International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 5, Issue 6, November- December 2023 |

DOI:10.15662/IJEETR.2023.0506028

AI-Based Data Engineering Pipelines for Real-Time Cybersecurity Threat Detection

Shashikala Valiki

Independent Researcher, India

shashikala.valiki.researcher@gmail.com

ABSTRACT: Timely detection and mitigation of cybersecurity threats is critical for organizations and the broader ecosystem. Numerous incident types, a wide attack surface and a dispersed threat landscape necessitate advanced detection capabilities across multiple areas. AI and ML techniques offer promise in improving the accuracy and efficiency of detection functions, in addition to enhancing elements of the data engineering processes that support them. Fundamentals of detection tasks, as well as Data Engineering in a Real-Time Analysis context, provide foundational guidance for research. A framework is proposed for deploying AI-enhanced data pipelines capable of supporting detection activities in real-time or near-real-time.

The pipeline architecture and individual components emphasize those aspects of data engineering necessary for timely threat detection—ingestion, feature extraction, transformation, quality, governance, compliance, and provenance. Challenge factors affecting these areas are identified, along with appropriate metrics, and a consolidation of support mechanisms and techniques that assist with real-time capability. Together with a set of design, classification, and evaluation guidelines, these provide a comprehensive foundation for the pipeline aspects of Data Engineering within Real-Time Analytics (swiftness, completeness, correctness, and cost-effectiveness).

KEYWORDS: Cybersecurity Threat Detection, AI-Driven Security Analytics, Real-Time Threat Detection, ML-Based Detection Systems, Security Data Pipelines, Threat Intelligence Systems, Feature Extraction for Security, Data Ingestion Pipelines, Security Data Transformation, Data Quality in Security, Security Data Governance, Data Provenance Tracking, Real-Time Analytics Systems, Detection Accuracy Optimization, Security Pipeline Architecture, Threat Detection Metrics, AI-Enhanced Data Engineering, Near Real-Time Monitoring, Security Compliance Systems, Adaptive Threat Detection.

I. INTRODUCTION

Contemporary digital society faces an incessant struggle against cybercriminal activity. Recent statistics estimate that the financial damage caused by cyberattacks is measured in trillions of dollars. Cybercriminals constantly develop new attack patterns, ultimately compromising the effectiveness of common cybersafety defensive strategies. Hence, attention regularly focuses on the development of innovative methods, processes, and tools to detect threat types, implement mitigations, and enforce response and recovery strategies.

Real-time detection of cyber threats still presents several challenges. In particular, the data ingested from logs or packets must be rapidly analyzed to detect abnormalities in behavior, not only at the network level but also at the endpoint and cloud levels. Solutions in edge-cloud architecture have emerged, and various data sources from endpoints, clouds, firewalls, and routers may be exploited to detect different types of threats. Nevertheless, a structured framework that identifies all components required to create complete data engineering pipelines focused on real-time data detection tasks and considers the integration of these components is still lacking.

II. BACKGROUND AND RELATED WORK

The cyber threat landscape has evolved significantly over the past decade, with an alarming rate of data breaches, ransomware attacks, hacktivism, state-sponsored espionage, advanced persistent threats (APTs), and attacks through the Internet of Things (IoT), Industrial Internet of Things (IIoT), and other devices. These diverse attacks often leverage multiple methods (e.g., phishing, social engineering) and increase in severity over time with the backing of well-established attackers. These trends necessitate a comprehensive active, multilayered defense strategy composed of preventive, detective, and responsive controls aligned with the CIA triad of confidentiality, integrity, and availability

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 5, Issue 6, November- December 2023 |

DOI:10.15662/IJEETR.2023.0506028

(or survivability) of data and services. Preventive controls are rarely 100% effective in mitigating all threats; therefore, it is critical to deploy detective and responsive measures that generate awareness about security incidents and enable appropriate incident response and recovery. Such measures are particularly relevant in situations where time is critical—a characteristic that applies to the majority of cyber threats.

As organizations move to the cloud, more centralized security models are also gaining traction to facilitate rapid threat detection and enable specialized teams with global threat visibility to respond to incidents across the entire organization. Advances in machine learning (ML) and artificial intelligence (AI) are making it increasingly possible to implement these controls in real time, either by fully automating suspicious activity detection and response or by generating alerts that expedite the work of security analysts. AI is providing the ability to analyze vast amounts of data with low latency, operate 24/7, and even implement defensive countermeasures. AI is also enabling advanced detection capabilities by thoroughly analyzing historical data and creating a comprehensive knowledge base of legitimate and illegitimate user behavior or network flow patterns. However, achieving this apparent utopia of fully-automated, self-learning, and self-healing detection-and-response AI systems is far from straightforward.

2.1. Cybersecurity Threat Landscape

Cyber threats manifest in many forms, and both the private and public sectors are constantly under attack. Trends indicate an increase in APTs and MASM attacks, as well as a rise in network attacks based primarily on malware. Government and military agencies are being targeted more than bank or credit card companies. The methods used by attackers vary depending on the resource level of the threat actor; while groups with a high maturity level use tailored attack strategies that can remain hidden for a long time, criminal organizations use less sophisticated tools, such as botnets or malware packages that can be installed on the dark web and rented. Both groups rely on shields, including obfuscation and encryption packages, in order to prevent or limit detection. Given these complex and constantly changing threats, a dynamic and adaptive detection strategy at all levels is essential.

Cybersecurity threats can be categorized from different perspectives. From a detective perspective, they can be classified into known, unknown, and zero-day threats, while from a threat actor standpoint, they fall into three categories: insiders, outsiders, and partners. From the perspective of the gathered data information, threats can be divided into system, application, and network threats. By analyzing attack patterns based on MITRE ATT&CK knowledge, it is possible to identify the success factors for each vector and determine how it can actually be detected. Cyber Security Operations Centers (CSOCs) are responsible for pen-testing, malware analysis, blue teaming, red teaming, SOC operations, threat intelligence, and incident management. As prevention is not sufficient, detection capabilities in this context must take the shape of an APT strategy that minimizes the failure to detect critical attacks and implements pre-defined playbooks or run books for known threats.

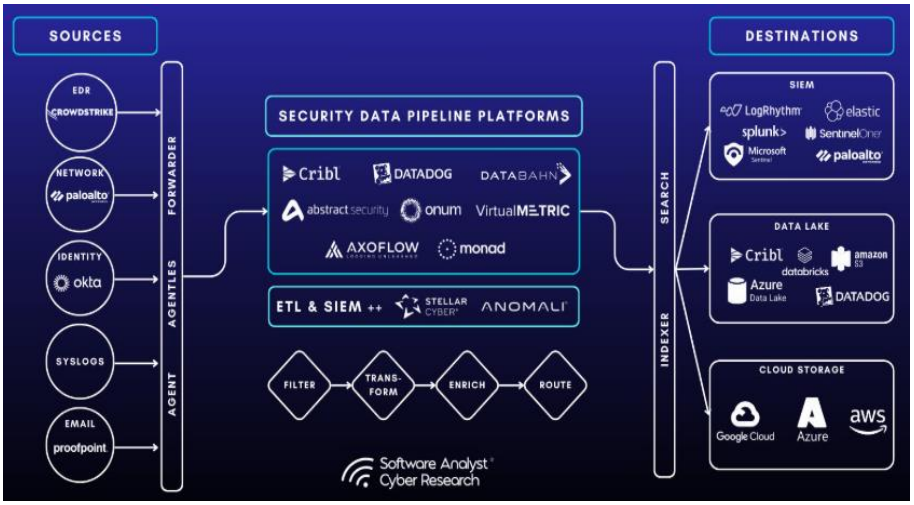


Fig 1: The Rise of Security Data Pipelines

2.2. Data Engineering in Real-Time Analytics

A common pattern in real-time analytic systems applies a streaming architecture combined with parallel processing. Data is continuously ingested from sources and sent as streams to complex event processors for immediate analysis.

International Journal of Engineering & Extended Technologies Research (IJEETR)

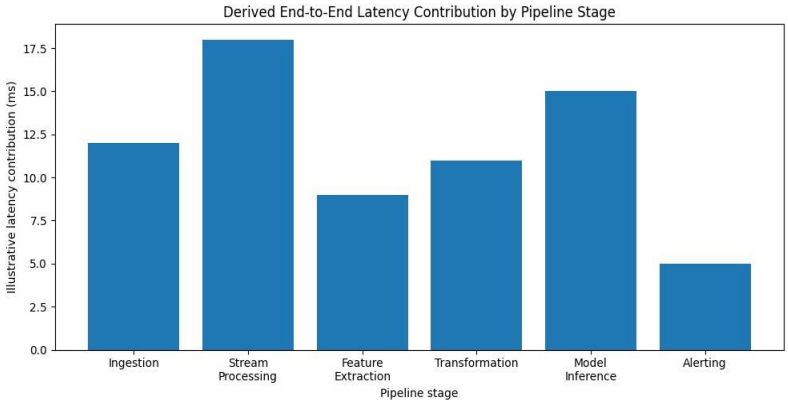
|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 5, Issue 6, November- December 2023 |

DOI:10.15662/IJEETR.2023.0506028

While data engineering for real-time analytics is well-established and documented, several aspects need further development to enable cyber security threat detection with timely, accurate results. Priority is therefor given to the design of such pipelines using directions in applied machine learning, data quality, and data governance to guide the integration of artificial intelligence components.

Real-time streaming analytics processes data in short time windows. Data engineers must choose an appropriate streaming platform, data ingestion pattern and schema that suit the application's requirements for timeliness, throughput and fault tolerance. Balancing trade-offs incurs additional complexity as pipelines are required to support detection of deceptive and malware-based threats in high-velocity, high-volume environments. Three AI-enhanced data engineering aspects are therefore important for cyber security threat detection support: introduction of machine learning models, ensuring data quality appropriate for detection tasks, and implementing suitable data governance policies.



Equation 1. End-to-end pipeline latency

The describe a real-time pipeline with several sequential stages:

- ingestion
- stream processing
- feature extraction
- transformation
- model inference
- alerting

Let:

- L_i = ingestion latency
- L_s = stream-processing latency
- L_f = feature-extraction latency
- L_t = transformation latency
- L_m = model-inference latency
- L_a = alerting latency

Then total end-to-end latency is the sum of stage latencies:

$$L_{\text{total}} = L_i + L_s + L_f + L_t + L_m + L_a$$

Step-by-step derivation

A single event enters the system.

After ingestion, elapsed time is:

$$T_1 = L_i$$

After stream processing:

$$T_2 = L_i + L_s$$

After feature extraction:

$$T_3 = L_i + L_s + L_f$$

After transformation:

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 5, Issue 6, November- December 2023 |

DOI:10.15662/IJEETR.2023.0506028

$$T_4 = L_i + L_s + L_f + L_t$$

After model inference:

$$T_5 = L_i + L_s + L_f + L_t + L_m$$

After alert generation:

$$T_6 = L_i + L_s + L_f + L_t + L_m + L_a$$

Therefore,

$$L_{\text{total}} = T_6 = L_i + L_s + L_f + L_t + L_m + L_a$$

III. METHODOLOGY

Research quality depends on complementing design choices with demonstrable rigor and theoretical grounding. A suitable methodology section justifies study choices and provides detailed context for readers' assessment of study quality. Details from deployed environments, frameworks, models, algorithms, and datasets enable comprehensive replication; additional specifics can enhance validation of an experiment bank's detection performance. Importantly, the research design reflects the objective—building and deploying AI-based data engineering pipelines for real-time threat detection.

Data source and experimental setup choices follow from the research goal, while coverage of the end-to-end design of a recognizable deployment affords valuable guidance. Together, these elements most clearly align with the audience's practical interest and need for solid theoretical formulation. Future work can take the overall framework and focus on further detail in methods, implementation, or deployment. Ideally, such an approach clarifies implementation of the AI-enhanced data pipeline framework across a range of detection problems, especially real-time operational capabilities for timely and accurate threat detection.

Table 1. Pipeline components and their role

Pipeline Component	Role in the Research Article	Main Evaluation Focus
Data Sources	Raw telemetry from network, endpoint, and cloud sources	Threat-surface coverage
Ingestion & Streaming	Move events with low delay and fault tolerance	Latency, throughput
Feature Extraction	Build model-ready security features	Feature usefulness, speed
Transformation	Normalize, enrich, reshape data	Schema fit, robustness
Data Quality Layer	Ensure correctness, completeness, timeliness	Trustworthy inputs
Governance & Compliance	Enforce privacy, lineage, retention, access control	Auditability, compliance
Model Integration	Run anomaly detection / classification / scoring	Precision, recall, F1
Monitoring & Feedback	Support SLAs, retraining, alerting, observability	Operational impact

3.1. Framework for AI-Enhanced Data Pipeline Design

Components of an AI-enhanced data engineering pipeline using well-defined data flows, integration points, and governance patterns.

A cyber threat detection task can be supported by one or more heterogeneous data-engineering pipelines that are integrated into a larger information system to monitor the environment. Each AI-enabled data pipeline comprises several components that help implement data ingestion, feature extraction and transformation, data quality and governance, and data provenance and lineage. These components are connected by a well-defined architecture and data flows. Addition of AI/ML-based upstream components enhances decision-making and provides an opportunity for run-time data governance. Furthermore, these AI-related components can be treated as integration points for individual contributors or changes.

A data pipeline for threat detection can include a variety of data sources and a combination of processes that help enable the end use case in real time with high fidelity. With the cyber threat landscape in mind, a dedicated framework comprising these components—together with their integration points, data flows, and basic data-governance patterns—improves and propagates data quality. Such a framework addresses the mapping of individual file-system, database, and external-service metadata files, as well as the flow and transformation of data among the various components. Path-tracking mechanisms validate data parameters and flags for the end users. Examples of data sources include packet captures from switch ports, network flow metadata, and cloud and on-premise endpoint telemetry, while data flows

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 5, Issue 6, November- December 2023 |

DOI:10.15662/IJEETR.2023.0506028

encompass sub-component integration and ingress into both AI/ML model integration points and operational dashboards.

IV. OBJECTIVE OF THE STUDY

The objective of this study is twofold: to define real-time detection requirements and to explain how the data engineering component of a cyber analytics pipeline can be implemented with the architectural capabilities associated with artificial intelligence. The expected contribution includes a framework for integrating AI techniques into cyber analytics data pipelines and a description of how the data engineering aspect must be designed to support real-time detection. To this end, two hypotheses are examined. The first posits that although the detection of a given threat occurs in near real time, the analytic model only provides an alert after a sufficient number of confirmed indications. The second asserts that an AI-enhanced data-engineering pipeline can incorporate data quality, provenance, governance, and model-integration considerations while still supporting real-time detection.

Real-time detection is defined as providing an alert as soon as sufficient evidence has accumulated to satisfy established criteria, thereby enabling defensive action before significant damage has occurred. Detection must be timely but not necessarily instantaneous. Short detection latency is important, as it is typically one of the main reasons for deploying a real-time analytics capability. Nevertheless, overheads must be kept within acceptable limits, so that the detection can genuinely be classified as real time.

Equation 2. Real-time requirement condition

Let:

- $L_{\max}$ = maximum acceptable latency threshold

For real-time operation:

$$L_{\text{total}} \leq L_{\max}$$

Substitute Equation 1:

$$L_i + L_s + L_f + L_t + L_m + L_a \leq L_{\max}$$

4.1. Study Aims and Research Goals

Real-time detection of cybersecurity and fraud threats is an area of keen interest for enterprises to mitigate financial losses, brand damage, and customer discontent. Cybersecurity data engineering pipelines with AI-enhanced components help operationalize detection capabilities and supplement security operations with timely insights. The threat landscape, relevant research, and a framework for architecting AI-enabled data pipelines for real-time analytic use cases are summarized. Four specific objectives characterize these pipelines, along with measurable metrics to assess success.

The study aims to outline an enabling architecture covering data ingestion and streaming, feature extraction, transformation and quality, governance, and AI model integration within the pipeline. Data engineering pipelines with real-time analytic properties are being studied, with a focus on timeliness, scale, and fault-tolerant deployment. Hypothesis testing is grounded in a collection of computational experiments evaluating experimental designs, implementations, datasets, and metrics, considered jointly.

Fig 2: Security Data Pipeline Platforms

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 5, Issue 6, November- December 2023 |

DOI:10.15662/IJEETR.2023.0506028

V. RESEARCH SUMMARY

Research progressed through a combination of a literature review and design science approach, addressing the following overarching research questions: what techniques enable the real-time detection of cyber threats? Which techniques can be captured in a framework for AI-enhanced data engineering pipelines for real-time detection? The contribution of the literature review phase lay in providing a rationale for, and informing the identification of, research topics for subsequent phases. The primary focus was therefore narrowed to the evaluation of AI-based techniques that, together with the underlying data-engineering pipeline, enable, extend, or improve the efficacy of detection in real time in order to reduce the opportunities for attackers to execute successful attacks.

Research built a comprehensive framework covering the end-to-end design of data-engineering pipelines—spanning data ingestion, stream processing, feature extraction and transformation, data quality, and governance and compliance design—enabling the integration of AI models. An architecture specified data flows, starting from source systems and encompassing the complete data-processing life cycle from source to consumed data. A discussion illustrated how the approach captures real-time constraints and supports the evaluation of real-time detection techniques across a comprehensive set of dimensions. Together, these elements instantiate an AI-enhanced data-engineering pipeline suitable for the real-time detection of cybersecurity threats.

5.1. Framework for AI-Enhanced Data Pipeline Architecture

AI-Enhanced Data Pipeline Architecture

An AI-driven data pipeline for real-time threat detection consists of four primary components: data sources, data-processing stages, model integration, and feedback loops for model-enhancement data. Each data source can be connected to any combination of the pipeline components, which together support the core threat-detection task. Two classes of data sources can be identified: those that generate direct inputs to detection models, and those that produce data that feed transitory, supporting components, but do not directly impact final model predictions.

Anomaly-detection models require data that capture the normal behaviour of endpoints within an organisation or, if protocol-level indicators are used, the standard behaviour of any protocol involved in network communication. These models are trained on normal data, and any deviation in future observations is deemed anomalous. Anomalies in the data are comparatively evaluated using alertness levels and different indicators to assign severity labels. Such labelling facilitates the tuning of SLAs beyond mere precision and recall metrics: detection times can impact the effect of some detections on operational workflows.

VI. ARCHITECTURE OF AI-DRIVEN DATA PIPELINES

Real-time analytics data pipelines facilitate the persistent acquisition and analysis of data as it streams from the source, enabling automated detection and alerting on significant events. The architecture of such pipelines comprises multiple components that ingest data from the source, prepare and transform it, feed it through one or more AI models, and present the results. These components must operate in a continuous manner to minimize latency and support constant processing of the data. The architecture of real-time detection data pipelines includes the entire workflow required for detection, encompassing the data source, data ingestion, stream processing, AI model consumption, and presentation of results.

For a given detection task using AI-based models, the architecture of the data pipeline includes the relevant data source, the method of data ingestion into the stream processing engine, the transformations applied to the data before they are presented to the model, and a description of the feedback loops that inform continuous retraining and updates to the model deployed in production. While this section uses the detection of network attacks as a concrete example, similar pipeline architectures can be defined for other types of detection tasks.

6.1. Data Ingestion and Stream Processing

Real-time analytics rely on streaming data processing systems that ingest data from multiple sources, modify its format for internal APIs, and make it available for downstream processing with a determined acceptable latency. Data streams may contain a few bytes per second or millions of records per second, so the management of the latency for data ingestion and processing represents a major challenge. With increasing speed, this latency can no longer be neglected, especially for low-volume data sources. Typically, data are organized in dedicated topics for each source, instance, and/or interest category and made available to interested parties for consumption. Due to the nature of threat detection

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 5, Issue 6, November- December 2023 |

DOI:10.15662/IJEETR.2023.0506028

tasks, the support of fault-tolerant mechanisms is essential. Therefore, the ingestion platform should keep all input data until the fate of each message or event is known, which may be a few seconds later.

Real-time data analysis implies that timestamps on alerts generation have a real relation with the events in the environment: when a threat is detected, a team should be alerted and ready to respond to it. Therefore, latency is essential to measure how close the end of processing is from "real time." Each application has an upper threshold time that should not be crossed, and alerts indicated with latencies below this limit are the most relevant. The probability of missing an anomaly also reflects the processing accuracy. A low task-specific F1 score indicates that latent attacks are not covered properly or not detected at all; therefore, an operational window should be detected with enough margin for action.

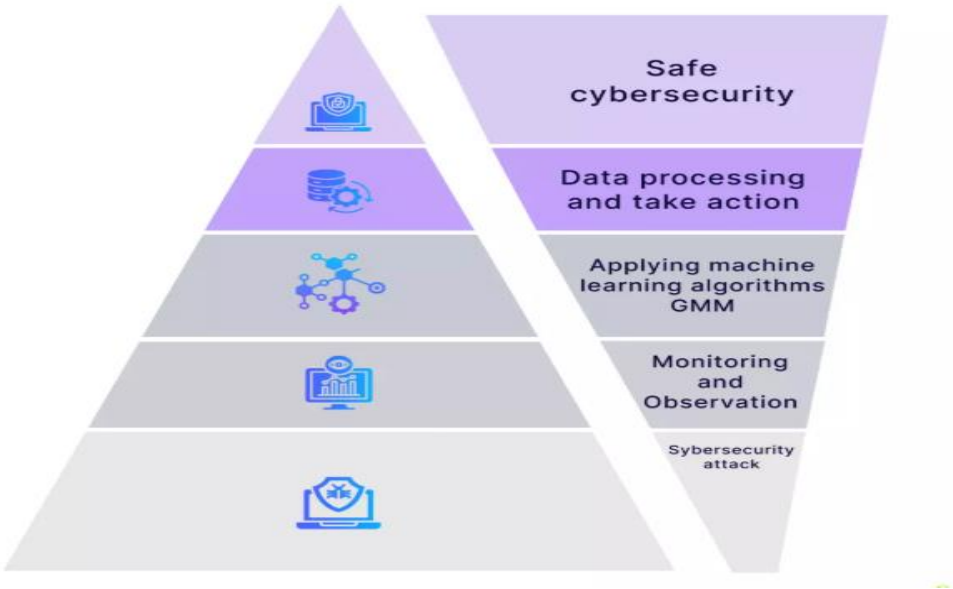


Fig 3: AI in Cybersecurity

6.2. Feature Extraction and Transformation

Although effective deep learning techniques may eliminate the need for legacy feature engineering in some contexts, they are typically characterized by long training times and large model sizes, which can hinder deployment resources and operational speed for real-time detection tasks. Universally applicable features like network flow statistics, however, are often overlooked within the real-time context.Feature extraction and transformation follow data ingestion and stream processing and may be divided into four categories: generic features of low dimensionality, general features of small dimensionality, detection-specific features of moderate dimensionality, and detection-specific features of high dimensionality. Feature extraction takes the shortest time and least processing from data sources with the lowest bandwidth.

Generic features may be temporally enriched statistics collected across distinct time windows as described in section 5.1. Normalization of such statistics—particularly in the context of Real-Time Feature Engineering—alleviates concerns regarding feature drifts and model robustness. For network flow statistics, an online feature store supports real-time embedding in detection system models. The large data volume and training requirements associated with detection-specific feature engineering are balanced by offline preparation and periodic updating during nonpeaking request periods.

Equation 3. Throughput

Let:

- N = number of processed samples/events
- Δt = elapsed time

Then throughput Q is:

$$Q = \frac{N}{\Delta t}$$

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 5, Issue 6, November- December 2023 |

DOI:10.15662/IJEETR.2023.0506028

Step-by-step derivation

If 500 events are processed in 10 seconds:

$$Q = \frac{500}{10} = 50 \text{ events/s}$$

In general, divide total processed volume by total processing time:

$$Q = \frac{\text{processed volume}}{\text{time}}$$

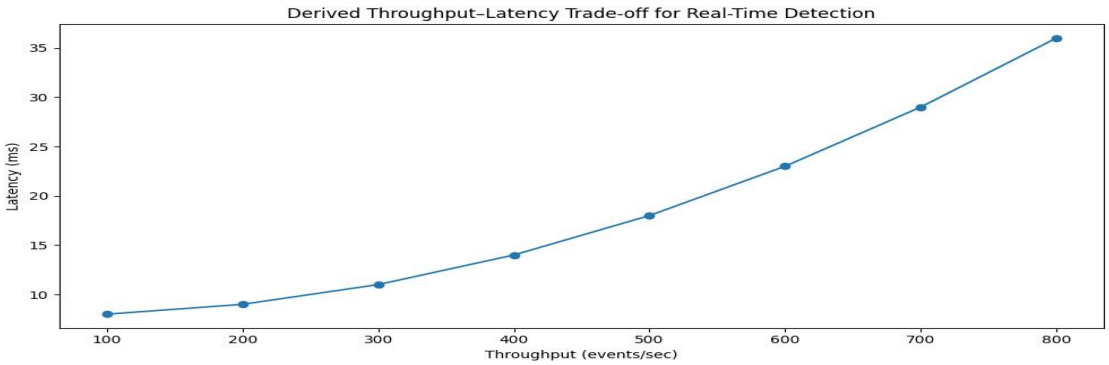
So the final formula is:

$$Q = \frac{N}{\Delta t}$$

VII. DATA QUALITY, GOVERNANCE, AND COMPLIANCE

Achieving stringent data quality requirements is crucial for developing trust in real-time analytics solutions used in high-stakes scenarios like cybersecurity, alongside concerns for GDPR compliance from regulators. Automated data quality monitoring enables detection and resolution of quality issues before model retraining prompts model staleness. Pioneering work on data quality dimension metrics establishes an initial framework, further expanded here within the specific context of real-time detection tasks. Since the quality of source data influences downstream processing, an additional feature quality dimension ensures the correct content, structure, and timing of feature extraction datasets.

To determine data origin, transformation history, and resulting quality dimensions, data lineage tracing underpins the data quality monitoring framework and enables data governance across the entire analytics stack. With evolving regulations in the area of personal data privacy, compliance with policies governing the storage, handling, and processing of sensitive data must be ensured. Thus, pipeline deployment provides an opportunity to implement privacy-preserving measures such as anonymization, differential privacy, and separation of duties, all of which affect the detection task's accuracy.



7.1. Data Provenance and Lineage

Data provenance refers to the lineage or origin of the data and the history of transformations performed on it from inception to the current state. Providing detailed data provenance is essential for ensuring accountability, facilitating data quality assessment, and detecting abnormal behavior. Data flow management systems offer basic lineage tracking, linking data products back to their origins and enabling auditing. However, additional mechanisms contribute to data lineage by capturing the impact of stream processing on data products and exposing complete pathway details.

Unlike an ETL (extraction, transformation, loading) process, a streaming data pipeline continuously reads input streams, searches for stateful and temporal patterns, takes actions, and produces output streams, all in relation to time. Pipeline definition and management capabilities in frameworks such as Apache NiFi, Apache Beam, and AWS Step Functions provide supported lineage, tracing the processing effect on data products and supporting lineage exploration. For full trail visibility, organizations also need to capture data flows from the source to their streaming pipeline products and integrate both aspects.

The architecture must define how to capture detailed lineage, enabling organizations to audit and assess accountability for stream processing effects on data products. Whenever a product of the data pipeline is received or generated, details

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 5, Issue 6, November- December 2023 |

DOI:10.15662/IJEETR.2023.0506028

of the data lineage—previous, current, and next destination—should be marked, along with the clock time used for generation.

Table 2. Real-time detection metrics

Metric	Meaning	Desired Direction	Why Important
Latency	Time to produce a detection after event arrival	Lower is better	Needed for real-time response
Throughput	Events/samples processed per unit time	Higher is better	Needed for scale
Precision	Fraction of predicted attacks that are truly attacks	Higher is better	Reduces false positives
Recall	Fraction of real attacks that are detected	Higher is better	Reduces false negatives
F1 Score	Harmonic balance of precision and recall	Higher is better	Balances both concerns
Operational Impact	Burden on analysts and systems	Lower is better	Determines real usability

7.2. Privacy-Preserving Techniques

Privacy-preserving mechanisms safeguard sensitive data from unauthorized disclosure while maintaining utility for decision-making. Several techniques, including anonymization, differential privacy, and access controls are employed to protect personally identifiable information in the gathered data. In the context of machine learning, privacy-preserving measures may impact the accuracy of cyber threat detection models. However, achieving the right balance should be a priority to protect the individuals involved, as well as organizations and the general public, from present and future cyber threats.

Anonymization techniques are the oldest and most common for privacy protection. Data anonymization entails removing unique identifiers, such as name and address, from datasets to prevent reputation or identity injury by data disclosure, while providing utility for model training and operationalization. Pseudonymization involves replacing values with fictional substitutes or “codes.” Although data records may contain values that relate to real-world identifiers, the substituted values are fictitious or indirectly identifiable only when cross-related with other data sources. While anonymized data cannot be used to trace back to individuals, it can serve secondary purposes without the need for the individuals’ consent.

VIII. EVALUATION METHODOLOGIES FOR REAL-TIME DETECTION

Evaluating detection methods that support real-time operation involves specific design choices. A thorough approach considers various aspects that affect the practices, challenges, and quality of the detection methods being evaluated.
An effective detection method is one that fulfills the desired objectives efficiently and accurately. Timeliness is highly relevant, especially for detection methods with low latency requirements. Latency quantifies the time taken by the model to produce predictions after the arrival of an input instance, so it must be measured from the model's perspective. Latency can also be evaluated at a broader level by combining the accumulative latency of each element in the inference stream.

Another aspect is throughput, which indicates the volume of traffic the detection method can process in a given time period, usually expressed in packets per second or bytes per second. These two parameters, latency and throughput, address the performance of the detection method. When a detection method serves as a component in a larger system, the overall operational impact must also be evaluated. The operational impact assesses the extent to which the detection method influences the functioning of other elements in the overall system.

8.1. Metrics for Timeliness and Accuracy

Measuring the trade-off between detection accuracy and timeliness considerations remains a priority when deriving detection performance evaluations , . Metrics evaluating these characteristics include latency, throughput, precision, recall, F1 score, and operational impact. Note that a malicious event, action, or behavior affecting an asset or dedicated service that may lead to disturbance, degradation, or denial of service needs to be supplied as ground truth.

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 5, Issue 6, November- December 2023 |

DOI:10.15662/IJEETR.2023.0506028

• Latency. Time taken for a model to process a single sample. The lower the latency, the better it is for real-time use cases.
• Throughput. Amount of samples processed by an online model in a given timeframe. While a higher value is preferred for detection, it needs to remain within the latency limit for it to serve its purpose.
• Precision. Percentage of ground-truth positive samples from newly predicted positive samples. Evaluating the false positives of a detection helps determine if it is really of use or not from the operational cost standpoint. A higher value is preferred.
• Recall. Percentage of newly predicted positive samples from actual positive ground-truth samples. Evaluation of false negatives indicates how many malicious actions were missed by the detection. A detect-and-respond strategy may tolerate a lower recall, while a prevent strategy cannot. A higher recall is always preferred.
• F1 score. Combines precision and recall values to provide a balance. High scores indicate a good detection model.
• Impact on the operational team. Explores the operational overhead introduced by the detection mechanism on the security team and supports closing or iterating on false positives in a timely manner.

Fig 4: Security Architecture for the Data Pipeline

8.2. Benchmarking and Datasets

An extensive set of publicly and privately available datasets, as well as benchmark data for different evaluation tasks, serve the validation of proposed detection mechanisms. For benchmarking, the latest version of the Problem Detection in Network Traffic (PDNT) challenge from Detection and Classification of Crowd Anomalies (DCCA) 2022 is considered. Real-Time Detection of Network Attacks (RTDNA) and Open Threat Research Data Repository (OTR) datasets are also utilized for validation. In support of upcoming works, datasets belonging to 2021 and 2022 Capture the Flag (CTF) competitions organized by Detectify Labs on the Hack The Box (HTB) forum are shared privately. As these CTF challenges are hosted on a public government website, data capturing is strictly prohibited. The reaction of existing tools and rules on the server is continuously monitored during the CTF competitions, and their observations are finally captured.

The very first publication appears on the capture of Windows Event Logs (WEL) of a real machine installation, where host data and activity telemetry are fed to a WEL-SIEM-Visualization pipeline. Using these logs, WindowsEnclosureMonitor analyzes both normal and abnormal behaviors, extracts signals from Telemetry, and helps detect potential attacks on Windows environments through extra feature extraction from categorized roles/services. The second publication materializes during the WindowsENF request–response time. A major Windows back-end related to credit card companies on the same time zone is analyzed, and abnormal traffic appears in the response time graph. The next publication finds a sudden increase in Wireshark number during specific timeframes, and a few potential Anomaly Detection Models Detection-Time-Agnostic model are efficiently ruled out in that area through incidence monitoring.

Table 3. Data-quality and governance dimensions

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 5, Issue 6, November- December 2023 |

DOI:10.15662/IJEETR.2023.0506028

Dimension	Interpretation in Cybersecurity Pipeline
Completeness	Required telemetry fields/events are present
Correctness	Values and labels reflect actual state
Timeliness	Data arrives soon enough to be actionable
Consistency	Formats and semantics match across sources
Provenance / Lineage	Source and transformation path are traceable
Privacy Compliance	PII and sensitive records are handled lawfully
Auditability	Decisions and transformations can be reviewed

IX. DEPLOYMENT AND OPERATIONAL CONSIDERATIONS

Deployment and operational aspects impact the effectiveness of data pipelines for real-time detection of cyber threats. Threat detection models developed on empirical datasets can be seamlessly integrated with existing security stacks to provide real-time protection against emerging attacks. Multiple deployment options are available depending on the use case and performance requirements of the detection task. Pipelines for network threat detection can be deployed at the network edge, where latency-sensitive detection tasks are performed using distributed stream processing systems such as Apache Flink, and a hybrid model deployment strategy is employed to meet end-to-end latency goals. Monitoring of endpoint telemetry, indicator of compromise (IOC) feeds, and cloud workloads is in general less time sensitive, and processes model predictions by polling the respective serving systems at regular intervals.

Data engineering pipelines supporting operational deployments need to consider aspects of monitoring, observability, and incident response. Detection models should not only generate alerts in case of positive predictions, but also provide contextual environment, time window, and severity information to assist cyber threat analysts in investigation. Dashboards should provide monitoring views for security analysts, and generate alerts if the SLAs defined for different detection tasks are violated. SLAs need not only monitor the health of the detection models with respect to accuracy, recall, and precision, but also define runbooks for incident handling.

9.1. Real-Time Deployment Architectures

Real-time deployment options fall into three categories: edge, cloud, and hybrid. The method of choice affects the overall shape of the streaming topology and where trained models reside. When deploying detection solutions, factors such as the applicability of edge processing appliances, real-time performance requirements, and the connectedness of elements in the security stack should be considered. In dedicated security deployments, various types of monitoring appliances on the edges collect different types of telemetry. The performance/test network traffic, endpoint telemetry, OS and application log files, network flow, Hyper-V, and connectors for Azure, Office 365, AWS and GCP are ingested for monitoring. Multiple types of detection logs are produced using the detections developed for each type of telemetry. Detection engines are developed for the network-based and telemetry-based data during training phase and when deployed they generate alerts when the detection logic gets triggered. The detection engines produce alert files using the individual logic defined for each of them.

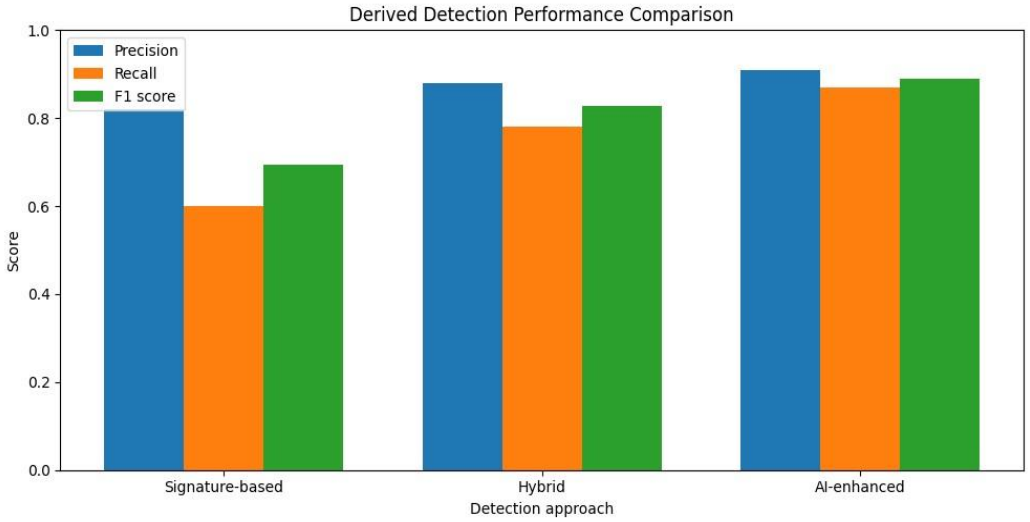
When the real-time requirements are not stringent, the workload can be shifted to the public cloud. Cloud-based processing resources can easily be scaled up when the volume or storage/compute needs increase. A single Instance may be sufficient to run the detection engine for a specific set of alerts. However, the cloud service should be selected carefully, keeping in mind the nature of the data being processed. The core feature extraction and classification process typically take time to complete, but once it reaches a good operating threshold, alerts get generated in real-time. During inference, the model produces the results at a much faster pace than training. Additionally, it is important to make SLA considerations for any cloud resources. For instance, the log sampling at 1 hour or 1.5 hour makes Automate and Ansible log analyses even slower. To avoid situations where alerts are generated late, a single window of logs being used for detection and ignoring all the logs more than this window for generation of alerts in the specific time period or the dedicated real-time or real-time-like data sample should be considered.

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 5, Issue 6, November- December 2023 |

DOI:10.15662/IJEETR.2023.0506028



Equation 4. Precision

Let:

- TP = true positives
- FP = false positives

Predicted positives are:

$$TP + FP$$

Among them, the correct positive predictions are TP.

Therefore precision P is:

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

Step-by-step derivation

1. Count all alerts raised by the model.
2. Split those alerts into:
   - correct alerts = TP
   - incorrect alerts = FP
3. Total alerts raised:

$$TP + FP$$

4. Fraction that are truly malicious:

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

9.2. Monitoring, Observability, and Incident Response

Dashboards, alerting mechanisms, service-level agreements, and runbooks streamline incident handling for the real-time data pipeline. Dashboards provide a comprehensive view of the pipeline's health for quick decision-making and mitigation strategies. Alerting mechanisms escalate issues based on severity and service-level agreements, either automatically to runbooks or manually to responsible stakeholders. The combination of automated alerts based on severity and complete incident runbooks with step-by-step mitigation details significantly facilitates fast responses to incidents.

Effective monitoring of the end-to-end pipeline enables swift detection of potential disruptions. Specialized engines, such as Apache Flink and Apache Kafka, enable monitoring of data flow rates, processing latency, node resource utilization, and availability of internal systems. Alerts are configured with specific thresholds and service-level agreements for triggering automatic alerts or assignment of manual escalation responsibilities based on incident severity and impact. Dashboards additionally help track service-level agreement compliance and recent data flow drops for a high-level overview. Alerts from upstream engines are consolidated in dedicated channels tied to runbooks or mitigation documents.

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 5, Issue 6, November- December 2023 |

DOI:10.15662/IJEETR.2023.0506028

Additional engines augment coverage by analyzing other critical sources, such as batch ingestion logs and data quality pipelines. Completeness, consistency, and accuracy checks are implemented with corresponding alerts for real-time or near-real-time dashboards. Analysis of deviations yields insights for updates to underlying data quality metrics and alert configurations. Monitoring of service-level agreement importance and consolidation in a single source helps prioritize updates.

Dedicated runbooks for each ingestion component and overall health monitoring eliminate knowledge gaps during the natural personnel churn inherent in security operations. Each runbook has clear ownership to ensure up-to-date documentation for quick and efficient incident resolution whenever required, thereby establishing an efficient security operation.

X. CASE STUDIES AND APPLICATIONS

Demonstrating practical deployments across diverse domains, two case studies illustrate the proposed data-engineering solution for real-time threat detection. Supporting network traffic as well as endpoint and cloud security, the presented approaches incorporate and evaluate various components of the detection pipelines.

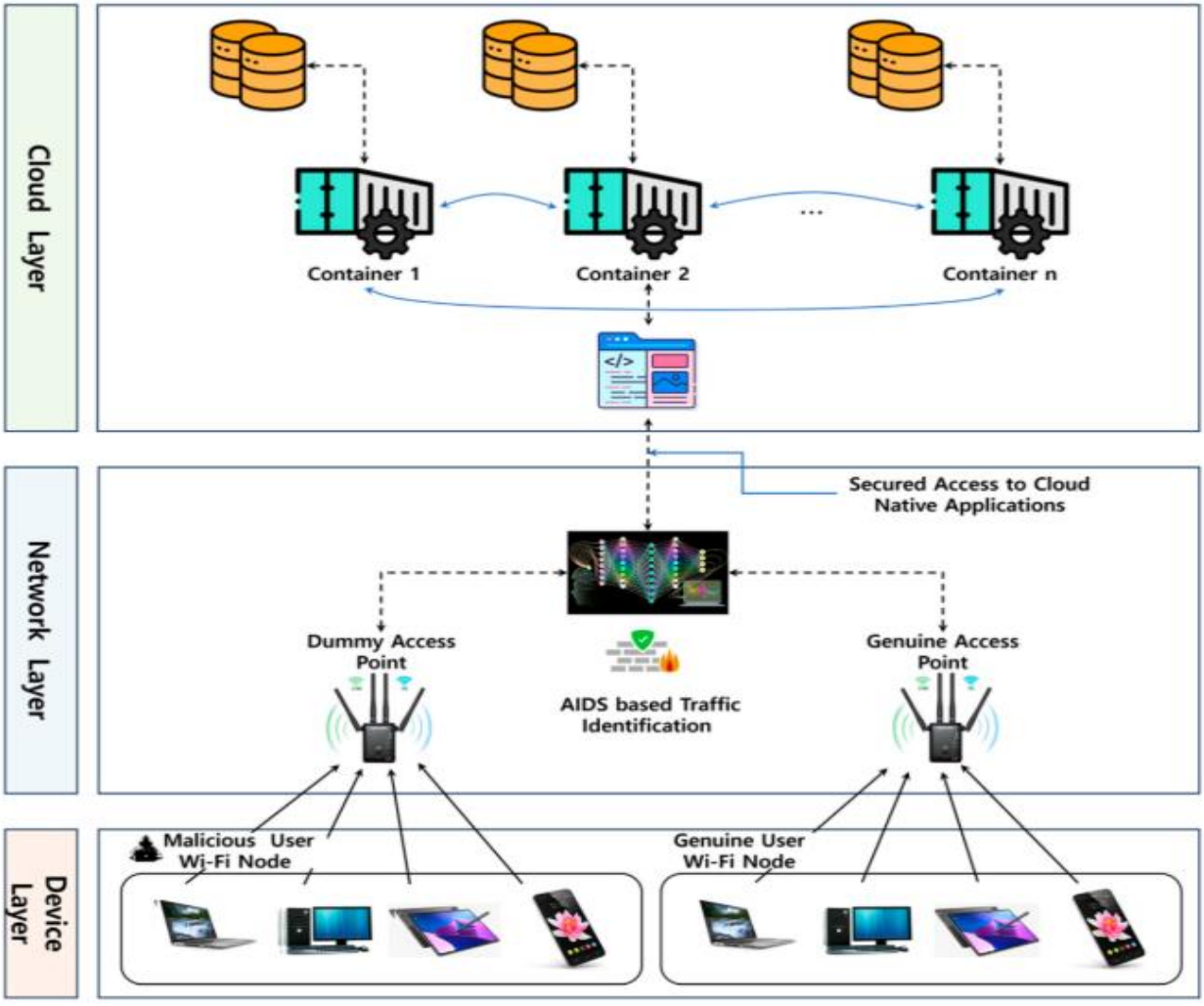


Fig 5: Cyber Threat Detection Framework for Secure Cloud-Native Microservices

1 Network Traffic Analytics

Monitoring network traffic for potential security threats has long been a priority beyond standard intrusion detection systems (IDS). Protocol analysis, particularly when delving into the intelligence section of flows, has been applied to

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 5, Issue 6, November- December 2023 |

DOI:10.15662/IJEETR.2023.0506028

detect planning behavior patterns of command-and-control sessions. Subtle deviations from expected traffic profiles should also raise suspicions. Earlier work established the feasibility of behavior modeling to reveal features that cluster well in normal operating conditions but lead to clear separability when the system encounters unexpected behavior. Similar analysis techniques have been employed for reasoning, validation, and verification, but achieving successful detection tasks requires not just sound reasoning but also supporting telemetry above the network layer.

Supported by the proposed data-engineering pipelines, network flow data are used to explore several approaches to analyze traffic for indications of compromise. A modified IDS signature matching expert system aims to generalize beyond known attack signatures through probabilistic reasoning and semantic similarity. Additionally, fixed-defined rule sets, independent of noted signature matching, and hybrid systems functioning as add-on detectors complement these efforts. The latter rely on a collection of rule-based conditions hypothesized to indicate malicious or other specific behavior patterns. Beyond flow telemetry and detection engines, protocols such as DNS or DHCP provide further data sources for suspected compromised traffic identification, although these are typically consumed by dedicated commercial security monitoring solutions.

10.1. Network Traffic Analytics

Cybersecurity and compliance practices typically require organizations to continuously monitor network traffic and actively detect anomalies. In particular, all inbound and outbound traffic must be inspected, classified, and recorded, with connections to known malicious IP addresses flagged for further action. Anomalies in network statistics, such as unusual bandwidth consumption, protocol distribution, or geographical distribution, should be detected. Packet data may also be inspected for sensitive information, and specific packets, such as those running known exploitation tools, should be flagged. At the detection and response levels, network traffic anomalies may provide signals for attack indicators at other levels, such as dropped packets from an external IP or repeated accesses to cloud services from multiple users in multiple locations.

Network traffic, however, presents unique challenges for real-time detection systems. First, the substantial volume of data requires algorithms capable of analysing high-capacity streams at high rates. Indeed, the scale of the industry makes the application of ML for on-device detection improbable, and cloud solutions commonly have latency requirements too stringent for standard model architecture to keep pace. Second, systems must remain responsive to changes in normal patterns; a sudden local surge in traffic from a single region might not be problematic, but the same increase from dozens of regions may indicate the early stages of a global-scale attack. Third, as knowledge of many attacks relies on detecting combinations of packets originating from inappropriate users, detecting deviation from the distribution of a set of normal behaviours, even if they are not clearly anomalous individually, is critical due to the high false-positive rate of signature-based techniques. Finally, despite minor cases of admissible breaches, the processing of any personally identifiable information (PII) in transit or at rest must comply with regulations such as the General Data Protection Regulation (GDPR).

Equation 5. Recall

Let:

- TP = true positives
- FN = false negatives

Actual positive cases are:

$$TP + FN$$

Among them, detected positives are TP.

So recall R is:

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

Step-by-step derivation

4. Count all truly malicious cases.
5. Separate them into:
   - detected attacks = TP
   - missed attacks = FN
6. Total actual attacks:

$$TP + FN$$

5. Fraction detected:

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 5, Issue 6, November- December 2023 |

DOI:10.15662/IJEETR.2023.0506028

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

10.2. Endpoint and Cloud Security

Comprehensive Orchestration of Network, Endpoint, Cloud, and OS Security

Implementing adequate security at the enterprise perimeter using firewalls, proxies, and routers does not guarantee network protection. New attack strategies (e.g., insider threats, social engineering) require defending endpoints and ensuring up-to-date, hardened configurations for security support, virtual machines, and infrastructures-as-a-service. Telemetry from web browsers, external drives, VDI platforms, security configuration, anti-virus software, Windows, and cloud AWS and Azure accounts—closely monitored and analyzed for worrying patterns (e.g., list of installed tools, anti-virus status, recent telemetric anomalies)—provide timely, contextual indicators of compromise or risk. Detection and alerting should occur with minimal delay (sub-second to few seconds). Data-driven approaches trained and tested on available datasets can be deployed quickly, reducing costs, time, and effort.

With the wide adoption of cloud services by organizations, security monitoring extends into these environments. Data from the cloud service provider's Management Console (Console, Security, AWS Config—the temporal service that stores the dashboard) is aggregated and events analyzed to minimize the time from telemetry ingestion to contextual awareness summary (minutes to seconds). The entire cloud stack can be monitored with regards to Configuration or Badge violations (indicating security best practices violations) or telemetry analysis: reconnaissance from source to destination devices has become very common, enabling cloud compromise detection on configuration structure or any type of lateral movement in the cloud.

XI. RESULTS

Results are presented in several parts. First, the data engineering requirements for real-time detection are summarized, as well as a framework to support the design of AI-enabled data engineering pipelines. Subsequently, an AI-enhanced architecture for real-time data engineering pipelines is proposed, before the structure of the pipeline is described. Finally, the methods for data ingestion and stream processing, feature extraction and transformation, and data quality, governance, and compliance are formalized.

The objective of the research was to support the design of detect-then-response security systems that offer real-time detection and response for a broad range of cyber threats and vulnerabilities. The evaluation criteria for achieving this objective were that all detection nodes share a common pipeline architecture and that the pipeline design methods explicitly address the data engineering requirements for real-time detection. Evaluation involved a rich set of AI-enhanced network, endpoint, and cloud detection applications. For each application, the pipeline architecture was identified. In particular, AI-enhanced approaches for data ingestion and stream processing, feature extraction and transformation, data quality support and validation, and compliance with data access and retention policies were established.

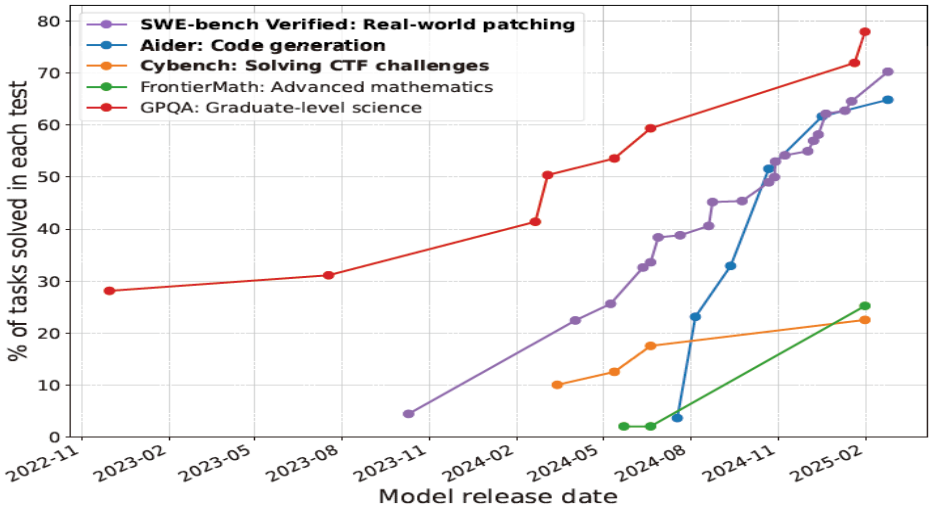


Fig 6: AI's Impact on the Cybersecurity Landscape

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 5, Issue 6, November- December 2023 |

DOI:10.15662/IJEETR.2023.0506028

XII. CONCLUSIONS

Economic and societal developments in cyberspace have far outpaced the ability of cybersecurity companies to devise effective countermeasures for increasingly complex, targeted, and automated attacks. As a result, it is imperative to enhance detection capabilities for a variety of attack types using real-time methodologies. The data engineering layers of detection systems have received relatively little attention in the academic literature, yet they are critical components that support detection tasks. This research makes several contributions to data engineering for real-time analytics through a framework that delineates the considerations, components, and flows in AI-enhanced data pipelines. The framework can guide the design of data engineering architectures and processes that underpin detection tasks. Furthermore, explicit operation and evaluation criteria have been articulated for real-time detection, and a suite of supporting deployment methodologies has been established.

Practical case studies complement the theoretical work and illustrate the feasibility of real-time detection for network-oriented threats, including C2 traffic, protocol-level anomalies, and anomalous flow patterns, as well as for cloud-oriented threats, such as data exfiltration and malware detection in cloud storage. Future work should focus on the identification of additional types of threats that can be detected in real time, the creation of replication packages for existing methodologies, and the definition of evaluation protocols that address both task performance and operational feasibility.

REFERENCES

[1] Kolla, S. K. (2023). Explainable AI and ML Models for Transparent Clinical Decision Support. Journal for ReAttach Therapy and Developmental Diversities, 6, 2444-2460.
[2] Davuluri, P. N. Integrating Artificial Intelligence into Event-Driven Financial Crime Compliance Platforms.
[3] Aitha, A. R. (2023). CloudBased Microservices Architecture for Seamless Insurance Policy Administration. International Journal of Finance (IJFIN)-ABDC Journal Quality List, 36(6), 607-632.
[4] Pamisetty, A. (2022). Big Data can Generate Major Opportunities for Manufacturing Supply Chains. International Journal of Scientific Research and Modern Technology, 1(12), 238–251. https://doi.org/10.38124/ijsrmt.v1i12.1186
[5] Moustafa, N., & Slay, J. (2015). UNSW-NB15 dataset. Military Communications Conference.
[6] Garapati, R. S. (2022). AI-Augmented Virtual Health Assistant: A Web-Based Solution for Personalized Medication Management and Patient Engagement. Available at SSRN 5639650.
[7] Inala, R. Designing Scalable Technology Architectures for Customer Data in Group Insurance and Investment Platforms.
[8] Kolla, S. H. (2021). Rule-Based Automation for IT Service Management Workflows. Online Journal of Engineering Sciences, 1(1), 1-14.
[9] Segireddy, A. R. (2020). Cloud Migration Strategies for High-Volume Financial Messaging Systems.
[10] Yandamuri, U. S. (2023). An Intelligent Analytics Framework Combining Big Data and Machine Learning for Business Forecasting. International Journal Of Finance, 36(6), 682-706.
[11] Singireddy, J. (2023). Finance 4.0: Predictive analytics for financial risk management using AI. European Journal of Analytics and Artificial Intelligence (EJAAI) p-ISSN, 3050-9556.
[12] Somasundaram, P. (2023). Improving real-time job monitoring for cloud-based data pipelines. International Journal of Computer Engineering and Technology, 14(3), 39–47.
[13] Davuluri, P. N. (2020). Event-Driven Architectures for Real-Time Regulatory Monitoring in Global Banking.
[14] Kolla, S. H. (2023). Deep Learning–Driven Retrieval-Augmented Generation for Enterprise ITSM Automation: A Governance-Aligned Large Language Model Architecture. Journal of Computational Analysis and Applications, 31(4).
[15] Singireddy, J. (2022). Leveraging Artificial Intelligence and Machine Learning for Enhancing Automated Financial Advisory Systems: A Study on AIDriven Personalized Financial Planning and Credit Monitoring. Mathematical Statistician and Engineering Applications, 71(4), 16711-16728.
[16] Amistapuram, K. Energy-Efficient System Design for High-Volume Insurance Applications in Cloud-Native Environments. International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering (IJIREEICE), DOI, 10.
[17] Mahesh Recharla, (2020), "Targeted Gene Therapy for Spinal Muscular Atrophy: Advances in Delivery Mechanisms and Clinical Outcomes", International Journal of Science and Research (IJSR), 9(12), 1921-1934. https://dx.doi.org/10.21275/SR20126161624, https://www.ijsr.net/getabstract.php?paperid=SR20126161624
[18] Kulkarni, A. R., Kumar, N., & Rao, K. R. (2023). Big data analytics and monitoring frameworks for scalable data pipelines. Big Data Mining and Analytics, 6(2), 139–153.

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 5, Issue 6, November- December 2023 |

DOI:10.15662/IJEETR.2023.0506028

[19] Botlagunta Preethish Nandan, "Data Analytics-Driven Approaches to Yield Prediction in Semiconductor Manufacturing," International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering (IJIREEICE), DOI 10.17148/IJIREEICE.2021.91217.
[20] Garapati, R. S. (2023). Optimizing Energy Consumption in Smart Build-ings Through Web-Integrated AI and Cloud-Driven Control Systems.
[21] Chowdhury, R. H. (2021). Cloud-based data engineering for scalable business analytics solutions: designing scalable cloud architectures to enhance the efficiency of big data analytics in enterprise settings. Journal of Technological Science & Engineering (JTSE), 2(1), 21-33.
[22] Vamsee Pamisetty, Lahari Pandiri, Sneha Singireddy, Venkata Narasareddy Annapareddy, Harish Kumar Sriram. (2022). Leveraging AI, Machine Learning, And Big Data For Enhancing Tax Compliance, Fraud Detection, And Predictive Analytics In Government Financial.
[23] Gottimukkala, V. R. R. (2021). Digital Signal Processing Challenges in Financial Messaging Systems: Case Studies in High-Volume SWIFT Flows.
[24] Aitha, A. R. (2023). Cloud-Native Big Data AI/ML Framework for Risk Intelligence and Fraud Control in Banking and Insurance Ecosystems. Available at SSRN 6157967.
[25] Sheelam, G. K., & Nandan, B. P. (2021). Machine Learning Integration in Semiconductor Research and Manufacturing Pipelines. International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), DOI, 10.
[26] Chakilam, C., Suura, S. R., Koppolu, H. K. R., & Recharla, M. (2022). From Data to Cure: Leveraging Artificial Intelligence and Big Data Analytics in Accelerating Disease Research and Treatment Development. Journal of Survey in Fisheries Sciences. https://doi.org/10.53555/sfs.v9i3.3619.
[27] Nagabhyru, K. C. (2023). Accelerating Digital Transformation with AI Driven Data Engineering: Industry Case Studies from Cloud and IoT Domains. Educational Administration: Theory and Practice, 29(4), 5898-5910
[28] Bonawitz, K., et al. (2023). Secure aggregation for federated learning. Google Research.
[29] Kalisetty, S., & Singireddy, J. (2023). Optimizing Tax Preparation and Filing Services: A Comparative Study of Traditional Methods and AI Augmented Tax Compliance Frameworks. Available at SSRN 5206185.
[30] Goutham Kumar Sheelam. (2022). Reconfigurable Semiconductor Architectures For AI-Enhanced Wireless Communication Networks. Kurdish Studies, 10(2), 1027–1040. https://doi.org/10.53555/ks.v10i2.3867.
[31] Yandamuri, U. S. (2021). A Comparative Study of Traditional Reporting Systems versus Real-Time Analytics Dashboards in Enterprise Operations. Universal Journal of Business and Management
[32] Dwaraka Nath Kummari, Srinivasa Rao Challa, “Big Data and Machine Learning in Fraud Detection for Public Sector Financial Systems,” International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), DOI: 10.17148/IJARCCE.2020.91221
[33] Sheelam, G. K., & Nandan, B. P. (2022). Integrating AI And Data Engineering For Intelligent Semiconductor Chip Design And Optimization. Migration Letters, 19, 2178-2207.
[34] Mangalampalli, B. M. (2023). AI-Driven Anomaly Detection in Healthcare Claims Data: A Business Intelligence Perspective. Journal of Rare Cardiovascular Diseases.
[35] Mukesh, A., & Aitha, A. R. (2021). Insurance Risk Assessment Using Predictive Modeling Techniques. International Journal of Emerging Research in Engineering and Technology, 2(4), 68-79.
[36] Palanichamy, R. S. T. (2023). AI and data governance: Enhancing security, privacy, and accountability. International Journal on Science and Technology, 14(1), 1–10
[37] Dwaraka Nath Kummari,. (2022). Machine Learning Approaches to Real-Time Quality Control in Automotive Assembly Lines. Mathematical Statistician and Engineering Applications, 71(4), 16801–16820. Retrieved from https://philstat.org/index.php/MSEA/article/view/2972
[38] Meda, R. End-to-End Data Engineering for Demand Forecasting in Retail Manufacturing Ecosystems.
[39] Mangala, N. (2022). Real-Time Data Quality Monitoring and Gating Frameworks in Cloud-Based Data Pipelines. International Journal of Research and Applied Innovations, 5(6), 8197-8219.
[40] Nasiri, S., Rahmani, A. M., & Rezaei, M. (2023). A systematic review of big data stream processing frameworks and applications. Journal of Big Data, 10(1), 67.
[41] Inala, R. (2021). A New Paradigm in Retirement Solution Platforms: Leveraging Data Governance to Build AI-Ready Data Products. Journal of International Crisis and Risk Communication Research, 286-310.
[42] Gottimukkala, V. R. R. (2023). Privacy-Preserving Machine Learning Models for Transaction Monitoring in Global Banking Networks. International Journal of Finance (IJFIN)-ABDC Journal Quality List, 36(6), 633-652.
[43] Malempati, M., Pandiri, L., Paleti, S., & Singireddy, J. (2023). Transforming financial and insurance ecosystems through intelligent automation, secure digital infrastructure, and advanced risk management strategies. Jeevani, Transforming Financial And Insurance Ecosystems Through Intelligent Automation, Secure Digital Infrastructure, And Advanced Risk Management Strategies (December 03, 2023).

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 5, Issue 6, November- December 2023 |

DOI:10.15662/IJEETR.2023.0506028

[44] Pamisetty, A. (2022). Integrating Big Data, AI, and Financial Modeling in Cloud-Based Insurance and Banking Ecosystems. AI, and Financial Modeling in Cloud-Based Insurance and Banking Ecosystems (December 05, 2022).
[45] Sriram, H. K., ADUSUPALLI, B., Singreddy, S., & Malempati, M. (2021). Revolutionizing Risk Assessment and Financial Ecosystems with Smart Automation, Secure Digital Solutions, and Advanced Analytical Frameworks. Murali, Revolutionizing Risk Assessment and Financial Ecosystems with Smart Automation, Secure Digital Solutions, and Advanced Analytical Frameworks (December 27, 2021).
[46] Kolla, T. (2023). Predictive ETL Failure Detection in Healthcare Data Pipelines Using Anomaly Detection Algorithms. International Journal of Medical Toxicology & Legal Medicine.
[47] Mangalampalli, B. M. Intelligent Data Profiling for Healthcare Data Lakes Using AI-Enhanced Analytics.
[48] Recharla, M., & Chitta, S. AI-Enhanced Neuroimaging and Deep Learning-Based Early Diagnosis of Multiple Sclerosis and Alzheimer's.
[49] Nasiri, S., et al. (2023). A systematic review of big data stream processing frameworks and applications. Journal of Big Data, 10(1), 67.
[50] Botlagunta, P. N., & Sheelam, G. K. (2020). Data-Driven Design and Validation Techniques in Advanced Chip Engineering. Global Research Development (GRD) ISSN, 2455-5703.
[51] Meda, R. (2020). Designing Self-Learning Agentic Systems for Dynamic Retail Supply Networks. Online Journal of Materials Science, 1(1), 1-20.
[52] Valiki, D., & Kummari, D. N. (2021). Rule-Based Decision Systems for the Automation of Audit Sampling. International Journal of Emerging Trends in Computer Science and Information Technology, 2(4), 105-114
[53] Mangala, N. (2021). CI/CD Pipeline Automation for Enterprise Data Artifacts Using Azure DevOps. Universal Journal of Business and Management, 1(1), 1-18. https://doi.org/10.31586/ujbm.2021.1363
[54] Nagubandi, A. R. (2023). Advanced Multi-Agent AI Systems for Autonomous Reconciliation Across Enterprise Multi-Counterparty Derivatives, Collateral, and Accounting Platforms. International Journal of Finance (IJFIN)-ABDC Journal Quality List, 36(6), 653-674
[55] Amistapuram, K. (2022). Fraud Detection and Risk Modeling in Insurance: Early Adoption of Machine Learning in Claims Processing. Available at SSRN 5741982.
[56] Gadi, A. L. , Gadi, A. L. Kannan, S. , Kannan, S. Nandan, B. P. , Nandan, B. P. Komaragiri, V. B. , & Komaragiri, V. B. (2021). Advanced Computational Technologies in Vehicle Production, Digital Connectivity, and Sustainable Transportation: Innovations in Intelligent Systems, Eco-Friendly Manufacturing, and Financial Optimization. Universal Journal of Finance and Economics, 1(1), 87-100. https://doi.org/10.31586/ujfe.2021.1296.
[57] Inala, R. Advancing Group Insurance Solutions Through Ai-Enhanced Technology Architectures And Big Data Insights.
[58] Kannan, S., Nuka, S. T., Pamisetty, V., Gadi, A. L., Krishna, H., & Koppolu, R. ENHANCING AGRICULTURAL EQUIPMENT AND MEDICAL DEVICES Pamisetty, V. (2020). Optimizing tax compliance and fraud prevention through intelligent systems: The role of technology in public finance innovation. Available at SSRN 5250796.
[59] Kummari, D. N., & Burugulla, J. K. R. (2023). Decision Support Systems for Government Auditing: The Role of AI in Ensuring Transparency and Compliance. International Journal of Finance (IJFIN)-ABDC Journal Quality List, 36(6), 493-532.
[60] Ring, M., Wunderlich, S., Grüdl, D., Landes, D., & Hotho, A. (2019). Flow-based intrusion detection datasets. Computers & Security, 86, 147–167.
[61] Adusupalli, B., Singireddy, S., & Pandiri, L. Implementing Scalable Identity and Access Management Frameworks in Digital Insurance Platforms. International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), DOI, 10.
[62] Segireddy, A. R. (2022). Terraform and Ansible in Building Resilient Cloud-Native Payment Architectures. International Journal of Intelligent Systems and Applications in Engineering, 10, 444-455.
[63] Gottimukkala, V. R. R. (2020). Energy-Efficient Design Patterns for Large-Scale Banking Applications Deployed on AWS Cloud. power, 9(12).
[64] Garapati, R. S., & Kanna, S. R. A Digital Twin-Enabled Predictive Maintenance Framework Leveraging Multi-Agent Reinforcement Learning and Industrial IoT Data.
[65] Pamisetty, V., Dodda, A., Lakarasu, P., Singireddy, J., & Challa, K. (2022). Optimizing Digital Finance and Regulatory Systems Through Intelligent Automation, Secure Data Architectures, and Advanced Analytical Technologies. Secure Data Architectures, and Advanced Analytical Technologies (December 10, 2022).
[66] Nagabhyru, K. C. (2023). From Data Silos to Knowledge Graphs: Architecting CrossEnterprise AI Solutions for Scalability and Trust. Available at SSRN 5697663.
[67] Pamisetty, A. (2021). A comparative study of cloud platforms for scalable infrastructure in food distribution supply chains.

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 5, Issue 6, November- December 2023 |

DOI:10.15662/IJEETR.2023.0506028

[68] Aiswarya, K., Reddy, P., & Kumar, V. (2023). Fault detection and mitigation strategies in data pipeline systems. International Journal of Data Engineering, 14(1), 22–34.
[69] Yandamuri, U. S. (2022). Big Data Pipelines for Cross-Domain Decision Support: A Cloud-Centric Approach. International Journal of Scientific Research and Modern Technology (IJSRMT).
[70] Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). KDD Cup dataset issues. CISDA.