Ensino que te conecta

BACHARELADO EM

ENGENHARIA DE SOFTWARE

ACTIVE MACHINE LEARNING APPLIED TO INTRUSION DETECTION IN USER AUTHENTICATION

GUSTAVO BEE CAMPOS ROCHA

Brasília - DF, 2025

GUSTAVO BEE CAMPOS ROCHA

ACTIVE MACHINE LEARNING APPLIED TO INTRUSION DETECTION IN USER AUTHENTICATION

Trabalho de Conclusão de Curso apresentado como requisito parcial para a obtenção de grau de Bacharel em Engenharia de Software, pelo Instituto Brasileiro de Ensino, Desenvolvimento e Pesquisa (IDP).

Orientadora
MSc. Lorena de Souza Bezerra Borges

Brasília - DF, 2025

Código de catalogação na publicação – CIP

R672a Rocha, Gustavo Beé Campos

Active machine learning applied to intrusion detection in user authentication / Gustavo Beé Campos Rocha. — Brasília: Instituto Brasileiro Ensino, Desenvolvimento e Pesquisa, 2025.

62 f. : il: color.

Orientador: Profa. Mes. Lorena de Souza Bezerra Borges

Trabalho de Conclusão de Curso (Graduação em Engenharia de Software) — Instituto Brasileiro Ensino, Desenvolvimento e Pesquisa – IDP, 2025.

1. Detecção de intrusão. 2. Aprendizado ativo. 3. Aprendizado de máquina. 4. Cibersegurança. I. Título

CDD 005.1

Elaborada por Natália Bianca Mascarenhas Puricelli – CRB 1/3439

GUSTAVO BEE CAMPOS ROCHA

ACTIVE MACHINE LEARNING APPLIED TO INTRUSION DETECTION IN USER AUTHENTICATION

Trabalho de Conclusão de Curso apresentado como requisito parcial para a obtenção de grau de Bacharel em Engenharia de Software, pelo Instituto Brasileiro de Ensino, Desenvolvimento e Pesquisa (IDP).

Aprovado em 02/12/2025

Banca Examinadora

gov.br Documento assinado digitalmente
LORENA DE SOUZA BEZERRA BORGES
Data: 19/12/2025 14:23:43-0300
Verifique em https://validar.iti.gov.br

MSc. Lorena de Souza Bezerra Borges- Orientadora

gov.br Documento assinado digitalmente
KLAYTON RODRIGUES DE CASTRO
Data: 20/12/2025 13:45:51-0300
Verifique em https://validar.iti.gov.br

MSc. Klayton Rodrigues de Castro- Examinador Externo

gov.br Documento assinado digitalmente
FABRICIO FERNANDES SANTANA
Data: 20/12/2025 13:26:08-0300
Verifique em https://validar.iti.gov.br

Specialist Fabrício Fernandes Santana- Examinador Externo

DEDICATÓRIA

Dedico este momento à minha família e amigos, pelo apoio incondicional em todos os momentos nos últimos anos.

Um agradecimento especial à minha mãe, que passou por momentos difíceis. Obrigado por nunca desistir, você é a minha maior inspiração, eu te amo profundamente.

E à minha namorada Helenna, minha companheira em todos os momentos. Obrigado por me apoiar e deixar tudo mais leve.

Sou extremamente grato por fazerem parte da minha vida.

AGRADECIMENTOS

À minha orientadora, Lorena, meu muito obrigado. Agradeço por toda ajuda, mas, acima de tudo, pela compreensão e paciência demonstradas durante a elaboração deste trabalho. O suporte que recebi na reta final foi essencial para que eu mantivesse o foco e a qualidade. Sua orientação foi fundamental para que eu pudesse concluir essa etapa.

ABSTRACT

The exponential growth of cyber threats poses significant challenges to traditional intrusion detection systems, which often struggle with the massive volume of data and the high cost of manual labeling. This study investigates the application of Active Learning as a strategy to enhance attack detection efficiency in user authentication systems. Using the CIC-IDS2017 and LANL benchmark datasets, the research evaluated three distinct scenarios: infrastructure brute force, web application attacks, and post-authentication lateral movement. The methodology employed an offline simulation based on uncertainty sampling, where the model iteratively selected the most informative instances for training. The results demonstrate that active models, trained with only 120 samples, achieved detection rates comparable to fully supervised baselines trained on hundreds of thousands of examples. Specifically in the Web Attack scenario, the approach achieved a Recall of 89% and a Precision exceeding 99%, representing a labeling effort reduction of over 99.9%. Additionally, the application of a hybrid architecture to the LANL dataset allowed for the effective filtering of over 29,000 false positives. The computational analysis confirmed the viability of the solution for real-time implementation, with inference latencies in the order of microseconds.

Keywords: Intrusion Detection, Active Learning, Authentication, Machine Learning, Cybersecurity.

RESUMO

O crescimento exponencial das ameaças cibernéticas impõe desafios significativos aos sistemas de detecção de intrusão tradicionais, que frequentemente enfrentam dificuldades com o volume massivo de dados e o alto custo da rotulagem manual. Este estudo investiga o uso do Aprendizado Ativo (Active Learning) como estratégia para aumentar a eficiência na detecção de ataques em sistemas de autenticação. Utilizando os conjuntos de dados CIC-IDS2017 e LANL, a pesquisa avaliou três cenários: força bruta em infraestrutura, ataques a aplicações web e movimentação lateral pós-autenticação. A metodologia empregou uma simulação offline baseada em amostragem por incerteza, onde o modelo selecionou iterativamente as instâncias mais informativas para treinamento. Os resultados demonstram que modelos ativos, treinados com apenas 120 amostras, alcançaram taxas de detecção comparáveis a baselines supervisionados treinados com centenas de milhares de exemplos. No cenário de ataques Web, a abordagem atingiu um Recall de 89% e Precisão superior a 99%, representando uma redução no esforço de rotulagem superior a 99,9%. Adicionalmente, a aplicação de uma arquitetura híbrida no dataset LANL permitiu a filtragem eficaz de mais de 29.000 falsos positivos. A análise computacional confirmou a viabilidade da solução para tempo real, com latências de inferência na ordem de microssegundos.

Palavras-chave: Detecção de Intrusão, Aprendizado Ativo, Autenticação, Aprendizado de Máquina, Cibersegurança.

LIST OF FIGURES

LIST OF TABLES

CONTENTS


1

1 INTRODUCTION

Over the years, the context of security in technological evolution has progressed significantly, enabling improved system protection and new functionalities in software projects, leading to more robust and efficient systems. However, in parallel to this process, more tools and strategies are created to exploit vulnerabilities in programs, networks, and projects [6].

The impact of these vulnerabilities extends beyond technical risks to significant economic losses. According to the report ”The Cost of Poor Software Quality in the US: A 2022 Report” by the Consortium for Information & Software Quality (CISQ) [7], the cost of poor software quality (CPSQ) has reached approximately $2.41 trillion annually in the United States alone. Within this staggering figure, operational software failures and security breaches represent a significant portion. This foundational issue directly correlates with modern cybersecurity challenges, reinforcing the economic imperative for robust, automated detection mechanisms to mitigate these costs before they escalate into catastrophic failures.

This financial criticality is even more pronounced when vulnerabilities lead to actual security incidents. The 2025 Cost of a Data Breach Report shows that the average global cost of a breach reached USD 4.44 million, marking a 9% reduction compared to the previous year, largely driven by faster containment of breaches. However, this global decline is countered by regions like the United States, where the record average cost increased by 9%, reaching USD 10.22 million. In the threat landscape, the tactic of many attackers is no longer to ”break in” but rather to ”log in,” actively exploiting vulnerabilities created by loose access controls and accounts with excessive permissions. Although phishing is the most frequent attack vector (16% of incidents), resulting in an average cost of USD 4.8 million, malicious insider attacks stood out as the most expensive vector, costing an average of USD 4.92 million. Such incidents demonstrate that speed of response is critical: security teams that extensively use AI and automation managed to reduce the breach time by 80 days and lower their average costs by USD 1.9 million, emphasizing that investment in modern, phishing-resistant authentication, such as passkeys, is essential to strengthen human and machine identities and mitigate these financial risks [8].

These authentication and access control vulnerabilities directly undermine funda-

mental security principles encapsulated in the CIA Triad—Confidentiality, Integrity, and Availability [9]. While Availability concerns service continuity, this research specifically targets the preservation of Confidentiality and Integrity by detecting unauthorized access attempts and anomalous account behaviors. Preventing adversaries from accessing sensitive data or manipulating system states through compromised credentials is essential to maintaining these core security pillars in authentication systems.

The practical manifestation of these principles is evident in industry standards such as the Open Web Application Security Project (OWASP) Top 10 [10], where Broken Access Control is ranked as the number one security risk. Additionally, Identification and Authentication Failures occupies the seventh position, confirming that securing user identities remains a persistent and critical challenge. These classifications encompass well-known invasion techniques such as SQL injection and brute-force attacks, which frequently target access credentials or privilege manipulation. Various solutions have been developed to combat them, such as standard authentication frameworks and multi-factor authentication (MFA) mechanisms. These traditional approaches aim to enforce the CIA principles through rule-based controls. However, new threats and attack methods are created daily, aiming to steal a user's confidential information, which demands constant monitoring and the rapid implementation of new protection measures to mitigate risks.

While these traditional security mechanisms are effective for rule-based security, they often face a critical limitation: the inability to actively learn from new attack patterns, relying heavily on predefined rules and signatures. In this dynamic scenario, one of the most effective strategies to address this problem is the application of machine learning (ML) models and algorithms. This technology enables systems not only to detect known attack patterns but also to identify anomalous behaviors in real-time, anticipating threats before they materialize. It is applied to the prior detection of attacks such as fraud detection, phishing, authentication attacks, and intrusion [11]. Thus, machine learning emerges as an essential layer of protection, complementing traditional solutions and increasing system resilience against even unknown vulnerabilities.

In this regard, one of the resources most in need of protection lies in user authentication processes, particularly access control systems for web applications, digital banking platforms, and corporate systems. These mechanisms play a fundamental role in safeguarding personal data [12].

Although machine learning models have become essential for preserving this information, their application faces some limitations. This is because software systems handle a large amount of data, requiring high computational power and often making it infeasible to execute these models in real time [13].

In this context, active learning differentiates itself from traditional ML approaches by offering a solution that optimizes the use of labeled data and reduces the demand for

high computational power, emerging as a promising method for efficient real-time threat detection. While conventional methods require complete and fully labeled datasets — a costly and, in some cases, unfeasible process in highly dynamic environments — active learning operates through a selective mechanism that identifies the most informative samples, such as certain anomalous behaviors, and requests human labeling for critical cases. The main idea of this technique is to create a good classifier with a small set of pre-labeled data, with minimal human and computational effort [14].

The main objective of this work is to apply the active machine learning method in the detection and prevention of invasions in authentication systems. To achieve this goal, the specific objectives are proposed as follows:

- To evaluate the machine learning techniques and types capable of identifying attack patterns in real-time, while addressing the challenge of high computational cost;
- To compare traditional machine learning models with active learning approaches;
- To quantify the solutions found using performance metrics;
- To evaluate the human cost (e.g., time/effort) associated with data labeling in active learning versus traditional learning scenarios.

By proposing a comparative evaluation between conventional models and approaches based on active learning, this work aims to contribute to the advancement in the field of cybersecurity.

This work proceeds as follows: Chapter 2 covers the *Theoretical Foundations* and *Related Works*; Chapter 3 describes the *Methodology*; Chapter 4 analyzes the *Results*; and Chapter 5 presents *Conclusions* and *Future Work*.

2

2 LITERATURE REVIEW

This chapter establishes the conceptual and academic foundations necessary to support the development of this research, being structured into two primary sections: the Theoretical Framework and Related Works. Initially, the fundamental principles of information security are addressed, providing the necessary context to understand authentication vulnerabilities and the Machine Learning algorithms applied to defense. Subsequently, a critical analysis of the state-of-the-art is presented to contextualize the study within the current academic landscape. This review highlights existing methodologies and their contributions while identifying the specific technological gaps that motivate the adoption of the Active Learning approach proposed in this work.

2.1 THEORETICAL FRAMEWORK

To establish a solid foundation for the proposed methodology, this section reviews the fundamental concepts underpinning the study. It begins by exploring the core principles of user authentication, covering factor categorization and the evolution of modern authentication mechanisms such as Multi-Factor Authentication (MFA). Subsequently, the discussion addresses the role of Machine Learning in cybersecurity, detailing the classification algorithms relevant to anomaly detection. Finally, the theoretical basis for the performance metrics used in this study is presented, providing the necessary mathematical context to interpret the experimental results.

2.1.1 Authentication

Authentication is the process of verifying a user's identity by providing credential data or evidence that confirms the declared identity [15].

Authentication typically relies on one or more categories of verification factors, which can be combined to increase system robustness and security, such as [6]:

- **Something you know**: This category includes static passwords, PINs, or security questions. While widely used due to their simplicity, these factors are vulnerable to attacks;
- **Something you have**: This factor requires the user to possess a physical or virtual object to authenticate. Examples include codes generated by authenticator

apps, one-time codes sent via SMS, hardware tokens, push notifications to approve login attempts, and smart cards. These methods add a layer of physical security, as an attacker would need not only knowledge (the password) but also possession of the device or token;

- **Something you are**: Often considered the most secure factor, this authentication relies on the user's unique biometric characteristics, which are extremely difficult to replicate. This includes facial recognition, fingerprint scanning, retina or iris scanning, voice pattern detection, and, at the most advanced level, behavioral biometrics.

To further enhance security, critical systems adopt MFA, which requires the combination of two or more distinct factors (e.g., password + SMS code). This approach is essential for mitigating threats like credential stuffing or password leaks, in addition to complying with regulatory requirements, such as the LGPD. There's also Two-Factor Authentication (2FA), which is a subset of MFA that specifically uses two factors from different categories. If more factors are added (e.g., password + token + biometrics), the system is still classified as MFA [6], [16].

MFA is widely applied in sectors that demand high security. In the financial sector, it protects bank accounts and card transactions, ensuring that only the authorized account holder has access to funds or performs confidential transactions. In healthcare, multi-factor authentication is essential for restricting access to sensitive medical records, guaranteeing patient privacy [16].

However, even with the adoption of robustness measures like Multi-Factor Authentication, authentication mechanisms remain a primary target for adversaries. One of the most prevalent methods to compromise these systems is the Brute Force Attack, which involves an automated trial-and-error approach to discover valid credentials [6]. These attacks manifest differently depending on the target layer, a distinction crucial for detection strategies. At the infrastructure level, attacks target network protocols such as Secure Shell (SSH) or File Transfer Protocol (FTP), interacting directly with server ports [9]. Conversely, at the application layer, adversaries target Web login forms via HTTP requests, simulating user interactions at superhuman speeds to guess passwords [10].

Beyond the initial entry point, a distinct category of threats becomes apparent during the Post-Authentication phase. In this specific scenario, the adversary has already managed to bypass authentication mechanisms. This unauthorized access is often achieved through various means, such as successfully guessing credentials, stealing them via phishing campaigns, or even originating from a malicious insider. Consequently, the primary security concern shifts to Lateral Movement [17]. This technique describes how a compromised account is utilized to traverse the network, accessing

multiple machines or servers in search of sensitive data. The detection of such threats presents a significant challenge, primarily because the credentials being used are technically valid [12], [15]. Therefore, the indicator of compromise is not an incorrect password, but rather an anomaly in the behavioral patterns of the user-computer interaction, such as accessing critical systems outside of standard working hours or connecting to unauthorized servers.

To address these post-authentication threats, modern security paradigms are shifting towards a Zero Trust architecture, which operates on the principle of 'never trust, always verify'. In this context, continuous monitoring of user behavior becomes mandatory [18]. However, the sheer volume of authentication logs generated by these systems makes manual analysis impossible, while traditional rule-based detection systems often struggle to distinguish between complex attack patterns and legitimate anomalies [11]. This limitation highlights the critical need for data-driven approaches capable of automating detection while adapting to new threats with minimal human intervention [19].

2.1.2 Machine Learning

Although these traditional mechanisms are effective for known threats, machine learning in user authentication offers several benefits, including enhanced security through robust identification methods, adaptive access control based on user behavior, and faster detection of suspicious activities. It can improve the user experience by streamlining access processes while continuously learning and adapting to evolving cyber threats, predicting future events in unseen data [12], [20].

Machine learning systems can be divided into different applications, such as supervised learning, where the model is trained using a labeled data set, which means that each input example has an associated correct output (label). The goal is for the algorithm to learn to map inputs to desired outputs, generalizing to new data. Unsupervised learning, on the other hand, operates on unlabeled data, dedicating itself to pattern identification through techniques like clustering, density estimation, or dimensionality reduction for visualization. Complementarily, reinforcement learning — inspired by behavioral theories — focuses on the sequential optimization of actions in dynamic environments, where the system learns through reward mechanisms to maximize its performance in complex tasks [21], [22].

The effective application of Machine Learning in cybersecurity necessitates a comprehensive understanding of various models capable of addressing both classification and anomaly detection tasks. While the broader paradigm encompasses supervised, unsupervised, and reinforcement learning, establishing a performance baseline is essential for evaluating specialized methodologies and understanding system limitations. Therefore, a detailed analysis of key classification algorithms, such as Logistic Regression and Decision Trees, and unsupervised methods, like Isolation Forest, is presented

here. This comparative foundation is essential for establishing a robust performance benchmark and enabling a thorough analysis of existing methodologies [12], [9], [19].

2.1.2.1 Logistic Regression

Among the key classification algorithms used as baselines, the Logistic Regression (LR) model is frequently chosen due to its computational efficiency, simplicity, and high interpretability, particularly in binary classification tasks (e.g., classifying behavior as normal or attack). Despite its name, LR is a probabilistic classification algorithm that utilizes the Sigmoid Function to map any real-valued input into a probability output ranging strictly between 0 and 1.

Mathematically, the model estimates the probability $P(Y = 1|X)$ by fitting data to a logistic curve. This linear decision boundary makes it highly effective for identifying volumetric attacks where the separation between normal and malicious traffic is relatively distinct. Furthermore, its probabilistic nature allows for precise calibration of detection thresholds. For instance, in a high-security environment, the threshold can be lowered to catch more potential threats, intentionally increasing the system's sensitivity to capture more potential threats, even if it means occasionally flagging legitimate traffic as suspicious. This tuneability allows the model to be adapted to different risk tolerances, establishing a flexible security baseline [21]."

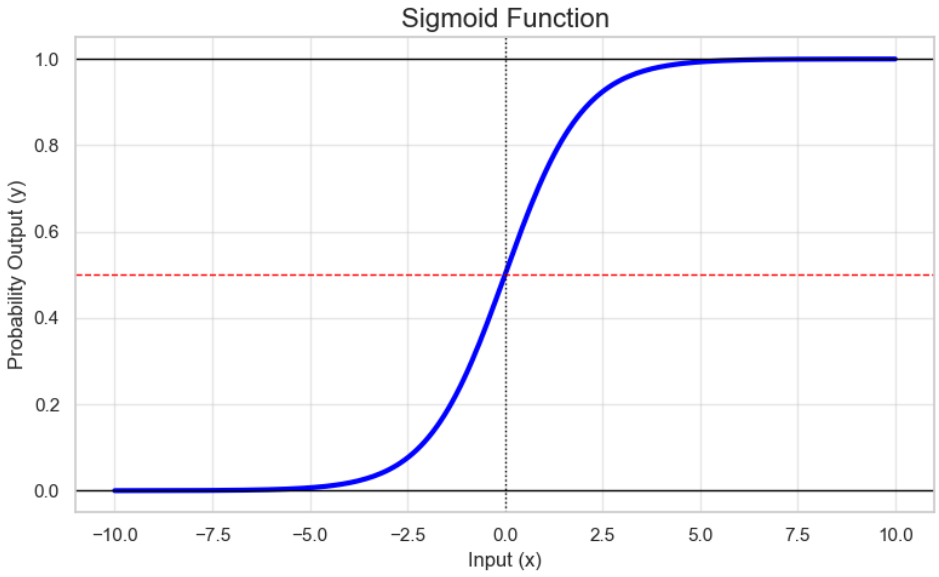

Figure 1: Representation of the Sigmoid Function. Source: The Author, based on Goodfellow et al. [1].

2.1.2.2 Decision Tree

The Decision Tree (DT) algorithm provides a highly interpretable model that simulates human decision-making based on sequential rules. Structurally, the model starts

from a root node and splits the dataset into subsets based on the value of input features (e.g., ”Is login hour $>$ 18:00?”). This process creates internal nodes and ultimately leads to leaf nodes, which represent the final classification outcome (e.g., ’Normal’ or ’Attack’).

The fundamental principle guiding these splits is the maximization of Information Gain or the minimization of Impurity. At each step, the algorithm selects the feature that best separates the classes, aiming to create leaf nodes that are as homogeneous (pure) as possible. In the context of intrusion detection, DTs are invaluable for their ability to generate explicit ”white-box” rules (e.g., ”If source IP is blacklisted AND login frequency $>$ 10/min, then Attack”). This transparency allows security analysts to audit the model’s logic and understand exactly why a specific event was flagged [21].

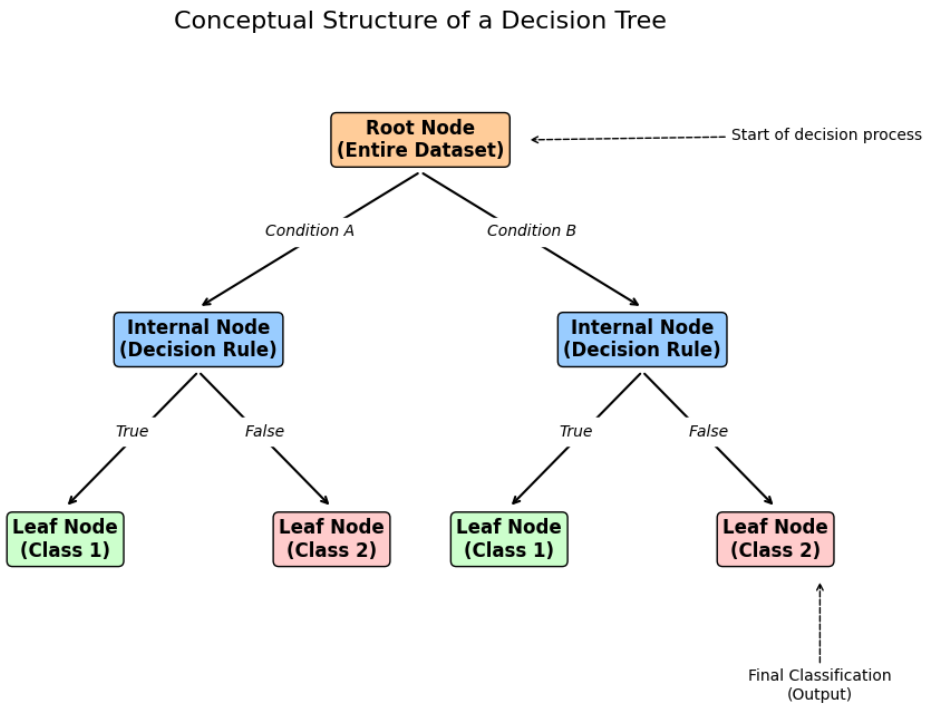

Figure 2: Conceptual structure of a Decision Tree algorithm. Source: The Author, based on Breiman et al. [2].

2.1.2.3 Random Forest

Building upon the principles of Decision Trees, the Random Forest (RF) algorithm operates as an ensemble learning method that aggregates the results from a multitude of individual trees. In cybersecurity, RF is highly valued for its ability to significantly improve accuracy and stability compared to a single Decision Tree, largely because it

mitigates the common problem of overfitting to the training data.

The RF model achieves this robustness through two key randomization mechanisms:

1. **Bagging (Bootstrap Aggregating):** Each individual tree is trained on a random subset of the data sampled with replacement;
2. **Feature Randomness:** At each node split, the algorithm considers only a random subset of features rather than all available features.

The final classification is determined by a majority vote among all trees. This ”wisdom of crowds” approach makes Random Forest exceptionally effective for high-dimensional data and provides a native measure of Feature Importance, allowing researchers to identify which attributes are most predictive of an attack [23].

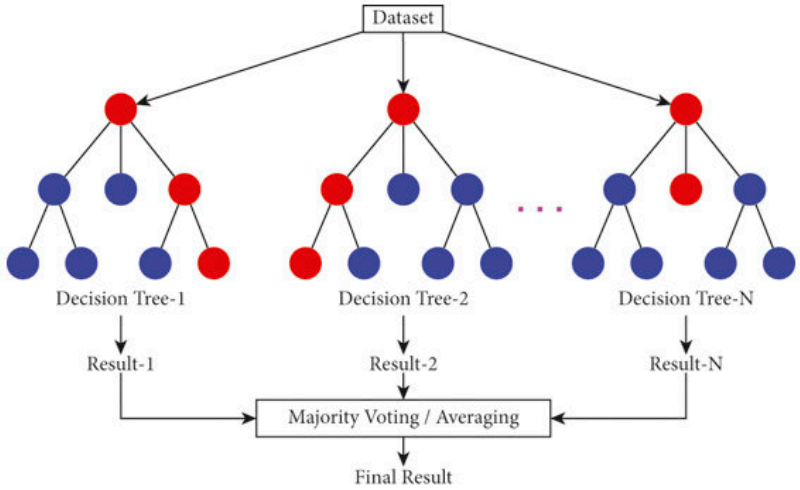

Figure 3: Conceptual diagram of the Random Forest algorithm, illustrating ensemble learning through the aggregation of multiple Decision Trees [3].

2.1.2.4 Isolation Forest

While the previous models are supervised, Isolation Forest (iForest) is an unsupervised algorithm explicitly designed for anomaly detection. Unlike traditional distance-based methods (like K-Means) that attempt to profile normal data points, iForest relies on the principle that anomalies are ”few and different”.

The algorithm builds an ensemble of random trees known as Isolation Trees. Because anomalies are statistically distinct from normal data, they are easier to ”isolate” by random partitioning. Consequently, anomalous points tend to terminate much closer to the root of the tree (requiring fewer splits to be separated), while normal points require more splits and end up deeper in the tree. The algorithm assigns an anomaly score based on the average Path Length: shorter average paths indicate a higher likelihood

of being an anomaly. This logarithmic efficiency ($\mathcal{O}(n \log n)$) allows iForest to scale to massive datasets, making it ideal for filtering millions of logs to find rare insider threats without requiring prior knowledge of attack signatures [24].

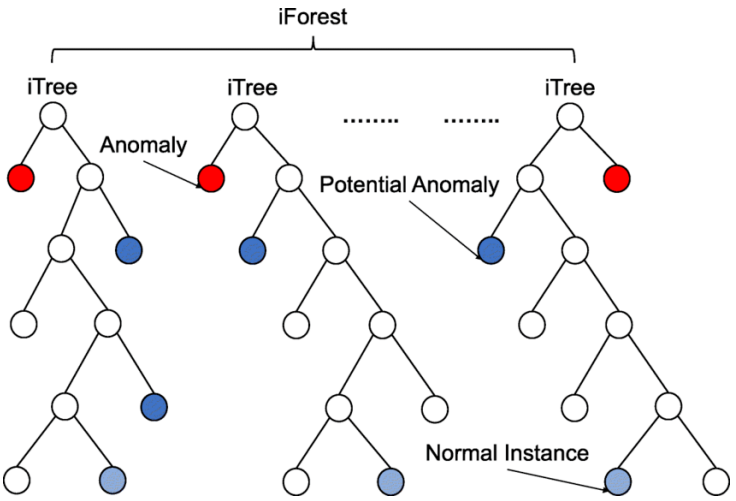

Figure 4: Conceptual structure of the iForest algorithm, illustrating how anomalies are isolated closer to the root of the Isolation Trees (iTrees) with fewer splits [4].

2.1.2.5 Active Learning

However, traditional intrusion detection systems face a critical limitation: the inability to actively learn from new attack patterns. This deficiency stems from an over-reliance on predefined rules and signatures, which restrict the identification of dynamic and uncataloged threats. As a result, novel or adaptive attack techniques often go unnoticed, exposing networks to significant vulnerabilities. The lack of real-time adaptation mechanisms reduces the effectiveness of these systems, highlighting the urgent need for more flexible and proactive approaches to anomaly detection [12].

Among the approaches presented earlier, there is active learning, which combines supervised and unsupervised learning techniques. The central idea behind active learning is that if a machine learning algorithm can select the data it uses for learning, it can achieve greater accuracy with fewer training labels. Active learners issue queries, typically in the form of unlabeled data instances, which are then labeled by an oracle (e.g., a human annotator). Active learning is useful for many modern machine learning problems because, while unlabeled data is abundant or easy to obtain, obtaining labels is difficult, time-consuming, or expensive [25].

Regarding implementation, three approaches stand out in active learning [14]:

- **Pool-Based Sampling**: The algorithm analyzes the entire available unlabeled dataset (the pool) and selects the most informative samples to be labeled. However, a challenge with this approach is the potentially high memory consumption it can require;

- **Stream-Based Selective Sampling**: In this approach, unlabeled data points are evaluated individually as they arrive, and the algorithm itself decides, in real time, whether or not to request their label. However, a disadvantage arises from the lack of guarantee that the data scientist will adhere to the planned budget;
- **Membership Query Synthesis**: It allows the model to generate personalized queries by creating synthetic examples or combining features from existing data to maximize learning. However, its application is limited, as it is not suitable for all situations or problem types.

2.1.2.6 Performance and Prediction Parameters

After training and selecting the most suitable model for a given task, it is essential to evaluate its performance and predictive capability. The confusion matrix is one of the primary methods for formally evaluating machine learning models. Also known as an error matrix, it's a table that details the performance of a classification or prediction model [26]. Thus, the confusion matrix becomes an essential tool for evaluating binary classification models—especially in cybersecurity contexts—by categorizing results into four different groups: True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN) [11], as illustrated in Figure 5.

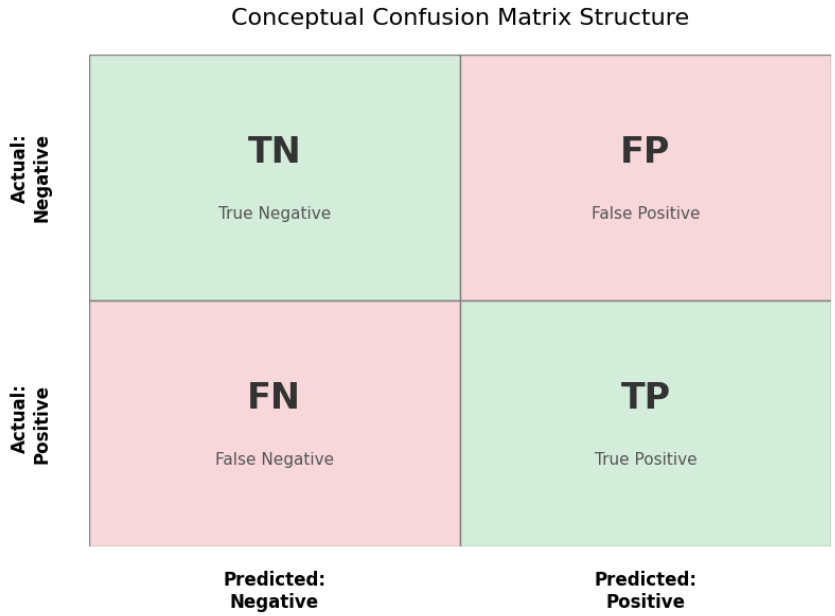

Figure 5: Conceptual structure of the Confusion Matrix. Source: The Author, based on Fawcett [5].

The TP represent legitimate behaviors, while TN reflect the accuracy in detecting real threats. Among the classification errors, FP generate incorrect alarms, and FN occur when the system considers normal traffic as a threat. These presented elements

allow for various performance metrics to be calculated, going beyond a simple error rate.

From these elements, the following performance metrics are calculated [11]:

- **Precision (Positive Predictive Values)**: This is the proportion of true positive classifications relative to the total number of instances classified as positive, as expressed in Eq. 1.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{1}$$

Precision serves as the primary indicator of operational efficiency in Security Operations Centers (SOCs). A low precision score implies a high rate of FPs, which significantly contributes to ”alert fatigue.” This phenomenon occurs when security analysts become desensitized to warnings due to the overwhelming volume of noise, potentially causing real threats to be ignored [11]. Therefore, achieving high precision is essential to ensure that the proposed detection system remains usable in real-world environments, minimizing the effort wasted on investigating benign events.

- **Recall (Sensitivity or True Positive Rate)**: The Equation 2 represents the model's ability to correctly detect real threats, calculating the ratio between true positives and the total number of existing positive instances.

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2}$$

In the specific context of cybersecurity research, this metric is frequently prioritized over others due to the high stakes involved in intrusion detection [11, 26]. This preference arises from the asymmetry of risk, where a FNs implies a successful intrusion that bypassed the defense, potentially leading to catastrophic data breaches or system compromises. Therefore, maximizing Recall is essential to ensure a robust security posture, guaranteeing that the system minimizes the probability of leaving an attack undetected.

- **F1 Score**: In Equation 3, a metric is established that combines both precision and recall metrics, serving as a classification of the model's accuracy, harmonizing both metrics.

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

The F1 Score acts as a harmonic balance between Precision and Recall, providing a single, reliable indicator of the model's quality. In intrusion detection scenarios, legitimate traffic dominates the dataset, which can allow a model to achieve high accuracy simply by ignoring rare attack events. The F1-Score prevents this by penalizing extreme imbalances. It ensures that a high score is only awarded

if the model is both effective at detecting threats (Recall) and trustworthy in its alerts (Precision), offering a more honest evaluation of performance in real-world conditions [11].

2.2 RELATED WORKS

To situate the present study within the current academic landscape, a critical review of the state-of-the-art was conducted. This section is organized into two complementary analyses. First, an overview of the benchmark datasets frequently cited in the cybersecurity authentication literature is presented [11], [12], [13], establishing the standards typically used for experimental validation. Subsequently, specific related works are analyzed, highlighting their methodologies, main contributions, and reported limitations, thereby identifying the research gaps that this work aims to address.

2.2.1 Benchmark Datasets in the Literature

To validate intrusion detection approaches based on active learning, it's fundamental to use datasets that reflect the complexity and dynamism of real threats. These datasets must meet specific criteria such as a variety of attacks, encompassing both known threats and anomalous patterns; an adequate balance between normal and malicious events; and rich metadata that includes temporal information, traffic origin, and authentication context. Table 1 compares the main datasets used in the literature, highlighting their suitability for active learning in cybersecurity scenarios for user authentication.

Table 1: Dataset Descriptions

Dataset	Year	Data Type	Attacks Covered
TON_IoT [27]	2020	IoT and Logs	DDoS, Injection, Brute Force
CIC-IDS2017 [28]	2018	Network Traffic	SSH-Patator, FTP-Patator, Web Brute Force, DDoS
LANL Auth Logs [29]	2016	Authentication Logs	Brute Force, Unauthorized Access, Lateral Movement
UNSW-NB15 [30]	2015	Network Traffic	Intrusion Detection, Exploits, Fuzzers
Kddcup99 [31]	2009	Network Traffic	Intrusion Detection, DoS

Following the overview presented in Table 1, a detailed analysis of each dataset's characteristics and limitations is essential to understand the landscape of available data for intrusion detection research.

In the specific domain of the Internet of Things, the TON_IoT [27] dataset stands out by aggregating telemetry data from sensors with network logs. It collects data from seven IoT devices and includes columns specific to IoT protocols, such as MQTT and

Modbus, alongside standard network flow features. Although highly relevant for IoT security, its specific context relies heavily on sensor data and machine-to-machine protocols, which diverges significantly from the enterprise authentication environment typically targeted in general IT security research.

Representing a significant evolution in network security, the CIC-IDS2017 [28] provides captures of complete network flows structured over a five-day period, designed to reflect realistic background traffic mixed with diverse attack scenarios. The dataset is organized chronologically to simulate a typical work week, starting with Monday, which is dedicated exclusively to Benign traffic, establishing a robust baseline of normal user behavior and background network noise without any malicious activity. Subsequently, Tuesday introduces Brute Force attacks targeting infrastructure protocols, specifically SSH and FTP services, while the period from Wednesday to Friday covers a broad spectrum of other threats, ranging from Denial of Service (DoS/DDoS) and Heartbleed to Web Application attacks (SQL Injection, XSS) and Botnets.

Unique among the reviewed sources, the LANL Auth Logs [29] focus exclusively on authentication events rather than packet-level traffic. Its structure is simplified but highly specific, containing columns such as Time, Source User, and Destination Computer. This structure is ideal for analyzing user behavior and detecting post-authentication threats such as lateral movement, addressing a critical gap that traditional network-flow datasets often fail to capture.

Slightly preceding these, the UNSW-NB15 [30] was developed to include a hybrid of real modern normal activities and synthetic attack behaviors. This dataset is notable for its comprehensive feature set of 49 attributes, categorized into flow, basic, content, time, and generated features. While it covers nine families of attacks, its focus remains broad, covering general network exploits rather than specialized authentication anomalies. This characteristic limits its direct applicability to specific problems like credential-based attacks, which require a more granular view of authentication flows.

Finally, historically serving as the standard, the KDD Cup 99 [31] dataset is derived from the DARPA98 dataset [32]. It contains approximately 4.9 million connection records described by 41 features. However, despite its foundational role, it is now considered outdated due to the lack of modern traffic patterns and the presence of significant redundancy, which can bias machine learning models towards unrealistic results. Consequently, its inclusion in modern studies serves primarily as a historical reference.

This review highlights the diversity of available datasets, each tailored to specific security domains ranging from modern IoT environments to legacy networks and behavioral analysis.

2.2.2 State-of-the-Art Analysis

This state-of-the-art analysis examines contemporary research in machine learning

applications for cybersecurity authentication. The selected studies span various approaches, ranging from conventional supervised learning to more advanced interactive methods, highlighting both their innovative contributions and methodological limitations. By synthesizing these findings, this work establishes the research landscape that informs the investigation into Active Learning for intrusion detection, while simultaneously validating the research opportunity identified in the literature.

To provide a structured overview of the literature, Table 2 presents a consolidated summary of the related works, highlighting their main contributions, limitations, and relevance to the present study. Following this overview, a detailed discussion of each study is provided.

Table 2: Summary of Related Works and Connection to Present Study

Author	Year	Focus Area	Main Contribution	Dataset(s) Used	Reported Limitations
Tadi [33]	2023	Behavioral Authentication in APIs	Combined behavioral authentication with ML to create adaptive systems	Synthetic API Logs	Absence of experimental validation in real-world environments.
Ramakrishnan [18]	2023	ML in Access Control (IAM/RBAC)	Achieved high accuracy (95%) and low latency with conventional ML	Kaggle (Amazon Access)	Prolonged training time of the SVM model alone.
Albert-Weiss et al. [34]	2022	Active Learning (KDPP) in Agriculture	Optimized learning with limited labeled data using KDPP	Proprietary (Melon Samples)	High computational cost and performance challenges against random sampling.
Wiefling et al. [35]	2022	Risk-Based Authentication (RBA) in SSO	Developed an ML-based method to optimize RBA parameters using real data	Real-world SSO Service Logs	Limited exploration of sophisticated models (e.g., Siamese, OCSVM) for risk scoring.
Dang [19]	2020	Active Learning in Intrusion Detection (IDS)	Demonstrated strategic selection of rare samples using Active Learning with Naive Bayes	CIDS'12	Restricted to the study of a single algorithm (no comparative analysis).

In the specific context of cybersecurity, Tadi [33] addressed the security challenges in APIs during accelerated digital expansions, utilizing a synthetic dataset of API logs to model user behavior. The research distinguished itself by combining behavioral authentication with Machine Learning techniques, demonstrating the potential to create adaptive systems. This combination is a significant contribution, reinforcing the concept that dynamic, ML-based controls are necessary to secure modern user access. However, a crucial limitation reported was the absence of experimental validation in real-world environments, which represents a significant gap in the presented results.

This limitation directly supports the necessity of the present work's methodology, which includes experimental validation and quantification using performance metrics, ensuring the practical effectiveness of the proposed Active Learning approach.

In parallel, Ramakrishnan [18] explored innovative AI applications in access control, specifically within RBAC and IAM systems, employing the Amazon Access Challenge dataset from Kaggle. By utilizing algorithms like Random Forest and SVM on this data, the study achieved an impressive 95% accuracy with a response time of 0.15 seconds, standing out for its simultaneous processing of complex policies. The work contributes by demonstrating the high accuracy and low latency achievable by conventional ML models in identity systems, reinforcing the viability of ML in critical security environments. However, a key reported limitation was the prolonged training time of the SVM alone. This limitation is central to our research, as the high training cost associated with traditional supervised models like SVM is precisely the bottleneck that Active Learning aims to resolve in the context of real-time threat detection.

Looking at broader applications outside security, Albert-Weiss et al. [34] investigated the application of non-destructive techniques to evaluate the quality of 'Galia' melons, utilizing a proprietary dataset of 30 fresh samples analyzed via spectral sensors. The study proposed the integration of K-Determinantal Point Processes (KDPP) with Active Learning. This constitutes their main contribution: optimizing learning in scenarios with limited labeled data by strategically selecting samples. As limitations, the authors highlighted the high computational cost and the challenges in outperforming random sampling due to the dynamic nature of agricultural parameters. While the application domain (agriculture) is different, their focus on Active Learning in situations with scarce labeled data is similar to the challenges faced in real-time threat detection in cybersecurity. Furthermore, their finding regarding the high computational cost is a critical point that the present work aims to address by focusing on optimizing the efficiency of Active Learning methods in authentication systems.

Returning to dynamic authentication, Wiefling et al. [35] conducted a pioneering study on Risk-Based Authentication (RBA) in a Single Sign-On (SSO) service, analyzing real-world login logs from a large-scale production environment. Their main contribution was developing a machine learning-based method for optimizing RBA parameters using this real data, representing the first large-scale analysis of RBA in a real environment. This focus on using ML to dynamically adjust authentication controls directly supports the feasibility of the Active Learning approach. However, a key limitation reported was the limited exploration of more sophisticated models, such as Siamese neural networks or One-Class Support Vector Machine (OCSVM), for calculating risk scores. This limitation reinforces the need for our work to compare different and modern Active Learning techniques that may offer superior performance in dynamic authentication environments.

Finally, regarding intrusion detection systems, Dang [19] innovated by proposing an Active Learning approach with Naive Bayes, tested on the CIDS'12 benchmark dataset. The method's main contribution was its effectiveness in the strategic selection of rare samples, challenging traditional density-based approaches. This strategic selection is a key aspect that aligns with the goal of detecting anomalous, low-frequency attacks in user authentication. The research, however, was limited to the study of a single algorithm (Naive Bayes), leaving room for comparative investigations with other techniques. This gap directly informs the methodology of the present work, which proposes a comparative evaluation between different Active Learning approaches and conventional models to identify the most efficient solution for authentication systems.


3

3 METHODOLOGY

The present research is classified as applied, as it seeks to address practical challenges related to security and efficiency in intrusion detection systems within authentication contexts. Regarding its approach, the study is strictly quantitative and is based on the statistical analysis of large volumes of network traffic data and behavioral biometrics. From a procedural point of view, the method is defined as experimental. It involves the controlled manipulation of variables, particularly the size of training sets and query strategies, to systematically compare the performance of Supervised, Unsupervised, and Active Learning paradigms.

3.1 GENERAL ARCHITECTURE

To ensure the robustness and reproducibility of the proposed experiments, a specific computational environment was established. The experiments were conducted using specific computational resources hosted on an Ubuntu Linux (version 24.04.1) operating system. The primary language used was Python (version 3.12.3) [36], executed in Jupyter Notebook environments [37].

The software stack was organized into three functional layers. First, data manipulation, cleaning, and structuring relied on the Pandas (v2.3.3) [38] and NumPy (v2.3.3) [39] libraries, while Matplotlib (v3.10.7) [40] and Seaborn (v0.13.2) [41] were utilized for data visualization and the generation of performance graphs. Subsequently, the Scikit-learn (v1.7.2) library [42] was utilized as the core engine for machine learning implementation, providing the necessary infrastructure for data preprocessing, dimensionality reduction, and the training of both supervised and unsupervised models. Finally, the modAL framework (v0.4.2.1) [43] was integrated to enable the active learning experiments, managing the query strategies and the oracle simulation.

With the computational framework established, the data selection focused on covering distinct layers of authentication attacks. The study's universe comprises authentication events and network traffic logs. For the specific experimental scope of this research, two distinct, publicly validated benchmark datasets were selected. These were chosen specifically because they provide the necessary granularity to analyze authentication anomalies at different layers (Network, Application, and Post-Authentication), ensuring a comprehensive evaluation of the proposed detection models.

Table 3 details the specific datasets utilized in this work, highlighting the attack vectors extracted and their specific application within the study.

Table 3: Benchmark Datasets Utilized in the Experiments

Dataset	Year	Selected Attack Vectors	Key Features	Application in this Study
CIC-IDS2017 [28]	2017	SSH-Patator, FTP-Patator, Web Brute Force	Network Flow Metrics (e.g., Flow Duration, Packet Counts, TCP Flags)	Train and compare Supervised vs. Active Learning models for external brute force.
LANL [29]	2014	Lateral Movement, Insider Threat	Authentication Logs (Time, Source User, Destination Computer)	Evaluate Unsupervised Detection of post-authentication anomalies and AL refinement.

Following this selection, the datasets were processed to isolate specific scenarios:

1. **CIC-IDS2017 (Intrusion Detection Evaluation Dataset):** This dataset was the primary source for analyzing Brute Force attacks [28]. Two specific subsets were utilized to evaluate the model's robustness across different attack vectors:
    - **Infrastructure Layer:** Using the Tuesday capture, which contains SSH and FTP brute force attacks against network protocols;
    - **Application Layer:** Using the Thursday morning capture, which contains Web-based brute force attacks against HTTP authentication forms.
2. **LANL (Los Alamos National Laboratory):** Employed for the detection of post-authentication anomalies [29]. This dataset provided millions of user-computer authentication events, allowing for the analysis of lateral movement patterns [17] and insider threats where the attacker already possesses valid credentials.

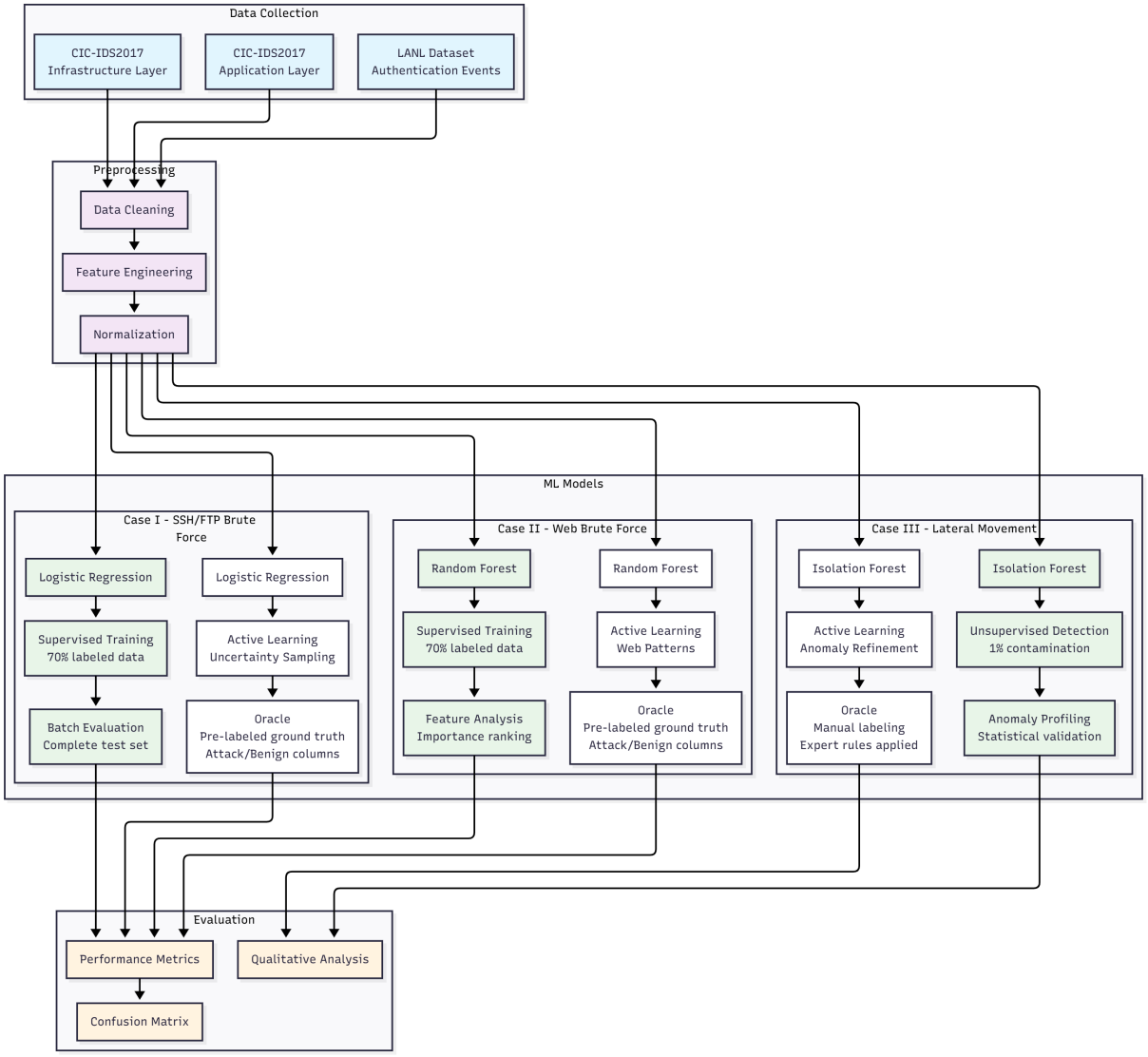

Figure 6: General architecture of the experimental framework, illustrating the parallel evaluation of traditional supervised learning and active learning approaches across three distinct cybersecurity scenarios.

As illustrated in Figure 6, the experimental workflow was guided by a unified analytical framework, systematically applied across all datasets to ensure methodological consistency and fair comparison. This structured approach unfolded through four interconnected phases: Data Collection, Data Preprocessing, Scenario-Specific Modeling, and Model Evaluation. By standardizing these stages, the methodology allows for a direct assessment of the efficiency gains provided by Active Learning in contrast to traditional supervised methods. The specific implementation details of this pipeline for each attack scenario are elaborated in the following subsections.

3.2 CASE STUDY I: INFRASTRUCTURE LAYER ATTACKS (SSH/FTP

BRUTE FORCE)

This study focused on the detection of high-volume authentication attacks targeting network protocols, specifically utilizing the Tuesday capture from the CIC-IDS2017 dataset as the primary data source.

To prepare the data for binary classification, a rigorous preprocessing phase was executed. The target variable was standardized by mapping instances originally labeled as FTP-Patator and SSH-Patator to the positive class (Attack), while all traffic labeled as BENIGN was assigned to the negative class (Normal). Furthermore, to ensure numerical stability for the machine learning algorithms, the dataset underwent a sanitization process to handle artifacts generated by traffic capture tools, where infinite values and missing entries were imputed with zero.

The Logistic Regression classifier was selected as the primary algorithm for this task. This choice is justified by its computational efficiency and interpretability, making it an ideal baseline for detecting volumetric attacks where the separation between classes is expected to be largely linear [21]. Furthermore, it is widely recognized in cybersecurity literature as a standard benchmark for intrusion detection systems, providing a necessary reference point to evaluate the gains of more complex models [11]. Unlike ”black-box” models, Logistic Regression provides immediate insight into the relationship between features and attack probability. Finally, its extremely low computational cost during the inference phase makes it highly suitable for real-time authentication monitoring, as highlighted in recent reviews on machine learning for user authorization [12].

The experimental protocol comprised two distinct scenarios utilizing this classifier:

- **Supervised Baseline (Upper Bound):** In this scenario, the model was trained using 70% of the preprocessed dataset. This approach represents the ideal condition of abundant labeled data, serving to establish a performance benchmark regarding maximum achievable Recall and Precision;

- **Active Learning Simulation:** To simulate a resource-constrained environment, the model was initialized with a minimal seed of 20 balanced samples (10 benign and 10 attacks). An Uncertainty Sampling strategy was then employed to iteratively query the oracle for the labels of the most ambiguous instances. The simulation was conducted for 100 query iterations to test the hypothesis that high detection rates can be achieved with minimal human labeling effort.

3.3 CASE STUDY II: APPLICATION LAYER ATTACKS (WEB BRUTE FORCE)

The application-layer threat detection employs Random Forest to handle complex,

non-linear attack patterns. This supervised baseline evaluates the CIC-IDS2017 Thursday dataset for identifying multifaceted web attacks such as Brute Force, XSS, and SQL Injection camouflaged in normal web traffic.

To facilitate binary classification, a rigorous data preprocessing phase was executed to consolidate the specific attack labels. The primary threat analyzed was the Web Attack – Brute Force, involving automated attempts to guess passwords via HTTP POST requests, which was mapped to the positive class (Attack). To create a comprehensive threat profile, instances of XSS and SQL Injection were also included in this class. Conversely, normal web browsing traffic was mapped to the negative class (Benign).

Crucially, to prevent the model from learning trivial artifacts or ”memorizing” specific network entities, a strict feature selection process was applied. Non-numeric identifiers such as `Flow ID`, `Source IP`, `Destination IP`, and `Timestamp` were explicitly removed. This ensures that the classification is based solely on the behavioral metrics of the traffic (e.g., flow duration, packet size) rather than on specific IP addresses or the time of day the attack occurred, mitigating the risk of temporal bias.

Regarding the modeling strategy, strict measures were taken to avoid data leakage. The dataset was split into training (70%) and testing (30%) sets using stratified sampling before any normalization was applied. The StandardScaler was fitted exclusively on the training partition and then applied to the test partition. This guarantees that the model remained completely blind to the statistical distribution of the test set during the training phase.

Given the nature of the data, the Random Forest (RF) classifier was selected as the core algorithm. This choice is justified by its proven capability to handle complex and heterogeneous datasets common in intrusion detection scenarios [44]. Recent studies demonstrate that this algorithm offers high accuracy, robustness against overfitting, and excellent generalization capacity, particularly in cybersecurity environments targeting web vulnerabilities [45]. Additionally, Random Forest provides native interpretability through *Feature Importance* analysis, allowing for a deeper understanding of the factors characterizing an attack [46].

Finally, the validation protocol was replicated to ensure methodological consistency. The supervised baseline established the performance ceiling, while the Active Learning simulation—utilizing a minimal seed of 20 samples and 100 query iterations—verified if efficiency gains persist even when applying a non-linear model to the nuanced patterns of web traffic.

3.4 CASE STUDY III: POST-AUTHENTICATION ANOMALIES (LATERAL MOVEMENT)

This study focused on the detection of Lateral Movement and Insider Threats using the Los Alamos National Laboratory (LANL) dataset. Unlike the previous scenarios,

this dataset represents a significant challenge as it is strictly unsupervised, containing millions of user-computer authentication events without any explicit indication of malicious activity. This characteristic reflects the real-world reality of most SOCs, where the vast majority of available data remains unclassified.

Given the raw nature of the logs, extensive feature engineering was performed to capture behavioral context rather than simple connection statistics. The process involved the derivation of novelty metrics, specifically a boolean flag indicating whether a user was accessing a specific computer for the first time in the recorded history, which serves as a strong indicator of lateral movement. Additionally, frequency metrics were calculated to identify bursts of activity by counting the number of logins per user and per computer within fixed time windows. To address the highly skewed nature of network traffic, where legitimate automated processes can generate thousands of events, a logarithmic transformation ($\log(1 + x)$) was applied to all frequency features. This normalization was crucial to prevent the anomaly detection model from being biased towards high-volume but benign machine behavior [47].

Due to the absence of ground truth labels, a supervised approach was not initially feasible, leading to the implementation of a two-stage hybrid architecture. The first stage employed the Isolation Forest algorithm [48] for initial anomaly detection. Unlike distance-based methods that require heavy computation, Isolation Forest efficiently isolates anomalies by randomly partitioning the feature space. This characteristic makes it particularly suitable for high-dimensional datasets with large volumes of data, where anomalies are defined as few and distinct instances. Consequently, the model was trained on the entire dataset to assign anomaly scores to millions of events, effectively filtering the data to isolate the top 1% most suspicious outliers.

Subsequently, an Active Learning module was applied to the filtered anomalies to mitigate the false positive rate inherent in unsupervised methods, where unusual but legitimate behavior is often flagged. In this refinement phase, a human analyst acted as an oracle to label a small subset of 30 samples based on their behavioral context. This allowed the model to learn the distinction between true lateral movement attacks, such as first access to critical servers at unusual hours, and legitimate administrative tasks, enabling the automatic classification of the remaining alerts.

3.5 PERFORMANCE EVALUATION AND METRICS

Given the diverse nature of the experimental scenarios, which encompass both supervised and unsupervised learning paradigms, a comprehensive evaluation framework was adopted. This approach was designed to assess not only the predictive power of the models but also their operational efficiency, addressing the dual goal of maintaining high detection capability while minimizing human effort and computational resources.

For the experiments involving the CIC-IDS2017 dataset, where ground truth labels were available, the evaluation centered on the classifier's ability to distinguish benign traffic from specific attack vectors. In this context, performance was quantified using standard metrics derived from the confusion matrix [26]. Specifically, Recall (or sensitivity) was prioritized as the primary measure of success. In cybersecurity, the cost of missing an attack is often higher than that of a false alarm; therefore, maximizing the detection rate of actual threats is paramount to minimizing security breaches. However, relying solely on recall can be misleading if the model generates excessive false alarms. To address this, Precision and the F1-Score were also evaluated to ensure a balanced assessment, guaranteeing that the detection system remains operationally viable without overwhelming analysts with noise.

Beyond traditional predictive metrics, the study introduced a comparative analysis of efficiency. The labeling cost was evaluated by contrasting the massive volume of data required to train the baseline supervised models against the minimal query budget used in the Active Learning approach. This comparison serves to directly validate the hypothesis that intelligent sampling strategies can drastically reduce the human effort involved in dataset annotation.

In the case of the unsupervised LANL dataset, where prior labels were absent, traditional metrics such as accuracy were not applicable during the initial detection phase. Consequently, the evaluation strategy shifted towards a qualitative validation of the anomalies identified by the Isolation Forest algorithm. This process involved analyzing the statistical divergence between events classified as normal and those flagged as anomalous, verifying whether the model successfully isolated distinct behavioral patterns consistent with lateral movement, such as login frequency and access times. Furthermore, the efficacy of the Active Learning refinement was assessed by its ability to filter FPs, effectively measuring the reduction in alert volume that a human analyst would need to investigate, thereby quantifying the operational gain of the proposed hybrid architecture.


4

4 RESULTS AND DISCUSSION

This chapter presents the experimental results obtained from the three case studies. The analysis focuses on quantifying the trade-off between detection performance and labeling effort, comparing the proposed Active Learning approach against traditional Supervised baselines. Furthermore, the computational viability of the models for real-time application is assessed. Finally, a critical discussion interprets these findings in light of the research objectives and existing literature.

4.1 CASE STUDY I: INFRASTRUCTURE LAYER (SSH/FTP BRUTE FORCE)

In this scenario, the detection of volumetric attacks against network protocols was evaluated using Logistic Regression.

4.1.1 Supervised Baseline Performance

The initial analysis focused on the supervised detection of volumetric attacks (SSH and FTP Brute Force) using the Tuesday capture from the CIC-IDS2017 dataset. The combined dataset comprised a total of 975,827 network flows, exhibiting a severe class imbalance typical of real-world environments: legitimate traffic (`BENIGN`) represented 98.6% of the data, while attack instances (`FTP-Patator`, `SSH-Patator`) accounted for only 1.4% (13,835 flows).

Regarding the data preprocessing outcomes, the cleaning phase successfully identified and mitigated numerical instability in specific features. Notably, `Flow Bytes/s` and `Flow Packets/s` contained 701 instances of null (`NaN`) or infinite values. These artifacts, resulting from flows with zero duration, were sanitized by imputation, ensuring the stability of the mathematical model for the training phase.

Subsequently, the baseline performance was established using the Logistic Regression model. Trained on 70% of the data ($N_{train} \approx 683,000$) with class weighting enabled (`class_weight='balanced'`), the model produced the results detailed in Table 4.

Table 4: Performance Metrics for Logistic Regression (Infrastructure Layer)

Class	Precision	Recall	F1-Score
Benign (0)	1.00	0.98	0.99
Attack (1)	0.47	0.99	0.64

The performance analysis reveals a clear strategic trade-off prioritized by the Logistic Regression baseline. Primarily, the model achieved Maximized Safety, evidenced by a near-perfect Recall of approximately 99.8% for the Attack class.

To visually corroborate these findings, Figure 7 presents the Confusion Matrix generated from the test set. The graphical representation explicitly validates the results, as detailed below.

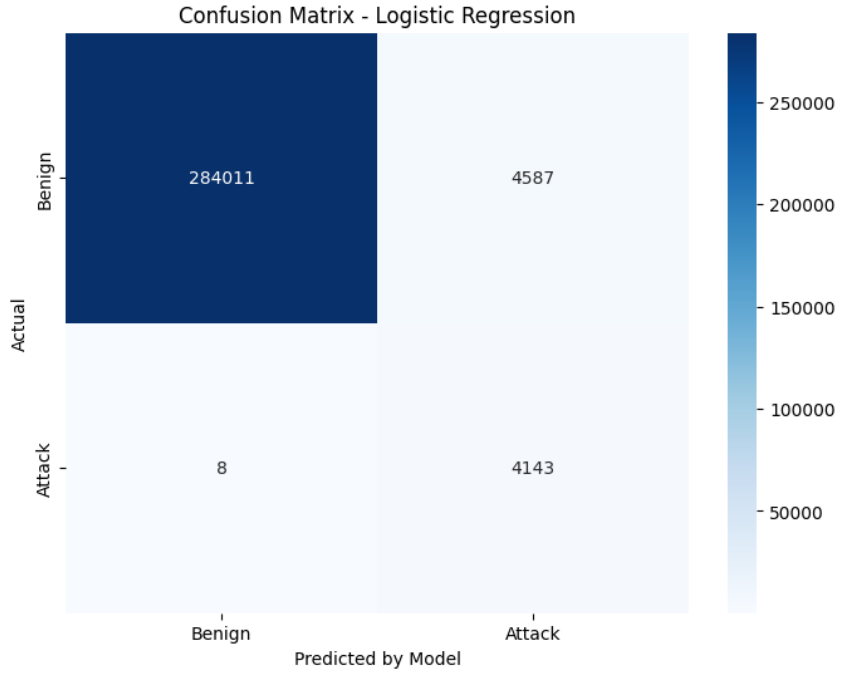

Figure 7: Confusion Matrix for the Logistic Regression Baseline (Infrastructure Layer).

Based on the quadrant analysis of the matrix, three critical observations can be drawn regarding the model's operational behavior:

- **High Detection Rate:** The bottom-right quadrant shows that 4,143 brute force attempts were correctly identified;
- **Minimal Missed Attacks:** Crucially, the bottom-left quadrant (FN) indicates only 8 missed attacks. In a dataset with over 4,000 malicious instances, missing only 8 demonstrates the model's high sensitivity to threats;

- **Operational Noise:** However, the top-right quadrant reveals 4,587 FPs. This implies that for every true attack detected, the model generated roughly one false alarm on legitimate traffic (Precision of 0.47). While this ensures security, it creates a significant operational overhead for analysts, justifying the investigation into more precise models or refinement techniques.

4.1.2 Active Learning Efficiency

Following the baseline evaluation, the Active Learning simulation was conducted to assess performance under severe data constraints. The model, using the same Logistic Regression architecture with `class_weight='balanced'`, was initialized with a seed of only 20 samples and allowed to query the oracle 100 times. This process resulted in a final training set of only 120 labeled instances.

Despite utilizing approximately 99.98% less training data than the supervised baseline (120 vs. 683,078 samples), the active model demonstrated a significant ability to learn the core attack patterns. The quantitative results are detailed in Table 5.

Table 5: Performance Metrics for Active Learning (Infrastructure Layer)

Class	Precision	Recall	F1-Score
Benign (0)	1.00	0.98	0.99
Attack (1)	0.30	0.71	0.42

To provide a visual perspective on these findings, Figure 8 illustrates the learning trajectory, showing a steep ascent in accuracy during the initial queries. This reinforces the efficiency of `Uncertainty Sampling` in rapidly acquiring knowledge. Complementing this, Figure 9 presents the final Confusion Matrix, which explicitly details the operational trade-offs accepted by drastically reducing the labeling budget.

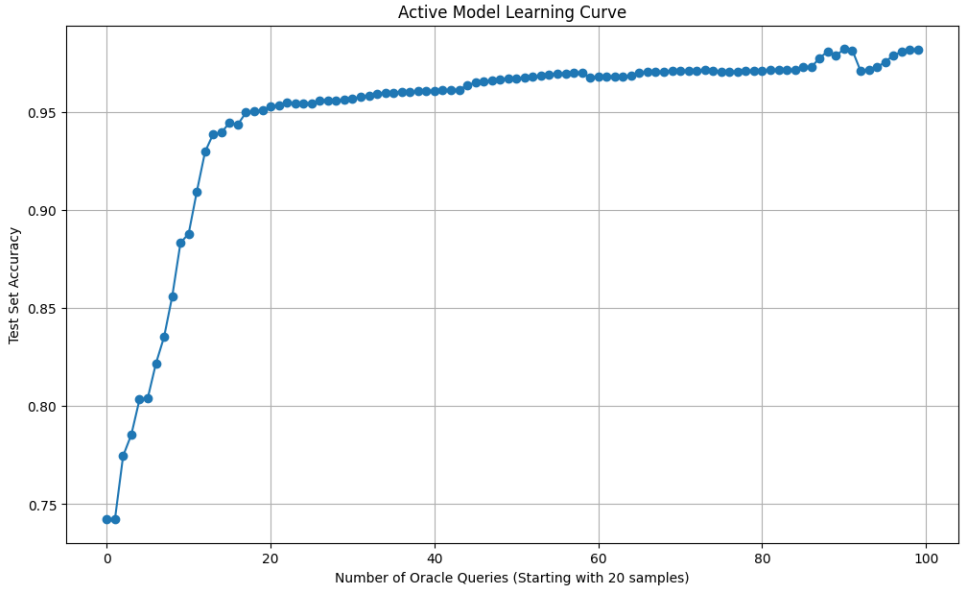

Figure 8: Active Learning Curve (Infrastructure Layer), showing rapid performance gains in the initial queries.

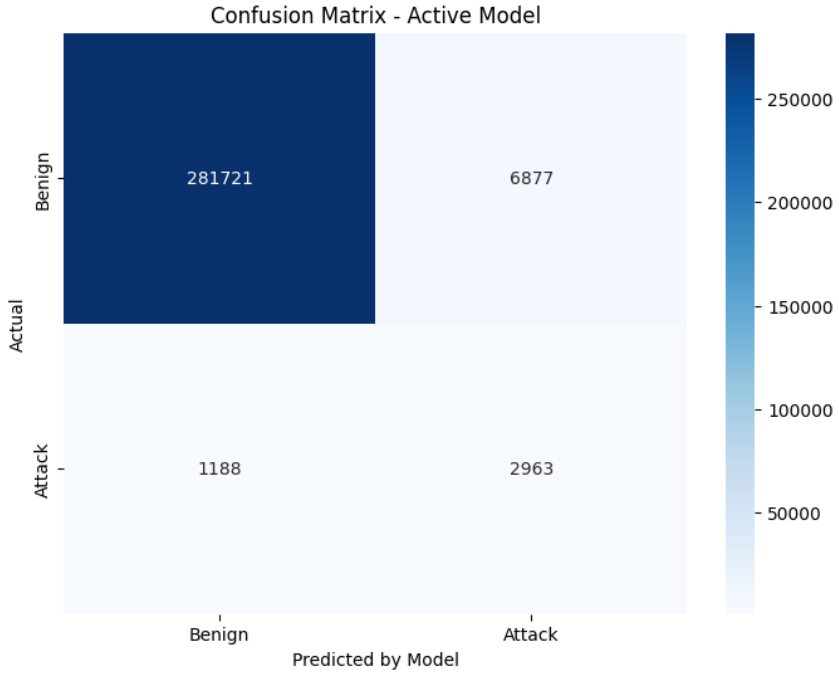

Figure 9: Confusion Matrix for the Active Learning Model (120 samples).

A joint analysis of the quantitative metrics and the graphical breakdown reveals the following key insights:

- **Detection Capability (Recall 0.71):** With a minimal budget, the model successfully identified the majority of brute force attacks (2,963 TPs). Achieving nearly

71% recall with such data scarcity proves that the model effectively prioritized the most informative attack signatures;

- **The Cost of Frugality (Missed Attacks):** The most significant impact of using limited data is evident in the bottom-left quadrant (FN), showing 1,188 missed attacks. Unlike the baseline model that missed almost nothing, the active model failed to identify approximately 29% of threats, representing the trade-off for using 99.98% less training data;

- **Operational Noise (Precision 0.30):** The top-right quadrant indicates 6,877 FPs. This high number of false alarms suggests a conservative ”flag-everything” strategy, which aligns with the safety-first priority in cybersecurity [6], where generating false alarms is often preferable to missing a breach;

- **Efficiency Analysis:** While the model does not achieve the perfect safety of the fully supervised baseline, it successfully creates a viable detection system with a resource expenditure that is orders of magnitude lower, validating its value as a rapid deployment strategy that directly addresses the challenge of high labeling costs.

It is crucial to note that the performance gap compared to the supervised baseline (Recall 1.00 vs. 0.71) is an expected consequence of the extreme reduction in training data size (from $\approx$683k to 120 samples). The Active Learning model did not ’worsen’ the data; rather, it reached a performance ceiling limited by the minimal information budget provided, demonstrating the efficiency of the sampling strategy in extracting maximum value from limited interactions.

4.2 CASE STUDY II - APPLICATION LAYER (WEB ATTACKS)

The application-layer threat detection employs Random Forest to handle complex, non-linear attack patterns. This supervised baseline evaluates the CIC-IDS2017 Thursday dataset for identifying multifaceted web attacks such as Brute Force, XSS, and SQL Injection camouflaged in normal web traffic.

4.2.1 Supervised Baseline Performance

The combined dataset comprised a total of 700,284 network flows. Similar to the infrastructure scenario, the class distribution was severely imbalanced, with legitimate traffic representing 99.7% of the data. The attack class was composed of a heterogeneous mix of web-based threats, including *Web Attack – Brute Force* (1,507 instances), XSS (652), and SQL Injection (21), totaling 2,180 malicious instances.

Data preprocessing followed the established pipeline, including the removal of non-numeric artifacts and missing values. Subsequently, the Random Forest classifier was

trained on 70% of the data ($N_{train} \approx 490,000$) using the `class_weight='balanced'` parameter to mitigate the imbalance. The model was evaluated on a test set containing 210,086 flows, yielding the quantitative results presented in Table 6.

Table 6: Performance Metrics for Random Forest (Application Layer)

Class	Precision	Recall	F1-Score
Benign (0)	1.00	1.00	1.00
Attack (1)	1.00	0.97	0.99

The performance of the Random Forest classifier presents a distinct contrast to the linear model used in the first case study. The confusion matrix (Figure 10) reveals two critical insights:

- **Perfect Precision (Zero False Positives):** The top-right quadrant shows a value of 0. This indicates that not a single legitimate user session was incorrectly flagged as an attack. This exceptional precision validates the Random Forest's ability to model complex, non-linear decision boundaries in web traffic, effectively eliminating operational noise;
- **Robust Detection (17 False Negatives):** The bottom-left quadrant indicates only 17 missed attacks out of 654. This corresponds to a Recall of 97.4%. While slightly lower than the perfect recall of the Logistic Regression in Case I, the trade-off for perfect precision is highly favorable in a web environment where traffic volume is massive.

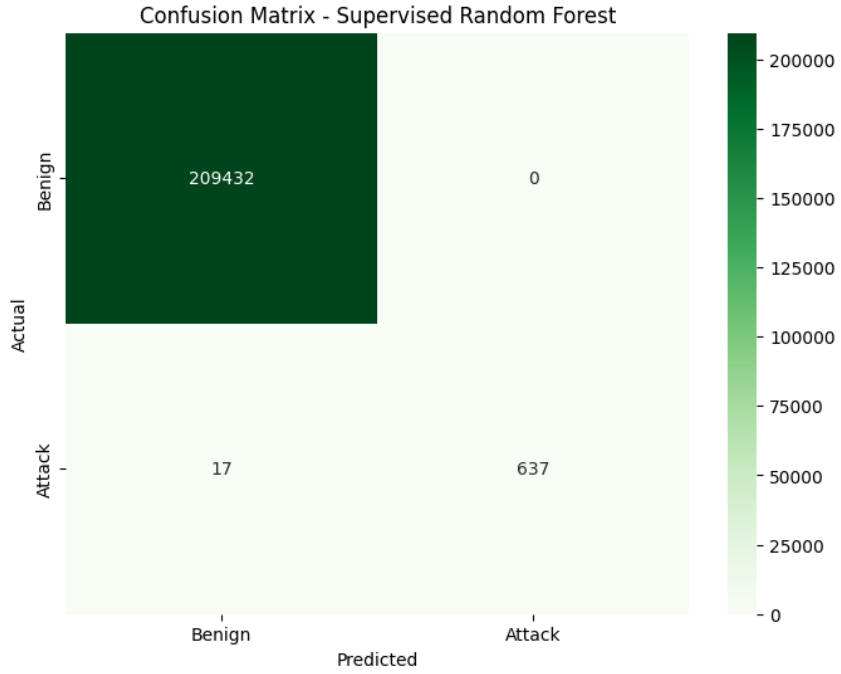

Figure 10: Confusion Matrix for the Random Forest Baseline.

Furthermore, the interpretability analysis provided by the Random Forest algorithm offers insight into the attack signatures. Figure 11 displays the top 10 most discriminatory features. Notably, `Max Packet Length` and `Fwd IAT Min` (Forward Inter-Arrival Time) emerged as the most critical indicators. This suggests that Web Brute Force attacks are primarily distinguishable by the specific size of the request packets and the unnatural timing between them, rather than just volume.

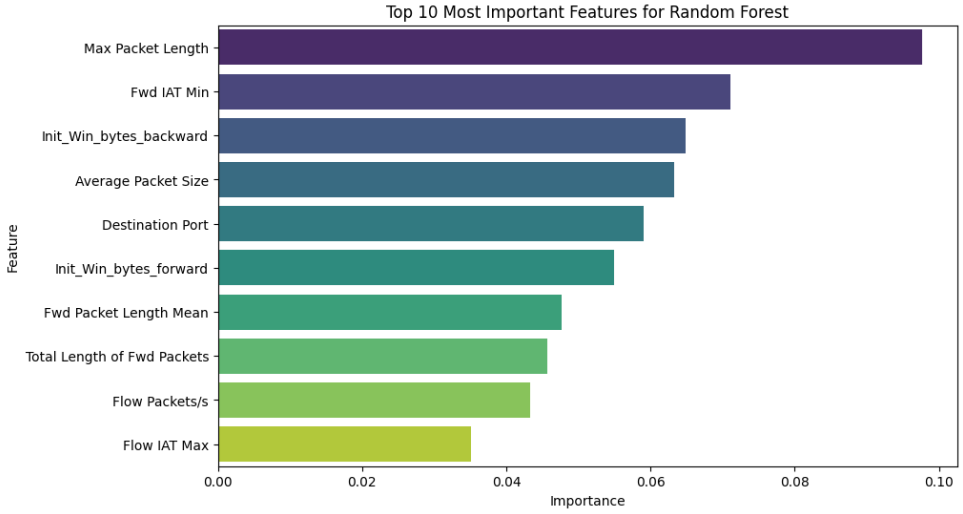

Figure 11: Top 10 Feature Importance for Random Forest, highlighting packet length and timing as key attack indicators.

4.2.2 Active Learning Efficiency

To evaluate the efficiency of the proposed method in the application layer scenario, the Active Learning simulation was executed using the Random Forest classifier. The model was initialized with the same constraints as the previous case study: a seed of 20 samples and a budget of 100 oracle queries, resulting in a final training set of 120 labeled instances.

The quantitative results, detailed in Table 7, demonstrate that the active learning strategy successfully adapted to the complex patterns of web traffic even with minimal data.

Table 7: Performance Metrics for Active Learning (Application Layer)

Class	Precision	Recall	F1-Score
Benign (0)	1.00	1.00	1.00
Attack (1)	1.00	0.89	0.94

The learning trajectory and final classification performance are visualized in Figure 12 and Figure 13, respectively. A joint analysis of these artifacts reveals compelling insights into the cost-benefit ratio of the approach:

- **Maintained Precision (Zero False Positives):** Remarkably, the active model maintained the perfect precision of the baseline (1.00). The top-right quadrant of the confusion matrix shows zero false alarms. This indicates that the Random Forest, guided by uncertainty sampling, learned to be extremely selective, flagging only traffic that exhibits clear attack signatures, which is ideal for minimizing analyst fatigue;
- **High Detection Capability (Recall 0.89):** With only 120 samples, the model correctly identified 583 out of 654 attacks (89%). While this represents a drop compared to the baseline's 97% recall, capturing nearly 90% of threats with less than 0.03% of the training data is a significant efficiency milestone. The 71 missed attacks (FNs) suggest that the model prioritized learning the most dominant attack patterns first;
- **Efficiency Analysis:** The experiment validates that Active Learning is highly effective even for complex, non-linear web attack vectors. The model rapidly converged to a high-performance state, proving that a massive reduction in labeling effort does not necessarily compromise the system's reliability (precision), although a trade-off in sensitivity (recall) is observed.

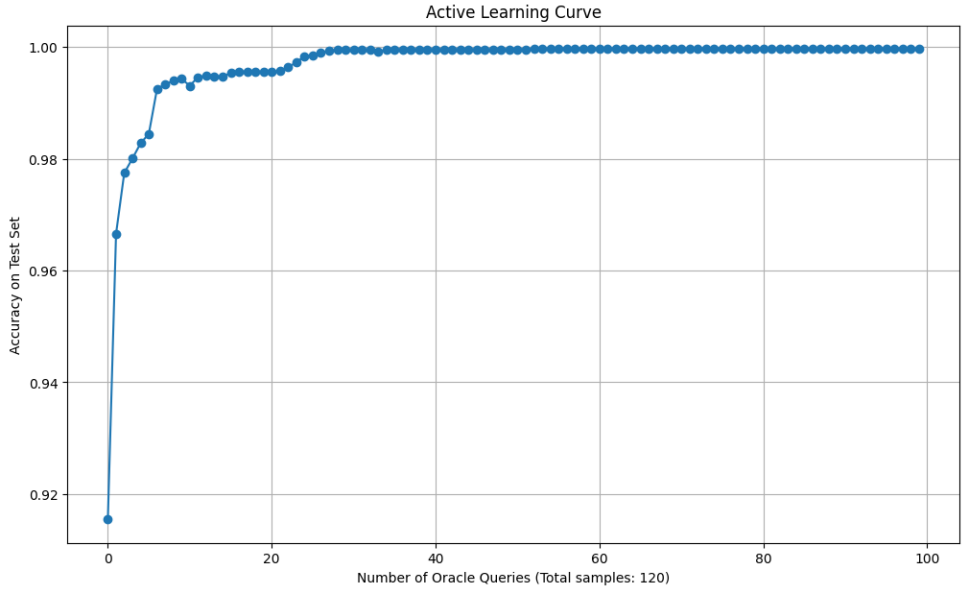

Figure 12: Active Learning Curve (Application Layer), illustrating the rapid convergence of accuracy.

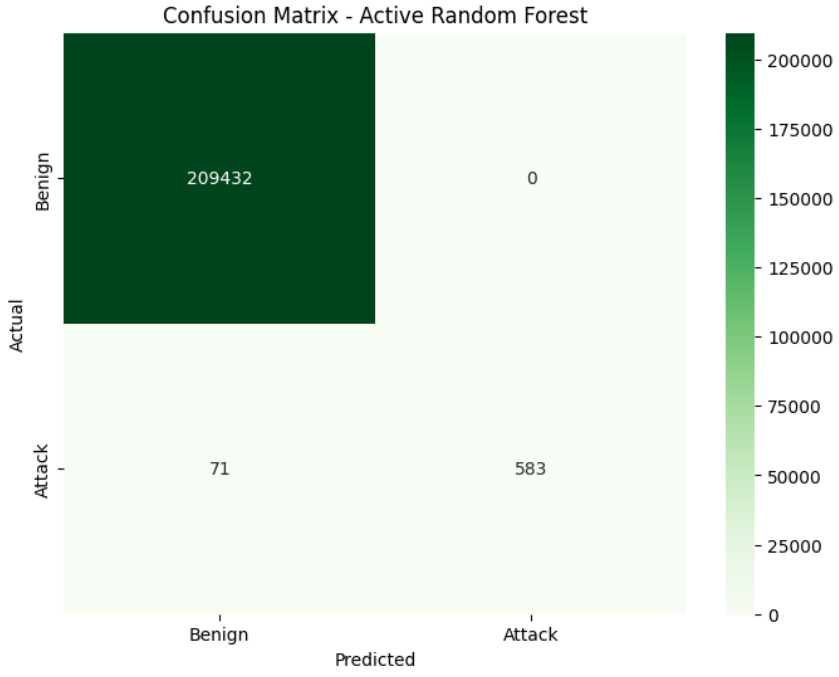

Figure 13: Confusion Matrix for the Active Learning Model (Application Layer), showing perfect precision (0 FP) and solid recall (89%), with only 71 missed attacks (FN).

4.3 CASE STUDY III - POST-AUTHENTICATION ANOMALIES (LANL)

This study addressed the challenge of detecting insider threats and lateral movement in an unsupervised environment. Due to the computational constraints of pro-

cessing the massive raw logs, a representative subset of approximately 5 million authentication events was utilized for this analysis.

4.3.1 Unsupervised Anomaly Detection

The Isolation Forest algorithm was trained on the dataset to identify deviations from normal behavior patterns without prior knowledge of attack signatures. The model was configured with a contamination rate of 1%, resulting in the isolation of 124,716 anomalous events out of the total sample.

To validate the efficacy of this unsupervised approach, a statistical comparison was performed between the events classified as Normal and those flagged as Anomalous. The results, visualized in Figures 14 and 15, reveal a distinct ”fingerprint” for potentially malicious behavior:

- **Novelty as a Key Indicator:** The `is_first_access` feature was the strongest discriminator. First-time access occurred in only 0.52% of normal traffic versus 96.4% of anomalies, validating that lateral movement is a primary risk indicator;
- **Temporal Anomalies:** The model identified that anomalous events showed a tendency towards off-hours (average hour $\approx$ 06:38 AM), contrasting with normal traffic centered around business hours (average hour $\approx$ 10:30 AM);
- **The Frequency Paradox:** Anomalies exhibited a lower frequency of logins per user ($\approx$ 43) compared to normal traffic ($\approx$ 709). This indicates that the model successfully distinguished between the high-volume ”noise” of automated system accounts and the low-volume, stealthy behavior characteristic of human attackers moving laterally.

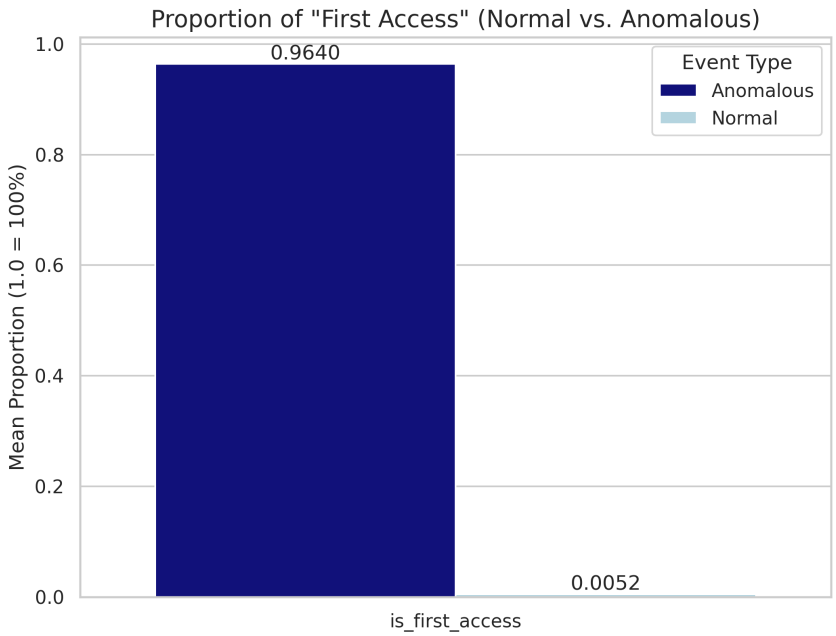

Figure 14: Comparison of 'First Access' probability: Anomalies vs. Normal behavior.

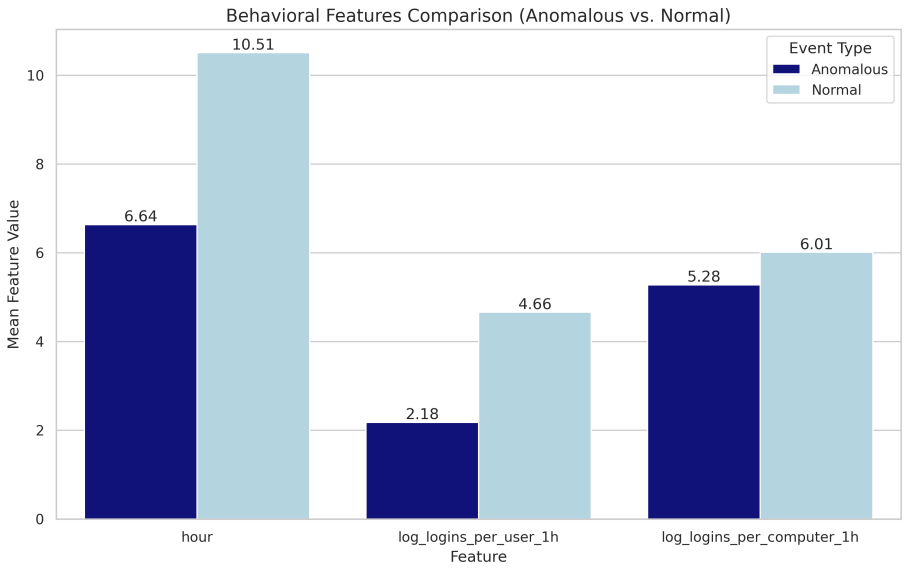

Figure 15: Comparison of temporal and frequency features: Anomalous events occur during off-hours (≈6:40 AM) and show lower user login frequency compared to normal automated traffic (≈10:30 AM).

4.3.2 Active Learning Refinement

To mitigate the false positive rate inherent in unsupervised detection, an Active

Learning module was applied to the pool of 124,716 isolated anomalies. In this phase, an automated oracle simulated the decision-making of a security analyst, labeling a seed of 20 samples and responding to 100 strategic queries based on model uncertainty, totaling 120 labeled instances.

The trained model was then deployed to automatically classify the remaining unlabelled anomalies. The results of this automated triage are summarized in Table 8.

Table 8: Automated Triage Results by the Active Learning Model (LANL)

Classification	**Count**	**Percentage**
High Risk (Real Attack - 1)	95,426	76.5%
Low Risk (False Positive - 0)	29,170	23.5%

These results demonstrate the critical value of the Active Learning layer. While the unsupervised model successfully filtered the massive dataset to its most suspicious 1%, it still included a significant portion (23.5%) of events that, upon closer behavioral inspection considering time context and frequency patterns, were deemed low risk. The Active Learning model successfully identified and filtered out these 29,170 false positives, reducing the analyst's workload by nearly a quarter while maintaining focus on the highest-risk activities.

To validate the real-time viability of the proposed hybrid architecture, a comprehensive latency analysis was conducted. The total computational cost was calculated for both normal traffic processing and the worst-case scenario (anomaly refinement).

Table 9: Computational Cost Analysis of the Hybrid Pipeline

Pipeline Stage	**Latency per Event (ms)**	**Throughput (Events/sec)**
Stage 1: Unsupervised Filter	0.0005 ms	$\approx$ 2,000,000
Stage 2: Active Refinement	0.0007 ms	$\approx$ 1,428,000
Total Latency (Worst Case)	**0.0012 ms**	$\approx$ **833,000**

The results demonstrate that even in the worst-case scenario, where an event is flagged as anomalous and requires secondary classification by the Active Learning model, the total processing time remains at approximately 1.2 microseconds. This confirms that the addition of the Active Learning layer introduces negligible overhead, maintaining the system's capability to process high-throughput network traffic in real-time.

5

5 CONCLUSION

This work demonstrates that combining Unsupervised filtering with Active Learning refinement constitutes a scalable and efficient methodology for modern intrusion detection, addressing the critical bottleneck of data labeling in an era of exponentially growing cyber threats. The primary objective of evaluating the efficacy and efficiency of this strategy was successfully met. Motivated by the challenge of managing massive volumes of unlabeled log data, the study confirmed that intelligent data selection significantly reduces the human cost of labeling without compromising detection performance.

5.1 SYNTHESIS OF RESULTS

The experimental outcomes provided strong evidence supporting the initial hypothesis that Active Learning can effectively optimize intrusion detection systems. The study successfully addressed each specific objective:

- **Addressing Real-Time Viability (Objective 1):** The analysis confirmed that the proposed architecture operates with negligible latency, validating its suitability for high-throughput network environments where real-time processing is mandatory. The empirical measurement of the most complex scenario (hybrid pipeline) confirmed that the system introduces no perceptible delay to authentication processes;
- **Validation via Comparative Analysis (Objectives 2 & 3):** The comparison between paradigms revealed distinct outcomes depending on the attack vector;
  - In the Web Attack scenario, the active model rapidly converged to performance levels closely mirroring the supervised baseline, demonstrating high efficiency;
  - In the Infrastructure scenario, a strategic trade-off was identified: while the active model successfully learned the core attack signatures with minimal data, a slight reduction in recall compared to the baseline highlights that Active Learning serves primarily as a rapid prototyping strategy, which can be further refined with incremental labeling.

- **Reduction of Human Effort (Objective 4):** This was the most significant contribution. Across all case studies, the methodology demonstrated that robust classifiers could be trained using a minimal fraction of the data required by traditional methods. By reducing the labeling requirement by over 99%, the approach proved to be a scalable solution for the critical bottleneck of manual data annotation;
- **Effectiveness in Unsupervised Contexts (Case Study III):** Complementing the supervised scenarios, the hybrid architecture applied to the LANL dataset proved effective in unsupervised domains. The combination of Isolation Forest filtering with Active Learning refinement successfully isolated behavioral anomalies (such as lateral movement) while filtering out a significant volume of false positives, streamlining the workflow for security analysts.

5.2 LIMITATIONS

Despite the positive results, the study identified practical limitations that must be acknowledged. A significant challenge observed during the execution of Case Study III (LANL) was the cognitive load on the Oracle. Although Active Learning reduces the quantity of labels, it increases the complexity of the decision-making process.

In the experiment, acting as the Oracle required intense manual analysis of context (time, frequency, user history) for each query to distinguish between a true lateral movement and a false positive. This confirms that while the method reduces the volume of work, it demands high expertise from the analyst. Furthermore, the experiments were conducted as ”offline simulations,” where the Oracle’s response time was not factored into the learning loop speed, which could be a bottleneck in a real-time training scenario.

To advance the findings of this research, several directions for future investigation are recommended. First, the integration of Cost-Sensitive Active Learning could be explored by incorporating the ”difficulty” or ”time cost” of labeling into the query strategy. This would enable the model to better balance information gain against the effort required from the analyst.

Furthermore, developing a pipeline for Online Learning Integration is crucial for real-world deployment. Future studies should focus on models that update their parameters continuously in a streaming environment, rather than in batch simulations, to test resilience against concept drift in live network traffic.

REFERENCES

References

[1] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA: MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org

[2] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*. Belmont, California: Wadsworth International Group, 1984.

[3] D. M. Almeida, A. Teixeira, S. Mendes, J. Câmara, and C. Martins, "Automated prediction of good dictionary examples (gdex): A comprehensive experiment with distant supervision, machine learning, and word embedding-based deep learning techniques," *Complexity*, vol. 2021, pp. 1–16, 2021. [Online]. Available: https://www.researchgate.net/publication/354354484_Automated_Prediction_of_Good_Dictionary_EXamples_GDEX_A_Comprehensive_Experiment_with_Distant_Supervision_Machine_Learning_and_Word_Embedding-Based_Deep_Learning_Techniques

[4] Y. Regaya, F. Fadli, and A. Amira, "Point-denoise: Unsupervised outlier detection for 3d point clouds enhancement," *Multimedia Tools and Applications*, vol. 80, pp. 1–17, 07 2021.

[5] T. Fawcett, "Introduction to roc analysis," *Pattern Recognition Letters*, vol. 27, pp. 861–874, 06 2006.

[6] R. Anderson, *Security Engineering: A Guide to Building Dependable Distributed Systems*, 3rd ed. Indianapolis, IN: John Wiley & Sons, 2020. [Online]. Available: https://www.cl.cam.ac.uk/fml/seclab/book/

[7] H. Krasner, "The cost of poor software quality in the us: A 2022 report," Consortium for Information & Software Quality (CISQ), Needham, MA, Tech. Rep., 2022. [Online]. Available: https://www.it-cisq.org/the-cost-of-poor-quality-software-in-the-us-a-2022-report/

[8] IBM Security. (2025) Cost of a data breach report 2025. [Online]. Available: https://www.ibm.com/reports/data-breach

[9] W. Stallings and L. Brown, *Computer Security: Principles and Practice*, 4th ed. Boston, MA: Pearson, 2018.

[10] OWASP Foundation, "Owasp top ten," https://owasp.org/www-project-top-ten/, 2023.

[11] K. Shaukat, S. Luo, V. Varadharajan, I. A. Hameed, and M. Xu, “A survey on machine learning techniques for cyber security in the last decade,” *IEEE access*, vol. 8, pp. 222 310–222 354, 2020.

[12] Z. T. Pritee, M. H. Anik, S. B. Alam, J. R. Jim, M. M. Kabir, and M. F. Mridha, “Machine learning and deep learning for user authentication and authorization in cybersecurity: A state-of-the-art review,” *Computers & Security*, vol. 140, p. 103747, 2024.

[13] A. Al Siam, M. Alazab, A. Awajan, and N. Faruqui, “A comprehensive review of ai’s current impact and future prospects in cybersecurity.” *IEEE Access*, 2025.

[14] F. Olsson, “A literature survey of active machine learning in the context of natural language processing,” 2009.

[15] S. Tumin and S. Encheva, “A closer look at authentication and authorization mechanisms for web-based applications,” in *5th World Congress: Applied Computing Conference*, vol. 12, 2012.

[16] P. A. Networks, “What is multi-factor authentication (mfa)? examples and methods,” https://www.paloaltonetworks.com.au/cyberpedia/what-are-multi-factor-authentication-mfa-examples-and-methods. [Online]. Available: https://www.paloaltonetworks.com.au/cyberpedia/what-are-multi-factor-authentication-mfa-examples-and-methods

[17] A. Hagberg, A. Kent, N. Lemons, and J. Neil, “Credential hopping in authentication graphs,” in *2014 International Conference on Signal-Image Technology Internet-Based Systems (SITIS)*. IEEE Computer Society, Nov. 2014.

[18] S. Ramakrishnan, “Revolutionizing role-based access control: The impact of ai and machine learning in identity and access management,” *Journal of Artificial Intelligence & Cloud Computing*, vol. 2, pp. 1–7, 2023.

[19] Q.-V. Dang, “Active learning for intrusion detection systems,” in *2020 RIVF International Conference on Computing and Communication Technologies (RIVF)*. IEEE, 2020, pp. 1–3.

[20] M. Hazratifard, F. Gebali, and M. Mamun, “Using machine learning for dynamic authentication in telehealth: A tutorial,” *Sensors*, vol. 22, no. 19, p. 7655, 2022.

[21] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[22] R. Sutton and A. Barto, “Reinforcement learning: An introduction,” *IEEE Transactions on Neural Networks*, vol. 9, no. 5, pp. 1054–1054, 1998.

[23] IBM. (2025) Random forest (floresta aleatória). [Online]. Available: https://www.ibm.com/br-pt/think/topics/random-forest

[24] DataCamp. (2024) Isolation forest: Um guia completo para detecção de anomalias. [Online]. Available: https://www.datacamp.com/pt/tutorial/isolation-forest

[25] B. Settles, “Active learning literature survey,” 2009.

[26] X. Deng, Q. Liu, Y. Deng, and S. Mahadevan, “An improved method to construct basic probability assignment based on the confusion matrix for classification problem,” *Information sciences*, vol. 340, pp. 250–261, 2016.

[27] A. Alsaedi, N. Moustafa, Z. Tari, A. Mahmood, and A. Anwar, “Ton$_{i}ottelemetrydataset : Anewgenerationdatasetofiotandiiotfordata - drivenintrusiondetectionsystems,"$ *IEEE Access*$, vol. 8, pp.165\,130 - -165\,150, 2020.$

[28] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, “Toward generating a new intrusion detection dataset and intrusion traffic characterization,” in *Proceedings of the 4th International Conference on Information Systems Security and Privacy*. SCITEPRESS, 2018, pp. 108–116.

[29] A. Kent, *Cyber security data sources for dynamic network research*, 05 2016, pp. 37–65.

[30] N. Moustafa and J. Slay, “Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set),” in *2015 Military Communications and Information Systems Conference (MilCIS)*, 2015, pp. 1–6.

[31] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, “A detailed analysis of the kdd cup 99 data set,” in *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, 2009, pp. 1–6.

[32] R. Lippmann, J. W. Haines, D. J. Fried, J. Korba, and K. Das, “Evaluating intrusion detection systems: The 1998 darpa off-line intrusion detection evaluation,” in *Proceedings DARPA Information Survivability Conference and Exposition. DISCEX’00*, vol. 2. IEEE, 2000, pp. 12–26.

[33] S. R. C. C. T. Tadi, “Strengthening api security: Behavioral authentication and intelligent threat mitigation with data-driven models,” *International Journal of Advanced Research in Science, Communication and Technology*, pp. 1343–1349, 2023.

[34] D. Albert-Weiss and A. Osman, “Interactive deep learning for shelf life prediction of muskmelons based on an active learning approach,” *Sensors*, vol. 22, no. 2, p. 414, 2022.

[35] S. Wiefling, P. R. Jørgensen, S. Thunem, and L. L. Iacono, “Pump up password security! evaluating and enhancing risk-based authentication on a real-world large-scale online service,” *ACM Transactions on Privacy and Security*, vol. 26, no. 1, pp. 1–36, 2022.

[36] G. Van Rossum and F. L. Drake, *Python reference manual*. Centrum voor Wiskunde en Informatica Amsterdam, 1995.

[37] T. Kluyver, B. Ragan-Kelley, F. Pérez, B. Granger, M. Bussonnier, J. Frederic, K. Kelley, J. Hamrick, J. Grout, S. Corlay *et al.*, “Jupyter notebooks-a publishing format for reproducible computational workflows,” in *Positioning and power in academic publishing: Players, agents and agendas*. IOS Press, 2016, pp. 87–90.

[38] W. McKinney *et al.*, “Data structures for statistical computing in python,” in *Proceedings of the 9th Python in Science Conference*, vol. 445. Austin, TX, 2010, pp. 51–56.

[39] C. R. Harris, K. J. Millman, S. J. Van Der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith *et al.*, “Array programming with numpy,” *Nature*, vol. 585, no. 7825, pp. 357–362, 2020.

[40] J. D. Hunter, “Matplotlib: A 2d graphics environment,” *Computing in science & engineering*, vol. 9, no. 3, pp. 90–95, 2007.

[41] M. L. Waskom, “seaborn: statistical data visualization,” *Journal of Open Source Software*, vol. 6, no. 60, p. 3021, 2021.

[42] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, “Scikit-learn: Machine learning in python,” *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[43] T. Danka and P. Horvath, “modAL: A modular active learning framework for Python,” *arXiv preprint arXiv:1805.00979*, 2018, available on arXiv: https://arxiv.org/abs/1805.00979. [Online]. Available: https://github.com/modAL-python/modAL

[44] A. Modi and K. Navadiya, “Anomaly detection in cybersecurity using random forest,” *International Journal of Engineering Research & Technology (IJERT)*, vol. 5, no. 02, 2025.

[45] C. Singh, V. Vijayalakshmi, and H. Raj, “A machine learning approach for web application vulnerability detection using random forest.” IJRASET, 2022.

[46] K. Kostas, “Anomaly detection in networks using machine learning,” Ph.D. dissertation, 08 2018.

[47] A. S. Pope, D. R. Tauritz, and M. Turcotte, “Automated design of tailored link prediction heuristics for applications in enterprise network security,” in *Proceedings of the Genetic and Evolutionary Computation Conference Companion (GECCO ’19)*. New York, NY, USA: Association for Computing Machinery, 2019, pp. 1634–1642.

[48] F. T. Liu, K. M. Ting, and Z.-H. Zhou, “Isolation forest,” in *2008 Eighth IEEE International Conference on Data Mining*, 2008, pp. 413–422.

idp

Ensino que
te **conecta**