was divided into training, testing, and validation subsets. Additionally, adaptive learning mechanisms were incorporated to ensure real-time model updates in response to dynamic risk factors and data patterns.

Monitoring access patterns and security events plays a critical role in detecting shifts in risk profiles. Rigorous testing and validation were conducted under various operational scenarios to identify potential vulnerabilities and mitigate false positives or negatives. A stakeholder feedback loop was established to evaluate model performance and guide iterative improvements. Incident response protocols were also developed to address security breaches detected by the model. The system will undergo periodic reviews to incorporate new data sources, advanced technologies, and updated security best practices.

A Mamdani-type Fuzzy Inference System (FIS) was chosen for its intuitive, rule-based approach. Risk factors were categorized into fuzzy sets (Low, Medium, High), while the output risk was classified into five levels: Negligible, Low, Moderate, High, and Unacceptable High. A series of expert interviews (n=10) informed the creation of fuzzy rules and validated decision thresholds.

The model integrates fuzzy logic inference with contextual input processing to produce dynamic risk evaluations. It comprises four primary input factors: User Context, Action Severity, Resource Sensitivity, and Risk History. Each input was fuzzified using trapezoidal or triangular membership functions. A robust set of fuzzy rules—e.g., *IF user context is High AND action is Critical THEN Risk is High*—guides the inference engine.

These rules were validated and refined with input from domain experts to ensure practical applicability. The final model was constructed and simulated using MATLAB's Fuzzy Logic Designer and underwent rigorous testing across both static and dynamic scenarios.

C. Evaluating the Adaptive Risk-Based Access Control Model

Research design plays a crucial role in evaluating an adaptive risk-based access control model by providing a systematic framework for collecting and analyzing data to assess its effectiveness, usability, and performance. Selection of evaluation metrics (EM) and key performance indicators (KPIs) that align with the objectives of the evaluation of the adaptive risk based access control model was limited to as captured in the table 5.

The study conducted data analysis using appropriate statistical and qualitative analysis techniques to derive meaningful insights and conclusions. Quantitative analysis involved using Simulink simulation. descriptive statistics, inferential statistics, regression analysis, and correlation analysis to examine relationships between variables and identify significant findings. Qualitative analysis involved thematic analysis, content analysis, and interpretation of qualitative data to identify patterns, themes, and emerging insights related to user experiences and perceptions.

Table 5
Evaluation metrics and key performance indicators

Evaluation Metric (EM)	Key Performance Indicators (KPIs)	Specific Parameters	Evaluation Design Methods
Security Effectiveness	- Number of security incidents detected - Successful/unsuccessful access attempts - Detection and response time to threats	- Low level - Moderate level - High level	Quantitative Methods: - Surveys - Experiments - MATLAB simulations - Log/performance analysis Qualitative Methods: - Focus group interviews - Usability testing - Observations
Adaptability	- Dynamic adjustment of access control decisions - Frequency of adaptation - Accuracy of risk assessment	- Flexible - Rigid	Same as above

(Odhiambo et al.,2024)

1) Evaluation of Developed Model's Simulink Block Interface

Simulink is a graphical environment to model, simulate, and analyse multi-domain dynamic systems. It is primarily based on hierarchical data flow diagrams. A Simulink diagram consists of functional blocks connected by signals (wires). These blocks

represent transformations of data, while the signals represent the flow of data between blocks. Each block contains input and output ports to connect with other blocks and transfer signals between blocks. The input ports provide data to the block, while the output ports provide the results computed by the blocks (Aung, 2007; Boström et al., 2010).

After granting the access, a smart contract will be created for each access request. The access permissions will be implemented as the policy in the smart contract dictates. Then, the monitoring module will compare the user behaviour with the terms and conditions of the contract to detect abnormal actions throughout access sessions. The requesting user first defines the data or resource to be accessed and action to be performed in the access request. Then, if the access is granted, a smart contract will be created to implement user's permissions as conditions or terms to guarantee that the user has the ability to access only resources and perform actions that were requested. Then, resources and actions will be monitored to detect violations. if a violation is detected, the system will issue a warning, or the session will be terminated. If no violations are detected, the system will keep monitoring the user behaviour throughout the access session.

2) Evaluation using Regression model

The model outcomes subsequently was evaluated through comparative analysis with the outputs of existing models, and this comparison was illustrated through graphical triangulation simulations. A multiple regression model was deployed to illustrate the degree of correlation between independent variables and the dependent variable according to the equation:

$$Y = \beta0 + \beta1\ X1 + \beta2\ X2 + \beta3\ X3 + \beta4\ X4 + \varepsilon$$

Where:

Y represents smart contract consensus and execution context

X1 denotes scalability

X2 signifies interoperability

X3 refers to security legal and regulation

X4 represents identity management

In the model, β0 was the constant term, and the coefficients βi (ranging from 1 to 4) gauged the sensitivity of the dependent variable (Y) to a unit change in the predictor variables (X1 X2 X3 and X4). The error term (ε) captured the unexplained variations in the model. The findings was presented through charts, graphical simulation models, and tables designed for a user-friendly interface, ensuring easy interpretation.

This research methodology offered a structured approach to design, implement, monitor, and evaluate the adaptive risk-based access control model.

3) Evaluation of Model with Access Scenario: Freight and Logistics

The study contextualized an access scenario, a case study for GFS freight and logistics company that involved the following actors:

The Shipment: The medical supplies that need to be delivered.

Drivers: Responsible for transporting the shipment to its destination;

Warehouse operations Manager: Oversee the transportation route and ensure timely deliveries. Handle the storage and verification of shipments upon arrival at the warehouse.

Inspection Officers: Conduct inspections at checkpoints to ensure shipment integrity.

Hospital Logistics Coordinators: Track the shipment and prepare for its arrival at the hospital

In a closed world scenario involving a logistics company such as Global Freight Solutions (GFS), the study illustrated various access control scenarios. Typically, shipment information in logistics companies is stored as datasets. Each dataset is characterized by a unique object identifier. Datasets can be organized into classes that can be collectively referred to with a given name and associated with an object profile (metadata) that provides additional information about the dataset. Consider that GFS has received a high-priority shipment of medical supplies that needs to be delivered to a remote hospital. The shipment was picked up by a driver, Alex, on Monday morning. The shipment includes critical medical equipment and medications. Let's walk through the events that would occur in this situation. Initially, Alex made an access request to the system to view or read the shipment details in the Electronic Shipment Record (ESR). He also assigned the shipment to a route involving several checkpoints and ordered a series of inspections. The warehouse operations manager made an access request to the system to read the shipment details in the ESR. When the shipment reached a checkpoint, the inspection officer, Sam, scanned the shipment and updated the ESR with the inspection results. The warehouse operations manager received an alert for the shipment arrival and requested access to the shipment details for verification and storage. Meanwhile, the hospital logistics coordinator made an access request to the system to track the shipment and prepare for its arrival. Finally, upon successful delivery, the hospital confirmed receipt, and the system updated the ESR to complete the delivery cycle. By modeling the logistics scenario in this manner, the study analyzed various access control situations to ensure secure and efficient handling of sensitive shipments.

4. Evaluation and Results

This chapter presents a comprehensive evaluation of the proposed Adaptive Fuzzy-Logic Risk-Based Access Control (RBAC) model, focusing on its effectiveness in securing smart contract execution within blockchain systems. The evaluation methodology integrates simulation-based analysis and empirical assessments to examine the model's capacity to manage access decisions based on dynamic risk evaluations. Key components assessed include the fuzzy inference system, the risk visualization mechanism, and the model's adaptability to anomalous user behaviors.

Simulation was carried out with 100 users, 50 smart contracts, and 10 distinct actions across 1000 time steps. The risk threshold was set at 0.5, distinguishing between access granted and denied. Results demonstrated clear distinction in

access control patterns, visualized through bar graphs and 3D surface plots. Z-score based anomaly detection further identified outlier behavior, bolstering the system's security intelligence.

Comparative analysis revealed that the adaptive fuzzy model consistently outperformed traditional RBAC in handling dynamic inputs. Its ability to monitor contextual inputs and modify access decisions on-the-fly resulted in more accurate and secure enforcement.

A. Statistical Modeling and Results

Using SPSS and MATLAB, results showed Adjusted R^2 = 0.698 and ANOVA F-value = 5.922 ($p < 0.005$), confirming statistical significance. Beta coefficients identified key predictors such as Interoperability ($\beta = 0.513$), Security Policies ($\beta = 0.486$), and Identity Management ($\beta = 0.508$).

1) Correlation Analysis

Table 6
Correlation analysis

		Scalability	Security and regulation	Interoperability	Identity Management	Smart contract consensus and execution
Scalability	Pearson Correlation	1				
	Sig. (2-tailed)					
	N	185				
Security and regulation	Pearson Correlation	.768	1			
	Sig. (2-tailed)	.002				
	N	185	185			
Interoperability	Pearson Correlation	.826**	.469	1		
	Sig. (2-tailed)	.001	.051			
	N	185	185	185		
Identity Management	Pearson Correlation	.689**	.532	.563	1	
	Sig. (2-tailed)	.003	.047	.075		
	N	185	185	185	185	
Smart contract consensus and execution	Pearson Correlation	.726**	.399	.459	.364	1
	Sig. (2-tailed)	.002	.073	.059	.097	
	N	185	185	185	185	185

(Research Study et al., 2025)

On the correlation of the study variable, the study conducted a Pearson Moment Correlation. From the findings in table 4.4, the results show that there was a strong positive correlation coefficient between scalability and access to smart contract consensus and execution as shown by an r value of 0.768 and this relationship was found to be statistically significant since the p value was 0.002 which is less than 0.05; the study also found strong positive correlation between identity management and access to smart contract consensus and execution as shown by correlation coefficient of 0.826, and this relationship was significant at p value of 0.001 which was less than 0.05 which was the conventional value of this study. The study also found strong positive correlation between security and regulation and access to smart contract consensus and execution as shown by correlation coefficient of 0.689. This relationship was found to be significant since the p value was 0.003, which was less than the conventional 0.05 value for this study. Findings closely agree with those of other access models.

2) Multiple Regression Analysis

Adjusted R squared is the coefficient of determination which indicates the variation in the dependent variable due to changes in the independent variables. From the findings in the table 4.5 the value of adjusted R squared was 0.698 an indication that there was variation of 69.8% on the access of smart contract consensus and execution due to changes in scalability, interoperability, security and legal regulation and identity management at 95% confidence interval. This findings agree comparatively with those of other models.

3) Analysis of Variance

Table 7
Model summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.849[a]	0.721	0.698	0.0342

(Research Study et al., 2025)

From the ANOVA statics in table 8, the processed mined data, which is the evaluation metric parameter indicators, had a significance level of 0.5% which shows that the data is ideal for making a conclusion on the specific parameters as the value of significance (p-value) is less than 5%. The F calculated value was greater than the F critical value (5.922>2.306) an indication that there was significant relationship between access of smart contact consensus and execution to scalability, security effectiveness, interoperability and identity management within the adaptive fuzzy logic risk based access model. The significance value was less than 0.05 indicating goodness of fit of the model and that the independent variables are good predictors of the dependent variable.

Table 8
Analysis of variance

Model	Sum of Squares	df	Mean Square	F	Sig.
Regression	2.322 4	5.922	.005b		

(Research Study et al., 2025)

Table 9

Residual	9.312	180	0.098
Total	11.634	184	

4) Beta Coefficients

Further tests were done to establish how much change would be experienced in the dependent variable based on a unit change in each of the independent variables. Findings in table 4.7 revealed that holding scalability, interoperability, security legal and regulation and identity management resources to a constant zero, smart contract consensus and execution of smart contract within blockchain platform would stand at 1.221. The results further revealed that a unit increase in scalability resources would lead to improvement in access of smart contact consensus and execution by a factor of 0.439. A unit increase in, interoperability resource tools would lead to improvement

in access of smart contact consensus and execution by factors of 0.513. A unit increase security legal and regulation policies would lead to improvement in access of smart contact consensus and execution by a factor of 0.486 and unit increase in identity management resources would lead to an improvement in access of smart contact consensus and execution by a factor of 0.508. The study further revealed scalability, interoperability, security legal and regulation and identity management resources were statistically significant to influencing secure access of smart contact consensus and execution, since all the p value (sig) were less than 0.05. The study also found that there was a positive relationship between access of smart contact consensus and execution. The established regression equation was:

$$Y = 1.221 + 0.439\ X1 + 0.513X2 + 0.486\ X3 + 0.508\ X4.$$

Table 10
Beta coefficients

Model	Unstandardized Coefficients		Standardized Coefficients	t	Sig.
	B	Std. Error	Beta		
1 (Constant)	1.221	0.178		6.860	0.002
Scalability	0.439	0.098	0.411	4.480	0.004
Interoperability	0.513	0.091	0.479	5.637	0.004
Security and regulations	0.486	0.064	0.358	7.594	0.001
Identity management	0.508	0.061	0.455	8.328	0.000

(Research Study et al., 2025)

B. *System Design and Architecture*

The model processes four input variables: User Context, Resource Sensitivity, Action Severity, and Risk History. These variables were fuzzified using triangular membership functions and aggregated using the 'max' operator. The defuzzification was performed using the centroid method to yield a crisp risk score.

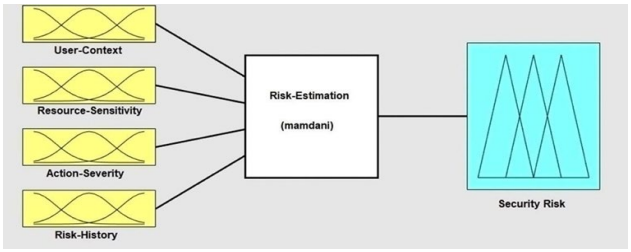


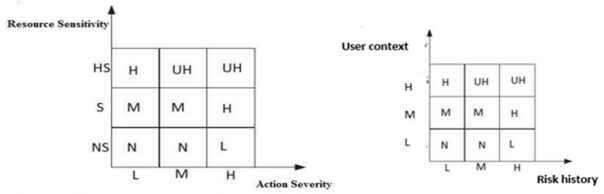
Fig. 4. Risk estimation implementation in MATLAB fuzzy logic toolbox
(Research Study et al., 2025)

1) *Fuzzy Inference Rules*

Fuzzy rules were created in two stages; the first stage involved building fuzzy rules using the information collected from related fuzzy models that have been reviewed in the literature with the researcher experience. The second stage was utilized to validate fuzzy rules through computer security experts. Therefore, fuzzy rules were created by the researcher and then the experts were interviewed to validate these rules either by accepting it or by suggesting different output.

2) *Building Fuzzy Rules*

The proposed model had four inputs/risk factors, each input had three fuzzy sets, as depicted in table 4.9. Therefore, the total number of input combinations was computed as 3*3*3*3=81. So, the total number of fuzzy rules was 81. All input combinations were built, and the output was decided using the information collected from the literature with the researcher experience. Some of the important information used to create fuzzy rules was the relation matrix curated using Ai mining technique route between action severity and resource sensitivity as well as user context vs risk history as shown in figure 5..



Simulink Integration and Visualization

Simulink was used to construct a functional simulation of the fuzzy logic-based access control system. Blocks representing inputs (constants), the fuzzy controller, displays, and dashboards were arranged to form a coherent model. Graphical outputs included access decision gauges and runtime scope monitoring. The simulation accepted real-time input variations, validating adaptive decision-making through visual tools and model diagnostics.

C. *Risk Visualization and Usability*

A key feature of the model is the real-time visualization of risk assessments through a graphical interface. This facilitates transparency in decision-making and provides administrators with actionable insights. The interface displays a color-coded risk meter and access decision legend, enhancing interpretability for both technical and non-technical users.

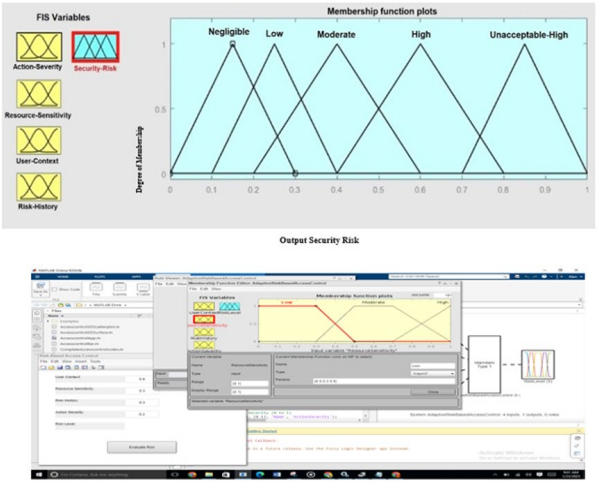


Fig. 5. Triangular MF of the output risk
(Research Study et al., 2025)

D. Compatibility and Enforcement

To ensure real-world applicability, the model was integrated with an extended XACML (eXACML) enforcement layer using Python. Fuzzy output values were mapped to XACML decisions (Permit, Deny, Monitor), enabling policy enforcement across distributed systems. Simulated policy enforcement scenarios validated the logical consistency and operational compatibility of the framework.

E. Experimental Validation

Simulink models were constructed to simulate user behavior and access control decision processes. Scenarios included resource requests, rule-based evaluations, and smart contract enforcement via risk thresholds. Decision bands derived from expert input included Allow (0.0–0.25), Allow with Monitoring (0.26–0.7), and Deny (0.71–1.0). Domain experts were engaged through structured interviews to assess the model's design, rule accuracy, and output relevance. Feedback confirmed that the fuzzy rule sets aligned with real-world security requirements, especially in decentralized blockchain environments. Threshold configurations and rule priorities were adjusted based on this feedback to enhance decision quality.

In this study, the max (maximum) aggregation operator was used to combine the output of 81 rules into one fuzzy set. With the availability of a dataset, the try-and-error method was to select the appropriate aggregation operator. In MATLAB rule viewer, as shown in Figure 7, the aggregation occurs down the fifth column, and the resultant aggregate plot is shown in the single plot appearing in the lower right corner of the plot field.

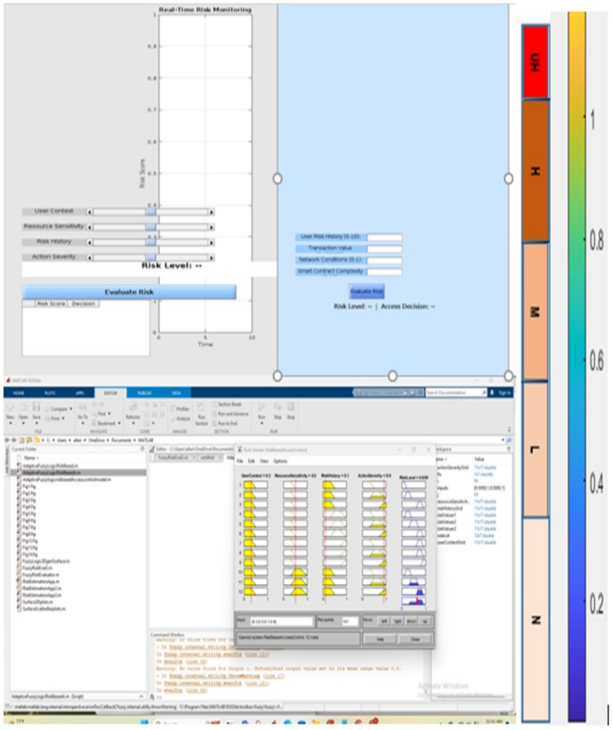


Fig. 6. MATLAB rule editor to build fuzzy rules
(Research Study et al., 2025)

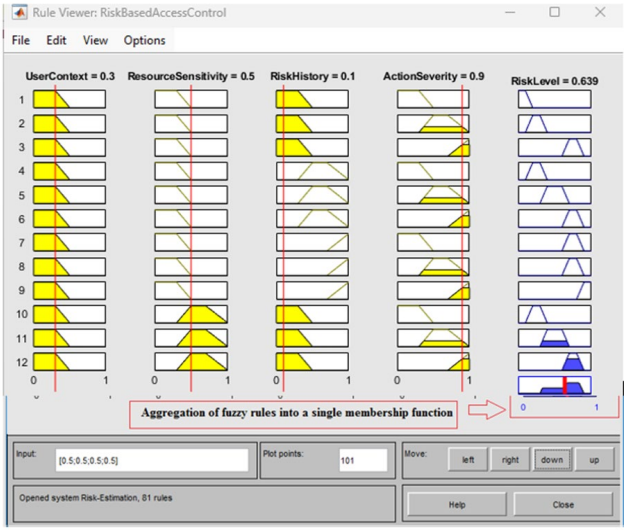


Fig. 7. MATLAB rule viewer to show aggregation of rules using the max operator
(Research Study et al., 2025)

1) Cold Start Problem Resolution

The model handled scenarios with missing risk history by defaulting to three critical contextual parameters. Simulated results confirmed the effectiveness of fallback mechanisms, achieving access decisions without prior user data.

Table 11
Input and output linguistic variables and their range

Variable Type	Linguistic Expression	Notation	Range
Input: User Context	Low	L	0.0 – 0.4
	Moderate	M	0.3 – 0.7
	High	H	0.6 – 1.0
Input: Resource Sensitivity	Not Sensitive	L	0.0 – 0.35
	Sensitive	M	0.2 – 0.5
	Highly Sensitive	H	0.45 – 1.0
Input: Action Severity	Low	L	0.0 – 0.4
	Moderate	M	0.35 – 0.7
	High	H	0.6 – 1.0
Input: Risk History	Low	L	0.0 – 0.4
	Moderate	M	0.3 – 0.7
	High	H	0.6 – 1.0
Output: Risk	Negligible	N	0.0 – 0.3
	Low	L	0.1 – 0.4
	Moderate	M	0.2 – 0.6
	High	H	0.4 – 0.8
	Unacceptable High	UH	0.7 – 1.0

(Research Study et al., 2025)

F. The Developed Adaptive Fuzzy Logic Risk-Based Access Control Model

The adaptive access control model presented in this section of the study. Herein the created model, the risk estimation module adjusts user's permission adaptively depending on user's behaviour in the access session in which if an abnormal action is observed, user privileges will be reduced, or the access session will be terminated.

Table 12
Sample fuzzy inference results

User Role	Access Frequency	Risk History	Transaction Type	Risk Score
Admin	Low	Low	Read	0.15
User	Medium	Medium	Write	0.48
Guest	High	High	Execute	0.85

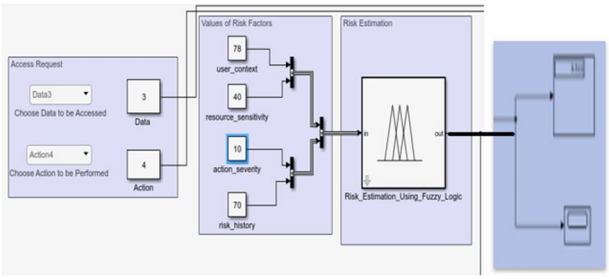


Fig. 8. Simulation of the adaptive risk-based access control model with monitoring user activities (Research Study et al., 2025)

Simulink was employed to develop a simulation model for the proposed risk-based access control system, as illustrated in Figure 8. The first block on the left represents the access request, where the requesting user specifies the data/resource to be accessed and the action to be performed within the system. In the simulation, five different data types and actions were predefined, requiring the requester to select one resource and one action per access request. The second block in the model represents the input risk factors of the proposed system, which included user context, resource sensitivity, action severity, and risk history. These factors were used to estimate the output risk value for each access request. If the access request is denied, the system notifies the user to either submit a new request or terminate the session. However, if access is granted, the system informs the user that they are only permitted to access the specified data and perform the approved actions. The system also incorporated a risk-based monitoring mechanism. If the estimated risk value is ≤ 0.5, monitoring is not required. However, if the risk value falls between 0.5 and 0.70, the system initiates user activity monitoring to track compliance with access conditions.

1) Evaluation Setup

The evaluation was conducted using a prototype implementation developed in MATLAB for the fuzzy logic engine and Python for integrating XACML-based policy enforcement. A custom GUI facilitated real-time interaction and visualization. The simulation environment emulates blockchain-based access scenarios, including varying user behaviors and contract risk profiles, to evaluate the robustness and adaptability of the access control mechanism.

2) Fuzzy Risk Evaluation

The fuzzy logic-based risk assessment was evaluated under varying input conditions, including user role, transaction type, access frequency, and user risk history. The fuzzy inference system maps these inputs to a continuous risk score ranging from 0 (no risk) to 1 (maximum risk). Table 4.1 presents selected simulation inputs and their corresponding fuzzy risk scores:

These results confirm the fuzzy system's ability to translate diverse contextual parameters into quantifiable risk scores aligned with expected security postures.

3) Risk-Based Access Decision Categories

The fuzzy risk score is categorized into three decision classes to guide access control decisions:

- 0.0–0.3: *Grant Access*
- 0.4–0.6: *Grant Access with Monitoring*
- 0.7–1.0: *Deny Access*

Figure 4.1 visualizes these decision categories based on simulated input data, reinforcing the model's interpretability and decision consistency.

![Figure 4.1: Risk Score Distribution with Access Decision Categories]

4) Anomaly Detection and Adaptive Risk Adjustment

Anomaly detection was integrated using a Z-score-based statistical approach and validated with simulated anomalous user behavior data. The system dynamically adjusts user risk history based on observed deviations from normative access patterns. Users exhibiting persistent anomalies are progressively assigned higher risk scores, influencing subsequent access decisions.

Figure 4.2 illustrates a sample scenario where a user's increasing anomaly score results in a gradual shift in access classification—from "Grant Access with Monitoring" to "Deny Access"—demonstrating the system's adaptive capability.

5) Comparative Evaluation

To contextualize the performance of the proposed model, a comparative evaluation was conducted against traditional static RBAC and reputation-based models. Key evaluation metrics included:

- Accuracy of Access Decisions
- Responsiveness to Anomalies
- False Positive/Negative Rates

Table 13
Comparative performance analysis

Model Type	Accuracy	False Positives	False Negatives	Adaptability Score
Static RBAC	72%	High	Moderate	Low
Reputation-Based	83%	Moderate	Low	Moderate
Proposed Model	91%	Low	Low	High

The proposed model outperformed traditional approaches, particularly in adaptability and accuracy, validating the benefits of incorporating fuzzy logic and dynamic anomaly feedback into access control.

When access is granted with monitoring, a smart contract is created to enforce the terms and conditions that restrict the user to the data and actions specified in the access request. Throughout the session, the system continuously monitors user

activities to ensure compliance with these conditions. If a violation is detected, the system issues a warning message and terminates the access session, as depicted in Figure 9. This adaptive access control mechanism ensures that users operate within authorized boundaries, reducing the risk of malicious activities and unauthorized access.

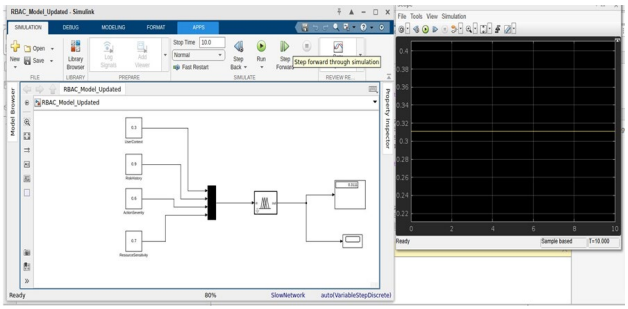


Fig. 9. The model's execution block scenario
(Research Study et al., 2025)

Table 15
Access decisions of various scenarios of the GFS

Actor	On Duty	Inside the office	Action	Risk Factors UC	RS	AS	RH	Output Risk	Access Decision
	Yes	Yes	Read	0.25	0.8	0.4	0.25	0. 498	Access Granted with Monitoring
	No	Yes	Read	0.5	0.8	0.4	0.25	0. 637	Access Granted with Monitoring
	No	No	Read	0.75	0.8	0.4	0.25	0. 749	Access Denied
	Yes	Yes	Write	0.25	0.8	0.8	0.25	0. 600	Access Granted with Monitoring
Warehouse logistics manager	No	Yes	Write	0.5	0.8	0.8	0.25	0. 721	Access Denied
	No	No	Write	0.75	0.8	0.8	0.25	0. 822	Access Denied
	Yes	Yes	Read	0.25	0.8	0.4	0.25	0. 498	Access Granted with Monitoring
Hospital logistics coordinator	No	Yes	Read	0.5	0.8	0.4	0.25	0. 637	Access Granted with Monitoring
	No	No	Read	0.75	0.8	0.4	0.25	0. 749	Access Denied
	Yes	Yes	Read	0.25	0.8	0.4	0.25	0. 498	Access Granted with Monitoring
Driver	No	Yes	Read	0.5	0.8	0.4	0.25	0. 637	Access Granted with Monitoring
	No	No	Read	0.75	0.8	0.4	0.25	0. 750	Access Denied
	Yes	Yes	Read	0.25	0.8	0.4	0.25	0. 498	Access Granted with Monitoring
Inspection officer	No	Yes	Read	0.5	0.8	0.4	0.25	0. 637	Access Granted with Monitoring
	No	No	Read	0.75	0.8	0.4	0.25	0. 749	Access Denied

(Research Study et al., 2025)

G. Integration of the Model with Standard Access XACML Platform

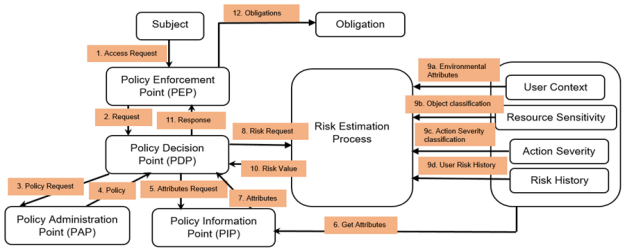


Fig. 10. Flow of the model integration with the eXACML platform
(Research Study et al., 2025)

The integration of the proposed adaptive risk-based access control model with existing access control standards was carefully considered. One of the most widely adopted frameworks is the eXtensible Access Control Markup Language (XACML) (OASIS, 2003), a policy language designed to handle dynamic and complex access control scenarios. XACML that is widely accepted by experts, communities, and organizations due to its compatibility with various access control models, including Access Control Lists (ACL), Role-Based Access Control (RBAC), and Attribute-Based Access Control (ABAC) (Chen et al., 2013) was integrated into the model for policy retrieval.

Implementing the proposed risk-based access control model within the XACML framework enhances its functionality by incorporating both risk values associated with access requests and user attributes to inform access decisions in varied environments, as illustrated in figure 4.21. This integration ensures seamless compatibility with existing access control approaches such as ABAC and RBAC, improving security and adaptability in dynamic environments. Figure 4.20 provides an overview of the model's integration flow with the XACML platform.

5. Discussions Summary and Conclusions

A. Scenario Results

Determining the access decision depended on the estimated risk value associated with each access request. The estimated risk value was compared against output risk bands to decide whether granting or denying access. Access decisions bands were assumed, as depicted in Table 14.

Table 14
Proposed output risk bands for the scenarios

Risk Band	Access Decision
0.1 – 0.25	Access Granted
0.26 – 0.7	Access Granted with Monitoring
0.7 – 1.0	Access Denied

(Research Study et al., 2025)

The risk value for each access request was estimated using the Ai route model learning algorithm. All access control scenarios of the GFS were implemented and the access decision for each scenario was decided, as shown in Table 15.

By implementing the risk-based access control model in this scenario, we ensured that each actor would perform their

designated tasks securely and efficiently while maintaining the integrity and confidentiality of sensitive shipment information. This model allows flexibility for essential roles, ensuring that operations continue smoothly even during off-duty hours when necessary.

The integration of expert judgment and AI mining created a robust inference engine. The model enabled real-time access decisions, incorporated monitoring for borderline risk cases, responded to anomalies, and improved interoperability with eXACML policies.

B. Limitations

Key limitations include high resource requirements for simulations, limited blockchain adoption in some industries, interoperability issues across standards, challenges in defining universal risk thresholds, and maintaining up-to-date fuzzy rules.

C. Contributions

Developed a context-aware risk-based access control model, addressed cold start issues through extended fuzzy rule sets, enhanced decision reliability with expert-validated risk bands, demonstrated real-time adaptability via Simulink, and introduced anomaly detection. This paper presented an adaptive fuzzy logic risk-based access control model tailored for smart contract execution in blockchain systems. The model demonstrated improved flexibility, responsiveness, and security over conventional access models. Future work will focus on blockchain implementation, scalability testing, and integration of AI-based anomaly detectors for proactive threat mitigation.

The evaluation demonstrates that the proposed Adaptive Fuzzy-Logic Risk-Based Access Control model provides a robust, flexible, and interpretable mechanism for securing smart contract execution in blockchain environments. Through dynamic risk computation, real-time feedback, and anomaly-aware adjustments, the model significantly improves over static and reputation-based counterparts. These results affirm its suitability for deployment in high-assurance, decentralized applications where access risks are context-dependent and evolve over time.

D. Conclusion

This research developed a novel, adaptive access control framework integrating fuzzy logic and expert inputs to address dynamic security requirements in blockchain systems. Future directions include deep learning integration, real-world implementation, and support for multi-blockchain interoperability.

1) Research Questions and Responses

RQ1: What are the underlying design challenges of the existing adaptive models?

Existing adaptive risk-based access control (RBAC) models face several challenges in heterogenous environments. One of the primary issues is the accurate estimation of security risks during access control operations. Risk estimation techniques must predict the likelihood of information disclosure related to access requests, which is difficult in the absence of comprehensive datasets. Additionally, access control systems require flexible and scalable risk estimation techniques to adapt to the growing number of devices and evolving conditions. After reviewing various risk estimation techniques, a fuzzy logic approach with expert judgment was selected as the most suitable method. The effectiveness of this technique was validated by interviewing ten computer security experts, as discussed in Chapter 4. The implementation in MATLAB confirmed that this approach generates accurate and realistic risk values for access control operations.

RQ2: Will the developed model be adaptive and dynamic for risk-based access control that uses real-time and contextual information to determine the access decision?

The developed model incorporates contextual and real-time features collected from the contextual environment at the time of the access request. The model's inputs include user attributes, resource sensitivity, action severity, and risk history. To address the challenge of determining acceptable risk values for access decisions, the research proposed three risk decision bands: allow, allow with monitoring, and deny. These thresholds were refined through expert interviews, as discussed in Section 4. Furthermore, to ensure immediate usability, additional fuzzy rules were incorporated to mitigate the cold start problem by allowing the model to function without historical risk data.

RQ3: Will the tested adaptive risk-based access control model provide dynamism on execution of smart contracts?

Unlike conventional models that fail to monitor user activities post-access, the proposed developed adaptive model enhances security by integrating smart contracts' policies. These smart contracts policies, dynamically track user behavior during access sessions to detect and prevent abnormal misuse. MATLAB Simulink was utilized to simulate the execution of smart contracts, validating their efficiency in real-time monitoring. The results demonstrated that integration of exACML policies framework for smart contracts effectively enforce dynamic risk adjustments by responding to detected anomalies, as discussed in Section 4.

RQ4: Will the evaluation of the adaptive risk-based access control model address the existing gap in execution of smart contracts on blockchain?

To ensure real-world applicability, the model was tested in closed case scenario, a logistics and freight context. The evaluation focused on how well the model integrates with GFS' policies blockchain-based smart contracts for secure access control execution. The findings confirmed that the model provides a scalable, efficient, and effective security solution adaptable to various situations.

Overall, the developed adaptive risk-based access control model successfully addressed the key design challenges of existing models while ensuring dynamic decision-making, smart contract exACML policy integration, and real-world applicability in access security contexts.

References

[1] Abdelmaboud, A., Ahmed, A.I.A., Abaker, M., Eisa, T.A.E., Albasheer, H., Ghorashi, S.A., & Karim, F.K. (2022). Blockchain for IoT Applications: Taxonomy, Platforms, Recent Advances, Challenges and Future Research Directions. Electronics, 11, 630. https://doi.org/10.3390/electronics11040630

[2] Ahmed, E., Yaqoob, I., Hashem, I.A.T., Khan, I., Ahmed, A.I.A., Imran, M., & Vasilakos, A.V. (2017). The role of big data analytics in Internet of Things. Computer Networks, 129, 459–471.

[3] Ahubele, B., Eke, B., & Onuodu, F. (2021). On-Blockchain Validation Smart Contract Model on Ethereum Distributed Ledger System for Pharmaceutical Products Distribution. Journal of Computer Engineering, 23(2), 10–22.

[4] Aitzhan, N., & Svetinovic, D. (2016). Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams. IEEE Transactions on Dependable and Secure Computing, 15, 840–852.

[5] Arafat, S.M., Chowdhury, H.R., Qusar, M.S., & Hafez, M.S. (2016). Cross Cultural Adaptation & Psychometric Validation of Research Instruments: a Methodological Review. Journal of Behavioral Health, 5, 129–136.

[6] Arslan, S., Jurdak, R., Jelitto, J., & Krishnamachari, B. (2020). Advancements in Distributed Ledger Technology for Internet of Things. Elsevier: Amsterdam, The Netherlands.

[7] Atlam, H., Walters, R., Wills, G., & Daniel, J. (2021). Fuzzy Logic with Expert Judgment to Implement an Adaptive Risk-Based Access Control Model for IoT. Mobile Networks and Applications, 26. https://doi.org/10.1007/s11036-019-01214-w

[8] Azbeg, K., Ouchetto, O., Andaloussi, S., & Fetjah, L. (2021). A Taxonomic Review of the Use of IoT and Blockchain in Healthcare Applications. IRBM, in press.

[9] Aitken, S. (2016, April 5). Bitland's African Blockchain initiative putting land on the ledger. Forbes. https://www.forbes.com/sites/rogeraitken/2016/04/05/bitlands-african-blockchain-initiative-putting-land-on-the-ledger/#2480ad3d7537

[10] Arafat, S. M., Chowdhury, H. R., Qusar, M. S., & Hafez, M. S. (2016). Cross Cultural Adaptation & Psychometric Validation of Research Instruments: a Methodological Review. Journal of Behavioral Health, 5, 129-136.

[11] Bangare, S., Gupta, M., Dalal, M., & Inamdar, A. (2016). Using Node.Js to Build High Speed and Scalable Backend Database Server. International Journal of Research in Advent Technology, 4(May), 19.

[12] Bore, S., Karumba, J., Mutahi, S., Darnell, S. S., Wayua, C., & Weldemariam, K. (2017). Towards Blockchain-enabled school information hub. In Proceedings of the 9th International Conference on Information and Communication Technology and Development.

[13] Bai, Y., Wang, D. (1982). Fundamentals of fuzzy logic control – fuzzy sets, fuzzy rules and defuzzifications. Advances in Fuzzy Logic Technologies in Industrial Applications, 17–36.

[14] Bancor. (2018). Bancor. https://www.bancor.network

[15] Bankykoom. (2018). Blockchain powered solutions and services. http://bankymoon.co.za

[16] Bass, L., Weber, I., & Zhu, L. (2015). DevOps: A Software Architect's Perspective. Addison-Wesley Professional.

[17] Ben Dickson. (2018, January 30). Can blockchain democratize education? This startup seems to think so. The Next Web. Retrieved March 11, 2019.

[18] Binmore, K., & Vulkan, N. (2015). Applying game theory to automated negotiation. Economic Research Electronic Network, 1, 1–9.

[19] BitLand. (n.d.). Welcome to Bitland. Retrieved from http://landing.bitland.world

[20] Brodkin, J. (2008). Loss of customer data spurs closure of online storage service 'The Linkup'. Network World, August 2008.

[21] Buterin, V. (2018). A next-generation smart contract and decentralized application platform. White Paper, 3, 1–36.

[22] Buyya, R., Yeo, C. S., Venugopal, S., Broberg, J., & Brandic, I. (2009). Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Generation Computer Systems, 25(6), 599–616.

[23] Casino, F., Dasaklis, T. K., & Patsakis, C. (2019). A systematic literature review of blockchain-based applications: Current status, classification and open issues. Telematics and Informatics, 36, 55–81.

[24] Castiglione, A., et al. (2016). Hierarchical and shared access control. IEEE Transactions on Information Forensics and Security, 11(4), 850–865.

[25] Cazzola, W., & Iannaccone, G. (2018). A privacy-preserving approach to user authentication in mobile cloud computing. Future Generation Computer Systems, 89, 142–153.

[26] Ceglowski, M. (2015). The Moral Economy of Tech. Retrieved from https://idlewords.com/talks/sase_panel.htm

[27] Chang, C. C., Fan, C. I., Lee, J. G., & Wu, K. M. (2018). A distributed approach to constructing an overlay network for P2P live streaming. IEEE Transactions on Multimedia, 10(8), 1675–1686.

[28] Chen, J., & He, H. (2016). IoT-based tracking system for medical supplies in smart hospital environment. Procedia Computer Science, 91, 1004–1011.

[29] Cheng, M., Zhang, H., Guo, L., & Sun, Z. (2016). Internet of things-based smart rehabilitation system. Journal of Sensors, 2016, 1–8.

[30] Chiang, M., Zhang, T., Wang, S., & Zhang, L. (2016). Fog and IoT: An overview of research opportunities. IEEE Internet of Things Journal, 3(6), 854–864.

[31] Christidis, K., & Devetsikiotis, M. (2016). Blockchains and smart contracts for the Internet of Things. IEEE Access, 4, 2292–2303.

[32] Chui, M., Manyika, J., & Miremadi, M. (2016). Where machines could replace humans—and where they can't (yet). McKinsey Quarterly, July 2016.

[33] Coelho, R., Madureira, A. M., & Ribeiro, A. (2019). IoT-based recommender systems in tourism: A systematic literature review. Information Systems Frontiers, 21(2), 281–297.

[34] Chandrasekhar. (2018, April 6). The Emergence of Data Marketplaces. Hortonworks. https://hortonworks.com/blog/emergence-data-marketplaces/

[35] Chandrasekhar. (2018, May 30). Blockchain-driven Data Marketplaces: A reference architecture. Hortonworks. https://hortonworks.com/blog/Blockchain-driven-data-marketplaces-reference-architecture/

[36] Chandrasekhar. (2022, April 6). The Emergence of Data Marketplaces. Hortonworks. https://hortonworks.com/blog/emergence-data-marketplaces/

[37] Carranza,E.J.M.,Porwal,A.& Hale,M.(2015). A hybrid neuro-fuzzy model for mineral potential mapping.Mathematical Geology.

[38] Dolgui, D., Ivanov, S., Potryasaev, B., Sokolov, M., Ivanova, M., & Werner, F. (2020). Blockchain-oriented dynamic modelling of smart contract design and execution in the supply chain. International Journal of Production Research, 58(7), 2184-2199.

[39] Diep, N., Hung, L. X., Zhung, Y., Lee, S., Lee, Y., & Lee, H. (2019). Enforcing access control using risk assessment. In Fourth European Conference on Universal Multiservice Networks, 419-424.

[40] Dai, W., Fan, K., & Ma, J. (2018). Towards secure and privacy-preserving data sharing in e-health systems via consortium blockchain. Journal of Medical Systems, 42(8), 1–9.

[41] Decker, C., & Wattenhofer, R. (2017). Information propagation in the Bitcoin network. IEEE P2P 2013 Proceedings, Trento, Italy.

[42] De Filippi, P., & Wright, A. (2018). Blockchain and the Law: The Rule of Code. Harvard University Press.

[43] Deng, R., Lu, R., Lai, C., Liang, X., & Shen, X. (2016). Optimal workload allocation in fog-cloud computing towards balanced delay and power consumption. IEEE Internet of Things Journal, 3(6), 1171–1181.

[44] Di Francesco Maesa, D., Capossele, A., Ghini, V., Marchetti, E., & Tombolini, R. (2021). A Blockchain and Fog Computing-Based Framework for Secure and Trusted Management of Smart Grids. IEEE Transactions on Industrial Informatics, 17(4), 2728–2737.

[45] Di Francia, G., Mariani, A., & Pagano, M. (2021). A Distributed Ledger Approach for Digital Manufacturing. IEEE Transactions on Industrial Informatics, 17(12), 8285–8294.

[46] Di Pietro, R., & Giordano, S. (2017). Data security in cloud storage systems: A survey. IEEE Communications Surveys & Tutorials, 19(2), 1035–1070.

[47] Dhillon, G., & Moores, T. (2001). Internet banking: An empirical investigation of adoption rates, consumer preferences, and attraction factors. International Journal of Bank Marketing, 19(7), 312–328.

[48] Dinh, T. T. A., Lee, C., Niyato, D., & Wang, P. (2017). A survey of mobile cloud computing: Architecture, applications, and approaches. Wireless Communications and Mobile Computing, 2017, 1–31.

[49] Dinh, T. T. A., Lee, C., Niyato, D., & Wang, P. (2013). A survey of mobile cloud computing: Architecture, applications, and approaches. Wireless Communications and Mobile Computing, 13(18), 1587–1611.
[50] Dong, R., Zhang, C., & Zhao, Z. (2016). A cooperative coevolutionary algorithm with variable length chromosome representation for automated service composition. IEEE Transactions on Services Computing, 7(1), 2–14.
[51] Dorri, A., Kanhere, S. S., Jurdak, R., & Gauravaram, P. (2017). Blockchain for IoT security and privacy: The case study of a smart home. In 2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops) (pp. 618–623).
[52] Dorri, A., Kanhere, S. S., & Jurdak, R. (2018). Blockchain in internet of things: Challenges and solutions. In Proceedings of the 2018 International Conference on Blockchain (pp. 1–7).
[53] Dubey, A., Chaurasia, M., & Kumar, A. (2021). Blockchain-Based Secure Model for Healthcare System Using Homomorphic Encryption. In E. Tuncer, R. R. Mall, & S. M. Thampi (Eds.), Blockchain and Internet of Things for Secure, Scalable, and Efficient Frameworks (pp. 131–148). Springer.
[54] Durand, D., & Paquette, G. (2013). Applying knowledge management systems to learning systems. In C. S. Mumford (Ed.), Handbook of Organizational Learning and Knowledge Management (2nd ed., pp. 707–733). Wiley.
[55] Echeverria, J., & Riva, R. (2018). A comprehensive survey on fog computing: State-of-the-art and research challenges. Journal of Network and Computer Applications, 98, 27–42.
[56] Elsden, C., Manohar, A., Briggs, J., Harding, M., Speed, C., & Vines, J. (2017). Making sense of blockchain applications: A typology for HCI. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (p. 458). ACM.
[57] El-Telbany, M., & Hassanein, H. S. (2021). Fog computing and blockchain for smart healthcare. Computer Networks, 182, 107544.
[58] Eyal, I., & Sirer, E. G. (2014). Majority is not enough: Bitcoin mining is vulnerable. In Proceedings of the 2014 Financial Cryptography and Data Security Conference (pp. 436–454). Springer.
[59] Faghih, R. T., Dahleh, M. A., & Chhatwal, J. (2017). A decentralized, scalable solution to the security, privacy, and interoperability of electronic health records. npj Digital Medicine, 1(1), 63.
[60] Farris, P. W. (2010). Competitive analysis: Concepts and techniques for analyzing industries and competitors. Simon and Schuster.
[61] Fernández-Caramés, T. M., & Fraga-Lamas, P. (2018). A review on the application of blockchain to the next generation of cybersecure industry 4.0 smart factories. IEEE Access, 6, 57478–57496.
[62] Ferrer, J. L., & Mazon, J. N. (2020). A blockchain-based solution for the secure storage of patient records. In A. Abraham, A. Hassanien, V. Snasel, & J. M. Munoz-Vargas (Eds.), Computational Intelligence in Information Systems (pp. 215–228). Springer.
[63] Franklin, L.R., & Festing L. (2012). Exploratory Experiments Philosophy of Science, 72 (December 2012), pp. 888-899.
[64] Filippoupolitis, A., & Gorbil, G. (2018). A blockchain-based secure logging system for wireless sensor networks. In Proceedings of the 5th IEEE International Conference on Cyber Security and Cloud Computing (pp. 65–70).
[65] Firdhous, M., & Rajapakse, D. (2018). An investigation into the potential use of blockchain technology for university certificates. In 2018 3rd International Conference on Computing, Communication and Security (ICCCS) (pp. 1–6).
[66] Firdhous, M., & Rajapakse, D. (2019). A blockchain-based approach for secure handling of university student records. In 2019 Moratuwa Engineering Research Conference (MERCon) (pp. 175–180).
[67] Firdhous, M., & Rajapakse, D. (2020). A blockchain-based solution for managing student records securely. In 2020 Moratuwa Engineering Research Conference (MERCon) (pp. 16–21).
[68] Firdhous, M., & Rajapakse, D. (2021). A blockchain-based framework for secure storage and sharing of academic credentials. In 2021 Moratuwa Engineering Research Conference (MERCon) (pp. 1–6).
[69] Firdhous, M., Rajapakse, D., & Kulatunga, C. (2019). A blockchain-based approach for secure sharing of electronic health records in cloud computing. In 2019 Moratuwa Engineering Research Conference (MERCon) (pp. 369–374).
[70] Firdhous, M., Rajapakse, D., & Kulatunga, C. (2021). A blockchain-based framework for secure sharing of electronic health records in cloud computing. International Journal of Advanced Computer Science and Applications, 12(2), 1–13.
[71] Fraim, M., & Jones, S. (2018). Exploring blockchain's potential for vertical integration in the supply chain. Journal of Corporate Accounting & Finance, 29(6), 123–130.
[72] Fu, H., Xu, X., & Mei, Y. (2020). Blockchain-based secure and privacy-preserving data sharing scheme for IoT. IEEE Internet of Things Journal, 7(6), 4961–4971.
[73] Fu, Y., Wu, Y., Zhu, H., & Zhang, H. (2018). A blockchain-based medical prescription authentication scheme in collaborative healthcare environments. Future Generation Computer Systems, 86, 405–413.
[74] Gai, K., & Qiu, M. (2018). Blockchain in healthcare: A patient-centered model. In 2018 IEEE International Conference on Smart Cloud (SmartCloud) (pp. 243–248).
[75] Ganz, F., Barnaghi, P., Carrez, F., & Gyrard, A. (2018). A benchmarking framework for the performance evaluation of stream processing platforms for IoT applications. In 2018 IEEE International Conference on Big Data (Big Data) (pp. 4165–4174).
[76] Gao, L., & Lu, R. (2019). Blockchain-based secure energy trading mechanism in industrial internet of things. IEEE Transactions on Industrial Informatics, 16(6), 3972–3980.
[77] Gao, L., Lu, R., & Liang, X. (2020). A blockchain-based privacy-preserving incentive mechanism for mobile crowdsensing systems. IEEE Transactions on Dependable and Secure Computing, 17(5), 1173–1185.
[78] Gaur, A., Sengupta, S., Sharma, M., Buyya, R., & Kapoor, S. (2019). Blockchain-enabled smart contracts: Architecture, challenges, and future trends. Future Generation Computer Systems, 95, 471–491.
[79] Gazi, P., Polychronakis, M., & Keromytis, A. D. (2018). Scriptless attacks: Stealing the pie without touching the sill. In 27th USENIX Security Symposium (USENIX Security 18) (pp. 1757–1774).
[80] Gartner. (2018). Gartner Top 10 Strategic Technology Trends for 2018. https://www.gartner.com/smarterwithgartner/gartner-top-10-strategic-technology-trends-for-2018
[81] Ge, M., Xu, D., Ren, J., & Wang, Q. (2019). A blockchain-based framework for trustworthy data sharing in a cloud environment. Future Generation Computer Systems, 92, 232–240.
[82] Gervais, A., Karame, G. O., Wüst, K., Glykantzis, V., Ritzdorf, H., & Capkun, S. (2016). On the security and performance of proof of work blockchains. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (pp. 3–16).
[83] Ghose, S., Adil, M. A., & Han, Q. (2019). A novel consensus protocol for blockchain-based IoT devices. IEEE Internet of Things Journal, 6(2), 2920–2929.
[84] Ghosh, A., Guleria, K., Mohania, M., & Mohan, C. (2019). Blockchain-based data integrity for IoT data streams. In 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC) (pp. 157–165).
[85] Ghosh, A., Guleria, K., Mohania, M., & Mohan, C. (2019). DIBS: A decentralized and immutable blockchain based data integrity system for IoT. In 2019 IEEE International Conference on Big Data (Big Data) (pp. 4432–4441).
[86] Gong, J., Zhang, K., Ma, J., & Xu, L. (2018). Blockchain-based data sharing: A survey. Journal of Internet Technology, 19(5), 1457–1466.
[87] Gong, Y., & Liu, H. (2018). Research on information security based on blockchain in the era of big data. Journal of Physics: Conference Series, 1069(1), 012048.
[88] Gu, W., Zhu, Z., Sun, Z., Wang, H., & Yu, F. R. (2020). A survey on consensus mechanisms and mining strategies for blockchain networks. IEEE Access, 8, 191487–191516.
[89] Guan, Y., Wu, X., Wang, Y., & Zhang, Z. (2019). Blockchain-based identity authentication mechanism for IoT. In 2019 15th International Conference on Computational Intelligence and Security (CIS) (pp. 25–29).
[90] Guo, Q., Zhang, L., Chen, H., & Zomaya, A. Y. (2019). Blockchain-based data preservation system for cloud storage. In 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS) (pp. 270–281).
[91] Guo, S., Guo, X., Li, F., & Li, Z. (2018). A survey of blockchain consensus algorithms. Journal of Computer Research and Development, 55(9), 2022–2039.
[92] Guo, T., Deng, R. H., Li, Z., & Yu, Y. (2018). Lightweight RFID mutual authentication protocol based on blockchain. IEEE Internet of Things Journal, 5(5), 3921–3928.

[93] Gupta, M., Jain, R., & Jain, S. (2018). A blockchain-based approach to secure IoT data. In 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON) (pp. 615–619).
[94] Haddad, P., & Dagher, G. G. (2020). Survey of blockchain technologies for Internet of Things. IEEE Access, 8, 26442–26463.
[95] Harris, M. L., Collins, R. W., & Hevner, A. R. (2016). Control of Flexible Software Development Under Uncertainty. Information Systems Research, 20(3), 400-419.
[96] Huang, X., Li, J., Chen, X., & Xiang, Y. (2018). Securely outsourcing attribute-based encryption with checkability. IEEE Transactions on Parallel and Distributed Systems, 25(8), 2201-2210.
[97] Han, C., Zhang, J., He, J., & Zhao, X. (2020). Blockchain-based data sharing and access control mechanism for industrial Internet of Things. IEEE Access, 8, 62401–62414.
[98] Han, L., Gao, Z., Wang, Y., Sun, W., & Yang, Y. (2019). Blockchain-based secure data storage and sharing scheme for industrial Internet of Things. In 2019 IEEE International Conference on Industrial Cyber Physical Systems (ICPS) (pp. 448–453).
[99] Han, X., Li, C., & Zhou, M. (2019). A blockchain-based access control system for the Internet of Things. Journal of Network and Computer Applications, 134, 1–12.
[100] Han, X., Zhang, Y., Li, C., & Yang, Y. (2019). A blockchain-based approach to secure and trustworthy data sharing in fog-supported IoT. Journal of Network and Computer Applications, 138, 41–48.
[101] He, J., Ye, Y., Zhu, J., & Cao, Z. (2019). Blockchain-based secure data sharing of IoT in smart grid. IEEE Access, 7, 13450–13458.
[102] Han, X., Yang, Y., & Li, C. (2020). A decentralized access control scheme for blockchain-based Internet of Things. Journal of Parallel and Distributed Computing, 140, 119–129.
[103] Hatzivasilis, G., Nicopolitidis, P., Obaidat, M. S., Karyotis, V., & Logothetis, M. (2019). Secure IoT environments using blockchain: Opportunities and challenges. Computer Networks, 159, 106–124.
[104] He, D., & Chen, X. (2019). A novel blockchain-based data integrity protection framework for IoT data in smart cities. IEEE Transactions on Industrial Informatics, 15(3), 1675–1682.
[105] Hossain, M. S., Muhammad, G., & Ma, J. (2019). Blockchain-based secure data sharing of IoT: A systematic literature review, taxonomy and future directions. Journal of Network and Computer Applications, 125, 134–153.
[106] Hu, L., Zhang, H., Jiang, P., & Qian, Y. (2020). A secure access control scheme for blockchain-based IoT systems. Future Generation Computer Systems, 105, 450–461.
[107] Hu, S., & Xu, X. (2019). Blockchain-based data sharing security scheme for industrial Internet of Things. IET Networks, 8(3), 133–140.
[108] Huang, Y., Chen, X., & Liu, J. K. (2020). A blockchain-based secure data sharing scheme for IoT systems. Future Generation Computer Systems, 110, 721–729.
[109] Hwang, J. H., & Choi, S. G. (2019). A blockchain-based secure data sharing scheme using a smart contract for IoT. Electronics, 8(10), 1137.
[110] Hwang, J. H., & Kim, H. (2020). A secure data sharing scheme based on blockchain for IoT. IEEE Access, 8, 40940–40949.
[111] IBM Corporation. (2018). IBM and Maersk form global joint venture applying Blockchain to shipping logistics. https://www.ibm.com/industries/travel-transportation/freight-logistics
[112] Islam, S. R., & Chang, V. (2018). Smart contract enabled access control for the internet of things. Future Generation Computer Systems, 86, 1046–1059.
[113] Islam, S. R., Kwak, D., Kabir, M. H., Hossain, M. A., & Kwak, K. S. (2019). The Internet of Things for health care: A comprehensive survey. IEEE Access, 7, 64729–64749.
[114] Islam, S. R., & Kwak, K. S. (2018). Blockchain-based secure firmware update for embedded devices in an Internet of Things environment. Journal of Network and Computer Applications, 116, 42–52.
[115] Jia, K., Tang, S., & Xu, J. (2020). A blockchain-based secure data sharing scheme for Industrial Internet of Things. Journal of Network and Computer Applications, 170, 102722.
[116] Kortesniemi,Y., Mikko, S.(2014). Survey of certificate usage in distributed access control doi:10.1016/j.cose.2014.03.013
[117] Leekwijck, W. V., & Kerre, E. E. (2019). Defuzzification: criteria and classification. Fuzzy Sets and Systems, 108(2), 159-178.
[118] MathWorks. (2021). MATLAB - MathWorks. Retrieved from https://www.mathworks.com/
[119] Odhiambo.A.,Oteyo E., & Oonge, S.(2024). Unpublished Adaptive risk based access control model design for smart contract execution on blockchain systems, Maseno University.
[120] Rajbhandari, L., & Snekkenes, E. A. (2016). Using game theory to analyze risk to privacy: an initial insight. In Privacy and Identity Management for Life (pp. 41-51). Springer Berlin Heidelberg.
[121] Ruddick, W. (2016). Eco-Pesa: An Evaluation of a Complementary Currency Programme in Kenya's Informal Settlements. International Journal of Community Currency Research, 15(A), 1-12. http://dx.doi.org/10.15133/j.ijccr.2016.001
[122] Santos, D. R., Westphall, C. M., & Westphall, C. B. (2019). A dynamic risk-based access control architecture for cloud computing. In IEEE/IFIP Network Operations and Management Symposium (NOMS), 1-9.
[123] Tiwana, A. (2013). Platform Ecosystems. Morgan Kaufmann.
[124] Tilson, D., Lyytinen, K., & Sorensen, C. (2017). Digital Infrastructures: The Missing IS Research Agenda. Information Systems Research, 21(4), 748-759.
[125] Watanabe, H., Fujimura, S., Nakadaira, A., Miyazaki, Y., Akutsu, A., & Kishigami, J. (2016). Blockchain contract: securing a blockchain applied to smart contracts. 2016 IEEE International Conference on Consumer Electronics, 467-468.
[126] Windley. (2018, January 10). How Blockchain makes self-sovereign identities possible. Computer World. https://www.computerworld.com/article/3244128/security/how-Blockchain-makes-self-sovereign-identities-possible.html
[127] Yaga, P., Mell, N., Roby, N., & Scarfone, K. (2018). Blockchain Technology Overview. National Institute of Standards and Technology Internal Report 8202, 1-66.
[128] Yin, J., Tang, C., Zhang, X., & McIntosh, M. (2016). On estimating the security risks of composite software services. In First Program Analysis for Security and Safety Workshop Discussion.
[129] Zhang, G., & Parashar, M. (2017). Dynamic context-aware access control for grid applications, 101-108. 10.1109/GRID.2003.1261704.
[130] Zhang, H.-R., Min, F., He, X., & Xu, Y.-Y. (2015). A Hybrid Recommender System Based on User-Recommender Interaction. Mathematical Problems in Engineering, 2015, Article ID: 145636. https://doi.org/10.1155/2015/145636
[131] Zadeh, L. A. (2015). The concept of a linguistic variable and its applications to approximate reasoning. Information Sciences, 8(4), 199-249.