STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **15**, April 2026, pp 2996–3013.
Published online in International Academic Press (www.IAPress.org)

Anomaly Detection in Endpoint Security: Leveraging Baseline Deviation Techniques for Enhanced Protection

Kamran Asgarov*

Department of Engineering Mathematics and Artificial Intelligence, Azerbaijan Technical University, Azerbaijan

Abstract The aim of the study was to develop approaches to using deviations from the baseline to detect anomalies in endpoint security in order to improve the effectiveness of threat detection. The work included the analysis of anomaly detection in endpoint security and the development of baseline deviation approaches to improve security. The results of the study included the creation of basic applications for analyzing network traffic using Z-scores, as well as for identifying correlated events based on time marks and values, which made it possible to detect anomalous activities. Process diagrams for classifying anomalous events and responding to them using machine learning (ML) techniques were demonstrated. In addition, approaches such as dynamic baseline update, multivariate analysis of variance, temporal contextual models, integration with event correlation analysis, and risk-based variance ranking systems have been developed. Dynamic baseline updates allowed real-time adaptation to changes in system behavior, multivariate analysis revealed complex relationships between parameters, and temporal contextual models took into account cyclical patterns and trends in the data. On the other hand, integration with event correlation analysis revealed interdependencies between different types of activities, while risk-based variance ranking systems prioritized detected anomalies, allowing for faster response to the most critical threats. The results also included an analysis of the benefits, limitations, and application cases of each approach, covering areas such as virtual private networks, management and data acquisition systems (SCADA), and Internet of Things (IoT) devices. The results confirm that the proposed approaches reduce false positives, increase the accuracy of anomaly detection, and increase the resilience of cybersecurity systems.

Keywords Machine Learning, Dynamic Data Update, Multivariate Analysis, Temporal Contextual Models, Event Correlation Analysis, Risk-Based Ranking Systems

AMS 2010 subject classifications 62M10, 62H30, 62H12

DOI: 10.19139/soic-2310-5070-3128

1. Introduction

Today's cyberattacks are becoming increasingly sophisticated, creating significant challenges to protect endpoint devices such as computers, mobile devices, and Internet of Things devices. Anomaly detection is a critical area for threat prevention, as traditional signature-based methods are ineffective at detecting new or unknown attacks. Current challenges include processing large amounts of data and adapting to dynamic changes in system behavior. A promising solution is the use of machine learning (ML) methods, which allow you to automatically detect deviations from normal behavior and respond to threats in a timely manner.

The study by Karimli [1] had demonstrated the limitations of traditional security methods in cloud computing and emphasizes the need to develop new security mechanisms that can assess risks and adapt to the dynamic conditions of cloud services. This suggests that standard signature-based or static rule-based approaches do not provide an adequate level of protection in complex environments with high data dynamics. In this context, one can

*Correspondence to: Kamran Asgarov (Email: asgarovkamran2@gmail.com). Department of Engineering Mathematics and Artificial Intelligence, Azerbaijan Technical University. 25 Huseyn Cavid Ave., Baku, Azerbaijan (AZ1073).

ISSN 2310-5070 (online) ISSN 2311-004X (print)

discuss the application of lightweight machine learning models that are able to quickly analyze telemetry data of cloud systems with minimal computational load, as well as stream algorithms that allow updating behavior profiles in real time without saving a complete historical dataset. Similarly, the work of Makhmudova and Dashdamirova [2] highlighted security challenges in the information society, focusing on threats that affect personal, public and state interests. This study highlighted the need to integrate multivariate analysis of deviations and event correlations to identify complex anomalies that do not manifest as apparent failures or irregularities. From a practical point of view, this approach requires computational optimization: simplified statistical models, a reduced set of functions, or streaming data processing methods can be used for endpoints, allowing you to efficiently detect threats without overloading device resources.

Research by Bagirov [3] had focused on detecting new malware that cannot be identified by traditional signature methods and demonstrates the effectiveness of using machine learning with n-grams extracted from executable files. This study illustrated the need for combined models that combine statistical approaches and learning algorithms to adaptively recognize emerging threats. In the context of practical implementation, it is advisable to consider optimizing algorithms through lightweight models that reduce memory and computation, as well as the application of streaming data processing to quickly respond to potentially dangerous files. The work of Asgarov et al. [4] had demonstrated the effectiveness of unsupervised machine learning techniques for real-time anomaly detection on endpoint devices using telemetry data such as CPU (Central Processing Unit) load, memory consumption, and network traffic. This confirmed the feasibility of multivariate analysis and integration of correlation models to accurately determine anomalous behavior. In practical applications for resource-constrained environments, it is recommended to combine lightweight models with adaptive basic update engines, allowing for rapid adjustments to normal behavior profiles without significantly increasing runtime or memory.

On the other hand, research by Williams and Mbakwe-Obi [5] highlighted the importance of integrating anomaly detection and predictive modeling systems to increase database protection. The authors had demonstrated that the combination of such methods allows for early threat detection and proactive vulnerability identification, which strengthens security and ensures compliance with critical requirements. The research by Rehman and Bukhari [6] had focused on the application of machine learning algorithms and anomaly analysis to improve the accuracy of forecasting and optimize trading strategies, which simultaneously reduces costs and increases the liquidity of financial markets, demonstrating the practical application of analytical methods in complex and dynamic systems. At the same time, Chiranjeevi and Malathi [7] had proposed an innovative approach to detecting anomalies in video streams using dynamic graphs and convolutional neural networks, which had allowed analyzing spatio-temporal patterns and achieving high accuracy (98.72%), surpassing existing methods and improving the efficiency of video surveillance and early warning systems.

Research by Mo and Zhang [8] had demonstrated that anomaly detection algorithms integrated with edge computing capabilities enable efficient real-time data processing and improve the protection of university networks even under resource-constrained and high-load environments. The author Lu [9] had developed an integrated system based on graph algorithms and machine learning, which increases the accuracy and speed of anomaly localization, while providing a high level of automation and intelligent cybersecurity management. In addition, Paul [10] had emphasized the need to use anomaly detection techniques to protect APIs, noting that traditional approaches based on static rules and signatures are proving insufficient against new threats, and suggests the application of anomaly analysis to detect non-standard patterns in API traffic. The analysis of these works had allowed to identify the key areas of effective implementation of anomaly detection systems: integration of predictive modeling and adaptive machine learning algorithms, the use of graph and convolutional neural networks for complex data, the use of edge computing for real-time processing, and the optimization of computing resources through light and stream models. All of these approaches demonstrate practical utility and at the same time point out the limitations of traditional methods, emphasizing the need to develop optimized models for real-world endpoint and network environments.

However, this research focused on the design and research of anomaly detection methods in endpoint security, including machine learning algorithms, multivariate and temporal deviation analysis, integration with event correlation analysis, and risk ranking systems. The objectives of the work included analyzing the effectiveness

of threat detection in endpoint security, developing software solutions for anomaly detection, and developing and comparing approaches to baseline deviations.

2. Materials and methods

The study was conducted in two phases: detection of anomalies in endpoint security and baseline deviation techniques to improve safety. The first phase focused on researching and applying machine learning and statistical analysis techniques to identify anomalies in endpoint data. A simple program was developed to analyze network traffic based on deviations from normal behavior using Z-scores. Z-score formula in network traffic analysis (1):

$$Z = \frac{X - \mu}{\sigma}, \tag{1}$$

where X is the value of the current packet, μ is the average value of the packets, and σ is the standard deviation.

The code was written in C++ using the Online C++ Compiler – Online Editor platform. It analyzes a network traffic dataset, where each record is represented by a "NetworkData" structure with two parameters: packet size and packet count. For each packet, a Z-score was calculated, which showed how much the current packet size differed from the average size. A deviation greater than 2σ was considered an anomaly. A second program, also written in C++, analyzed the correlations of events, checking whether the two types of events (e.g., login attempts and network traffic) occurred approximately in time and had meaningful values. If the timestamp difference between the two events was less than 5 minutes, and their values exceeded a predetermined threshold (in this case, 10), the program classified them as correlated. It went through all possible pairs of events and typed a message every time a correlation was detected. In addition, process diagrams for classifying anomalous events using machine learning and a notification/response system were demonstrated [11, 12].

Emphasis was placed on the development of approaches aimed at improving the effectiveness of threat detection using basic deviations. Dynamic baseline updates using adaptive machine learning algorithms were considered to account for changes in system behavior [13], as well as multivariate analysis of variance, which included interdependencies between different parameters such as network traffic and CPU load [14]. In addition, temporal contextual models were used to explain cyclical patterns and trends in the data, which made it possible to more accurately predict normal behavior [15]. The study also examined the integration of event correlation analysis to identify associations between different events, such as entry attempts and network traffic spikes [16], as well as risk-based variance ranking systems [17]. To ensure the correctness and reproducibility of the results, systematic comparisons with the most up-to-date basic approaches to anomaly detection, such as isolation scaffolding and autoencoders, are included in the assessment. Such comparisons are carried out using identical data divisions into training, validation and test samples, as well as consistent evaluation protocols, which makes it impossible to interpret the advantages of individual models in a biased way. The unity of experimental conditions to compare the results at the level of both individual metrics and their aggregate dynamics, providing an objective assessment of the relative effectiveness of the proposed approaches. Together, the use of an extended set of performance indicators and standardized comparative experiments forms a transparent and evidence-based basis for analyzing the results. This approach allows not only to demonstrate the advantages of a specific anomaly detection model, but also to clearly outline its strengths and weaknesses in comparison with established basic solutions, which is a prerequisite for scientifically based and practically significant implementation in cybersecurity systems.

The research was implemented as a step-by-step process that included data preparation, parameter tuning, machine learning model development, and integration of statistical methods to identify anomalies in endpoint security. At the initial stage, network traffic data and endpoint events were pre-processed. Each record in the dataset was represented by a structure that included key parameters such as batch size, number of packets, timestamps, and event type. The data were normalized to eliminate large-scale differences between different parameters, and missing or abnormal values were corrected by averaging or filling in based on statistical estimates to avoid skewing the analysis results. Preprocessing options included defining thresholds for clipping extreme values and setting time intervals for correlation analysis. At the second step, the parameters of the basic methods for detecting anomalies were configured. For Z-scores, the mean and standard deviation of packet size and number of packets within the

selected time window were calculated, followed by a Z-score for each packet with a deviation threshold greater than two sigma, allowing potential anomalies to be flagged. For the correlation analysis of events, a time threshold of five minutes and a threshold of event values were determined, exceeding which made it possible to classify events as related. The algorithm went through all possible pairs of events in the dataset and recorded messages about the detected correlation, which ensured the detection of complex relationships between different types of activity.

In the third step, machine learning models were developed to classify anomalous events. Simple statistical models and more complex deep learning techniques have been applied. Statistical approaches included multivariate analysis of deviations using main components to identify key areas of data variation, as well as ranking deviations by risk level. For time models, autoregressive integrated moving averages and LSTM (Long Short-Term Memory) models were used, which made it possible to take into account short-term and long-term data memory and predict normal behavior patterns. The parameters of the models were adjusted experimentally: for LSTM, the layer size, the number of neurons, the number of learning epochs and the learning rate were determined, and for statistical models, the number of components and the method for estimating covariances were chosen. At the final step, all methods were integrated into a single analytical system. Z-score and correlation analysis data were used as input for machine learning models, allowing for improved accuracy in anomaly classification. After training, the models were applied to test data sets, the classification results were stored in structured tables indicating the type of anomaly, the level of risk and the causal explanation. Each approach was accompanied by documentation on the advantages, limitations and examples of application, which ensured the transparency and reproducibility of the study, as well as the possibility of further improvement of the methods in real operating conditions.

Each approach was accompanied by tables with its advantages, limitations and examples. For example, applications for dynamic basic update included Virtual Private Network (VPN), Supervisory Control and Data Acquisition (SCADA), and IoT. Examples of multivariate analysis of variance were principal component analysis and distributed denial of service (DDoS), while temporal contextual models contained autoregressive integrated moving average and long-term short-term memory (ARIM-LSTM).

3. Results

3.1. Detection of anomalies in endpoint security

Securing endpoints – such as workstations, servers, and mobile devices – requires active monitoring and anomaly detection to identify potential threats or attack attempts. Endpoint protection systems include several layers of security: antivirus solutions, intrusion prevention systems (IPS), vulnerability management tools, and activity monitoring solutions. It is crucial not only to detect known threats, but also to identify previously undetected anomalies that may indicate zero-day vulnerabilities or new attack methods.

The main methods for detecting anomalies in endpoint security include machine learning (ML) algorithms, statistical methods, and correlation rules. One of the most common approaches is clustering and classifying algorithms to detect deviations from the normal behavior of the system. These methods build models based on historical data on the behavior of systems and subsequently detect anomalies in real time. Known machine learning algorithms include k-mean clustering, decision trees, and random forests that analyze parameters such as network request frequency, incoming traffic volume, and application activity to detect unusual behavior. An example of a code snippet for analyzing network traffic based on deviations from normal behavior is given (APPENDIX A). Here the Z-score method is used to detect anomalies (Formula 1). Consequently, the comparative analysis highlighted a trade-off between interpretability and efficiency (statistical and rule-based methods) versus expressiveness and detection power (Isolation Forests and Autoencoders), reinforcing the need for hybrid architectures that combine lightweight baseline monitoring at the endpoint level with more complex unsupervised or deep learning models deployed at edge or centralized processing layers.

Code (APPENDIX A) analyzes network traffic, determining whether packet sizes differ from normal behavior using standard deviation and Z-scores. The output indicates that a packet size of 300 bytes has been marked as abnormal compared to an average of 175 bytes. Because the deviation is greater than 2 sigma (SD=50), the

system marks this packet as anomalous, indicating unusual network activity or a potential attack. Evaluating the computational performance of anomaly detection systems involves providing quantitative benchmarks for runtime, memory consumption, and CPU and GPU (Graphics Processing Unit) resource usage for incrementally increasing datasets. This approach allows tracing how the performance of algorithms changes as the number of endpoints, events, or observation time intervals increases, as well as identify potential scaling bottlenecks. Capturing these metrics in standardized conditions ensures transparency of evaluation and allows to compare different models not only in terms of detection accuracy, but also in terms of their suitability for practical use in environments with different computational constraints. The feasibility of applying such approaches to resource-constrained endpoints, such as IoT sensors or mobile devices, is evaluated taking into account available computing power, the amount of RAM (Random Access Memory), and energy constraints. In this context, hardware assumptions are clearly articulated, including the type of processor, the presence or absence of hardware acceleration, power consumption limits, and the ability to periodically exchange data with centralized services. Such refinement makes it possible to correctly interpret the results and determine which components of anomaly detection should be performed locally and which should be moved to peripheral or cloud processing levels.

For environments with low power consumption, simplified or approximate models are offered that reduce the computational load without significant loss of detection quality. Such approaches include quantization of models, which reduces the bit depth of parameters and reduces memory consumption, the use of a reduced set of the most informative functions, as well as the use of streaming statistics, which allows to update baselines incrementally without saving full historical data. Such optimizations ensure that anomaly detection systems are adapted to the conditions of limited resources and increase their suitability for long-term autonomous operation. In enterprise environments with a large number of endpoints and high event intensity, scalable deployment strategies are key. These include distributed and threaded architectures, which enable real-time data processing and allow for a uniform distribution of computing load across infrastructure nodes. Such architectures support horizontal scaling, integration with existing telemetry collection and centralized analysis systems, and provide low latency in anomaly detection even in the face of growing data volumes. Together, the combination of quantitative analysis of resource consumption, optimized models for limited devices, and scalable deployment architectures forms a holistic vision of the practical implementation of anomaly detection systems in multi-level operating environments. Classification techniques such as decision trees or random forests are also actively used to detect anomalous endpoint security events (Figure 1). These algorithms are trained on labeled data and can classify new data based on the model studied.

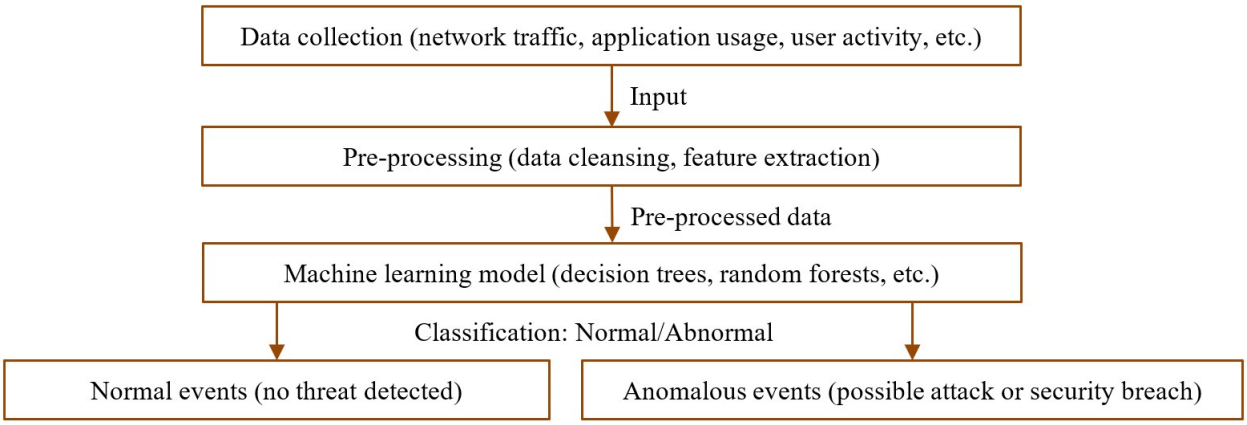

Figure 1. The process of classifying anomalous events using ML.

Source: compiled by the author on the basis of Dey and Rahman [11].

Another critical component of endpoint protection systems is rule-based correlation algorithms. This approach involves analyzing and correlating security events between networks, systems, and applications to detect deviations from normal states. Correlation rules are often used to aggregate event data, such as unusual login attempts, network traffic spikes, or unauthorized attempts to access resources. When analyzed taking into account several factors, such deviations may indicate potential threats. For example, the correlation of events between systems monitoring incoming and outgoing connections can reveal abnormal network patterns, such as a sudden spike in data transmission over a short period – a potential sign of a DDoS attack or data extraction attempts. These approaches are often implemented through security information and event management (SIEM) systems, which collect and analyze event logs from various devices and systems to identify suspicious or anomalous patterns (APPENDIX B).

Adaptive anomaly detection focuses on identifying deviations from normal behavior and adjusting system models to maintain accuracy in dynamic environments. The approach continuously monitors behavioral indicators such as process activity, session duration, network traffic, and access patterns, detecting short-term anomalies and long-term structural changes. Adaptive mechanisms, including controlled updating of baselines and change-point detection, allow the system to respond to confirmed shifts while minimizing the impact of transient fluctuations. These methods ensure that the anomaly detection system remains both sensitive to genuine threats and resilient to noise, providing robust protection in cyber-physical environments [18]. Practical workflows for deploying systems with explanation mechanisms involve their tight integration with the organization's existing security infrastructures. At the endpoint level, such systems interact with EDR (Endpoint Detection and Response)/EPP (Endpoint Protection Platform) class solutions, using pre-existing telemetry data on processes, file operations, network activity, and user behavior detection, but also the reasons that led to it. At the level of centralized event analysis, information about anomalies and related explanations is correlated in SIEM systems with logs from other sources, including network devices, servers, and cloud services. Compatibility with SIEM enables the construction of end-to-end analytical scenarios in which behavioral anomalies are associated with other indicators of compromise and business context. As part of such workflows, the analyst gets the opportunity to track the development of the incident from the initial deviation to the potential consequences, based on the explained results, and not only on numerical risk assessments. This facilitates more informed decision-making about escalation, response, or false positive closure of incidents, and facilitates auditing and improving anomaly detection models within the existing cybersecurity ecosystem. This example implements a simple correlation system that monitors events related to an unusual number of login attempts and anomalous network traffic. If both events occur simultaneously, this may indicate a potential threat, such as an intrusion attempt. number of login attempts or traffic volume) are significant and occurred at intervals of 5 minutes. The result demonstrates that the system showed a correlation between individual events if they had high values and coincided in time.

Evaluating the effectiveness of anomaly detection systems involves reporting a comprehensive and consistent set of performance indicators that allows to comprehensively characterize the quality of detection in different operating conditions. The key metrics in this context are the share of true positive positives (TPRs), which reflects the system's ability to detect real threats, and the share of false positives (FPRs), which characterizes the level of false alarms and directly affects the operational workload of security analysts. Accuracy and completeness are used to assess the balance between correctness and completeness of detection, while the F1 score provides an integral indicator that summarizes these two characteristics and is especially informative in the case of class imbalances characteristic of cybersecurity tasks. For a deeper analysis of the behavior of models at different thresholds, ROC (Receiver Operating Characteristic) curves and precision-completeness curves are used, which allow visually and quantifying the trade-off between sensitivity and specificity, or between completeness and accuracy. Such curves provide insight into the stability of models over a wide range of operating modes and facilitate the selection of optimal thresholds for practical use. Additionally, the withdrawal delay is taken into account, which reflects the time between the occurrence of anomalous activity and its detection by the system. This indicator is critical for real-time response scenarios, as even high accuracy values can lose practical value under conditions of significant detection time delay.

In addition to algorithms, an important aspect is the implementation of an alert and response system (Figure 2). Once an anomaly is detected, the system must not only log it, but also send alerts to administrators so that they

can take steps to mitigate further damage. This requires integrating monitoring tools with an incident management system, allowing for faster response to potential threats and minimizing endpoint security risks.

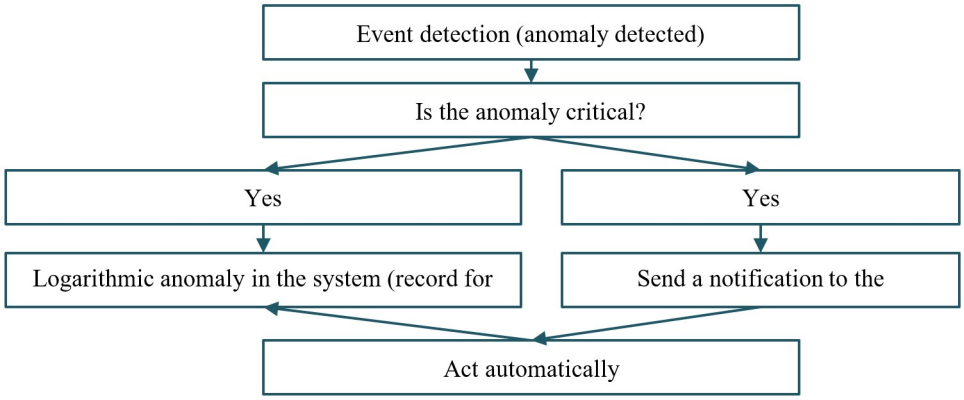

Figure 2. Notification and response system diagram.

Source: compiled by the author on the basis of the US Cyber Defense Agency [12].

It is also worth noting that with the development of technology and the emergence of new threats, methods for detecting anomalies must be constantly improved. Regularly updating threat databases and adapting models to new types of attacks are key elements of endpoint protection. For example, when anomalous activity is detected in an application or operating system, machine learning systems can adjust their models to effectively detect and prevent emerging attack vectors, such as zero-day exploits Endpoint security anomaly detection plays a key role in preventing threats by providing robust protection through machine learning, statistical analysis, and event correlation techniques.

The resilience of anomaly detection systems to realistic attack scenarios is assessed taking into account the ability to resist deliberate attempts by attackers to adapt their behavior to expected patterns of normal activity. Particular attention is paid to mimicry scenarios, when malicious actions are deliberately disguised as legitimate processes, evasion of detection by splitting activity or using permitted tools, as well as ”slow and low” attacks, implemented through minimal, time-stretched deviations. Under such conditions, the effectiveness of baseline deviation methods is determined by their ability to detect not only sharp anomalies, but also cumulative, weakly expressed changes in behavioral patterns that gradually form a threatening profile. To reduce vulnerability to manipulation of baseline indicators, reliable statistical methods are being implemented, focused on resistance to noise, emissions and targeted data distortion. In particular, robust estimates of averages and variances, sliding windows with the controlled influence of new observations, as well as ensemble approaches combining several independent models are used. Such methods limit the possibility of gradual ”poisoning” of the basic model, in which the attacker tries to slowly shift the normal profile of behavior in a direction favorable to himself. Additionally, training of opponents is used, within which models are trained taking into account simulated attacks that simulate mimicry or evasion. This leads to the development of sensitivity to characteristic, albeit subtle, signs of malicious activity and increases the overall robustness of the system.

A key element in increasing resilience remains the clear formulation of threat models, which describe the expected capabilities, resources, and limitations of the attacker. Such models determine how much knowledge the attacker has about the system, whether he has access to historical data or training mechanisms, what his time horizons are, and acceptable risks of detection protection without overestimating or underestimating the threats. As a result, the combination of assessment of resilience to realistic attack scenarios, the use of robust and adversarial training methods and formalized threat models ensures a more reliable functioning of anomaly detection systems in environments with a high level of adaptability of attackers. These techniques allow for the detection of deviations from normal behavior by analyzing both clear threats and subtle anomalies, which is critical for protecting against new types of attacks, including zero-day vulnerabilities. The alert and response system further increases operational

efficiency while minimizing the impact of incidents. To strengthen security, basic deviation techniques should be applied, allowing for even more accurate and faster detection of anomalies based on established norms of system behavior.

The speed and stability of adaptation of anomaly detection systems based on basic indicators are analyzed as key factors that determine the ability of such approaches to correctly respond to legitimate changes in the functioning of information systems without losing sensitivity to threats. In real-world environments, user behavior, software configuration, load intensity, and network patterns are constantly changing, necessitating continuous updating of the underlying model of normal behavior. Excessively rapid adaptation leads to the risk of ”blurring” anomalies when potentially harmful activity is mistakenly integrated into the baseline profile as normal. At the same time, too slow adaptation leads to an increase in the number of false positives in response to legitimate changes, which reduces trust in the system and complicates its practical use. Adaptation rate analysis involves evaluating how quickly the model responds to confirmed changes in workflows without losing the stability of behavioral profiles. An optimal balance is achieved under conditions when short-term fluctuations and random bursts of activity are filtered out, and persistent and repetitive changes are gradually integrated into the baseline indicators. Stability of adaptation is considered as the ability of a system to maintain consistency and predictability of decisions over time, without showing sharp fluctuations in the level of anxiety in the absence of real incidents [19]. This is especially important in environments with cyclical or seasonal activity, where normal patterns can vary significantly depending on the time of day, day of the week, or reporting period. Adaptive anomaly detection in cyber-physical systems employs mechanisms for controlled updating of baseline indicators to balance stability and responsiveness. Adaptive learning rates adjust the speed of model updates based on confidence in the legitimacy of observed changes: during stable behavior, updates slow to maintain a consistent baseline, while confirmed long-term changes trigger gradual adaptation, avoiding abrupt shifts. Change-point detection identifies significant deviations in statistical or behavioral patterns, allowing transient anomalies to be distinguished from structural transformations. These mechanisms enable temporary separation of transition periods from normal operation and facilitate controlled retraining of models, ensuring both stability and adaptability in dynamic system environments. [20].

3.2. Basic deflection methods to improve safety

To effectively detect anomalies, it is necessary to determine the normal behavior of the system that serves as a baseline. Establishing a baseline involves collecting data on common endpoint activities, such as network traffic, CPU and memory usage, and user activities. This data is collected over a long period to account for seasonal and temporal variations. Once collected, the data is processed using statistical analysis and machine learning techniques. During pre-processing, the data is cleared of noise and anomalies that can distort the model. Algorithms such as moving averages or exponential smoothing are applied to generate a basic model.

To ensure the accuracy and efficiency of anomaly detection systems, it is critical to develop approaches that take into account the dynamic nature of modern information systems. The first approach involves dynamic baseline updates (Table 1). This method uses adaptive machine learning and statistical techniques to update the baseline in real time. It takes into account changes in system behavior caused by factors such as load fluctuations, time of day (such as peak hours), or the deployment of new applications. Online learning algorithms process data streaming, updating the underlying model without the need for a complete retraining.

This method allows for rapid adaptation to changes, minimizes false positives, and ensures effective detection of anomalies. In addition, it is worth noting the approach of multivariate analysis of deviations (Table 2). This approach involves applying multidimensional statistical and machine learning techniques to identify anomalies by analyzing correlations between different system parameters. Unlike one-dimensional analysis, where parameters are evaluated individually, a multidimensional approach takes into account their interactions, providing a more comprehensive understanding of the behavior of the system. For example, network traffic, CPU usage, disk activity level, and other metrics are analyzed simultaneously. Techniques such as principal component analysis, k-mean clustering, and deep learning techniques can be applied to identify hidden patterns and identify deviations that cannot be detected from a one-dimensional perspective.

Table 1. Benefits, limitations, and examples of dynamic baseline updates.

Dignity	Limitations	Examples of use
Reduction of false positives taking into account the current state of the system	High computational resource requirements for adaptive analysis	Continuous monitoring of network traffic in corporate VPNs
Ability to take into account seasonal and temporal fluctuations in behavior	Need for high accuracy of the initial base to ensure correct updates	Protect servers in cloud environments where load can vary significantly depending on the time of day or client requests
Support for new threat models arising from changes in user and application behavior	Risk of accumulating errors during adaptive updates if the data is poorly processed in advance	Implementation in SCADA systems
Real-time work in IoT devices	Data quality dependence: noise and random spikes can negatively affect the model	Detection of anomalies in the operation of IoT devices in smart homes

Source: compiled by the author on the basis of Arvisais [13].

Table 2. Advantages, limitations, and examples of multivariate anomaly analysis.

Advantages	Limitations	Examples of use
Taking into account complex interdependencies between parameters	High computational complexity	Detection of attacks that simultaneously increase network traffic and CPU load, such as DDoS attacks
High accuracy in detecting hidden anomalies	A significant amount of training data is required	Monitoring user activity to detect combined anomalies, such as large data downloads at the same time as scripting
Ability to detect multi-vector attacks affecting multiple system parameters	Difficulty interpreting results for non-professional staff	Analyze virtual machine performance to detect malicious activity based on correlations between network, processor, and disk settings

Source: compiled by the author on the basis of [14].

Multivariate deviation analysis is particularly effective in complex systems with numerous parameters, where methods for detecting one-dimensional anomalies lose effectiveness due to the inability to account for data dependencies. On the other hand, contextual models should also be considered (Table 3). These models are based on analyzing the behavior of the system and the user, taking into account temporal factors such as daily, weekly or seasonal cycles. This approach uses time-series and prediction models to identify deviations, given that the behavior of the system can vary significantly depending on the time of day, day of the week, or month. For example, user activity is usually higher on weekdays than on weekends, while the load on the system at night may be minimal. Models such as autoregressive integrated moving average, long-term short-term memory, and Holt-Winters are used to predict normal behavior based on time patterns. They help to detect anomalies effectively, ignoring predictable fluctuations, thereby reducing false positives.

Table 3. Advantages, limitations, and examples of temporal context models.

Advantages	Limitations	Examples of use
Taking into account time cyclicality reduces false positives	High requirements for the quality of historical data and the volume of training	Detection of anomalies in the corporate network of users (for example, bulk data downloads at night)
Ability to predict changes in system behavior	Sensitivity to abrupt changes in behavioral patterns not related to time factors	Analyze server load based on peak hours to prevent performance-based attacks
Effective detection of abnormalities associated with atypical activity over a period of time	Complexity of Model Setup for Dynamically Evolving Systems	Monitoring of industrial systems, where metrics depend on equipment operation schedules

Source: compiled by the author on the basis of Sleeman et al. [15].

Temporal contextual models are especially useful for systems with pronounced periodicity of behavior, such as corporate networks, IoT devices, or industrial systems, where the underlying behavior changes depending on the time of day or operational schedules. In turn, the approach of integrating the analysis of deviations with the correlation of events combines the detection of anomalies with the analysis of the relationships between different events of the system (Table 4). Here, in addition to individual deviations from the baseline, patterns and coincidences between different events, such as a spike in network traffic and an increase in the number of login attempts, are also considered. These events can be interrelated, for example, during an attack attempt, such as a DDoS, when the number of requests and the load on the server increases at the same time. This approach increases the accuracy of threat detection through more comprehensive analysis and contextual awareness, helping to eliminate false positives and identify real-world incidents faster.

Table 4. Advantages, limitations, and examples of integration with event correlation analysis.

Advantages	Limitations	Examples of use
Increased accuracy of threat detection by taking into account the interdependence of events	Large computational overhead to handle large volumes of events	Attack detection when sudden spikes in network traffic coincide with an increase in the number of login attempts, potentially indicating DDoS
Reduction of false positives by filtering uncorrelated activity	A significant amount of data required for the correct configuration of the algorithm	Correlation of events in security monitoring systems to detect data breaches, such as unusual user activity along with increased network traffic
Quickly identify complex attacks that combine multiple types of anomalous events	Difficulty interpreting results for staff unfamiliar with machine learning and correlation techniques	Detection of anomalous combinations of events in SCADA systems, for example, jumps in equipment temperature with sudden spikes in network traffic, indicating potential threats

Source: compiled by the author on the basis of [16].

The integration of variance analysis with event correlation is a key tool for systems where threat detection depends on a holistic analysis of many factors, such as network traffic, user activity, hardware status, and other

parameters. This approach not only detects anomalous events, but also determines their relationships, thereby increasing security. Another approach worth considering is the risk-based anomaly ranking system (Table 5). This method assesses threats based on several factors, including the magnitude of the deviation, the frequency with which they occur, and the context in which they occur. It prioritize threats by focusing resources on the most critical incidents, minimizing losses, and accelerating response to high-risk events. Each deviation detected during monitoring is evaluated according to predetermined criteria, such as the degree of deviation from normal system behavior (e.g., increased traffic or CPU load), the frequency of such deviations, and their context (e.g., time of day, user type, or actions related to the deviation). These parameters are then used to assign a risk score to each deviation, helping security services and specialists determine how quickly and to what extent to respond to a particular event.

Table 5. Benefits, limitations, and examples of risk-based anomaly ranking systems.

Advantages	Limitations	Examples of use
Improve incident response by prioritizing high-risk threats	Accurate, high-quality data is needed to accurately assess risks	Risk assessment during sudden spikes in network traffic with frequent login attempts, potentially indicating high-probability attacks
Reduce resource overhead by focusing on most critical events	Challenges in tuning and optimizing the model for proper risk assessment, especially in dynamic systems	Implementation of cloud services in security monitoring, where rapid response to changes in user activity (potentially evidence of data breaches) is critical
Provides better automation of decision-making to respond to threats	It is possible to lose less critical, but important information about the threat if it is not taken into account when assessing risks	Implementation of SCADA in protection systems to assess the risks associated with abnormal equipment load in case of non-standard deviations

Source: compiled by the author on the basis of Chen et al. [17].

Analysis of Table 5 demonstrates that risk-based anomaly ranking systems provide several significant benefits, including improved incident response through prioritization of high-risk threats, reduced resource overhead by focusing on the most critical events, and enhanced automation of decision-making for timely threat mitigation. However, their effectiveness heavily depends on the accuracy and quality of data used for risk assessment, as well as proper tuning and optimization of models in dynamic systems, with the potential risk of overlooking less critical but still important information relevant to comprehensive threat analysis. Practical applications of such systems include risk assessment during sudden spikes in network traffic or frequent login attempts, monitoring user activity in cloud services for rapid response to potential data breaches, and evaluating abnormal equipment loads in SCADA systems to detect threats under non-standard deviations. Therefore, the table highlights that risk-based anomaly ranking systems efficiently optimize resources and prioritize actions, yet require high-quality data and careful configuration to ensure the completeness and accuracy of threat assessment.

In the context of insider trading, baseline deviation techniques show their value due to their ability to analyze the long-term behavioral patterns of users and service accounts. Such approaches reveal gradual or hidden changes in access profiles, atypical activity outside of working hours, abnormal data transfer volumes, or unusual combinations of operations with sensitive information. Because insiders often operate within formally permitted authority, signature and rule-based systems show limited effectiveness, while behavioral analysis allows for the identification of potential threats based on statistical and contextual deviations from established patterns of normal activity. Supply chain attacks are seen as one of the most sophisticated categories of modern cyber threats due to the use of trusted update channels, third-party libraries, or partner services. In this case, baseline deviation techniques show effectiveness mainly during the operation phase, when compromised components begin to exhibit behavior that does not correspond to typical models of system functioning [21]. Abnormal network connections, uncharacteristic

requests to external resources, changes in privilege levels, and atypical interactions between software modules indicate a hidden compromise.

At the same time, these approaches have significant limitations and limit conditions of application. Their effectiveness largely depends on the quality and stability of the basic model of normal behavior, which is formed on the basis of historical data. In dynamic environments with frequent changes in configurations, user roles, or workloads, the risk of false positives increases, making it difficult to interpret the results. In addition, a short training period or a limited amount of data leads to an incomplete picture of normal behavior, which reduces the system's ability to correctly distinguish anomalies from acceptable deviations. The baseline-deviation-based approach remains most effective in relatively stable environments with well-defined workflows, where behavioral patterns are highly repeatable. It is found to be least effective in conditions of high volatility, massive updates, or drastically changing use cases, as well as against slow, well-camouflaged attacks that intentionally mimic normal activity for long periods of time. In such cases, baseline deviation techniques need to be combined with contextual analysis, event correlation, and other protection mechanisms to achieve an appropriate level of cyber resilience.

Thus, anomaly detection approaches play a key role in improving the security of information systems, each of which has its own benefits and applications. Dynamic Basic Update is ideal for systems with changing loads, such as corporate VPNs or cloud environments, where it is important to take into account time changes and quickly adapt to new threats. Multivariate analysis of variance is effective for complex systems with numerous interdependent parameters, such as DDoS attacks, where understanding the interactions between different metrics is critical. Temporal contextual models are best applied to systems with pronounced cyclical behavior, such as corporate networks or industrial systems, where the temporal context reduces false positives. Integration with event correlation improves the accuracy of threat detection by combining multiple events – for example, correlating network traffic spikes with an increase in login attempts – which is especially important for preventing sophisticated attacks. A risk-based anomaly ranking system helps focus efforts on the most critical threats, which is vital in high-load or dynamic environments such as cloud services and SCADA systems. Each approach can be tailored to specific needs, allowing for a more accurate and faster response to threats.

4. Discussion

The study demonstrated the use of dynamic baseline update, multivariate analysis of variance, and temporal contextual models to improve the effectiveness of threat detection. On the other hand, the work of Zhang and Lázaro [22] provides an overview of existing anomaly analysis techniques, including statistical, machine learning, and hybrid approaches, with a focus on their automation and interpretation. In contrast to the generalized nature of their analysis, current research focuses on the practical application and comparison of specific methods in real-world settings. The results also demonstrated approaches to integrating machine learning techniques into threat detection systems, including anomalous event classification and automated response. Meanwhile, Jones' work [23] emphasizes the application of large language models for security analysis, encompassing threat detection, anomaly analysis, and automated response, as well as discussions of their future prospects. Unlike the aforementioned research, the current work is focused on the application of machine learning and statistical analysis techniques to detect anomalies in endpoints, which allows taking into account the temporal and behavioral aspects of the system's operation.

The results are focused on prioritizing real-time threats to endpoint security using basic rejection techniques. Instead, the work of Aggera [24] investigated methods for protecting containerized Development & Operations (DevOps) environments, including behavior monitoring and anomaly detection. Both studies emphasize the importance of AI in improving anomaly detection, but current research specifically focuses on end targets, given their unique characteristics and behaviors. This study also included analysis of network traffic using Z-score as well as DDoS in the context of multivariate analysis of variance and integration with event correlation analysis. On the contrary, the work of Pardosi [25] proposed a solution to protect servers from DDoS attacks using the Nginx reverse proxy, which effectively filters traffic and reduces the impact of attacks. Both studies emphasize

the importance of proactive defense methods; However, current research focuses on improving endpoint security through machine learning and statistical analysis, complementing server-centric solutions proposed in other work.

This study examines machine learning techniques to improve the accuracy of threat detection, including dynamic updates, multivariate analysis of variance, temporal contextual models, event correlation analysis, and deviation ranking systems. In contrast, the work of Rajesh et al. [26] uses deep learning techniques such as transfer learning and the "You only watch once" algorithm to detect anomalies in real-time video streams. Both approaches focus on timely detection of anomalies, but the current results focus on endpoint protection, while the mentioned studies complement this direction by offering solutions for video surveillance analysis and security improvement. In addition, the work of Gan et al. [27] proposes deep learning techniques for poorly controlled anomaly detection, including transformer-based networking to improve performance in the face of limited anomalous data. Instead, current research focuses on applying machine learning and statistical analysis techniques to protect endpoints. The research is complementary because both aim to improve methods for detecting anomalies, but differ in their emphasis on data types and system-specific aspects. However, the results of the current study allow you to more effectively adapt to dynamic changes in the system and minimize false positive results.

This study examines methods for detecting anomalies at endpoints. Meanwhile, the work of Ouyang et al. [28] offers a cloud-based, fully automated public Turing test to distinguish between Computers and Humans architecture (CAPTCHA) with multi-level enembling and semi-supervised learning to improve anomaly detection and reduce the load on the cloud system. While the mentioned research focuses on optimizing CAPTCHA for efficient anomaly detection using ensemble methods, current research focuses on deeper data processing using machine learning and statistical endpoint protection techniques, increasing the accuracy and adaptability of the system in real-world operating conditions. Using ML, the study demonstrated approaches to improve the efficiency of anomaly detection at endpoints. The work of Okusi [29] emphasizes the application of AI and machine learning to protect critical infrastructure, including optimizing threat detection and staff training. While the paper presented explores the large-scale applications of AI and ML to protect infrastructure, the current research focuses on specialized endpoint protection techniques that take into account temporal and behavioral aspects. Both approaches emphasize the importance of machine learning in improving security, but the study carried out is distinguished by detailed methods that ensure adaptability and accuracy.

Overall, this study demonstrated the use of ML to detect endpoint anomalies, including dynamic baseline updates and risk-based variance ranking systems. Meanwhile, the work of Adelusi [30] emphasizes a multi-layered approach to endpoint protection, incorporating zero-trust architecture and machine learning techniques. Both studies confirm the importance of adaptive ML technologies for endpoint security, aligning their necessity and complementing each other through different aspects of approaches. On the other hand, authors Rehman and Ahmad [31] applied machine learning techniques to protect cloud infrastructure, including anomaly detection, intrusion prevention systems, and predictive analytics. Similar to current work that focuses on endpoints, the study cited highlights the importance of machine learning in improving security. Thus, the results of both studies coincide, demonstrating the effectiveness of machine learning techniques in both endpoint protection and cloud infrastructure security, confirming the versatility of these approaches at different levels of cyber defense.

This study mainly focuses on baseline deviation to detect anomalies in endpoint security. Meanwhile, the work of Nzekwe and Ozurumba [32] explores database security approaches using neural networks and Bayesian inference to detect Structured Query Language (SQL) injections, unauthorized access, and data breaches. Both studies emphasize the importance of analytically oriented solutions and are consistent with the need for machine learning techniques to improve cyber resilience. However, current work emphasizes the integration of different analysis methods to prioritize threats in real-time, ensuring high accuracy and adaptability in the face of modern cyber threats. The current study investigates the classification of anomalous events using machine learning and alert and response system architecture. The work of Chaudhuri [33] presents the "AI4Falcon" architecture, based on generative AI, for threat prediction and automated response. Both studies agree on the need for adaptive machine learning techniques to improve the accuracy and speed of threat detection, complementing each other with different aspects of their approaches. At the same time, the research carried out focuses on analytical methods, while AI4Falcon demonstrates the potential of generative AI.

As mentioned, this study focuses on baseline deviation techniques to detect anomalies in endpoint security. In turn, the work of Jeffrey et al. [34] explores methods for detecting anomalies in cyber-physical systems with a focus on threats to critical infrastructure. Both studies emphasize the importance of anomaly detection to improve safety; However, current research focuses on endpoints and machine learning, while the research cited focuses on cyber-physical systems and resource constraints. Despite these differences, the results of both studies are consistent in emphasizing the value of analytics solutions in improving cybersecurity. It is worth mentioning the work of Kaul and Khurana [35], who investigated the use of AI to improve the security of application programming interfaces, including machine learning to detect anomalies and improve authentication. In contrast to this study, which focuses on endpoint protection and correlation analysis, the above work focused on application programming interfaces in distributed systems. Both studies confirm the importance of machine learning in improving security, but the results are aimed at analyzing endpoints rather than application programming interfaces, highlighting differences in applications and demonstrating the versatility of AI techniques in cybersecurity contexts.

The work of Shafi and Mirjat [36] is notable because it highlights the application of AI for anomaly detection, behavioral analysis, and automated incident response, demonstrating improved detection accuracy, reduced response times, and effective risk management. In contrast to this work, which explores a wide range of applications of AI in cybersecurity, the current study looks in detail at approaches to improve endpoint security. The research also focused on the development of software for analyzing network traffic and identifying correlational events based on timestamps and values. Instead, the results of Elsaid and Zulkernin [37] focus on the development of PredictDeep, a framework that uses graph analytics and deep graph neural networks to detect and predict anomalies in cloud analytics applications. PredictDeep emphasizes the importance of graph structure analysis for anomaly detection and offers solutions for cloud infrastructure. Both studies demonstrate the effectiveness of machine learning in security, but current research focuses on endpoint protection, while PredictDeep focuses on optimizing the security of cloud systems.

The results of this study are consistent with the conclusions of Kommisetty et al. [38], which emphasize the importance of automating the detection, prediction, and response of anomalies in cyber defense. However, unlike the conceptual approach of these authors, which aims to mathematically generalize predictions and analyze dependencies to develop universal solutions, the current study focuses on classifying anomalous events and responding to them using machine learning techniques. Both approaches agree on the need for automation and increased adaptability in cyber defense, but current research complements the conceptual approaches, adapting them to real-world operational conditions, taking into account the temporal and behavioral aspects of endpoint operation. In addition, the results of this study are also consistent with the findings of Cappello [39], who investigated the role of modern endpoint detection and response systems, endpoint protection platform, and traditional antivirus solutions in ensuring the security of endpoint devices. Both studies emphasize the importance of adaptive, multi-layered approaches to endpoint protection, highlighting the need for machine learning techniques and data integration to improve the accuracy of threat detection. However, unlike the cited papers, which focus on the functional aspects and comparison of EDR, EPP and antivirus technologies, the current study focuses on the application of baseline deviation techniques, such as multivariate deviation analysis and temporal contextual models, to improve the security of endpoint devices.

While this study cites IoT as an example of dynamic baseline update and temporal contextual models, the work of Mandala [40] emphasizes the comparison of clustering, density analysis, and autoencoders to detect anomalies in IoT sensor data, such as smart locks. Both studies confirm the effectiveness of IoT and adaptive data analysis techniques in dynamic environments and emphasize resilience to external attacks. The results of this work complement the findings of the current study, demonstrating similarities in IoT-enabled security approaches, but the current study offers methods to improve the accuracy of threat detection and minimize false positives. Finally, the conclusions of Kaya et al. [41] support the findings of this study by showing that ML faces endpoint limitations due to variability in malware behavior and label noise in data. Although the above study shows a significant decrease in accuracy when migrating models from sandboxes to endpoints (from 90% to below 50%), the current study focuses on methods for detecting anomalies in endpoint security, including machine learning algorithms, statistical and correlational event analysis, and baseline deviation approaches. This methodology reduces the impact of variables and increases the accuracy of threat detection. In this way, the presented research is in line with the works under

study, confirming the importance of ML for safety, while at the same time standing out for its unique focus on the temporal and behavioral aspects of endpoints and offering integrated approaches to improve accuracy and reduce false positives.

5. Conclusions

The study found that effective endpoint protection is impossible without a combination of statistical approaches, machine learning techniques, and event correlation capable of detecting both overt and implicit threats, including zero-day attacks, fileless malware, and complex multi-vector scenarios. Particular attention is paid to the basic methods of deviation, which form the basis of behavioral analysis and allow you to record atypical changes in the functioning of systems even in the absence of known attack signatures. The results obtained indicate that the key factors of the practical effectiveness of anomaly detection systems are the quality of the basic model of normal behavior, the balanced speed of its adaptation and the ability to work in conditions of limited computing resources. It is shown that the use of dynamic baselines, multivariate analysis, temporal contextual models and event correlation mechanisms significantly reduces the level of false positives and increases the accuracy of detection. Additional practical value is provided by the implementation of explainable mechanisms that make the results of the models transparent to security analysts and simplify the processes of response, auditing and improving systems.

The contribution of the article to the industry is to form a holistic, systematized vision of anomaly detection in endpoint security, combining algorithmic, architectural and operational aspects. The proposed approach allows not only to compare methods in terms of accuracy, but also to assess their suitability for real implementation, taking into account scalability, resource constraints, and resistance to adaptive attacks. This provides a foundation for future research and practical development of intelligent cyber defense systems capable of operating in dynamic and heterogeneous environments. Further studies could focus on evaluating deep learning architectures, such as recurrent neural networks or autoencoders, for adaptive anomaly detection, particularly in scenarios involving IoT devices, industrial control systems, or edge computing nodes, where limited computational resources and high data variability challenge the robustness and efficiency of security models.

REFERENCES

1. L. Karimli, *Cloud technology security issue*, Proceedings of Azerbaijan High Technical Educational Institutions, vol. 34, no. 11, pp. 114–121, 2023.
2. R. Makhmudova, and K. Dashdamirova, *Analysis of information security problems in the information society environment*, Problems of Information Society, vol. 2, pp. 83–94, 2021.
3. E. Bagirov, *Malware detection based on N-gram analysis*, In Proceedings of the I International Scientific Conference of Students and Young Researchers, pp. 431–433, 2020. Available: $https://www.researchgate.net/profile/Elshan-Baghirov/publication/378268059_Zrrli_proqramlarin_n-qram_analizi_sasinda_askarlanmasi/links/65d05ce101325d46520f6986/Zrrli-proqramlarin-n-qram-analizi-sasinda-askarlanmasi.pdf$
4. K. N. Asgarov, Y. N. Imamverdiyev, and M. M. Abutalibov, *Unsupervised machine learning methods for real-time anomaly detection in endpoints*, Journal of Modern Technology and Engineering, vol. 9, no. 3, pp. 141–155, 2024.
5. M. Williams, and T. C. Mbakwe-Obi, *Integrated strategies for database protection: Leveraging anomaly detection and predictive modelling to prevent data breaches*, World Journal of Advanced Research and Reviews, vol. 24, no. 3, pp. 1098–1115, 2024.
6. N. Rehman, and S. Bukhari, *Precision finance: Leveraging AI for enhanced market efficiency and security*, 2024. Available: https://doi.org/10.13140/RG.2.2.30952.89604
7. V. Chiranjeevi, and D. Malathi, *Anomaly graph: Leveraging dynamic graph convolutional networks for enhanced video anomaly detection in surveillance and security applications*, Neural Computing and Applications, vol. 36, no. 20, pp. 12011–12028, 2024.
8. J. Mo, and Z. Zhang, *Research on university network data anomaly detection and security protection algorithm based on edge computing*, SPIN, vol. 15, no. 2, 2440009, 2024.
9. Y. Lu, *Enhanced network security protection through data analysis and machine learning: An application of GraphSAGE for anomaly detection and operational intelligence*, Journal of Computing and Information Technology, vol. 31, no. 4, pp. 233–249, 2024.
10. J. Paul, *The role of anomaly detection in API security: A machine learning approach*, 2021. Available: https://www.researchgate.net/publication/385587499_The_Role_of_Anomaly_Detection_in_API_Security_A_Machine_Learning_Approach
11. S. K. Dey, and M. M. Rahman, *Effects of machine learning approach in flow based anomaly detection on software defined networking*, 2019. Available: https://doi.org/10.20944/preprints201911.0113.v1

12. America's Cyber Defence Agency, *Incident Response Plan (IRP) Basics*, 2022. Available: https://www.cisa.gov/sites/default/files/publications/Incident-Response-Plan-Basics_508c.pdf
13. K. Arvisais, *Making the case for dynamic baselines*, 2023. Available: https://www.renoster.co/resource/making-the-case-for-dynamic-baselines
14. RoboticsBiz, *Six anomaly detection techniques – Pros and Cons*, 2021. Available: https://roboticsbiz.com/six-anomaly-detection-techniques-pros-and-cons/
15. J. Sleeman, T. Finin, and M. Halem, *Temporal understanding of cybersecurity threats*, In Proceedings of the 6th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE, Baltimore, pp. 115–121, 2020.
16. *Event types and use cases for event correlation*, 2020. Available: https://www.bigpanda.io/blog/event-types-and-use-cases-for-event-correlation/
17. J. Chen, J. Zhang, R. Qian, J. Yuan, and Y. Ren, *An anomaly detection method for wireless sensor networks based on the improved isolation forest*, Applied Sciences, vol. 13, no. 2, pp. 702, 2023.
18. A. Lara-Gutierrez, C. Fernandez-Gago, and J. A. Onieva, *A framework for drift detection and adaptation in AI-driven anomaly and threat detection systems*, Journal of Reliable Computing, vol. 12, no. 4, pp. 233–251, 2025.
19. S. A. Okolie, C. A. Amadi, J. N. Odii, E. C. Nwokorie, and U. C. Onyemauche, *Anomaly detection in heterogeneous cybersecurity data*, Franklin Open, vol. 13, 100426, 2025.
20. P. Moriano, S. C. Hespeler, M. Li, and M. Mahbub, *Adaptive anomaly detection for identifying attacks in cyber-physical systems: A systematic literature review*, arXiv, pp. 1–35, 2024.
21. CyberPhore, *Insider Threat Detection and Prevention: Complete Internal Security Guide 2025*, 2025. Available: https://cyberphore.com/insider-threat-detection-and-prevention-complete-internal-security-guide-2025/
22. W. Zhang, and J. P. Lazaro, *A survey on network security traffic analysis and anomaly detection techniques*, International Journal of Emerging Technologies and Advanced Applications, vol. 1, no. 4, pp. 1–9, 2024.
23. R. Jones, *Techniques and approaches for leveraging LLMs in security analysis*, In *Application of Large Language Models (LLMs) for Software Vulnerability Detection*, IGI Global, London, pp. 75–104, 2024.
24. S. Aghera, *Containerised endpoint security for DevOps environments*, International Journal on Recent and Innovation Trends in Computing and Communication, vol. 11, no. 3, pp. 539–546, 2023.
25. V. Pardosi, *Cost-effective DDoS mitigation: Leveraging Nginx reverse proxy for enhanced server protection*, Jurnal Teknologi Informasi dan Pendidikan, vol. 17, no. 2, pp. 371–382, 2024.
26. D. Rajesh, M. Ganesan, S. Kumarakrishnan, and P. N. Kumar, *Leveraging transfer learning and YOLO for scalable anomaly detection in surveillance systems*, Research Square, pp. 1–42, 2024.
27. H. Gan, H. Zheng, Z. Wu, C. Ma, and J. Liu, *TFD-Net: Transformer deviation network for weakly supervised anomaly detection*, IEEE Transactions on Network and Service Management, vol. 22, no. 1, pp. 941–954, 2024.
28. Z. Ouyang et al., *A cloud endpoint coordinating CAPTCHA based on multi-view stacking ensemble*, Computers & Security, vol. 103, 102178, 2021.
29. O. Okusi, *Leveraging AI and machine learning for the protection of critical national infrastructure*, Asian Journal of Research in Computer Science, vol. 17, no. 10, pp. 1–11, 2024.
30. J. B. Adelusi, *Endpoint security strategies for safeguarding digital infrastructure*, 2023. Available: https://www.researchgate.net/profile/Joshua-Adelusi/publication/387225060_Endpoint_Security_Strategies_for_Safeguarding_Digital_Infrastructure/links/6764b90fe74ca64e1f20b39c/Endpoint-Security-Strategies-for-Safeguarding-Digital-Infrastructure.pdf
31. N. Rehman, and N. Ahmad, *Leveraging machine learning in cybersecurity: Data-driven insights for enhanced information security and cloud infrastructure protection*, 2024. Available: https://doi.org/10.13140/RG.2.2.27545.43367
32. C. J. Nzekwe, and C. J. Ozurumba, *Advanced modelling techniques for anomaly detection: A proactive approach to database breach mitigation*, International Journal of Science and Research Archive, vol. 13, no. 2, pp. 2839–2909, 2024.
33. T. K. Chawdhury, *Beyond the falcon: A generative AI approach to robust endpoint security*, 2024. Available: https://www.dlyog.com/papers/ai4falcon
34. N. Jeffrey, Q. Tan, and J. R. Villar, *A review of anomaly detection strategies to detect threats to cyber-physical systems*, Electronics, vol. 12, no. 15, 3283, 2023.
35. D. Kaul, and R. Khurana, *AI to detect and mitigate security vulnerabilities in APIs: Encryption, authentication, and anomaly detection in enterprise-level distributed systems*, Eigenpub Review of Science and Technology, vol. 5, no. 1, pp. 34–62, 2021.
36. M. Shafi, and N. A. Mirjat, *Enhancing cybersecurity with AI: From anomaly detection to threat mitigation*, Bulletin of Engineering Science and Technology, vol. 1, no. 3, pp. 20–39, 2024.
37. M. A. Elsayed, and M. Zulkernine, *PredictDeep: Security analytics as a service for anomaly detection and prediction*, IEEE Access, vol. 8, pp. 45184–45197, 2020.
38. P. D. Kommisetty, B. M. Kuppala, and H. V. Buvvaji, *Transforming cyber defense: Anomaly detection and predictive analytics for automated threat response*, International Journal of Engineering and Computer Science, vol. 11, no. 8, pp. 25585–25600, 2022.
39. M. Cappello, *A comprehensive analysis of EDR (endpoint detection & response), EPP (endpoint protection platform), and antivirus security technologies*, 2024. Available: https://dione.lib.unipi.gr/xmlui/handle/unipi/16751
40. S. K. Mandala, *Revolutionizing security: Leveraging unsupervised anomaly detection and autoencoder techniques to enhance device security*, Zenodo, pp. 1–6, 2023.
41. Y. Kaya et al., *Demystifying behavior-based malware detection at endpoints*, In Proceedings of the 2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML), IEEE, pp. 921–940, 2024.

APPENDIX A

```cpp
// Example code for analysing network traffic
#include <iostream>
#include <vector>
#include <cmath>

struct NetworkData {
  double packetSize;
  double packetCount;
  // Other network traffic parameters
};

// Function for analysing deviations
bool isAnomaly(const NetworkData& data, double avgSize, double stddev) {
  double zScore = (data.packetSize - avgSize) / stddev;
  return std::abs(zScore) > 2; // If the devation is more than 2 sigma
}

int main() {
  std::vector<NetworkData> trafficData = {
    {100, 50}, {200, 60}, {150, 55}, {300, 75}
  };

  double avgSize = 175;
  double stddev = 50;

  for (const auto& data : trafficData) {
    if (isAnomaly(data, avgSize, stddev)) {
      std::count << "Anomaly detected: " << data.packetSize << " packages" << std::end1;
    }
  }
  return 0;
}
```

Figure 3. Example of a network traffic analysis program.

Source: compiled by the author.

APPENDIX B

```cpp
#include <iostream>
#include <vector>
#include <cmath>

struct Event {
  std::string eventType; // Type of the event
  std::time_t timestamp; // Type of the event
  double value; // Value of the event
};

// Function to check correlation between two types of events
bool checkCorrelation(const Event& event1, const Event& event2, double threshold) {
  // Check if events occur close in time and have significant values
  double timeDiff = std::difftime(event1.timestamp, event2.timestamp);
  if (std::abs(timeDiff) < 300) { // Events happen within 5 minutes
     return event1.value > threshold && event2.value > threshold;
  }
  return false;
}

int main() {
  // Sample events
  std::vector<Event> events = {
    {"LoginAttempt", 1615123456, 15}, // Login attempts
    {"NetworkTraffic", 1615123480, 5000}, // Network traffic
    {"LoginAttempt", 1615123500, 12}, // Login attempts
    {"NetworkTraffic", 1615123520, 3000}, // Network traffic
 };

 // Check correlation between events
 double threshold = 10; // Threshold for event significance
 for (size_t i = 0; i < events.size(); ++i) {
   for (size_t j = i + 1; j < events.size(); ++j) {
     if (checkCorrelation(events[i], events[j], threshold)) {
       std::cout << "Correlation detected between event " << i + 1 << " and event " << j + 1
<<std::end1;
     }
   }
 }
 return 0;
}
```

Figure 4. Example of a correlation system program.

Source: compiled by the author.