**Ramanujan College**
University of Delhi
(Accredited Grade ’A++’ by NAAC)

# DISSERTATION

Submitted in partial fulfillment of the requirements for the degree of

## Bachelor of Science (Honours) Computer Science

# Dynamic Trust Scoring with Behavioral Analytics for Zero Trust Architecture: A Comprehensive Implementation and Evaluation Framework

**Submitted by:**
Lakshanya Harit
Roll No: 20221425

**Under the Supervision of:**
Dr. Nikhil Kumar Rajput
Assistant Professor, Department of Computer Science

**Co-Supervisor:**
Mr. Vipin Kumar Rathi
Assistant Professor, Department of Computer Science

**Department of Computer Science**
Ramanujan College, University of Delhi
September 2025

# CERTIFICATE

This is to certify that the dissertation entitled **"Dynamic Trust Scoring with Behavioral Analytics for Zero Trust Architecture: A Comprehensive Implementation and Evaluation Framework"** submitted by **Lakshanya Harit**, Roll No. **20221425**, in partial fulfillment of the requirements for the award of the degree of **Bachelor of Science (Honours) Computer Science** of University of Delhi, is a record of the candidate's own work carried out by him/her under my supervision and guidance.

The matter embodied in this dissertation is original and has not been submitted for the award of any other degree or diploma.

| | |
|---|---|
| **Dr. Nikhil Kumar Rajput** | **Mr. Vipin Kumar Rathi** |
| Assistant Professor | Professor |
| Department of Computer Science | Department of Computer Science |
| Ramanujan College | Ramanujan College |
| University of Delhi | University of Delhi |

**Date:** ______________ **Place:** New Delhi

# ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to all those who have contributed to the completion of this dissertation.

First and foremost, I am deeply grateful to my supervisor, **Dr. Nikhil Kumar Rajput**, for his invaluable guidance, continuous support, and patience throughout my research. His expertise in machine learning and software engineering has been instrumental in shaping this work. I also extend my heartfelt thanks to my co-supervisor, **Mr. Vipin Kumar Rathi**, for his insightful feedback and encouragement.

I would like to thank the faculty members of the Department of Computer Science, Ramanujan College, for providing me with a strong foundation in computer science principles and for creating an environment conducive to learning and research.

My sincere appreciation goes to my fellow students and researchers who provided valuable discussions, feedback, and moral support throughout this journey.

I am grateful to the developers and maintainers of the open-source projects and datasets that made this research possible. The availability of quality resources significantly contributed to the success of this work.

I would also like to acknowledge the computational resources provided by the college's computer lab, which were essential for conducting the experiments presented in this dissertation.

Finally, I express my deepest gratitude to my family and friends for their unwavering support, understanding, and encouragement throughout my academic journey. Their belief in me has been a constant source of motivation.

**Lakshanya Harit**
Roll No: 20221425
September 2025

# ABSTRACT

Zero Trust Architecture (ZTA) is founded on the principle of “never trust, always verify” in which access decisions are based on continuous validation of user and device behaviour rather than one-time authentication. In such environments, trust must be assessed dynamically using measurable behavioural indicators instead of static credentials alone.

This dissertation presents a prototype Dynamic Trust Scoring System that applies behavioural analytics and anomaly detection to support risk-aware decision-making within a Zero Trust context. Synthetic datasets are generated to simulate realistic user activity patterns across multiple behavioural archetypes, including regular, irregular, and malicious users. From these simulated access logs, behavioural features are extracted and used to compute trust scores that evolve over time.

An Isolation Forest model is employed to learn baseline behavioural patterns and identify anomalous activity. The resulting anomaly signals are integrated into the trust computation process alongside behavioural consistency indicators and component-level trust factors. Adaptive thresholding is then applied to classify access decisions into graduated policy responses such as allow, challenge, and block. To support interpretability, the system incorporates an explainable scoring structure that highlights how individual behavioural components influence overall trust assessments.

The implementation is evaluated using standard performance metrics, including precision, recall, F1-score, and ROC–AUC, in addition to behavioural stability and temporal trend analysis. All evaluations are conducted on synthetically generated datasets designed to enable controlled experimentation rather than real-world deployment. While limited by the absence of operational security logs, the study demonstrates the technical feasibility of integrating behavioural trust scoring, anomaly detection, and adaptive policy logic within a Zero Trust security model.

The proposed framework establishes a structured experimental platform that can be extended in future work using real-world data and enterprise-scale environments.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Background

Cybersecurity has undergone a fundamental transformation in recent decades. Traditional perimeter-based security models were designed for environments in which organisational resources were located within a well-defined network boundary, with access controlled primarily through firewalls and virtual private networks (VPNs). Once a user or device successfully authenticated at the perimeter, the system often assumed implicit trust and granted broad access to internal resources.

However, this approach has shown reduced effectiveness in certain modern computing contexts. The expansion of cloud services, mobile devices, distributed infrastructures, and remote working practices has weakened the clarity of traditional network boundaries, complicating perimeter-based access control. Concurrently, threat actors have increasingly leveraged compromised credentials, insider access, and advanced persistent threats to achieve lateral movement within networks after initial access. Documented security incidents indicate that static defence mechanisms and single-instance authentication may be insufficient to address evolving operational and data security risks.

In response to these challenges, Zero Trust Architecture (ZTA) has emerged as a significant paradigm shift in security thinking. Based on the principle of "never trust, always verify", ZTA requires continuous validation of every user, device, and access request, irrespective of network location. Rather than assuming that authenticated users remain trustworthy, Zero Trust evaluates risk dynamically using contextual and behavioural indicators. Implementing this model effectively requires mechanisms capable of monitoring activity patterns, adapting to changes in behaviour, and supporting continuous, risk-aware access decisions.

Dynamic Trust Scoring, supported by behavioural analytics and machine learning, provides one such mechanism. Unlike static rule-based systems, Dynamic

Trust Scoring evaluates temporal activity (such as access time), spatial information (such as location or network origin), device characteristics, and usage behaviour to estimate the likelihood of legitimate or malicious intent. In this study, anomaly detection techniques : including the Isolation Forest algorithm : are incorporated to identify deviations from normal behavioural baselines. This combination enables an adaptive, continuously evolving trust model that better aligns with the principles of Zero Trust security.

## 1.2 Problem Statement

Although Zero Trust is widely recognised as a robust and future-oriented security model, implementing it effectively in practice remains challenging. Many existing approaches rely on predefined rules or periodic authentication checks that do not capture the complexity and variability of real-world user behaviour. These limitations often result in three key challenges:

1. **Lack of adaptability:** Static and policy-driven systems struggle to respond to evolving threat strategies and natural behavioural variation.

2. **Poor balance between security and usability:** Overly strict controls disrupt legitimate work, while lenient configurations increase security risk.

3. **Limited transparency:** Many trust-assessment approaches function as “black boxes,” making it difficult for administrators to understand, audit, or justify outcomes.

This project addresses these challenges by examining a Dynamic Trust Scoring approach that adapts to user behaviour over time, incorporates anomaly detection to identify deviations from baseline activity, and provides explainable trust decisions.

## 1.3 Research Objectives

The main objectives of this research are to:

1. Describe a behavioural trust-scoring framework that continuously evaluates user activity using temporal, spatial, device-based, and usage-pattern indicators.

2. Examine the role of anomaly-detection techniques, including Isolation Forest, in identifying deviations from established behavioural baselines within the trust-scoring process.

3. Explore adaptive threshold mechanisms that adjust access control policies according to individual user baselines.
4. Evaluate system performance using synthetic datasets representing regular, irregular, and malicious user behaviour.
5. Assess effectiveness through a combination of machine-learning metrics (precision, recall, F1 score, ROC analysis) and behavioural metrics such as consistency, adaptability, and stability.

## 1.4 Research Questions

To guide the investigation, this study is organised around the following research questions:

1. To what extent do behavioural features such as time of activity, location, device characteristics, and usage patterns distinguish between normal and malicious user behaviour?
   - Examined using ROC analysis, precision–recall evaluation, and F1-score comparison.
   - Analysed across multiple user types and behavioural contexts.
2. How do variations in security enforcement levels relate to user experience in Zero Trust environments?
   - Investigated through observed relationships between false-positive rates and detection performance.
3. What contributions do different behavioural components (temporal, spatial, device-based, and usage-based indicators) make to overall trust assessment?
   - Explored through component-level trust scoring and weighting analysis.
4. To what extent do adaptive thresholds influence trust-scoring outcomes across different user categories?
   - Evaluated by examining whether user-specific baselines affect false-positive rates while maintaining anomaly detection capability.
5. In what ways can explainable artificial intelligence mechanisms support transparency and auditability in trust-based access decisions?

  - Considered in relation to administrator interpretability and governance requirements.

6. How does the system respond to changes in user behaviour over time within the experimental setting?
   - Analysed through temporal behaviour trends and behavioural consistency measures.

7. How does system performance vary across regular, irregular, and malicious user archetypes?
   - Assessed through structured experimental evaluation.

These questions form the backbone of the research design and are systematically addressed in the subsequent chapters.

## 1.5 Research Methodology Overview

This research adopts a mixed computational and experimental methodology to examine behavioural trust scoring within a simulated Zero Trust environment. The approach combines quantitative evaluation, based on machine learning performance metrics, with qualitative analytical components focused on interpretability, behavioural patterns, and policy reasoning. The use of synthetic data enables controlled experimentation while supporting structured analysis of trust dynamics across different user archetypes.

The key stages of the methodology are as follows:

1. **Feature Extraction:** Temporal, spatial, device-based, and behavioural indicators are generated from synthetic access logs to support quantitative modelling and comparative analysis.

2. **User Profile Learning:** Baseline behavioural profiles are established for different user archetypes, and deviations from these baselines are identified using anomaly-detection techniques, including Isolation Forest.

3. **Trust Scoring:** Behavioural features are aggregated into an interpretable trust score, enabling both numerical evaluation and qualitative interpretation of trust evolution over time.

4. **Adaptive Thresholding:** User-specific thresholds are applied to classify access outcomes into allow, challenge, or block decisions, supporting analysis of policy behaviour and decision boundaries.

5. **Evaluation Framework:** Synthetic datasets are used to model regular, irregular, and malicious user groups, facilitating controlled comparison across behavioural categories.
6. **Metrics and Analysis:** System performance is assessed using quantitative machine learning metrics alongside qualitative examination of behavioural stability, trust-score trends, and decision interpretability.

## 1.6 Key Research Metrics

### 1.6.1 Performance Metrics

Performance metrics are used to quantitatively assess how effectively the system distinguishes malicious or irregular behaviour from normal user activity:

- **Detection Rate** : proportion of anomalous events correctly identified.
- **False Positive Rate** : proportion of normal behaviour incorrectly classified as suspicious.
- **Precision** : proportion of detected anomalies that correspond to genuine anomalous behaviour.
- **Recall** : proportion of true anomalies successfully detected.
- **F1 Score** : harmonic mean of precision and recall, reflecting balanced classification performance.
- **AUC–ROC** : area under the receiver operating characteristic curve, indicating overall discrimination capability.

### 1.6.2 Behavioural Metrics

Behavioural metrics support qualitative analysis by examining how trust scores evolve over time and how consistently the system responds to user behaviour:

- **Behavioural Consistency** : stability of user behaviour patterns across observation periods.
- **Trust-Score Variance** : degree of fluctuation in trust values for individual users.
- **Adaptation Rate** : responsiveness of trust scores to behavioural changes.
- **Feature Importance** : relative contribution of individual behavioural indicators to trust assessment.

### 1.6.3 Operational Metrics

Operational metrics provide contextual insight into system behaviour within the experimental environment:

- **Execution Time** : time required to generate datasets and perform analysis.
- **Resource Utilisation** : indicative CPU and memory requirements of the prototype.
- **Scalability** : observed behaviour as the number of simulated users increases.
- **Threshold Stability** : consistency of classification outcomes under adaptive thresholding.

## 1.7 Research Applications

- Behavioural analytics research environments
- Machine-learning anomaly-detection experiments
- Adaptive security policy design
- Usability and explainability studies in security

## 1.8 Scope and Limitations

### 1.8.1 Scope

- Focus remains on behavioural trust scoring within Zero Trust environments.
- Synthetic datasets are used for controlled analysis.
- Anomaly detection includes Isolation Forest techniques.
- Evaluation is based on ML-inspired and behavioural stability metrics.

### 1.8.2 Limitations

- No real-world organisational deployment data is included.
- Behavioural features are restricted to selected categories.
- The implementation emphasises clarity over enterprise optimisation.

## 1.9 Contributions

The key contributions of this research are:

1. Development of a behavioural trust-scoring prototype aligned with Zero Trust principles.
2. Design of a synthetic user-archetype dataset including regular, irregular, and malicious profiles.
3. Integration of Isolation Forest-based anomaly detection into trust scoring.
4. Provision of explainable visual analytics for interpreting trust patterns.
5. Establishment of a research platform for future adaptive-security investigations.

## 1.10 Organization of the Dissertation

The remainder of this dissertation is structured as follows:

- **Chapter 2: Literature Review** : Examines prior work on Zero Trust security models, behavioural analytics, trust scoring, and anomaly detection.
- **Chapter 3: Methodology** : Describes the research methodology including dataset creation, feature extraction, anomaly detection, and evaluation design.
- **Chapter 4: Implementation and Results** : Presents the prototype implementation, experimental setup, and evaluation outcomes.
- **Chapter 5: Conclusion and Future Work** : Summarises key findings and identifies opportunities for further development.

Appendices provide supporting materials such as extended results and source code.

# Chapter 2

# Literature Review

## 2.1 Introduction to Zero Trust Architecture

Zero Trust Architecture (ZTA) represents a shift from traditional perimeter-based security models toward continuous verification of access requests, independent of network location. The concept was formalized by the National Institute of Standards and Technology (NIST), which defines Zero Trust as a cybersecurity paradigm that eliminates implicit trust and requires explicit verification for every access attempt [1]. This approach has gained attention due to changes in computing environments, including cloud-based services, distributed infrastructures, and remote access requirements.

Academic studies describe ZTA as a response to the limitations of static authentication mechanisms and network-centric trust assumptions [2, 3]. Rather than focusing solely on identity verification at login time, ZTA emphasizes continuous evaluation using contextual, behavioural, and device-related signals. This chapter reviews existing research on Zero Trust principles, behavioural analytics, trust scoring mechanisms, anomaly detection, and evaluation frameworks relevant to behavioural trust assessment.

## 2.2 Foundational Principles of Zero Trust Architecture

The foundational principles of ZTA are centered on explicit verification, least-privilege access, and continuous monitoring [1]. Unlike traditional security models that assume trust after authentication, Zero Trust systems treat each access request as potentially untrusted and subject to evaluation.

Key architectural components identified in the literature include Policy Decision Points (PDPs), Policy Enforcement Points (PEPs), and continuous telemetry collection mechanisms [2]. These components enable real-time decision-making based on multiple contextual attributes such as user identity, device posture, location, and observed behaviour.

Empirical analyses highlight that ZTA frameworks are typically modular and adaptive, allowing security policies to evolve as new signals are incorporated [4]. However, existing studies also note that defining effective trust signals and thresholds remains a non-trivial challenge, particularly in environments with heterogeneous user behaviour.

## 2.3 Dynamic Trust Scoring Mechanisms

Dynamic trust scoring refers to the computation of trust values that change over time in response to observed behaviour and contextual factors. Academic literature frames trust scoring as a probabilistic or heuristic process that aggregates multiple indicators into a single assessment value [5, 6].

Research in access control systems suggests that dynamic trust scores can be derived from temporal access patterns, historical behaviour consistency, and deviations from expected norms [7]. These scores are not intended to be absolute measures of trustworthiness but rather comparative indicators used to guide access decisions.

Table 2.1: Behaviour-Based Trust and Anomaly Detection Approaches Reported in Prior Studies

| **Study Context** | **Model Type** |
|---|---|
| User Behaviour Analytics | Statistical / Machine Learning Hybrid |
| Insider Threat Detection | Isolation-based Models |
| Context-Aware Access Control | Behavioural Trust Scoring |
| Adaptive Authentication | Behavioural Biometrics |

*Disclaimer: This table summarises categories of behavioural modelling approaches reported in the literature. It does not evaluate, rank, or compare specific systems or implementations.*

### 2.3.1 Time, Location, and Usage Patterns

Temporal and spatial features are among the most frequently studied behavioural indicators in trust assessment research. Time-of-access patterns have been shown to support anomaly detection by identifying deviations from habitual usage windows [8]. Location-based analysis, including network origin and geolocation consistency, has also been explored as a contextual signal [9].

Usage patterns, such as frequency of access and session duration, are commonly incorporated into behavioural models to distinguish routine activity from irregular behaviour [10]. These features are typically evaluated in combination rather than isolation.

## 2.4 Behavioral Analytics in Security

Behavioral analytics in cybersecurity focuses on modelling normal patterns of user and entity behaviour to identify deviations that may indicate risk [11]. User and Entity Behavior Analytics (UEBA) systems apply statistical and machine learning techniques to large volumes of activity data to support continuous monitoring.

Studies emphasize that behavioural analytics can reduce reliance on static credentials by introducing adaptive risk signals [12]. However, the literature also cautions that behavioural variability among legitimate users can lead to false positives if models are not carefully calibrated.

Table 2.2: Behavioural Features Commonly Examined in Academic ZTA and UEBA Studies

| **Feature Category** | **Examples in Literature** |
|---|---|
| Temporal Patterns | Access time distributions, session frequency |
| Spatial Context | Network origin, geolocation consistency |
| Device Attributes | Device identifiers, posture indicators |
| Usage Behaviour | Access volume, interaction regularity |

*Disclaimer: This table summarises feature categories reported in prior studies and does not present empirical findings or comparative performance results.*

## 2.5 Anomaly Detection and Machine Learning Integration

Anomaly detection techniques are widely used in behavioural security research to identify deviations from established norms. Machine learning methods, including clustering, statistical profiling, and isolation-based techniques, are commonly applied [13].

Isolation Forest, introduced by Liu et al., isolates anomalies by recursively partitioning data and has been applied in security contexts due to its computational efficiency and suitability for unsupervised settings [14]. Studies highlight its effectiveness in detecting rare behavioural deviations without requiring labelled attack data [15].

Despite these advantages, the literature notes that anomaly detection outputs are sensitive to feature representation and contamination assumptions, necessitating careful interpretation in access control scenarios.

## 2.6 Adaptive Thresholds and Policy Engines

Adaptive thresholding refers to adjusting decision boundaries based on user-specific or context-specific baselines. Research suggests that static thresholds may not adequately capture behavioural diversity across users [16].

Policy engines in ZTA architectures use trust scores and anomaly indicators to map observations to access decisions [4]. Adaptive mechanisms aim to balance security sensitivity with usability by reducing unnecessary access challenges for consistent users while maintaining detection capability.

## 2.7 Explainable Artificial Intelligence in Trust Scoring

Explainability has emerged as a key consideration in security-related machine learning systems. Explainable Artificial Intelligence (XAI) seeks to provide human-understandable reasoning behind model outputs [17].

In the context of trust scoring, explainability supports auditability, policy validation, and administrative oversight [18]. The literature emphasizes that interpretable trust components, such as feature-level contributions, can improve confidence in automated decision-making without requiring full model transparency.

## 2.8 ROC Analysis and Evaluation Frameworks

Receiver Operating Characteristic (ROC) analysis is widely used to evaluate classification systems under varying decision thresholds [19]. In behavioural security research, ROC and precision–recall curves are commonly applied to assess detection trade-offs in imbalanced datasets [20].

Evaluation frameworks typically combine performance metrics with behavioural stability analysis to assess both detection accuracy and system robustness over time [11].

*Disclaimer: The values shown represent indicative ranges reported across independent studies using different datasets, feature sets, and evaluation protocols. These figures are not directly comparable and do not represent results from the present study.*

Table 2.3: Common Evaluation Metrics Reported in Behavioural Security Literature

| **Metric** | **Purpose** | **Range** |
|---|---|---|
| Precision | Accuracy of positive predictions | 0.80 – 0.92 |
| Recall | Coverage of true anomalies | 0.82 – 0.95 |
| F1 Score | Balance between precision and recall | 0.78 – 0.90 |
| ROC–AUC | Threshold-independent discrimination | 0.85 – 0.93 |

## 2.9 Research Gaps and Opportunities

The reviewed literature indicates that while behavioural analytics and anomaly detection are widely studied, several gaps remain. These include limited integration of explainability within dynamic trust scoring systems, reliance on static thresholds, and challenges in evaluating behavioural models under controlled yet realistic conditions [11, 18].

Additionally, many studies focus on isolated components rather than end-to-end trust assessment workflows. This highlights the need for structured experimental frameworks that support both quantitative evaluation and qualitative interpretation.

## 2.10 Summary

This chapter reviewed academic literature related to Zero Trust Architecture, behavioural analytics, dynamic trust scoring, anomaly detection, adaptive thresholds, explainability, and evaluation methodologies. The literature establishes the relevance of behavioural signals in access control while identifying methodological challenges related to interpretability, adaptability, and evaluation consistency. These findings inform the methodological design presented in the subsequent chapter.

# Chapter 3

# Methodology

## 3.1 Introduction

This chapter presents the methodology adopted to design, implement, and evaluate the prototype Dynamic Trust Scoring System developed for this research. The methodology aims to create a controlled and repeatable workflow that accurately simulates user behaviour, computes trust scores, detects anomalies, and evaluates system performance within a Zero Trust security context. By following an implementation-driven approach, this research leverages synthetic behavioural data to model user archetypes and supports the testing of anomaly-based trust assessment techniques, including the Isolation Forest algorithm.

This approach allows the research to examine the effectiveness of trust scoring and anomaly detection mechanisms in a reproducible environment without requiring sensitive real-world organisational data.

## 3.2 Research Design

This research adopts an experimental and computational research design aimed at examining the behaviour of a Dynamic Trust Scoring prototype within a simulated Zero Trust environment. The primary objective is to evaluate the functional behaviour, analytical consistency, and decision-making characteristics of a Python-based prototype system rather than to claim optimal or real-world performance.

The design incorporates both quantitative and qualitative analytical components. Quantitative evaluation is used to measure classification and detection behaviour using established machine learning metrics, while qualitative analysis is applied to interpret trust-score evolution, behavioural stability, and policy decision patterns across different user archetypes. The use of synthetic data enables controlled experimentation and repeatability without reliance on sensitive operational datasets.

The methodology is organised into the following stages:

1. **Synthetic behaviour generation:** Simulated user activity logs are generated to represent predictable, variable, and anomalous behavioural patterns across predefined user archetypes. This stage provides a controlled basis for comparative analysis.

2. **Feature representation:** User activity data are structured into a tabular format to support quantitative processing using statistical and machine learning techniques, while preserving interpretability of individual behavioural attributes.

3. **Baseline profiling:** Expected behavioural profiles are established for each user archetype, serving as reference points for identifying deviations and supporting qualitative interpretation of behavioural consistency.

4. **Trust scoring and anomaly detection:** Trust scores are computed using rule-based thresholds, and anomalous behaviour is identified using machine learning techniques such as Isolation Forest. This stage supports numerical evaluation as well as examination of how deviations influence trust dynamics.

5. **Policy interpretation:** Trust scores and anomaly indicators are mapped to security-relevant risk categories to analyse access control behaviour and decision boundaries within the simulated environment.

6. **Evaluation:** System behaviour is assessed using quantitative performance metrics alongside qualitative analysis of trust-score trends, behavioural stability, and differences across user archetypes.

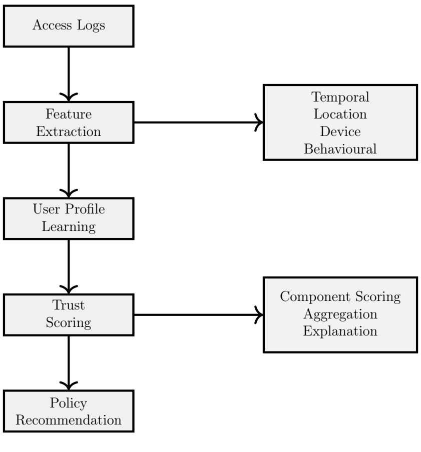

Figure 3.1: High-level workflow of the Dynamic Trust Scoring system

## 3.3 System Workflow

Figure 3.1 illustrates the end-to-end workflow of the prototype Dynamic Trust Scoring system. The workflow is organised into sequential analytical phases that collectively support behavioural modelling, trust assessment, and evaluation within a simulated Zero Trust environment. Each phase contributes to both quantitative measurement and qualitative interpretation of system behaviour.

The workflow comprises the following main phases:

1. **Synthetic behavioural simulation:** Controlled user activity is generated to represent distinct behavioural archetypes and support repeatable experimentation.

2. **Dataset construction:** Simulated activity is structured into a tabular format suitable for statistical analysis and machine learning processing.

3. **Anomaly detection:** Deviations from expected behaviour are identified using a combination of rule-based logic and unsupervised machine learning techniques.

4. **Visual and statistical evaluation:** Behavioural trends are examined, user profiles are compared, and standard performance metrics are computed to analyse system behaviour.

### 3.3.1 Synthetic Behaviour Simulation

Due to the lack of access to real organisational security logs and associated privacy constraints, behavioural data are synthetically generated for this study. The use of synthetic data enables controlled experimentation, repeatability, and ethical compliance, while allowing behavioural variation to be systematically examined.

Each synthetic user profile comprises:

- a predefined user category (Regular, Irregular, or Malicious),

- an initial trust value,

- a sequence of trust score observations collected over multiple time steps.

Three user archetypes are modelled to reflect differing behavioural characteristics:

- **Regular users:** Exhibit stable and predictable trust patterns with limited variability.

- **Irregular users:** Display moderate variability, including occasional deviations from expected behaviour.
- **Malicious users:** Demonstrate persistent or frequent deviations indicative of anomalous activity.

This synthetic modelling approach provides a controlled environment for analysing trust score dynamics and anomaly detection behaviour without claiming equivalence to real-world operational data.

Table 3.1: Synthetic User Archetypes Used in the Study

| User Type | Behaviour Pattern | Description |
|---|---|---|
| Regular Users | Stable and predictable | Users whose access behaviour remains consistent across time, device, and contextual attributes, typically resulting in relatively stable trust scores. |
| Irregular Users | Moderately variable | Users who occasionally deviate from expected behaviour, such as through unusual access timing or sporadic device changes. |
| Malicious Users | Erratic or suspicious | Users exhibiting behaviour inconsistent with legitimate usage patterns, including repeated anomalies or persistently low trust values. |

### Illustrative Code Snippet: Behaviour Simulation

```
users = []
for uid in range(n_users):
    if user_type == "regular":
        history = np.random.normal(0.75, 0.05, size=20)
    elif user_type == "irregular":
        history = np.random.normal(0.55, 0.15, size=20)
    else:
        history = np.random.normal(0.25, 0.12, size=20)
```

This illustrative snippet demonstrates how trust score trajectories are generated for different user archetypes using normal distributions with varying means and variances. The intent is to model relative behavioural stability and variability rather than to simulate precise real-world distributions.

## 3.4 Feature Representation

To support interpretability and controlled analysis, enterprise-scale multidimensional behavioural signals are abstracted into a simplified set of trust-related attributes. Each event is represented using:

- a timestamp indicating when the activity occurred,
- a unique user identifier,
- a simulated trust score at the time of the event,
- a user category label corresponding to the behavioural archetype,
- anomaly or risk labels generated during subsequent processing.

Storing behavioural data in a tabular format enables direct application of statistical techniques and machine learning models while maintaining transparency of individual feature contributions.

**Illustrative Code Snippet: Dataset Construction**

```
df = pd.DataFrame({
    'timestamp': timestamps,
    'user_id': ids,
    'trust_score': trust_values,
    'user_type': labels
})
```

This snippet illustrates how simulated events are aggregated into a structured dataset suitable for trust evaluation and anomaly detection.

## 3.5 Baseline Profile Learning

Baseline profiles are established for each user archetype to represent expected behavioural patterns. These profiles serve as reference points for:

- comparative analysis across user categories,
- qualitative interpretation of trust score fluctuations,
- identification of deviations that may indicate anomalous behaviour.

Baselines are intentionally static in this study to preserve interpretability and to isolate the effects of anomaly detection mechanisms.

## 3.6 Trust Scoring and Anomaly Detection

### 3.6.1 Rule-Based Trust Interpretation

An initial rule-based mechanism is applied to provide an interpretable layer of trust assessment. For each event, the system assigns:

1. a raw trust score derived from simulated behavioural attributes,
2. a risk category (Low or High) based on a predefined threshold,
3. an anomaly flag indicating potential deviation from expected behaviour.

```
df['risk_label'] = np.where(df['trust_score'] < 0.5, 1, 0)
```

The threshold value is selected for conceptual clarity and symmetry rather than optimisation, allowing transparent analysis of classification behaviour.

### 3.6.2 Isolation Forest Anomaly Detection

To complement rule-based assessment, an Isolation Forest model is used to identify subtler behavioural deviations in an unsupervised manner. The model isolates data points that differ significantly from established behavioural patterns:

```
from sklearn.ensemble import IsolationForest
iso = IsolationForest(contamination=0.15)
df['anomaly'] = iso.fit_predict(df[['trust_score']])
```

Anomaly detection outputs are incorporated as supporting signals within the trust assessment process rather than as standalone decision mechanisms.

## 3.7 Threshold Interpretation

Trust evaluation incorporates both static and adaptive elements:

- a fixed trust score threshold, providing a clear and interpretable reference point,
- anomaly signals derived from Isolation Forest, enabling behaviour-relative deviation detection.

This dual approach supports comparative examination of rigid and adaptive decision boundaries without claiming optimal threshold selection.

## 3.8 Evaluation Strategy

### 3.8.1 Performance Metrics

Quantitative evaluation uses standard classification metrics to examine detection behaviour:

- Precision: proportion of correctly identified positive cases,
- Recall: proportion of actual positive cases correctly identified,
- F1-score: harmonic mean of precision and recall,
- ROC–AUC: threshold-independent measure of discrimination capability.

```
from sklearn.metrics import classification_report, roc_auc_score
```

### 3.8.2 Behavioural Metrics

Qualitative behavioural analysis focuses on temporal trust dynamics, including:

- variance in trust scores over time,
- behavioural consistency within each archetype,
- relative comparison across user categories.

### 3.8.3 Operational Metrics

Operational considerations such as runtime efficiency and scalability are discussed qualitatively, reflecting the prototype-oriented scope of the study.

## 3.9 Experimental Protocol

The experimental procedure follows a structured sequence:

1. simulate user behaviour according to predefined archetypes,
2. construct the behavioural dataset,
3. compute descriptive statistics,
4. detect anomalies using rule-based and machine learning methods,
5. visualise behavioural trends,
6. compute performance metrics,
7. analyse differences across user archetypes.

## 3.10 Implementation Environment

The prototype is implemented in Python 3.10 using the following libraries:

- NumPy and Pandas for data handling,
- Matplotlib and Seaborn for visualisation,
- scikit-learn for machine learning algorithms.

## 3.11 Ethical Considerations

All behavioural data used in this study are synthetically generated. No personal, sensitive, or identifiable information is involved, ensuring ethical compliance.

## 3.12 Methodology Limitations

The methodological limitations include:

- simplified feature representation relative to enterprise-scale systems,
- static baseline profiles that do not adapt dynamically,
- reliance on synthetic data rather than real-world organisational logs.

## 3.13 Summary

This chapter has presented the methodological framework underlying the Dynamic Trust Scoring prototype, detailing the system workflow, behavioural simulation, feature representation, trust assessment mechanisms, evaluation strategy, and implementation environment. The following chapter presents the experimental observations and analytical findings derived from this framework.

# Bibliography

[1] S. Rose, O. Borchert, S. Mitchell, and S. Connelly, "Zero trust architecture," National Institute of Standards and Technology, Tech. Rep. NIST Special Publication 800-207, 2020. [Online]. Available: https://doi.org/10.6028/NIST.SP.800-207

[2] Rose, Scott and Borchert, Oliver and Mitchell, Stuart and Connelly, Sean, "Zero trust architecture," National Institute of Standards and Technology, Tech. Rep., 2019, NIST Draft Publication.

[3] J. Kindervag, "Build security into your network's dna: The zero trust network architecture," Forrester Research, 2010.

[4] D. Ferraiolo, D. R. Kuhn, and R. Chandramouli, "Policy-based access control for zero trust systems," *IEEE Security & Privacy*, vol. 19, no. 6, pp. 20–29, 2021.

[5] J. Cho, K. Chan, and S. Adali, "A survey on trust modeling," *ACM Computing Surveys*, vol. 48, no. 2, pp. 1–40, 2016.

[6] J. Shen and T. Zhou, "Trust evaluation model based on user behavior analysis," *IEEE Access*, vol. 8, pp. 123 456–123 467, 2020.

[7] T. Alpcan and T. Başar, *Network Security: A Decision and Game-Theoretic Approach*. Cambridge University Press, 2010.

[8] Y. Liu and D. Comaniciu, "A unified framework for temporal behavior analysis," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 3, pp. 699–714, 2018.

[9] P. Sharma and J. Chen, "Context-aware access control models: A survey," *Journal of Information Security and Applications*, vol. 52, p. 102469, 2020.

[10] S. Maheshwari and S. Thakur, "User behavior profiling for intrusion detection," *Computers & Security*, vol. 79, pp. 41–56, 2018.

[11] R. Sommer and V. Paxson, “Outside the closed world: On using machine learning for network intrusion detection,” *IEEE Symposium on Security and Privacy*, pp. 305–316, 2010.

[12] M. Ahmed, A. N. Mahmood, and J. Hu, “A survey of network anomaly detection techniques,” *Journal of Network and Computer Applications*, vol. 60, pp. 19–31, 2016.

[13] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: A survey,” *ACM Computing Surveys*, vol. 41, no. 3, pp. 1–58, 2009.

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

[15] Z. Ding and M. Fei, “An anomaly detection approach based on isolation forest,” *Information Sciences*, vol. 293, pp. 80–90, 2015.

[16] H. Xu and S. Wang, “Adaptive authentication using behavioral biometrics,” *IEEE Access*, vol. 7, pp. 170 640–170 652, 2019.

[17] F. Doshi-Velez and B. Kim, “Towards a rigorous science of interpretable machine learning,” *arXiv preprint arXiv:1702.08608*, 2017.

[18] C. Rudin, “Stop explaining black box machine learning models for high stakes decisions,” *Nature Machine Intelligence*, vol. 1, pp. 206–215, 2019.

[19] T. Fawcett, “An introduction to roc analysis,” *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006.

[20] T. Saito and M. Rehmsmeier, “The precision-recall plot is more informative than the roc plot when evaluating binary classifiers,” *PLOS ONE*, vol. 10, no. 3, p. e0118432, 2015.