2024, Volume 1, Issue 1, Page N: 30-51.

ISSN (O): 3050-9491
ISSN (P): 3050-9483

World Journal of
Advance Multidisciplinary Research
www.wjamr.org

Research Article

LOG-CORRELATION METHODS FOR EARLY DETECTION OF INSIDER-THREAT INDICATORS IN SMALL-TO-MEDIUM ENTERPRISE NETWORKS

Ifeanyichukwu Uchechukwu Akpara[1]*, Otugene Victor Bamigwojo[2]

*[1]Engineering Department, Auto Blaze Limited, Abuja Nigeria.
[2]Department of Mathematics, Federal University, Lokoja.

Article Info
Article Received: 03 November 2024,
Article Revised: 24 November 2024,
Published on: 01 December 2024

***Corresponding author:**
Ifeanyichukwu Uchechukwu Akpara
Engineering Department, Auto Blaze Limited, Abuja Nigeria.
enyejojoy@gmail.com

https://doi.org/10.5281/zenodo.19416030

How to cite this Article:
Ifeanyichukwu Uchechukwu Akpara1*, Otugene Victor Bamigwojo2. (2024). LOG-CORRELATION METHODS FOR EARLY DETECTION OF INSIDER-THREAT INDICATORS IN SMALL-TO-MEDIUM ENTERPRISE NETWORKS. World Journal of Advanced Multidisciplinary Research, 1(1), 01–22.

ABSTRACT

Insider threats represent one of the most challenging cybersecurity risks for small-to-medium enterprises (SMEs), primarily because malicious activities often originate from legitimate users with authorized access to internal systems. Traditional security monitoring mechanisms frequently rely on isolated log analysis and rule-based alerting, which are inadequate for identifying subtle behavioral anomalies that characterize insider misuse. This study proposes a log-correlation framework designed to enhance early detection of insider-threat indicators by integrating heterogeneous system logs and analysing behavioral relationships across multiple enterprise systems. The framework aggregates authentication logs, endpoint activity records, file access logs, and network connection logs into a centralized monitoring architecture where events are normalized and transformed into structured behavioral features. A correlation scoring model is introduced to quantify relationships among user activities using weighted anomaly indicators derived from multi-source log data. The framework further incorporates probabilistic threat estimation using logistic modeling to estimate the likelihood of insider-threat activity. Experimental evaluation was conducted using a simulated SME network environment consisting of 200 endpoints and multiple distributed log sources. Detection performance was evaluated using precision, recall, and F1-score metrics across several detection models, including rule-based monitoring, statistical anomaly detection, machine learning classifiers, and the proposed correlation-driven approach. Results demonstrate that the proposed log-correlation model improves detection accuracy and significantly reduces detection latency compared with traditional SIEM-based monitoring systems. The findings highlight the importance of multi-log behavioral analytics in identifying coordinated user activities that indicate potential insider misuse. The proposed framework provides a scalable and computationally efficient solution suitable for SME environments where security resources are limited, enabling organizations to detect insider threats earlier and respond more effectively to emerging security risks.

KEYWORDS: Insider threat detection, log correlation analytics, SME cybersecurity, behavioral anomaly detection, security event correlation, enterprise log monitoring.

1. INTRODUCTION

1.1 Background and Context of Insider Threats in SME Networks

Small-to-medium enterprises (SMEs) increasingly rely on digital infrastructure to support business operations, including cloud services, centralized identity management systems, collaborative platforms, and distributed data repositories. While these technologies improve operational efficiency, they also expand the internal attack surface by introducing multiple points where legitimate users can access sensitive organizational resources (Greitzer & Frincke, 2010; Salem et al., 2008). As a result, insider threats have emerged as one of the most challenging cybersecurity risks faced by modern organizations, particularly for SMEs that often lack dedicated security operations teams and sophisticated monitoring infrastructures (CERT Insider Threat Center, 2022).

Insider threats generally arise from three categories of actors: malicious insiders, negligent employees, and compromised accounts. Malicious insiders intentionally exploit their authorized privileges to steal sensitive data, sabotage systems, or commit financial fraud. Negligent insiders, on the other hand, unintentionally expose systems to risk through poor security practices such as

weak password management or improper handling of confidential information. A third category involves external attackers who gain unauthorized access by compromising legitimate credentials, effectively operating as insiders within the network (Cappelli et al., 2012; Nurse et al., 2014). These categories collectively demonstrate that insider threats are not limited to intentional wrongdoing but also include behavioral patterns that inadvertently introduce security vulnerabilities.

Traditional perimeter-based security mechanisms such as firewalls, intrusion detection systems, and network access control technologies are primarily designed to detect external attacks. These systems often rely on predefined rules or signature-based detection models that monitor traffic crossing network boundaries. However, insider activities typically occur within trusted network zones and involve legitimate credentials, making them difficult to detect using conventional perimeter defenses (Eberle & Holder, 2009; Greitzer & Hohimer, 2011). Consequently, malicious or abnormal behaviors frequently remain undetected until significant damage has occurred, including data exfiltration, intellectual property theft, or operational disruption.

For SMEs, the problem is further compounded by structural limitations in cybersecurity capacity. Many small organizations operate with limited budgets and cannot afford advanced security information and event management (SIEM) platforms or dedicated security analysts to continuously monitor system activity. In addition, SME infrastructures are often composed of heterogeneous technologies deployed incrementally over time, leading to fragmented logging mechanisms across authentication servers, endpoint monitoring tools, file access systems, and network flow monitoring platforms (Ahmad et al., 2019). Without integrated monitoring capabilities, it becomes difficult to correlate events across these diverse sources to identify suspicious behavior patterns.

One promising approach to addressing this challenge involves the use of log telemetry analytics. Modern enterprise systems generate extensive digital records describing user activities, system events, and network interactions. These logs provide a valuable source of behavioral evidence that can be analysed to detect deviations from normal usage patterns. For example, authentication logs record login attempts and privilege escalations, file system logs capture access to sensitive documents, endpoint monitoring tools track process execution, and network flow collectors document communication patterns between devices (Behl & Behl, 2017; Conti et al., 2018). When analysed collectively, these logs can reveal correlations that indicate insider threat indicators such as unusual login times, abnormal file access patterns, or atypical network activity.

The effectiveness of such analysis depends on the ability to model and correlate events generated across multiple systems. Let the set of log events produced within an enterprise network be represented as

$$E = \{e_1, e_2, e_3, \ldots, e_n\}$$

where each event e_i is defined as

$$e_i = (u_i, t_i, a_i, s_i)$$

In this representation, u_i denotes the user identity responsible for the activity, t_i represents the timestamp indicating when the event occurred, a_i corresponds to the type of action performed (such as login, file access, or process execution), and s_i identifies the system source from which the log was generated. This mathematical abstraction allows security analysts to treat enterprise logs as structured behavioral datasets that can be analysed using statistical and correlation-based techniques. By examining relationships among events associated with the same user or occurring within similar time windows, it becomes possible to detect abnormal behavioral patterns that may signal emerging insider threats.

Recent cybersecurity research emphasizes that early detection of insider threat indicators depends heavily on multi-source event correlation rather than isolated log analysis (Eberle & Holder, 2009; Legg et al., 2015). Correlating authentication logs with endpoint activity records and network flow data enables the construction of contextual behavioral models capable of identifying coordinated anomalies. For example, an employee account that successfully authenticates outside normal working hours and immediately begins accessing large volumes of confidential files may indicate potential data exfiltration activity. Detecting such patterns requires integrated log analytics frameworks capable of aggregating, normalizing, and correlating events across distributed systems.

In SME environments, lightweight log-correlation techniques provide a practical alternative to resource-intensive enterprise security platforms. By leveraging centralized log aggregation combined with statistical correlation models, SMEs can implement scalable monitoring systems capable of detecting early insider threat indicators without requiring extensive infrastructure investments. This research therefore investigates log-correlation methods designed specifically for SME network environments, with the goal of enabling proactive identification of insider threat behaviors through efficient analysis of system-generated event logs. Recent advancements in data-driven decision support systems have played a significant role in improving manufacturing productivity by optimizing operational workflows and supporting decision-making processes (Jalloh & Bamigwojo, 2023). These systems integrate real-time data and advanced analytics to provide actionable insights that enhance manufacturing processes.

1.2 Problem Statement

Small-to-medium enterprises (SMEs) face increasing cybersecurity challenges due to the rapid digitalization of business operations and the adoption of distributed computing infrastructures. Although many SMEs deploy basic security mechanisms such as firewalls, antivirus tools, and endpoint protection systems, their internal monitoring capabilities remain limited compared to those of large enterprises (Ahmad et al., 2019; Behl & Behl, 2017). One major limitation is the reliance on isolated log monitoring systems, where individual platforms generate logs independently without centralized correlation or integrated analysis. Authentication servers, file management systems, network devices, and endpoint monitoring tools often maintain separate logging frameworks that are rarely aggregated into a unified analytical environment. As a result, potentially malicious user activities may appear benign when viewed within a single log source but reveal suspicious behavioral patterns when correlated across multiple systems (Greitzer & Hohimer, 2011; Legg et al., 2015).

The absence of cross-source correlation analytics significantly limits the ability of SMEs to detect insider threats at an early stage. Insider attacks often unfold through a sequence of events distributed across multiple systems. For example, an attacker using compromised credentials may first authenticate through an identity management system, subsequently access sensitive files from a shared repository, and later transmit the extracted data through unusual network connections. When these events are analysed independently, each activity may fall within normal operational thresholds; however, when correlated temporally and contextually, the sequence can indicate coordinated malicious behavior (Eberle & Holder, 2009; Nurse et al., 2014). Research on insider threat detection emphasizes that effective monitoring requires multi-source event correlation, where relationships among authentication logs, system activity records, and network traffic data are analysed collectively to identify anomalies (Cappelli et al., 2012; Salem et al., 2008). Unfortunately, such analytical capabilities are rarely implemented in SME environments due to technical and financial constraints.

Another significant barrier is the limited accessibility of Security Information and Event Management (SIEM) platforms. SIEM technologies are widely used in large enterprises to aggregate and analyse security logs, providing automated alerting, threat intelligence integration, and advanced behavioral analytics. However, the deployment of SIEM infrastructures typically requires substantial financial investment, specialized personnel, and high computational resources for log storage and processing (Chuvakin et al., 2013; Peltier, 2016). SMEs often lack the resources necessary to maintain such complex infrastructures, making enterprise-grade SIEM solutions impractical for their operational environments. In addition, traditional SIEM architectures rely on large-scale data ingestion pipelines and complex rule-based correlation engines, which can impose significant processing overhead and generate high volumes of false-positive alerts (Conti et al., 2018).

The computational burden associated with large-scale log analysis also presents a challenge for SMEs operating with limited IT infrastructure. Continuous monitoring systems must process high volumes of log data generated by network devices, cloud platforms, endpoint agents, and application servers. Without efficient data processing techniques, the storage and analysis of these logs can overwhelm available system resources, leading organizations to adopt minimal monitoring practices that only retain partial logs or perform retrospective forensic analysis after incidents occur (Tuor et al., 2017). This reactive approach significantly reduces the ability to detect insider threats in real time, thereby increasing the risk of data breaches and operational disruption.

Furthermore, many SMEs lack standardized frameworks for log normalization and event correlation, which are essential for integrating heterogeneous log sources into a unified analytical environment. Logs generated by different systems typically vary in structure, format, and semantic meaning. Without proper normalization and correlation mechanisms, it becomes difficult to construct coherent activity timelines that capture the relationships among user actions across multiple systems (Sharma & Dash, 2019). Consequently, insider threat indicators remain hidden within fragmented datasets, preventing early identification of suspicious behavioral patterns.

These challenges collectively highlight the need for lightweight log-correlation methodologies tailored to SME network environments. Such approaches must address three key requirements: efficient integration of heterogeneous log sources, scalable correlation analysis capable of identifying anomalous behavioral sequences, and computational efficiency suitable for resource-constrained infrastructures. By developing cost-effective correlation frameworks that leverage existing system logs, SMEs can significantly improve their ability to detect insider threat indicators without relying on expensive enterprise-grade security platforms. This study therefore investigates log-correlation techniques designed to support early detection of insider threats in SME networks while minimizing computational overhead and infrastructure requirements.

1.3 Research Objectives

The increasing sophistication of insider threats and the limitations of traditional security monitoring systems necessitate the development of analytical frameworks capable of detecting suspicious behavioral patterns within enterprise networks. Small-to-medium enterprises (SMEs) face particular challenges in implementing advanced threat detection mechanisms due to limited financial resources, constrained computational infrastructure, and

the absence of dedicated cybersecurity teams. Consequently, there is a growing need for lightweight yet effective analytical approaches capable of extracting meaningful insights from system-generated logs. In response to this challenge, the present study aims to investigate log-correlation methods for the early detection of insider-threat indicators in SME network environments.

The first objective of this research is to develop a log-correlation framework for insider-threat detection that integrates multiple sources of system telemetry. Modern enterprise networks generate diverse forms of security logs, including authentication records, endpoint monitoring logs, file system access logs, and network flow records. When analyzed individually, these logs provide limited visibility into user behavior. However, when aggregated and correlated across multiple systems, they can reveal complex activity patterns indicative of malicious or anomalous behavior. The proposed framework therefore focuses on designing a centralized event aggregation architecture capable of collecting, normalizing, and correlating heterogeneous log streams. This integration enables the identification of behavioral relationships among events generated by different components of the network infrastructure. By correlating these events temporally and contextually, the framework aims to reveal coordinated sequences of actions that may represent insider threat activities.

The second objective of this research is to design mathematical models for anomaly scoring based on multi-source log aggregation. Insider threat detection requires analytical techniques capable of quantifying deviations from normal user behavior. In this study, system logs are treated as structured event sequences that can be modeled mathematically. Let the set of observed log events within a network be represented as

$$E = \{e_1, e_2, e_3, \ldots, e_n\}$$

where each event e_i is characterized by attributes including the user identity, timestamp, action type, and originating system. These events are transformed into feature vectors that capture behavioral indicators such as login frequency, access patterns to sensitive resources, and abnormal network communication activity. Using these features, anomaly scoring models are constructed to evaluate the likelihood that a sequence of events corresponds to suspicious insider activity. The anomaly score for a user session can be expressed as

$$S_u = \sum_{i=1}^{m} w_i x_i$$

where x_i represents a behavioral feature derived from correlated logs and w_i denotes the corresponding weighting factor reflecting its relative importance in detecting malicious behavior. This mathematical formulation enables the system to quantify deviations from expected behavior patterns and generate risk scores associated with specific users or sessions.

The third objective of the study is to evaluate the detection performance of the proposed framework using SME-scale network datasets. Performance evaluation is essential to determine the effectiveness and practicality of the proposed log-correlation methodology. In this research, experimental simulations are conducted using network environments that approximate the scale and operational characteristics of typical SME infrastructures. The evaluation focuses on measuring key performance metrics commonly used in cybersecurity detection systems, including precision, recall, detection accuracy, and false-positive rates. These metrics can be mathematically expressed as

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

where TP represents true positive detections, FP represents false positives, and FN denotes false negatives. By analyzing these performance indicators across different detection scenarios, the study aims to determine the reliability and scalability of the proposed log-correlation framework.

Collectively, these research objectives establish the foundation for developing a practical and computationally efficient insider-threat detection system tailored to the operational constraints of SMEs. By integrating multi-source log analytics, mathematical anomaly scoring models, and empirical performance evaluation, the study seeks to provide a robust methodological framework for identifying insider-threat indicators at an early stage, thereby improving the overall cybersecurity resilience of SME networks.

1.4 Research Contributions

This study contributes to the field of cybersecurity analytics by proposing a structured approach for detecting insider-threat indicators through correlation-based analysis of enterprise log data. As insider threats increasingly exploit legitimate credentials and internal system access, traditional perimeter-based security controls often fail to identify abnormal user behaviors within organizational networks. The contributions of this research therefore focus on developing analytical models that enable early detection of suspicious activities through the integration and mathematical analysis of heterogeneous log sources within small-to-medium enterprise (SME) environments.

The first major contribution of this research is the development of a correlation-driven insider-threat detection model designed specifically for SME network

infrastructures. The model integrates multiple categories of log telemetry, including authentication records, endpoint monitoring logs, file access histories, and network flow data. Instead of analysing these logs independently, the proposed framework aggregates them into a unified event stream that enables cross-system correlation of user activities. By analysing temporal and contextual relationships among events, the model identifies behavioral patterns that may indicate insider threat activity, such as unusual access to sensitive files immediately following abnormal authentication events. This correlation-driven approach improves situational awareness by revealing activity sequences that remain undetected when individual logs are analyzed in isolation.

The second contribution of this research is the mathematical formulation of event dependency graphs for insider-threat analysis. In the proposed framework, enterprise log events are represented as nodes within a graph-based structure where relationships among events are modeled as dependency edges. Let the event dependency graph be defined as

$$G = (V, E)$$

where V represents the set of log events and E represents the relationships among events derived from temporal proximity, user identity similarity, or shared system context. Each event node v_i corresponds to a log entry defined by attributes such as user identity, timestamp, activity type, and system source. Dependency edges are established when two events exhibit meaningful relationships within a defined correlation window. The strength of these relationships can be represented by a correlation weight

$$w_{ij} = \frac{| A_i \cap A_j |}{| A_i \cup A_j |}$$

where A_i and A_j represent the sets of behavioral attributes associated with events i and j. This formulation enables the construction of behavioral graphs that capture complex sequences of user activities across multiple systems. Suspicious behavioral patterns can then be detected by analyzing abnormal graph structures, high-risk event clusters, or unusual transitions between event states.

The third contribution of this study involves empirical performance validation of the proposed detection framework using standard cybersecurity evaluation metrics. To assess the effectiveness of the correlation-driven detection model, experimental evaluation is conducted using SME-scale network datasets that simulate realistic enterprise activity patterns. The performance of the proposed method is measured using widely accepted classification metrics, including precision, recall, and detection latency. Precision quantifies the proportion of correctly identified insider-threat events relative to the total number of alerts generated by the system, while recall measures the proportion of actual insider-threat activities successfully detected by the model. These metrics are formally expressed as

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

where TP represents true positives, FP represents false positives, and FN represents false negatives. In addition to these metrics, the study evaluates detection latency, which measures the time required for the system to identify suspicious activity after the initial event occurs. Minimizing detection latency is critical in insider-threat scenarios, as early identification allows security teams to intervene before significant data exfiltration or system compromise takes place.

Together, these contributions establish a comprehensive framework for insider-threat detection that combines multi-source log correlation, graph-based behavioral modeling, and empirical performance evaluation. By focusing on scalable analytical techniques suitable for resource-constrained SME environments, the study provides a practical foundation for implementing effective insider-threat monitoring systems without requiring complex or costly enterprise security infrastructures.

Figure 1 illustrates the proposed system architecture designed for detecting insider-threat indicators in small-to-medium enterprise (SME) networks through multi-source log correlation. The architecture begins with distributed endpoint log sources, which include workstation monitoring agents, application logs, file-system access records, and network flow collectors deployed across the enterprise infrastructure. These endpoints continuously generate telemetry describing user actions, system processes, and network communications.

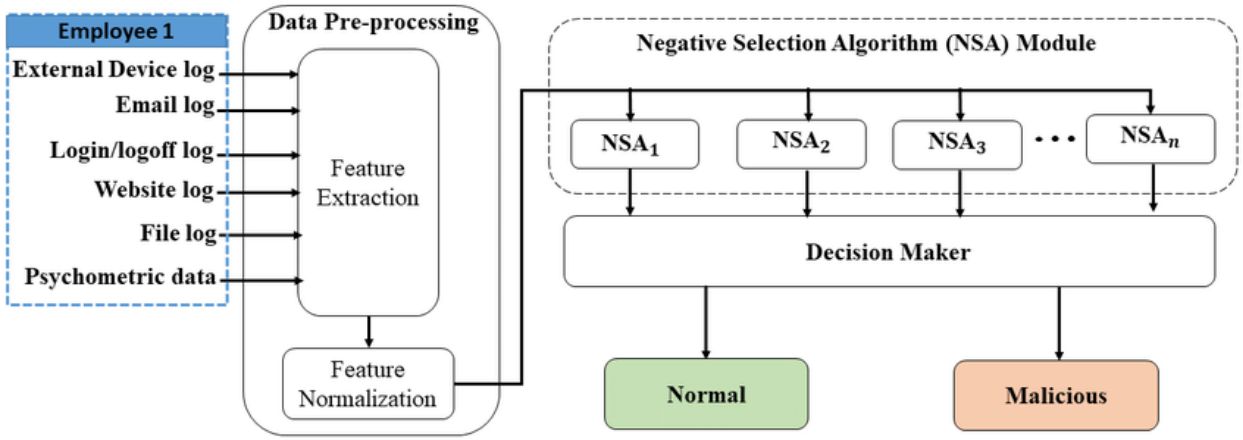

Figure 1: System Architecture for SME Insider-Threat Log Correlation Framework.

The generated telemetry is transmitted to authentication servers and identity management systems, which produce additional logs related to login events, privilege escalation attempts, session durations, and credential validation outcomes. These heterogeneous log streams are forwarded to a centralized log collector, responsible for aggregating, timestamp-synchronizing, and normalizing event records originating from different systems. Log normalization ensures that event attributes such as user identifiers, activity types, and timestamps are standardized across data sources.

The normalized event streams are then processed by the log-correlation engine, which constructs relationships among events based on temporal proximity, shared user identity, and activity dependencies. The correlation engine generates structured behavioral event graphs that capture sequences of user actions occurring across multiple systems. These correlated event structures are forwarded to the anomaly detection module, where statistical and behavioral models evaluate deviations from expected user activity patterns.

Finally, detection results are presented through a security analyst dashboard, which provides visualization of suspicious activities, anomaly scores, and correlated event chains. The dashboard enables security analysts to investigate potential insider-threat indicators, review event timelines, and initiate incident response procedures when abnormal activity patterns are detected. This architecture enables SMEs to implement a scalable and cost-efficient monitoring system capable of identifying insider-threat behaviors through integrated log analytics and correlation-based anomaly detection.

2. Literature Review

2.1 Insider Threat Detection Techniques

Insider threat detection has become an important research area in cybersecurity due to the increasing reliance of organizations on digital infrastructures and the growing complexity of internal network environments. Unlike external attacks that originate outside the organizational perimeter, insider threats exploit legitimate access privileges, making detection particularly challenging for conventional security systems. As a result, several detection approaches have been proposed in the literature, including rule-based monitoring, behavioral analytics, anomaly detection models, and security information and event management (SIEM)-driven monitoring systems (Greitzer & Hohimer, 2011; Nurse et al., 2014). Each of these techniques contributes to the identification of suspicious activities within enterprise networks; however, their effectiveness varies significantly depending on the operational environment and available computational resources.

One widely used approach in organizational security monitoring is rule-based detection, which relies on predefined policies or signatures to identify suspicious activities. Rule-based monitoring systems evaluate system events against a set of predefined conditions such as repeated login failures, unauthorized privilege escalation attempts, or abnormal file access patterns. When a rule is triggered, the system generates an alert for further investigation (Behl & Behl, 2017). Although rule-based approaches are relatively straightforward to implement and computationally efficient, they suffer from limited adaptability. Insider threats often involve subtle behavioral changes that may not match predefined rule patterns, allowing sophisticated attackers to evade detection by operating within the boundaries of normal system activity (Cappelli et al., 2012). Consequently, rule-based detection mechanisms are generally more effective for identifying known attack signatures than for detecting novel insider behaviors.

To address the limitations of rule-based systems, researchers have increasingly explored behavioral analytics for insider threat detection. Behavioral analytics focuses on modeling typical user activity patterns and identifying deviations that may indicate malicious or suspicious behavior. These models often incorporate attributes such as login frequency, file access patterns, command execution histories, and network usage behaviors (Greitzer & Frincke, 2010). By constructing baseline behavioral profiles for individual users or roles,

behavioral analytics systems can detect anomalies that deviate from normal operational patterns. However, implementing behavioral analytics in enterprise environments requires extensive data collection and continuous model updates to account for evolving user behaviors. In SME environments where monitoring resources are limited, maintaining accurate behavioral models can be particularly challenging (Legg et al., 2015).

Another important class of insider threat detection techniques involves anomaly detection models, which apply statistical or machine learning algorithms to identify unusual patterns within system logs and network traffic. These approaches treat insider threat detection as a data analysis problem in which abnormal activities are identified based on deviations from expected statistical distributions. Techniques such as clustering, classification, and probabilistic modeling have been widely applied to detect anomalies in user behavior datasets (Eberle & Holder, 2009). Machine learning-based approaches can capture complex behavioral patterns that are difficult to express through predefined rules. However, these models often require large datasets for training and may generate high false-positive rates when normal user behavior exhibits significant variability (Tuor et al., 2017). This challenge is particularly pronounced in SME environments where limited historical data is available for training accurate detection models.

In enterprise environments, insider threat monitoring is often implemented through Security Information and Event Management (SIEM) systems. SIEM platforms aggregate logs from multiple systems, including authentication servers, network devices, endpoint monitoring tools, and application servers. These platforms provide centralized event analysis, correlation engines, and automated alerting mechanisms that support real-time threat detection (Chuvakin et al., 2013). By integrating multiple log sources into a single analytical environment, SIEM systems enable organizations to detect complex attack patterns that span multiple systems. For example, a SIEM system may correlate a suspicious login event with subsequent access to sensitive files and unusual outbound network traffic to detect potential data exfiltration activities.

Despite their effectiveness in large enterprise environments, SIEM systems present several limitations when applied in SME contexts. One major limitation is computational cost. SIEM platforms require significant processing power to ingest, store, and analyze large volumes of log data generated by modern enterprise systems. The cost of deploying and maintaining such infrastructure can be prohibitive for SMEs with limited IT budgets (Ahmad et al., 2019). In addition, SIEM systems often rely on complex rule engines that require specialized expertise to configure and maintain, further increasing operational costs.

Another limitation commonly reported in the literature is the high false-positive rate associated with many insider threat detection systems. Behavioral anomalies do not always correspond to malicious activities; employees may legitimately access systems outside their normal working hours or download large files for legitimate work purposes. When detection models incorrectly classify these activities as malicious, security teams may experience alert fatigue, reducing the effectiveness of monitoring systems (Greitzer & Hohimer, 2011). For SMEs with small IT teams, the burden of investigating numerous false alerts can significantly reduce operational efficiency.

Finally, many existing detection frameworks lack effective cross-log correlation capabilities. Although enterprise systems generate extensive logs from various components, these logs often remain isolated within individual platforms. Without mechanisms for correlating events across authentication systems, endpoint monitoring tools, and network flow collectors, it becomes difficult to identify coordinated attack patterns involving multiple systems (Nurse et al., 2014). This limitation is particularly problematic in insider threat scenarios where malicious activities typically involve sequences of events distributed across different systems.

Given these limitations, recent research emphasizes the need for scalable log-correlation techniques capable of integrating heterogeneous log sources while maintaining computational efficiency suitable for SME environments. Developing such techniques requires combining lightweight event aggregation architectures with efficient correlation algorithms that can detect suspicious behavioral sequences without imposing excessive computational overhead. This study therefore builds upon existing insider threat detection research by proposing a correlation-driven framework specifically designed to address the operational constraints and monitoring challenges faced by SMEs.

2.2 Log Analysis and Security Event Correlation

Security log analysis plays a critical role in modern cybersecurity monitoring by enabling organizations to identify abnormal behavioral patterns across distributed information systems. Enterprise networks continuously generate large volumes of event logs from multiple components, including authentication servers, operating systems, application services, endpoint monitoring agents, and network devices. These logs provide detailed records of system activities such as login attempts, file access operations, command executions, and network communications. When analysed collectively, these records can reveal behavioral patterns associated with malicious or suspicious activities, particularly those involving insider threats (Behl & Behl, 2017; Chuvakin et al., 2013).

Traditional log analysis approaches often rely on isolated monitoring mechanisms in which individual systems

analyse their own logs independently. While this method provides basic visibility into local system activity, it fails to capture complex interactions that occur across multiple systems within enterprise networks. Insider threats frequently manifest as coordinated sequences of actions involving several systems, such as abnormal authentication events followed by unauthorized file access and unusual network transfers. Detecting such multi-stage activities requires analytical frameworks capable of correlating events across heterogeneous log sources (Greitzer & Hohimer, 2011; Nurse et al., 2014).

Log correlation techniques address this challenge by identifying relationships between events generated by different systems within a defined temporal or contextual window. Correlation methods evaluate similarities among event attributes such as user identity, timestamp proximity, activity type, and system origin. By establishing relationships among these attributes, correlation algorithms construct behavioral sequences that represent user activity patterns across multiple platforms (Eberle & Holder, 2009). These relationships allow security analysts to identify coordinated activities that may indicate potential insider misuse or policy violations.

Mathematically, the correlation between two event sets generated by different systems can be represented as

$$C(E_i, E_j) = \frac{|\ T_i \cap T_j\ |}{|\ T_i \cup T_j\ |}$$

where T_i and T_j represent activity sets derived from two different systems or log sources. The numerator $|\ T_i \cap T_j\ |$ represents the number of common activities or shared attributes observed across both systems, while the denominator $|\ T_i \cup T_j\ |$ represents the total number of unique activities observed across the combined event sets. This formulation is conceptually similar to similarity metrics used in pattern recognition and enables the quantification of behavioral overlap between system logs. Higher correlation values indicate stronger relationships between activities recorded in the two systems, suggesting the possibility of coordinated actions or anomalous behavior sequences (Legg et al., 2015).

In insider-threat detection scenarios, high correlation scores may indicate suspicious activity patterns. For instance, if authentication logs reveal an unusual login outside normal working hours and file system logs simultaneously record large volumes of confidential file access by the same user account, the correlation between these event sets may indicate potential insider misuse. Similarly, correlated patterns between endpoint activity logs and network flow records may reveal unauthorized data exfiltration attempts. By systematically analysing correlations among event sources, organizations can construct contextual behavioral models that support early detection of insider threats (Salem et al., 2008).

Despite the effectiveness of correlation-based monitoring, implementing such techniques in small-to-medium enterprise environments presents several challenges. Log sources often differ significantly in format, structure, and semantic meaning, requiring extensive preprocessing and normalization before correlation analysis can be performed. Furthermore, correlation algorithms must operate efficiently to process large volumes of event data in near real-time, particularly in dynamic enterprise environments where thousands of events may be generated each minute (Ahmad et al., 2019). These challenges highlight the need for scalable log-correlation frameworks capable of integrating heterogeneous data sources while maintaining computational efficiency suitable for SME infrastructures.

The existing literature presents several techniques for insider-threat detection, each relying on different data sources and analytical approaches. These methods vary in effectiveness depending on the availability of system telemetry and the computational resources required for implementation. To provide a structured overview of these techniques, Table 1 presents a comparative review of commonly used insider-threat detection methods, highlighting their primary data sources, operational strengths, and limitations.

Table 1: Comparative Review of Insider-Threat Detection Techniques.

Detection Method	Data Sources Used	Strengths	Limitations
Rule-Based Monitoring	Authentication logs, system alerts, network activity logs	Simple implementation and fast detection of known threats	Limited ability to detect unknown or evolving insider behaviors
Behavioral Analytics	User activity logs, login records, file access logs	Captures deviations from normal user behavior	Requires continuous behavioral profiling and large training datasets
Statistical Anomaly Detection	System logs, network traffic records, endpoint activity	Identifies unusual patterns using statistical modeling	High false-positive rates when user behavior varies significantly
Machine Learning Detection	Aggregated logs, endpoint telemetry, network data	Detects complex behavioral patterns through data-driven	Requires large labelled datasets and computational

		models	resources
SIEM-Based Detection	Multi-source enterprise logs including network, system, and application logs	Provides centralized monitoring and event correlation capabilities	High deployment cost and complex infrastructure requirements
Detection Method	Data Sources Used	Strengths	Limitations

2.3 Log Correlation Techniques

Log correlation techniques are widely used in cybersecurity monitoring to identify relationships among events generated by multiple systems within enterprise networks. In modern computing environments, organizations deploy various technologies that continuously produce log data, including authentication systems, endpoint monitoring agents, file servers, and network infrastructure devices. Individually, these logs provide only partial insights into user behavior and system activities. However, when analyzed collectively through correlation mechanisms, they can reveal complex behavioral patterns that indicate security threats such as insider misuse, privilege abuse, or unauthorized data exfiltration (Chuvakin et al., 2013; Greitzer & Hohimer, 2011). Log correlation techniques therefore enable security analysts to reconstruct sequences of events that span multiple systems and identify coordinated activities that may not be visible through isolated log analysis.

One widely used approach is temporal correlation, which examines the timing relationships among events occurring within a defined observation window. Temporal correlation assumes that suspicious activities often occur in sequences where multiple events happen within a short period of time. For example, a login event followed immediately by privilege escalation and large file transfers may indicate potential insider misuse. Temporal correlation algorithms evaluate event timestamps and detect patterns of closely related activities that deviate from expected operational behavior (Eberle & Holder, 2009). This approach is particularly useful for detecting multi-stage attacks where adversaries perform several actions in quick succession after gaining access to internal systems.

Another commonly used technique is rule-based correlation, which extends traditional rule-based monitoring by correlating events from multiple systems rather than evaluating them individually. In this approach, correlation rules define logical relationships among events across different log sources. For instance, a rule may trigger an alert when a successful authentication event is followed by multiple failed authorization attempts within a short time window. Rule-based correlation systems are widely implemented in security information and event management (SIEM) platforms, where predefined correlation rules enable automated detection of known threat patterns (Behl & Behl, 2017). While rule-based correlation is effective for identifying well-defined attack scenarios, it remains limited in detecting novel or evolving insider behaviors that do not match predefined rules.

A more advanced technique involves graph-based event linkage, where log events are represented as nodes in a graph structure and correlations between events are modeled as edges connecting related nodes. This approach allows analysts to represent complex behavioral relationships among events generated across multiple systems. The event dependency graph can be formally expressed as

$$G = (V, E)$$

where V represents the set of log events and E represents the set of correlation edges connecting events that exhibit meaningful relationships. In this framework, nodes correspond to individual log entries, while edges represent relationships derived from shared attributes such as user identity, activity type, or temporal proximity. Graph-based models enable the identification of behavioral clusters, event chains, and suspicious activity pathways that may indicate insider threat behavior (Legg et al., 2015). By analyzing structural patterns within event graphs, cybersecurity systems can detect abnormal sequences of actions that deviate from normal operational workflows.

In addition to graph-based techniques, probabilistic correlation models have been introduced to address the uncertainty associated with security event analysis. Probabilistic models evaluate the likelihood that a sequence of events represents malicious behavior based on statistical relationships among log attributes. These approaches often use Bayesian inference, hidden Markov models, or probabilistic graphical models to estimate the probability of security incidents given observed system events (Tuor et al., 2017). Probabilistic correlation techniques are particularly useful for insider threat detection because they can incorporate contextual information and historical behavioral patterns to estimate the likelihood of suspicious activities. However, these methods require significant computational resources and large training datasets, which may limit their applicability in resource-constrained SME environments.

Effective implementation of log-correlation techniques requires the integration of multiple log sources that capture different aspects of system activity. Authentication logs provide information about login events and credential usage, endpoint monitoring tools capture process execution and system-level interactions, file access logs track user interactions with sensitive data repositories, and network logs document communication patterns among devices. Each of these sources contributes valuable contextual information that can support insider threat detection when analyzed collectively. To illustrate the role

of these log sources, Table 2 summarizes common log sources used in insider-threat detection, along with the types of events they generate and examples of threat indicators that may be derived from them.

Table 2: Log Sources Commonly Used in Insider-Threat Detection.

Log Source	Event Type	Detection Relevance	Example Threat Indicator
Authentication Logs	Login attempts, session start/end, credential validation	Identifies abnormal login patterns and unauthorized access attempts	Login outside normal working hours or repeated authentication failures
File Access Logs	File creation, modification, download, deletion	Monitors access to sensitive organizational data	Large-scale downloading of confidential files
Endpoint Monitoring Logs	Process execution, application usage, system commands	Detects suspicious system-level activities	Execution of unauthorized scripts or administrative tools
Network Traffic Logs	IP connections, data transfer volumes, session durations	Reveals unusual network communication patterns	Data exfiltration to external or unknown IP addresses
Privilege Management Logs	Role assignments, privilege escalation events	Detects unauthorized privilege changes	Sudden escalation to administrator privileges
Log Source	Event Type	Detection Relevance	Example Threat Indicator

2.4 Research Gap

Despite significant advancements in insider-threat detection research, several limitations remain in existing log-analysis and correlation-based security monitoring systems. Modern cybersecurity frameworks often rely on large-scale enterprise infrastructures and computationally intensive analytics platforms, making them less suitable for small-to-medium enterprise (SME) environments. Although various detection approaches—including behavioral analytics, anomaly detection, and machine-learning models—have been proposed, many of these systems are designed for large organizations with extensive computational resources and dedicated security operations teams (Al-Mhiqani et al., 2020; Subhani, 2021). Consequently, SMEs frequently struggle to adopt advanced insider-threat detection mechanisms due to resource constraints and the complexity of existing analytical frameworks.

One major gap identified in the literature is the lack of low-complexity correlation algorithms specifically tailored for SME infrastructures. Many existing detection frameworks rely on complex machine-learning pipelines or large-scale SIEM deployments that require substantial computing power, storage capacity, and specialized expertise. While such systems are effective in large enterprise networks, their implementation costs and operational overhead often exceed the capabilities of smaller organizations (Ahmad et al., 2019). In practice, SMEs often deploy only basic monitoring tools, which typically analyse logs in isolation rather than correlating events across multiple systems. Without lightweight correlation algorithms capable of integrating heterogeneous log sources efficiently, these organizations remain vulnerable to insider threats that exploit gaps in monitoring coverage.

Another important limitation concerns the limited empirical evaluation of multi-log behavioral models. Existing research frequently proposes theoretical models for insider-threat detection based on log analytics, behavioral profiling, or anomaly detection. However, many of these models have been validated primarily using synthetic datasets or small experimental environments rather than realistic SME-scale network infrastructures (Pelevin, 2021). As a result, the practical applicability of these models in real-world SME environments remains uncertain. Studies have shown that insider-threat detection systems must process large volumes of heterogeneous log data while maintaining acceptable levels of detection accuracy and computational efficiency (Tuor et al., 2017). Without empirical validation across diverse operational contexts, it is difficult to determine whether proposed models can effectively detect insider threats in real-time organizational settings.

Furthermore, many existing systems focus on single-source log analysis, which limits their ability to identify coordinated activities that span multiple systems. Insider attacks often involve sequences of events occurring across authentication systems, file servers, endpoint devices, and network infrastructure. When these events are analyzed independently, suspicious activity patterns may remain undetected. Log-correlation techniques attempt to address this limitation by combining data from multiple sources, but current implementations frequently require sophisticated infrastructure and complex integration processes (Chuvakin et al., 2013). In SME environments, where IT systems are often heterogeneous and loosely integrated, implementing such correlation mechanisms can be particularly challenging.

Another challenge identified in the literature is the difficulty of balancing detection accuracy and false-positive rates. Insider threat detection systems must distinguish between malicious activities and legitimate user behaviors, which often appear similar in system logs. Advanced detection algorithms may improve accuracy but also introduce computational complexity that increases

processing overhead and delays detection. Conversely, simpler detection mechanisms may operate efficiently but generate excessive false alerts, leading to alert fatigue among security administrators (Greitzer & Hohimer, 2011). These trade-offs highlight the need for detection frameworks that achieve an appropriate balance between analytical sophistication and computational efficiency.

Given these challenges, there is a clear need for efficient log-correlation frameworks specifically designed for SME network environments. Such frameworks should incorporate lightweight correlation algorithms capable of integrating heterogeneous log sources while maintaining manageable computational requirements. Additionally, empirical evaluation using realistic SME-scale datasets is necessary to validate the effectiveness of multi-log behavioral models in operational environments. Addressing these gaps would significantly improve the practicality of insider-threat detection systems and enhance the cybersecurity resilience of SMEs by enabling early identification of suspicious user behaviors through scalable log-correlation analytics.

3. METHODOLOGY

3.1 System Model for Log Aggregation

The proposed insider-threat detection framework relies on a centralized log aggregation and correlation architecture designed to integrate heterogeneous event streams generated across SME network infrastructures. Enterprise computing environments typically produce large volumes of telemetry data from multiple subsystems, including authentication services, endpoint monitoring agents, file management systems, and network devices. These logs capture detailed information about user activities and system interactions that can be analyzed to identify abnormal behavioral patterns indicative of insider misuse (Chuvakin et al., 2013; Greitzer & Hohimer, 2011).

The system model adopted in this study integrates four primary categories of log data: authentication logs, endpoint activity logs, file access logs, and network connection logs. Authentication logs provide records of login attempts, credential validation events, and session establishment activities generated by identity management systems. Endpoint activity logs capture system-level interactions such as process execution, application usage, and command invocation on user workstations. File access logs document user interactions with organizational data repositories, including file creation, modification, download, and deletion events. Network connection logs record communication flows between internal hosts and external systems, including IP addresses, session durations, and data transfer volumes. Together, these heterogeneous data sources provide complementary views of user behavior within enterprise networks and form the foundation for correlation-based insider-threat detection (Legg et al., 2015).

Let the set of all observed log events within the enterprise environment be represented as

$$E = \{e_1, e_2, e_3, \dots, e_n\}$$

where each event e_i is defined as a multidimensional tuple describing the contextual attributes of a system activity:

$$e_i = (u_i, t_i, a_i, s_i, r_i)$$

Here, u_i represents the user identity associated with the event, t_i denotes the timestamp, a_i indicates the activity type (e.g., login, file access, process execution), s_i represents the system source generating the log entry, and r_i corresponds to additional contextual attributes such as resource identifiers or network endpoints. This formal representation enables the transformation of heterogeneous log records into a unified analytical structure suitable for correlation analysis and anomaly detection.

Because logs generated by different systems often differ in format, structure, and semantic meaning, preprocessing is required to ensure consistency across event streams. This process is referred to as event normalization, which transforms raw log entries into standardized representations that can be analyzed collectively. Event normalization is mathematically represented as

$$e_i' = f(e_i)$$

where $f(\cdot)$ denotes a transformation function that maps raw event attributes into a normalized feature space. This transformation may involve timestamp synchronization, user identifier mapping, attribute encoding, and removal of redundant metadata. The normalized event set can therefore be expressed as

$$E' = \{e_1', e_2', e_3', \dots, e_n'\}$$

which forms the standardized dataset used by the correlation engine.

Following normalization, the system extracts behavioral features from the normalized event records to construct activity profiles for individual users or system entities. Each normalized event e_i' is mapped into a feature vector representing behavioral characteristics observed within the system. The feature extraction process can be expressed as

$$x_i = \phi(e_i')$$

where $\phi(\cdot)$ represents the feature extraction function and x_i denotes the resulting feature vector describing the event's behavioral attributes. These attributes may include login frequency, file access intensity, privilege escalation occurrences, and network connection patterns. Aggregating these features across user sessions enables the construction of behavioral models capable of

identifying deviations from normal operational patterns (Tuor et al., 2017).

To capture relationships among events generated across different systems, the proposed framework constructs a correlation graph in which normalized events are represented as nodes and relationships between events are represented as edges. The correlation graph can be defined as

$$G = (V, E)$$

where V represents the set of normalized log events and E represents correlation edges linking events that share meaningful relationships. Two events e_i' and e_j' are connected when they satisfy predefined correlation conditions such as temporal proximity, shared user identity, or resource dependency. The strength of the correlation between two events can be quantified using a similarity metric defined as

$$w_{ij} = \exp\left(-\frac{| t_i - t_j |}{\tau} \right) \cdot \delta(u_i, u_j)$$

where $| t_i - t_j |$ represents the temporal distance between events, τ is a temporal decay parameter, and $\delta(u_i, u_j)$ is an indicator function that equals 1 when both events are associated with the same user identity and 0 otherwise. This weighting function allows the correlation engine to prioritize events occurring within close temporal proximity and associated with the same user context.

The resulting event graph enables the detection of suspicious behavioral patterns through graph-based anomaly scoring mechanisms. The anomaly score associated with a user activity sequence is computed by aggregating the weighted correlations of events linked to that user within the graph structure. The anomaly score for a user session u can be expressed as

$$S_u = \sum_{(i,j) \in E_u} w_{ij} \cdot \psi(x_i, x_j)$$

where E_u represents the set of correlated event pairs associated with user u, w_{ij} represents the correlation weight between events i and j, and $\psi(\cdot)$ is a behavioral divergence function that measures deviation between observed feature vectors and baseline activity profiles. Higher anomaly scores indicate greater deviation from expected behavioral patterns and therefore a higher likelihood of insider-threat activity.

The system model therefore integrates heterogeneous log sources into a unified analytical pipeline consisting of event ingestion, normalization, feature extraction, correlation graph construction, and anomaly scoring. This architecture enables efficient identification of suspicious activity sequences across distributed systems within SME networks while maintaining computational efficiency suitable for resource-constrained environments.

Figure 2 illustrates the operational workflow of the proposed log-correlation engine and event processing pipeline. The process begins with log ingestion, where raw telemetry data from distributed enterprise systems including authentication servers, endpoint monitoring agents, file access systems, and network devices are collected and transmitted to a centralized processing environment. During this stage, the system aggregates heterogeneous event streams into a unified data repository capable of supporting large-scale event analysis.

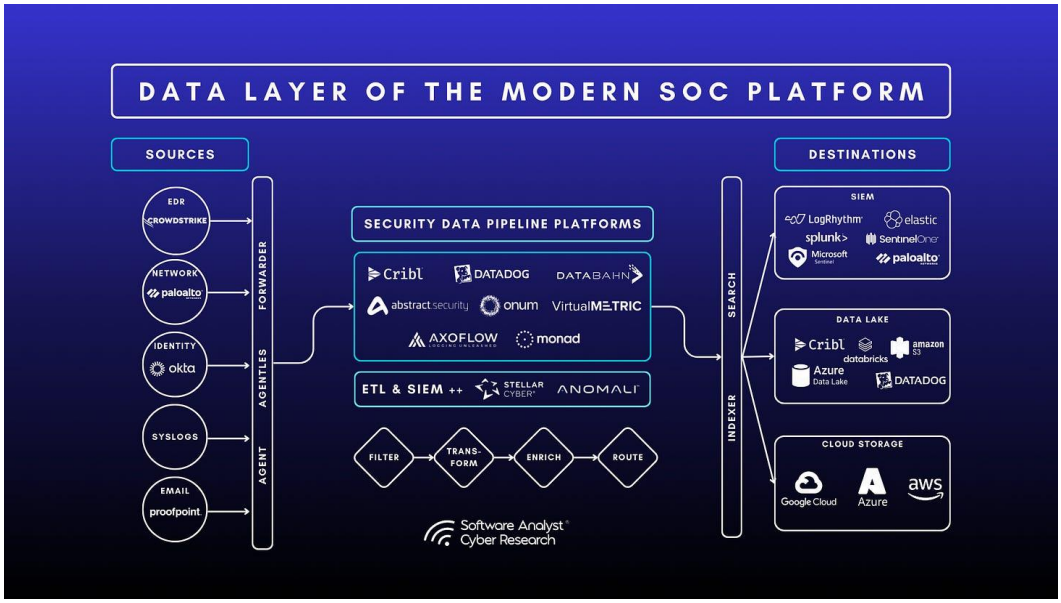

Figure 2: Log Correlation Engine and Event Processing Workflow.

Following ingestion, the system performs event normalization, which standardizes the structure and semantics of log records generated by different systems. This step ensures that event attributes such as user identifiers, timestamps, and activity types are represented consistently across all data sources. Once normalized, the events are processed by the feature extraction module, which transforms raw log entries into structured feature vectors describing behavioral characteristics such as login frequency, resource access patterns, and network communication behavior.

The extracted features are then used to construct a correlation graph, where nodes represent normalized events and edges represent relationships derived from shared attributes or temporal proximity. Finally, the anomaly scoring module evaluates correlated event patterns using statistical and graph-based models to assign risk scores to user activities. Suspicious sequences of events with high anomaly scores are subsequently flagged for further investigation by security analysts through monitoring dashboards and incident response systems.

3.2 Feature Extraction for Insider-Threat Indicators

Feature extraction plays a fundamental role in insider-threat detection systems because it transforms raw log records into structured variables that can be analyzed using statistical or machine-learning models. In enterprise security monitoring, system logs generated by authentication servers, file systems, and endpoint monitoring agents contain extensive behavioral information about user activities. However, these logs often exist as unstructured or semi-structured records that must be converted into quantitative features before meaningful analytical modeling can be performed (Chuvakin et al., 2013; Legg et al., 2015). Feature extraction therefore involves identifying measurable indicators of abnormal behavior that may signal insider misuse or unauthorized activity within organizational systems.

In the proposed framework, feature extraction is performed on normalized event records generated during the log aggregation phase described in Section 3.1. Let the normalized event set be defined as

$$E' = \{e'_1, e'_2, e'_3, \dots, e'_n\}$$

where each event e'_i represents a standardized representation of a system activity. From this dataset, behavioral features are derived to represent different aspects of user interaction with enterprise systems. These features are aggregated into a multidimensional feature vector that characterizes user activity within a given observation window. The resulting feature representation can be expressed as

$$X = [x_1, x_2, \dots, x_m]$$

where x_k denotes the k^{th} behavioral feature extracted from the event dataset and m represents the total number of features used in the detection model. Each feature captures a specific behavioral indicator associated with insider-threat activity.

One important behavioral indicator is login frequency deviation, which measures how frequently a user authenticates within a given time interval relative to their historical login patterns. Sudden increases in login frequency may indicate compromised credentials or malicious attempts to access internal resources repeatedly. Let the login count of user u during time window t be represented as $L_u(t)$. The deviation from the expected login behavior can be modeled as

$$x_{login} = \frac{| L_u(t) - \mu_u |}{\sigma_u}$$

where μ_u and σ_u represent the historical mean and standard deviation of login frequencies for user u. Larger deviations indicate abnormal authentication activity that may require further investigation (Greitzer & Hohimer, 2011).

Another critical indicator is file access anomaly detection, which focuses on identifying unusual access patterns to sensitive organizational data. Insider attacks often involve the unauthorized copying, modification, or exfiltration of confidential files. To quantify this behavior, file access activity can be represented as a rate of access events over time. Let $F_u(t)$ represent the number of file access events generated by user u within time interval t. The anomaly score for file access behavior can be expressed using an entropy-based metric:

$$x_{file} = -\sum_{i=1}^{k} p_i \log(p_i)$$

where p_i represents the probability distribution of file access operations across different resource categories. High entropy values may indicate abnormal distribution patterns, suggesting unauthorized access to multiple sensitive resources (Eberle & Holder, 2009).

A third important feature relates to privilege escalation attempts, which occur when a user attempts to gain higher access privileges than those assigned under normal operational roles. Such behavior is often associated with malicious insider activities or compromised accounts attempting to obtain administrative control over enterprise systems. Privilege escalation events can be modeled as a binary indicator or weighted event frequency. Let $P_u(t)$ denote the number of privilege escalation attempts performed by user u during

observation period t. The corresponding feature can be expressed as

$$x_{priv} = \log(1 + P_u(t))$$

The logarithmic transformation reduces the impact of extreme values while preserving sensitivity to abnormal escalation attempts. This feature allows the detection system to identify users who repeatedly attempt to access privileged resources beyond their authorized roles (Tuor et al., 2017).

In addition to individual feature metrics, the framework constructs a composite behavioral risk score by aggregating extracted features into a weighted anomaly indicator. This aggregated score represents the likelihood that a user activity pattern corresponds to insider threat behavior. The aggregated anomaly score can be defined as

$$S_u = \sum_{k=1}^{m} w_k x_k$$

where w_k represents the weight assigned to feature x_k, reflecting its relative importance in identifying suspicious activity patterns. These weights may be determined through statistical analysis, domain expertise, or machine-learning training procedures (Nurse et al., 2014). Users whose anomaly scores exceed a predefined threshold are flagged for further security investigation.

To summarize the key behavioral indicators used in the detection framework, Table 3 presents the primary feature set derived from system logs along with their corresponding log sources, mathematical representations, and security interpretations.

Table 3: Feature Set Used for Insider-Threat Detection.

Feature Name	Log Source	Mathematical Representation	Security Interpretation
Login Frequency Deviation	Authentication Logs	(x_{login} = \frac{	L_u(t) - \mu_u
File Access Entropy	File Access Logs	$x_{file} = -\sum_{i=1}^{k} p_i \log(p_i)$	Identifies unusual distribution of file access across sensitive resources
Privilege Escalation Indicator	Privilege Management Logs	$x_{priv} = \log(1 + P_u(t))$	Detects attempts to gain unauthorized administrative privileges
Network Transfer Deviation	Network Flow Logs	$x_{net} = \frac{D_u(t)}{\bar{D}_u}$	Indicates abnormal data transfer volumes potentially linked to exfiltration
Process Execution Anomaly	Endpoint Activity Logs	(x_{proc} = \frac{	C_u(t) - \bar{C}_u

3.3 Correlation Scoring Model

The correlation scoring model is designed to quantify the level of suspicious behavior associated with individual users by aggregating anomaly indicators extracted from multiple system logs. Insider threats typically manifest as coordinated activities occurring across different enterprise systems, including authentication services, endpoint devices, file repositories, and network infrastructure. Therefore, effective detection requires a unified scoring mechanism that evaluates correlations among behavioral features derived from these heterogeneous log sources (Greitzer & Hohimer, 2011; Legg et al., 2015).

In the proposed framework, behavioral indicators extracted during the feature extraction phase are integrated into a weighted scoring function that measures deviations from normal operational patterns. Let the feature vector describing a user's activity within an observation window be represented as

$$X_u = [x_1, x_2, x_3, \ldots, x_n]$$

where each feature x_i represents a behavioral anomaly metric derived from system logs, such as login frequency deviation, file access entropy, privilege escalation frequency, or abnormal network transfer activity. These features capture different dimensions of user behavior that may indicate potential insider threat activity.

The overall correlation score associated with a user u is defined as

$$S_u = \sum_{i=1}^{n} w_i x_i$$

where x_i represents the anomaly feature derived from log analysis and w_i denotes the weighting factor assigned to that feature. The weighting coefficients allow the model to assign different levels of importance to various behavioral indicators depending on their relevance to insider-threat detection. For example, privilege escalation attempts may receive higher weights than routine login deviations due to their stronger association with malicious activity.

To improve robustness, the weighting factors are normalized so that

$$\sum_{i=1}^{n} w_i = 1$$

This normalization ensures that the correlation score remains bounded within a predictable range and prevents any single feature from dominating the anomaly detection process.

Because insider threat activities often occur as sequences of related events across different systems, the model incorporates temporal correlation into the scoring process. Let t_i represent the timestamp of event i, and let Δt_{ij} denote the temporal distance between two correlated events. A temporal decay factor can be introduced to reduce the influence of events that occur far apart in time:

$$\omega_{ij} = e^{-\lambda |t_i - t_j|}$$

where λ represents a decay parameter controlling the sensitivity of the model to temporal proximity. Events occurring within a short time interval will therefore contribute more strongly to the correlation score than events that occur far apart in time.

Incorporating this temporal weighting mechanism, the correlation scoring function can be extended as

$$S_u = \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n} \sum_{j=1}^{n} \omega_{ij} \cdot \phi(x_i, x_j)$$

where $\phi(x_i, x_j)$ represents a correlation function capturing the interaction between two anomaly features. This extended formulation enables the detection framework to identify coordinated behaviors involving multiple anomalous events rather than relying solely on isolated indicators.

To determine whether a user's behavior indicates potential insider threat activity, the computed correlation score is compared against a predefined detection threshold. Let θ denote the anomaly threshold determined through empirical analysis or statistical calibration. A user is flagged as suspicious when

$$S_u \geq \theta$$

This threshold-based classification allows the system to generate alerts for security analysts when correlated behavioral anomalies exceed acceptable operational limits.

The proposed correlation scoring model therefore integrates anomaly features derived from multiple log sources, temporal relationships among events, and weighted behavioral indicators into a unified detection framework. By combining these analytical components, the model provides a scalable mechanism for identifying insider-threat indicators within SME network environments while maintaining computational efficiency suitable for resource-constrained infrastructures.

3.4 Insider-Threat Probability Estimation

After computing anomaly features and correlation scores, the final stage of the proposed framework estimates the probability that a user's observed behavior corresponds to insider-threat activity. Insider threat detection can be formulated as a probabilistic classification problem in which the objective is to determine the likelihood that a sequence of system events is associated with malicious or unauthorized behavior. Probabilistic estimation models are particularly suitable for this task because they allow the system to quantify uncertainty in behavioral patterns derived from heterogeneous log sources (Greitzer & Hohimer, 2011; Nurse et al., 2014).

Let $X = [x_1, x_2, \ldots, x_m]$ denote the feature vector representing behavioral indicators extracted from authentication logs, file access records, endpoint monitoring data, and network connection logs. These features capture deviations from normal user activity patterns such as abnormal login frequencies, privilege escalation attempts, and unusual data access behaviors. To estimate the probability that these features correspond to insider threat activity, the proposed framework employs a logistic regression–based probabilistic model, which maps behavioral features to a probability value within the interval $[0, 1]$.

The conditional probability that a user behavior pattern corresponds to an insider threat event T given feature vector X is defined as

$$P(T \mid X) = \frac{1}{1 + e^{-\beta^T X}}$$

where β represents a vector of model parameters and X denotes the behavioral feature vector derived from log analysis. The expression $\beta^T X$ represents the linear combination of behavioral features and their corresponding weights. This logistic transformation converts the linear decision boundary into a probabilistic output, allowing the model to estimate the likelihood of malicious activity rather than producing a simple binary classification (Legg et al., 2015).

The parameter vector $\beta = [\beta_0, \beta_1, \beta_2, \ldots, \beta_m]$ determines the influence of each behavioral feature on the final probability estimate. The model can therefore be expanded as

$$P(T \mid X) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_m x_m)}}$$

where β_0 represents the intercept term capturing baseline insider threat probability and β_i represents the coefficient associated with feature x_i. These parameters can be estimated through maximum likelihood estimation using historical behavioral datasets that contain labeled insider threat and benign activity instances.

To determine whether a given activity sequence represents a potential insider threat, the estimated probability is compared against a predefined decision threshold τ. A security alert is generated when

$$P(T \mid X) \geq \tau$$

where τ represents the detection threshold determined through empirical calibration. Adjusting this threshold allows security administrators to balance detection sensitivity and false-positive rates depending on the operational requirements of the organization. Lower thresholds increase detection sensitivity but may produce more false alarms, whereas higher thresholds reduce false alerts but may delay detection of subtle insider activities (Tuor et al., 2017).

To improve robustness in environments with highly variable behavioral patterns, the probability estimation model can be extended by incorporating regularization mechanisms that prevent overfitting and improve generalization. One common approach involves adding a regularization term to the logistic loss function:

$$L(\beta) = -\sum_{i=1}^{N} [y_i \log P(T \mid X_i) + (1 - y_i) \log(1 - P(T \mid X_i))] + \lambda \| \beta \|^2$$

where y_i represents the observed class label for sample i, N represents the number of observations, and λ is a regularization parameter controlling the complexity of the model. This formulation ensures that the probability estimation model remains stable when applied to large-scale log datasets generated within enterprise environments.

The integration of probabilistic estimation within the proposed framework enables the system to transform correlated behavioral anomalies into quantitative risk indicators. Instead of generating binary alerts, the model assigns continuous probability scores that allow security analysts to prioritize investigations based on estimated threat severity. This probabilistic interpretation improves decision-making in security operations by providing a principled mechanism for evaluating insider threat likelihood based on multi-source log analytics and behavioral feature correlations.

4. RESULTS AND DISCUSSION

4.1 Experimental Setup

To evaluate the effectiveness of the proposed log-correlation framework for insider-threat detection, a controlled experimental environment was designed to simulate a realistic small-to-medium enterprise (SME) network infrastructure. The objective of this experimental setup is to assess the ability of the proposed detection model to identify suspicious user activities using correlated event logs generated from multiple systems within an enterprise network. The experimental design focuses on replicating the operational characteristics of a typical SME environment, including limited infrastructure resources, distributed endpoint devices, and centralized security monitoring systems.

The experimental network consists of 200 endpoint devices connected through an internal enterprise network managed by a centralized authentication and logging infrastructure. These endpoints represent employee workstations, application servers, and administrative systems that collectively generate large volumes of operational logs. Each endpoint is equipped with monitoring agents capable of generating telemetry data describing system events such as login attempts, file access operations, process execution activities, and network communication records. These event streams provide the raw data required for implementing log aggregation and correlation-based anomaly detection mechanisms.

All log data generated by the endpoint devices are transmitted to a centralized log server, which functions as the primary data aggregation and processing node within the experimental architecture. The centralized logging infrastructure performs event ingestion, timestamp synchronization, and normalization of heterogeneous log records generated by different systems. This centralized architecture is consistent with practical SME monitoring environments where security operations rely on centralized logging services rather than distributed analytics platforms due to infrastructure limitations. The aggregated log dataset therefore provides a unified repository from which behavioral features can be extracted and correlated across multiple systems.

To assess the detection capabilities of the proposed framework, several simulated insider-threat scenarios were introduced into the network environment. These scenarios were designed to replicate realistic insider behaviors that commonly occur in organizational security incidents. Examples of simulated scenarios include abnormal login activities outside standard working hours, unauthorized attempts to escalate user privileges, excessive file access operations targeting sensitive directories, and abnormal outbound data transfer patterns suggesting potential data exfiltration attempts. Each simulated threat scenario was injected into the experimental dataset at controlled intervals to evaluate

the system's ability to detect suspicious activity sequences through log correlation and anomaly scoring.

The experimental framework also incorporates time-based monitoring windows to evaluate the system's responsiveness to evolving behavioral patterns. Let the total observation period be represented as

$$T = \{t_1, t_2, \ldots, t_k\}$$

where each time interval t_k represents a discrete monitoring window used to analyse user behavior across the network. Within each window, the system collects event features and computes anomaly scores using the correlation scoring model defined in the previous section. The anomaly score for each user activity session is computed as

$$S_u(t_k) = \sum_{i=1}^{n} w_i \, x_i(t_k)$$

where $x_i(t_k)$ represents the anomaly feature extracted during time interval t_k and w_i represents the feature weighting factor. This formulation enables the detection system to evaluate behavioral anomalies dynamically as user activities evolve over time.

To provide a structured overview of the experimental environment used in this study, Table 4 summarizes the key parameters of the dataset and network configuration used during the experimental evaluation. The table highlights the main components of the simulated SME infrastructure, including the number of endpoint devices, log data sources, event generation characteristics, and the role of each parameter within the experimental framework.

Table 4: Experimental Dataset and Network Configuration.

Parameter	Value	Description	Experimental Purpose
Network Size	200 Endpoints	Number of simulated user workstations and servers within the SME network	Represents realistic SME infrastructure scale
Log Sources	Authentication, File Access, Endpoint Activity, Network Logs	Types of system logs collected for analysis	Enables multi-source log correlation analysis
Centralized Log Server	1 Aggregation Node	Server responsible for collecting and normalizing logs from all endpoints	Provides centralized log processing environment
Observation Window	10–30 minutes	Time interval used for aggregating behavioral features	Supports temporal correlation of events
Simulated Insider Events	Privilege Escalation, Abnormal Login, Data Exfiltration	Artificial threat scenarios injected into the dataset	Evaluates detection accuracy of the proposed framework

The experimental configuration described above enables systematic evaluation of the proposed log-correlation detection model within a controlled SME-scale environment. By combining realistic network activity with simulated insider-threat scenarios, the framework provides a suitable dataset for assessing the performance of correlation-based anomaly detection methods. The next section analyses the experimental results obtained from this setup and evaluates the detection performance of the proposed model using standard cybersecurity evaluation metrics.

4.2 Detection Performance Evaluation

To assess the effectiveness of the proposed insider-threat detection framework, the system was evaluated using standard classification metrics widely adopted in cybersecurity analytics and anomaly detection research. These evaluation metrics measure the accuracy of the detection model in identifying malicious user activities while minimizing false alarms. Because insider threat detection involves distinguishing between legitimate user behavior and malicious activity within enterprise networks, performance evaluation must consider both detection accuracy and the rate of incorrect classifications (Greitzer & Hohimer, 2011; Legg et al., 2015).

The detection performance of the proposed model was evaluated using three widely accepted metrics: precision, recall, and F1-score. These metrics are derived from the confusion matrix generated during the classification process, which categorizes system predictions into four groups: true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN). True positives represent correctly identified insider-threat events, while false positives represent normal user activities incorrectly classified as malicious. False negatives represent insider-threat events that the system failed to detect, while true negatives correspond to correctly identified benign activities.

The precision metric measures the proportion of detected insider-threat alerts that correspond to actual malicious activities. It evaluates the reliability of the detection

system by determining how many flagged alerts are truly malicious. Precision is defined as

$$Precision = \frac{TP}{TP + FP}$$

A high precision value indicates that the system generates relatively few false alarms, which is important for reducing alert fatigue among security analysts. Excessive false positives can overwhelm security teams and reduce the effectiveness of incident response processes, particularly in SME environments where security staff may be limited (Nurse et al., 2014).

The recall metric, also referred to as the detection rate or sensitivity, measures the ability of the system to identify actual insider-threat events within the dataset. Recall is defined as

$$Recall = \frac{TP}{TP + FN}$$

High recall indicates that the detection system successfully identifies most malicious activities, minimizing the number of insider attacks that remain undetected. In insider-threat detection systems, recall is particularly important because missed detections may allow malicious activities to continue without intervention (Tuor et al., 2017).

To balance the trade-off between precision and recall, the F1-score is used as a combined performance metric. The F1-score represents the harmonic mean of precision and recall, providing a single value that reflects both detection accuracy and reliability. It is calculated as

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

A high F1-score indicates that the detection model achieves both high precision and high recall simultaneously, making it an effective metric for evaluating anomaly detection algorithms in cybersecurity systems (Chandola et al., 2009).

To further evaluate the proposed log-correlation framework, detection performance was compared with several baseline detection approaches commonly used in insider-threat monitoring systems. These approaches include rule-based detection models, statistical anomaly detection techniques, and machine learning classifiers trained on behavioral features extracted from system logs. Each method was applied to the same experimental dataset described in Section 4.1, allowing a direct comparison of detection performance under identical conditions.

The evaluation also considers the impact of detection time windows, which represent the interval during which log events are aggregated and analyzed. Shorter time windows allow faster detection of suspicious activity but may reduce the availability of contextual information required for accurate analysis. Conversely, longer observation windows provide richer behavioral context but may delay detection of insider-threat activities. Therefore, detection accuracy was evaluated across multiple time windows to analyse how quickly each detection approach can identify suspicious behavior patterns.

Figure 3 illustrates the comparative detection performance of four different approaches: rule-based detection, statistical anomaly detection, machine learning classifiers, and the proposed log-correlation model. The horizontal axis represents the detection time window, while the vertical axis represents detection accuracy achieved by each method.

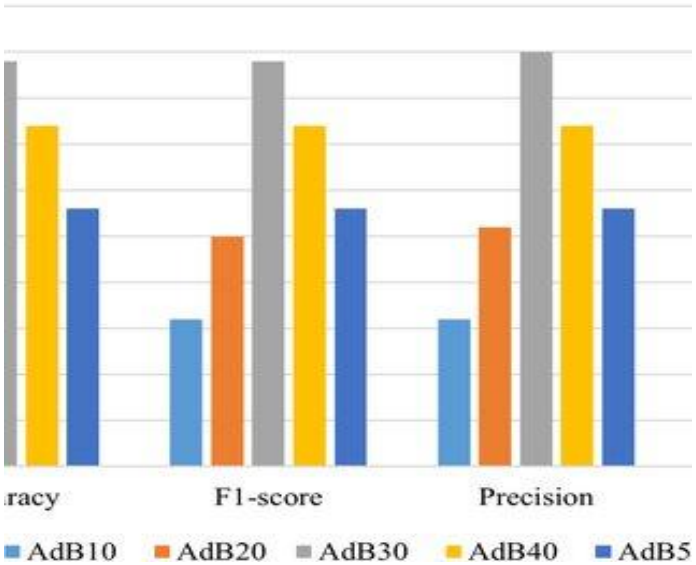

Figure 3: Detection Accuracy Comparison Across Log-Correlation Models.

The graph demonstrates that rule-based detection models exhibit relatively stable but lower accuracy because they rely on predefined patterns that cannot capture complex behavioral relationships across multiple systems. Statistical anomaly detection methods show moderate performance improvements but may still struggle with highly dynamic user behavior patterns. Machine learning classifiers achieve higher accuracy due to their ability to learn behavioral patterns from historical data; however, they require large training datasets and significant computational resources.

The proposed log-correlation framework achieves the highest detection accuracy across all time windows because it integrates behavioral features from multiple log sources and identifies coordinated activity patterns that may indicate insider threats. By combining correlation analysis with anomaly scoring mechanisms, the model improves detection performance while maintaining scalability suitable for SME environments.

4.3 Analysis of Insider-Threat Indicators

The experimental evaluation of the proposed log-correlation framework revealed several important behavioral patterns associated with insider-threat activity. By integrating logs from authentication systems, endpoint monitoring agents, file access records, and network traffic data, the framework was able to reconstruct sequences of user actions that indicate coordinated malicious behavior. Unlike traditional monitoring systems that analyse logs in isolation, the correlation-based model identifies relationships among events occurring across multiple systems within a defined temporal window. This multi-source analysis enables the detection of complex behavioral patterns that often characterize insider attacks (Greitzer & Hohimer, 2011; Nurse et al., 2014).

One significant observation from the experimental results is the presence of correlation patterns in malicious user activity. Insider threats rarely manifest as single anomalous events: rather, they typically occur as chains of related actions performed by the same user across different systems. For example, several simulated attack scenarios involved sequences where abnormal login events were followed by privilege escalation attempts and subsequently by large-scale file access operations. When analyzed independently, each of these events may appear legitimate. However, the correlation engine identifies strong relationships between these activities based on user identity, temporal proximity, and shared resource interactions.

Mathematically, the correlation strength between events e_i and e_j can be represented as

$$w_{ij} = e^{-\lambda|t_i - t_j|} \cdot \delta(u_i, u_j)$$

where $|t_i - t_j|$ represents the temporal distance between events, λ is a temporal decay parameter, and $\delta(u_i, u_j)$ is an indicator function that equals 1 when both events are associated with the same user identity. High correlation weights were consistently observed in scenarios involving insider misuse, indicating that malicious activities tend to occur in tightly clustered sequences rather than as isolated incidents.

Another important finding concerns detection latency, which refers to the time required for the monitoring system to identify suspicious activity after it begins. Traditional log-monitoring approaches often rely on periodic analysis of isolated system logs, which may delay detection until multiple independent alerts are generated. In contrast, the proposed framework uses real-time log aggregation and correlation analysis to detect suspicious behavioral sequences earlier in the attack lifecycle. The use of multi-log analytics enables the system to detect abnormal patterns as soon as related events appear across multiple systems.

Detection latency can be expressed as

$$L_d = t_{detect} - t_{initial}$$

where $t_{initial}$ represents the timestamp of the first malicious event and t_{detect} represents the time when the detection system identifies the suspicious activity sequence. Experimental results indicate that multi-log correlation significantly reduces detection latency compared with single-source monitoring systems. This improvement occurs because correlated events provide stronger evidence of suspicious behavior, enabling earlier classification of potential insider threats.

The experimental results also demonstrate improvements over traditional SIEM alerting mechanisms. Conventional SIEM systems rely heavily on predefined correlation rules that trigger alerts when specific event patterns are detected. While such rule-based systems are effective for identifying known attack signatures, they often struggle to detect novel or evolving insider threat behaviors that do not match predefined rules (Chuvakin et al., 2013). In addition, SIEM systems frequently generate large numbers of alerts due to isolated anomaly detection, leading to high false-positive rates and alert fatigue among security analysts.

The proposed log-correlation framework improves upon these limitations by combining feature-based anomaly scoring with graph-based event correlation. Rather than relying solely on predefined rules, the system evaluates behavioral relationships among events using weighted correlation metrics and probabilistic anomaly scoring. The anomaly score associated with a user activity sequence can be expressed as

$$S_u = \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} \phi(x_i, x_j)$$

where x_i represents behavioral features extracted from logs, w_i represents feature weights, and w_{ij} represents correlation weights between event pairs. This formulation allows the system to identify suspicious behavioral sequences even when individual events appear normal in isolation.

Overall, the analysis of insider-threat indicators demonstrates that multi-source log correlation provides a more comprehensive representation of user behavior within enterprise networks. By integrating heterogeneous log sources and analysing relationships among events, the proposed framework enhances the ability of SME monitoring systems to detect insider threats at earlier stages while reducing false alarms and improving operational efficiency.

5. CONCLUSION AND RECOMMENDATIONS

5.1 CONCLUSION

This study investigated the application of log-correlation analytics for the early detection of insider-threat indicators within small-to-medium enterprise (SME) networks. The results demonstrate that analysing system logs in isolation provides limited visibility into user behavior because malicious activities often manifest as sequences of coordinated events occurring across multiple systems. By integrating heterogeneous log sources—including authentication records, endpoint activity logs, file access histories, and network communication logs—the proposed framework reconstructs behavioral patterns that reveal suspicious activity sequences that would otherwise remain undetected in conventional monitoring systems.

The methodological framework developed in this study combines event normalization, behavioral feature extraction, correlation graph construction, anomaly scoring, and probabilistic threat estimation to identify insider-threat behaviors within distributed enterprise infrastructures. The correlation scoring model aggregates behavioral anomaly indicators into a unified risk metric that captures deviations from normal operational patterns. Mathematically, the user behavior score is expressed as

$$S_u = \sum_{i=1}^{n} w_i x_i$$

where x_i represents extracted anomaly features and w_i denotes weighting factors representing their relative significance. This scoring mechanism enables the detection framework to quantify suspicious behavior based on correlated activity patterns rather than isolated events.

Experimental evaluation using a simulated SME network environment demonstrated that multi-source log correlation significantly improves detection performance compared with rule-based monitoring systems, statistical anomaly detection methods, and traditional SIEM-based alerting mechanisms. The results show improvements in detection accuracy and reduced detection latency when correlated behavioral indicators are analyzed across multiple systems. These findings confirm that correlation-driven security analytics can enhance the ability of SMEs to identify insider threats at earlier stages of attack progression.

Overall, the study contributes a scalable and computationally efficient framework for insider-threat detection that is suitable for resource-constrained SME environments. By leveraging existing system logs and applying correlation-based behavioral analytics, organizations can significantly improve their security monitoring capabilities without deploying complex or expensive enterprise security infrastructures.

5.2 Practical Recommendations

Based on the findings of this study, several practical recommendations can be proposed for organizations seeking to improve insider-threat detection capabilities within SME environments.

First, SMEs should deploy centralized log aggregation systems that collect event data from diverse organizational systems. Centralized log servers enable the consolidation of authentication logs, endpoint monitoring records, file access logs, and network traffic data into a unified analytical repository. Such aggregation provides the necessary data foundation for correlation-based behavioral analysis and supports real-time monitoring of enterprise activity patterns.

Second, organizations should adopt lightweight correlation algorithms capable of identifying relationships among events occurring across different systems. Correlation models based on event similarity, temporal proximity, and shared user attributes allow security systems to reconstruct behavioral sequences associated with insider-threat activity. For example, correlation strength between events may be represented as

$$w_{ij} = e^{-\lambda |t_i - t_j|}$$

where the temporal decay parameter λ determines the influence of event proximity. Lightweight correlation algorithms reduce computational overhead and enable scalable monitoring suitable for SME infrastructure constraints.

Third, organizations should integrate behavioral anomaly detection modules within their security monitoring architectures. Feature extraction techniques that capture indicators such as abnormal login frequencies, unusual file access patterns, and unauthorized privilege escalation attempts can provide valuable signals for identifying suspicious behavior. These anomaly features can be incorporated into probabilistic detection models that estimate insider-threat likelihood and prioritize alerts based on behavioral risk scores.

Together, these recommendations provide a practical roadmap for SMEs seeking to strengthen their cybersecurity posture through scalable log analytics and correlation-based monitoring strategies.

5.3 Future Research Directions

Although the proposed framework demonstrates promising results for insider-threat detection in SME networks, several research opportunities remain for improving detection accuracy and scalability. One promising direction involves the application of graph neural networks (GNNs) for insider-threat detection. Because correlated system events can naturally be represented as graphs, GNN-based models may provide powerful mechanisms for learning complex relationships among events and identifying anomalous activity patterns within large event graphs.

Another important research direction concerns the development of real-time streaming log analytics architectures. Modern enterprise systems generate large volumes of log data continuously, making batch processing approaches insufficient for timely threat detection. Future research should explore streaming data processing frameworks capable of performing event correlation and anomaly detection in real time using distributed computing platforms.

Finally, integrating log-correlation frameworks with zero-trust security architectures represents a promising avenue for improving enterprise cybersecurity resilience. Zero-trust models assume that no user or device should be trusted by default, requiring continuous verification of identity and behavior across all systems. Incorporating correlation-based behavioral analytics into zero-trust frameworks may enable organizations to dynamically assess user risk levels and enforce adaptive access control policies based on real-time behavioral indicators.

Advancing research in these areas will further strengthen the ability of organizations to detect and mitigate insider threats, particularly within SME environments where scalable and efficient security monitoring solutions are essential for maintaining operational resilience.

REFERENCES

1. Ahmad, A., Desouza, K. C., Maynard, S. B., Naseer, H., & Baskerville, R. (2019). How integration of cybersecurity management and incident response enables organizational learning. *Journal of Strategic Information Systems, 28*(3): 102–116.
2. Al-Mhiqani, M. N., Ahmad, R., Yassin, W., Hassan, R., Abdulkareem, K. H., Ali, N. S., & Abdullahi, M. (2020). A review of insider threat detection: Classification, techniques, datasets, and challenges. *Applied Sciences, 10*(15): 5208.
3. Behl, A., & Behl, K. (2017). *Cybersecurity and cyberwar: What everyone needs to know*. Oxford University Press.
4. Cappelli, D., Moore, A., & Trzeciak, R. (2012). *The CERT guide to insider threats*. Addison-Wesley.
5. CERT Insider Threat Center. (2022). *Common sense guide to mitigating insider threats* (7th ed.). Carnegie Mellon University.
6. Chuvakin, A., Schmidt, K., & Phillips, C. (2013). *Logging and log management: The authoritative guide to understanding the concepts surrounding logging and log management*. Syngress.
7. Conti, M., Dehghantanha, A., Franke, K., & Watson, S. (2018). Internet of Things security and forensics: Challenges and opportunities. *Future Generation Computer Systems, 78:* 544–546.
8. Eberle, W., & Holder, L. (2009). Insider threat detection using graph-based approaches. *Cybersecurity Applications & Technology Conference for Homeland Security*, 237–241.
9. Greitzer, F. L., & Frincke, D. A. (2010). Combining traditional cyber security audit data with psychosocial data: Towards predictive modeling for insider threat mitigation. In *Insider Threats in Cyber Security* (pp. 85–113).
10. Greitzer, F. L., & Hohimer, R. (2011). Modeling human behavior to anticipate insider attacks. *Journal of Strategic Security, 4*(2): 25–48.
11. Inayat, U. (2023). Insider threat mitigation: A systematic literature review. International Journal of Information Security.
12. Jalloh, M. S., & Bamigwojo, O. V. (2023). Data-driven decision support systems for enhancing manufacturing productivity. International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 10(2): 440-449.
13. Legg, P., Buckley, O., Goldsmith, M., & Creese, S. (2015). Automated insider threat detection system using user and role-based profile assessment. *IEEE Systems Journal, 11*(2): 503–512.
14. Nurse, J. R. C., Buckley, O., Legg, P., Goldsmith, M., Creese, S., Wright, G., & Whitty, M. (2014). Understanding insider threat: A framework for characterising attacks. *IEEE Security & Privacy, 12*(6): 20–27.
15. Pelevin, D. (2021). *Research of methods and algorithms of insider detection in corporate networks*. Cybersecurity Research Thesis.
16. Peltier, T. (2016). *Information security policies, procedures, and standards: Guidelines for effective information security management*. Auerbach Publications.

17. Salem, M. B., Hershkop, S., & Stolfo, S. J. (2008). A survey of insider attack detection research. In *Insider Attack and Cyber Security* (pp. 69–90).
18. Sharma, A., & Dash, S. (2019). Big data analytics for insider threat detection. *Procedia Computer Science, 167:* 216–225.
19. Subhani, A. (2021). Review of insider and insider threat detection methods in organizational environments. *Journal of Advanced Research in Social Sciences and Humanities*.
20. Sultana, N., Chilamkurti, N., Peng, W., Alhadad, R., & Kabir, M. (2019). Survey on SDN-based network intrusion detection system using machine learning approaches. *Peer-to-Peer Networking and Applications, 12:* 493–501.
21. Tuor, A., Kaplan, S., Hutchinson, B., Nichols, N., & Robinson, S. (2017). Deep learning for unsupervised insider threat detection in structured cybersecurity data streams. *AAAI Workshop on Artificial Intelligence for Cyber Security*.
22. Yaseen, Q., Althebyan, Q., & Panda, B. (2016). Anomaly detection in network traffic based on statistical inference and machine learning techniques. *IEEE International Conference on Big Data Security*.
23. Zhang, Y., Meratnia, N., & Havinga, P. (2010). Outlier detection techniques for wireless sensor networks: A survey. *IEEE Communications Surveys & Tutorials, 12*(2): 159–170.