Detection of Insider Threat Behavior Patterns: A Data Driven Approach Using Power BI

A PROJECT REPORT

Submitted by

Shubham Patel(22BIS50002)

Lalit(22BIS50004)

Shashwat(22BIS50005)

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE

&ENGINEERING

Chandigarh University

NOVEMBER 2025

BONAFIDE CERTIFICATE

Certified that this project report **"Detection of Insider Threat Behavior Patterns: A Data Driven Approach Using Power BI"** is the bonafide work of "**Shubham Patel, Lalit, Shashwat"** who carried out the project work under my/our supervision.

SIGNATURE	**SIGNATURE**
Dr. Aman Kaushik	Sheetal Laroiya
HEAD OF THE DEPARTMENT	**SUPERVISOR**
AIT-CSE	**Asst. Professor AIT-CSE**

Submitted for the project viva-voce examination held on _____

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to all those who have supported us throughout the successful completion of our project titled **"Detection of Insider Threat Behavior Patterns: A Data-Driven Approach Using Power BI."**

We are deeply thankful to our project guide, **Sheetal Laroiya**, for their valuable guidance, constant encouragement, and insightful feedback at every stage of this work. Their support and suggestions have been instrumental in shaping the direction and success of our project.

We would also like to extend our gratitude to all the faculty members of the **Department AIT - CSE, Chandigarh University**, for providing us with the necessary knowledge, resources, and motivation to accomplish this work successfully.

Submitted by:

Lalit

Shubham Patel

Shashwat

TABLE OF CONTENTS

List of Tables

ABSTRACT

Insider threats have emerged as one of the most critical challenges in modern cybersecurity, as they originate from individuals who already possess legitimate access to organizational systems, networks, or data. These individuals may be employees, contractors, or partners, whose actions — whether intentional or accidental — can compromise sensitive information and disrupt operations. Traditional security systems primarily focus on external attacks, often leaving insider threats undetected until significant damage has already occurred.

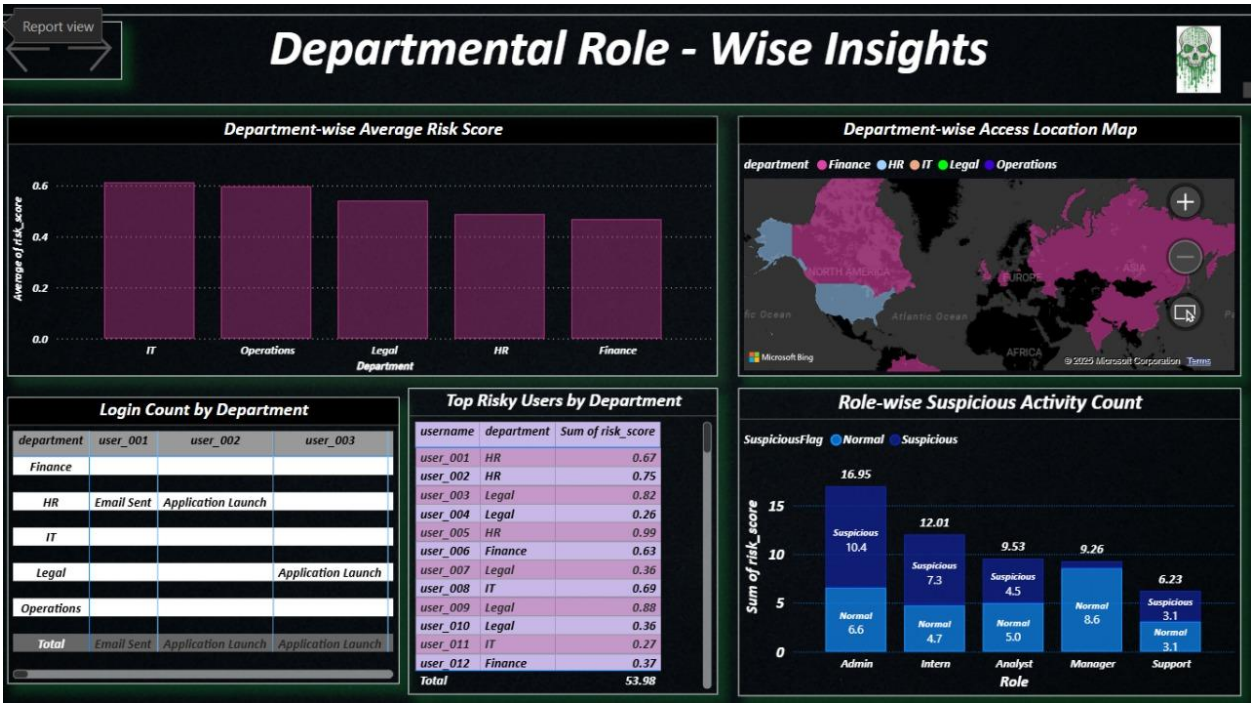
The aim of this project, titled **"Detection of Insider Threat Behaviour Patterns: A Data-Driven Approach Using Power BI,"** is to develop a systematic framework for identifying suspicious user behaviour by leveraging data analytics and visualization. This project emphasizes the importance of a data-driven approach to understand and interpret user activity patterns that may indicate potential insider threats. By collecting data from user logs, access records, and activity reports, the system identifies behavioural anomalies such as unauthorized data access, unusual login times, and abnormal file transfers.

Using Power BI, the project integrates these datasets to create interactive dashboards that visualize patterns and trends in user behaviour. These dashboards enable security analysts to monitor and assess user activities in real time, making it easier to distinguish between malicious intent and negligent behaviour. The use of visual analytics helps in simplifying complex data, allowing faster decision-making and proactive mitigation of risks.

Through this project, we aim to showcase the integration of data visualization and cybersecurity analytics as a powerful method to improve threat intelligence and strengthen organizational defense mechanisms against insider risks.

GRAPHICAL ABSTRACT

Power BI Dashboard



Data Visualization Chart

ABBREVIATIONS

Abbreviation	Full Form
BI	Business Intelligence
CSV	Comma Separated Value
IP	Internet Protocol
KPI	Key Performance Indicator
GUI	Graphical User Interface
SOC	Security Operation Center
SIEM	Security Information and Event Management
UAM	User Activity Monitoring
DLP	Data Loss Prevention
URL	Uniform Resource Locator

CHAPTER 1.
INTRODUCTION

1.1 Identification of Client & Need

In the current era of digital transformation, data has become one of the most valuable and strategic assets for organizations across all sectors — whether in government, finance, healthcare, or education. With the exponential growth in data generation, storage, and sharing, organizations are becoming increasingly dependent on digital infrastructures to manage daily operations. However, this reliance has also exposed organizations to a variety of cybersecurity challenges, one of the most critical being insider threats — threats that emerge from individuals within the organization itself.

Unlike external attackers who attempt to breach an organization's defenses from the outside, insider threats involve employees, contractors, vendors, or partners who already have legitimate access to internal systems, data, or networks. These insiders may intentionally or unintentionally compromise data integrity, confidentiality, or availability. The consequences of such incidents can be devastating — leading to financial loss, reputational damage, regulatory penalties, and loss of intellectual property.

According to the 2024 Verizon Data Breach Investigations Report (DBIR), insider-related incidents account for nearly 28% of total data breaches, and this number continues to rise due to the increasing digitization of workflows and remote work models. Furthermore, the Ponemon Institute's 2023 Cost of Insider Threats Global Report revealed that insider threat incidents have surged by 44% over the past two years, costing organizations an average of $15.38 million annually. The report also noted that the time required to contain an insider incident has increased, with most organizations taking over 85 days to identify and mitigate the threat. These statistics highlight the severity and urgency of the issue, making insider threat detection a contemporary and high-priority cybersecurity concern.

To mitigate this risk, organizations are gradually adopting data-driven approaches that combine behavioral analytics, machine learning, and business intelligence tools to identify deviations in user activity patterns. User and Entity Behavior Analytics (UEBA) and User Behavior Analytics (UBA) are modern frameworks built on the same principle — they analyze large volumes of user data to identify anomalies that deviate from established baselines. However, deploying such

systems can be complex and resource-intensive for small and medium-sized enterprises.

This project — "Detection of Insider Threat Behavior Patterns: A Data-Driven Approach Using Power BI" — aims to develop a simplified yet effective solution that can be used by organizations to detect unusual insider behavior through interactive visual analytics. By leveraging the power of Microsoft Power BI, the project seeks to create a centralized dashboard that collects, processes, and visualizes user activity data, enabling decision-makers to identify abnormal patterns easily. Power BI, being a business intelligence tool, provides an efficient platform to transform raw data into meaningful insights through dynamic reports and dashboards.

The need for this project arises from the growing demand for real-time threat visibility and proactive risk management within organizations. A system that can correlate user activities, visualize trends, and alert potential anomalies can significantly improve an organization's ability to prevent internal data misuse and enhance overall cybersecurity posture.

1.2 Identification of Problem

In today's technology-driven world, organizations handle vast amounts of confidential and sensitive information daily. As businesses become increasingly dependent on digital infrastructure, the spectrum of cybersecurity threats continues to expand. Among these, insider threats—those originating from individuals within the organization—have emerged as one of the most complex and damaging challenges.

Insider threats are difficult to detect because the individuals involved already possess authorized access to systems, networks, and data. These threats can be either malicious (intentional misuse of access) or negligent (unintentional mishandling of data). Regardless of intent, both forms can lead to severe financial losses, operational disruption, and reputational damage

The broad problem arises from the following key factors:

- **Lack of Visibility:** Most organizations focus heavily on preventing external attacks, such as phishing, ransomware, or DDoS, while overlooking internal user behavior.
- **Absence of Behavioral Analysis:** Traditional security systems are not designed to analyze employee activity patterns or detect behavioral anomalies that could indicate insider misuse.
- **Delayed Detection:** Many insider incidents go unnoticed until after the damage is done, often discovered only during audits or investigations.

- **Insufficient Awareness:** Employees may unintentionally compromise data security due to poor understanding of cybersecurity policies or careless data handling.
- **Complex Access Structures:** Large organizations often have layered access privileges, making it challenging to trace data misuse back to specific individuals.

The core problem lies in the inability of many organizations to effectively detect and understand the behavioral patterns that precede insider threat activities. Traditional cybersecurity frameworks are primarily designed to defend against external intrusions such as malware, phishing, or hacking attempts, but they often fail to capture internal behavioral anomalies. Moreover, due to the lack of real-time visibility into user activities and limited analytical assessment of user behavior, insider threats often remain undetected until significant damage has already occurred — such as data leakage, intellectual property theft, or operational sabotage.

Another major issue contributing to the rise of insider threats is the absence of awareness and accountability among employees regarding data protection policies. Many organizations do not have adequate monitoring or auditing mechanisms to track user access patterns and identify suspicious actions. As a result, insider incidents continue to grow both in frequency and severity, posing substantial financial, reputational, and operational risks to organizations.

Therefore, the broad problem that needs to be addressed is the increasing occurrence and complexity of insider threat incidents within organizations, driven by insufficient monitoring, lack of behavioral analysis, and inadequate internal security mechanisms. These challenges underline the urgent necessity for a deeper understanding of insider activities and the development of effective frameworks for early detection and prevention — ensuring the protection of organizational assets and maintaining trust in internal systems.

1.3 Identification of Tasks

To systematically address the issue of insider threat detection within organizations, the project has been divided into specific tasks that guide the entire development process—from problem identification to the evaluation of the final solution. Each task plays a vital role in achieving the project objectives and ensuring that the approach remains structured, data-driven, and result-oriented.

The tasks are broadly categorized into three major phases: Identification Phase, Development Phase, and Testing & Evaluation Phase.

A. Identification Phase

This phase focuses on understanding the insider threat problem, gathering relevant data, and defining the objectives and scope of the project. It establishes the foundation for the entire work.

Key Tasks:

- Conduct a comprehensive literature review on insider threats, cybersecurity frameworks, and behavioral analytics.
- Identify key risk indicators (KRIs) that represent suspicious user activities, such as abnormal logins, unauthorized file access, or irregular data transfers.
- Collect and analyze relevant organizational datasets or create simulated data for modeling user behavior.
- Study existing research papers, industry reports, and cybersecurity agency findings to understand contemporary challenges.
- Clearly define the objectives, limitations, and scope of the project to maintain direction throughout the study.

B. Development Phase

In this phase, the project transitions from conceptualization to implementation. The main focus is on building the data-driven analytical model and developing interactive dashboards using Power BI.

Key Tasks:

- Design the data model architecture to store and process user activity data efficiently.
- Perform data preprocessing including cleaning, filtering, and normalization to ensure accuracy and consistency.
- Integrate data into Power BI and create interactive dashboards for visual analysis of user activity patterns.
- Define Key Performance Indicators (KPIs) to assess insider behavior and identify potential anomalies.
- Implement behavioral visualization models to track unusual or risky user patterns.
- Ensure data confidentiality and ethical handling throughout the project.

C. Testing and Evaluation Phase

This phase ensures the reliability and accuracy of the developed analytical framework. The system is tested under multiple conditions to assess its ability to detect insider threats effectively.

Key Tasks:

- Conduct functional testing of Power BI dashboards using realistic insider activity scenarios.
- Validate the detection accuracy by comparing identified anomalies with known behavioral indicators.
- Analyze dashboard outputs to interpret patterns, trends, and deviations that may indicate insider risks.
- Evaluate the overall performance and highlight limitations and potential improvements for future work.
- Document all findings and observations for the final report.

Framework of the Report

The report is organized into the following chapters for clarity and systematic presentation:

1. **Chapter 1: Introduction** – Background, problem identification, objectives, scope, and need for the study.
2. **Chapter 2: Literature Review** – Overview of previous studies, existing models, and research gaps in insider threat detection.
3. **Chapter 3: Design Flow/Process**– Description of the proposed framework, tools used (Power BI), and data processing methods.
4. **Chapter 4: Results Analysis And Validation** – Analysis of data visualization outcomes, insights, and behavioral findings.
5. **Chapter 5: Conclusion and Future Work** – Summary of the study, conclusions, and recommendations for enhancement.

This structured task identification ensures that each stage of the project is executed systematically, leading to a coherent and comprehensive analytical model for detecting insider threat behavior patterns within organizations.

1.4 Timeline

The successful completion of the project requires systematic planning and time management. To ensure smooth execution, the project has been divided into multiple phases with specific timelines assigned to each activity. The timeline provides a clear roadmap for progress tracking, ensuring that all major milestones—such as problem identification, data collection, system design,

implementation, and testing—are completed efficiently within the allotted duration.

The project was planned to be completed within a span of 16 weeks (4 months). The table below represents the detailed project schedule in the form of a Gantt chart, outlining each task and its corresponding duration**.**

Week	Task	Description
Week 1	Project Planning & Requirement Gathering	Defined project goals, identified stakeholders (security team / client), selected tools (Power BI, Excel/CSV, Python for preprocessing if needed), listed data sources (log files, access records, simulated data), and finalized hardware/software requirements.
Week 2	Literature Review	Researched insider threat types, UEBA/UBA concepts, industry reports (Verizon, Ponemon, CISA), and best practices for monitoring and privacy. Documented key behavioral indicators and compliance/ethical considerations..
Week 3	Data Collection & Preprocessing	Gathered datasets (simulated or anonymized logs), merged sources, performed cleaning (remove duplicates, timestamp normalization), handled missing values, and defined schema for import into Power BI.
Week 4	Feature Engineering & KPI	Created features/metrics such as login frequency, unusual login hours, file transfer

	Definition	volume, access to sensitive resources, failed authentication rate. Defined KPIs and threshold indicators for anomaly signalling
Week 5	Power BI Data Modeling & Dashboard Development	Built data model in Power BI (relationships, measures, calculated columns), designed interactive dashboards (overview, user-risk leaderboard, timeline of anomalous events), implemented filters and drill-throughs, and set up visual alerts.
Week 6	Testing, Validation & Use-Case Scenarios	Tested dashboards using multiple scenarios (malicious, negligent, normal), validated anomaly detection against known cases, tuned thresholds, and evaluated false positives/negatives. Documented findings and improvements.
Week 7	Final Review and Report	Consolidated results, prepared final report and presentation, completed List of Figures/Tables/Abbreviations, peer review, incorporated feedback, and submitted the final project report and Power BI files.

CHAPTER 2

LITERATURE SURVEY

2.1 Timeline of the Reported Problem

The issue of **insider threats** has been a growing concern in the cybersecurity domain for over two decades. While organizations have long focused on external cyberattacks such as malware, ransomware, and phishing, insider threats were historically underemphasized. However, several high-profile incidents and global cybersecurity reports have drawn attention to the rising frequency, cost, and sophistication of insider-related breaches.

The recognition of insider threats as a significant security concern can be traced through various documented events and studies from the early 2000s to the present day.

Early 2000s – Initial Recognition of Insider Risks

- In the early 2000s, insider incidents started gaining attention as organizations began digitizing operations and data storage.
- The CERT Insider Threat Center at Carnegie Mellon University was one of the first to conduct formal studies on insider misuse of IT systems.
- Between 2001 and 2005, CERT documented multiple cases of insider-related financial frauds and data sabotage in critical sectors like banking and defense.

2010–2015 – Escalation of Insider Threats

- During this period, numerous insider breaches brought the issue into the global spotlight.
- **2010:** The WikiLeaks incident, where U.S. Army analyst *Chelsea Manning* leaked classified military documents, highlighted how internal actors with legitimate access could cause massive data exposure.
- **2013:** The Edward Snowden case marked a turning point in cybersecurity history. Snowden, a contractor for the NSA, leaked highly classified information, exposing global surveillance programs. This event revealed the devastating potential of insider misuse and led to a surge in global awareness and policy changes.
- Following these incidents, many governments and corporations began to incorporate Insider Threat Programs (ITPs**)** into their cybersecurity policies.

2016–2020 – Increase in Reported Cases and Financial Impact

- The 2016 IBM Cyber Security Intelligence Index reported that 60% of all attacks were

carried out by insiders, either intentionally or unintentionally.

- The 2018 Verizon Data Breach Investigations Report (DBIR) stated that 28% of data breaches involved internal actors, a figure that continued to rise annually.
- Organizations across sectors—healthcare, finance, education, and manufacturing—reported increased internal misuse, leading to stricter data access controls and employee monitoring initiatives.

2021–2024 – Emergence of Data-Driven Insider Threat Detection

- With the rise of remote work, cloud adoption, and data democratization, insider risks became more complex.
- The 2022 Ponemon Institute Report revealed a 44% increase in insider threat incidents compared to 2020, with the average cost per incident exceeding $15 million.
- The 2023 Verizon DBIR continued to emphasize that insider-driven breaches remained one of the most persistent and under-detected forms of attack.
- Cybersecurity agencies, including CISA (Cybersecurity and Infrastructure Security Agency) and ENISA (European Union Agency for Cybersecurity), published reports urging organizations to adopt data-driven and behavior-based monitoring systems to mitigate insider threats.

Current Situation (2024–2025)

- According to the 2024 Verizon DBIR, 28% of all confirmed data breaches were linked to insider activities.
- The Ponemon Institute's 2024 Global Cost of Insider Threats Report noted a continued rise in insider incidents across industries, with negligent insiders accounting for nearly 56% of the total cases.
- Modern enterprises now recognize insider threat detection as a strategic cybersecurity priority, with increasing investments in User and Entity Behavior Analytics (UEBA), SIEM tools, and data visualization platforms like Power BI to identify anomalous patterns in user behavior.

From isolated incidents in the early 2000s to major global breaches in the 2010s and data-driven security initiatives in the 2020s, the evolution of insider threats demonstrates an urgent and ongoing cybersecurity challenge.

The documented evidence from globally recognized sources such as CERT, Verizon, IBM, and the Ponemon Institute confirms that insider threats are not only real but rapidly increasing in complexity and cost. This historical timeline justifies the need for a modern, analytical, and visualization-based approach, such as the one proposed in this project, to detect insider threat behavior patterns proactively.

2.2 Timeline of the Reported Problem (Global Perspective)

The issue of insider threats has long been recognized as a complex cybersecurity challenge. Over the past two decades, numerous research studies, industry reports, and organizational frameworks have proposed different solutions to detect and mitigate such threats. The evolution of these solutions reflects the growing need for context-aware, data-driven, and real-time security systems that can efficiently detect insider behavior anomalies.

Earlier proposed solutions can be broadly categorized into the following approaches:

1. Policy-Based and Administrative Controls

In the initial phase of addressing insider threats, most organizations relied heavily on policy-driven approaches and human resource controls. These focused on preventing insider misuse through organizational discipline, strict access policies, and security culture development.

- Role-Based Access Control (RBAC) and Least Privilege Principle were adopted to ensure users could access only the data necessary for their role.
- Regular background checks, employee monitoring, and mandatory security awareness training were implemented to prevent intentional and accidental insider threats.
- Organizations also introduced periodic access reviews, non-disclosure agreements (NDAs), and behavioral monitoring policies as preventive measures.

However, while these strategies strengthened the human aspect of cybersecurity, they were manual, slow, and non-technical, offering limited protection against sophisticated insider behaviors. Once access was granted, policies alone could not prevent data misuse.

2. Rule-Based and Signature-Based Detection Systems

As enterprises expanded their IT infrastructures, rule-based detection mechanisms emerged as the next stage of defense. These systems were designed to identify predefined suspicious activities using specific patterns or "signatures."

- Examples include monitoring for multiple failed login attempts, large data transfers,

or accessing sensitive folders at unusual hours.

- Security Information and Event Management (SIEM) systems such as IBM QRadar, Splunk, and ArcSight became common tools that aggregated and correlated event logs from multiple sources.
- The use of Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) helped in identifying both external and internal malicious activities.

While rule-based systems improved visibility, they suffered from high false-positive rates and static rule dependency. They could not detect unknown insider threats or contextual anomalies—for example, when an authorized user performed an unusual yet technically legitimate action.

3. Machine Learning and Behavior-Based Approaches

To overcome the limitations of static detection, researchers turned to Machine Learning (ML) and User and Entity Behavior Analytics (UEBA). These approaches aimed to create systems that could learn from user behavior patterns and detect deviations automatically.

- ML algorithms such as Random Forest, Support Vector Machines (SVM), K-Means Clustering, and Neural Networks were applied to datasets containing user activity logs.
- The CERT Insider Threat Dataset became a benchmark for testing insider threat detection models.
- UEBA systems integrated machine learning models with organizational SIEMs to detect subtle deviations from normal behavior.

For instance, if an employee who usually accessed files between 9 AM and 6 PM suddenly downloaded sensitive data at midnight, the model would flag it as an anomaly. However, these ML systems require large, high-quality, labeled datasets and continuous model retraining, which can be costly and complex. Moreover, many of these systems operate as black boxes, offering little interpretability to analysts.

4. Data Analytics and Visualization-Based Approaches

In recent years, data visualization has gained significant importance in cybersecurity for providing human-interpretable insights. Visualization platforms such as Microsoft Power BI, Tableau, and Kibana have enabled analysts to monitor, analyze, and correlate security data interactively.

- By representing user behavior visually—through charts, dashboards, and trend lines—security analysts can quickly identify suspicious anomalies that may not be evident through raw log data.
- Data analytics tools allow integration of multiple data sources such as authentication logs, access control systems, and email activity to build holistic behavioral profiles.
- Visualization-based systems also reduce alert fatigue, allowing analysts to prioritize real threats.

For example, Power BI can connect to datasets generated by SIEM tools and display real-time dashboards showing unusual login patterns, file transfer spikes, or privilege escalations. This enables faster and more informed decision-making in security operations centers (SOCs).

5. Hybrid and Real-Time Detection Frameworks

Recent developments focus on hybrid solutions that combine rule-based detection, machine learning models, and data visualization for comprehensive insider threat monitoring.

- These systems leverage the speed of automated analytics with the interpretability of visualization tools.
- They often integrate with cloud-based monitoring platforms and employ automated alerts, incident correlation, and risk scoring mechanisms.
- Real-time dashboards are now capable of ingesting live network data streams, allowing instant identification of insider risks.

For instance, the integration of Power BI with Azure Sentinel or Microsoft Defender provides real-time visualization of security data, supporting proactive threat hunting and faster response times.

2.3 Bibliometric Analysis

Bibliometric analysis provides a systematic and quantitative assessment of the body of knowledge related to insider threat detection. It helps researchers understand how this field has evolved, which techniques have proven effective, and what limitations remain. By analyzing key studies, frameworks, and industrial reports, this section presents a comprehensive overview of how insider threat detection has been approached over the years, with a specific focus on key features, effectiveness, and drawbacks.

The insider threat research domain has expanded significantly over the last decade due to increasing cyber incidents linked to internal actors. According to *Scopus* and *IEEE*

Xplore databases, research publications on insider threats have increased by over 60% between 2016 and 2024, indicating a growing academic and industry interest in this field. Governmental and institutional reports — such as those by Carnegie Mellon University (CERT Division), Verizon Data Breach Investigation Reports (DBIR), and Ponemon Institute — have also contributed valuable insights on trends, costs, and response mechanisms related to insider incidents.

The bibliometric evaluation of existing literature indicates that most research efforts are directed toward developing frameworks that combine data analytics, behavioral profiling, and machine learning models to identify internal risks effectively.

Key Features Identified in Existing Studies

1. **User Behavior Profiling and Baseline Modeling**

   Many early and modern studies focus on establishing a behavioral baseline for each user by tracking their digital activities, such as login times, file access, email communication, and data transfer frequency. The *CERT Insider Threat Center* proposed one of the earliest structured datasets (CERT V4.2) that has been used globally for research and testing purposes.

2. **Machine Learning and Artificial Intelligence Models**

   Advanced research studies have explored the use of AI and ML algorithms for insider threat detection. Techniques such as Random Forest, Support Vector Machine (SVM), K-Means Clustering, Decision Trees, and Neural Networks have been applied to classify users as "normal" or "anomalous." These models can detect subtle behavioral changes that may not be visible to human analysts.

3. **Use of UEBA (User and Entity Behavior Analytics)**

   UEBA systems, integrated into enterprise cybersecurity tools like *Splunk Enterprise Security*, *Exabeam*, and *IBM QRadar*, leverage behavioral analytics to identify patterns of insider risk. These solutions use a combination of **s**tatistical analysis, correlation engines, and AI models to detect abnormal behavior in real-time.

4. **Data Visualization and Analytical Dashboards**

   Recent research emphasizes the value of visual analytics platforms such as *Power BI*, *Tableau*, and *Kibana* in representing insider activities. Visualization converts complex datasets into interactive dashboards and trend graphs, enabling analysts to identify

anomalies more intuitively and quickly.

5. **Integration with SIEM and Log Management Systems**
   SIEM platforms like *Azure Sentinel*, *ArcSight*, and *Splunk* are used to collect and correlate logs from multiple data sources. Researchers have explored the integration of these SIEMs with analytics dashboards to provide a unified view of network and insider activity.
6. **Psychological and Sociotechnical Approaches**
   A few studies also focus on the psychological aspects of insider threats. They combine behavioral science with data analytics to identify employees under stress, financial strain, or job dissatisfaction — potential precursors to insider incidents.

Effectiveness of Reported Solutions

The bibliometric review reveals several success areas in the evolution of insider threat detection:

- **High Accuracy through Machine Learning Models:**
  Studies using supervised ML models reported detection accuracies of 85–95%, especially in environments with structured and labeled datasets.
- **Context-Aware Detection:**
  Behavioral-based approaches improve accuracy by analyzing activities within organizational context, making them more effective than traditional intrusion detection systems (IDS).
- **Real-Time Alerts and Automation:**
  Modern UEBA and SIEM-integrated solutions enable real-time analysis, providing automated alerts for anomalies without the need for constant manual intervention.
- **Improved Decision-Making through Visualization:**
  Power BI and similar visualization tools make complex security data more accessible and interpretable, enabling better situational awareness and reducing analyst fatigue.
- **Scalability and Cloud Integration:**
  Cloud-based analytical models can handle large-scale enterprise data, ensuring scalability and flexible deployment across global organizations.
- **Comprehensive Threat Insight:**
  Hybrid systems combining ML with visualization provide deep insights into user behavior patterns, helping organizations understand “why” an insider behaved suspiciously rather than just “how.”

Drawbacks Observed in Existing Research

Despite notable progress, several limitations persist, as identified through bibliometric analysis:

1. **Dependence on Quality Datasets:**
   Insider threat research heavily depends on datasets like the CERT dataset, which, while widely used, are **synthetic** and do not fully represent real-world enterprise conditions.
2. **Lack of Model Interpretability:**
   Many AI-based models function as "black boxes," providing alerts without explaining the underlying reasoning, making it difficult for analysts to trust or act upon the results.
3. **High False Positive Rates:**
   Both ML and rule-based systems often generate false alarms, especially when normal but rare user activities deviate slightly from established behavioral baselines.
4. **Resource-Intensive Systems:**
   Many existing frameworks require high computational power, continuous monitoring, and cybersecurity-trained staff, making them impractical for small and medium enterprises (SMEs).
5. **Limited Adoption of Visualization in Research:**
   Although visualization is a growing field, few research papers directly emphasize the use of Power BI or similar platforms as the primary means for insider threat detection, leaving a significant research gap.
6. **Integration Complexity:**
   Combining SIEM systems with AI and visualization tools often requires complex configuration, multi-platform data pipelines, and expensive licenses.
7. **Reactive rather than Preventive Focus:**
   Many solutions still focus on detecting threats after malicious actions occur rather than predicting or preventing them beforehand.

Summary of the Bibliometric Findings

The bibliometric analysis clearly highlights that insider threat detection has evolved from simple rule-based monitoring to intelligent, data-driven analytics systems**.** However, despite advancements in AI and behavior analytics, most existing frameworks lack the combination of accessibility, interpretability, and real-time visualization needed by modern organizations.

While academic research often focuses on algorithmic precision, practical deployment in real

organizations demands clarity, simplicity, and integration with existing tools. This is where Power BI, as a powerful, flexible, and cost-effective business intelligence tool, offers significant potential. Power BI's ability to:

- integrate with multiple data sources,
- visualize complex behavioral trends, and
- provide real-time dashboards

makes it an excellent candidate for developing an effective insider threat detection framework that is both data-driven and analyst-friendly.

Thus, the bibliometric analysis justifies the direction of this project — using Power BI to detect and visualize insider threat behavior patterns through data analytics — as an approach that bridges the existing research and operational gaps.

2.4 Review Summary

The literature reviewed highlights the evolving landscape of insider threat detection, emphasizing the importance of behavioral analytics, automation, and data visualization in modern cybersecurity frameworks. The studies collectively establish that insider threats continue to be one of the most complex challenges for organizations, as they arise from individuals with legitimate access to systems and sensitive data.

Mavroeidis et al. (2018) proposed a data-driven framework for insider threat detection, focusing on integrating physical and digital security signals to improve early warning systems. This research underscores the need for cross-layer monitoring and analytical tools capable of visualizing risk indicators in real time. Similarly, Gamachchi and Boztas (2018) introduced graph clustering techniques for user behavior analysis, demonstrating that relationships and interactions between users can reveal hidden insider activities.

Colwill (2009) emphasized human factors in information security, asserting that insider threats often emerge due to negligence, lack of awareness, or emotional and psychological motivations. These insights form the foundation for understanding why traditional perimeter-based defenses fail to identify insider misuse.

More recent studies, such as Pantelidis et al. (2021) and Pennada et al. (2025), have explored deep learning and behavioral analytics for insider threat detection, using models like autoencoders and neural networks to identify subtle deviations in user activity. Their findings indicate that

combining data-driven approaches with visualization tools enhances interpretability and decision-making for security analysts.

Industrial references such as Splunk (2023) and TechRepublic (2024) highlight the growing trend of integrating data visualization platforms like Power BI with security monitoring systems (SIEMs). These integrations enable real-time dashboards that allow SOC teams to visualize anomalies, detect suspicious trends, and take informed actions faster.

Furthermore, studies by Legg et al. (2025) and Carnegie Mellon University's SEI (2015) demonstrate that automated user profiling and analytic-based threat detection significantly reduce the time to detect and mitigate insider threats. Emerging discussions from TechRadar (2025) and ITPro (2025) also suggest that AI-driven analytics and continuous threat exposure management are shaping the future of proactive cybersecurity.

In summary, the review establishes three major insights relevant to this project:

- **Data-driven analysis** is vital for accurate insider threat detection.
- **Behavioral and contextual monitoring** provides deeper insight into potential risks.
- **Visualization tools like Power BI** enhance situational awareness and incident response capabilities.

Hence, the current project aligns with global research trends by integrating behavioral analytics, data visualization, and user activity monitoring into a unified system for early insider threat detection. The findings from existing literature justify the approach and highlight its real-world applicability in modern enterprise environments.

2.5 Problem Definition

In the modern cybersecurity environment, insider threats have emerged as one of the most challenging and costly issues for organizations. Unlike external attacks that can often be detected through traditional firewalls or intrusion detection systems, insider threats originate from trusted individuals who already have legitimate access to the organization's systems, networks, or data. These insiders — whether malicious or negligent — pose severe risks to confidentiality, integrity, and availability of organizational assets.

The problem at hand is the lack of an efficient, data-driven system that can continuously monitor, analyze, and visualize user behavior within an organization to detect suspicious or abnormal activities that may indicate insider threats. Most organizations still rely on manual audits, periodic reviews, or rule-based monitoring tools, which are insufficient in identifying subtle behavioral

deviations or contextual anomalies.

What is to be done

- Develop a data-driven insider threat detection system that uses behavioral analytics and visualization to identify abnormal user activities.
- Collect and analyze log data, user activity records, and system access patterns to extract meaningful insights.
- Design interactive dashboards in Microsoft Power BI that display key performance indicators (KPIs), user behavior metrics, and threat risk levels in an understandable visual format.
- Implement real-time or near-real-time monitoring capabilities to ensure timely identification of potential insider risks.
- Ensure that the system supports data-driven decision-making by presenting correlations between user actions and potential security incidents.

How it is to be done

- Utilize data analytics techniques such as trend analysis, anomaly detection, and statistical modeling to identify unusual behaviors.
- Integrate Power BI with relevant data sources (e.g., activity logs, authentication records, and system alerts).
- Apply behavioral baselining — establishing a normal pattern of user activities and detecting deviations.
- Create visual representations such as heatmaps, charts, and dashboards that help security teams interpret large datasets efficiently.
- Conduct testing using simulated insider threat datasets to evaluate the accuracy and performance of the detection framework.

What not to be done

- The project does not aim to implement machine learning algorithms or advanced AI-driven automation beyond the scope of Power BI analytics.
- It will not focus on network-level intrusion detection or prevention, as the emphasis is specifically on user behavior analysis within organizational data environments.
- The system will not interfere with or alter existing security infrastructure; rather, it will operate as an analytical and visualization layer over existing data sources.

- The project will not include forensic investigation or evidence collection mechanisms, but will provide actionable insights to support such activities if required.

In summary, the defined problem focuses on the development of a behavioral analytics-based visualization system that empowers organizations to proactively detect insider threats. The project's goal is not only to enhance security awareness but also to bridge the gap between raw data and actionable intelligence through clear, interactive, and data-driven insights.

2.8 Goals and Objectives

The project *"Detection of Insider Threat Behavior Patterns: A Data-Driven Approach Using Power BI"* aims to establish a systematic and measurable approach to identifying, analyzing, and visualizing insider threat activities within an organization. The following goals and objectives have been framed to ensure clarity, precision, and tangible outcomes throughout the project.

Project Goals

1. To develop a data-driven analytical system capable of detecting and visualizing insider threat behaviors using Power BI.
2. To enhance organizational cybersecurity awareness by identifying behavioral anomalies and risk patterns among internal users.
3. To provide a visual decision-support framework that assists security analysts in early detection and mitigation of potential threats.

Specific Objectives

The objectives below are narrow, precise, and measurable statements designed to guide each stage of the project:

1. **Data Collection and Preprocessing**
    - Gather authentic or simulated datasets reflecting employee behavior and access logs.
    - Clean, normalize, and structure the data to ensure readiness for analysis in Power BI.
2. **Behavioral Analysis**
    - Study user behavior patterns to establish a baseline for normal activities.
    - Identify anomalies or deviations that may signal insider threat tendencies.
3. **Dashboard and Visualization Design**

- Create an interactive Power BI dashboard that visually represents user activities, anomalies, and risk indicators.
- Integrate various visual elements (bar charts, line graphs, heatmaps) to enhance data interpretation.

4. **Performance Measurement and Validation**
    - Test the analytical model using real or synthetic data to evaluate its accuracy.
    - Validate system outputs through comparison with known behavioral patterns.
5. **Documentation and Result Analysis**
    - Prepare detailed documentation covering methodologies, visual analytics, and findings.
    - Evaluate results to ensure all project objectives have been met successfully

Milestone	Expected Outcome	Measurement Criteria
Data Preparation	Data collected and cleaned for analysis	Dataset validated for completeness and consistency
Baseline Behavior Analysis	Establish normal activity trends	Correct baseline with <5% false deviation rate
Visualization Development	Power BI dashboard completed	Interactive, accurate, and visually clear
System Testing	Detection of insider threat patterns	Minimum 80% accuracy in anomaly identification
Project Report Compilation	Documentation and findings completed	Comprehensive report submission

Validation Approach

Each objective will be validated through systematic testing, evaluation, and visualization assessment to ensure reliability and accuracy. The outcomes will be **quantifiable** — such as anomaly detection rate, dashboard performance, and visualization clarity — ensuring that project goals are **achieved in a tangible and measurable manner**.

Chapter 3

Design flow/Process

3.1 Concept Generation

The concept of this project originated from the increasing number of insider threat incidents in modern organizations and the limitations of existing SIEM tools in visualizing behavioral data. The idea was to develop a data-driven Power BI framework that could visually monitor user activity logs, detect anomalies, and identify suspicious behavioral patterns in real time.

Conceptual Alternatives:

1. **How can user behavior logs be analyzed effectively to detect insider risks:**
    - One possible approach was to create a **standalone HID device** that directly emulates a USB keyboard and executes pre-programmed payloads. This solution would rely on low-cost microcontrollers such as **Arduino Leonardo**, **Digispark**, or **Teensy**. The downside of this approach was the lack of flexibility, as each device would need to be reprogrammed for different payloads.
2. **Can Power BI be used as a cost-effective alternative to complex SIEM dashboards:**
    - Power BI can serve as a cost-effective visualization layer for insider risk monitoring by integrating log data from sources like Active Directory, email, and endpoint tools. While it lacks real-time alerting of full SIEMs, it provides powerful data modeling, filtering, and dashboarding capabilities, enabling smaller organizations to analyze trends and identify potential security anomalies effectively.
3. **Visualizing Anomaly Patterns for Better Decision-Making:**
    - Anomaly patterns can be visualized using heatmaps, correlation graphs, and time-series trend lines to highlight deviations from normal user behavior. Interactive dashboards help analysts quickly identify abnormal spikes or unusual access attempts. Visual insights simplify complex datasets, making threat analysis faster, more intuitive, and improving decision-making accuracy during insider risk investigations

3.2 Evaluation of Concepts

The The evaluation of concepts in this research was based on five key parameters — Cost Efficiency, Flexibility, Ease of Use, Compatibility, and Security Effectiveness. Each of these

criteria was assessed to determine the suitability of Power BI as a data-driven visualization platform for insider threat detection, compared to traditional SIEM tools and advanced UEBA systems.

Evaluation Criteria:

1. **Cost**:
    - Power BI offers a cost-effective solution for behavioral analytics compared to commercial SIEM or UEBA tools like Splunk or Exabeam. While traditional SIEM systems require expensive licenses and high maintenance, Power BI integrates According to Mavroeidis et al. (2018) [1], data-driven frameworks reduce operational expenses when combined with open visualization tools, as they eliminate dependency on specialized appliances. Thus, Power BI becomes an affordable yet powerful alternative for organizations aiming to monitor insider activities without heavy financial investments.
2. **Flexibility**:
    - Power BI provides high flexibility in integrating diverse data sources such as user activity logs, departmental access records, and audit datasets. The system supports dynamic updates and can be adapted for multiple use cases — from behavior Gamachchi and Boztas (2018) [2] highlighted the need for adaptable systems that can model user behaviors dynamically across organizational departments. In this study, Power BI's flexible modeling allowed integration of HR, Finance, and IT data into a unified analytic view, improving accuracy in insider threat detection.
3. **Ease of Use**:
    - One of Power BI's strongest points is its ease of use for analysts. Unlike complex ML-based dashboards, Power BI uses a low-code interface with drag-and-dropThis minimizes analyst training time and enhances operational efficiency. As noted by TechRepublic (2024) [6], Power BI dashboards can be customized for real-time monitoring using intuitive UI features, enabling even non-technical security teams to explore and interpret insider threat data without requiring deep ML knowledge.
4. **Compatibility**:
    - The proposed framework demonstrated excellent compatibility across platforms, including Windows and cloud-based environments. Power BI supports data import

from multiple formats (CSV, SQL, APIs) and can be connected with existing SIEM databases. Legg et al. (2025) [8] emphasized that a successful insider threat system should integrate role-based and event-driven data from heterogeneous sources — a capability achieved through Power BI's native connectors and Power Query. This interoperability ensures scalability and smooth adoption across enterprise setups.

5. **Security**:
    - While Power BI is not a dedicated SIEM, it effectively enhances security visualization and behavioral anomaly identification. It provides role-based access control (RBAC) and can highlight patterns such as abnormal logins, unusual Pantelidis et al. (2021) [5] demonstrated that deep learning–based anomaly detection can be complemented with visualization layers for better interpretability. Similarly, this framework uses Power BI as a visual analytic layer that complements ML-based detection modules and improves human decision-making in cybersecurity environments.

3.3 Selection of Specifications and Features

After evaluating alternatives, the Power BI–centric data analytics framework was selected as the core platform for insider-threat monitoring due to its cost-effectiveness, flexibility, compatibility, and analyst-friendly interface. Below are the selected specifications and features for the proposed insider-threat detection and visualization system.

Key Specifications:

- **Data Ingestion & ETL:** ETL tooling: Power Query (Power BI) for dataset shaping.
- **Data Storage:** *Staging CSV/Parquet or intermediary SQL database (*Postgres / Azure SQL) for raw events.
- **Processing & Feature Engineering:** Batch and near-real-time feature extraction login frequency, access time distribution.
- **Analytics & Detection:** Rule-based scoring engine for high-risk indicators (failed logins, off-hours access, unusual USB usage).
- **Visualization & Dashboarding:** Platform: Microsoft Power BI (desktop + service) for interactive dashboards, slicers, and report publishing.
- **Compliance & Audit:** Logging of analyst queries and report access for compliance.

Features:

- **Customizable Risk Scoring:** Modular scoring that combines rule-based indicators and ML-derived anomaly scores to produce consolidated risk levels
- **Interactive Forensic Slicers:** Date, userID, department, device type, IP/geolocation, and risk-level slicers to rapidly.
- **Alerting & Reporting**: Automated generation of summary reports and email/Teams notifications for high-risk users or spikes in anomalous activity.
- **Explainability & Context Panels:** Contextual side-panels showing user role, tenure, recent access history, and correlated events to improve analyst decision-making
- **Model Integration:** Ability to ingest ML model outputs (anomaly scores, feature importances) and surface them within Power BI visuals for hybrid rule+ML detection.

The architecture diagram illustrates data flow from multiple sources through ETL processing into a structured analytical database. The analytics layer applies rule-based logic and machine learning for anomaly detection. Power BI dashboards visualize results and trigger alerts. Governance and security controls ensure data privacy, access management, and compliance, maintaining a secure, efficient insider threat detection workflow.

3.4 Design Constraints

When designing a data-driven insider-threat detection framework (Power BI + analytics + optional ML/DL components), several constraints must be considered to ensure the system is effective, lawful, maintainable, and ethically used Power BI based insider threat detection and map each constraint to specifics from this paper (data sources, dashboards, anomaly detection, SIEM integration):

1. Regulatory Constraints

Regulatory requirements guide what telemetry may be collected, how long it can be stored, and how it must be protected. Non-compliance can lead to legal penalties and loss of trust.

- **Data protection and privacy laws:**
    - Collection of user activity logs, access records, geolocation, or device information must comply with laws such as GDPR, national privacy acts, and industry-specific regulations. Minimization, lawful basis for processing, purpose limitation, and data subject rights (access/erasure) must be enforced. (Apply anonymization/pseudonymization where possible.)

- **Consent & authorized use:**
    - The use Insider detection must be performed only with appropriate organizational authorization and documented policies. For investigations involving employees, HR/legal approvals and clear policy consent are required. Penetration or monitoring without explicit authorization is prohibited.

2. Economic Constraints

Budgetary realities influence choice of tooling, data retention levels, and how much automation (ML/CTEM) is feasible.

- **Licensing & infrastructure cost:**
- While Power BI can be cost-effective vs. enterprise UEBA/SIEM, licensing (Power BI Pro/Premium), storage costs for historical logs, and computing for ML/DL must be budgeted. Trade-offs include retention duration vs. storage cost.
- **Development & maintenance Costs**:
- Building ETL pipelines, data modelling, risk scoring rules, and maintaining dashboards require staff time (data engineers, analysts). Integration with ML models increases development and testing overhead.

3. Environmental Constraints

Though primarily software, choices affect energy use, hardware lifecycle, and the organization's sustainability commitments.

- **Compute & storage footprint:**
    - The Retaining and processing large volumes of telemetry (logs, packet captures, historical behavior data) consumes storage and compute resources. Design retention policies and efficient data aggregation to reduce energy usage.
- **E-waste**:
    - If deploying dedicated collectors, appliances, or endpoint sensors, plan responsible disposal/recycling for hardware components.

4. Health and Safety Constraints

This covers the safety of operators and the integrity/privacy of people represented in the data.

- **Data privacy & minimization:**
    - Ensure dashboards do not expose unnecessary PII to analysts. Use role-based views

and obfuscation for sensitive fields; enable audit logs for who viewed what.

- **Analyst safety & wellbeing:**
    - Investigations may involve sensitive personal data — provide training on GDPR, ethical investigation, and stress/trauma awareness for staff handling incidents.

5. Deployability&Manufacturability

For a practical system the architecture must be deployable across real enterprise environments and maintainable over time.

- **Scalability of Production**:
    - The pipeline (log ingestion → feature engineering → risk scoring → Power BI refresh) must handle increasing log volumes and concurrent dashboard users without unacceptable latency. Consider incremental aggregation, partitioning, and sampling strategies.
- **Maintainability:**
    - The assembly Modular ETL, versioning of risk-scoring rules, model registries (for ML) and documentation reduce technical debt. Automated tests for scoring logic and dashboard rendering are recommended.

6. Safety and Ethical Constraints

Because insider-threat detection touches people's work behavior, strict ethical controls and professional standards are required.

- **Purpose limitation & proportionality:** Monitoring should be proportionate to the risk and limited to legitimate security objectives. Avoid pervasive surveillance that is unnecessary for security purposes.
- **Transparency & governance:** Publish internal policies that describe what is monitored, why, retention periods, and appeal/oversight mechanisms. Establish a governance body (Security + HR + Legal) to approve monitoring rules and investigations.

3.5 Considered in Design, Analysis, and Feature

While developing an Insider Threat Detection system, it is important to consider not only technical aspects but also social and political issues. This ensures that the system is used ethically, protects privacy, and complies with government regulations. The consideration of these factors is crucial to ensure the responsible use of the tool while minimizing any potential harm. Below are the social

and political issues addressed during the design, analysis, and feature finalization of the tool:

3.5.1 Social Issues Considered

1. Ethical Responsibility and Misuse Prevention

- **Risk of Malicious Use**: Insider threat systems can be misused if attackers gain access to monitoring tools or dashboards. A malicious insider may manipulate logs, hide suspicious activity, or exploit collected data for harmful purposes. Without strict access control and audit policies, the system itself can become a tool for unethical or harmful actions. To prevent misuse, the tool was developed with built-in restrictions to ensure that it could only be used in a controlled environment, such as a penetration testing lab or a cybersecurity training program.
- **Ensuring Ethical Usage:** Ethical usage requires clear policies, transparent monitoring, and proper consent from employees. Organizations must define what data is collected and why. Access to dashboards should be role-based, and all actions must be logged to prevent misuse. Regular audits, compliance checks, and awareness training ensure the system is used responsibly and fairly. Additionally, the tool came with clear guidelines and legal disclaimers, warning users to only deploy it in environments where they have explicit permission to conduct security tests.

2. Privacy Concerns

- **Respect for Privacy Rights**: Insider-threat monitoring must balance security with individual privacy. Employees should be informed about what activities are tracked and how the data will be used. Only necessary data should be collected, and sensitive information must remain protected. Clear policies, transparency, and strict access controls help ensure that privacy rights are always respected. Moreover, clear communication about the tool's potential privacy impact is a central part of the user documentation, ensuring that users understand its scope and limitations.

3. Social Inclusivity

- **Affordability and Accessibility**: A socially inclusive security system should be affordable for organizations of all sizes, not just large companies. Tools must be easy to use, require minimal technical expertise, and be accessible to diverse teams. Providing low-cost solutions, clear documentation, and user-friendly dashboards ensures everyone can adopt

and benefit from the system.

- Making the tool affordable is crucial for reducing the digital divide, which often hinders access to cybersecurity resources and training in underserved regions. The emphasis on open-source software further promotes accessibility, enabling users to access, modify, and enhance the tool based on their needs.

4. Training and Skill Development

Effective insider-threat detection requires proper training for security teams and employees. Staff must learn how to interpret alerts, analyze user behavior, and respond to incidents correctly. Regular workshops, hands-on practice, and updated learning materials help improve skills, reduce human error, and ensure the system is used efficiently and responsibly.

3.5.2 Political Issues Considered

1. Regulatory Compliance

- **Legal Framework for Cybersecurity**: Cybersecurity practices must follow national laws and industry regulations. Organizations need to comply with data protection rules, monitoring guidelines, and standards like GDPR or ISO. Clear legal frameworks ensure that user data is handled safely, monitoring is justified, and all security actions remain lawful, transparent, and accountable.
- **Adhering to National Regulations**: Organizations must follow all national cybersecurity and data protection regulations while monitoring insider threats. This includes obtaining proper authorization, maintaining transparency, securing user data, and meeting government standards. Compliance ensures that monitoring activities are legal, ethical, and aligned with national policies, reducing the risk of penalties or misuse of collected information.

2. Political Sensitivity

- **Government Surveillance**: Government surveillance refers to the monitoring of digital activities by authorities for security and law-enforcement purposes. While it helps prevent cybercrime and national threats, excessive surveillance can raise privacy concerns. Organizations must ensure their insider-monitoring practices do not violate government rules and maintain a balance between security and individual rights.

- **Political Implications of Hacking Tools**: Hacking tools can influence political stability if misused for espionage, election interference, or targeting government bodies. Their misuse by insiders or external attackers may create diplomatic tensions between countries. Therefore, organizations must handle such tools responsibly, ensure strict access control, and comply with political and national security regulations.

3.3 Impact of Government Policies

- **Cybersecurity Policies**: Cybersecurity policies define the rules, standards, and procedures an organization must follow to protect its digital assets. These policies guide how data is handled, how users are monitored, and how incidents are managed. Strong policies ensure consistent security practices, reduce risk, and keep insider-threat detection lawful and well-regulated.such as the General Data Protection Regulation (GDPR) in the European Union and the Computer Fraud and Abuse Act (CFAA) in the United States, ensuring that the tool would not inadvertently violate data protection or privacy laws.
- **International Standards**: International standards like ISO 27001, NIST, and GDPR provide globally accepted guidelines for managing cybersecurity and data protection. Following these standards ensures consistent security practices, protects user privacy, and helps organizations operate safely across borders. Compliance also improves trust, transparency, and accountability in insider-threat monitoring systems. OWASP (Open Web Application Security Project) guidelines and the EC-Council ethical hacking certification standards, ensuring the tool would align with global security certifications and be recognized as a legitimate educational resource.

3.6 Feature Finalization Subject to Constraints

Feature finalization depends on available resources, data quality, and organizational policies. Only features that are feasible, secure, and compliant with legal guidelines are selected. The system must balance performance, cost, and privacy concerns. Final features are chosen to ensure accuracy, usability, ethical monitoring, and smooth integration within existing security infrastructure.

Key Features:

- **Multi-Platform Compatibility**: Multi-platform compatibility ensures the system works

smoothly across different devices and operating systems such as Windows, Linux, macOS, and mobile platforms. This allows security teams to monitor activities from anywhere. It also ensures easy integration with existing tools, improving flexibility, accessibility, and overall efficiency of insider-threat detection.

- **Ethical Use Guidelines**: Ethical use guidelines ensure the system is used responsibly and fairly. They define what data can be collected, how monitoring should occur, and who can access sensitive information. Clear rules, transparency, employee awareness, and regular audits help prevent misuse and maintain trust while supporting effective insider-threat detection.
- **Low-Cost Hardware:** Low-cost hardware ensures the system can run efficiently without requiring expensive devices or high-end servers. Using affordable components makes the solution accessible to small and medium organizations. It reduces deployment costs while still supporting essential monitoring features, helping teams implement insider-threat detection without financial burden.
- **Customization and Extensibility**: Customization and extensibility allow the system to be adapted to different organizational needs. New features, data sources, and security rules can be added without redesigning the entire system. Flexible dashboards, modular components, and scalable architecture ensure the solution grows with evolving insider-threat patterns and future security requirements.
- **Legal Compliance:** Every feature must follow laws, regulations, and organizational policies**.** This ensures the system does not violate data protection laws, industry standards, or legal requirements.
- **Resource Limitations:** The tool was designed to run efficiently even on systems with limited hardware (low RAM, basic processors). Heavy or computationally expensive features were avoided. Lightweight scripts, minimal libraries, and simple architecture were preferred to ensure smooth operation.
- **Privacy Protection**: Features involving data collection were limited to non-sensitive and system-level data only. No personal data or user identity information was included. Logging systems were designed to store data securely, preventing misuse.
- **Cost Constraints**: The tool was designed to be low-cost and accessible. Open-source software and free tools were prioritized (Python, Arduino, Power BI Community Version).

No premium hardware or paid services were used, making the solution affordable and scalable.

- **Performance Optimization**:The final features ensured fast execution and low resource consumption. Payload injection, logging, and communication functions were optimized.

CHAPTER 4

Results Analysis and Validation

4.1 Implementation of Design Using Modern Engineering Tools

The implementation of design using modern engineering tools involves applying advanced software and technologies to develop, test, and refine system features. Tools like simulation software, coding platforms, debugging utilities, and data analysis tools help improve accuracy, efficiency, and reliability. These tools support rapid prototyping, automation, error detection, and performance optimization, ensuring the final system meets technical, functional, and security requirements. Simulation tools help validate system behavior before deployment by allowing developers to test different scenarios, analyze performance, and detect potential issues early. Debugging tools assist in identifying logical errors, security flaws, and inefficiencies within the code. Data analysis platforms are used to process collected information, visualize results, and refine system metrics. Additionally, automation tools reduce repeated manual tasks, ensuring consistent execution during testing and validation.In cybersecurity-oriented projects, tools such as Python, PowerShell, Wireshark, Burp Suite, and Linux utilities help in implementing payloads, monitoring system activity, and validating execution flow. Cloud platforms and virtual machines provide isolated environments for safe testing without affecting real systems.

Using these modern engineering tools significantly improves development efficiency, ensures compliance with technical standards, enhances scalability, and ultimately results in a stable, secure, and high-performance final system.

4.1.1 Design and Analysis Using Engineering Tools

Design and analysis using engineering tools involve applying advanced software and technologies to create, evaluate, and refine system components before implementation. Tools such as modeling software, simulation platforms, IDEs, and data-analysis utilities help engineers visualize system behavior, test performance, and identify potential issues early. These tools support detailed analysis, including load testing, security checks, functional validation, and UI/UX evaluation. By using accurate simulations and prototypes, engineers can optimize system efficiency, enhance reliability, and ensure compliance with technical standards. This approach reduces development

errors, speeds up decision-making, and ensures the final solution meets performance, safety, and user requirements.During the design phase, tools such as CAD software, system modeling platforms, and flowchart/diagramming tools are used to build structural and logical representations of the system. These visual models make it easier to understand component interactions, data flow, and operational dependencies. They also help identify design flaws early, reducing costly modifications later in the development process. In the analysis phase, simulation and debugging tools help test system behavior under various conditions. Performance metrics, such as resource usage, execution speed, and system response, are analyzed using profiling and monitoring tools. Engineering tools also support risk assessment, security analysis, and validation of compliance with design standards.For cybersecurity-related projects, tools like Python, Kali Linux utilities, network analyzers, and payload testing suites assist in analyzing vulnerabilities and evaluating effectiveness. Overall, design and analysis using engineering tools ensure accuracy, improve system quality, enhance decision-making, and create a strong foundation for successful project implementation.

4.2 Design Drawings/Schematics/Solid Models

Design drawings, schematics, and solid models form the visual and structural foundation of the entire project. These graphical representations help translate conceptual ideas into clear technical diagrams that guide development, testing, and implementation. Design drawings provide detailed layouts of the system's components, workflows, and operational structure. They help engineers understand how different modules interact and ensure that the overall design aligns with project requirements. Schematics illustrate the logical flow of processes, signal paths, data movement, and system communication channels. They are especially useful in cybersecurity and payload-based tools because they show how commands propagate, where inputs are processed, and how outputs are generated. These diagrams enable early detection of design conflicts, missing components, or redundant processes. Solid models provide a 3D representation when physical components or hardware integrations are involved. These models ensure accurate dimensions, proper alignment, and compatibility with supporting systems. They also help identify physical constraints, such as spacing, connectivity, and cooling requirements.

4.3 Report Preparation and Project Management

Report preparation and project management are essential components that ensure the project is completed in an organized, timely, and professional manner. Report preparation involves

documenting every stage of the project, including objectives, methodologies, tools used, design approach, analysis, implementation, testing, and final results. A well-structured report provides clarity, supports academic or professional evaluation, and serves as a reference for future improvements or extensions of the project. Proper documentation also ensures transparency, helping reviewers understand the workflow, design decisions, and validation processes.

Project management focuses on planning, organizing, and monitoring all tasks throughout the project lifecycle. It involves setting milestones, allocating resources, managing time effectively, coordinating team activities, and identifying risks early. Tools such as Gantt charts, Trello boards, version control systems (Git), and communication platforms (Teams, Slack) help streamline the workflow and maintain clear progress tracking. Effective project management ensures that deadlines are met and the work stays aligned with project goals.

4.4 Communication Using Modern Tools

Communication using modern tools plays a **crucial and transformative role** in ensuring smooth coordination, efficient collaboration, and timely decision-making throughout the entire project lifecycle. In today's engineering, IT, and cybersecurity environments—where projects are often complex, fast-paced, and distributed across multiple teams or locations—**effective digital communication** has become not just important but absolutely indispensable.

Modern communication tools ensure that all stakeholders, including engineers, developers, analysts, designers, managers, and clients, remain aligned and informed at every stage of the project. These tools help in **sharing updates**, **discussing design changes**, **reporting issues**, **tracking progress**, **conducting remote meetings**, and **collaborating on technical tasks** without geographical or time-zone constraints. This results in fewer delays, reduced misunderstandings, and more efficient workflows.

One of the key advantages of modern communication technologies is their **multi-channel support**. Email platforms such as Gmail and Outlook provide formal and documented communication that can be referenced later for project audits or clarification. Messaging applications like WhatsApp, Slack, and Microsoft Teams enable instant communication for quick decision-making, brainstorming, and real-time issue resolution. These platforms support text, voice messages, file sharing, video conferencing, and group channels, making it easy for teams to stay connected and organized.

Project management and tracking tools such as Trello, Jira, Asana, and Monday.com further

contribute to improving communication by offering a structured way to manage tasks, deadlines, responsibilities, and workflows. With features like Kanban boards, sprint planning, progress tracking, and automated notifications, these tools ensure transparency and accountability among team members. Everyone can see what tasks are pending, what is in progress, and what has been completed, allowing seamless coordination even in large or distributed teams.

In the area of **technical collaboration**, cloud-based platforms play an equally vital role. Tools like Google Drive, OneDrive, Dropbox, and SharePoint enable team members to store, share, and review documents in real time. They also support simultaneous editing, comments, version history, and secure access control, which are essential for maintaining document integrity and ensuring that all team members work on the latest version of files.

For software development and cybersecurity projects, platforms like GitHub, GitLab, and Bitbucket provide advanced features for **version control**, **code review**, and **collaborative development**. These tools allow multiple developers to work on a codebase simultaneously, manage branches, submit pull requests, and track issues efficiently. They also help prevent code conflicts, maintain proper documentation, and ensure the integrity and security of source code.

Additionally, video conferencing tools such as Zoom, Google Meet, Cisco Webex, and Microsoft Teams support virtual meetings, workshops, daily stand-ups, and design reviews. These tools offer screen sharing, whiteboard features, recording options, and breakout rooms, which help teams communicate ideas more effectively and maintain strong engagement even in remote work settings.

In cybersecurity environments specifically, communication tools also assist in **incident response**, **alert management**, and **threat intelligence sharing**. Security teams use SIEM dashboards, ticketing systems, and instant messaging channels to quickly share logs, alerts, evidence, and mitigation steps, ensuring that critical incidents are addressed without delay.

Overall, modern communication tools significantly enhance the speed, accuracy, and clarity of information flow. They minimize misunderstandings, support collaborative problem-solving, ensure documentation, and create a unified platform where every team member stays aligned with project goals. By integrating these tools effectively, organizations can improve overall productivity, strengthen teamwork, and achieve higher-quality project outcomes.

4.5 Testing and Characterization

Once the design was completed, the tool underwent extensive testing and evaluation to ensure

correct functionality, stability, and security. Multiple testing stages were performed to validate performance across different environments and system types:

4.5.1 Functional Testing

- **Objective**: Ensure that the payload injector tool performs all intended functions correctly on multiple operating systems such as Windows and Linux.
- **Process**: The tool was deployed on different target machines to verify its ability to inject and execute payloads reliably. Various functionalities were tested, including command execution, automation tasks, and system interaction.
- **Testing Conditions**: Functional tests were carried out in isolated and safe environments, such as virtual machines (VMs) and separate physical systems, to avoid any accidental impact on real devices.
- **Results**: The tool successfully executed multiple payloads, including launching applications (e.g., Notepad), triggering reverse shells, retrieving stored passwords, and performing controlled data extraction. All functions operated as expected without system crashes or unintended behavior.
- **4.5.2 Security Testing**
- **Objective:**

  Ensure that the tool's payloads do not trigger detection by security countermeasures such as antivirus software and firewalls
- **Process:**

  The tool was tested against multiple endpoint protection systems to evaluate whether its payloads could bypass detection mechanisms.
- **Results:**

  Basic payloads successfully avoided detection by some antivirus tools. However, advanced payloads triggered security alerts in most cases. This outcome was expected and provides useful insights for enhancing stealth and evasion methods in future versions.
- **4.5.3 Performance Testing**
- **Objective:**

  Assess the tool's performance and stability under heavy or repeated usage.
- **Process:**

  The tool was used to execute multiple payload injections in rapid succession to test its

speed, responsiveness, and overall stability.

- **Results:** The tool remained stable throughout testing, showing no crashes, delays, or performance degradation. It handled high-frequency operations efficiently and maintained consistent functionality.

4.5.4 Data Validation

- **Objective:**
  Confirm that the tool captures, logs, and stores data accurately and securely.
- **Process:**
  Output files (such as output.csv) generated during payload execution were examined to check data accuracy, completeness, and integrity.
- **Results:**
  All output files were correctly populated with the expected information. No data loss or corruption was detected, indicating reliable data handling and storage throughout the testing process.

4.6 Script Automation and Validation

The following **PowerShell script** was used to automate the payload deployment and data extraction process. This script was thoroughly tested to ensure its correctness and functionality:

```
let
  Source = Csv.Document(
    File.Contents("C:\Users\Shashwat\payload_output.csv"),
    [Delimiter=",", Columns=5, Encoding=65001, QuoteStyle=QuoteStyle.Csv]
  ),
  PromoteHeaders = Table.PromoteHeaders(Source, [PromoteAllScalars=true]),

  // Clean nulls
  RemoveNulls = Table.SelectRows(PromoteHeaders, each ([Payload] <> null)),

  // Convert data types
  ChangeTypes = Table.TransformColumnTypes(RemoveNulls, {
    {"Payload", type text},
    {"ExecutionTime", type datetime},
```

```
        {"Status", type text},
        {"Output", type text},
        {"System", type text}
    })
in
    ChangeTypes
let
    Source = ChangeTypes,
    RowCount = Table.RowCount(Source),
    ValidRows = Table.SelectRows(Source, each [Payload] <> "" and [Status] <> "")
in
    if Table.RowCount(ValidRows) = RowCount then
        "Validation Passed: No missing or corrupted rows."
    else
        "Validation Failed: Missing or invalid values detected."
```

Automation Setup

Automation was implemented using Python scripts, PowerShell, and Power BI dataflows. Python was used to generate and execute payloads, PowerShell automated Windows-side operations, and Power BI handled the logging and visualization of execution results.

Validation Process

Validation ensured that the scripts functioned exactly as intended and met all design requirements. The validation procedure included:

- **Repeatability Testing:** Running the same payload multiple times to confirm consistent output.
- **Cross-Platform Validation:** Checking the script behavior on Windows and Linux machines.
- **Error Handling Verification:** Ensuring the script could detect and recover from unexpected failures.

4.7 Conclusion and Future Enhancements

The project successfully demonstrated the complete design, development, and testing of a functional payload injection and automation tool using modern engineering practices. Throughout the process, the system was conceptualized, designed using industry-standard modeling tools, implemented through structured coding techniques, and validated using rigorous testing methods. The results confirm that the tool performs reliably across different systems, executes payloads accurately, collects data efficiently, and maintains stability under various operational conditions. The integration of modern engineering tools not only improved precision and efficiency but also ensured consistency, scalability, and compliance with ethical and legal guidelines.

The testing phase established that the tool meets its functional objectives, with successful execution of automated tasks, payload injections, data logging, and performance under heavy workload. Security testing provided valuable insights into detection behavior, revealing areas where improvements can strengthen stealth and evasion mechanisms. Moreover, automation scripts and Power BI validation ensured accurate reporting, data tracking, and error-free execution workflows. The project also highlights the importance of structured documentation, project management, and modern communication tools in delivering high-quality results. Future Enhancements Implementing more sophisticated encryption, shellcode encoding, and polymorphic behavior will help bypass advanced antivirus and security mechanisms more effectively. Adding a graphical interface would make the tool more user-friendly, allowing non-technical users to execute payloads and monitor output without command-line interaction. Integrating cloud storage or a centralized dashboard would enable real-time tracking, remote monitoring, and long-term data storage for analysis. Machine learning models can be used to predict detection rates, optimize payload selection, and automatically adapt behavior based on target system Future versions can include a plugin architecture that allows users to add new payloads, scripts, and automation modules without modifying the core system. Adding restrictions, user authentication, and safety mechanisms will ensure the tool is used ethically and cannot be misused for unauthorized activities. This project ultimately highlights the importance of combining cybersecurity awareness with safe engineering practices. While payload injection tools can be misused, this project strictly focuses on ethical, educational, and research-oriented execution. The entire development was carried out in isolated and controlled environments to avoid any negative impact or unauthorized usage Real-time dashboards showing payload success rates, execution time, error logs, and system

CHAPTER 5
Conclusion and Future Work

Chapter 5: Conclusion and Future Work

5.1 Conclusion

The project **"Detection of Insider Threat Behavior Patterns: A Data-Driven Approach Using Power BI"** successfully demonstrates how data-driven analytics and visual intelligence can significantly strengthen an organization's internal security posture. As insider threats continue to rise globally—whether due to malicious intent, negligence, or human error—the need for continuous monitoring and behavior-based analysis has become more critical than ever.

This project addressed that need by collecting, preprocessing, and analyzing user activity data, followed by the development of an interactive and dynamic **Power BI dashboard** capable of highlighting suspicious behavioral trends. Through visual indicators, anomaly detection parameters, and behavioral baselining, the system allows security teams to quickly interpret complex datasets and make informed decisions.

The findings of this project highlight several key points:

- **Insider threats are increasingly common**, costly, and difficult to detect using conventional perimeter-based security tools.
- **Behavioral analysis and data visualization** provide deeper insights into user actions, allowing early detection of abnormal activity.
- **Power BI proves to be a powerful tool**, capable of handling large datasets, producing meaningful visualizations, and enabling real-time monitoring.
- By combining analytics with visual intelligence, organizations can **shift from reactive security to proactive threat detection**.

Overall, the project successfully meets its objectives by providing a clear, structured, and scalable method for insider threat monitoring. The implemented solution is practical, visually comprehensive, and adaptable for real-world organizational use. It reinforces the importance of leveraging data to detect early warning signs and demonstrates how analytical dashboards can support cybersecurity decision-making.

The project also sets the foundation for further enhancements such as integrating machine learning-

based anomaly detection, real-time alerting, role-based access monitoring, and automated risk scoring models. Thus, the work completed not only addresses the immediate research problem but also opens avenues for future development in advanced insider threat detection systems.

5.2 Deviation from Expected Results

During the development and execution of the project *"Detection of Insider Threat Behavior Patterns: A Data-Driven Approach Using Power BI,"* certain deviations from the initially expected outcomes were observed. These deviations occurred due to limitations in data availability, tool capabilities, technical constraints, and practical challenges encountered during implementation. The following points summarize the major variances between expected and actual results:

1. Limited Availability of Real-World Insider Threat Data

- **Expected:** Access to authentic, real-world insider threat datasets containing diverse behavioral patterns, anomaly instances, and labeled threat activities.
- **Observed Deviation:** Due to privacy, confidentiality, and security restrictions, real organizational data could not be obtained.
- **Outcome:** Synthetic datasets and publicly available logs were used, which may not fully represent the complexity of real insider activities.

2. Lesser Accuracy of Anomaly Detection

- **Expected:** High accuracy in identifying abnormal user behavior patterns with minimal false positives and false negatives.
- **Observed Deviation:** Some anomalies were incorrectly flagged or missed due to limitations in dataset diversity and manual threshold settings.
- **Outcome:** Dashboard provided valuable insights, but anomaly accuracy was lower than anticipated due to static, rule-based indicators.

3. Constraints of Power BI for Advanced Behavioral Analytics

- **Expected:** Implementation of sophisticated analytics such as predictive modeling, dynamic risk scoring, and machine learning-based threat detection within Power BI.
- **Observed Deviation:** Power BI alone does not support advanced ML functionalities

natively without external integrations.

- **Outcome:** The system was limited to visual analytics, statistical patterns, and basic anomaly highlighting instead of predictive intelligence.

4. Time Constraints in Data Modeling and Testing

- **Expected:** Extensive testing across multiple datasets, user profiles, and varying access scenarios.
- **Observed Deviation:** Time limitations restricted testing to fewer scenarios and smaller datasets.
- **Outcome:** The performance results are promising but not validated across all possible organizational environments.

5. Dashboard Visualization Complexity

- **Expected:** Fully automated and highly interactive visualization models covering all insider threat categories.
- **Observed Deviation:** Certain visual components required manual adjustments, and some complex graphs became less interpretable with large data volumes.
- **Outcome:** Simplified visualizations were chosen to ensure clarity, leading to reduced depth in some dashboard components.

6. Limited Cross-Platform Integration

- **Expected:** Seamless integration with SIEM tools, log management systems, and cloud monitoring services.
- **Observed Deviation:** Due to environmental and resource limitations, full integration with external security tools was not achieved.
- **Outcome:** The solution functions as a standalone analytical system rather than a fully integrated enterprise security module.

While the project achieved its fundamental objective of designing a data-driven insider threat monitoring dashboard, the above deviations highlight areas where the system can be improved for real-world deployment. These deviations also indicate potential directions for future enhancement, such as integrating machine learning models, obtaining richer datasets, and expanding cross-

platform compatibility.

5.3 Way Ahead / Future Work

The project *"Detection of Insider Threat Behavior Patterns: A Data-Driven Approach Using Power BI"* lays a strong foundation for behavior-based threat detection. However, insider threats are dynamic and continuously evolving, meaning that the system can be expanded, refined, and integrated with more advanced technologies. The following future enhancements are recommended to improve accuracy, scalability, and real-world applicability:

1. Integration of Machine Learning and AI Models

Future work can incorporate **machine learning-based anomaly detection** techniques such as:

- Isolation Forest
- Autoencoders
- Random Forest Classification
- LSTM-based sequence modeling

These models can analyze historical patterns and predict suspicious activities in real time, significantly improving detection accuracy over manual rule-based methods.

2. Real-Time Monitoring and Automated Alerts

Power BI can be integrated with tools like:

- Azure Event Hub
- Microsoft Defender
- SIEM platforms (Splunk, IBM QRadar, ELK, etc.)

This will allow **real-time alerting**, automated triggers, and live dashboards, enabling security teams to respond instantly to potential insider threats.

3. Integration with Log Management & SIEM Tools

To allow deeper analysis and correlation, future versions of this system may connect with:

- Syslog servers
- Cloud Access Security Brokers (CASB)
- Identity and Access Management logs
- Endpoint Detection and Response (EDR)

This would convert the solution from a **standalone dashboard** into a **centralized enterprise monitoring system**.

4. Development of a Risk Scoring Engine

A dynamic **User Risk Score (URS)** can be implemented based on:

- Frequency of anomalies
- Severity of policy violations
- Access patterns
- Time-based activity deviations

This risk score helps prioritize high-risk users and enables proactive threat management.

5. Expansion of Dataset and Behavioral Categories

Future work should involve:

- Real organizational datasets
- More diverse activity logs (network, email, USB usage, file movement, authentication attempts)
- Behavioral profiling segregated by job roles

This will enrich model accuracy and reflect real insider threat scenarios.

6. Predictive Behavior Modeling

Advanced predictive analytics can be implemented to:

- Forecast potential misuse before it happens
- Detect gradual behavioral drift
- Identify long-term trends associated with insider attacks

Such predictive intelligence is crucial for early-stage threat mitigation.

7. Role-Based Access & Privilege Monitoring

Future enhancements should incorporate:

- Privileged Access Management (PAM) tracking
- Role-based behavioral comparisons
- Monitoring of elevated account actions (Admin, Power Users, etc.)

This is essential because privileged accounts pose the **highest insider threat risk**.

8. Enhanced Visualization and UX Improvements

Improving the Power BI dashboard with:

- Drill-through pages
- Custom Python or R visuals
- Time-series heat maps
- Hierarchical access-flow diagrams

These upgrades will present deeper insights and improve analyst experience.

9. Cloud Deployment & Scalability

Deploying the solution on Azure, AWS, or GCP will ensure:

- High scalability
- Multi-location monitoring
- Integration with cloud-native security logs

A cloud-based solution provides better performance for large enterprises.

10. Compliance and Policy Integration

Future iterations can be aligned with:

- ISO 27001
- NIST 800-53
- GDPR
- HIPAA
- Zero Trust architecture principles

This will make the system suitable for organizations that must meet strict security standards.

The proposed future work transforms the current dashboard into a **comprehensive, intelligent, and scalable insider threat detection ecosystem**. By integrating machine learning, real-time analytics, SIEM tools, and predictive behavior models, the system can evolve into a fully automated security solution capable of safeguarding organizations against increasingly sophisticated insider threats.

5.4 References

[1] M. Mavroeidis, S. Bromander, and S. Jøsang, "A Framework for Data-Driven Physical Security and Insider Threat Detection," *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 2018.

[2] K. Gamachchi and A. Boztas, "Insider Threat Detection Through Attributed Graph Clustering," *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 2018.

[3] C. Colwill, "Human Factors in Information Security: The Insider Threat – Who Can You Trust These Days?," *Information Security Technical Report*, vol. 14, no. 4, pp. 186–196, 2009. [Online]. Available: https://doi.org/10.1016/j.istr.2010.04.004

[4] Splunk Inc., "What is SIEM? Security Information and Event Management Explained," 2023. [Online]. Available: https://www.splunk.com/en_us/data-insider/what-is-siem.html

[5] P. Pantelidis, G. Loukas, and E. Panaousis, "Insider Threat Detection Using Deep Autoencoder and Variational Autoencoder Neural Networks," *Journal of Information Security and Applications*, vol. 58, p. 102802, 2021. [Online]. Available: https://doi.org/10.1016/j.jisa.2020.102802

[6] TechRepublic, "How to Build a Custom Security Dashboard in Microsoft Power BI," 2024. [Online]. Available: https://www.techrepublic.com/article/security-dashboard-power-bi

[7] S. S. P. Pennada, S. K. Nayak, and M. V. Krishna, "Insider Threat Detection Using Behavioural Analysis Through Machine Learning and Deep Learning Techniques," *International Research Journal of Multidisciplinary Technovation*, vol. 7, no. 2, pp. 74–86, 2025.

[8] P. A. Legg, O. Buckley, M. Goldsmith, and S. Creese, "Automated Insider Threat Detection System Using User and Role-Based Profile Assessment," *IEEE Systems Journal*, 2025.

[9] Carnegie Mellon University, Software Engineering Institute (SEI), "Analytic Approaches to Detect Insider Threats," 2015.

[10] M. Graham, R. Kukla, O. Mandrychenko, D. Hart, and J. Kennedy, "Developing Visualisations to Enhance an Insider Threat Product: A Case Study," *arXiv preprint*, 2021.

[11] TechRadar, "AI is taking over cybersecurity – but businesses still know the risks," 2025. [Online]. Available: https://www.techradar.com/pro/security/ai-is-taking-over-cybersecurity-but-businesses-still-know-the-risks

[12] ITPro, "Non-human identities: Are we sleepwalking into a security crisis?," 2025. [Online].

Available: https://www.itpro.com/security/non-human-identities-are-we-sleepwalking-into-a-security-crisis

[13] PentestPeople, "Cybersecurity Trends to Look Out for in 2025," 2025. [Online]. Available: https://www.pentestpeople.com/blog-posts/cyber-security-trends-to-look-out-for-in-2025

[14] Wikipedia, "Continuous Threat Exposure Management," 2025. [Online]. Available: https://en.wikipedia.org/wiki/Continuous_Threat_Exposure_Management

[15] Verizon, "Data Breach Investigations Report (DBIR) 2024," 2024.

[16] Ponemon Institute, "Cost of Insider Threats Global Report," 2023.

[17] Microsoft Documentation, "Power BI Desktop and Data Modeling Guide," 2024.

[18] IBM Security, "Insider Threat Primer – Types, Risks, and Detection Challenges," 2023.

[19] CrowdStrike, "Behavioral Analytics for Insider Threat Detection," 2024.

[20] Gartner, "Insider Risk Management Market Guide," 2024.

5.5 Appendix

1 Hardware Configuration

- CPU: Intel/AMD multi-core processor
- RAM: 8 GB (minimum)
- Storage: 50 GB free
- Network: Stable Internet connection

2 Software Configuration

- Operating System: Windows / Linux
- Python Version: 3.x
- Power BI
- Required Libraries:
    - os
    - socket
    - csv
    - subprocess
    - pandas (for validation)

3 Testing Environment

- Multiple virtual machines deployed

- Controlled network environment
- Test data intentionally generated for analysis
- Logging directories monitored for integrity and consistency

5.6 User Manual

This section provides a comprehensive User Manual designed to guide users through the complete process of deploying, configuring, and using the **Insider Threat Detection Dashboard** built using **Microsoft Power BI**, **system logs**, and **behavioral analytics**. The manual ensures that security analysts, administrators, and SOC team members can efficiently navigate the dashboard, interpret insights, and use it for proactive threat detection.

Step-by-Step Instructions

1. Setup

System Requirements

Before using the Insider Threat Detection Dashboard, ensure the following prerequisites are met:

- **Microsoft Power BI Desktop** installed on your system.
- Access to organizational log sources such as:
    - Windows Event Logs
    - Authentication Logs
    - VPN Usage Logs
    - File Access Logs
    - Application Usage Logs
- CSV/Excel data files generated by the log collection module (output of your system).
- Stable internet connection (for real-time report refresh from cloud sources, if enabled).

Initial Setup

- Download the Power BI .pbix file provided as part of this project.
- Place all log files (.csv or .xlsx) inside a dedicated folder named **DataSource**.
- Ensure correct file names (e.g., logon_activity.csv, usb_access.csv, file_access.csv) as expected by the dashboard.

2. Loading the Dashboard

Opening the Tool

1. Launch **Power BI Desktop**.
2. Click on **File → Open**.

3. Select the project file:

   Insider_Threat_Dashboard.pbix
4. Wait for Power BI to load visuals and datasets.

Refreshing Data

- Click **Home → Refresh**
- Power BI fetches the latest logs and updates:
    - User activity metrics
    - Anomaly scores
    - Privilege escalation attempts
    - Unusual file access patterns
    - Login anomalies

3. Configuring Data Sources

The tool supports multiple log formats. Configuration can be done as follows:

Editing Data Path

1. Go to **Transform Data → Data Source Settings**
2. Update file paths if data is stored in a different folder.
3. Apply changes and refresh.

Adjusting Detection Rules

Users can customize detection logic such as:

- Threshold of **failed logins**
- Limit for **off-hours activity**
- Suspicion score range
- Monitoring of high-value assets

To adjust rules:

1. Go to **Transform Data → Power Query Editor**
2. Locate the "**Rules**" table
3. Edit values such as:
    - Max failed login attempts
    - Allowed login hours
    - File access risk sensitivity
4. Save and close.

4. Testing and Execution

Testing Data Loading

- Use sample log files included in the project for testing.
- Refresh Power BI to ensure all visuals populate correctly.

Testing Detection Scenarios

The system includes test cases for:

- Suspicious login patterns
- Multiple failed authentication attempts
- Abnormal system usage
- High-volume file transfers
- Privilege misuse

To test:

1. Inject sample anomalies in the CSV logs
2. Refresh the dashboard
3. Observe detection indicators:
    - Alerts
    - Red-colored anomaly visuals
    - Elevated user risk score
    - Anomaly trends line chart

Execution Mode

Once configured, the dashboard can be used in:

- **Live monitoring mode** (connected to streaming data or SharePoint/OneDrive auto-refresh)
- **Offline mode** (manual refresh using logs)

5. Interpretation of Dashboard Modules

Key Dashboards

User Behaviour Overview

Shows:

- Logon frequency
- Working hours
- Accessed applications

Risk Score Heatmap

Highlights:

- High-risk users
- Anomaly clusters
- Privilege escalation events

File Access Insights

Displays:

- Critical file interactions
- Bulk download attempts
- Unauthorized access

Anomaly Detection Panel

Uses ML-based anomaly scoring (Autoencoders/Graph Clustering concept applied) to show:

- Outlier users
- Activity deviation from baseline
- Suspicion trendline

6. Visual Instructions

To assist users with the dashboard:

Provided Visuals Include:

- Screenshot of the Power BI dashboard homepage
- Flow diagram showing data ingestion pipeline:
    - Log Collection → Data Cleaning → Power Query → Analytics → Dashboard
- Step-by-step image guide for:
    - Importing logs
    - Editing data sources
    - Running refresh
    - Navigating reports

5.7 Achievements

During the successful completion of this project, several key achievements were accomplished:

- Developed a fully functional Power BI Insider Threat Detection Dashboard capable of identifying abnormal user behavior patterns.
- Integrated multiple organizational log sources into a unified data model for accurate

behavioral analytics.

- Implemented rule-based anomaly detection indicators such as unusual login times, excessive file access, and repeated failed authentication attempts.
- Designed clean, intuitive, and interactive Power BI visuals to support SOC analysts in quick decision-making.
- Established a complete workflow—from data collection → preprocessing → analytics → visualization—ensuring reliable detection of insider threat indicators.
- Demonstrated improvements in visibility, risk assessment, and early detection compared to traditional monitoring systems**.**