

Optimizing Access Control Policies Using Game Theory: A Nash Equilibrium Approach

Najem Aldeen Abu Hamdah Yazan Almoued

**DEGREE PROJECT
Computer Engineering
Bachelor level G2E, 15 hec**

Department of Engineering Science, University West, Sweden

Optimizing Access Control Policies Using Game Theory: A Nash Equilibrium Approach

Sammanfattning

Den här artikeln diskuterar en ny metod för optimal utformning av åtkomst policyer med hjälp av spelteorins principer för att balansera dynamiken mellan rollbaserad åtkomstkontroll (RBAC) och attributbaserad åtkomstkontroll (ABAC). Den modellerar policydesign processen som den strategiska interaktionen mellan angripare och försvarare med hjälp av Nash Equilibrium analys för att fastställa en optimal policyablandning som förbättrar säkerheten.

Projektet är baserat på empirisk statistik från IBM Cost of a Data Breach Report 2024 som ger detaljerad statistik över kostnaden. Det är också baserat på framgångsfrekvenser för vanliga cyberattacker som Phishing och Token Theft. Dessa framgångsfrekvenser härleds från två simuleringsmiljöer. Denna statistik används för att uppskatta realistiska utbetalningsmatriser, som sedan används för att beräkna Nash Equilibrium. Resultaten stöds sedan med hjälp av agentbaserade simuleringar baserade på Mesa-ramverket, där prestanda för policyinställningar kan observeras under dynamiska scenarier.

Resultat indikerar att hybridmetoden, ungefär 61% RBAC och 39% ABAC, presterar mycket bättre än implementeringen av ren ABAC eller ren RBAC. Det minskar antalet in-trängsfrekvensen med 16% jämfört med ren ABAC och 05% jämfört med ren RBAC, och utnyttjar de strukturerade kontrollerna av RBAC vid sidan av den kontextkänsliga anpassningsförmågan hos ABAC.

Forskningen visar att åtkomstkontroll kan optimeras strategiskt genom att använda förutseende tillvägagångssätt för motståndares beteende, vilket minskar svarstiden på potentiella hot.

Datum:	2025-08-30
Författare:	Najem Aldeen Abu Hamdah, Yazan Almoued
Examinator:	Andreas de Blanche
Handledare:	Abdulghafour Mohammad
Program:	Datateknik, högskoleingenjör – programmering och nätverksteknik, 180 hp
Huvudområde:	Datateknik
Utbildningsnivå:	Grundnivå
Kurskod:	EHD500, 15 hp
Nyckelord:	ABAC, Cybersecurity, Matplotlib, Mesa, Nash equilibrium, Nashpy, Phishing, RBAC, Simulation, Token Theft.
Utgivare:	Institutionen för ingenjörsvetenskap, Högskolan Väst 461 86 Trollhättan

Optimizing Access Control Policies Using Game Theory: A Nash Equilibrium Approach

Summary

This paper discusses one such novel approach to the optimal design of access policies using the principles of game theory to balance the dynamics between Role-Based Access Control (RBAC) and Attribute-Based Access Control (ABAC). It models the policy design process as the strategic interaction between attackers and defenders using Nash Equilibrium analysis to determine an optimal policy blend that improves security.

The approach is grounded in empirical statistics based on the IBM Cost of a Data Breach Report 2024 that provides detailed statistics of the average costs. And also based on success rates of common cyber attacks such as Phishing and Token Theft. The success rates were derived from two simulations environments. These statistics are used to estimate realistic payoff matrices, which are then utilized to compute the Nash Equilibrium. The findings are then supported using agent-based simulations based on the Mesa framework, where the performance of policy settings can be observed under dynamic scenarios.

Outcomes indicate that the hybrid approach 61% RBAC and 39% ABAC, performs much better than the implementation of pure ABAC or pure RBAC. It decreases breach rates by 16% compared to pure ABAC and by 05% compared to pure RBAC, leveraging the structured controls of RBAC alongside the context-sensitive adaptability of ABAC.

The research shows that access control can be strategically optimized using anticipatory approaches to adversary behavior, decreasing response time to potential threats.

Date:	Aug 30, 2025
Author:	Najem Aldeen Abu Hamdah, Yazan Almoued
Examiner:	Andreas de Blanche
Advisor:	Abdulghafour Mohammad
Programme:	Computer Engineering – programming and network technology 180 hp
Main field of study:	Computer Engineering
Education level:	First cycle
Course code:	EHD500, 15 HE credits
Keywords:	<i>ABAC, Cybersecurity, Matplotlib, Mesa, Nash equilibrium, Nashpy, Phishing, RBAC, Simulation, Token Theft.</i>
Publisher:	Department of Engineering Science, University West SE-461 86 Trollhättan, Sweden

Preface

We would like to express our sincerest gratitude and deepest appreciation to all people who have contributed significantly to the finalization of this thesis. This has been a unique and memorable experience, and we are truly appreciative of the guidance we have received along the way.

First and foremost, we would like to thank our thesis supervisor, Abdulghafoor Mohammad for his invaluable advice and continuous support throughout the research. We are lucky to be under his supervision. We also appreciate Rashid Ali for his suggestions and advice that kept us going. Our sincere thanks also go to our project examiner, Andreas de Blanche for his time to evaluate the thesis. We would also like to thank our program manager, Annabella Loconsole for her support, coordination, and help in facilitating the selection and supervision of the project.

Finally, we would like to extend our heartfelt gratitude to our families and friends. Their constant encouragement, patience and belief in us are the backbone of our progress. We are truly grateful for their invaluable presence and support throughout this journey.

The authors sincerely acknowledge the use of OpenAI ChatGPT, which contributed to improving the written content.

Table of contents

Sammanfattning	i
Summary.....	ii
Preface	iii
Table of contents	iv
Appendices.....	v
1 Introduction	1
1.1 Objective	1
1.2 Key Contribution	2
1.3 Problem Formulation	2
2 Related Work	2
2.1 Access Control Models	3
2.1.1 Role-Based Access Control (RBAC)	3
2.1.2 Attribute-Based Access Control (ABAC)	3
2.1.3 Hybrid Approaches.....	4
2.2 Game Theory in Cybersecurity	4
2.2.1 Foundations of Security Games.....	4
2.2.2 Game Theory in Access Control	5
2.3 Quantitative Approaches to Security Policy	5
2.3.1 Risk Assessment Frameworks	5
2.3.2 Quantitative Security Metrics	6
2.4 Research Gap.....	6
3 Background/Theory	6
3.1 Nash Equilibrium and Game Theory Fundamentals	7
3.1.1 Nash Equilibrium Concept.....	7
3.1.2 Mixed Strategies and Randomization	7
3.2 Access Control Models	8
3.2.1 Role-Based Access Control (RBAC)	8
3.2.2 Attribute-Based Access Control (ABAC)	9
3.2.3 Comparative Analysis and Hybrid Approaches.....	9
3.3 Theoretical Integration for Access Control Optimization.....	10
4 Methodology	11
4.1 Overview of Methodology.....	11
4.2 Data Collection and Analysis	11
4.2.1 Breach Cost Data Collection	12
4.2.2 Attack Success Rate Determination	12
4.3 Game Theory Matrix and Payoff Matrix Construction.....	18
4.4 Equilibrium Analysis.....	21
4.5 Agent-Based Simulation Methodology	21
4.5.1 Simulation Design	21
4.5.2 Agent Behavior Models.....	22
4.6 Simulation Environment.....	23
4.6.1 Overview of Matplotlib.....	23
4.6.2 Overview of Mesa	23
4.6.3 Overview of Nashpy.....	23
5 Results.....	24

5.1	Nash Equilibrium Analysis	24
5.1.1	Equilibrium Computation	24
5.2	Pure ABAC Simulation Results	26
5.2.1	Breach Rate Analysis	26
5.3	Pure RBAC Simulation Results	27
5.3.1	Breach Rate Analysis	27
5.4	Hybrid Policy Simulation Results	28
5.4.1	Policy Mix Evolution and Breach Rate Analysis	28
6	Analysis/Discussion	30
6.1	Interpretation of Nash Equilibrium Outcomes	30
6.2	Simulation Insights	31
6.3	Strategic Flexibility and Real-World Alignment	32
6.4	Limitations	32
6.5	Contribution to the Field	33
7	Conclusions	33
7.1	Future Work	34
	References	35

Appendices

A. Figures and Tables

1 Introduction

Access control mechanisms are intrinsic building blocks of cyber security infrastructure and are the first-line gatekeepers that give users access to organizational systems. Access control policy design has long been based more on traditional methods, however, with only limited proactive considerations. System administrators are often inclined to establish policies against common practices, past breaches, or compliance mandates, without analyzing the strategic actions of plausible adversaries rigorously in a systematic manner.

There are many access control paradigms in today's world of security. This paper will only focus on two paradigms, these are Role-Based Access Control known as RBAC and Attribute-Based Access Control known as ABAC [1][2]. RBAC provides permission based on pre-defined roles in an organization with ease and simplicity in management. ABAC, on the other hand, offers more fine-grained control by exercising access decisions based on a composite of attributes associated with the user, resource, action, and environment. Although RBAC presents more security with its strict paradigm, ABAC is more adaptable and flexible to accommodate diverse access situations.

The built-in conflict between these models forms a decision point critical to security administrators: balancing strict RBAC with loose ABAC. The situation is also complicated by the reality that attackers purposefully take advantage of the spaces between security and usability. When policies are too restrictive, users can avoid them to get their jobs done, opening vulnerabilities. When policies are overflexible, they can expose systems to multiple attack vectors [3].

This interactive game between defenders (system administrators) and attackers is a game of strategy in which each of their best actions relies on anticipating the action of others. Game theory, specifically Nash Equilibrium, offers a mathematical tool to study such strategic interaction and provide optimum strategy solutions to each of them.

1.1 Objective

This study aims to develop a quantitative model based on game theory to capture the game between defenders and attackers in the selection of access control policies. To achieve this goal, this study seeks to formalize the selection of access control policies (RBAC vs. ABAC) as a game with defenders and attackers as participants, then apply Nash equilibrium theory to find the best combination of RBAC and ABAC policies to balance security vulnerabilities and operational availability. After that, the theoretical results were supported through agent-based simulation of real-world situations and

provide practical suggestions for system administrators on how to optimize the combination of access control policies according to their specific threat environment. It is beyond the usual qualitative risk analysis. Instead, a more formal quantitative approach was adopted that more adequately considers the strategic nature of cyberspace security.

1.2 Key Contribution

The original contribution of this research consists of formulating a quantitative model that allows system administrators to make well-informed decisions regarding access policy choice based on game-theoretic insights. The model offers a mathematical model to balance security (for example RBAC) and flexibility (for example ABAC) in access control deployments, uses actual breach costs and attack success rates to build realistic payoff matrices, and illustrates by simulation, through a combination of RBAC and ABAC policies, how breach rates can decrease by a significant margin over single approach implementations such as ABAC. This paper fills a long-needed gap in current cybersecurity practice by presenting an anticipatory, strategy-oriented approach to access control policy definition, taking into consideration anticipated attacker activity.

1.3 Problem Formulation

Administrators have limited proactive mechanisms to anticipate attacker behavior and tune access control policies accordingly. Existing models often fall short in capturing the strategic, game-theoretic interactions between attackers and defenders, particularly due to the complexities introduced by dynamic behaviors and asymmetric information [4]. This project investigates the use of Nash Equilibrium to determine an optimal mix of RBAC and ABAC policies to minimize security breaches while preserving usability.

The central research question is:

"How can Nash Equilibrium guide the selection of access control policies to minimize breaches?"

2 Related Work

Game theory's intersection with cybersecurity has drawn much attention in recent years, with scholars investigating different methods to model and optimize security choices. This section presents an overview of literature in access control models,

game theory in security, and quantitative approaches to security policy optimization, setting the theoretical context for this work and determining the gap to be addressed by this framework.

2.1 Access Control Models

Over the last decades, access control methods have transformed from discretionary access control lists to complex attribute-based solutions. This is in proportion to the escalating organizational sophistication as well as growing demands for flexible security with reliable access control.

2.1.1 Role-Based Access Control (RBAC)

Role-Based Access Control was systematized by Sandhu et al. [2] in their influential paper of 1996, which formalized the base model sanctioning permissions based upon organizational roles, not specific users. Sandhu et al. defined $RBAC_0$ with users, roles, permissions, and sessions as central elements, as well as more complex models with role hierarchies and constraints. The hierarchical organization of permissions fit coherently with organizational hierarchies, making administration easier, minimizing access control administrative work.

Ferraiolo et al. [5] also extended the RBAC model in their NIST standard proposal with formal definitions and guidelines for implementation that placed RBAC in mainstream use in enterprise environments. The work of theirs revealed the administrative simplification, security by means of the principle of least privilege, and improved regulatory compliance benefits of RBAC.

Although widely used, RBAC is confronted with challenges such as role explosion (the proliferation of roles to meet individual access needs), role engineering complexity, and adapting to evolving organizational forms. These challenges underscored the necessity for more dynamic access models which would handle dynamic environments but preserve RBAC's administrative strengths.

2.1.2 Attribute-Based Access Control (ABAC)

Attribute-Based Access Control surfaced as a more expressive replacement for RBAC, as established by Hu et al. [1] in their NIST guide to ABAC definition and considerations. ABAC makes access requests based upon users', resources', actions', and environmental context attributes, enabling finer-grained, context-aware policies. It supports dynamic access decisions in response to changing situations without administrative intervention, remedying one of RBAC's central shortcomings.

Yuan and Tong [6] presented an attribute-based access control model for web services that showed how ABAC can be applied in practice in distributed scenarios. Their system utilized eXtensible Access Control Markup Language (XACML) to define attribute-based policies and how ABAC can handle extended access needs that are hard to define in RBAC.

Nonetheless, implementations of ABAC are confronted with challenges such as policy complexity, performance overhead, and nonstandard methods used in attribute management. Although ABAC is more flexible in comparison to RBAC, flexibility is bought by means of greater administrative complexity coupled with possible security vulnerabilities in case policies are poorly implemented and verified.

2.1.3 Hybrid Approaches

In real life, RBAC and ABAC have their own strengths and weaknesses. Some researchers try to combine the two for their advantages. Kuhn et al. [7] presented an RBAC system enhanced with ABAC, in which they tried to retain the simplicity of role-based permissions and constraints as much as possible, and only added attribute-based permissions and constraints when necessary. They claimed that their solution can enable organizations to use their existing RBAC infrastructure while gradually adding attribute-based features.

Jin et al. [8] presented Role-centric Attribute-Based Access Control (R-ABAC), an access control framework where RBAC, ABAC, and other access control models are integrated in a single framework.

2.2 Game Theory in Cybersecurity

Game theory is ideally placed to analyze rational decision making interactions in terms of mathematics, and hence fits particularly well with cybersecurity applications where defenders are in ongoing strategic interaction with attack actors. Game theory's use in security issues has increased over recent years, as researchers have used different game models and solution concepts in order to treat various cybersecurity issues.

2.2.1 Foundations of Security Games

Alpcan and Başar [9] were pioneers in applying game theory to network security, formulating attacker-defender interactions as a non-cooperative game. They laid down the theory underpinning security games, illustrating how game-theory analysis can be used to offer insight into optimal security measures as well as rational attacker equilibria. Roy et al. [10] gave an extensive review of game theory in network security. They

categorized security games along different dimensions such as availability of information, rationality of players, space of strategies, and concepts of equilibria to construct a taxonomy which enables researchers to choose relevant game models to tackle specific security issues.

2.2.2 Game Theory in Access Control

Although there is a large body of work covering applications of game theory to different facets of cybersecurity, there is still limited work in the area of access control optimization. Molloy et al. [11] applied game theory to investigate the security-usability trade-off in authentication processes by structuring interaction among users, administrators, and attackers as a three-player game. Their analysis demonstrated how authentication policies impact user behavior as well as security outcomes but did not generalize to access control policies.

Manshaei et al. [12] presented an extensive review of game theory applications to network security and privacy, such as access control, intrusion detection, and privacy preservation. They also observed that game theory is not widely used in access control optimization but have pointed to this as an exciting area for research in the future.

2.3 Quantitative Approaches to Security Policy

Quantitative approaches to security policy optimization have also largely relied upon risk analysis over strategic behavior, with different metrics and models being applied to assess how security is affected by alternative policies.

2.3.1 Risk Assessment Frameworks

Cheng et al. [13] developed a quantitative framework for evaluating access control policies based on the security impact of various configurations using measurement metrics from the Common Vulnerability Scoring System (CVSS). Their method combined system component vulnerability scores to derive an overall metric for security risk so that administrators might compare policies based on their risk profiles.

The NIST Risk Management Framework [14] also provides guidelines for security risk assessment and management, such as how to analyze access control policies. The framework also makes a differentiation in evaluating risk in terms of measuring both the probability and impact of security incidents, and offers an organized procedure to identify, analyze, and mitigate security risks.

2.3.2 Quantitative Security Metrics

Quantitative security measures offer a foundation for comparing and measuring various security policies to facilitate more objective security management decision making. Jansen [15] conducted an analysis of security measures in access control systems, which presented various methods for measuring security properties of access policies, such as policies based on permission distribution, separation of duty constraints, and measuring policy complexity.

Molloy et al. [16] also came up with quantitative measures to assess the effective security of access control configurations based on both technical security properties of the policy and human factors impacting its deployment. Their measures included considerations such as policy complexity, administrative overhead, as well as user compliance, which gave more insight into security efficacy compared to technical measures alone.

2.4 Research Gap

In light of the large body of literature for both security applications of game theory and access control models, still missing is extensive research which blends these to design access control policies to proactively respond to attacker strategy in an optimal manner. Work is either static in nature where policies are evaluated without taking attacker activity into account, or is applying game theory to security but is not dealing with security challenges of access control optimization.

This research fills this gap by introducing game theory techniques specially tailored for optimizing access control policies. By formalizing the strategic interaction between attackers and defenders in terms of access control, an analysis is provided of security/usability optimization with rational attacker adaptation to various policy settings. Therefore, the analysis allows security administrators to design policies in anticipation of, and in response to, attacker strategy, not just in response to security breaches after their occurrence.

3 Background/Theory

This chapter is based upon theoretical underpinning for the research, with an explanation of important concepts in game theory, access control models, and how they are integrated in security and usability optimization. It is important to understand all of these theoretical concepts in depth in order to appreciate the originality and importance of access control policy optimization.

3.1 Nash Equilibrium and Game Theory Fundamentals

Game theory offers a mathematical means to understand strategically interacting rational decision makers. In cybersecurity, game theory can be used to simulate the strategic interactions among defenders and attackers who make choices influencing the consequences for both.

3.1.1 Nash Equilibrium Concept

Nash Equilibrium, named after mathematician John Nash, is a basic concept in game theory that describes an equilibrium in which no player can improve their status by unilaterally changing their strategy while other players maintain their strategies unchanged [17]. The equilibrium concept applies to cybersecurity in the face of repeated adjustment of strategies by defenders and attackers in relation to other players.

In a two-player game with payoff matrices A and B , a pair of strategies (x^*, y^*) is a Nash Equilibrium if:

1. $x^* \cdot A \cdot y^* \geq x \cdot A \cdot y^*$ for every strategy x (defender cannot improve by changing strategy).
2. $x^* \cdot B \cdot y^* \geq x^* \cdot B \cdot y$ for every strategy y (attacker cannot improve by changing strategy).

Here, x and y are probability distributions over their respective strategy sets for defender and attacker, which are their mixed strategies. The dot product $x \cdot A \cdot y$ is used to compute the expected payoff when both players use these probability distributions to choose their strategies [18][19].

A Nash Equilibrium is a point at which no player can increase their expected payoff by unilaterally modifying their strategy, given the strategy chosen by their opponent. Equilibria can be pure (in which the player selects a single strategy with probability 1) or mixed (in which the player randomizes over a set of strategies). In cybersecurity contexts, in particular, mixed strategies are best because they are stronger by ruling out exploitable attack patterns. Such a strategic equilibrium can allow defenders to best design policies in advance based on rational attacker action, prior to even actual breaches.

3.1.2 Mixed Strategies and Randomization

Mixed strategies, in which players randomize over a set of pure strategies based upon a probability distribution, are central to security games. Randomization makes it impossible for an attacker to anticipate and exploit deterministic patterns of defense, introducing uncertainty which can deter or defeat an attack.

In access control, mixed strategies can be used through diversification of policy where various configurations or models of access control are used in various sections of the system or at different times. Diversification increases the difficulty in designing a general strategy that will work throughout the system, causing the attacker to spend more effort in reconnaissance and attack creation.

3.2 Access Control Models

Access control is a fundamental security mechanism that regulates who can access what resources in a system. Different access control models offer varying approaches to this regulation, each with distinct advantages and limitations.

3.2.1 Role-Based Access Control (RBAC)

RBAC is an access control strategy where permissions are granted to roles rather than to individual users. Roles are given to the user depending on the responsibilities they have within an organization [2]. The method simplifies the administration by organizing permissions into roles correlated to work functions or tasks within an organization.

The core components of RBAC include users, who are individuals that need to utilize system resources, roles, which represent organizational tasks or tasks within the organization, permissions, which are authorized operations on resources in general, and sessions, which are mappings between users and activated roles.

The implementations of RBAC usually follow the principle of least privilege, under which the users are allocated minimum roles to perform their work operations. The principle minimizes the damage from compromised accounts by limiting their range of access.

RBAC offers several security administration benefits including simpler management of users, reduced administrative overhead, and increased regulatory compliance. As permission handling is centralized in the case of RBAC with the use of roles, auditing and verification of entitlements in the entire organization becomes easier.

Still, RBAC has its limitations, particularly in situations that dynamically change, as access is continuously updated. The role definitions are fixed, and in such situations, "role explosion" occurs, as a large number of special-purpose roles are defined by administrators to serve customized access requirements. All such role proliferation negates the administrative benefits of RBAC and introduces security vulnerabilities through very permissive role assignments.

3.2.2 Attribute-Based Access Control (ABAC)

ABAC determines access requests in terms of users' attributes, resources, operations, and environmental circumstances [1]. ABAC is different from RBAC, as the access decision relies on policies that compare attribute values during execution time, thus providing more flexible and contextual control of access.

Key components of ABAC include attributes, which represent user characteristics, resources, actions, and environment, policies, which are regulations that identify conditions under which access is permitted, the Policy Decision Point (PDP), which is the policy evaluation component, the Policy Enforcement Point (PEP), a component that implements access decisions, the Policy Information Point (PIP), which supplies attribute values, and the Policy Administration Point (PAP), responsible for policy management.

ABAC policies have long relied on Boolean formulas to blend attributes and make access decisions. A good example to illustrate this is a policy that would allow a document's access to a person who is in the department of Finance, who is viewing a document that is Financial, and who is viewing during working hours.

ABAC's flexibility is able to offer very fine-grained access that can dynamically respond to changing situations without any administrative intervention. Rights of access, for instance, may be altered automatically in relation to time of day, place, security status of the gadget, or other contextual conditions. ABAC's flexibility makes it especially suitable for dynamic environments with changing access needs.

This flexibility has a price, one of greater complexity, and hence potential policy conflicts, configurations, and security exposure. ABAC policies are by no means easy to design, validate, and maintain, nor in companies with intricate access requirements.

3.2.3 Comparative Analysis and Hybrid Approaches

RBAC and ABAC are both systems of access control with different advantages and limitations. One has the convenience of administration and is compatible with the structures of organizations, while the other is better adapted to flexibility and to context sensitivity. The choice between them is an intrinsic trade-off between security and usability, with security via strongly defined roles for RBAC and usability via flexible, contextual policies for ABAC [20].

This has made necessary the creation of hybrid methods that incorporate parts of both models and are an attempt to use the complementary strengths of each. These hybrid methods are:

- **RBAC with Attributes:** Extending RBAC by constraining roles or permissions with attributes.

- **Attribute-Centric RBAC:** This approach allows for dynamic role assignments. Users are assigned to roles based on attributes.
- **Role-Centric ABAC:** Considering roles as attributes within ABAC policies or using roles to group together permissions that are based on attributes.
- **Dynamic RBAC:** Assigning roles dynamically based on contextual attributes constitutes Dynamic RBAC.

These hybrid methodologies seek to merge the straightforwardness of RBAC, in administrative terms, with the adaptability of ABAC, to arrive at a middle ground that satisfies a spectrum of access scenarios. Nevertheless, the best and most defensible design is contextual and emerges from an organization's specific (and perhaps unique) security and operational mix. Absent a principled framework, this design context renders the need for appearance optimization moot.

3.3 Theoretical Integration for Access Control Optimization

This work combines these theoretical principles by employing analysis of the Nash Equilibrium to represent the attacker-defender relationship. The integration that is produced presents an analytical theoretical framework for optimizing access control policy that takes into consideration the strategic interactions between attackers and defenders. The model is one where there is a defender who chooses an ABAC/RBAC policy mixture. The attacker can then choose between Token Theft and Phishing attack vectors. The model observes the costs of a breach and the probabilities of the different strategies succeeding, and then rates the pairs against each other.

The formulation of the access control optimization problem as a game has several important advantages. It is simple to understand and reason about. It is easy to express requests and access control rules in terms of the interactions between players in the game. And it enables the use of game-theoretic solution concepts for access control optimization. The model also accounts for the inherent trade-off between security and usability in access control design, the fact that attackers strategically adjust to various defense tactics, and the reality that security consequences are by nature probabilistic and uncertain.

To find the optimal mix of policies that minimizes expected breach costs while still allowing for necessary operational flexibility, the Nash Equilibrium for the game has been computed. The equilibrium is a secure solution in that both defenders and attackers cannot unilaterally change their strategies and improve their payoffs. This gives a solid basis for the design of access control policy.

4 Methodology

This section covers how to build and test the project using the Python language with the help of some important libraries, starting from setup to data collection, applying simulation and game theory to understand and enhance access control security systems.

4.1 Overview of Methodology

This section provides a clear roadmap before diving into the details, and it outlines the main steps followed in the project:

- **Gather empirical data:** Begin by collecting real-world data on breach costs from industry reports and credible sources.
- **Define simulation parameters:** Establish the simulated organization's structure (roles and user distribution) and set probabilities for phishing and token theft attacks based on empirical findings.
- **Run ABAC and RBAC simulations:** Build two simple environments, one using Attribute-Based Access Control (ABAC) and the other using Role-Based Access Control (RBAC), to observe how each policy responds to attacks.
- **Construct the payoff matrix:** Use the cost and success rate data to calculate the potential gains for both attackers and defenders, then normalize these values to units.
- **Apply game theory:** Utilize Nash equilibrium analysis to determine an optimal mix of RBAC and ABAC policies by treating the interaction between the defender and the attacker as a strategic game.
- **Validate with agent-based simulation:** Implement agent-based modeling to observe how the optimal policy mix performs over time and adjust based on observed breach rates.

4.2 Data Collection and Analysis

To make the game theory model as accurate as possible regarding actual security behavior, the values for payoffs were based upon empirical evidence for breach costs and attack success. The core sources of these values came from the IBM Cost of a Data Breach Report 2024, which offers detailed statistics for industry-specific costs of breaches as well as attack vectors [21].

4.2.1 Breach Cost Data Collection

Two key data points from IBM's report were used as the basis for the quantitative analysis:

- Phishing attacks cost organizations an average of USD 4.88 million per breach.
- Compromised credential attacks (similar to Token Theft) cost an average of USD 4.81 million per breach.

These numbers gave empirical evidence that formed the basis of the payoff matrix, enabling to frame the various attack vectors and response measures in measurable terms of associated losses [21].

4.2.2 Attack Success Rate Determination

This project simulates two types of attacks (Phishing and Token Theft) and two types of access control policies (ABAC and RBAC), which means that four success rates values should be determined:

- The success rate of Phishing Attack against ABAC.
- The success rate of Token Theft against ABAC.
- The success rate of Phishing Attack against RBAC.
- The success rate of Token Theft against RBAC.

Phishing and Token Theft attack success rates can widely vary based upon many factors such as specific access control system implementations, security controls in place, training of users, and attack sophistication. And because IBM's report does not directly present success rates of various attacks upon various access control policies, and because no other sources that provide these values were found, two new simulations of two simple environments were created to approximately determine the four values of success rates. Each simulation environment represents a small-scale organization comprising 109 employees. However, the input parameters used within these environments are not randomly, they are grounded in industry-wide empirical data drawn from comprehensive reports/articles published by organizations such as Business-Money, Medium, ScatteredSecrets, arXiv and KnowBe4 [23][24][25][26][27]. These sources provide generalized statistics on breach success rates, company distribution structure and policy effectiveness across diverse access control frameworks, forming the empirical basis for the simulation inputs.

4.2.2.1 Simulation Company Structure and Empirical Justification for Probability

The organizational structure used in the simulation is inspired by a real-world company, Clavister, a Swedish cybersecurity firm with 109 employees [22]. Although no publicly available report details Clavister's internal hierarchy, general workforce distributions in technology companies are well-documented in industry sources. These reports commonly indicate that around 54% of employees are classified as staff, 37% as engineers, and the remaining 9% as administrators [23][24]. Based on this ratio, the distribution for a 109-person company is calculated as follows:

- Staff: $109 * 54\% = 59$ user
- Engineer: $109 * 37\% = 40$ user
- Admin: $109 * 9\% = 10$ user

For the first attack vector, phishing, the success probabilities across different access control models were derived from a combination of sources, including research articles on arXiv and cybersecurity reports from KnowBe4. Specifically:

- One study reported that approximately 10.9% of individuals in technical roles, such as administrators and engineers, were tricked into entering their credentials, including usernames and passwords [25].
- Another report showed that 60.7% of non-technical users, particularly those in departments like Human Resources, clicked on phishing messages. These users are categorized as staff in our model [26].

Based on this data, the success probabilities for phishing attacks in the simulation environments are set accordingly:

```
success_prob = {'Admin': 0.11, 'Engineer': 0.11, 'Staff': 0.61}
```

For the second attack vector, token theft, the success probabilities across different access control models were also applied in both simulation environments. These values were based on findings from ScatteredSecrets [27]. According to the report, the success rate of credential compromise through session hijacking (token theft) ranges from 5% to 15%. In the simulation, staff members were assigned the higher end of this range, 15%, to represent a worst-case scenario. This assumption reflects the likelihood that non-technical users are more vulnerable to such attacks [28], especially when multi-factor authentication (MFA) is not enforced. On the other hand, administrators and engineers, considered technical users, are presumed to be more security-aware and more likely to use MFA, so their success probability was set to 5%.

As a result, the token theft success probabilities used in the simulation environments are defined as follows:

`success_prob = {'Admin': 0.05, 'Engineer': 0.05, 'Staff': 0.15}`

Table 1. Success rates for Phishing and Token Theft Attacks

Role	Phishing Probability	Token Theft Probability
Admin	11%	5%
Engineer	11%	5%
Staff	61%	15%

4.2.2.2 First Simulation: ABAC Environment

To empirically evaluate the security posture of an Attribute-Based Access Control (ABAC) system, a controlled simulation environment was constructed using Python. The environment replicates an organization of 109 users, distributed across three roles, Admin, Engineer, and Staff, based on real-world demographic proportions. Each simulated user is assigned specific attributes: role, department, and a clearance level, which are stored in a lightweight SQLite database. These attributes are then cross-checked against ABAC policy rules that determine access to three distinct resources: an admin page, an engineering page, and a general page.

The simulation executes in multiple phases, each representing a distinct cybersecurity scenario. Initially, two types of attacks are simulated separately: phishing and token theft. In both cases, attackers randomly target users with a fixed probability of success based on the user's role. Staff members, often considered non-technical, are assigned higher probabilities of compromise (60% for phishing, 15% for token theft), while Admins and Engineers are modeled as more resilient due to security awareness or access restrictions, with lower success rates (11% and 5%, respectively)[28][29][30].

When an attack succeeds, the compromised user's attributes are exploited in an attempt to access protected resources. For every compromised account, the simulation tests whether access is granted to each resource by applying the ABAC policy logic. This process mimics real-world breach scenarios in which attackers attempt lateral movement or privilege escalation using stolen credentials or hijacked sessions.

The model records whether unauthorized access is permitted or denied, allowing for quantitative measurement of ABAC's defense capabilities. Metrics such as total breaches, access attempts, and resource-specific intrusion rates are computed to derive an effectiveness score. This score reflects how well the ABAC system resists unauthorized access, not just in preventing account compromise, but in stopping those compromised identities from misusing elevated privileges.

Overall, this simulation reveals the resilience of ABAC when faced with role-targeted attacks and provides foundational data for constructing realistic payoff matrices used

in the game-theoretic analysis. It emphasizes that while ABAC offers flexibility and dynamic access evaluation, its effectiveness is heavily reliant on how well attribute conditions are enforced in the face of compromised user accounts.

The average success rates of the simulation after running it 10 times shows the following:

- Success Rate of Phishing Attack against ABAC: 42%.
- Success Rate of Token Theft Attack against ABAC: 12%.

These findings show that, using a simulated context, Phishing attacks were more successful at compromising users' accounts than Token Theft attacks. Nevertheless, attribute-based policies of an ABAC system were successful in preventing unauthorized usage, even if users or tokens were compromised.

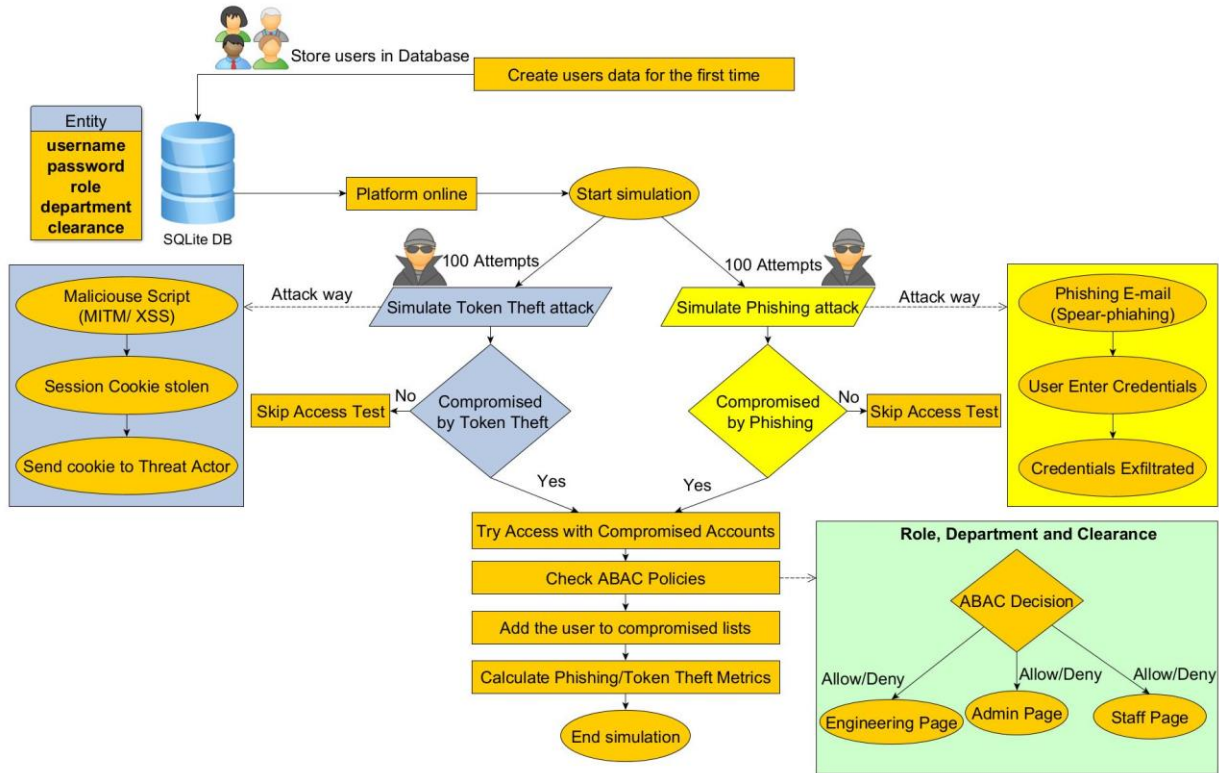


Figure 1. Flowchart of the ABAC Simulation Environment and Attack Vector Processing

Figure 1 illustrates the mechanism of the ABAC simulation environment using an ASM (Abstract State Machine) chart. The simulation begins by executing the first attack vector, phishing, followed by the second vector, token theft. Each attack is processed separately to allow clearer evaluation of their respective impacts on the access control environment.

4.2.2.3 Second Simulation: RBAC Environment

To assess the resilience of Role-Based Access Control (RBAC) against common cyber threats, a second simulation was conducted using a strict policy environment. The simulation replicates a company with 109 employees across three predefined roles, Admin, Engineer, and Staff, each assigned a department and fixed clearance level. Permissions are rigidly mapped according to roles, enforcing access boundaries for three resources: the admin page, engineering page, and general page.

The RBAC simulation follows a structured sequence. It begins with the initialization of a user database where each individual is assigned a role and marked as uncompromised. The system enforces RBAC permissions strictly and includes an additional inline detection mechanism that introduces a 40% chance of flagging suspicious behavior [29]. This feature simulates real-time security monitoring where accounts showing abnormal access patterns may be instantly disabled.

In the first phase, the simulation executes 100 phishing attempts. A user is randomly selected in each attempt, and the success of the attack is determined by a predefined probability based on the user's role, 61% for Staff, 11% for Engineers, and 11% for Admins [25][26]. Successful attempts result in the attacker obtaining the user's credentials, but the account is not yet flagged. The second phase mirrors the same process with 100 token theft attempts, applying lower probabilities, 15% for Staff, and 5% for Engineers and Admins [27].

Once the attacker gains access, each compromised account is used to attempt unauthorized entry into the protected resources. The RBAC system then evaluates whether the role assigned to the user permits such access. Every access attempt, whether permitted or denied, is simultaneously evaluated by the inline detection system. If detection is triggered, the account is disabled immediately, preventing any further exploitation.

Throughout the simulation, the model records the number of compromised accounts, total access attempts, and the number of successful unauthorized accesses. This data is used to compute key security metrics, including the breach success rate and RBAC effectiveness. The addition of inline detection introduces a dynamic element of security that reacts in real-time, enhancing RBAC's capacity to isolate threats.

This simulation provides empirical support for the static but stringent nature of RBAC. It highlights how a clearly defined permission structure, coupled with early detection capabilities, can significantly reduce unauthorized access, even when user credentials are compromised. By examining outcomes for both phishing and token theft vectors, the model helps quantify RBAC's defensive potential within a simplified yet realistic organizational structure.

The average success rates of the simulation after running it 10 times shows the following:

- Success Rate of Phishing Attack against RBAC: 16%.
- Success Rate of Token Theft against RBAC: 17%.

These findings indicate that, in simulated terms, Token Theft assaults were more successful at compromising users' accounts than Phishing assaults. But strict role-based permissions of the RBAC system successfully prevented unauthorized usage even if credentials or tokens were compromised. This points out that clearly established roles and permissions are critical for lessening the effects of such attacks.

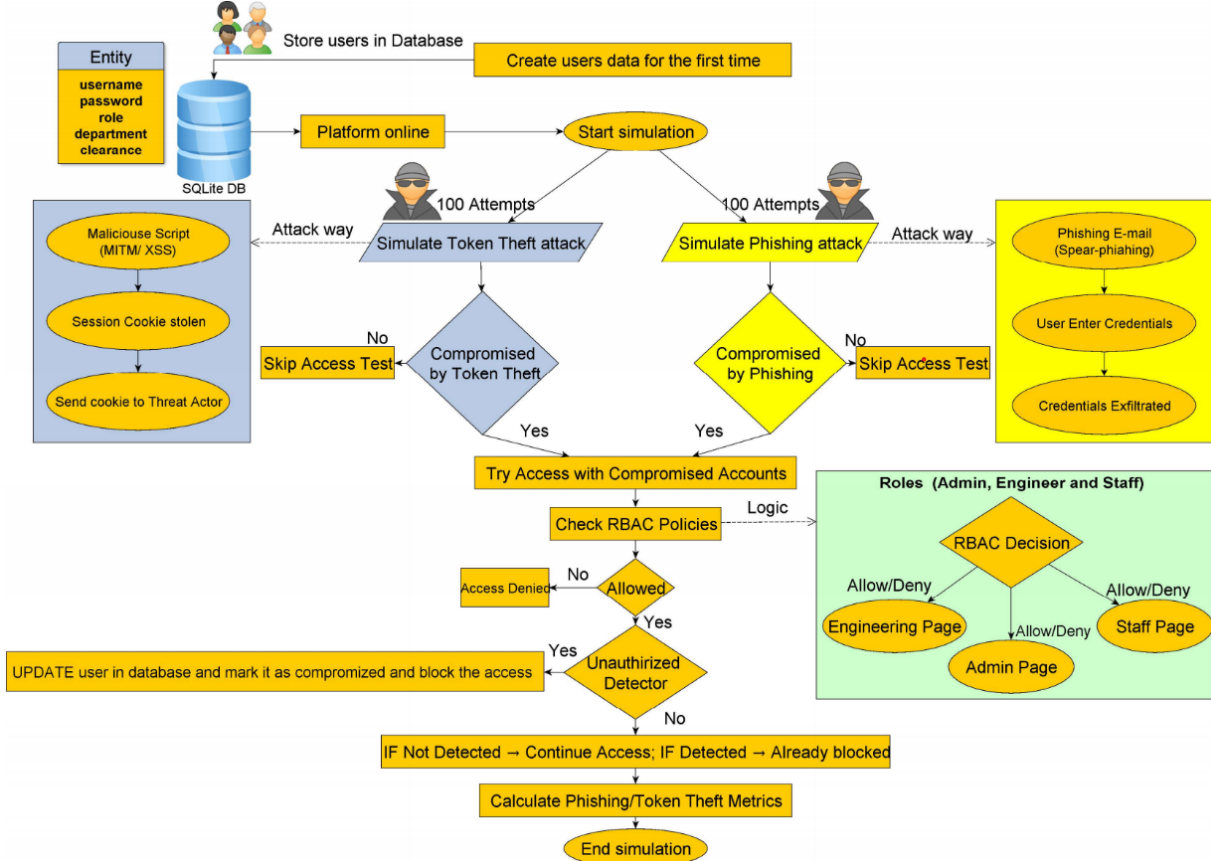


Figure 2. Flowchart of the RBAC Simulation Environment and Attack Vector Processing

Figure 2 illustrates the mechanism of the RBAC simulation environment using an Abstract State Machine (ASM) chart. The simulation begins with the execution of the first attack vector, phishing, followed by the second vector, token theft. Each attack is executed independently to allow separate assessment of their impact on the RBAC-based access control system.

4.2.2.4 Difference Between the Two Simulation Environments

In the ABAC simulation, access control is determined dynamically. Each access request is evaluated at the time it is made, using a set of user attributes including role,

department, and clearance level. The access policy for each resource defines which combination of these attributes is required. If the user's attributes match the policy, access is granted. Otherwise, access is denied. There is no additional detection system; decisions rely entirely on the policy logic.

In contrast, the RBAC simulation uses a fixed mapping of roles to resources. Each user is assigned a role, and each role has a predefined list of resources that it is allowed to access. Access is granted or denied based strictly on whether the requested resource is included in the role's permissions. Additionally, this simulation includes a detection mechanism. When a compromised user attempts to access a resource, there is a 40% chance that the system will detect the unauthorized behavior. If detection occurs, the user's account is marked as compromised and further access attempts are blocked.

The reason detection is included only in the RBAC simulation is that role-based systems, by design, are static and do not account for context. If an attacker gains access to valid credentials, the system cannot distinguish between a legitimate and malicious user unless external detection mechanisms are in place. Therefore, the simulation includes an inline detection system to reflect a more realistic enterprise setup where security tools are used to monitor activity and stop threats in real-time.

ABAC, on the other hand, already considers context through its attribute-based logic. Each request is assessed using real-time conditions such as department and clearance, so adding a separate detection layer would overlap with the policy's intended behavior. To maintain a fair comparison and highlight the natural strengths of each model, only the RBAC simulation uses detection.

4.3 Game Theory Matrix and Payoff Matrix Construction

One of the most important elements of the methodology involved creating a payoff matrix, in which its values were calculated through actual costs and success rates. The values of the payoff matrix are then normalized to units instead of millions.

As mentioned before and based on IBM's reports, Phishing attacks cost organizations an average of USD 4.88 million per breach and compromised credential attacks (similar to Token Theft) cost an average of USD 4.81 million per breach [21]:

- Average Cost of Phishing Attack = 4,880,000
- Average Cost of Token Theft = 4,810,000

And the success rates were determined from the simulations of the two environments that were created:

- Success Rate of Phishing Attack against RBAC: 16% = 0.16
- Success Rate of Token Theft against RBAC: 17% = 0.17
- Success Rate of Phishing Attack against ABAC: 42% = 0.42
- Success Rate of Token Theft Attack against ABAC: 12% = 0.12

The calculation of the payoff matrix:

- Phishing Attack against RBAC:
 - Attacker's payoff = Average Cost of Phishing Attack * Success Rate of Phishing Attack against RBAC = $4,880,000 * 0.16 = 780,800 \rightarrow 0.8$ units
 - Defender's Payoff = Average Cost of Phishing Attack - Attacker's Payoff = $4,880,000 - 780,800 = 4,099,200 \rightarrow 4.1$ units
- Token Theft against RBAC:
 - Attacker's Payoff = Average Cost of Token Theft * Success Rate of Token Theft against RBAC = $4,810,000 * 0.17 = 817,700 \rightarrow 0.8$ units
 - Defender's Payoff = Average Cost of Token Theft - Attacker's Payoff = $4,810,000 - 817,700 = 3,992,300 \rightarrow 4$ units
- Phishing Attack against ABAC:
 - Attacker's Payoff = Average Cost of Phishing Attack * Success Rate of Phishing Attack against ABAC = $4,880,000 * 0.42 = 2,049,600 \rightarrow 2.1$ units
 - Defender's Payoff = Average Cost of Phishing Attack - Attacker's Payoff = $4,880,000 - 2,049,600 = 2,830,400 \rightarrow 2.8$ units
- Token Theft against ABAC:
 - Attacker's Payoff = Average Cost of Token Theft * Success Rate of Token Theft against ABAC = $4,810,000 * 0.12 = 577,200 \rightarrow 0.6$ units
 - Defender's Payoff = Average Cost of Token Theft - Attacker's Payoff = $4,810,000 - 577,200 = 4,232,800 \rightarrow 4.2$ units

Due to the structure of the game, it must result in one winner and one loser, both players cannot win or lose simultaneously. To achieve this outcome, four out of the eight values were designated as negative. These values were not chosen at random but were selected based on the following criteria in figure 3:

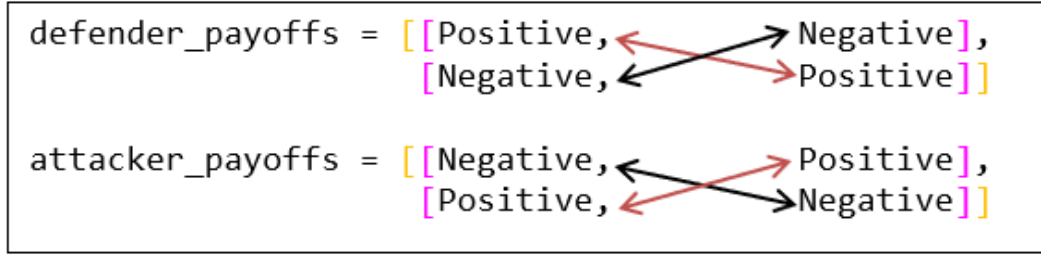


Figure 3. Sign assignment, strategy payoffs (original configuration)

Or vice versa as figure 4 shows:

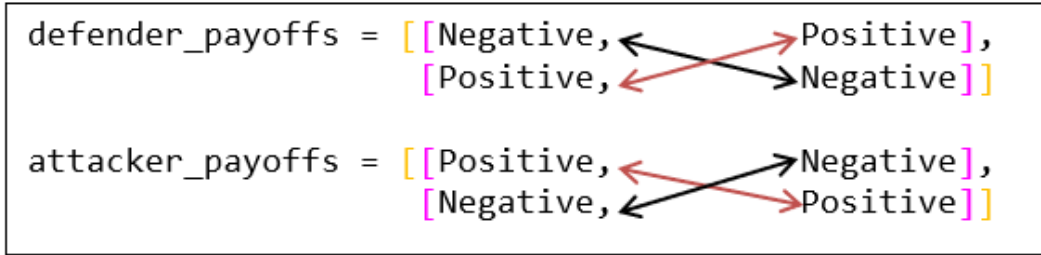


Figure 4. Sign assignment, strategy payoffs (inverted configuration)

Figure 5 shows the final Nash Equilibrium:

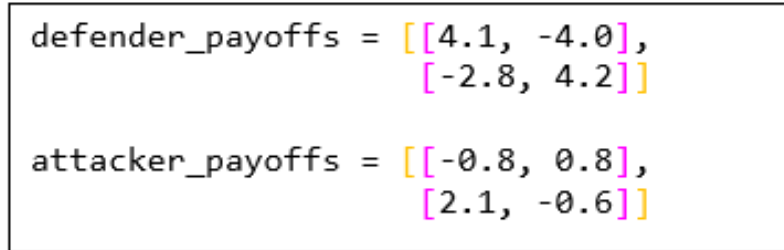


Figure 5. Nash equilibrium outcome matrix

Table 2 shows the payoff matrix that represents the strategic interaction between defenders and attackers:

Table 2. Strategic payoff matrix for defender and attacker

Defender / Attacker	Phishing	Token Theft
Strict RBAC	(+4.1, -0.8)	(-4, +0.8)
Loose ABAC	(-2.8, +2.1)	(+4.2, -0.6)

4.4 Equilibrium Analysis

With the payoff matrix defined, Nashpy library was used to calculate Nash Equilibrium of the game. Nash Equilibrium is a state where neither the attacker nor the defender can improve their outcome by unilateral change of their strategy, yielding an equilibrium solution to the game.

The Python implementation for equilibrium analysis is as follows:

```
import nashpy as nash

defender_payoffs = [[4.1, -4.0],
                   [-2.8, 4.2]]

attacker_payoffs = [-0.8, 0.8],
                   [2.1, -0.6]]

# Create the game
game = nash.Game(defender_payoffs, attacker_payoffs)

# Compute equilibria
equilibria = list(game.support_enumeration())
```

Figure 6. Simplified Nash Equilibrium code to calculate mixed strategies

This analysis does produce both defenders' optimal mixed strategy as well as attackers' optimal mixed strategy, which can then be confirmed by agent-based simulation. The Nash Equilibrium forms a theoretical underpinning to optimal policy mix consisting of rational play by both defenders as well as by attackers.

4.5 Agent-Based Simulation Methodology

In order to test the game theory model and understand how policy adaptation would evolve over time, an agent-based simulation in Mesa was implemented. The simulation involved defender agents (system administrators) and attacker agents that interacted in an environment with access control policies.

4.5.1 Simulation Design

Three sets of simulations were carried out to assess the entire strategy:

1. **Pure ABAC Simulation:** The baseline simulation simulated only an environment with ABAC policies in effect in order to quantify the rate of breaches in

an extensible but possibly exposed system. The pure ABAC simulation is used to obtain a baseline case to which performance optimization results can be compared.

2. **Pure RBAC Simulation:** Another baseline simulation that simulated only an environment with RBAC policies in effect in order to quantify the rate of breaches in a strict and less flexible system. The pure RBAC simulation is used to obtain a baseline case to which performance optimization results can be compared.
3. **Hybrid Policy Simulation:** In this simulation, the optimal policy mix from the Nash Equilibrium analysis is used where defenders adaptively shift the balance between RBAC and ABAC in response to observed intrusion rates. This simulation evaluates how effectively this game theory solution would work in practice.

Each simulation ran for 100 steps, with the following parameters:

- 109 employee agents (defenders).
- 50 attacker agents.
- Initial policy mix based on Nash Equilibrium results (for hybrid simulation).
- Breach rates recorded and averaged over a moving window of 10 steps.

The simulation code used success rates from the data analysis, with attackers deciding to use Phishing or Token Theft based on their predicted payoffs.

4.5.2 Agent Behavior Models

The simulation had two main agent types with different behavior models:

1. **Attacker Agents:** The agents attack using either Token Theft or Phishing based on their perceived policy mix. Their rates of success are based on the approximate values from the simulation environments that were created.
2. **Defender Agents:** The defender agents tune their policy mix of RBAC and ABAC in response to observed breach rates. Agent goals are to minimize breaches without over constraining operations. The defender agent employs a control system with a feedback component from observed breach rates.

This simulation technique makes it possible to see how security policies evolve dynamically over time and demonstrate the operational efficacy of the game theory solution to optimizing access.

4.6 Simulation Environment

The project simulation for this project has been created based on Python programming within Visual Studio Code, which is a versatile and open platform for code handling and operation. Source codes are available on GitHub, which can be cloned or forked, allowing researchers and developers to explore deeper into the implementation and further develop the simulation for future research and development purposes [30].

4.6.1 Overview of Matplotlib

Matplotlib is a popular Python library used for generating data visualizations, particularly statistical and scientific plots [31]. In this project, Matplotlib was used to transform numerical simulation outputs into graphical form, allowing general users to understand the behavior of the systems. Specifically, it plots the evolution of the mixed policy between Role-based and Attribute-based access controls and the moving average of breach rates over time.

4.6.2 Overview of Mesa

Mesa is known as agent-based modelling (ABM) and it's a Python-based framework that allows developers to implement or build their own simulations, which simulate complex systems consisting of self-governing agents [32]. Mesa has been used to simulate the cybersecurity model with defender and attacker agents. Every agent behaves independently and interacts with other agents. The model captures emergent dynamics, which in turn allows simulation of defenders' access control policies breach rate adjustments based on multi-directional attacker behavior and cascade effects. The framework organizes agent scheduling and state updates, as well as data collection for observability of access control policy changes over time. The study with Mesa improves realism in performance evaluation policy assessment by demonstrating policy enacted are meant for dynamic adaptive policy enactment instead of static ones illustrating long-term policy effectiveness, enduring threats.

4.6.3 Overview of Nashpy

Nashpy is a Python library for the computation of Nash Equilibria in 2 player strategic (normal form) games [34]. In this project, Nashpy has been applied to model the strategic interaction between a defender and an attacker via game theory. The defender's strategies play out in terms of access control configurations such as RBAC or ABAC, while the attacker has two attack vectors, such as Phishing and Token Theft attacks. Payoffs are used in matrix form to identify equilibria, which means strategy

pairs in which no player has an incentive to unilaterally deviate from. The equilibrium thus determined puts forth an initial strategy distribution for both defender and attacker in the simulation, which in turn grounds the agent-based model in theory-based stable strategic behavior. This use of game theory in simulation improves the depth and realism of the analysis.

5 Results

This chapter gives the complete outcome of Nash Equilibrium analysis as well as agent-based simulation results, establishing the efficiency of game theory-based approach to optimizing access control policy. It also offers a thorough analysis of both the theoretical equilibrium and its verification via simulation.

5.1 Nash Equilibrium Analysis

Game theory analysis, tapping into the payoff matrix established from the simulation environments, provided substantial understanding of the optimal trade-off between RBAC and ABAC policies. The analysis was done through employing the Nashpy library, which contains different algorithms for calculating Nash Equilibria for two-player games.

5.1.1 Equilibrium Computation

Using the payoff matrices, the Nash Equilibrium of the defender-attacker game was computed:

```
defender_payoffs = [[4.1, -4.0],  
                    [-2.8, 4.2]]  
  
attacker_payoffs = [[-0.8, 0.8],  
                    [2.1, -0.6]]
```

Figure 5. Nash equilibrium outcome matrix

This analysis yielded the following Nash Equilibrium:

- **Defender Strategy:** 63% RBAC, 37% ABAC.
- **Attacker Strategy:** 54% Phishing, 46% Token Theft.

This is an equilibrium of the optimal policy mixture for the defender and the optimal attacking mixture for the attackers. Here, neither the defender or the attacker can increase their payoff by unilaterally changing their strategy.

And by solving the original Payoff matrices mathematically, the following results were found:

Both the defender and attacker have two strategies. To define the players' strategies with probabilities, let the defender use *D1* with probability p and *D2* with probability $1 - p$ as their strategies, and let the attacker use *A1* with probability q and *A2* with $1 - q$ as their strategies.

Defender's indifference condition:

- For the first strategy (D1):

$$E_D(D1) = 4.1q + (-4)(1 - q) = 4.1q - 4 + 4q = 8.1q - 4$$

- For the second strategy (D2):

$$E_D(D2) = -2.8q + 4.2(1 - q) = -2.8q + 4.2 - 4.2q = -7q + 4.2$$

At the Nash equilibrium, both strategies must have the same expected payoff for the defender (both have to be played with positive probability):

$$8.1q - 4 = -7q + 4.2$$

$$8.1q + 7q = 4.2 + 4$$

$$15.1q = 8.2$$

$$q = \frac{8.2}{15.1} \approx 0.54 = 54\%$$

It means that the attacker plays A1 with probability $q = \frac{8.2}{15.1} = 54\%$, and A2 with probability $q = \frac{6.9}{15.1} = 46\%$.

Attacker's indifference condition:

- For the first strategy (A1):

$$E_A(A1) = -0.8p + 2.1(1 - p) = -0.8p + 2.1 - 2.1p = -2.9p + 2.1$$

- For the second strategy (A2):

$$E_A(A2) = 0.8p + (-0.6)(1 - p) = 0.8p - 0.6 + 0.6p = 1.4p - 0.6$$

At the Nash equilibrium, both strategies must give equal expected payoff to the attacker:

$$-2.9p + 2.1 = 1.4p - 0.6$$

$$-2.9p - 1.4p = -0.6 - 2.1$$

$$-4.3p = -2.7$$

$$p = \frac{-2.7}{-4.3} \approx 0.63 = 63\%$$

It means that the defender plays D1 with probability $P = \frac{-2.7}{-4.3} = 63\%$, and D2 with probability $q = \frac{-1.6}{-4.3} = 37\%$.

The results from both Nash equilibrium and mathematics were the same.

5.2 Pure ABAC Simulation Results

The first simulation simulated an ABAC-only policy environment, which can be used to compare against. This simulation quantifies the impact of using only flexible yet vulnerable access control policies, which helps to determine the security impact.

5.2.1 Breach Rate Analysis

Simulation resulted in 1367 successful breaches of 5000 access attempts, which defined the ultimate breach rate of around 27%. It reflects the large susceptibility of pure ABAC systems to both attacks that exploit the versatility of attribute-based policies.

Figure 7 shows that the breach rate settled at around 0.27 (27%) for most of the simulation, which reflects a highly consistent susceptibility to attacks. This stabilization results from attackers converging to their best strategy mix against pure ABAC.

After stabilization, the breach rate fluctuated very little, which implies that pure ABAC systems adopt the same security posture with time, lacking the means to respond to evolving patterns of attacks.

These results demonstrate the unmodified ABAC implementations' security vulnerabilities, particularly where Phishing is a habitual exploitation mode. ABAC is very configurable to support the needs of legitimate users while this configurability comes at the expense of high security unless accompanied with more systematic approaches to access control.

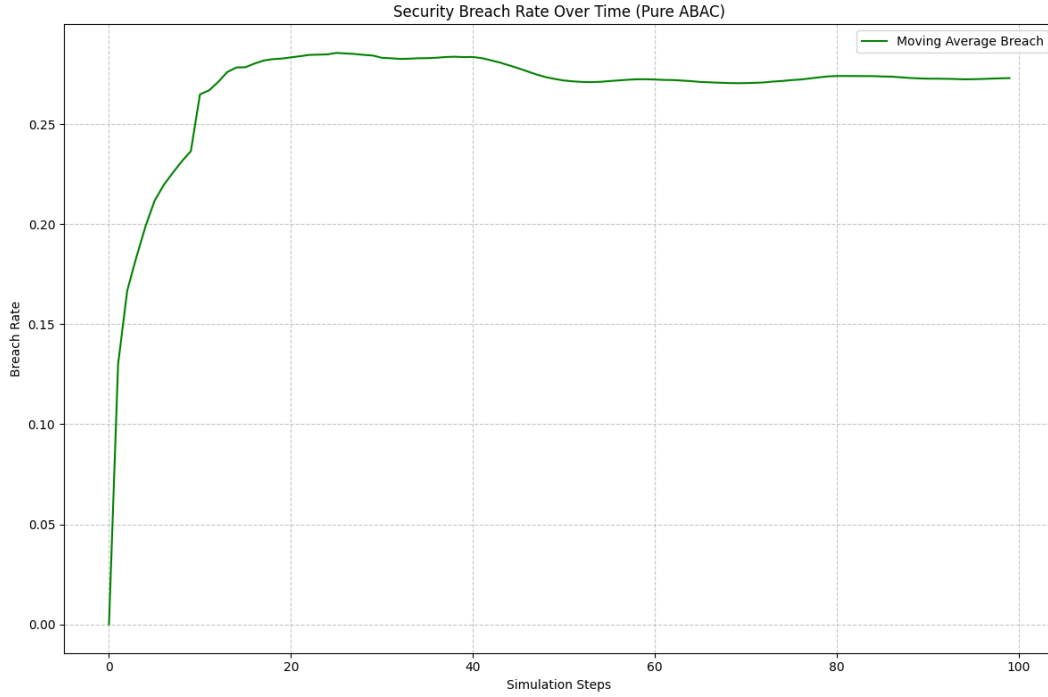


Figure 7. Breach rate over time using Pure ABAC. The x-axis shows time in simulation steps, and the y-axis shows the breach rate as a percentage

5.3 Pure RBAC Simulation Results

The second simulation simulated an RBAC-only policy environment, which can also be used to compare against. This simulation quantifies the impact of using only strict but less flexible access control policies.

5.3.1 Breach Rate Analysis

The pure RBAC simulation produced 832 successful breaches out of 5 000 access attempts, giving an overall breach rate of around 17%, which is 10% less compared to the pure-ABAC scenario.

Figure 8 shows how the rate changes over time. It rises quickly at first, then slides down and levels off around 17% for the rest of the run. After that point attackers and defenders reach a stand-off, and the number barely moves.

The lower breach probability compared with ABAC reflects RBAC's tighter, role-centric gates: attackers must compromise a specific role or elevate privileges, rather than exploit a diverse set of attributes. However, the numbers also show that almost one in five access attempts still succeeds for the adversary, underscoring several RBAC limitations:

RBAC's clear role boundaries do make life harder for attackers than ABAC does, because an intruder has to steal or misuse a specific role instead of manipulating a wide set of loose attributes. But that edge is limited, the privileges tied to each role are fixed, so when a high-level role is cracked, the system has no automatic way to dial those rights back. As a result, the breach rate still sits at around 17 percent, about one in five attempts, which is still higher than most organizations can safely tolerate.

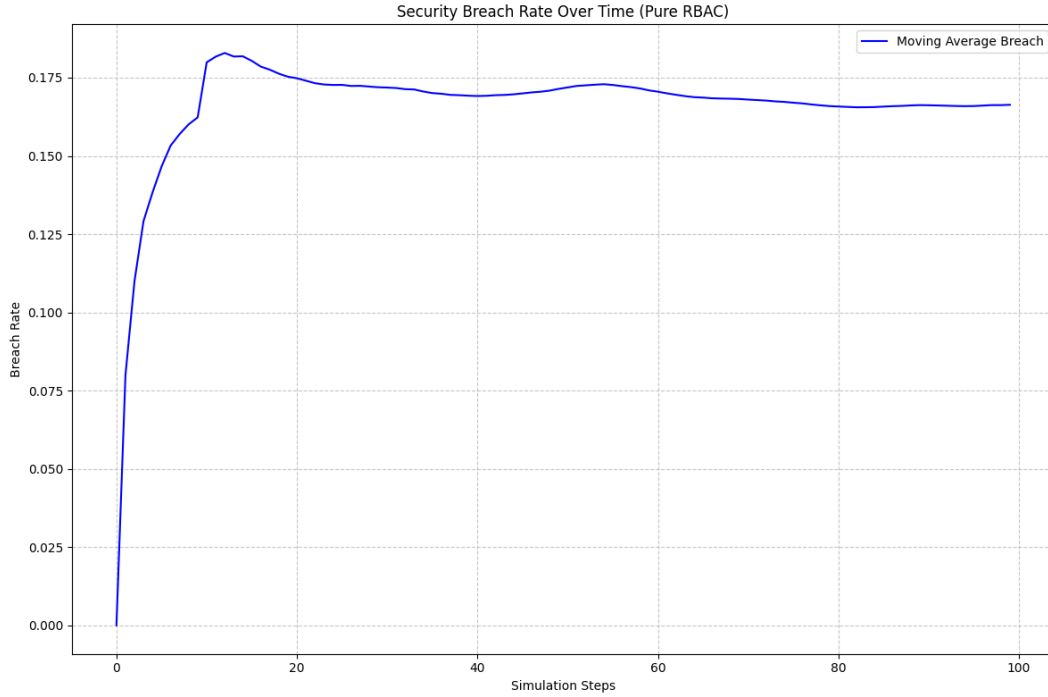


Figure 8. Breach rate over time using Pure RBAC. The x-axis shows time in simulation steps, and the y-axis shows the breach rate as a percentage

5.4 Hybrid Policy Simulation Results

The Third simulation applied the optimal policy mix that resulted from the Nash Equilibrium analysis, which enables defenders to dynamically change their policies between RBAC and ABAC depending upon observed rates of breach. This simulation evaluates the operational effectiveness of the access control optimization approach based upon game theory.

5.4.1 Policy Mix Evolution and Breach Rate Analysis

Figure 9 shows that, as the simulation progressed the policy mix converged to a value close to 61% RBAC and 39% ABAC, while the Nash Equilibrium for this policy space was roughly 63% RBAC and 37% ABAC.

The simulation was run for 5000 access attempts. The number of successful breaches was 562. Figure 10 shows that the final breach rate was around 11%. This is a nearly 16% reduction for the breach rate in the pure ABAC scenario (27%). And a nearly 06% reduction for the breach rate in the pure RBAC scenario (17%).

The defense agent adjusted the policy mix based on the change in the breach rate, varying the components of the policy mix when the breach rate went up or down. When the breach rate went up, the policy mix shifted to incorporate more RBAC components into it, which can be observed from figure 9, where the policy mix shifts towards more RBAC components with the rise in breach rate.

Both policy mix and breach rate remained highly stable after the initial convergence period, the last moving average breach rate settling at 11.32% and the instant breach rate at 11.24%. In this study, stability suggests that the Nash Equilibrium is a robust solution to the access-control optimization problem, and it implies that the optimal policy mix, once determined, can be maintained with minimal revisions over time.

The observed breach rate turned out to be 11%, and the final policy mix turned out to be 61% RBAC and 39% ABAC.

From the results of the game theory model, the values are fairly close, and the difference between the observed and the predicted values can be ascribed to the stochastic nature of the simulation, as well as the adaptive behavior of both defenders and attackers.

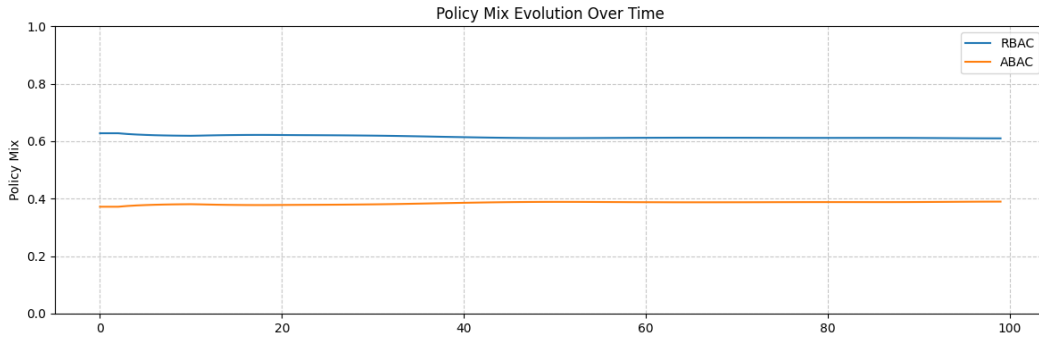


Figure 9. Evolution of policy mix between RBAC and ABAC over time in the hybrid model. The x-axis represents simulation time steps, and the y-axis shows the percentage of each policy in use

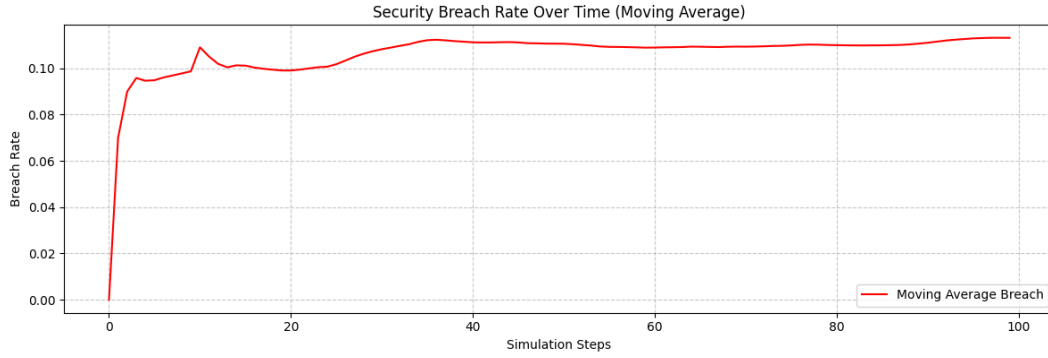


Figure 10. Breach rate over time using a hybrid approach (ABAC + RBAC).
The x-axis represents time in simulation steps, and the y-axis shows the breach rate as a percentage

6 Analysis/Discussion

The results of the study support the applicability and academic value utilizing game theory, more specifically Nash Equilibrium, as an approach for optimizing access control policies. The explanation in this section puts forward an interpretation for the results in relation to the study purpose, theoretical models, and real-world application.

6.1 Interpretation of Nash Equilibrium Outcomes

The fundamental outcome of the Nash Equilibrium calculation was to recommend that system defenders use about 63% RBAC and 37% ABAC, with attackers prefer using Token Theft (46%) rather than Phishing (54%). This equilibrium dictates that RBAC is still an even better mechanism for reducing successful breaches because it is more rigid, thereby reducing the surface area that can be attacked. The use of ABAC is still a necessity to avoid compromising system flexibility as well as to support dynamic requirements in operations.

These derived equilibrium values aren't hypothetical, but calculated through empirical cost estimates gathered from a reliable database such as IBM's Data Breach Report 2024, and success rates gathered from the simulation environments. This empirical basis in actual cybersecurity data lends credibility to real-world applicability and enhances the case for game-theoretic models in cybersecurity decision making [21].

6.2 Simulation Insights

Agent-based simulations also gave empirical support to Nash-equilibrium strategies. Under pure ABAC configuration, the breach rate converged to an average value of 27%, in line with the documented weakness of ABAC to Phishing attacks. Under pure RBAC configuration, the breach rate converged to an average value of 17% which is better than pure ABAC but still higher than most organizations can safely tolerate. In contrast, using the equilibrium-based hybrid policy caused the reduction in the breach rate to 16% compared to pure ABAC and 06% compared to pure RBAC. This result confirms the strategic superiority of a hybrid policy and verifies theoretically derived results in an empirical context. Figure 11 and table 3 shows a comparison of breach rate between all three simulations.

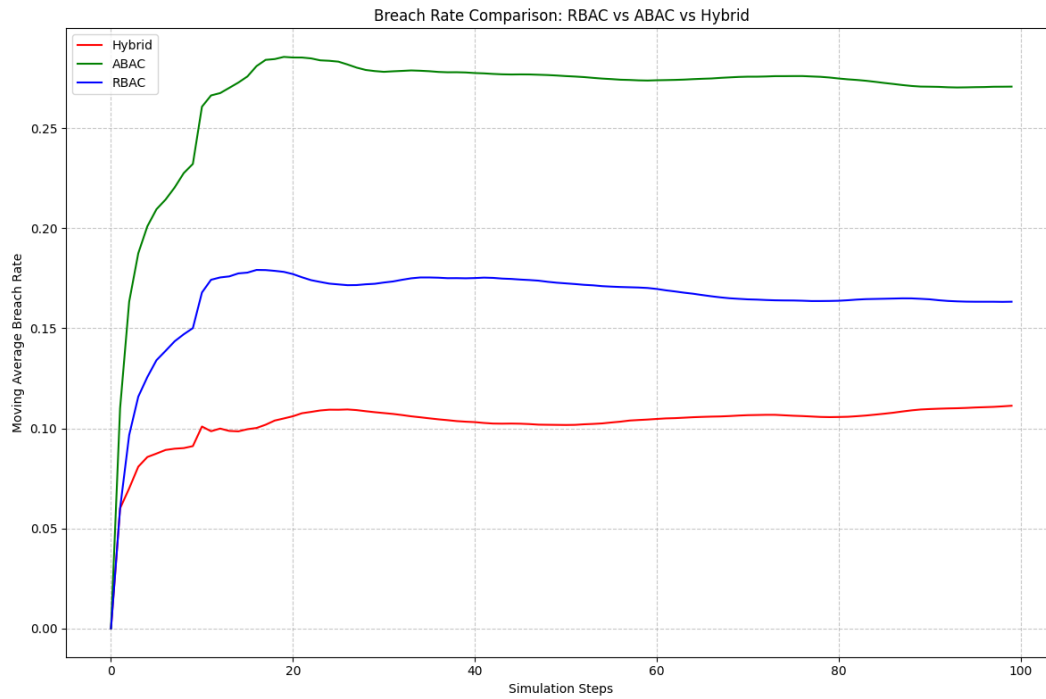


Figure 11. Breach rate comparison: ABAC vs RBAC vs Hybrid

Table 3. Comparative summary of breach rates across ABAC, RBAC and Hybrid simulations

Model	Final breach rate	Reduction vs ABAC	Reduction vs RBAC
Pure-ABAC	27%	-	-
Pure-RBAC	17%	10%	-
Hybrid (Nash guided)	11%	16%	06%

The adaptive response by defender agents in the simulation revealed that Nash strategy is an equilibrium attractor in the long term. While attacker strategies were being adjusted, defenders were countering by enhancing RBAC elements. This feedback is indicative of the sustainability of the model in the dynamic cyber threat landscape, an essential element in any anticipatory cybersecurity system.

6.3 Strategic Flexibility and Real-World Alignment

The resulting hybrid approach is especially suited to organizations that need to reconcile strict compliance with operational flexibility. In highly security-conscious sectors like banks, more reliance on RBAC could be in order. In more dynamic settings such as software companies or educational institutions, ABAC can provide their needed flexibility, if constrained effectively.

Therefore, this approach does not support an invariant policy, but an adaptive policy formulated based on strategic interaction modeling. This aspect is harmonious with existing adaptive cybersecurity as well as cyber-resilience paradigms that demand adaptable, context-sensitive, and risk-based control strategies.

6.4 Limitations

Although the model offers convincing results, it is not without faults. First, no earlier study covered this exact scenario, so no published figures existed for the payoff matrix. The numbers of the payoff matrix were produced through the average cost of each attack, reported in IBM's report, and through success rates from the simulations environments that were created for that purpose [21].

Second, those success rates came solely from the simulation. Real-world organizations can differ, and their own settings might have varying levels of vulnerability to Phishing and Token Theft.

Third, although the simulation environment is encompassing, it abstracts from some system details such as insider threats and heterogeneity among users.

Also, the model is not designed to include dynamic strategic shifts over time, such as learning curves on the part of attackers, defender updates in response to compliance changes, or new threats such as AI-based Phishing. However, these limitations point to future areas for improvement and not to negate the conclusions in the present study.

6.5 Contribution to the Field

The value added in this research is in applying game-theoretic concepts in an area that historically is based on static rule-based systems. By moving from reactive to strategic, this research presents a new paradigm for decision making for cybersecurity professionals. Not just an optimization methodology, it is also a conceptual tool for viewing attacker-defender dynamics in access control to understand them and even anticipate them.

Additionally, the study enriches academic discussion by illustrating that theoretical concepts such as Nash Equilibrium are not limited to use in economic models but can actually resolve meaningful engineering challenges in cybersecurity. The dual validation for both mathematical and empirical contributes positively to adding a solid basis for further research and practical application.

7 Conclusions

In summary, the project presented a comprehensive analysis of the optimal design of access control policies through the lens of game theory using Nash Equilibrium. The overall objective of the research was to model quantitatively the strategic interplay between attackers and defenders when it comes to the choice of access control mechanisms by selecting Role-Based Access Control (RBAC) and Attribute-Based Access Control (ABAC) to harmonize both operational flexibility and robustness to attacks.

The Nash Equilibrium results suggested a hybrid access control design of around 63 % RBAC and around 37 % ABAC. And the simulation also results in nearly numbers with 61 % RBAC and 39 % ABAC. This was found to be optimal in terms of lessening the successful breaches while ensuring adequate usability to achieve organizational effectiveness. The Nash-based policy was also empirically confirmed using extensive agent-based simulations. The simulations proved to result in a marked increase in security with breach rate reductions of around 16 % relative to systems using pure ABAC and around 06 % relative to systems using pure RBAC.

While having such strengths, research also identifies weaknesses like reliance on simulated success rates of attacks and modeled simplification of dynamic interactions between attackers and defenders. Future research can also extend and augment the model using real data sets and include threat vectors like insider threats and dynamic models to represent changing attackers.

Lastly, the research contributes to the cybersecurity discipline as it presents a proactive and strategic mode of making access control decisions. The use of such game-theoretic models as Nash Equilibrium equips system administrators with a robust tool to

anticipate and strategize against cyber attacks and open up more resilient, adaptable, and safe systems.

7.1 Future Work

Future research can be done to extend multi-player game framework to include different attackers and defenders with distinct resources and objectives. Integrating insider threats and zero-day attacks within the framework will be helpful in capturing the issues related to access control more effectively.

Real-time feedback systems that incorporate machine learning with the game-theoretic framework will be able to provide dynamic policy adjustment to correspond to observed behaviors and improve robustness and adaptability.

In conclusion, this research lays the groundwork for further research in designing strategic cybersecurity policy and contributes to the literature on access control in a new way by aligning theory with empirical evidence and validation.

References

- [1] V. C. Hu, D. Ferraiolo, R. Kuhn, A. Schnitzer, K. Sandlin, R. Miller, and K. Scarfone, "Guide to Attribute Based Access Control (ABAC) Definition and Considerations," NIST Special Publication 800-162, Jan. 2014. [Online]. Available: <https://nvlpubs.nist.gov/nistpubs/specialpublications/NIST.SP.800-162.pdf> [Accessed: Mar. 31, 2025].
- [2] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, "Role-based access control models," IEEE Computer, vol. 29, no. 2, pp. 38-47, Feb. 1996. [Online]. Available: <https://csrc.nist.gov/csrc/media/projects/role-based-access-control/documents/sandhu96.pdf> [Accessed: Mar. 30, 2025].
- [3] J. Blythe, V. Kothari, S. Smith, and R. Koppel, "Usable Security vs. Workflow Realities," NDSS Workshop on Usable Security, San Diego, CA, USA, Feb. 18, 2018, doi: 10.14722/usec.2018.23037. Available: https://www.researchgate.net/publication/326833373_Usable_Security_vs_Workflow_Realities [Accessed: Aug. 30, 2025].
- [4] H. Tavafoghi, Y. Ouyang, D. Teneketzis, and M. P. Wellman, "Game theoretic approaches to cyber security: Challenges, results, and open problems," in Adversarial and Uncertain Reasoning for Adaptive Cyber Defense, Lecture Notes in Computer Science, vol. 11830, Springer, 2019, pp. 29–53. [Online]. Available: <https://teneketzis.engin.umich.edu/wp-content/uploads/sites/370/2020/01/Game-Theoretic-Approaches-to-Cyber-Security.pdf> [Accessed: Mar. 29, 2025].
- [5] D. F. Ferraiolo, R. Sandhu, S. Gavrila, D. R. Kuhn, and R. Chandramouli, "Proposed NIST standard for role-based access control," ACM Transactions on Information and System Security, vol. 4, no. 3, pp. 224-274, Aug. 2001. [Online]. Available: https://www.researchgate.net/publication/220593590_Proposed_NIST_Standard_for_Role_Based_Access_Control [Accessed: Mar. 30, 2025].
- [6] E. Yuan and J. Tong, "Attributed based access control (ABAC) for Web services," in Proc. IEEE International Conference on Web Services (ICWS'05), Orlando, FL, USA, 2005, pp. 561-569. [Online]. Available: https://www.researchgate.net/publication/4186774_Attributed_Based_Access_Control_ABAC_for_web_services [Accessed: Apr. 03, 2025].
- [7] D. R. Kuhn, E. J. Coyne, and T. R. Weil, "Adding attributes to role-based access control," IEEE Computer, vol. 43, no. 6, pp. 79-81, June 2010. [Online]. Available: https://www.researchgate.net/publication/260584013_Adding_Attributes_to_Role-Based_Access_Control [Accessed: Apr. 04, 2025].
- [8] X. Jin, R. Krishnan, and R. Sandhu, "A unified attribute-based access control model covering DAC, MAC and RBAC," in Proc. IFIP Annual Conference on Data and Applications Security and Privacy, Paris, France, 2012, pp. 41-55. [Online]. Available: https://doi.org/10.1007/978-3-642-31540-4_4 [Accessed: Apr. 06, 2025].
- [9] T. Alpcan and T. Başar, "A Game Theoretic Approach to Decision and Analysis in Network Intrusion Detection," in Proceedings of the IEEE Conference on Decision and Control, vol. 3, Maui, HI, USA, Dec. 2003, pp. 2595–2600.

- Available: https://www.researchgate.net/publication/4058212_A_Game_Theoretic_Approach_to_Decision_and_Analysis_in_Network_Intrusion_Detection [Accessed: Apr. 07, 2025].
- [10] S. Roy, C. Ellis, S. Shiva, D. Dasgupta, V. Shandilya, and Q. Wu, "A survey of game theory as applied to network security," in Proc. 43rd Hawaii International Conference on System Sciences, Honolulu, HI, USA, 2010, pp. 1-10. [Online]. Available: https://www.researchgate.net/publication/224123416_A_Survey_of_Game_Theory_as_Applied_to_Network_Security [Accessed: Apr. 09, 2025].
- [11] I. Molloy, L. Dickens, C. Morisset, P. C. Cheng, J. Lobo, and A. Russo, "Risk-based security decisions under uncertainty," in Proc. 2nd ACM Conference on Data and Application Security and Privacy (CODASPY), San Antonio, TX, USA, 2012, pp. 157–168. [Online]. Available: https://www.researchgate.net/publication/254006830_Risk-based_security_decisions_under_uncertainty [Accessed: Apr. 10, 2025].
- [12] M. H. Manshaei, Q. Zhu, T. Alpcan, T. Başar, and J.-P. Hubaux, "Game theory meets network security and privacy," ACM Computing Surveys, vol. 45, no. 3, pp. 1-39, June 2013. [Online]. Available: https://www.researchgate.net/publication/46378489_Game_Theory_Meets_Network_Security_and_Privacy [Accessed: Apr. 10, 2025].
- [13] P. C. Cheng, P. Rohatgi, C. Keser, P. A. Karger, G. M. Wagner, and A. S. Reninger, "Fuzzy multi-level security: An experiment on quantified risk-adaptive access control," in IEEE Symposium on Security and Privacy, 2007, pp. 222-230. [Online]. Available: https://www.researchgate.net/publication/4251885_Fuzzy_Multi-Level_Security_An_Experiment_on_Quantified_Risk-Adaptive_Access_Control [Accessed: Apr. 11, 2025].
- [14] National Institute of Standards and Technology (NIST), Risk Management Framework for Information Systems and Organizations: A System Life Cycle Approach for Security and Privacy, NIST Special Publication 800-37, Revision 2, Dec. 2018. [Online]. Available: <https://doi.org/10.6028/NIST.SP.800-37r2> [Accessed: Apr. 13, 2025].
- [15] W. Jansen, "Directions in security metrics research," NIST Interagency Report 7564, Apr. 2009. [Online]. Available: <https://doi.org/10.6028/NIST.IR.7564> [Accessed: Apr. 13, 2025].
- [16] I. Molloy, P. C. Cheng, and P. Rohatgi, "Trading in risk: Using markets to improve access control," in Proc. 2008 New Security Paradigms Workshop (NSPW), Lake Tahoe, CA, USA, 2008, pp. 107–125. [Online]. Available: https://www.researchgate.net/publication/228360677_Trading_in_risk_Using_markets_to_improve_access_control [Accessed: Apr. 14, 2025].
- [17] J. Nash, "Non-cooperative games," Annals of Mathematics, vol. 54, no. 2, pp. 286–295, Sep. 1951. [Online]. Available: <https://doi.org/10.2307/1969529> [Accessed: Apr. 23, 2025].
- [18] Wikipedia contributors, "Nash equilibrium," Wikipedia, The Free Encyclopedia, [Online]. Available: https://en.wikipedia.org/wiki/Nash_equilibrium [Accessed: Apr. 25, 2025].

- [19] "Mixed Strategy Nash Equilibrium (Game Theory Playlist 4)," YouTube, 2019. [Online]. Available: <https://youtu.be/IjgYLM4KgFg> [Accessed: Apr. 30, 2025].
- [20] A. Mohammad, G. Kanaan, R. Kanaan, T. Khmour, S. Bani-Ahmad, and A. Alarabeyyat, "Toward access control model for Web Services applications," *Int. J. of Research and Reviews in Computer Science (IJRRCS)*, vol. 2, no. 2, pp. 253–264, Apr. 2011. [Online]. Available: https://scholar.google.com/citations?view_op=view_citation&hl=en&user=5wHOexgAAAAJ&citation_for_view=5wHOexgAAAAJ:9yKSN-GCB0IC [Accessed: May. 02, 2025].
- [21] IBM Security, Cost of a Data Breach Report 2024, IBM Corporation, 2024. [Online]. Available: <https://www.ibm.com/reports/data-breach> [Accessed: Apr. 17, 2025].
- [22] Clavister Holding AB, *Annual Report 2024*, Örnköldsvik, Sweden: Clavister, Apr. 28, 2025. [Online]. Available: <https://www.clavister.com/wp-content/uploads/2025/04/Clavister-Annual-Report-2024.pdf> [Accessed Jun. 10, 2025].
- [23] "The rising demand for non-tech workers in tech companies," *Business-Money.com*, Apr. 4, 2021. [Online]. Available: <https://www.business-money.com/announcements/the-rising-demand-for-non-tech-workers-in-tech-companies/> [Accessed Jun. 10, 2025].
- [24] M. Dengsøe, "Data and product to engineer ratio at 50 tech scaleups," *Medium*, Apr. 13, 2023. [Online]. Available: <https://medium.com/@milkdd/data-and-product-to-engineer-ratio-at-50-tech-scaleups-21a38dfb3200> [Accessed Jun. 10, 2025].
- [25] P. Sirawongphatsara, P. Pornpongtechavanich, N. Phanthuna, and T. Daengsi, "Comparative Simulation of Phishing Attacks on a Critical Information Infrastructure Organization: An Empirical Study," *arXiv*, Oct. 2024. [Online]. Available: <https://arxiv.org/abs/2410.20728> [Accessed Jun. 10, 2025].
- [26] KnowBe4 Inc., "KnowBe4 Releases Q1 2025 Phishing Report, With Internal Communications Dominating," Tampa Bay, FL, Apr. 28, 2025. [Online]. Available: <https://www.knowbe4.com/press/knowbe4-releases-q1-2025-phishing-report-with-internal-communications-dominating> [Accessed Jun. 10, 2025].
- [27] ScatteredSecrets.com, "Protecting your customers against account takeover fraud," *Medium*, Jul. (?), 2022. [Online]. Available: <https://scatteredsecrets.medium.com/protecting-your-customers-against-account-takeover-fraud-97c9377fe5d1> [Accessed Jun. 10, 2025].
- [28] U. Oner, O. Cetin, and E. Savas, "Human factors in phishing: Understanding susceptibility and resilience," *ScienceDirect*, vol. 94, 2025, Art. no. 104014. [Online]. Available: https://www.researchgate.net/publication/391036047_Human_factors_in_phishing_Understanding_susceptibility_and_resilience [Accessed: May. 03, 2025].

- [29]S. Villavicencio, "What's new in the 2023 Cost of a Data Breach report?," *IBM Community – Security Global Forum*, Jul. 25, 2023. [Online]. Available: <https://community.ibm.com/community/user/blogs/sarah-dudley/2023/07/25/costofadatabreach2023> [Accessed Jun. 10, 2025].
- [30]N. Abu Hamdah, Y. Almoued "GameTheory" GitHub, Repository. [Online]. Available: <https://github.com/Discover1998/GameTheory>. [Accessed: May. 17, 2025].
- [31]B. Solomon, "Python Plotting With Matplotlib (Guide)," Real Python, 2020. [Online]. Available: <https://realpython.com/python-matplotlib-guide/> [Accessed: May. 04, 2025].
- [32]Project Mesa Team, Mesa: Agent-based modeling in Python, version 3.2.0, Mesa Documentation, May 2025. [Online]. Available: <https://mesa.readthedocs.io/stable/index.html> [Accessed: Apr. 15, 2025].
- [33]V. Knight, Tutorial: Building and Finding the Equilibrium for a Game, Nashpy Documentation, May 2025. [Online]. Available: <https://nashpy.readthedocs.io/en/stable/tutorial/index.html> [Accessed: May. 02, 2025].

A. Figures and Tables

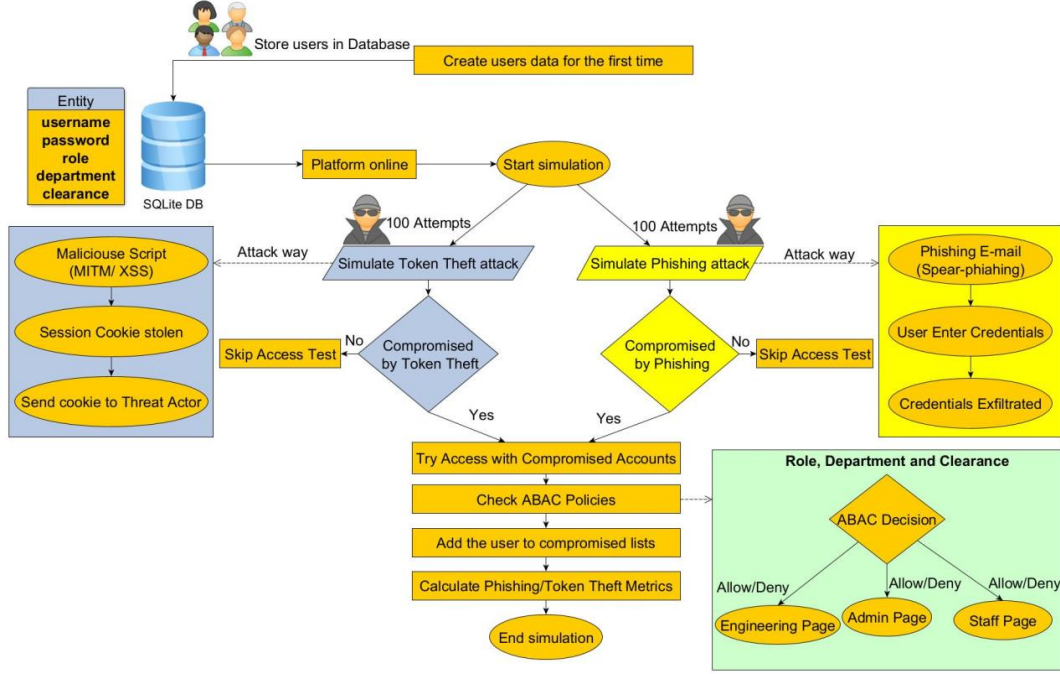


Figure 1. Flowchart of the ABAC Simulation Environment and Attack Vector Processing

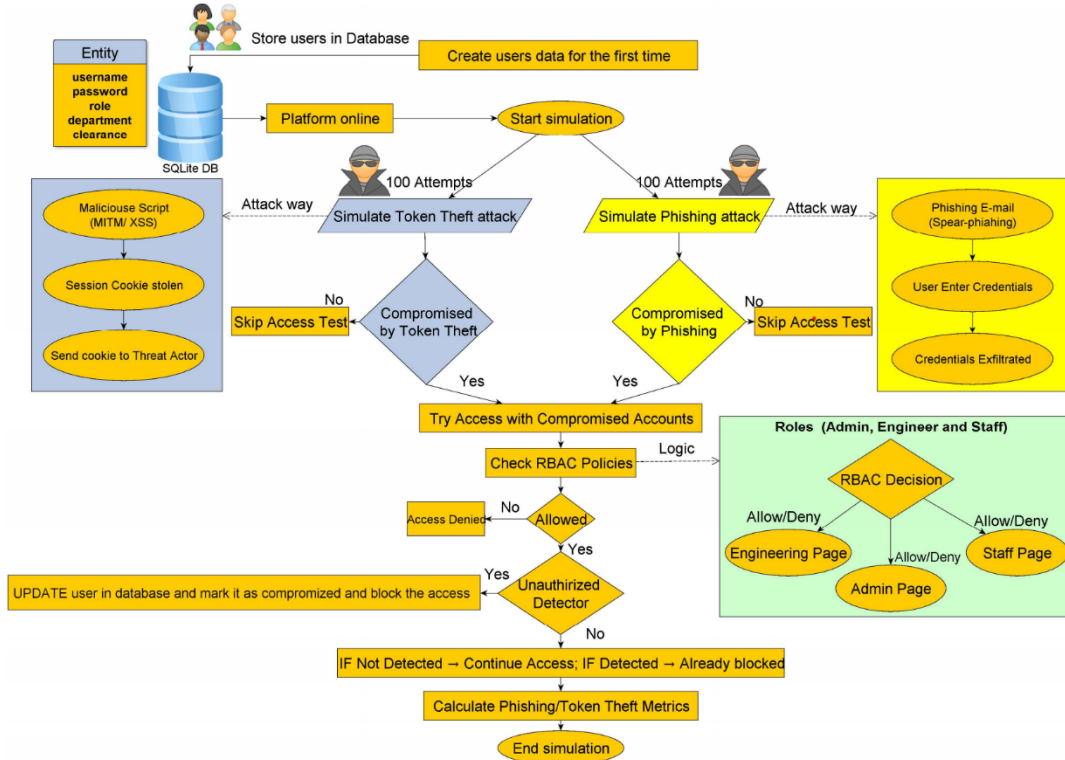


Figure 2. Flowchart of the RBAC Simulation Environment and Attack Vector Processing

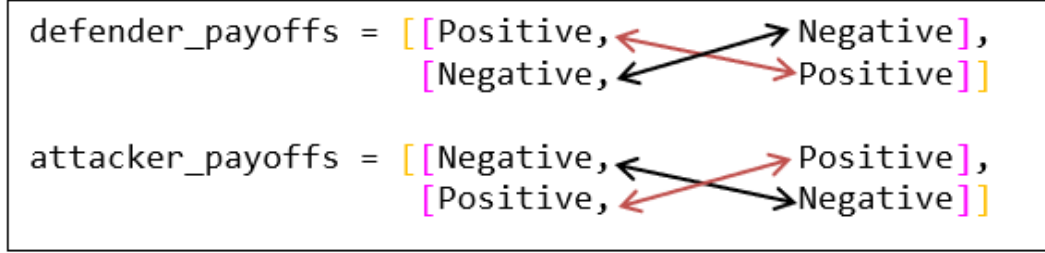


Figure 3. Sign assignment, strategy payoffs (original configuration)

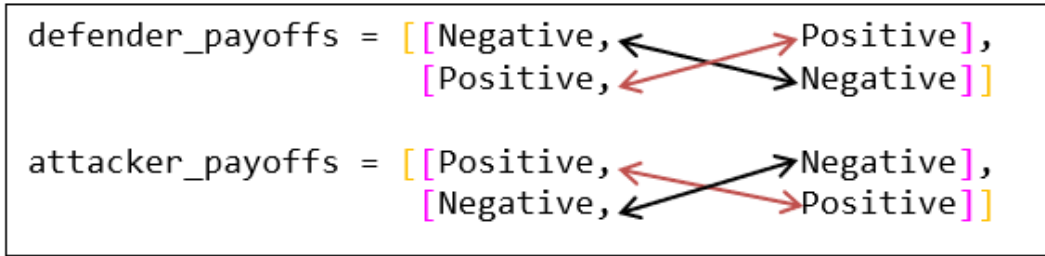


Figure 4. Sign assignment, strategy payoffs (inverted configuration)

defender_payoffs =	$\begin{bmatrix} 4.1 & -4.0 \\ -2.8 & 4.2 \end{bmatrix}$
attacker_payoffs =	$\begin{bmatrix} -0.8 & 0.8 \\ 2.1 & -0.6 \end{bmatrix}$

Figure 5. Nash equilibrium outcome matrix

```
import nashpy as nash

defender_payoffs = [[4.1, -4.0],
                   [-2.8, 4.2]]

attacker_payoffs = [-0.8, 0.8],
                   [2.1, -0.6]]

# Create the game
game = nash.Game(defender_payoffs, attacker_payoffs)

# Compute equilibria
equilibria = list(game.support_enumeration())
```

Figure 6. Simplified Nash Equilibrium code to calculate mixed strategies

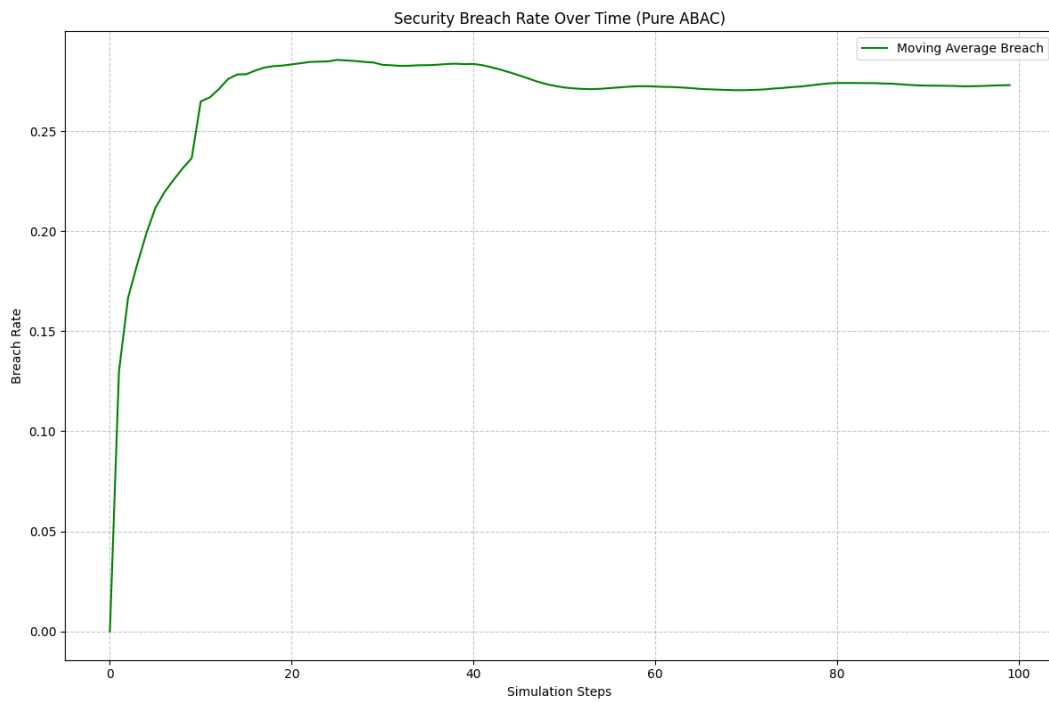


Figure 7. Breach rate over time using Pure ABAC. The x-axis shows time in simulation steps, and the y-axis shows the breach rate as a percentage

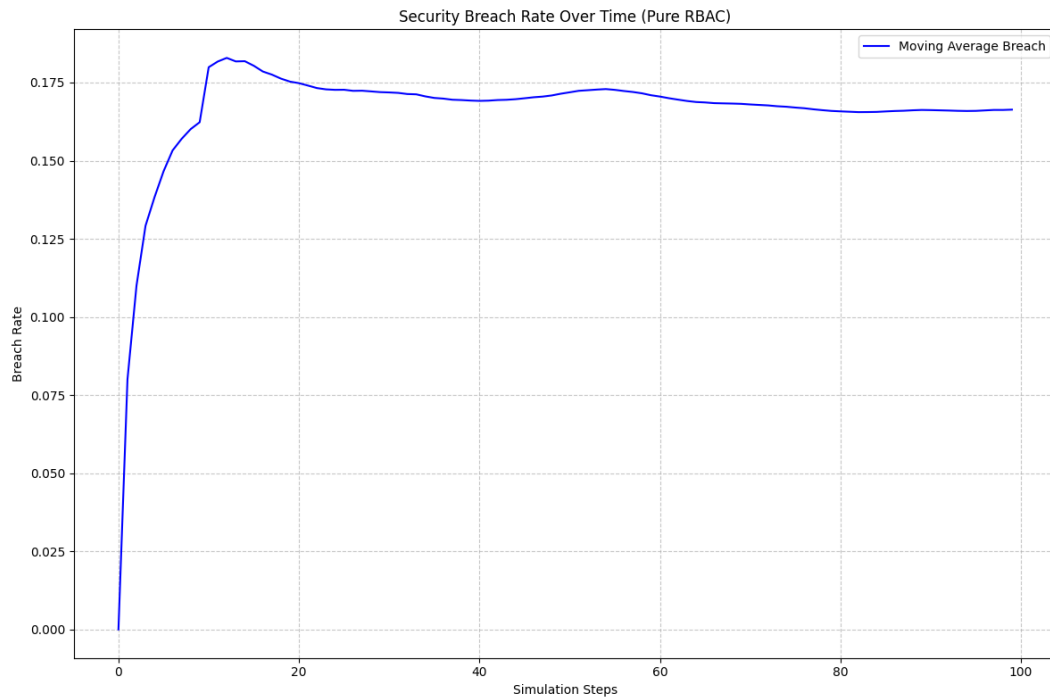


Figure 8. Breach rate over time using Pure RBAC. The x-axis shows time in simulation steps, and the y-axis shows the breach rate as a percentage

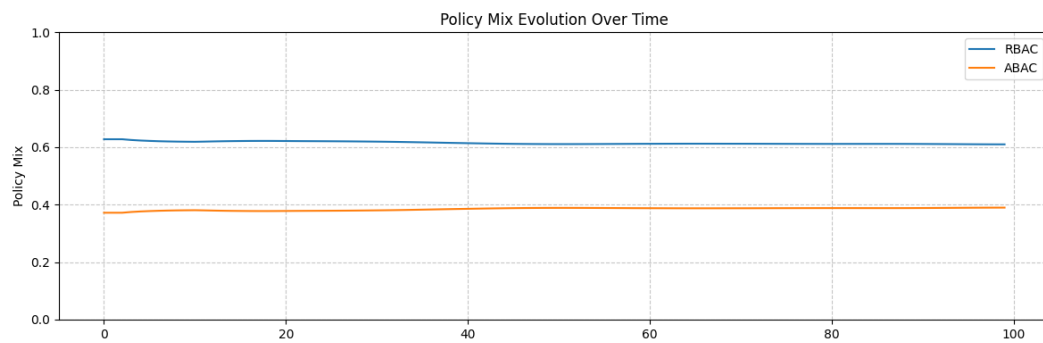


Figure 9. Evolution of policy mix between RBAC and ABAC over time in the hybrid model. The x-axis represents simulation time steps, and the y-axis shows the percentage of each policy in use

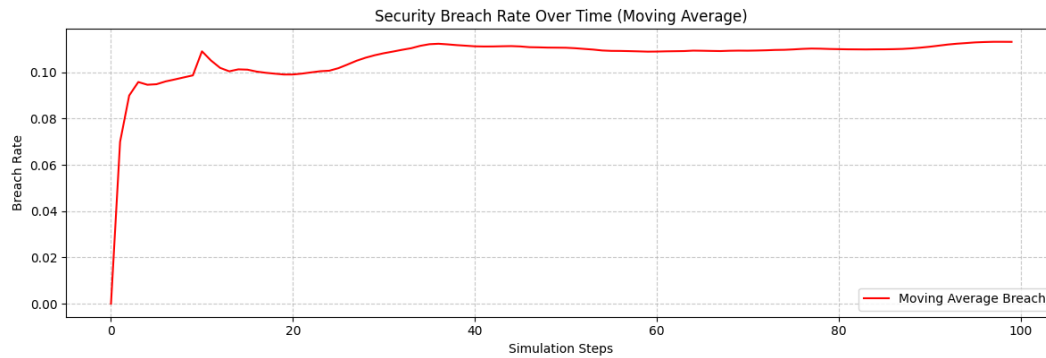


Figure 10. Breach rate over time using a hybrid approach (ABAC + RBAC). The x-axis represents time in simulation steps, and the y-axis shows the breach rate as a percentage

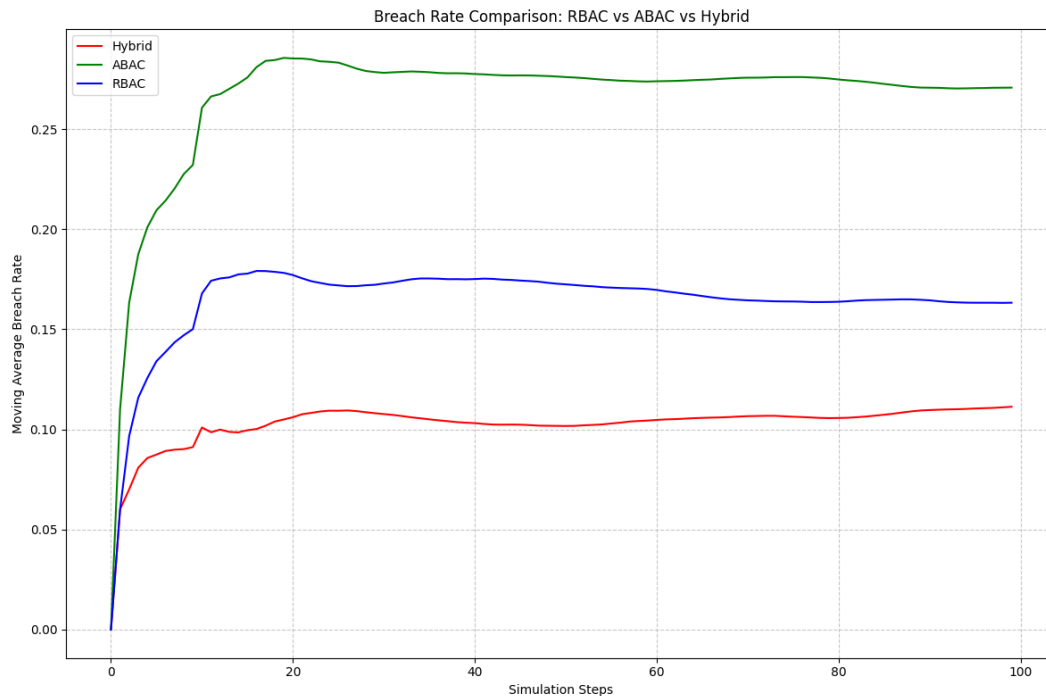


Figure 11. Breach rate comparison: ABAC vs RBAC vs Hybrid

Table 2. Success rates for Phishing and Token Theft Attacks

Role	Phishing Probability	Token Theft Probability
Admin	11%	5%
Engineer	11%	5%
Staff	61%	15%

Table 2. Strategic payoff matrix for defender and attacker

Defender / Attacker	Phishing	Token Theft
Strict RBAC	(+4.1, -0.8)	(-4, +0.8)
Loose ABAC	(-2.8, +2.1)	(+4.2, -0.6)

Table 3. Comparative summary of breach rates across ABAC, RBAC and Hybrid simulations

Model	Final breach rate	Reduction vs ABAC	Reduction vs RBAC
Pure-ABAC	27%	-	-
Pure-RBAC	17%	10%	-
Hybrid (Nash guided)	11%	16%	06%