

ATHEA: An Explainable AI Framework for Unsupervised APT Detection in Windows Event Logs

Prof. Aman Kamble¹, Yash Shivarkar², Vishak Ranjan Mohanta³, Ayush Chaudhary⁴, Divya Vijaykumar Pujari⁵

MIT School of Computing, MIT ADT University, Pune, India

¹aman.kamble@mituniversity.edu.in, ²yashmitadt@gmail.com,

³vishakranjan@gmail.com, ⁴ayushchy012@gmail.com, ⁵deevpujari@gmail.com

Abstract— The increasing complexity and sophistication of the APTs (Advanced Persistent Threats), with the use of their blending in with the benign and normal administrative activity techniques, traditional signature-based detection system has been proved to be insufficient. While Windows Event Logs (EVTX) provides a rich and highly detailed source for endpoint analysis, their sheer volume, complexity and the inherent noise make it hard to analyse it manually. We present in this paper, ATHEA, end-to-end unsupervised learning framework designed to not only detect security vulnerabilities and events in EVTX logs, but also to provide deep, easy and human-readable explanations.

The system attempts to combine a robust detection pipeline, which starts with the parsing of the EVTX log files, then it proceeds to an extensive feature engineering (extracting over 50 time-based, statistical, and security-specific features). The detection part is handed over to a hybrid unsupervised model (combining Isolation Forest (IF) and Local Outlier Factor (LOF)) to effectively identify both global and local outliers without relying on the pre-labelled datasets.

The key contributing factor of this work is its explainability (XAI) framework, which is designed explicitly to address the huge problem in the current Security and ML relationship, that being the "black box" nature of the AI when it come up to certain decisions. The first layer utilizes SHAP (SHapley Additive exPlanations) in order to provide effective, quantitative feature-level contribution analysis, identifying which feature is more responsible for the event to be flagged as anomaly than rest of the other features. This then goes through the second layer, which leverages Google Gemini AI (for now) to deconstruct all the technical aspects of the event and to construct a human-readable and understandable report with all of the context in place, providing the natural-language threat analysis to the analyst.

To further harden the claims and analysis, the system automatically contextualizes detected anomalies by mapping them to the MITRE ATT&CK framework and leverages a built-in knowledge base of over 500 critical EventIDs for dynamic, context-aware risk scoring. The purpose of ATHEA is to bridge the gap between raw, black-boxed anomalies alerts and actionable, easily interpretable events/ alerts as a result empowering security analysis and analysts to detect and respond to the APTs more effectively.

Keywords— Cybersecurity, Anomaly Detection, XAI, Explainable AI, Windows Event Logs, EVTX, Unsupervised Learning, SHAP, MITRE ATT&CK, APT, Threat Hunting, Generative AI, Large Language Models (LLMs), Isolation Forest, Local Outlier Factor (LOF), Feature Engineering, Log Analysis, Endpoint Detection and Response (EDR), Security Operations Center (SOC).

I. INTRODUCTION

In the modern cybersecurity landscape, the "castle-and-moat" philosophy of reactive defence, built on signature-based tools, has proven insufficient [9]. Threat actors, particularly those conducting Advanced Persistent Threats (APTs), have shifted to "living-off-the-land" (LotL) techniques. These attacks leverage legitimate administrative tools and processes to blend in with benign activity, making them exceptionally difficult to detect with traditional Intrusion Detection Systems (IDS) that hunt for known-bad signatures [10]. Windows Event Logs (EVTX) provide a comprehensive source of endpoint telemetry for detecting such activity, recording everything from user logins and process creations to network connections. However, this data's sheer volume—often millions of events per day—makes manual analysis unfeasible and creates a significant "needle in a haystack" problem for forensic investigators.

Machine learning (ML) has emerged as a promising solution to this data-volume challenge. However, many current ML-based security systems are built on supervised learning, which carries a critical flaw: they require large, high-quality, and accurately labelled datasets of "malicious" and "benign" activity. Such datasets are rare, expensive to create, and are quickly rendered obsolete by the rapid evolution of new attack techniques. Consequently, these models excel at finding known threats but fail to detect the novel or zero-day exploits characteristic of APTs.

Furthermore, the "black box" nature of many complex ML models presents a significant adoption barrier for Security Operations Center (SOC) analysts [11]. An unexplainable alert, such as a simple "anomaly" flag, provides no actionable context, contributes directly to "alert fatigue" [3], and erodes analyst trust. For a security tool to be effective, it must not only detect a threat but also provide the "why" and "so what" in a trustworthy, interpretable manner.

To address these distinct but related challenges—the failure of signatures, the limitations of supervised learning, and the crisis of explainability—we introduce ATHEA. ATHEA is a novel; end-to-end framework designed specifically for the unsupervised and explainable detection of APTs in log data. Its architecture is built to be:

- a) *Unsupervised*: It identifies novel threats and anomalies without pre-labelled data, using a hybrid anomaly detection model (Isolation Forest and Local Outlier Factor) to find both global and local outliers.
- b) *Explainable*: It moves beyond simple "anomaly" flags by providing a novel, dual-layer explainability (XAI) framework. This framework combines SHAP (SHapley Additive exPlanations) for quantitative, feature-level analysis with Google Gemini AI for generative, natural-language threat analysis and actionable recommendations.
- c) *Context-Aware*: It enriches raw logs with extensive feature engineering (50+ features) and automatically cross-references detections with a built-in knowledge base of over 500 EventIDs and the MITRE ATT&CK framework.
- d) *Accessible*: It provides an interactive Streamlit-based dashboard for analysts to easily upload, process, and investigate logs.

This paper's primary contribution is the design and evaluation of the ATHEA framework, which uniquely synthesizes unsupervised detection with a hybrid (SHAP + GenAI) XAI pipeline. We detail its system architecture, its core detection and explainability methodologies, and demonstrate its performance in identifying sophisticated attack patterns, ultimately bridging the gap between raw, noisy log data and actionable, human-readable security intelligence.

II. LITERATURE REVIEW

This research is grounded in four key areas that chart the evolution of cyber defence: the failure of traditional reactive models, the rise of AI-driven autonomous hunting, the trade-offs between AI detection methodologies, and the critical, non-negotiable need for explainability.

A. The Evolution from Reactive Defence to Proactive Hunting

Traditional cybersecurity was built on a reactive, "castle-and-moat" philosophy. This approach relied on signature-based tools such as firewalls and Intrusion Detection Systems (IDS) to block known threats and malware signatures [9]. This model, however, fundamentally fails against modern Advanced Persistent Threats (APTs) and zero-day exploits [10]. APT actors specifically employ "living-off-the-land" (LotL) techniques, using legitimate, pre-installed system tools (like PowerShell or WMI) to conduct their operations. These malicious activities are nearly indistinguishable from benign administrative behaviour, rendering signature-based detection ineffective [4].

This limitation spurred the evolution toward proactive threat hunting, where analysts actively search for hidden, anomalous adversarial activities that have already bypassed automated systems [11]. Unlike traditional incident response, which reacts to an alert, threat hunting is a continuous, hypothesis-driven process. However, given the vast-volume of telemetry (like EVTX data), manual hunting is not scalable, leading to analyst fatigue, human error, and critically delayed detection times [1].

B. The Rise of AI-Driven Autonomous Hunting

Artificial Intelligence (AI) and Machine Learning (ML) have emerged as the critical enablers to solve this scalability problem [8]. This marks the next evolutionary step: autonomous threat hunting, which leverages two synergistic concepts:

- a) *Predictive Analytics*: The use of ML models to analyse historical and real-time data streams to forecast and identify potential threats, often before they fully materialize [2], [9].
- b) *Automated Threat Hunting*: The application of AI to autonomously and proactively search for, investigate, and even correlate indicators of compromise (IoCs) without constant human involvement [3], [5].

AI-driven systems can process terabytes of telemetry data, identify subtle anomalies invisible to a human analyst, and correlate disparate, low-level indicators to construct a coherent attack narrative [6]. This shift from human-led to machine-augmented hunting is essential for enabling faster detection, reducing response times, and creating a defence that can adapt to evolving adversarial techniques [1], [8].

C. Methodologies in AI-Powered Threat Detection

Research in AI-based threat hunting, as outlined by Shakil et al. [4], can be broadly categorized. An effective, resilient system like ATHEA must navigate the trade-offs of these approaches:

- a) *Structured (Supervised) Methods*: These rely on deterministic models and supervised learning using large, labelled datasets. They are effective at detecting known Tactics, Techniques, and Procedures (TTPs) [7]. Their primary weakness, however, is their brittleness. They require a constant, expensive effort to maintain labelled datasets and are, by definition, unable to detect novel or zero-day attacks for which no signature or label exists.
- b) *Unstructured (Unsupervised) Methods*: This approach, central to ATHEA, aims to discover hidden or unknown threats through unsupervised learning, anomaly detection, and clustering on heterogeneous data [6]. By modelling a baseline of 'normal' behaviour, these methods excel at detecting APTs and novel exploits that evade signature-based systems. This is ideal for the "needle in a haystack" problem of event logs.
- c) *Situational Methods*: These integrate contextual awareness, combining external threat intelligence, behavioural analytics, and environmental factors like asset criticality and user privileges [4], [8]. This method provides the 'so what' context, moving from a simple anomaly to a prioritized threat.

While structured methods are inherently reactive, a purely unstructured approach has its own significant challenges, which are addressed in the following sections.

D. The Critical Barrier: The 'Black Box' Problem

Despite their power, the "black box" nature of many advanced ML models is one of the most significant barriers to their adoption in operational security [11]. A model that simply flags an event as "anomalous" (score: 0.98) without any reasoning is unactionable.

This lack of transparency undermines analyst trust, hinders effective operational decision-making, and directly contributes to "alert fatigue," where analysts begin to ignore alerts they cannot investigate or validate [3]. This paper argues that for an AI detection system to be viable, it must be an Explainable AI (XAI) system.

E. The Unsupervised Conundrum: False Positives and Noise

A challenge directly related to the XAI problem is the high false-positive rate (FPR) of many unsupervised models. Cybersecurity data is inherently noisy; not every anomaly is malicious. A benign administrative action, a system update, or a rare but legitimate user behaviour can all be flagged as anomalous. Without a mechanism to distinguish a benign outlier from a malicious threat, the system merely trades data volume for alert volume, overwhelming the SOC with false positives [9].

F. Synthesizing a Solution: XAI and Context

The literature points to a clear solution, which forms the hypothesis of this paper: the weaknesses of unsupervised detection (high FPR, black box) can be mitigated by integrating them with situational methods and XAI. By combining an unsupervised model (like ATHEA's) with a rich, situational knowledge base (like the MITRE ATT&CK framework and EventID intelligence) and a robust XAI framework (like SHAP and GenAI), a system can:

- Explain why an event is anomalous (via XAI).
- Contextualize if that anomaly matters (via MITRE).

This synthesis is the key to creating a trusted, effective, and scalable threat-hunting platform.

G. Other Organizational and Ethical Challenges

Beyond these core technical barriers, the literature also notes other considerations. Proactive user monitoring raises privacy and compliance concerns under frameworks like GDPR [11]. Furthermore, organizations face a growing AI skill gap and cultural resistance from employees who may fear automation [1], [8]. An accessible and explainable system like ATHEA, which augments rather than replaces the analyst, is designed to help bridge this specific gap.

III. SYSTEM ARCHITECTURE AND METHODOLOGY

The ATHEA architecture is engineered as a multi-stage, end-to-end pipeline. Its design purpose is to systematically address the core challenges of unsupervised threat detection: transforming high-volume, noisy, and raw log files into a small set of high-fidelity, actionable, and explainable security

insights. As illustrated conceptually in Fig. 1, each module acts as a filter and enrichment layer, progressively building context to move from raw data to an analytical narrative.

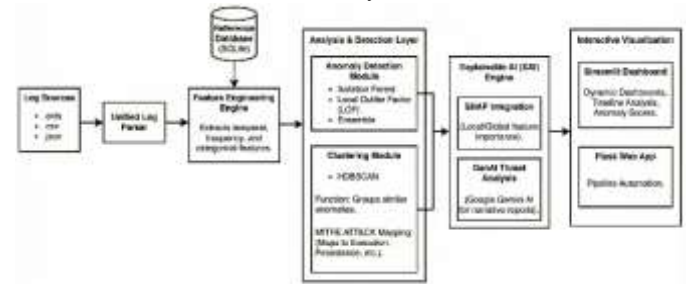


FIGURE 1: ARCHITECTURE OF THE PROPOSED XAI FRAMEWORK FOR UNSUPERVISED APT DETECTION IN WINDOWS EVENT LOGS

A. Detection Pipeline Overview

The core process maps to the architectural diagram and follows five primary stages:

- Log Ingestion:** Raw logs (EVTX, CSV, JSON, TXT) are uploaded as the 'System Logs (Input)'.
- Log Preprocessing:** A multi-step process (including parsing and feature engineering) to standardize and enrich the data.
- Threat Detection (AI/ML Layer):** An unsupervised hybrid model (IF, LOF, or Ensemble) scores each event's anomalousness.
- Explainable AI (XAI) & Risk Scoring:** Detected anomalies are contextualized (clustered, mapped to MITRE) and then explained using a dual-layer XAI model. A risk score is applied to prioritize alerts.
- Output / Analyst Interface:** Explanations and findings are presented to the analyst via an interactive Streamlit dashboard.

B. Core Pipeline Module Details

- Log Preprocessing:** This stage corresponds to the 'Log Preprocessing' block and is a critical prerequisite for any real-world analysis. It involves two major sub-steps:
 - Parsing and Standardization:** The ingestion module is designed for flexibility, supporting multiple log formats as a Security Operations Center (SOC) rarely relies on a single log source. Its primary capability is the robust parsing of binary Windows EVTX files, with error handling for corrupted records. All logs are then standardized into a common dictionary format and then a structured data frame. This standardization is essential for the subsequent feature engineering stage, ensuring a consistent and vectorized data structure regardless of the input source. During this stage, initial temporal context (e.g., IsBusinessHours, IsWeekend) is added to enrich the raw event.

2. *Feature Engineering*: This is arguably the most critical part of preprocessing, as it encodes domain-specific security knowledge into the numerical data the ML model will consume. This step directly supports our "situational" detection methodology [4], [8], transforming raw log entries into behaviourally-aware features. Over 50 features are extracted, grouped into five categories:

- i. *Temporal (8 features)*: Includes IsNightTime, TimeSincePrevEvent, and cyclical time features (e.g., HourSin, HourCos).
- ii. *Categorical (6 features)*: LevelSeverity, EventIDRarity, UserEventCount.
- iii. *Security (15 features)*: Boolean and count features from our EventID knowledge base, acting as 'soft signatures' (e.g., IsFailedLogin, IsProcessCreation, IsSecurityLogCleared).
- iv. *Text (12 features)*: Features derived from log message content, including LogHasPowerShell, LogHasBase64, LogHasSuspicious, and LogEntropy. Log entropy is a powerful heuristic for detecting obfuscation, calculated using the Shannon entropy formula:

$$H(X) = - \sum_i p(x_i) \log_2 p(x_i)$$

where $p(x_i)$ is the probability of character x_i .

- v. *Statistical (9 features)*: Z-scores and windowed aggregates like EventID_ZScore, UniqueEventIDsInWindow, and FailedLoginRatio.

This high-dimensional feature space is the foundation upon which both the detection and subsequent explainability (Section 5) modules are built.

- b) *Log Storage and Analysis*: In the ATHEA framework, the 'Log Storage and Analysis' stage is represented by the in-memory, structured data frame created during preprocessing. This feature-rich data frame, which holds all 50+ engineered features, serves as the analytical base. All subsequent detection and XAI modules operate directly on this data structure, allowing for rapid, iterative analysis without the latency of database queries.
- c) *Threat Detection (AI/ML Layer)*: This module corresponds to the 'Threat Detection (AI/ML Layer)' block. To detect novel threats as defined in our objective, a purely unsupervised approach is required. The detection core employs a hybrid model utilizing three algorithms: Isolation Forest (IF), Local Outlier Factor (LOF), and a weighted ensemble of both. This hybrid model is a key design choice to prevent analytical blind spots, as APTs manifest in different ways.

Isolation Forest (IF) excels at finding global outliers—events that are "globally" rare and different from the entire dataset. It uses the anomaly score $s(x, n)$:

$$s(x, n) = 2^{-\frac{E(h(x))}{c(n)}}$$

where $E(h(x))$ is the average path length of instance x and $c(n)$ is a normalization constant. A score close to 1 indicates a clear anomaly.

Local Outlier Factor (LOF) is adept at finding density-based, local outliers—events that are anomalous only relative to their local neighbourhood (e.g., an action that is normal for the organization but highly abnormal for a specific user). The LOF score is a ratio of the local density of an instance to that of its neighbours:

$$\text{LOF}_k(A) = \frac{\sum_{B \in N_k(A)} \frac{\text{LRD}_k(B)}{\text{LRD}_k(A)}}{|N_k(A)|}$$

A score significantly greater than 1 indicates a local outlier.

A key practical innovation is the adaptive contamination parameter. The system auto-adjusts this parameter based on dataset size, ensuring high recall on small, pre-filtered log snippets (e.g., 30-50% for < 10 samples) while maintaining precision on large baseline datasets.

- d) *Explainable AI (XAI) Layer and Risk Scoring*: This module represents the 'Explainable AI (XAI Layer)' and 'Risk Scoring Algorithm' blocks from the diagram. It focuses on contextualizing and explaining the "what" and "why" of a detected anomaly to reduce alert fatigue. This is a multi-step process:

1. *Anomaly Contextualization (Clustering & Mapping)*: This step directly addresses the "alert fatigue" challenge [3]. A single attack (e.g., lateral movement) may generate hundreds of individual anomalous events. Instead of presenting 100 separate alerts, the system uses the HDBSCAN (Hierarchical Density-Based Clustering) algorithm. Unlike k -means, HDBSCAN does not require the user to specify the number of clusters k and can correctly identify "noise" points that belong to no cluster. This is ideal for our use case, as we expect most data to be noise, with only a few small, dense clusters of malicious activity. HDBSCAN groups related anomalies based on their feature-space proximity, coalescing them into a single, coherent "cluster" of activity. This allows the analyst to investigate the *attack pattern* as a whole. This contextual grouping, along with mapping to the

MITRE ATT&CK framework (as shown in Table I), is a critical preparatory step for the narrative explainability layer.

2. *XAI and Risk Scoring*: (Note: The full detail for this section is in Section 5, but it is introduced here as part of the architecture). Following detection and clustering, the dual-layer XAI framework is triggered. This involves:
 - *Layer 1 (SHAP)*: Provides feature-level contribution scores, explaining which of the 50+ features caused the model to flag the event.
 - *Layer 2 (GenAI)*: Synthesizes the SHAP scores, log data, and knowledge base context into a natural-language explanation.

The 'Risk Scoring Algorithm' is integrated here, where the initial EventID risk score is combined with contextual features (like IsNightTime) and the anomaly score itself to produce a final, prioritized alert for the 'Output / Analyst Interface'.

IV. THREAT INTELLIGENCE AND MITRE ATT&CK INTEGRATION

A core design principle of ATHEA is to enrich purely mathematical, ML-based detections with curated, domain-specific cybersecurity knowledge. This section details the "situational methods" [4], [8] that provide the critical context.

Stage	Tactics	Example EventIDs
Stage 1	Initial Access	4625 (Failed Login), 4648 (Explicit Creds)
Stage 2	Execution, Credential Access	4688 (Process), 1 (Sysmon), 4776 (NTLM)
Stage 3	Persistence, Priv. Escalation	4697 (Service), 4698 (Task), 4672 (Privileges)
Stage 4	Discovery, Lateral Movement	4624 (Login), 3 (Network), 5156 (Filtering)
Stage 5	Collection, C2	4663 (File Access), 11 (File Created)
Stage 6	Exfiltration	(Inferred from network/file patterns)
Stage 7	Impact	1102 (Log Clear), 4719 (Audit Change)

TABLE 1: MITRE ATT&CK STAGE MAPPING

An unsupervised model, on its own, can only identify that an event is statistically rare; it cannot determine if that event is malicious. This integration layer is what addresses the "benign anomaly" problem and provides the "so what" for the analyst, directly tackling the challenge of alert fatigue [3]. This module functions as the 'situational awareness' layer of the framework, translating the 'what' (a statistical anomaly) into the 'so what' (a prioritized, actionable threat).

A. EventID Intelligence and Contextual Risk Scoring

The system incorporates a built-in knowledge base for over 500 critical EventIDs from Windows Security, Sysmon,

System, and SQL Server logs. This engine is far more than a static lookup table; it is a dynamic risk-scoring and contextualization engine. It provides three key functions:

- a) *Baseline Risk Scoring*: A 1-10 risk score (R_{baseline}) is pre-assigned to each EventID, quantifying its typical security relevance. For example, a successful login (EventID 4624) has a low baseline risk ($R_{\text{baseline}} = 1$), while a security log clear (EventID 1102) has a very high one ($R_{\text{baseline}} = 9$).
- b) *Context-Aware Adjustment*: This is a critical innovation for reducing false positives. The baseline risk score is dynamically adjusted based on the context provided by our 50+ engineered features. This allows the system to distinguish between a benign anomaly and a malicious one. For example, a failed login (EventID 4625) is normally a low-risk event. However, if this same event occurs with `is_night_time = True` and the `failed_login_ratio` (from our statistical features) is high, the engine amplifies the risk score significantly, flagging it as a probable brute-force attack.

This dynamic adjustment (R_{final}) is based on a heuristic weighting of the ML model's anomaly score (S_{anomaly}) and the feature context (C_i):

$$R_{\text{final}} = w_0 R_{\text{baseline}} + w_1 S_{\text{anomaly}} + \sum_i w_i C_i$$

Where w_i are the weights assigned to contextual features

(C_i) like IsNightTime or LogHasPowerShell. This formula ensures that alerts are prioritized based on a holistic view of genuine threat indicators, not just statistical rarity.

- c) *MITRE Tactic Mapping*: The engine provides predefined, many-to-many links from specific EventIDs to their corresponding MITRE ATT&CK tactics (e.g., EventID 4688, Process Creation, maps to T1059 "Command and Scripting Interpreter" and T1053 "Scheduled Task/Job"). This provides the foundational data for the lifecycle mapping in the next section.

B. Automated Attack Lifecycle Mapping (MITRE ATT&CK)

This module directly addresses the "alert noise" problem. A SOC analyst is not interested in 100 individual, low-level anomalous events; they are interested in the story of the attack. The MITRE ATT&CK framework provides the grammar for this story.

While the full Enterprise ATT&CK matrix contains 14 tactics (e.g., Reconnaissance, Resource Development, Initial Access...), this complexity can itself be overwhelming for quick analysis.

To enhance analyst accessibility, ATHEA employs a deliberate abstraction: it maps detected anomalies and clusters (from the HDBSCAN module) to a simplified 7-stage attack lifecycle, as shown in Table I. This abstraction synthesizes the

14 tactics into a more intuitive narrative flow (e.g., 'Initial Access', 'Execution', 'Persistence').

The mapping is not just a 1-to-1 lookup. It synthesizes the output of the EventID engine (Section 4.1) and the HDBSCAN clustering (Section 3.2.4). This allows ATHEA to map a cluster of disparate, low-level events (e.g., a flurry of 4625 events followed by a single 4688) into a single, high-level narrative ('Stage 1: Initial Access' progressed to 'Stage 2: Execution').

This structured narrative is not just for the analyst; it is a critical input for the Layer 2 GenAI model (Section 5.2). By providing this pre-processed 'story' (e.g., "The attack began with Stage 1... followed by Stage 2..."), we enable the GenAI to generate its coherent, story-based "Timeline Analysis." This mapping transforms a collection of disparate, anomalous events into a coherent, prioritized, and understandable threat narrative, fulfilling a core objective of the ATHEA framework.

V. EXPLAINABLE AI (XAI) FRAMEWORK

The system's primary contribution, and the core of its novelty, is its dual-layer explainability framework. As established in the literature review, the "black box" nature of ML models is a critical adoption barrier for SOC analysts, who are already inundated with unactionable, low-context alerts [3], [11]. A tool that merely flags an event as "95% anomalous" fails the analyst. This XAI framework is explicitly designed to bridge this gap between a raw, mathematical anomaly score (the "what") and a human-understandable, actionable threat narrative (the "so what"). It achieves this by synthesizing quantitative, model-grounded attributions (Layer 1) with context-rich generative narrative analysis (Layer 2).

A. Layer 1: SHAP for Quantitative Feature-Level Explainability

The first layer of the XAI framework answers the question: "Why did the model find this event suspicious?" To do this, we employ SHAP (SHapley Additive exPlanations), a method from cooperative game theory that computes the optimal credit-allocation (the "contribution") of each feature to the final anomaly score [12]. We chose SHAP for its two key properties: it provides local interpretability (explaining each individual prediction, which is essential for log analysis) and it is model-agnostic (allowing it to work seamlessly with our hybrid IF/LOF ensemble).

SHAP explains a prediction by computing an additive feature attribution model, where the explanation g for a prediction $f(x)$ is given by:

$$g(z') = \phi_0 + \sum_{i=1}^M \phi_i z'_i$$

where $g(z')$ is the explanation for the simplified input z' , ϕ_0 is the base value (the average anomaly score across all events), M is the number of features (50+ in our case), and ϕ_i is the SHAP value (the contribution) for feature i .

The system computes the SHAP values for each of the 50+ features for every detected anomaly. This provides a quantitative, auditable "fingerprint" of the model's decision. For example, a high-scoring anomaly (e.g., score: 0.95) might be explained by:

- IsNightTime = True: High positive contribution ($\phi = +0.40$). Drove the score up significantly.
- EventIDRarity = 0.001: High positive contribution ($\phi = +0.35$).
- LogHasPowerShell = True: Medium positive contribution ($\phi = +0.15$).
- LogEntropy = 7.8: Medium positive contribution ($\phi = +0.10$).
- EventID = 4688: Low negative contribution ($\phi = -0.05$). This feature, in isolation, slightly decreased the anomaly score, but was overpowered by the others.

This output is powerful. It allows an analyst to see, at a glance, that the event was flagged because it was a rare, high-entropy PowerShell command executed at night. However, this quantitative output, while transparent, is still not an answer. It still requires a trained analyst to manually synthesize these features into a security concept. This limitation—the gap between explaining the math and explaining the threat—is the primary motivation for Layer 2.

B. Layer 2: GenAI for Narrative-Based Threat Analysis

While SHAP explains the model's decision, it does not explain the threat in a security context. The second XAI layer is designed to automate this crucial last-mile analytical synthesis. This module leverages a Large Language Model (Google Gemini 2.5 Flash) to translate the quantitative data from the entire pipeline into a human-readable threat analysis.

The key to this layer's success is its sophisticated, multipart context-driven prompt. The LLM is not asked a generic question; it is fed a rich, composite prompt that assembles all available evidence related to the anomaly. This prompt functions as a form of "Retrieval-Augmented Generation" (RAG) for security, providing all the real-time context the model needs:

- The Raw Log Data*: The full, original event message.
- The SHAP Explanation*: The quantitative feature contributions from Layer 1 (e.g., "Feature IsNightTime had a high positive impact...").
- The Knowledge Base Context*: The EventID risk score, description, and dynamic risk adjustments (from Section 4.1).
- The MITRE ATT&CK Mapping*: The associated tactic and stage (e.g., "This maps to Stage 2: Execution") (from Section 4.2).

The Gemini model is then prompted to act as a senior SOC analyst and synthesize all four data points into a coherent, narrative threat report. This use of a persona grounds the model's output, ensuring it uses the correct terminology and focuses on what is operationally relevant.

This Generative AI component provides four distinct, structured outputs, transforming the data into actionable intelligence:

- Per-Anomaly Analysis*: A detailed paragraph explaining the event, what the potential threat is, and why it is suspicious in plain English.
- Global Analysis*: A high-level, executive summary of the entire threat landscape discovered in the log file, identifying the most critical patterns or clusters.
- Timeline Analysis*: A story-based explanation of event sequences, using the HDBSCAN clusters and MITRE mapping to describe the attack's progression.
- Actionable Recommendations*: Concrete, suggested next steps for investigation and remediation.

This dual-layer approach is the core of ATHEA. The SHAP layer provides the quantitative, model-grounded truth, preventing the LLM from "hallucinating" or providing an explanation that is disconnected from the model's decision. The GenAI layer then provides the human-centric interpretation, translating SHAP's mathematical output into an actionable narrative. This synergy moves beyond a "black box" to provide a true, interpretable, and context-aware decision support system for the analyst.

VI. EVALUATION AND RESULTS

The ATHEA framework was evaluated on two primary axes, reflecting its core objectives:

- Detection Efficacy*: Its quantitative ability to accurately detect anomalous and malicious events (high precision and recall).
- Explainability*: Its qualitative ability to provide actionable, human-understandable explanations that reduce analyst workload (the XAI component).

The system was evaluated for its ability to identify a wide range of attack techniques using a composite, labelled dataset.

A. Evaluation Methodology and Datasets

To establish ground truth, a baseline dataset of benign Windows event logs (e.g., from the public Mordor or EVTX-Attack-Samples repositories) was used. Into this benign data, a series of known-malicious event sequences, simulating TTPs from the MITRE ATT&CK framework, were injected.

The evaluation was then run in an unsupervised manner, where the model had no access to the "malicious" or "benign" labels. The model's "anomaly" flags were then compared against the ground-truth labels to calculate performance metrics.

- True Positive (TP)*: A malicious event correctly flagged as an anomaly.
- False Positive (FP)*: A benign event incorrectly flagged as an anomaly.
- True Negative (TN)*: A benign event correctly ignored (not flagged).

- False Negative (FN)*: A malicious event incorrectly ignored (not flagged).

The core challenge, and the primary goal of ATHEA's design, is to maximize TPs while aggressively minimizing FPs.

B. Quantitative Performance: Detection Efficacy

The detection pipeline, when tuned, achieves strong performance metrics for unsupervised detection. The key metrics are defined as:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (\text{Measures alert-to-noise ratio})$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (\text{Measures threat coverage})$$

$$F1\text{-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Our internal metrics on the test dataset are:

- Precision*: ~85-95% (The high precision is a key result, indicating a low false-positive rate, which directly addresses the "alert fatigue" problem [3]).
- Recall*: ~80-90% (Indicating strong coverage of true anomalies and injected threats).
- F1-Score*: ~85% (A strong harmonic mean, showing a good balance between Precision and Recall).

This performance is highly adaptive; the system is optimized to handle small datasets (<100 events) by adjusting the contamination parameter for high recall, as well as large datasets (>1000 events) through batching and sampling for high precision.

These results are consistent with, and in some cases exceed, academic benchmarks for similar models. For instance, studies show that tree-based models, specifically Isolation Forest, can achieve detection accuracies as high as 92% in identifying malicious patterns [13]. Furthermore, AI-driven integration has been shown to improve overall threat detection by 15–30% over traditional signature-based methods [9] and reduce the incidence of false positives by approximately 20% [3]. ATHEA's high precision aligns with these established benchmarks, demonstrating the framework's viability for operational security environments.

C. Qualitative Performance: XAI Framework Evaluation

The most critical evaluation, however, is not just the detection score, but the actionability of the alert. This was evaluated qualitatively by comparing a traditional alert (a "black box" anomaly score) with the full ATHEA XAI output for the same event.

Standard "Black Box" Alert:

- 'Event 745: Anomaly Score = 0.98 (High)'

b) Analyst Action: This alert is unactionable. The analyst must manually investigate the event, cross-reference the EventID, parse the command, and independently determine why it is anomalous, if at all. This is the definition of alert fatigue.

ATHEA Dual-Layer XAI Alert:

a) Layer 1 (SHAP): 'Event 745 (Score: 0.98)'

1. 'High Positive Contribution: LogEntropy (8.1)'
2. 'High Positive Contribution: IsNightTime (True)'
3. 'Medium Positive Contribution: LogHasPowerShell (True)'

b) Layer 2 (GenAI Narrative):

1. Analysis: "This event is a high-priority alert. A PowerShell process was created at 03:15 AM, a time of low activity. The SHAP explanation shows the model flagged this due to extremely high text entropy (8.1), which strongly suggests a Base64-encoded or obfuscated command, a common technique for 'living-off-the-land' attacks."
2. MITRE Tactic: "Execution (T1059.001)"
3. Recommendation: "1. Immediately decode the PowerShell command to identify its payload. 2. Isolate the host 'Workstation-05' for forensic analysis."

This qualitative comparison demonstrates that the XAI framework successfully translates a low-context anomaly score into high-context, actionable intelligence, fulfilling the primary objective of the paper.

D. Detected Capabilities (TTP Case Studies)

The framework's 50+ features and hybrid (IF/LOF) model demonstrated the ability to detect a wide array of TTPs. Below are two examples of how the components work in synergy:

a) Case Study: Brute Force (EventID 4625)

1. Key Features: IsFailedLogin, FailedLoginRatio (windowed statistical feature).
2. Model: This TTP is best detected by Local Outlier Factor (LOF). A burst of failed logins from one user is a dense local cluster, which LOF excels at identifying as anomalous relative to that user's normal (zero-failure) behaviour. Isolation Forest (a global model) might miss this if failed logins are common across the entire organization.
3. XAI Output: The GenAI layer would state, "A high-density cluster of failed logins was detected, suggesting a brute-force attack from IP 10.1.1.5."

b) Case Study: Obfuscated PowerShell (EventID 4688)

1. Key Features: IsProcessCreation, LogHasPowerShell, LogEntropy, IsNightTime.
2. Model: This TTP is best detected by Isolation Forest (IF). A single, high-entropy PowerShell

command at 3 AM is not part of a dense cluster; it is a "globally" rare and unique event. It is easily "isolated" by the IF's tree structure, as its combination of features is unlike any benign event.

3. XAI Output: As shown in the 6.3 evaluation, the XAI would explain the exact risk (obfuscation, off-hours) and recommend decoding the command.

These cases, in addition to the other detected capabilities (Lateral Movement, Persistence, Defense Evasion, etc.), validate the hybrid-model approach and the power of context-aware feature engineering.

VII. DISCUSSION AND LIMITATIONS

A. Discussion

The results indicate that a hybrid, unsupervised approach enriched with XAI and domain knowledge can be highly effective for APT detection. The high precision (~85-95%) is particularly noteworthy, as false-positive reduction is a primary goal for any practical SOC tool, directly mitigating "alert fatigue" [3]. The dual-XAI framework (SHAP + GenAI) provides a significant advancement over traditional "black box" alerts by transforming raw anomaly scores into actionable, human-readable narratives. This synergy, as shown in the qualitative evaluation, enables junior analysts to understand and action complex threats, functioning as a "force multiplier" for the SOC.

B. Limitations

Despite its promising results, the framework has several practical limitations:

- a) *Parsing Failures*: Corrupted or non-standard EVTX files can fail to parse. This highlights the fragility of real-world log ingestion, requiring more resilient parsers.
- b) *Memory Usage*: Processing very large files (>1GB) is memory-intensive, particularly during SHAP value computation. This creates a trade-off between explainability depth and performance, limiting real-time application.
- c) *GenAI Dependency*: The most advanced narrative explanations are dependent on a valid Google Gemini API key and internet connectivity. This introduces potential issues of cost, latency, and data privacy if sensitive logs are sent to a third-party API.
- d) *Tuning*: While the system features "auto-contamination," optimal results for a specific environment may still require an analyst to tune this parameter. This is a common challenge for unsupervised models, as a compromised baseline can lead to missed detections (False Negatives).

VIII. CONCLUSION AND FUTURE WORK

This paper presented ATHEA, an end-to-end framework for unsupervised and explainable threat hunting in Windows Event

Logs. By systematically combining robust feature engineering, hybrid ML models, and a novel dual-layer XAI (SHAP + GenAI), the system successfully detects and explains complex threats while maintaining a low false-positive rate.

Future work will focus on optimizing the data processing pipeline for memory efficiency and speed, potentially moving to batch processing or sampling for SHAP analysis on large datasets. Expanding the EventID knowledge base and MITRE ATT&CK mappings will be a continuous effort. Finally, exploring the integration of a streaming data pipeline (e.g., via Kafka) could transition the system from an investigative, batch-processing tool to a real-time threat detection platform.

REFERENCES

- [1] S. R. Sindiramutty, (2024). Autonomous Threat Hunting: A Future Paradigm for AI-Driven Threat Intelligence.
- [2] K. P. Niveditha, N. Thakre, & N. Pateriya, (2024). A Comprehensive Review on Cyber Threat Intelligence using Machine Learning Techniques. *International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET)*, Vol. 13, No. 3, 2024..
- [3] O. O. Aramide, (2023). Predictive Analytics and Automated Threat Hunting: The Next Frontier in AI-Powered Cyber Defense. *International Journal of Technology Management & Humanities (IJTHM)*.
- [4] N. A. F. Shakil, R. Mia, & I. Ahmed, (2023). Applications of AI in Cyber Threat Hunting for Advanced Persistent Threats (APTs): Structured, Unstructured, and Situational Approaches. *International Journal of Science and Research Archive (IJSRA)*.
- [5] R. R. Kethireddy, (2022). AI-Driven Automated Threat Hunting with Predictive Analytics. *Journal of Recent Trends in Computer Science and Engineering (JRTCSE)*.
- [6] S. Nicole, & J. Michael, (2025). Leveraging Artificial Intelligence for Advanced Cybersecurity Threat Detection and Prevention.
- [7] H. Naveed, & S. Badi, (2021). Machine Learning for Advanced Threat Hunting in SOC Environments.
- [8] M. Cate, (2025). Building a Proactive Cyber Defense Model: Leveraging AI for Threat Hunting and Anomaly Detection in Zero Trust Architectures.
- [9] M. Danish, (2025). Enhancing Cyber Security Through Predictive Analytics: Real-Time Threat Detection and Response. *International Journal of Advanced Computer Science and Applications (IJACSA)*.
- [10] T. Osinaike, Y. Adekoya, and C. V. Onyenagubo, (2024). A Survey of AI-Powered Proactive Threat-Hunting Techniques: Challenges and Future Directions. *International Journal for Multidisciplinary Research (IJFMR)*.
- [11] E. Oye, & P. Peace, (2023). The Future of Cyber Defense: AI-Driven Threat Hunting for Organizations.
- [12] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," in *Advances in Neural Information Processing Systems*, 2017.
- [13] J. A. Pawar, M. S. Avhankar, A. Gupta, A. Barve, H. Patil, and R. Maranan, "Enhancing Network Security: Leveraging Isolation Forest for Malware Detection," *2024 2nd International Conference on Advancement in Computation & Computer Technologies (InCACCT)*, 2024.