INTEGRATING ROLE AND ATTRIBUTE-BASED ACCESS CONTROL WITH PREDICTIVE ANALYTICS FOR DYNAMIC ACCESS DECISIONS

Basireddy Vijaya Lakshmi[1], Dr. Nagarjuna Karyemsetty[1]
[1] Department of Computer Science and Engineering,
Koneru Lakshmaiah Education Foundation, Guntur, India
Emails: vijayabasireddy.2002@gmail.com, nagarjunak@kluniversity.in

ABSTRACT

Access control plays a vital role in information security, ensuring that only authorized individuals can access specific resources. This study introduces a hybrid access control model that merges Role-Based Access Control (RBAC) with Attribute-Based Access Control (ABAC) to improve both flexibility and scalability. RBAC streamlines management by assigning permissions according to defined roles, whereas ABAC provides detailed access control based on user attributes, resource characteristics, and environmental factors. By combining these two models, the proposed method overcomes the shortcomings of traditional RBAC systems while still allowing for effective management. The paper outlines the implementation, architecture, and assessment of the hybrid RBAC-ABAC model, showcasing its ability to strike a balance between security and usability.

I. INTRODUCTION

Access control mechanisms have become crucial for protecting sensitive data and resources in various domains such as enterprise systems, cloud computing, and IoT environments. Traditional Role-Based Access Control is widely used for the simplicity and ease of administration. In RBAC, permissions are assigned to roles, and users inherit permissions [2,3,4,5] based on their assigned roles. However, RBAC is not flexible [1,7] enough to address dynamic and context-based access requirements.

ABAC, or Attribute-Based Access Control uses attributes such as user attribute, resource attribute, environmental conditions, and so forth to make access control decisions. ABAC provides fine-grained as well as context-aware access control as suitable for modern systems with challenging security requirements. However, for the same reason, when it comes to scale issues, ABAC can also be quite difficult to administer.

This paper proposes a hybrid access control model that combines the simplicity of RBAC with the flexibility of ABAC. The hybrid model combines the simplicity of RBAC with the flexibility of ABAC [1] to address the shortcomings of each approach. This solution is evaluated in terms of scalability, security, and usability.

II. LITERATURE SURVEY

Hybrid access control research has focused on integrating Role-Based Access Control (RBAC) and Attribute-Based Access Control (ABAC) to improve security, flexibility, and policy granularity. Ameer et al. [1] applied an ABAC-RBAC model in smart home IoT systems, enhancing adaptability and security, while Long and Yan [2] introduced RACAC to unify policy validation with improved usability. Jyothsna et al. [3] confirmed the feasibility of such hybrids using custom access policy datasets.

In multi-domain settings, Attia et al. [4] combined ABAC and RBAC for multimodal applications, and Varadharajan et al. [5] proposed RC-ABAC to strengthen role-attribute integration. FRABAC by Attia et al. [6] and the model by Hasiba et al. [7] improved interoperability, scalability, and policy management. Pal and Jadidi [8] explored IoT protocol-based hybrids, identifying gaps, while Fatima et al. [9] compared ABAC and RBAC, highlighting hybrid advantages. Kaiwen and Lihua [10] enhanced IoT security policy evaluation using an attribute-role hybrid model.

Overall, these works show that RBAC-ABAC hybrids offer a balanced, scalable, and adaptable solution for complex environments such as IoT and multi-domain systems.

Author(s)	Title	Dataset used	Findings	Output

Ameer, S., Benson, J., & Sandhu, R. (2022)	Hybrid approach towards secure access control in smart home appliances.	IoT device activity data (synthetic and real-world scenarios).	Proposed hybrid ABAC-RBAC model for smart homes, enhancing security and flexibility.	Improved security and flexibility in smart home IoT systems.
Long, S., & Yan, L. (2019)	Racac: An approach toward rbac and abac combining access control	Simulated user role and attribute datasets for access policy validation.	Combined RBAC and ABAC to create a unified RACAC model, improving granularity and policy expression.	Enhanced access control policy flexibility and usability.
Jyosthna,P. M. etal. (2024)	Enhancing Security and Flexibility with Combined RBAC and ABAC Access Control Models	Custom access policy datasets for security testing.	Explored combining RBAC and ABAC to address challenges in security and flexibility.	Demonstrated feasibility of combined models with improved security policies.
Attia, H. B., et al. (2018)	A new hybrid access control model for security policies in multimodal applications environments	Simulated multimodal application access logs.	Introduced a hybrid model combining ABAC and RBAC for multimodal environments.	Enhanced policy consistency and adaptability in multimodal application environments.
Varadharajan , V.,et al. (2015)	Policy based role centric attribute based access control model policy RC-ABAC	Hypothetical datasets modeling role and attribute combinations.	Proposed RC-ABAC, a policy-based hybrid model emphasizing role-centric attributes.	Improved access control granularity with role and attribute integration.
Attia, H. B., et al. (2018)	FRABAC: A new hybrid access control model for the heterogeneous multi-domain systems	Simulated multi-domain access scenarios.	Developed FRABAC for heterogeneous environments, addressing multi-domain challenges.	Increased interoperability and security in multi-domain access control.
Pal,S. & Jadidi,Z. (2021)	Protocol-based and hybrid access control for the IoT: Approaches and research opportunities	IoT protocol datasets and case studies.	Examined IoT-specific hybrid access control models and identified research gaps.	Highlighted potential for hybrid models in IoT with identified gaps and solutions.
Hasiba, B. A., et al. (2017)	A new hybrid access control model for multi-domain systems	Multi-domain simulated access datasets.	Proposed a hybrid access control model for multi-domain systems to improve scalability and policy management.	Achieved better scalability and policy management in multi-domain systems.
Fatima, A., et al. (2016)	Towards Attribute-Centric Access Control: an ABAC versus RBAC argument	Comparative analysis using custom attribute-role datasets.	Provided a detailed comparison of ABAC and RBAC, identifying scenarios where hybrid models are more effective.	Showed strengths and weaknesses of ABAC vs. RBAC and benefits of hybrid approaches.
Kaiwen, S., & Lihua, Y. (2014)	Attribute-role-based hybrid access control in the internet of things	IoT attribute-role datasets for security policy evaluation.	Proposed a hybrid model for IoT environments combining roles and attributes.	Enhanced IoT attribute-role datasets for security policy evaluation.

Stepien, B., et al. (2012)	CatBAC:A generic framework for designing and validating hybrid access control models	Generic framework tested on hypothetical access datasets.	Developed CatBAC, a generic framework for designing hybrid models combining RBAC and ABAC.	Framework validated with improved flexibility and security policy expression.
Penelova, M. (2021)	Hybrid Role and Attribute Based Access Control Applied in Information Systems	Simulated enterprise information system datasets.	Examined hybrid access control models applied to enterprise information systems for better security.	Demonstrated enhanced flexibility and security in information systems.
Hassan, M. A. M. (2020)	A New Model of Attribute Based Access Control (ABAC) for RDBMS Enterprise Applications	Relational database user activity datasets.	Developed an ABAC model for relational database systems, enhancing security and scalability.	Improved security in RDBMS applications with ABAC implementation.
Zhang, R., et al. (2021)	ABSAC: Attribute-Based Access Control Model Supporting Anonymous Access for Smart Cities	Introduced ABSAC for smart cities, focusing on anonymous access while preserving security.	Enabled secure and anonymous access control in smart city environments.	Smart city user access scenarios and anonymized datasets.
Nirmalrani, V., & Sakthivel, P. (2015)	A hybrid access control model with multilevel authentication and delegation to protect distributed resources	Distributed resource access and authentication logs.	Proposed multilevel authentication and delegation in hybrid access control models.	Improved resource protection through multilevel authentication and delegation.

III. EXSITING SYSTEM

3.1 Role-Based Access Control (RBAC)

Role-Based Access Control (RBAC) [3,8,11] is a highly accepted access control mechanism which reduces complexity in the management of access by allocating permissions to roles rather than users. In RBAC:

Roles: Represent the job functions or responsibilities of an organization, such as manager, employee, administrator, etc.

Users: Those who are assigned to the roles.

Permissions: Access rights to execute actions on resources, like read, write, delete, etc.

Role Assignment: Users inherit the permissions assigned to their roles.

RBAC simplifies administrative overhead, especially in large organizations where users with similar responsibilities can share permissions. This reduces the complexity of access management and ensures consistency.

However, RBAC has limitations when dynamic access requirements arise. For instance, RBAC lacks the flexibility to accommodate context-based conditions such as location, time, or device used for access. These challenges can be addressed through integration with Attribute-Based Access Control (ABAC).

Flow of RBAC:

a) Define Roles: Define roles according to organizational needs.

b) Map Permissions to Roles: Map permissions to specific roles.

c) Assign Users to Roles: Assign users to roles, depending on their responsibilities.

d) Force Access: Users can perform actions because of their role-associated permissions.

Limitations

1) Lack of Context Awareness - RBAC cannot handle dynamic conditions such as location, time, or device type.

2) Rigid Permission Structure - changes in access requirements require manual role reconfiguration.

3) Role Explosion Problem - large organizations with complex needs may require excessive role definitions, making management difficult.

4) Limited Fine-Grained Control - permissions are coarse-grained and tied only to roles, reducing adaptability.

3.2 Attribute-Based Access Control (ABAC)

Attribute-Based Access Control (ABAC) [6,8,13] improves access control since it considers a set of attributes to make decisions. These attributes include the following:

User Attributes: details about the user, name, job title, department, or security clearance.

Resource Attributes: resource information, such as file type, sensitivity level, or ownership.

Environmental Attributes: Information related to the context or time of access, location, or device used.

ABAC access decisions are based on policies that evaluate these attributes dynamically. This method provides fine-grained control and is highly adaptable to changing environments.

Flow of ABAC:

a) Define Attributes: Identify relevant user, resource, and environmental attributes.

b) Create Policies: Write policies that specify access conditions based on attribute values.

c) Evaluate Attributes: At runtime, evaluate the attributes of the user and resource.

d) Enforce Access: Allow or deny access based on policy evaluation.

Fine-grained control from ABAC is ideal for dynamic access scenarios, but managing and maintaining attributes and policies is extremely hard at scale.

Limitations

1) High Administrative Overhead - defining and managing numerous attributes and policies is complex.

2) Performance Overhead - evaluating attributes at runtime may slow down access decisions in high-volume systems.

3) Policy Conflicts - overlapping or contradictory policies can occur, requiring careful resolution.

4) Scalability Issues - managing attributes across multiple domains can be challenging at the enterprise level.

IV. PROPOSED SYSTEM

4.1 Hybrid RBAC-ABAC Model

The hybrid RBAC-ABAC [18,19,20] model combines RBAC for static, role-based permissions and ABAC for dynamic, attribute-based access control. The hybrid approach combines the simplicity of RBAC with the flexibility of ABAC to achieve efficient and fine-grained access control.

Architecture:

Role-Based Access Control: The users are assigned roles that define static permissions.

Attribute-Based Access Control: Policies evaluate additional attributes about the user, the resource, and the environment during runtime.

Policy Enforcement: Access is granted only when the role-based permissions and the attribute-based conditions are both met.

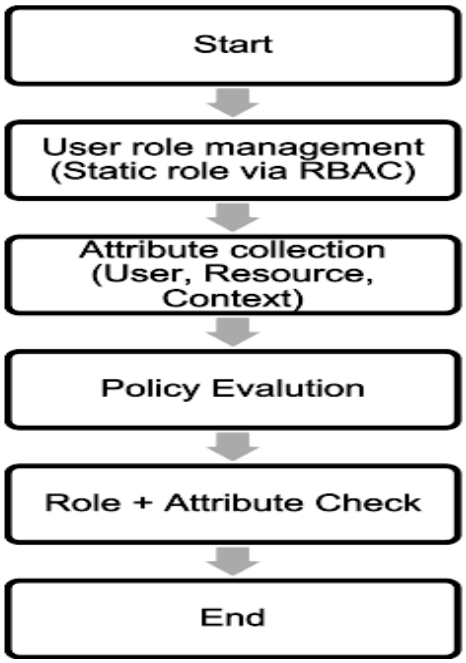

Figure-1 Hybrid RBAC-ABAC Model

Core Definitions

Definition-1

Users (UUU): The set of entities (e.g., people, processes) requesting access.U={u1, u2,…,un}U = \{u_1, u_2, \dots, u_n\}U={u1,u2,…,un}

Roles (RRR): Job functions or responsibilities within the system, each associated with permissions. R={r1,r2,…,rm}R = \{r_1, r_2, \dots, r_m\}R={r1,r2,…,rm}

Permissions (PPP): Actions users or roles can perform on resources.P={p1,p2,…,pk}P = {p_1, p_2, dots, p_k}P={p1,p2,…,pk}

Definition-2 Role Hierarchy (RHRHRH): A partial order on roles, where r1≥r2r_1 geq r_2r1≥r2 implies r1r_1r1 inherits r2r_2r2's permissions. RH⊆R×RRH subseteq R times RRH⊆R×R . **Attributes**

(A)-Properties describing users, resources, or the environment. Examples: user.department,resource.type , timeofaccess .

Definition-3 Policy Functions(PF) : Rules that determine access based on attributes and context.

$$PF:A\times Context\rightarrow\{True,False\}$$

Definition-4 Function Definitions

Assigned Permissions (assigned_permissionsassigned_permissionsassigned_permissions): Maps a role to its set of permissions.

assigned_permissions:R→2 Passigned_permissions: R to 2^Passigned_permissions:R→2P

User Roles (subject_rolessubject_rolessubject_roles): Maps a user to their assigned roles.

subject_roles:U→2Rsubject_roles: U \to 2^Rsubject_roles:U→2R

Definition-5

$$can\_access(u,p)=\begin{cases}True, & \text{if } \exists r\in subject\_roles(u), p\in assigned\_permissions(r),\\ True, & \text{if } PF(attributes(u),context(p))=True\\ False, & otherwise\end{cases}$$

Definition-6 Role Hierarchy: Administrator ≥ Manager ≥ Employee

Policy Function : PF(user.id,resource.owner_id)=(user.id==resource.owner_id)

Statement: A user u can access a permission ppp if:

1. U is active.
2. There exists an active role r∈subject_roles(u)r in subject_roles(u)r∈subject_roles(u) such that p∈assigned_permissions(r)p in assigned_permissions(r)p∈assigned_permissions(r), or a policy function evaluates to True.

Step 1: Assume u requests access to p.
If u is inactive, can_access(u,p)=False.

Step 2: Evaluate roles:
If ∃r∈subject_roles(u) and p∈assigned_permissions(r), then can_access(u,p)=True.

Step 3: Evaluate policy functions:
If PF(attributes(u),context(p))=True, then can_access(u,p)=True.

Step 4: can_access(u,p)=True can_access(u, p) = True can_access(u,p)=True if either condition holds.

Advantages

1) Combines strengths: RBAC's simplicity and ABAC's flexibility.
2) Fine and coarse-grained control: Stable role permissions with dynamic, context-based rules.
3) Better security: Requires a match of both role and attribute.
4) Prevents role explosion: Fewer roles are needed, as attributes handle variations.
5) Supports dynamic access: Adapts to real-time conditions such as location, time, and device.
6) Scalable and consistent: Functions well in large, complex organizations.
7) Improved compliance: Clear roles with policies that can be audited and are aware of context.

V. METHODOLOGY

This research presents a hybrid framework that merges Role-Based Access Control(RBAC) with Attribute Based Access Control (ABAC) to enhance the efficiency of authorization processes in access control systems.

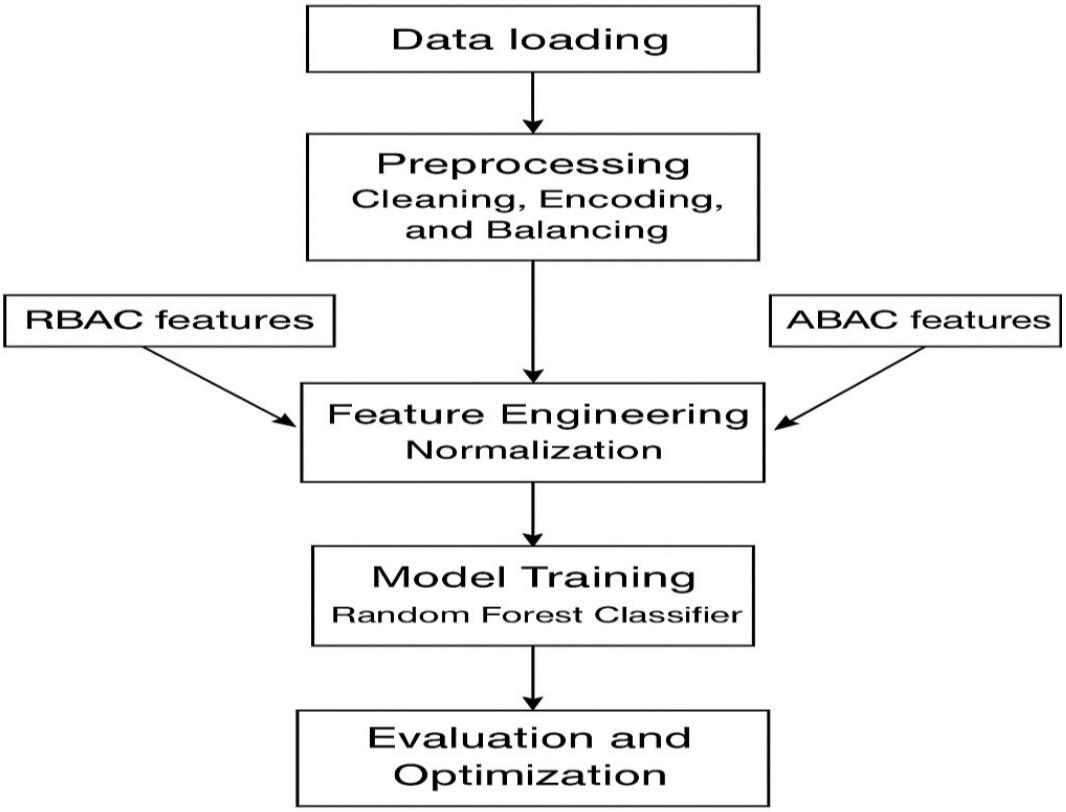

Figure-2 Flowchart representing the methodology

5.1 Data Acquisition and Preprocessing

Dataset Selection: The UCI Adult dataset was chosen due to its rich characteristics with varied demographic and occupational attributes, which allows simulation of real-world access control situations.

Data Cleaning: Missing values were identified and removed to provide the dataset with integrity.

Encoding: Categorical variables such as occupation, marital status, and native country were encoded by the label encoding techniques, which convert them into a form to be used by machine learning algorithms.

5.2 Feature Engineering

Normalization: The normalization of the feature scales was carried out to eliminate pre-existing biases in scale for numerical features like age, hours worked per week.

Synthetic Data Balancing: The dataset exhibited class imbalance, with certain user categories underrepresented. To address this, SMOTE (Synthetic Minority Oversampling Technique) was applied to generate synthetic samples and ensure balanced class distribution.

5.3 Hybrid Model Design

Role-Based Features: Role assignments were derived from categorical attributes such as occupation and work class.

Attribute-Based Features: Attribute-based controls incorporated granular properties, including age, education, and capital gain, ensuring fine-grained decision-making.

Hybrid Integration: Features derived from both RBAC and ABAC were integrated into a unified feature vector for machine learning-based classification.

5.4 Machine Learning Framework

Algorithm Selection: A Random Forest Classifier [6,7,9,10,11,12,15] was used due to its good handling of non-linear relations and high-dimensional inputs.

Model Training: This classifier was trained on the processed datasets with use of an 80-20 train-test split.

Optimization: Hyperparameter tuning was performed to optimize the number of estimates, maximum depth, and other model parameters.

5.5 Evaluation Metrics

Performance Metrics: Evaluation metrics included the confusion matrix, classification report, and ROC-AUC score.

Decision Thresholding**:** Post-evaluation, thresholds were fine-tuned to minimize false positives and negatives in access control decisions.

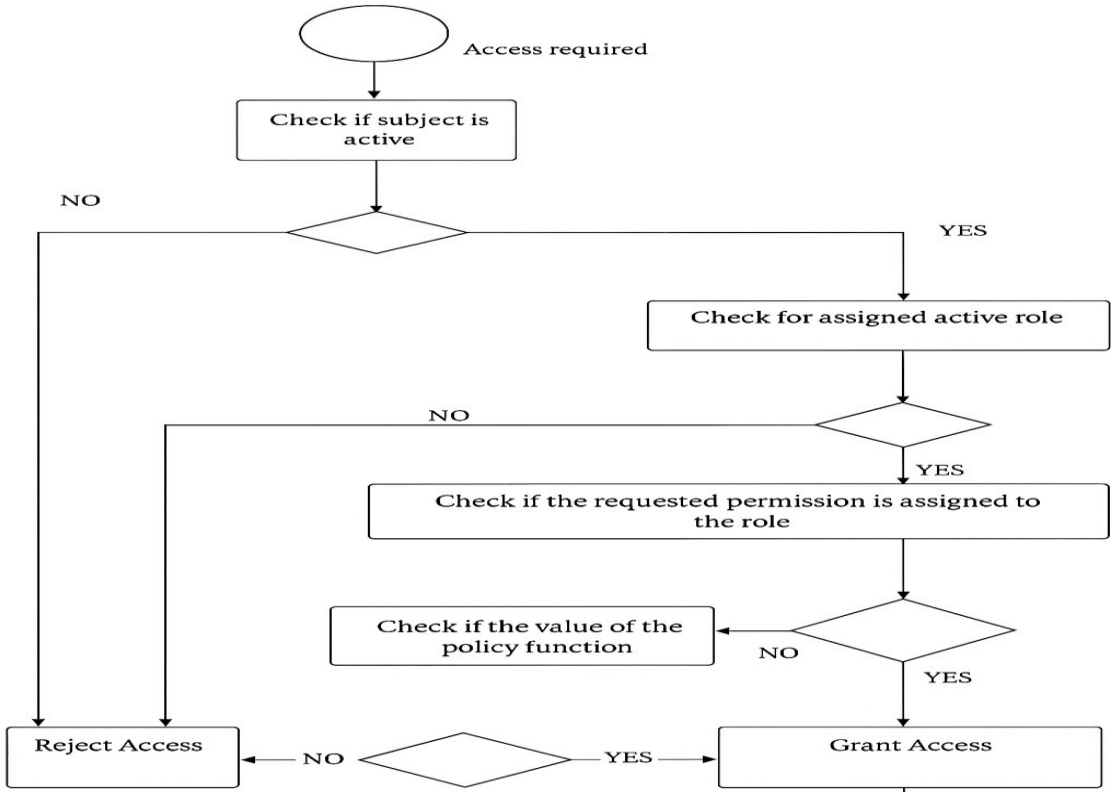

Figure-3 Process of access control in HRABAC

Certainly, let's break down the flowchart step by step:

1. Access Request:

- The process begins with an access request. This could be from a user attempting to access a resource or perform an action.

2. Check if Subject is Active:

- The system first checks if the user (or subject) making the request is active. If the user is inactive or their account is disabled, access is denied.

3. Check for Assigned Active Role:

- If the subject is active, the system checks if they have been assigned an active role. Roles are typically used to group users with similar access requirements. If no active role is assigned, access is denied.

4. Check if Requested Permission is Assigned to the Role:

- Assuming an active role is assigned, the system checks if the specific permission required for the requested access is granted to that role. If the permission is not granted to the role, access is denied.

5. Check if the Value of Policy Function is True:

- If the role has the required permission, the system evaluates a "policy function." This function can implement additional conditions or constraints based on specific policies or rules. If the policy function evaluates to false, access is denied.

6. Grant Access:

- Finally, if all the previous checks pass, the system grants the requested access to the user.

VI. RESULTS

	Precision	**Recall**	**F1-score**	**Support**
0	0.9	0.89	0.90	7417
1	0.67	0.69	0.68	2352
accuracy			0.84	9769
micro avg	0.79	0.79	0.79	9769
weighted avg	0.85	0.84	0.84	9769

Table-1 Classification report

MODEL DESCRIPTION	**ROC-AUC SCORE**
HYBRID RBAC-ABAC MODEL	0.9009
ABAC MODEL	0.7323

Table-2 Performance of the models was evaluated using the ROC-AUC metric

The results demonstrate that the HRABAC model significantly outperforms the attribute-based control model, highlighting the advantages of integrating role-based features into access control systems.

Metric	**RBAC**	**ABAC**	**HRABAC**
Configurability	High	Low	High
Granularity	Low	High	High
Scalability	Moderate	Low	High

Table-3 Comparison Metrics

6.1 Visualization

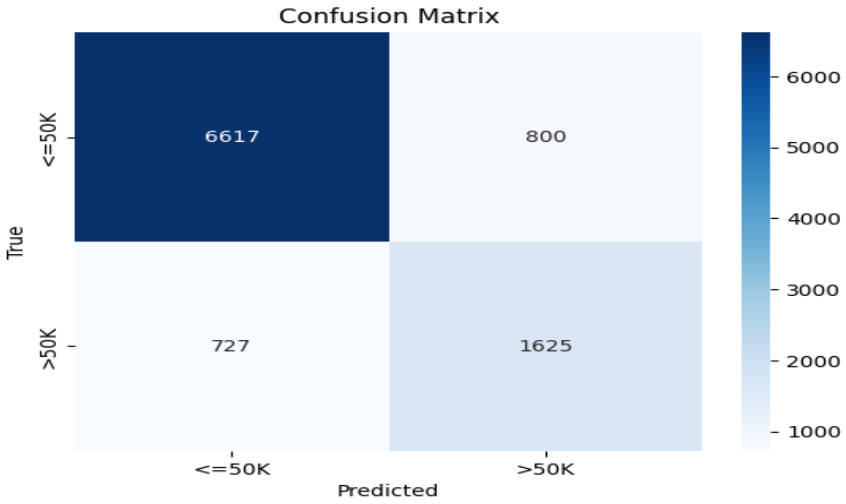

Figure-3 Confusion matrix

Figure-3 visualizes the classification performance, highlighting low false-positive and false-negative rates for the hybrid model. This aligns with its high ROC-AUC score and validates the efficacy of the hybrid framework

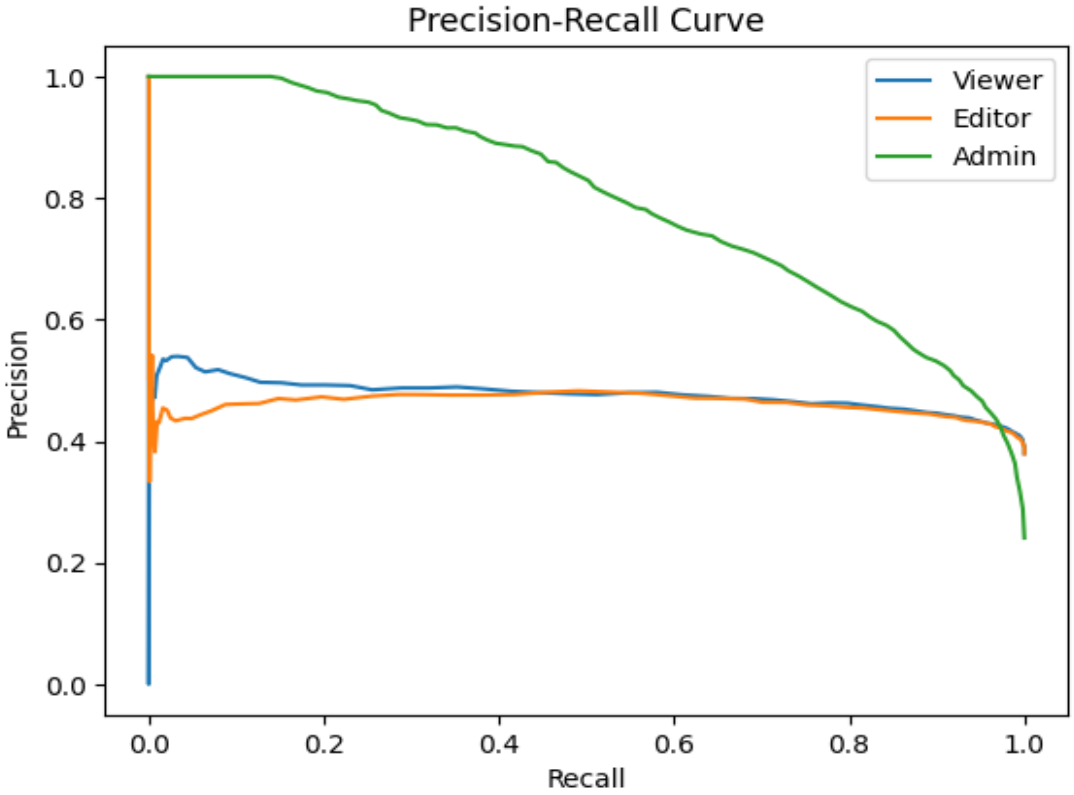

Figure-4 ROC Curve Analysis

Hybrid RBAC-ABAC Model: The ROC curve for this model demonstrated a high AUC score of 0.9009, indicating excellent discrimination between classes. The curve closely approaches the top-left corner, suggesting strong sensitivity and specificity.
ABAC Model: The ROC curve for the ABAC model showed an AUC score of 0.7323. While it performed better than random classification (AUC = 0.5), the curve is farther from the top-left corner compared to the hybrid model, reflecting moderate predictive power.

VII. CONCLUSION

This work generalized on the previous works done in a compartmentalized manner with a hybrid Role Based and Attribute Based Access Control (RBAC-ABAC) model. Results showed that compared to attribute-based access control used in isolation, its hybrid model performed better, with ROC-AUC score 0.9009 vs. 0.7323. Role-based and attribute-based characteristics assisted in constructing better decisions leading to a solid and versatile approach for access control systems.

VIII. FUTURE SCOPE

Future work can focus on the following aspects to further enhance the framework:

1. **Dynamic Access Control**: Incorporate real-time monitoring and dynamic attribute updates to improve responsiveness to changing access requirements.
2. **Scalability**: Test the framework on larger and more complex datasets to evaluate its scalability and performance.
3. **Integration with Blockchain**: Explore the use of blockchain technology for secure and tamper-proof storage of access control policies.
4. **Explainability**: Develop methods to provide interpretable decision-making insights, allowing administrators to understand and trust the model's predictions.
5. **Performance Optimization**: Experiment with advanced machine learning algorithms or deep learning models to further improve accuracy and reduce computation time.

REFERENCES

[1]Ameer, S., Benson, J., & Sandhu, R. (2022). Hybrid approaches (ABAC and RBAC) toward secure access control in smart home IoT. *IEEE transactions on dependable and secure computing*, *20*(5), 4032-4051.

[2] Long, S., & Yan, L. (2019, December). Racac: An approach toward rbac and abac combining access control. In *2019 IEEE 5th International Conference on Computer and Communications (ICCC)* (pp. 1609-1616). IEEE.

[3] Jyosthna, P. M., Mandapati, A. V., Teja, M. S., Ray, S. K., & Kumar, B. Y. S. (2024, April). Enhancing Security and Flexibility with Combined RBAC and ABAC Access Control Models. In *2024 10th International Conference on Communication and Signal Processing (ICCSP)* (pp. 576-581). IEEE.

[4]Attia, H. B., Kahloul, L., & Benharzallah, S. (2018). A new hybrid access control model for security policies in multimodal applications environments. *J. Univ. Comput. Sci*, *24*, 392-416.

[5] Varadharajan, V., Amid, A., & Rai, S. (2015, December). Policy based role centric attribute based access control model policy RC-ABAC. In *2015 International Conference on Computing and Network Communications (CoCoNet)* (pp. 427-432). IEEE.

[6] Attia, H. B., Kahloul, L., & Benharzallah, S. (2018). FRABAC: A new hybrid access control model for the heterogeneous multi-domain systems. *International Journal of Management and Decision Making*, *17*(3), 245-278.

[7] Pal, S., & Jadidi, Z. (2021). Protocol-based and hybrid access control for the iot: Approaches and research opportunities. *Sensors*, *21*(20), 6832.

[8] Hasiba, B. A., Kahloul, L., & Benharzallah, S. (2017, April). A new hybrid access control model for multi-domain systems. In *2017 4th International Conference on Control, Decision and Information Technologies (CoDIT)* (pp. 0766-0771). IEEE.

[9] Fatima, A., Ghazi, Y., Shibli, M. A., & Abassi, A. G. (2016). Towards Attribute-Centric Access Control: an ABAC versus RBAC argument. *Security and Communication Networks*, *9*(16), 3152-3166.

[10] Kaiwen, S., & Lihua, Y. (2014). Attribute-role-based hybrid access control in the internet of things. In *Web Technologies and Applications: APWeb 2014 Workshops, SNA, NIS, and IoTS, Changsha, China, September 5, 2014. Proceedings 16* (pp. 333-343). Springer International Publishing.

[11] Stepien, B., Khambhammettu, H., Adi, K., & Logrippo, L. (2012, June). CatBAC: A generic framework for designing and validating hybrid access control models. In *2012 IEEE International Conference on Communications (ICC)* (pp. 6721-6726). IEEE.

[12] Penelova, M. (2021). Hybrid Role and Attribute Based Access Control Applied in Information Systems. *Cybernetics and Information Technologies*, *21*(3), 85-96.

[13] Hassan, M. A. M. (2020). *A New Model of Attribute Based Access Control (ABAC) for RDBMS Enterprise Applications* (Master's thesis, Princess Sumaya University for Technology (Jordan)).

[14] Zhang, R., Liu, G., Li, S., Wei, Y., & Wang, Q. (2021). ABSAC: Attribute-Based Access Control Model Supporting Anonymous Access for Smart Cities. *Security and Communication Networks*, *2021*(1), 5531369.

[15] Nirmalrani, V., & Sakthivel, P. (2015). A hybrid access control model with multilevel authentication and delegation to protect the distributed resources. *Journal of Pure and Applied Microbiology (JPAM)*, *9*(2015), 595-609.

[16] Cheng, X., Dai, F., Hu, M., & Gui, Q. (2019). An improved privacy-preserving and security hybrid access control mechanism. In *Wireless Sensor Networks: 12th China Conference, CWSN 2018, Kunming, China, September 21–23, 2018, Revised Selected Papers 12* (pp. 169-180). Springer Singapore.

[17] Renuse, S., Mahalle, P. N., Shinde, G. R., & Sable, N. P. (2023, November). A Comparative Study of Access Control Models for Ubiquitous Computing Systems. In *Proceedings of the 5th International Conference on Information Management & Machine Intelligence* (pp. 1-6).

[18] Routh, A. K., & Ranjan, P. (2024, March). A Comprehensive Review on Granularity Perspective of the Access Control Models in Cloud Computing. In *2024 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI)* (Vol. 2, pp. 1-6). IEEE.

[19] Belhadaoui, H., Filali, R., & Malassé, O. (2023). A Role-Attribute Based Access Control Model for Dynamic Access Control in Hadoop Ecosystem. *IAENG International Journal of Computer Science*, *50*(1).

[20] Fernández, M., Mackie, I., & Thuraisingham, B. (2019, March). Specification and analysis of ABAC policies via the category-based metamodel. In *Proceedings of the Ninth ACM conference on data and application security and privacy* (pp. 173-184).

[21]Aktoudianakis, E. (2016). *Relationship based access control*. University of Surrey (United Kingdom).

[22] Benattia, H. (2019). *Formal Modelling and Verification of Security Policies in Cloud Computing* (Doctoral dissertation, Université Mohamed Khider-Biskra).

[23] Kashmar, N., Adda, M., & Ibrahim, H. (2022). Access Control Metamodels: Review, Critical Analysis, and Research Issues. *J. Ubiquitous Syst. Pervasive Networks*, *16*(2), 93-102.

[24]Almohammad Alsaleh, S. (2024). Permission-Based Dynamic Access Control Models for Enhanced Data Security: Integrating Contextual Awareness and Role Flexibility for Secure Healthcare Data Management.

[25] Danilescu, M. (2020). Comparative study of access control methods in enterprise information systems, based on RBAC, ABAC, and TBAC policies. *EIRP Proceedings*, *15*(1).