# Enhancing Cloud Access Control: Leveraging Machine Learning for Security Score Prediction and Improvement

MSc Research Project
Master of Science in Cloud Computing

Anurag Singh
Student ID: x23180013

School of Computing
National College of Ireland

Supervisor: Aqeel Kazmi

# National College of Ireland
# Project Submission Sheet
# School of Computing

| | |
|---|---|
| **Student Name:** | Anurag Singh |
| **Student ID:** | x23180013 |
| **Programme:** | Master of Science in Cloud Computing |
| **Year:** | 2024 |
| **Module:** | MSc Research Project |
| **Supervisor:** | Aqeel Kazmi |
| **Submission Due Date:** | 29/01/2025 |
| **Project Title:** | Enhancing Cloud Access Control: Leveraging Machine Learning for Security Score Prediction and Improvement |
| **Word Count:** | 7512 |
| **Page Count:** | 20 |

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

**ALL** internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

| | |
|---|---|
| **Signature:** | |
| **Date:** | 29/01/2025 |

## PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

| | |
|---|---|
| Attach a completed copy of this sheet to each project (including multiple copies). | □ |
| **Attach a Moodle submission receipt of the online project submission**, to each project (including multiple copies). | □ |
| **You must ensure that you retain a HARD COPY of the project**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. | □ |

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

| **Office Use Only** | |
|---|---|
| Signature: | |
| Date: | |
| Penalty Applied (if applicable): | |

# Enhancing Cloud Access Control: Leveraging Machine Learning for Security Score Prediction and Improvement

Anurag Singh
x23180013

**Abstract**

Flexibility and scale have made cloud services essential for modern organizations, still, big security challenges surrounding access control management come with that. Role Based Access Control (RBAC) and Attribute Based Access Control (ABAC) serve as the fundamentals, however, existing methods are insufficient to describe the dynamic nature of the cloud from the perspective of cloud access control enforcement that changes along with it. In recent years, machine learning has been employed to improve cloud access control via anomaly detection, policy mining, and predictive analysis. Despite this, most previous solutions are limited to detection and do not integrate predictive solutions with actionable feedback for real-time remediation. To address these limitations, this research leverages predictive analytics alongside automated feedback to proactively remediate cloud access control by closing the accurate prediction and remediation gap. Applying the ML models to make security score predictions and feedback in access control configuration. XGBoost is the most robust model due to optimization across hyperparameters of RandomizedSearchCV. Through a novel feedback mechanism, vulnerabilities are automatically identified based on predictions. It consists of a simple yet functional user interface and Flask-based REST API which gives real-time insights and actionable recommendations. The solution is deployed on top of AWS services (Cloud9, S3, Elastic Beanstalk, CodePipeline) for scalability and convenient Integration. The results show proactive security management with reduced risk exposure and better compliance. Through the combination of predictive analytics with dynamic feedback, this research pushed the boundaries of cloud security and demonstrated that adaptive learning and cloud interoperability will be necessary for future development.

***Keywords***— Access Control, Machine Learning, Cloud9, S3, Elastic Beanstalk, XGBoost, Flask Framework, Hyperparameter Tuning, Auto-Remediation, CodePipeline, Compliance.

## 1 Introduction

The increasing popularity and utilization of cloud services have changed how organizations store, share, and protect their information. Being regarded as scalable, flexible, and cost-effective, cloud environments are now fundamental to contemporary business operations. But, with the shift towards cloud systems, there are new concerns and security vulnerabilities tied to the handling of cloud configurations like access controls that can grant permission to interact with applications and information. Organizations require an effective access control method because risks that may accompany cloud services include unauthorized access, data leakage, and non-compliance. However, original access control models such as RBAC and ABAC do not meet the demands coming from the cloud environment that is continuously evolving and should be scalable Nobi et al. (2022). This thesis proposes the implementation of various Machine Learning techniques in optimizing access control in cloud computing networks. New solutions in the field of ML present a revolutionary approach and enable building applications that meet real-time security requirements. The metrics used during the validation were precision, precision, recall, F1 score, Mean Time to Detect (MTTD), and Mean Time to Resolve (MTTR), to validate the effectiveness of the solution. Through proactive security metrics calculations, Machine Learning-based access control solutions can help organizations make better decisions about access control measures, thus minimizing the risks of unauthorized access to information and resources, and achieving compliance with the data protection requirements and regulations Cappelletti et al. (2019).

## 1.1 Background and Importance

Due to increased cloud adoption, it is imperative to identify far more sophisticated strategies for securing access and permission management. Past studies have established that the traditional approach of using static access control models to address the changes that occur in cloud environments is not effective Heaps et al. (2021). This is further compounded by the characteristics of the cloud infrastructure that are unique including multi-tenancy, and distributed data management, that require security models to be fast adapting while meeting high standards of security Sanders et al. (2019). Machine learning provides an innovative solution to these challenges which include; the ability to recognize patterns within big data, the ability to adapt to dynamic security threats, and the ability to develop robust security frameworks Karimi and Joshi (2018).

The dataset carries different forms of access control together with security scores meaning that Research on access control features that affect cloud security is enhanced. This way, the present thesis aims to help boost cloud security practices by using ML algorithms to anticipate these scores and provide suggestions for upgrading particular configurations. Of late, researchers have mostly investigated the configuration changes in an ML setting without assessing the transformability that can be operated by ML in cloud security systems Narouei and Takabi (2019). This research extends from these studies by constructing ML models that not only provide security scores but also recommend ways through which these scores can be improved.

## 1.2 Research Question and Objectives

This thesis poses the following research question: **Is machine learning applicable to predict security scores in cloud access control systems accurately enough to provide feedback that would allow for remedial actions ?**

To address this research question, the study pursues the following objectives:

- **Predict Security Scores :** The project applies XGBoost machine learning algorithms to produce security score predictions from access control configurations while detecting important security-related features.
- **Provide Feedback :** The system must include a mechanism that provides suggestions that present applicable modifications to enhance access control security ratings.

- **Build Scalable Infrastructure :** Flask will serve as the base technology to build a scalable back-end framework that links to an interactive visualization interface that displays security scores and results of feature analysis and security feedback.
- **Leverage AWS Services :** The system deployment depends on AWS tools including Cloud9 for development work and S3 for data storage along with Elastic Beanstalk for hosting and CodePipeline for CI/CD features.
- **Evaluate ML Models :** Assess the cloud security enhancement capability of these ML models by using accuracy, precision, recall, F1 score, MTTR, and MTTD.

This work tries to separate itself by combining prediction with feedback mechanisms, as it makes it possible for organizations to improve the definition of access control settings according to real-time information from the ML models studied by Liu et al. (2021) Xiang et al. (2019).

## 1.3 Contribution

This research presents an innovative ML-based solution to the problems of access control in cloud security, a novel solution that significantly surpasses conventional ways. The novelty of the study is in being able to not only predict security scores for access control configurations but also suggest actions to improve the scores. This work circumvents some of the longstanding unresolved challenges in cloud security practice by bridging the gap between predictive analytics and dynamic feedback.

Its key contributions, include the application and evaluation of multiple machine learning algorithms such as Decision Tree, Random Forest, Gradient Boosting, and XGBoost, to optimize access control mechanisms. In this regard, among these, XGBoost was found to outperform in terms of prediction accuracy and feature importance, rendering it the best fit for controlling complex and dynamic cloud environments. Finally, the research further refined the XGBoost model with hyperparameter tuning to ensure the scalability and adaptability of the model to real-world applications. This research extends the previous studies by adding the prediction and feedback focus into the model and provides a more suitable approach for access control in cloud environments since it is more dynamic and adaptable Yang (2019).

## 1.4 Structure of the Thesis

The thesis contemplates the interaction of cloud security and machine learning (ML) to handle dynamic complexity in access control systems. Role Based Access Control (RBAC) and Attribute Based Access Control (ABAC) based access control are traditional models, that cannot be able to adapt in a scalable, dynamic cloud environment which leads to inefficiency and security hazards. Then, using ML models this research aims to predict security scores as well as provide useful feedback on how to improve cloud access control. A methodology is developed for preprocessing cloud-based access data, feature selection, model development, and rigorous evaluation using metrics like accuracy, f1 score, mean time to detect (MTTD), and mean time to resolve (MTTR).

A REST API with a user-friendly interface for real-time insights is used for implementation with Flask and it runs from AWS services as it needs to be Scalable. The results show that predictive analytics to identify potential risks and automated remediation are leading to major improvements in proactive risk management and compliance. This paper bridges existing research gaps by combining prediction with feedback mechanisms to support dynamic adaptation to security threats. The work concludes with recommendations for future work including adaptive learning to support real-time policy updates feedback and integration with hybrid cloud ecosystems to create robust, data-driven security frameworks.

# 2 Related Work

The application of machine learning (ML) in cloud access control is important to deal with cybersecurity challenges. Traditional access control models, like Role Based Access Control (RBAC), Attribute Based Access Control (ABAC), Discretionary Access Control (DAC), and Mandatory Access Control (MAC) are the foundational security frameworks, but cannot address the dynamic, complex, and evolving security needs of modern cloud computing systems. These models are often manually provisioned, which leads to inefficiency and security risks, which are especially dangerous when used in large-scale systems where over-allocating and breaches are always possible Cotrini et al. (2018). Thus, as more attention is paid to cybersecurity, there is a trend in literature to develop more ML-based approaches toward improving the flexibility, efficiency, and robustness of access control systems Mohammad Nur Nobi (2022).

## 2.1 The Need for Automation in Access Control

Amid the growing use of Cloud environments, the problem of access control is quite exceptional due to the nature of Cloud environments, which are highly scalable, often having to accommodate numerous users and the expansive permissions they need that may not be readily manageable through scripts. Research shows that conventional access control solutions lose efficiency and become difficult to manage in environments where many objects are opened and closed Heaps et al. (2021). For example, Cotrini et al. (2018) designed an ML approach named Rhapsody, which can mine ABAC rules from sparse access logs and revealed that the application gap left by conventional models in policy mining and management can be conquered by the aid of ML. This type of self-governing rule generation and enforcement is critical for today's complex cloud infrastructures which see constant fluctuations in access patterns. Additionally, Maanak Gupta and Sandhu (2018) discussed how effective ML is in cloud security indicating that through analytical processing, the meaning of the access control logs may be performed more proactively using the ML techniques.

## 2.2 ML for Policy Decision-Making and Rule Mining

Studies have revealed that the machine learning algorithm is useful for automating policy decision-making and rule discovery in access control. Application of the ML-based models provides a dynamic solution together with the increasing amount of data to determine the access permissions necessary for protecting complex systems against unauthorized access. For instance, Cappelletti et al. (2019) have analyzed the integration of symbolic and non-symbolic methods of ML to solve policy-related decisions focusing on the ability of ML to monitor access control policies and modify them accordingly to challenge security threats. At the same time, Karimi and Joshi (2018) suggested improving policy mining in ABAC environments based on the ML-based clustering, which makes it possible to group similar roles or permissions, while it is helpful in terms of cloud administrators to simplify the process of policy creation and management. These papers demonstrate how achievable utilizing ML is to minimize the efforts to administrate policies, enhance security, and lessen reliance on human beings in access control systems.

## 2.3 Enhancing Security Through Predictive Analysis

A critical advantage of ML in demand is prediction ability of threats that would enable early actions to be taken against them. Writing on permission decision engines for ABAC, Liu et al. (2021) employed a Random Forest model to forecast access results from prior data, indicating elevated levels of accuracy in automatic access determination and the discovery of outliers. Such a strategy creates a constant check for likely security threats hence minimizing the chances of intrusions. In addition, Nobi et al. (2022) also analyzed the application of deep learning in access control to develop the DLBAC model that uses artificial neural networks instead of policy rules. This model can on its own, derive access control rules from the metadata of the user and resources and apply these rules to new cases with little assistance from humans. However, the authors do find that such models face difficulties in explainability, which is a problem with deep learning models due to the complexity of the models and how they might not be easy for users to understand.

## 2.4 A Shift Towards Real-Time, Proactive Security

In legacy access control solutions, access control is reactive, and a compliance check when something anomalous happens. This level of security while a good start, is not enough in this rapidly moving modern cloud environment. The reactive methods are inherently limited as these work to address vulnerability and breach following their having already offered a risk of possible delays in remediation and exposure to the threat. Machine learning is pushing the envelope towards real-time, real proactive access control. Using systems that perform predictive analytics, ML does the work of continuously watching access control configuration and security scores for changes that might indicate potential problems before those issues can be exploited. This real-time approach allows the dynamic modification of access policies to maintain security in the presence of evolving threats. Based on these advancements, this thesis develops such ML models through prediction and recommendation using a dataset of access control settings and their corresponding security scores. For instance, in its current configuration, predictive analytics will provide security scores based on the probability of vulnerability appearing. These predictions will inform administrative recommendations for immediate, data-driven adjustments to access control settings to improve access. This research therefore powerfully balances its assurance and its threat mitigation functions by enabling administrators to respond proactively using constant monitoring and real-time threat mitigation so that risks can be reduced and compliance preserved in changeable cloud environments as called for by Xiang et al. (2019).

## 2.5 Automating Remediation in Cloud Access Control Security Using ML

Machine learning promises to help solve another critical problem in the fast-paced cloud—automating the remediation of undesired access, beyond just improving detection and decisions in access control. Automated remediation is the dynamic response of an organization to detect security anomalies or impending threats via real-time adjustments of access policies, permissions, or configuration without human intervention. Studies, for example, Xiang, Feng and Wu (2019), have highlighted the need to integrate ML-based automation with predictive models to make proactive recommendations and corrective actions. For example, reinforcement learning is suggested as very suitable for learning self-adaptive systems that can control security by learning from environmental feedback Nguyen et al. (2021). In addition, Anwar and Khan (2020) used hybrid ML frameworks that integrate anomaly detection with prescriptive analytics for policy change recommendations that mitigate risks without administrative delays. Not only do they shorten response times but also minimize human errors, which are often the cause of cloud security breaches. The potential for automated remediation is significant, yet there remain challenges around their explainability, accountability, and understanding of how to maintain a balance between automation and administrative oversight to be sure these systems are secure and transparent.

## 2.6 Summary and Research Gap

In today's cloud security literature, most of the related research encompasses only those techniques based on post-incident reporting, such as anomaly identification or post-investigation examination. Conversely, this research takes a more preventive approach whereby the system tries to anticipate possible security risks based on a subject's access control parameters and suggest proactive feedback for security measures. This study presents an innovative idea by delivering coming Quantifiable Anticipated of Security, as well as emphasizing intermittent estimation, which contrasts itself with previous contingent methodologies. Such emphasis on the predictive, ML-based analytic data for access control demonstrates the existing change from traditional reactive security approaches to intelligent, data-based security approaches sufficient for current fluid cloud environments and services Di Giulio et al. (2017). The current literature review lays the ground concerning the application of machine learning to enhance access control in cloud systems. This paper also establishes the idea of applying ML more broadly than what has been provided by traditional access control constraints by considering the role of ML in automated decision-making in policy, and real-time monitoring, as well as statistical dangers, which enhance the cloud security protocols. Further work should build upon the proposal of utilizing symbolic and non-symbolic ML combined approaches to maximize the flexibility of policies and increase the transparency of the system Servos and Osborn (2017).

# 3 Methodology

This study selects a structured data approach, utilizing the ML framework to assess and optimize the access control frameworks deployed in a cloud networking environment. The methodology adopted is reproducible, valid, and done taking into consideration the best practices of research methodology, with each phase from data preprocessing to model testing and outcome analysis documented so that other researchers could replicate the study. As shown in the research methodology diagram below Figure 1

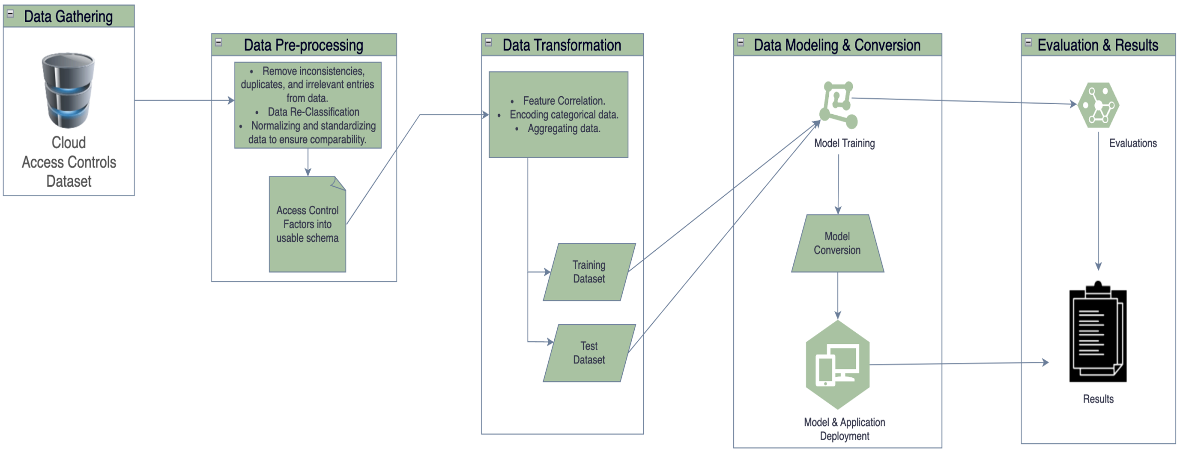

Data Gathering
Cloud Access Controls Dataset
Data Pre-processing
• Remove inconsistencies, duplicates, and irrelevant entries from data.
• Data Re-Classification
• Normalizing and standardizing data to ensure comparability.
Access Control Factors into usable schema
Data Transformation
• Feature Correlation.
• Encoding categorical data.
• Aggregating data.
Training Dataset
Test Dataset
Data Modeling & Conversion
Model Training
Model Conversion
Model & Application Deployment
Evaluation & Results
Evaluations
Results

Figure 1: Research Methodology Diagram

## 3.1 Data Collection and Preprocessing

The data for this study include access control settings and product security ratings in a cloud environment. Fifty factors that define access control assessments include but are not limited to authentication methods, regulatory, PAM, and multi-tenancy security. These factors are all related to the aspect that can be involved in access control which helps to provide various data sets that can be used to analyze security threats and risks that concern a system.

- **Data Selection:** The input dataset is designed to contain the necessary access control settings that exhibit the users' activities, systems' rights, and compliance information required for the analysis of security risks.
- **Categorical Encoding:** Factors affecting access control security in the cloud like user roles, access levels, authorization models, and so on are encoded categorically since non-numerical data have to be treated correctly before feeding the machine learning models.
- **Feature Correlation and Selection:** Feature selection is performed for validation of the correlation; and, therefore, which one feature influences scores in security to the greatest degree. LassoCV (Cross-validation) is useful especially in selecting features since it sets a preference of making unwanted feature coefficients zero, therefore erasing feeble features. While pursuing generalization, this thesis applies hyperparameter tuning with RandomizedSearchCV using the XGBoost model to enhance efficiency.
- **Data Standardization and Split:** The data are normalized to scale every value to a certain range needed for training the models that are sensitive to the scale of the features used like the gradient boosting models. The data set is subjected to a test and training split to determine the model accuracy over the unseen data set.

## 3.2 Machine Learning Model Development

The essence of this methodology is the training and assessment of several ML models with security score outputs, as well as vulnerability detection and suggestions for security enhancement. The following models are implemented, each selected for its suitability in access control evaluation within cloud environments:

- **Decision Tree:** This model has a way of making us understand our access control configurations since it shows how the different decision-making processes are arranged hierarchically in doing a security evaluation.
- **Random Forest:** Applied in its utilitarian fashion for variance reduction and enhanced predictive capability as well as for the ability to manage large complex datasets.
- **Gradient Boosting:** Applied for its capability of improving the model's accuracy by refining the inaccuracies present in subsequent models, suitable to perform predictive analysis on datasets containing access control data.
- **XGBoost:** Utilized for the state-of-the-art boosting method and feature selection that allows increasing the degree of prognostication and the efficiency of the model.

After every model has been trained to the processed data set, hyperparameters are then adjusted by using RandomSearchCV to overcome overfitting. The choice of hyperparameter and subsequently developed model ensures high efficiency in terms of predictive accuracy and simultaneously does not reduce the model's interpretability along with a reasonable computational complexity.

### 3.3 Prediction and Evaluation on Different Data Scenarios

After testing the predictive system in different data scenarios to check its robustness and adaptability. It first checks how well the model learns on the training data (already seen data). This dataset's performance validates the internalization of patterns in historical configurations by the model. The second is because it tests with unseen data (test data). After all, testing on unseen data is crucial to verify the model's generalization capabilities which are required to tackle real-world problems. The model's predictions are tested across a set of real-time scenarios, through manual data entry of hypothetical access configurations. The actual usability of this practical system is underscored in dynamic, real-world environments, whereby the system predicts security risks for configurations that it has never seen before. These also make sure the predictive system is under the need for a complete security framework.

## 4 Design Specification

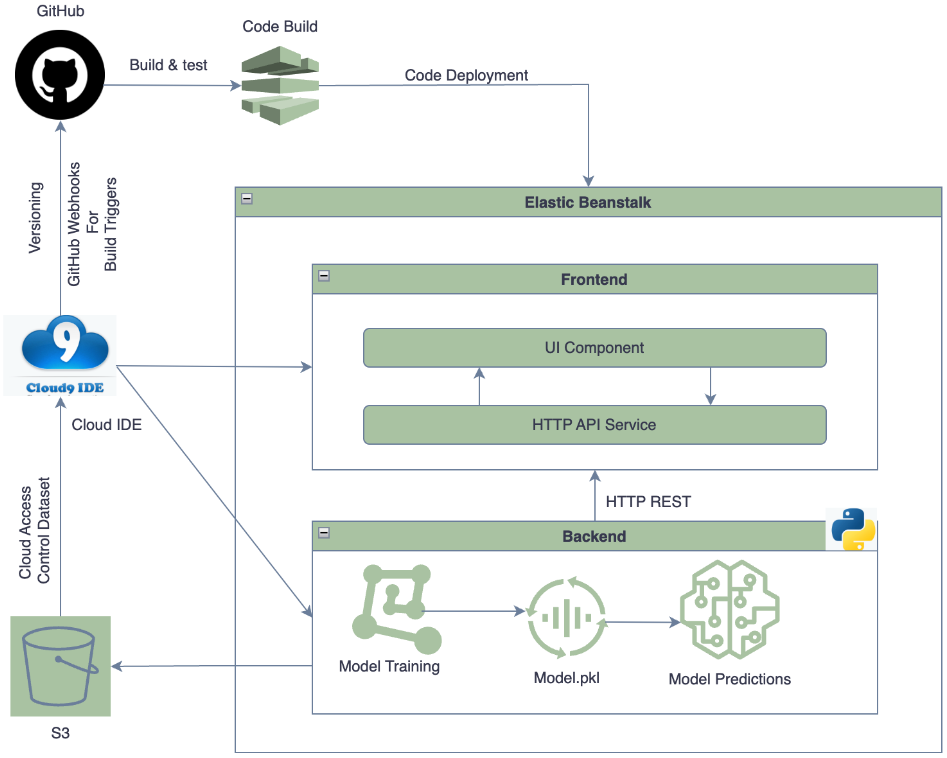


Figure 2: Research Architecture Diagram

The architecture, framework, and techniques are outlined in this section for the implementation of the cloud access control management proposed system. The design focuses on integrating predictive analytics with actionable feedback mechanisms such that dynamic security score prediction and automated remediation are possible. A machine learning (ML) driven framework is built around scaled cloud infrastructure and user-centric interfaces supporting this system. Figure 2

## 4.1 Data Processing Layer

Input data is prepared by the data processing layer for historical cloud access configurations along with security event logs. In this method, the data undergoes cleaning, normalization, and feature engineering using Python-based libraries like Pandas and Scikit. This standardizes the data so that it can be analyzed. One of the more critical components is feature selection which uses XGBoost's built-in feature importance feature and manual analysis to identify variables that have a large impact on security scores. This step helps steer the processed data to be accurate and relevant and gets us in a good position to build the machine learning models. Such methods are therefore compatible with state of art approaches as captured by their survey Mohammad Nur Nobi (2022) of applications of machine learning to access control.

## 4.2 Machine Learning Layer

The system consists of a machine learning layer that consists of model soldiers like Decision Tree, Random Forest, Gradient Boosting, and XGBoost. XGBoost stands out among these because it exhibits far superior predictive accuracy and also has very good feature ranking abilities. Finally, the model is tuned for hyperparameters with RandomizedSearchCV which further improves the model's generalization on diverse datasets. The input data is processed by this layer into security scores that represent the risk of access control configurations. Another XGBoost also helps to provide the importance of features, a meaningful information that can be used by the actionable feedback mechanisms. This approach extends the works of Maanak Gupta and Sandhu (2018) who spoke of ML as a means of improving access control within big data architectures.

## 4.3 Application and Deployment Layer

The system is operationalized in the application and deployment layer using a Flask Framework-based REST API and an interactive user interface. Real-time predictions and feedback generation API endpoints are provided for easy integration of the organizational workflows. By placing a graphical user interface on top of frameworks such as Bootstrap, CSS, and HTML, this work provides administrators the ability to visualize security scores for scoring feeds, interpret feature importance, and dynamically implement suggested remediation actions. Dealing with a massive cloud environment, its deployment is well controlled using AWS services such as Cloud9 for development, S3 for storage of cloud access control dataset as outlined by Chaudhuri (2024) in their analysis, Code Pipeline for smooth code integration, and Elastic Beanstalk for hosting and does allow for scalability and reliability.

## 4.4 Framework and Functional Requirements

For backend API development, I have chosen to use Flask, for infrastructure I have used AWS, and Bootstrap/HTML to create a user-friendly interface. In contrast, functional requirements mandate high predictive accuracy, real-time delivery of feedback, and the ability to scale appropriately for hybrid and large-scale cloud environments. Modularity is ensured by the system as well, which supports adding more models or tools in the future. Also, according to Nobi et al. (2022)vision on replacing classical models with automated deep learning solutions, modularity is a key feature, facilitating future integration of new models or more sophisticated ML techniques such as deep learning.

# 5 Implementation

First, the dataset including user permissions, security policies, and access-related data is stored in Amazon S3 (Simple Storage Service). It is an intermediary between data collection and processing and serves as S3, scalable and secure cloud storage. The dataset from S3 is processed in Google Colab, a cloud-based Jupyter notebook environment. Data analysis, visualization, and machine learning model training and development on the dataset are possible on Google Colab. This workflow in Figure 3 showcases the integration of data storage in S3 with computational power on Google Colab to enable quick, vigorous analysis of cloud access control data.

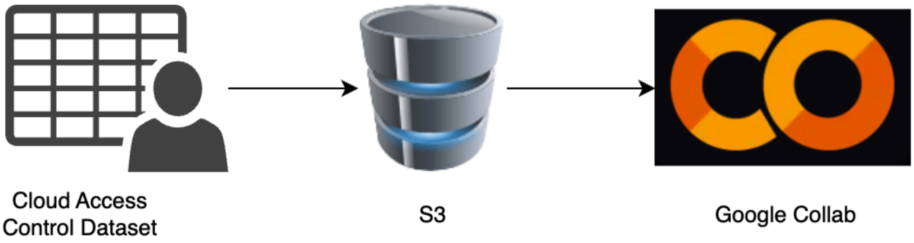
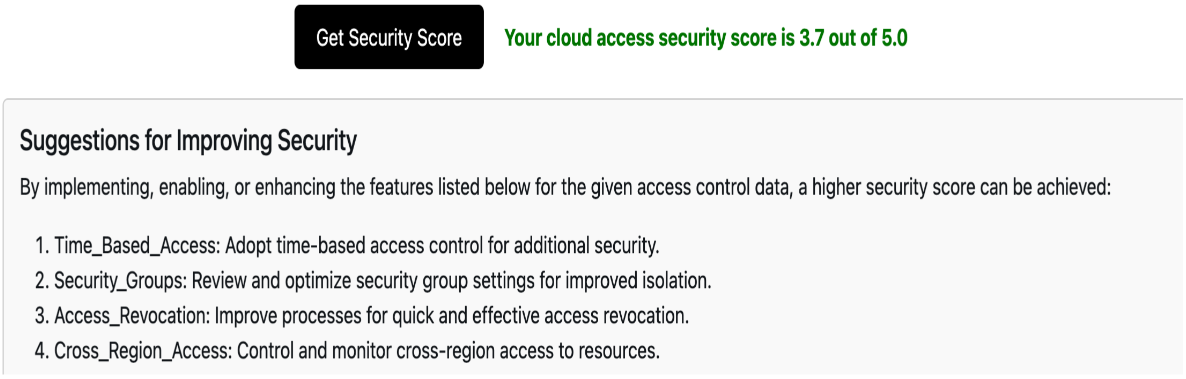


Figure 3: Integration of cloud storage and ML computational tools.

The final implementation applies the proposed solution as a machine learning (ML) driven system to predict the security scores and suggest actionable feedback based on which access control policies in cloud environments can be improved. Trained ML models, APIs for prediction and feedback as well as a user-friendly interface for interacting with the system are the output. To address both predictive accuracy and adversarial vulnerabilities, the tools and technologies were selected in such a way that scalability, security, and applicability in the real world are ensured. Preprocessing was done to the transformed data extracted from public repositories to remove inconsistencies and improve model learning. The models were engineered to capture the access control analysis requirements like role usage pattern, and access frequency to feature align with the nuances of cloud security configurations. XGBoost showed consistent accuracy and feature interpretability, which is in line with the findings of Hassan (2022) who suggest the need for robust ML models to manage security posture. For model development the implementation used Python and to build a REST API Flask was used. The backend for delivering real-time security predictions and feedback is served using this API. First, a Bootstrap-based graphical user interface (GUI) for inputting policies that are enabled or disabled that impact cloud access control security is a key feature of the system.

This model predicts the security score by the input of these features and provides feedback to dynamically improve the score as shown in this Figure 4 Figure 5. This interactive capability facilitates the administrator's testing and refining access configurations in real-time to improve the overall cloud security in line with Ait el Hadj (2023) recommendations concerning the use of ML in access control to improve the real-time. Decisions like the use of S3 to store the data and Cloud9, CodePipline, and EC2 services for model deployment, and ElasticBeanstalk to host the application-enabled solution of the data management and scalability problems. Finally, the system was evaluated in terms of black-box attack vulnerability and adversarial query sensitivity, resembling methods proposed by Usama et al. (2019). Risk of model extraction was mitigated Reith et al. (2019) by using secure API endpoints and encryption protocols.



Your cloud access security score is 3.7 out of 5.0

Suggestions for Improving Security

By implementing, enabling, or enhancing the features listed below for the given access control data, a higher security score can be achieved:

1. Time_Based_Access: Adopt time-based access control for additional security.
2. Security_Groups: Review and optimize security group settings for improved isolation.
3. Access_Revocation: Improve processes for quick and effective access revocation.
4. Cross_Region_Access: Control and monitor cross-region access to resources.

Figure 4: User Interface for security score enhancement suggestions.

## 5.1 Feedback Mechanisms

As a result of the predictions and the security scores, this research proposes specific protocols that will help improve the security of access control in cloud environments. With granular access controls, implemented as Gupta et al. (2017) suggest by not overspecifying permissions, you narrow the attack surface down to what is strictly necessary by roles. Frequent monitoring increases security while banning activities from high-threat regions further reduces risks from unauthorized access. The work of Nobi et al. (2022) stresses the importance of automated access revocation policies that adapt and change permissions dynamically based on real-time threat analysis. Cappelletti et al. (2019) also point out that simple penetration testing and vulnerability assessment are also critical to uncovering and solving the rising security holes, in particular within dynamic cloud systems. They fit with best practices for cloud security management through both proactive monitoring and adaptive (re)learning using machines.

This methodology relies on a comprehensive dataset of the connected devices coupled with ML techniques to overcome the inability of traditional access control systems. This study follows a methodological rhythm by describing each step of the research and thus provides reproducibility and enhanced confidence in the findings. Additional layers of security could be created in exploring frameworks for detecting anomalous configurations in access policies van Ede et al. (2022). Furthermore, it would also look into the system's efficacy in comparative studies of the cloud control standards, as done by Hegde et al. (2024). Ultimately, move one step further and incorporate the full suite of comprehensive cybersecurity tools as suggested by Coppola et al. (2023), to frame the challenges in cloud security posture management in an encompassing framework.

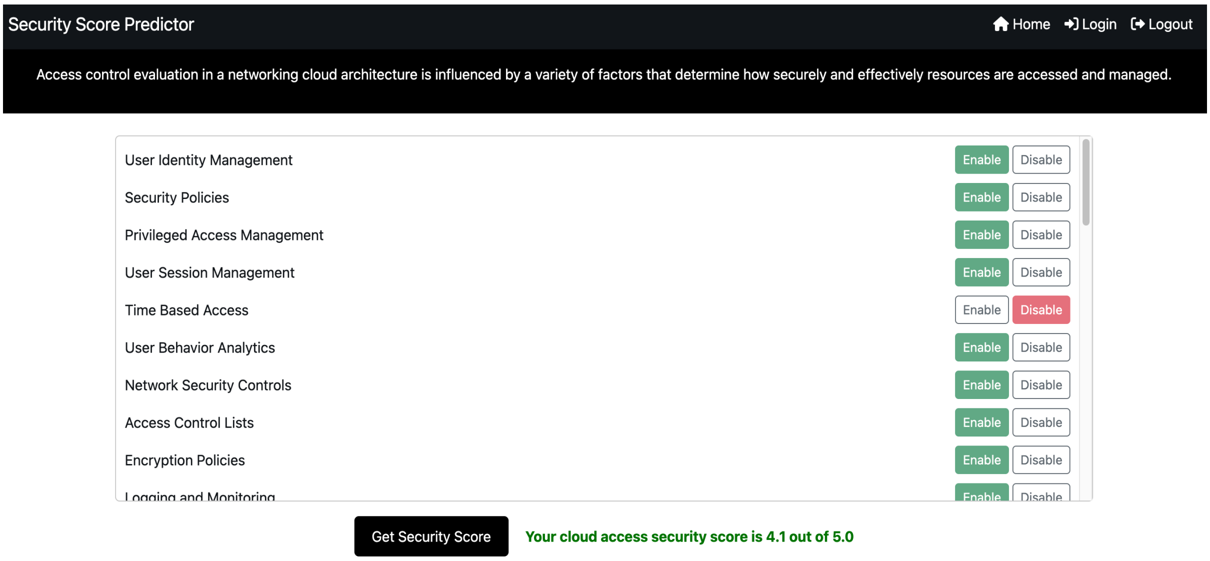
The feedback system reviews the predictions of the ML model to produce specific recommendations that strengthen the access control configurations. The system uses XGBoost's native feature importance capabilities to detect security-score influencing elements which include user permissions in addition to access frequency and role complexity. The system provides administrators with detailed insights about important elements that require their immediate focus through its ranking mechanism. The analysis produces dynamic feedback that points out security issues such as overpermitted access weak authentication or role redundancies, so the system will present specific resolution steps. These recommendations promote security through digital identity solutions that involve role deactivation and MFA deployment with the refinement of excessive access parameters.

The system implements a seamless feedback framework through its frontend Bootstrap

HTML and backend Flask API interactive web-based user interface which prioritizes usability needs. Through this interface, administrators receive real-time security scores with clear visual interpretations of important features. Through simulated impact analysis administrators can verify how different configuration changes impact security scores using this system. The detailed approach makes the feedback both data-centered and actionable, yet makes it available to organizations so they can actively improve their access control settings.

## 5.2 Prediction Process

To achieve the above goals, the system is based on the fact that XGBoost itself is capable of conveying generalization: it can predict security scores accurately and it provides recommendations for better cloud security. This functionality has a critical piece in there, that is feature selection, where feature importance ranking generated by XGBoost indicates the features that are the most correlated with security risks. The model is then refined in this way: the dataset is refined, reducing noise and directing model attention to what matters most. Using RandomizedSearchCV, a robust approach to implement hyperparameter tuning to explore the best hyperparameter combinations efficiently without exhaustive computation to optimize model performance. By doing so this work prevents overfitting and avoids loss of predictive accuracy on diverse cloud configurations. Training is done both on the models and for multiple models, they compare XGBoost against others like Random Forest and Gradient Boosting to ensure the highest performance. The best deployment model is selected based on the metrics of accuracy, F1 score, and AUC-ROC. It manages a large dataset quite well and is extremely good with imbalanced classes, which are important for cloud security. The model takes in configurations and produces a security score, and based on feature importance ranks it suggests configurations in the prediction phase.



Access control evaluation in a networking cloud architecture is influenced by a variety of factors that determine how securely and effectively resources are accessed and managed.

| | |
|---|---|
| User Identity Management | [✓ Enable] Disable |
| Security Policies | [✓ Enable] Disable |
| Privileged Access Management | [✓ Enable] Disable |
| User Session Management | [✓ Enable] Disable |
| Time Based Access | Enable [✓ Disable] |
| User Behavior Analytics | [✓ Enable] Disable |
| Network Security Controls | [✓ Enable] Disable |
| Access Control Lists | [✓ Enable] Disable |
| Encryption Policies | [✓ Enable] Disable |
| Logging and Monitoring | [✓ Enable] Disable |

Your cloud access security score is 4.1 out of 5.0

Figure 5: User Interface for security score prediction.

For instance, it might identify a high level of permissions, or a lack of good access controls and suggest specific actions to improve the security score. These insights let administrators proactively deploy risk mitigation resources in a focused, data-driven way to help enhance the security of cloud environments. This mindset fits Heaps et al. (2021) proactive methodologies tied to machine learning beyond when it detects a violation and instead offers actionable insights.

## 5.3 Ethical considerations

This section analyses the ethical issues that surfaced in the constitutions of this research. Data used was sourced from Kaggle a public repo site that has open source data and is used to comply with copyright and usage regulations Truong et al. (2020). For this research, only public datasets are exclusively used. The datasets included no personally identifiable information (PII) hence complying with global data protection laws including the General Data Protection Regulation (GDPR). All datasets were stripped of any identifying information to maintain anonymity and reduce the possibility of misuse before they were analyzed. This was only data used for research and the focus was only to develop and evaluate machine learning models for the cloud access control leaving no room for ethical breaches. By using AWS services — S3 for secure storage and encryption and Google Collab for secure machine learning model training and deployment — secure processing and management of the data were also facilitated Zulkifley et al. (2020). These were good safeguards during the research process to guarantee solid security and compliance.

The design builds on the principles of ethical design and guardrails for strict data governance, creating trust and accountability in which the system can operate within ethics and law and within the limits of ensuring the security of cloud environments. Furthermore, Qayyum et al. (2021) recommendations for having secure and robust ML systems also support this system's continuous monitoring to prevent and handle the occurrence of potential bias in predictions.

# 6 Results and Discussion

In this section, machine learning models are evaluated to predict security scores and provide actionable feedback to enhance cloud access control systems.

## 6.1 Model Evaluation

Evaluation of the models is done using metrics such as accuracy, balanced accuracy, precision, recall, F1 score, mean time to detect (MTTD), and mean time to resolve (MTTR) and the implications on cloud security in the given table 1 below. The research calculated various metrics to measure the predictive accuracy and duty cycle (responsiveness to security threats) of the machine learning (ML) models to evaluate their performance. Such a fundamental measure of accuracy is defined as how often the model makes the right prediction and shows how reliable the model is. In addition, this paper uses precision, recall, and F1 scores to assess the quality of the models at finding vulnerabilities without being too lenient or overly strict. These metrics are important to minimize security risks, ensuring that the model sees real threats but numerical disruptions from unnecessary false alarms. It takes the Mean Time to Detection (MTTD) and Mean Time to Remediation (MTTR) from the system's reaction in recognizing a threat and proposing remediation accordingly. The model's performance is synchronized to the real-time security demands of the cloud environment.

| Model | Accuracy | Balanced Accuracy | Precision | Recall | F1 Score | MTTR (secs) | MTTD (secs/sample) |
|---|---|---|---|---|---|---|---|
| Decision Tree | 0.7873 | 0.4790 | 0.7933 | 0.7873 | 0.7902 | 2.2393 | 0.000001 |
| Random Forest | 0.8458 | 0.3719 | 0.8687 | 0.8458 | 0.7911 | 17.7755 | 0.000033 |
| Gradient Boosting | 0.8996 | 0.4759 | 0.9090 | 0.8996 | 0.8820 | 67.3864 | 0.000012 |
| XGBoost | 0.9671 | 0.6089 | 0.9664 | 0.9671 | 0.9653 | 11.7821 | 0.000027 |

Table 1: Performance evaluation of various machine learning models.

## 6.2 Evaluation of ML Algorithms to Improve Security

In this research ML algorithms are used to improve security in cloud access control systems by dynamically predicting the security scores and making the improvements actionable. The data preprocessing started with encoding categorical access control configurations, and security scores with techniques like categorical encoding for user roles, access levels, and authorization methods. To select the best variables for security, the research used LassoCV and XGBoost's in-built feature importance. XGBoost showed the best performance due to its high accuracy, favorable handling of complex datasets, and preventative treatment against overfitting with hyperparameter tuning using RandomizedSearchCV.

The system was integrated with a feedback mechanism that allowed for the analysis of predictions to identify vulnerabilities and suggest concrete actions to enhance access control configurations, reduce unnecessary permissions, and ensure tightly enforced authentication policies. The system included automated remediation provided by ML predictions, and the policies were dynamically changed as new threats arose. Scalability and operational reliability were ensured by deployment on AWS infrastructure and its effectiveness was validated through metrics including precision, recall, F1 score as well as Mean Time to Detect (MTTD) and Mean Time to Resolve (MTTR). Access Control was transformed using this ML-driven approach to become a proactive, real-time system that can adapt to changing security needs.

## 6.3 Implications and Comparative Analysis

The performance of various models is compared in terms of accuracy, balanced accuracy, precision, recall, F1 Score, and Latency (MTTR) graphs, showing significant differences between metrics on different parts of the graph.

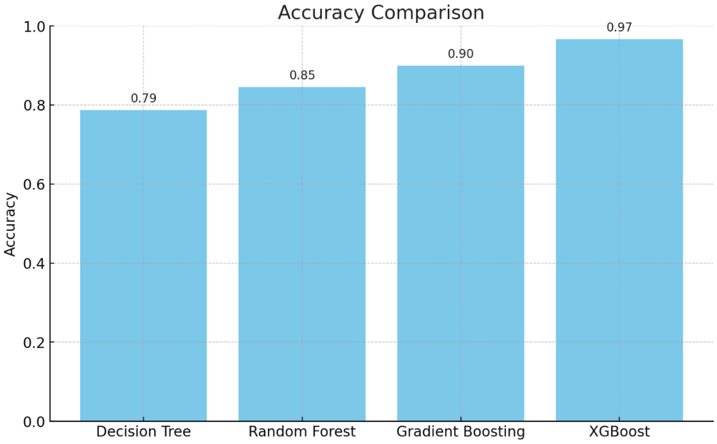


Figure 6: Accuracy comparison of machine learning models

In Figure 6 Gradient Boosting (89.96%), Random Forest (84.58%), and Decision Tree (78.73%) are all outperformed by the XGBoost model a single point, giving it a performance of 96.71%. This demonstrates the stability of XGBoost in performing precise predictions for security scores. XGBoost is different because of its ability to remember complicated data patterns. Unlike models such as the Decision Trees and Random Forests, XGBoost uses additive learning, each time it builds a new tree to correct previous tree errors. With the regularization approach of L1 and L2, this approach prevents overfitting.

As a bonus, XGBoost is also rather efficient, using parallelized tree building and optimization to deal with large datasets quickly and well. In the figure XGBoost achieves an accuracy of 96.71%, beating Gradient Boosting (89.96%), Random Forest (84.58%), and Decision Tree (78.73%). It is further improved in its performance and adaptability by using a hypertuned XGBoost model. Resources utilization and efficiency can also be improved by changing computational parameters such as subsample and colsample by tree. This is especially important in cloud security assessments where precise predictions are essential or in applications that require high computational efficiency. To sum it up, hypertuning exposes XGBoost's full potential, the default capabilities are already outstanding but with hypertuning, you will get the most accurate, fastest, and reliable model training and prediction in the critical domain.

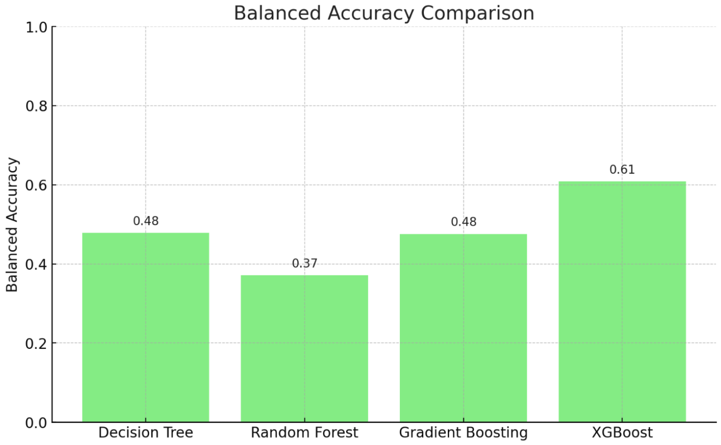


Figure 7: Balanced accuracy of machine learning models, showing their performance when accounting for imbalanced datasets by averaging sensitivity and specificity.

In balanced accuracy as shown in the Figure 7, XGBoost beat other models giving them a score of 60.89%, while other models did poor handling of the imbalanced dataset. This implies a much stronger ability to handle cases with different amounts of positive and negative classes. In Figure 8 the F1 score for XGBoost is 96.53%, the best amongst all reported models, to indicate that it has balanced precision and recall. On the side of latency as well, it showed a latency of around 11.78 seconds (MTTR), this being slightly better than Decision Tree's 2.24 seconds, but much better than Gradient Boosting's 67.38 seconds. This shows that XGBoost can make both timely and accurate predictions.

The graph clearly shows a trade-off between latency (Mean Time to Resolve, MTTR) and model performance (F1 score). It achieves the highest F1 score of 96.53% thus proving its superior balance of precision and recall. The balance in this case proves that the model can still find truly positive, whilst avoiding false positives or false negatives, for example when making security predictions. Compared to these other models (Gradient Boosting and Random Forest) F1 scores get lower, which means there is not as robust performance.

Another important factor is Latency. XGBoosts 11.78 seconds latency is significantly less than a Gradient boosting 67.38 seconds. As a result, the fast return of XGBoost helps to provide predictions quickly, which is important for applications that have to wait near live or real-time. Though Decision Tree is a bit faster (2.24 seconds) than the other models, with a worse F1 score of 79.02%, it is poorer at predicting. This equilibrium between high F1 score and low latency makes XGBoost optimal for those applications where Accuracy and timeliness are of top priority. XGBoost is reliable in using rapid and precise answers in fields such as cybersecurity, which are characterized by a need for rapid and swift responses to make proactive decisions.

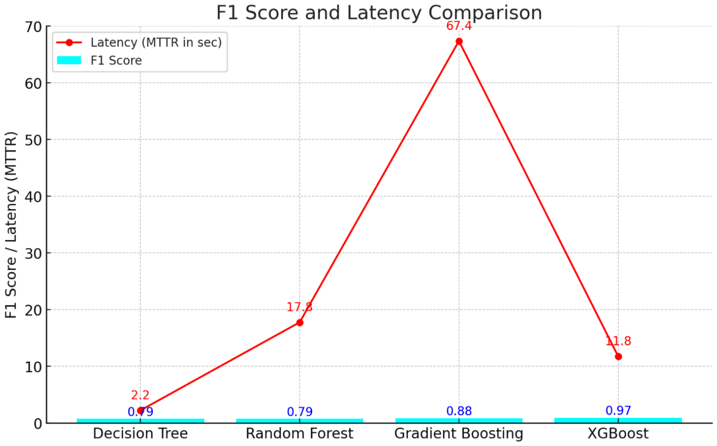


Figure 8: F1 score combines precision and recall into a single metric, while latency(MTTR) indicates the model's processing time in seconds.

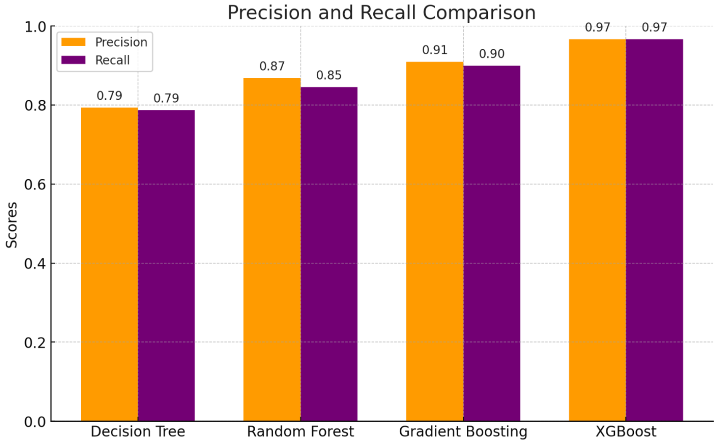


Figure 9: Precision and recall scores for each model, demonstrating their ability to handle false positives and false negatives respectively.

This Figure 9 compares precision and recall scores across four machine learning models. Recall measures the ability to capture all actual positive cases and precision measures the ability of a model to identify positive instances out of all correctly predicted positive cases. The XG Boost model has the highest scores of 97% both for precision and recall. Since it is the one that can very well minimize false positives and false negatives, it implies that this. Precision and recall metrics are at 91% and 90%, respectively, which shows gradient boosting does well in getting errors balanced. Since the Random Forest model has a slightly higher recall (87%) than its precision (85%), a bit more false positives and a little less true positives. The higher the results, the better the model does at predicting whether a given case is positive or not, with Precision of 79% and Recall of 79%, the Decision Tree model ranks the lowest, having the least ability to identify positive cases.

## 6.4 Error Analysis

Despite the performance gain, the training metrics of the XGBoost model slightly exceeded those of its testing metrics in Table 2. Error analysis of the XGboost model's performance on the training and testing dataset is also discussed in the section. Being a capable model that can predict very well, as seen with the training accuracy of 99.69% and the testing accuracy of 96.71%, the model is still an overfitted one. The higher training metrics indicate overfitting: a model climbs the training data like a Ferrari, then slightly less on unseen testing data. Impressive, however, poor balanced accuracy scores of 37.19% of Random Forest and 47.90% of Decision Tree models are then observed for imbalanced datasets. In datasets with different class distributions, balanced accuracy is very important as it calculates the model's capacity to make correct predictions in minority classes. These models are seen to have poor balancing accuracy, indicating their inability to generalize to the under-represented classes.

While XGBoost performed better than other models, error analysis shows that current imbalanced data handling poses challenges. Additionally, although Gradient Boosting achieves the highest F1 score and accuracy among all other builders, its high computational latency makes it unsuitable for real-time systems. It results in slight overfitting and shows that the generalization capability on an imbalanced dataset has to be improved.

| Model | Accuracy | Balanced Accuracy | Precision | Recall | F1 Score | Mean Squared Error |
|---|---|---|---|---|---|---|
| TRAIN | 0.9969 | 0.7458 | 0.9969 | 0.9969 | 0.9969 | 0.0030875 |
| TEST | 0.9671 | 0.6089 | 0.9664 | 0.9671 | 0.9653 | 0.0329 |

Table 2: Prediction evaluation metrics for the XGBoost model on train and test datasets.

## 6.5 Discussion

These results suggest the importance of using machine learning models that are both accurate adaptable and efficient in dealing with the challenges of real-world cloud environments. The paper demonstrates that XGBoost is a strong performer on a wide range of performance metrics, and therefore a strong candidate for following up on like predicting security scores or recommendations of change. Its robustness comes from its capability to achieve good value of precision and recall as well as both of them, resulting in producing reliable cloud security decisions with high reliability. The unique aspect of this approach is that a built-in feedback mechanism is included via feature importance analysis. This enables to figure out which configs could drive the biggest increase in the security score saving time to improve compliance.

XGBoost can be deployed at scale as a top enterprise solution to real-time data-driven security improvements for large infrastructures running on Cloud services in AWS. XGBoost still faces some limitations, most noticeably the need to prevent overfitting through methods that are prone to overfitting. In the future, these techniques need to be fine-tuned to generalize better across diverse cloud environments. Additionally, investigating how adaptive learning could influence the model to continuously adapt to ever-changing security threats would allow the model to react to new threats in real-time. Furthermore, the solution could be expanded to a hybrid cloud ecosystem and further strengthened by functioning simultaneously in different operational environments. By making this more applicable, it would also become more resilient to a broader set of security challenges. This allows it to deliver significant enhancements in accuracy, efficiency, and versatility, which aligns with increasing cloud environment needs to delivers a real-time solution for securing cloud infrastructures.

## 6.6 Proposed Enhancements

Potential future improvements to this system would take advantage of adaptive learning approaches to handle real-time policy changes in response to emerging security threats. The system asymmetrically integrated reinforcement learning to continuously refine its predictive and prescriptive capabilities, tailored to newly emergent risk landscapes in the wild. Enabling cross-cloud interoperability to allow for policy management across diverse hybrid and multi-cloud environments would need to extend this adaptability to different hybrid and multi-cloud environments. It could also solve one of the gravest cloud security challenges of all: providing consistent and coordinated protection over collections of increasingly interdependent infrastructure. Adding natural language processing (NLP) to policy interpretation and remediation can support an innovative enhancement.

The system could translate technical insights into mechanical recom- mendations to empower non-technical stakeholders including business executives or compliance officers to make an informed decision. It not only adds to the usability of the system but is essential to getting greater collaboration of both technical and non-technical teams to develop systems that will protect enterprise-scale environments. Furthermore, the drawback inspiration of nature-inspired frameworks as proposed inNarouei and Takabi (2019) might provide new optimization techniques. Placing savings into evolutionary algorithms and swarm intelligence that effectively optimizes access control policies, the system would be more efficient and deal with more complex threat patterns effectively. These bio-inspired strategies could allow for adaptive processes to simulate adaptive processes where the system could self-organize and respond autonomously to security challenges.

# 7 Conclusion and Future Work

This research presented the solution of a critical challenge to make a dynamic and scalable access control configuration management in cloud computing. The goal of the work was to accurately predict security scores, provide actionable feedback for remediation, and deliver automated solutions for the improvement of the cloud access control policies in an integrated fashion dynamically. This paper has conducted a comparative analysis of various models on ML and found out that XGBoost was performing much better than the other models to take it a notch higher, XGBoost's hyperparameters are optimized using RandomizedSearchCV. As a result, a robust model capable of accurately making predictions and useful recommendations for improving cloud security configuration was formed.

Finally, the research showed practical utility by using the Flask framework for a convenient REST API coupled with a simple user interface and optimal deployment on AWS services. The proposed solution provides a proactive security management framework that mitigates possible risks, improves compliance, and gives cloud administrators data-driven insight. This work advances cloud access control security from the current reactive paradigm to a predictable and adaptive one through the fusion of predictive modeling and automated remediation.

There are exciting possibilities for future work to extend this framework's capabilities. Implementation of adaptive learning mechanisms enables new real-time policy changes and evolving threat patterns to be accommodated, enabling the system to adapt to continually shifting security landscapes. Another promising avenue lies in cross-cloud interoperability that would allow consistent form factor control across hybrid cloud ecologies, and mitigate risks associated with multi-cloud ecosystems. Furthermore, incorporating natural language processing (NLP) within the feedback mechanism would allow for policy interpretation to be accessible to non-technical stakeholders; collaboration across organizational domains would be invited. This research lays the foundation for a new paradigm in cloud security that is predictive, automated, and universally accessible. If enhanced, the system would change the paradigm of security management in cloud computing and provide a future intelligent scalable solution to the ever-increasing complexities of modern Cloud infrastructures.

# References

Ait el Hadj, M. (2023). Exploring the role of machine learning in enhancing access control systems: A comprehensive review, *International Journal of Computing and Digital Systems* .

Anwar, M. and Khan, A. (2020). Hybrid machine learning frameworks for automated security remediation in cloud environments, *Computers & Security* p. 101783.

Cappelletti, L., Valtolina, S., Valentini, G., Mesiti, M. and Bertino, E. (2019). On the quality of classification models for inferring abac policies from access logs, *2019 IEEE International Conference on Big Data (Big Data)*, pp. 4000–4007.

Chaudhuri, M. (2024). Cloud Access Control Parameter Management, `https://www.kaggle.com/datasets/brijlaldhankour/cloud-access-control-parameter-management/data?select=access_control_dataset_with_security_score.csv`. [Accessed 10-11-2024].

Coppola, G., Varde, A. S. and Shang, J. (2023). Enhancing cloud security posture for ubiquitous data access with a cybersecurity framework based management tool, *2023 IEEE 14th Annual Ubiquitous Computing, Electronics  Mobile Communication Conference (UEMCON)*.

Cotrini, C. et al. (2018). Rhapsody: Mining abac rules from sparse access logs, *ACM Transactions on Information and System Security* (4): 1–26.

Di Giulio, C., Sprabery, R., Kamhoua, C., Kwiat, K., Campbell, R. H. and Bashir, M. N. (2017). Cloud standards in comparison: Are new security frameworks improving cloud security?, *2017 IEEE 10th International Conference on Cloud Computing (CLOUD)*, pp. 50–57.

Gupta, M., Patwa, F. and Sandhu, R. (2017). Object-tagged rbac model for the hadoop ecosystem, pp. 63–81.

Hassan, Ahmad, U. (2022). Challenges and solutions in cloud security posture management: A systematic review, *Comput. Secur.* .

Heaps, J., Krishnan, R., Huang, Y., Niu, J. and Sandhu, R. (2021). Access control policy generation from user stories using machine learning, *in* K. Barker and K. Ghazinour (eds), *Data and Applications Security and Privacy XXXV*, Springer International Publishing, Cham.

Hegde, T., Gangl, J., Babenko, S. and Coffman, J. (2024). Cloud security frameworks: A comparison to evaluate cloud control standards, Association for Computing Machinery.

Karimi, M. and Joshi, A. (2018). Enhancing policy mining with ml-based clustering in abac environments, *Information Security Journal: A Global Perspective* pp. 100–112.

Liu, L. et al. (2021). A permission decision engine scheme for abac using random forests, *Journal of Cybersecurity and Privacy* pp. 98–110.

Maanak Gupta, F. P. and Sandhu, R. (2018). An Attribute-based Access Control Model For Secure Big Data Processing In Hadoop Ecosystem. In ACM Workshop on Attribute-Based Access Control.

Mohammad Nur Nobi, Maanak Gupta, L. P. R. K. (2022). Machine Learning in Access Control: A Taxonomy and Survey, `https://ar5iv.labs.arxiv.org/html/2207.01739`.

Narouei, M. and Takabi, H. (2019). Improving abac policies with machine learning, *Journal of Information Security* pp. 210–219.

Nguyen, H., Pham, Q. and Do, T. (2021). Reinforcement learning for adaptive security management in cloud computing, *ACM Transactions on Autonomous and Adaptive Systems* pp. 1–23.

Nobi, M. N., Krishnan, R., Huang, Y., Shakarami, M. and Sandhu, R. (2022). Toward deep learning based access control, *Proceedings of the Twelfth ACM Conference on Data and Application Security and Privacy*, Association for Computing Machinery, p. 143–154.

Qayyum, A., Qadir, J., Bilal, M. and Al-Fuqaha, A. (2021). Secure and robust machine learning for healthcare: A survey, *IEEE Reviews in Biomedical Engineering* pp. 156–180.

Reith, R. N., Schneider, T. and Tkachenko, O. (2019). Efficiently stealing your machine learning models, Association for Computing Machinery, p. 198–210.

Sanders, G. et al. (2019). Enhancing access control with ml for large-scale enterprises, *Computers Security* pp. 101–115.

Servos, D. and Osborn, S. L. (2017). Abac and its extension into cloud environments, *Computers & Security* pp. 75–90.

Truong, Q., Nguyen, M., Dang, H. and Mei, B. (2020). Housing price prediction via improved machine learning techniques, *Procedia Computer Science* pp. 433–442.

Usama, M., Qayyum, A., Qadir, J. and Al-Fuqaha, A. (2019). Black-box adversarial machine learning attack on network traffic classification, *2019 15th International Wireless Communications  Mobile Computing Conference (IWCMC)*, pp. 84–89.

van Ede, T., Khasuntsev, N., Steen, B. and Continella, A. (2022). Detecting anomalous misconfigurations in aws identity and access management policies, Association for Computing Machinery.

Xiang, S., Feng, Y. and Wu, T. (2019). Real-time cloud security frameworks based on predictive analytics, *Journal of Cloud Computing* pp. 245–258.

Xiang, X. et al. (2019). Policy monitoring in access control systems using machine learning, *Security and Privacy Journal* **10**(5): 378–390.

Yang, Y. (2019). Decision tree applications in access control for cyber-physical systems, *IEEE Internet of Things Journal* pp. 1753–1763.

Zulkifley, N. H., Rahman, S. A., Ubaidullah, N. H. and Ibrahim, I. (2020). House price prediction using a machine learning model: A survey of literature, *International Journal of Modern Education and Computer Science* pp. 46–54.