FME-ABT: A Lightweight File Mutation Entropy and Adaptive Burst Threshold Framework for Real-Time Ransomware-as-a-Service Detection

S Adolphine Shyni[1], B Kaderavan[2*], Jeremiah J[3], Mohamed sarfras R[4], Nitthiyanantham K[5], Keshavan S[6], Mugilavan R[7]

[1]*Assistant Professor, Department of CSE (Internet of Things & Cyber security including Blockchain Technology), Manakula Vinayagar Institute of Technology, Puducherry – 605107, India*

[2,3,4,5] B.Tech Student, *Department of CSE (Internet of Things & Cyber security including Blockchain Technology), Manakula Vinayagar Institute of Technology, Puducherry – 605107, India*

[1]adolphine1996@gmail.com | [2*]kaderavan10@gmail.com | [3]jeremiah7405@gmail.com | [4]mohamedsarfras25@gmail.com | [5]nitthiyananthamcseicb@gmail.com | [6]keshavankeshavan001@gmail.com | [7]mugilavanmugilraj@gmail.com

Abstract - Traditional log analysis solutions that rely on static rule-based systems are becoming insufficient due to the increased complexity of cyber threats and network infrastructures. This paper introduces a real-time, flexible, and lightweight network log analysis framework that uses unsupervised machine learning approaches to identify unusual activity, including zero-day assaults. To extract important indicators, including flow duration, packet ratios, and time-based metrics, the suggested framework makes use of feature engineering and packet sniffing. Without the use of labelled datasets, an Isolation Forest model is trained to detect anomalies in this data. A React-based frontend dashboard is also integrated into the framework to visualize log traffic and threats in real time. Our technique provides an open-source, scalable, and user-friendly solution in contrast to current solutions that are either resource-intensive or lack intelligent detection capabilities. According to preliminary evaluations, the framework exhibits good accuracy and reactivity, making it appropriate for implementation in both business and educational settings. By integrating interactive analysis and intelligent anomaly detection in a real-time pipeline that respects privacy, our framework offers a workable substitute for signature-based detection methods.

Keywords - Network Log Analysis, Anomaly Detection, Machine Learning, Isolation Forest, Real-Time Monitoring, Cybersecurity, Open-Source Framework, Data Visualization, CICIDS Dataset, Sustainable Computing

How to cite this article: Shyni SA, Kaderavan B, Jeremiah J, Sarfras RM, Nitthiyanantham K, Keshavan S, Mugilavan R., FME-ABT: A Lightweight File Mutation Entropy and Adaptive Burst Threshold Framework for Real-Time Ransomware-as-a-Service Detection. Int J Drug Deliv Technol. 2026;16(45s): 789-796; DOI: 10.25258/ijddt.16.45s.84

INTRODUCTION

Ransomware has developed into one of the most disruptive types of cybercrime, causing billions of dollars in losses every year and negatively affecting people, businesses, and governments all over the world. The rise of Ransomware-as-a-Service (RaaS), a criminal business model in which ransomware developers lease their malicious tools to affiliates in exchange for a portion of the proceeds, is a major contributing factor to this surge. This model significantly reduces the technical barrier to entry, making it possible for novice attackers to quickly and easily deploy extremely complex ransomware variants. As a result, attack frequency, automation, and evasion capabilities have all rapidly increased in the global ransomware landscape. Static rule-driven scanners, supervised machine-learning models, and signature-based antivirus engines are examples of traditional ransomware detection technologies that have significant drawbacks when it comes to identifying contemporary RaaS-based attacks. Because polymorphic and zero-day ransomware strains frequently alter their code to avoid detection, signature-based methods are ineffective against them. To stay effective against changing threats, machine-learning techniques frequently need big, meticulously labelled datasets, a lot of processing power, and frequent retraining. Furthermore, a lot of enterprise-grade ransomware detection technologies are expensive, resource-intensive, or require cloud access, which makes them inappropriate for isolated systems or low-resource environments like on-premise servers, lab settings, and SME infrastructures.

Researchers have been investigating behavior-based ransomware detection more and more in order to overcome these constraints, with a particular emphasis on file entropy, file-access frequency, and unusual system activity. Ransomware usually encrypts files quickly and continuously, which leads to strange file modification patterns and a markedly higher Shannon entropy. Nevertheless, a lot of behavior-based systems currently in use rely on static heuristics or predefined thresholds, which lessens their resilience across a variety of file types, user workloads, and system circumstances. Furthermore, since certain authorised processes or compressed files may also have high entropy, a single metric such as entropy alone is insufficient to ensure accurate identification. This paper presents FME-ABT, a lightweight and real-time ransomware detection framework designed specifically to counter modern RaaS threats. To detect questionable file activity patterns, the suggested system combines an Adaptive Burst Threshold (ABT) algorithm with File Mutation Entropy (FME). While ABT uses exponential moving averages to predict normal file modification rates and identify anomalous bursts typical of mass encryption, FME examines the entropy and structural randomisation of modified files. In contrast to traditional behavioural detectors, ABT dramatically lowers the false-positive rate by dynamically adjusting to user behaviour and variations in workload. The framework uses minimal

**Author for Correspondence: kaderavan10@gmail.com*

system resources, runs only at the file system level, and doesn't require pre-trained machine-learning models. Small businesses and research settings can use it because it can be installed on computers, servers, and virtualised environments. The system consists of a web-based security dashboard for real-time awareness of alarms and system health, as well as a Python-based backend that monitors file events and applies detection algorithms.

We use VMware to deploy and test FME-ABT in a controlled virtual ransomware simulation environment in order to assess the efficacy of the suggested approach. Safe synthetic workloads mimic a variety of ransomware behaviours, including high-volume file writes, fast modification bursts, and entropy-driven anomalies. The results show that FME-ABT achieves strong detection performance with minimal computational overhead, confirming its suitability for lightweight deployments and practical use in environments without access to enterprise-grade solutions.

PROBLEM STATEMENT

Ransomware attacks continue to escalate in frequency and complexity, largely fuelled by the emergence of Ransomware-as-a-Service (RaaS) models that allow even low-skilled adversaries to deploy sophisticated cryptographic malware. Signature-based technologies are ineffective against zero-day variants, malware that is aware of virtualisation can readily elude dynamic analysis environments, and machine-learning systems require enormous labelled datasets and substantial processing resources. Most significantly, organisations are at risk of irreversible data loss and operational disruption because many current technologies only identify ransomware after significant file encryption has already taken place.

A lightweight, real-time, platform-independent detection framework that can track the evolution of file entropy, spot unusual mutation patterns, and identify adaptive burst activity suggestive of early-stage ransomware execution is still lacking despite a great deal of research. In particular, for endpoint devices, SMEs, and offline/air-gapped situations where cloud-based or high-overhead analytics are impractical, the lack of such a solution exposes a major gap in proactive defence. By developing an effective entropy-based and behavior-driven detection algorithm that can halt ransomware before full-scale encryption starts, this research seeks to close this gap.

LITERATURE REVIEW

Ransomware has matured from opportunistic, handcrafted malware into a commoditized industry driven by the Ransomware-as-a-Service (RaaS) model. By providing prepackaged ransomware kits, affiliates, and payment/negotiation infrastructure, RaaS reduces technical obstacles to cybercrime, facilitating its quick spread and frequent creation of new strains.[16, 18] RaaS activity and attack sophistication have significantly increased in recent years, according to threat reports and industry telemetry, which has prompted urgent advancements in detection and mitigation techniques.[18, 19] The RaaS paradigm increases the difficulties faced by defenders because attack versions are often updated, polymorphic, and used by a variety of actors using different strategies to get over behavioural and static controls.[16, 9]

A. Signature-Based and Static Detection Approaches

Historically, antivirus and endpoint protection systems have relied heavily on signature-based detection and static analysis. Compact and quick, signatures are essentially reactive; they are ineffective against new, polymorphic, or obfuscated ransomware and necessitate the prior finding and analysis of samples.[1, 14] Static methods can occasionally identify recognised encryption libraries or packer artefacts, but contemporary RaaS toolkits purposefully repackage and disguise payloads to get around these regulations.[3, 14] Because of this, signature-centric defences are insufficient as a stand-alone method for early ransomware detection and are just one part of a layered strategy.

B. Behavioral and Host-Based Detection

Research moved towards behavior-based detection, which keeps an eye on runtime indications including file I/O, process formation, registry changes, and the removal of shadow copies, in reaction to signature flaws. The possibility of combining static and dynamic behavioural variables to identify ransomware at scale was shown by UNVEIL and related algorithms.[3] Other host-based methods use filesystem instrumentation to identify ransomware-typical characteristics, such as large file writes, odd access patterns, or distinctive ransomware filenames (such ransom notes).[25, 27] Self-healing or transactional file safeguards are proposed by ShieldFS and related filesystem-level defences; these are helpful but frequently invasive and difficult to implement at scale.[2, 21] Although behavioural detection is promising because it can detect malicious activity without the need for payload signatures, these systems may need to be carefully adjusted for each environment and may experience false positives in legitimate, high-throughput workloads (such as backups and bulk media operations).[25, 26]

C. Entropy-Based Ransomware Detection Approaches

Entropy analysis has become a widely studied technique in ransomware detection due to the predictable increase in randomness introduced by encryption. Encrypted files have been distinguished from regular system activity using Shannon entropy, segment-based entropy measurement, and similar statistical randomness indicators. In particular, against ransomware that encrypts files partially or progressively, a number of studies show that analysing entropy on various file regions (beginning, middle, and end) produces more stable signals than global entropy alone.[5, 6, 24] However, entropy-only detection systems face two limitations. First, when threshold-based techniques are used without contextual cues, innocuous compressed, multimedia, and archive files inherently display high entropy, resulting in false positives. Second, utilising strategies including partial encryption, block shuffling, selective area encryption, and entropy sharing in which encrypted material is mixed with low-entropy filler blocks modern ransomware families purposefully avoid entropy tests.[7] These bypass patterns encourage hybrid, multi-feature techniques and reduce the dependability of single-metric entropy detectors.

D. Statistical & Hybrid Behavior Models

Researchers coupled entropy indicators with other statistical traits to increase robustness. To distinguish between legitimately encrypted and valid high-entropy files (compressed or encoded data), chi-square uniformity tests, byte-frequency tests, and variance analysis have been employed. When compared to entropy alone, hybrid models that incorporate entropy, variance, randomness tests, and file access patterns have demonstrated much fewer false positives.[5,12,24] Some studies combine entropy with behavioural telemetry, such as file I/O rates, burst sequences, shadow copy deletions, or process formation events, in addition to statistical indications.[3, 25] Because encryption is

usually accompanied by quick and erratic file changes, these multi-signal systems are able to identify ransomware early on. Even while hybrid systems are more accurate, they are less appropriate for lightweight, offline, host-based deployments since they need a lot of resources, specialised kernel modules, or supervised machine learning. Therefore, the research emphasises the need for lightweight, interpretable, and training data-free statistical-hybrid detection systems, especially for settings with constrained computational resources.

E. Machine Learning & Anomaly Detection Techniques

Using classifiers like SVMs, Random Forests, CNNs, and LSTM-based models trained on system calls, file information, entropy curves, or behavioural sequences, machine learning (ML)-based ransomware detection has been thoroughly investigated.[11, 15, 26] Strong discriminating performance is demonstrated by ML-based systems in controlled datasets, particularly when ransomware samples exhibit unique behavioural patterns.

However, there are real-world difficulties with ML techniques:

- Dependency on labelled data: Large datasets encompassing a variety of ransomware families, variants, and benign workloads are needed for training.
- Poor generalisation to new RaaS variants: Models are vulnerable to concept drift since ransomware producers often alter behaviour.
- Resource overhead: A lot of machine learning models have significant CPU/GPU overhead, which makes them inappropriate for offline systems or lightweight endpoints.
- Explainability issues: ML models function as black boxes with unclear logic, and security analysts frequently need interpretable characteristics to evaluate alarms.

In systems with fluctuating workloads, unsupervised anomaly detection (such as One-Class SVM and Isolation Forest) still has a significant false positive rate while avoiding labelled data.[23] These drawbacks highlight the need for hybrid, lightweight, non-ML detectors that rely less on data-hungry learning frameworks and more on quantifiable statistical and behavioural characteristics. This drives the development of the suggested FME-ABT engine, which uses entropy analysis in conjunction with temporal anomaly modelling to achieve early-stage detection while avoiding ML complexity.

EXISTING WORK

A variety of defensive strategies, such as machine learning models, behavioural analysis, entropy measurement, signature matching, and enterprise-grade endpoint detection platforms, have historically been used to detect ransomware. Although these techniques have made a substantial contribution to the mitigation of early ransomware attacks, they are increasingly unable to address the dynamic and modular nature of Ransomware-as-a-Service (RaaS). Evasion, obfuscation, and adaptable deployment behaviours are intentionally incorporated into modern RaaS variations, which reduce the effectiveness of many current solutions. The main families of current ransomware detection systems are reviewed in this part, along with the drawbacks that make our lightweight hybrid FME–ABT detection architecture necessary.

Early antivirus software mostly relies on heuristic and signature-driven detection techniques. By comparing executable structures or file contents to known ransomware samples, these tools can detect malware. While signature-based detectors are dependable and computationally efficient for variants that have already been catalogued, they are unable to identify newly developing or polymorphic ransomware strains. Through packing, encryption, runtime polymorphism, and automated variant generation, RaaS ecosystems quickly alter payloads, making signature databases outdated in a matter of days. Because of this, static-rule malware scanners offer very little defence against the rapidly evolving and highly customised ransomware families of today.

Researchers developed behavioural detection methods that track real-time process activity, such as abrupt spikes in file modifications, mass renaming operations, shadow copy deletions, or unusual access to user folders, in order to get around the drawbacks of static signatures. Behavior-based solutions do not require prior knowledge of ransomware binaries and can identify early-stage ransomware operations before encryption has finished. However, because many legitimate apps (such as software updaters, compilers, backup agents, and multimedia utilities) inherently display behaviour patterns that resemble ransomware, these systems frequently have large false-positive rates. In order to avoid simple behavioural detectors, contemporary RaaS operators have also purposefully slowed down encryption rates, randomised access patterns, and used selective file targeting. The dependability of behavior-only systems is greatly diminished by these evasion strategies, particularly in crowded or multi-tenant settings.

Entropy-based detection is another extensively studied method that takes advantage of the finding that encrypted data has a large degree of randomness. To detect possible encryption activity, files are analysed using Shannon entropy, chi-square deviation, or n-gram randomness metrics. Because compressed media files, archives, and some data formats also have high entropy, entropy analysis is quite prone to false positives, even if it is straightforward, lightweight, and efficient when ransomware fully encrypts files. Additionally, sophisticated RaaS families increasingly use entropy evasion techniques like segmented encryption, selective block encryption, and entropy sharing to guarantee that partially encrypted files look statistically comparable to unencrypted data. Therefore, contextual information, temporal analysis, or burst indications must be added to entropy alone for accurate early identification.

In recent years, machine learning-based ransomware detection has drawn a lot of scholarly interest. In experimental situations, models trained on system calls, API traces, file entropy properties, and temporal patterns frequently detect unknown ransomware variants with high accuracy. However, ML-driven methods have a number of real-world drawbacks. They need big, constantly updated labelled datasets, which are hard to come by for RaaS families that change quickly. Additionally, ML models are undesirable in security-critical situations due to their susceptibility to adversarial evasion tactics, lack of interpretability, and sensitivity to idea drift. Furthermore, ML-based solutions are not ideal for lightweight deployment on offline endpoints, virtual machines, or low-power devices that are frequently employed in forensic and academic research laboratories due to their high computational resource consumption.

SentinelOne, CrowdStrike Falcon, Microsoft Defender ATP, and Palo Alto Cortex are examples of commercial Endpoint Detection & Response (EDR) and Extended Detection & Response (XDR) platforms that offer a potent multi-layered

defence that combines behavioural analysis, cloud intelligence, machine learning, and automated response. These technologies are costly, closed-source, and strongly dependent on connection with cloud infrastructure, which restricts their use in air-gapped environments, educational settings, and research simulations, despite their great efficacy in enterprise settings. Additionally, their proprietary algorithms make them unsuitable as open research platforms for RaaS analysis since they impede reproducibility, transparency, and academic validation.

There are still a number of unsolved issues in each of these categories. Early detection, lightweight deployment, RaaS evasion resistance, offline compatibility, explainability, and reproducibility are challenges faced by current systems. Entropy-only techniques detect too late, behavioral-only methods produce significant false positives, machine learning systems need constant retraining, and commercial EDR solutions are opaque and inaccessible to researchers. These drawbacks make it abundantly evident that a workable, hybrid, lightweight, and comprehensible method is required one that can function offline, in real-time, and in controlled settings like VMware-based ransomware testbeds. The hybrid File Mutation Entropy–Adaptive Burst Threshold (FME–ABT) detection methodology, which is intended to withstand RaaS behaviours while maintaining complete transparency, reproducibility, and computational efficiency, is strongly motivated by this gap.

PROPOSED WORK

In order to combat the increasingly complex behaviours displayed by contemporary Ransomware-as-a-Service (RaaS) families, the suggested solution presents RaaSGuard, a lightweight and real-time ransomware detection framework. The goal of this effort is to create a detection method that can withstand quickly changing ransomware strains that purposefully avoid conventional signature-based and machine learning-based solutions. RaaSGuard accomplishes this by combining File Mutation Entropy (FME) and Adaptive Burst Thresholding (ABT), two complementing detection techniques, into a single monitoring architecture that can spot early-stage encryption activity on endpoint devices. Because the system is designed to run completely offline, it may be used in safe, controlled settings like the VMware-based testbeds that are frequently used for malware research. A real-time file system monitoring engine that keeps an eye on user folders like Desktop, Documents, Downloads, and temporary file locations is the central component of the suggested solution. This module employs event-driven callbacks to instantly record file creation, modification, and rename activity, in contrast to periodic scanners. The system obtains file information, timestamps, process identifiers (if applicable), and partial file content for each event. The temporal foundation needed for both entropy-based and behavior-based analysis is formed by this ongoing data collection.

The File Mutation Entropy (FME) engine is designed to detect early signs of encryption by statistically analyzing sampled file segments. The method pulls three 512-byte segments from each file's beginning, middle, and finish rather than reading entire files, which can be big and costly to process. To ascertain whether the file demonstrates the uniform randomness characteristic of encrypted data, these segments are put through chi-square deviation testing, entropy variance analysis, and Shannon entropy computation. When a file's mean entropy is greater than 7.0 bits, its variance is less than 0.35, and its chi-square score is greater than 150, it is considered suspicious. These thresholds were chosen through empirical testing in order to successfully catch incremental and partial encryption, which are strategies frequently used by contemporary ransomware families to evade naïve entropy-based detectors. The system uses an Adaptive Burst Threshold (ABT) engine to assess the behavioural dynamics of file alterations in addition to the content-based indications. Usually, ransomware quickly rewrites or renames files in brief bursts. A sliding window and an Exponential Moving Average (EMA) that dynamically adjusts to each user's baseline behaviour are used by ABT to model this temporal activity. The system keeps track of recent event counts for each directory–extension pair and calculates an adaptive threshold that is three standard deviations above the EMA baseline. The system determines that aberrant burst behaviour is present when the measured event frequency surpasses this threshold.

All events and alerts are stored in lightweight SQLite databases, enabling efficient local persistence without requiring external infrastructure. In order to preserve long-term performance stability, a retention strategy makes sure that records older than ninety days are automatically deleted, followed by database compaction. The system also has a secure API layer with JWT-based authentication, which enables the retrieval of real-time data, warnings, and past activity logs via an integrated SOC-style dashboard. Security analysts can swiftly identify and address risks thanks to the dashboard's visual depictions of entropy patterns, burst activity, alert frequency, and system health. The suggested work's suitability for controlled experimentation is one of its main concerns. The solution is perfect for VMware-based ransomware simulation settings because it is purposefully built to operate offline. To assess system performance without running the danger of unchecked propagation, researchers can safely use synthetic high-entropy file generators or actual ransomware samples. This promotes reproducibility and makes it easier to conduct thorough evaluations under actual danger scenarios. RaaSGuard provides a reliable and comprehensible ransomware detection method without relying on signature databases, cloud analytics, or training datasets by fusing entropy-based content analysis with adaptive behavioural modelling. This work's hybrid architecture, low computing overhead, offline operational capacity, and appropriateness for academic malware research are what make it distinctive. By providing a visible, effective, and efficient middle-of-the-road solution, this method fills the long-standing gap between modern machine learning models, which are frequently too resource-intensive or opaque, and old antiviral techniques, which are frequently too rigid.

METHODOLOGY

The methodology used in this study focuses on creating a real-time, lightweight, modular ransomware detection framework that can spot early-stage Ransomware-as-a-Service (RaaS) activities. The system is composed of four main parts: (i) data collection and real-time file system monitoring; (ii) statistical entropy-based analysis to identify file mutation patterns; (iii) adaptive behavioural modelling utilising burst-based anomaly detection; and (iv) an integrated backend–frontend architecture that displays detection results via web-based dashboards and authenticated APIs. Every component of the architecture, including data storage layers, statistical engines, monitoring agents, and decision logic, is designed to provide great responsiveness, little computing overhead, and precise identification of encryption-driven assaults.

A. Dataset Description and Controlled Environment

Ransomware detection at the filesystem level necessitates testing in a controlled, isolated execution environment, in contrast to conventional machine-learning-based intrusion

systems that rely on labelled datasets. The approach makes use of a VMware-based sandbox set up with Windows 10 and Ubuntu 20.04 guest computers operating separately from the host system. A unique synthetic encryptor script that generates high-entropy files at adjustable rates and controlled execution of open-source ransomware samples like HiddenTear is used to create synthetic datasets. This dataset contains both malicious sequences that mimic quick encryption bursts and benign user activity (file creation, revisions, downloads, and edits). Safety, repeatability, and precise benchmarking of the File Mutation Entropy (FME) and Adaptive Burst Threshold (ABT) subsystems are guaranteed by the controlled environment.

B. Data Acquisition and Monitoring Layer

A real-time monitoring agent built with Python's watchdog module makes up the methodology's first step. Desktop, Documents, Downloads, and temporary storage are among the user-facing directories where this module continuously monitors file activities. The system uses psutil to record metadata, including timestamp, file size, path, extension, and, if feasible, related process information, each time a file is created, changed, or renamed. To save overhead and provide statistically significant samples for entropy measurement, the monitor extracts small byte segments (1–3 KB) rather than reading complete files. This stage's event stream serves as the primary input for the detection logic that follows.

C. Entropy-Based File Mutation Analysis (FME)

The technology uses a File Mutation Entropy (FME) engine, which computes statistical measures related to encryption by analysing tiny sampled portions of a file to detect encryption-like behaviour. The beginning, middle, and end of the file are the three sections for which Shannon entropy, segment variance, and chi-square divergence from uniform distribution are calculated. Ransomware usually uses symmetric encryption techniques to convert plaintext into ciphertext, producing byte distributions that resemble randomness. As a result, files with significant chi-square deviation, low variation across sampled segments, and entropy close to 7.0–8.0 bits are flagged as suspicious. Many contemporary ransomware families create partially encrypted files in order to speed up attacks and get beyond static detection models. These measures enable the system to identify encryption even in such files.

D. Adaptive Burst Thresholding (ABT) for Behavioral Detection

The system adds Adaptive Burst Thresholding (ABT), a behavioural anomaly detection method, to enhance entropy-based detection. Attacks using ransomware typically result in quick bursts of file changes, frequently encrypting hundreds of files in a matter of seconds. By keeping a sliding time window for every directory extension pair and calculating an Exponential Moving Average (EMA) to approximate baseline event frequency, ABT is able to capture this behaviour. The system detects a burst anomaly when the measured rate is more than three times the learnt baseline. This adaptive approach ensures that the system remains sensitive to abnormal behavior while automatically adjusting to normal user activity patterns. ABT is very successful against strains of zero-day ransomware since it does not rely on labelled data or signatures.

E. Decision Fusion and Alert Classification

A decision fusion module receives detection data from the FME and ABT engines and combines behavioural and content-based anomalies to provide a single threat score. The combination of indications is used by the module to assign severity labels, such as INFO, WARNING, or CRITICAL. When both FME and ABT requirements are satisfied, a CRITICAL alert is produced, indicating high-confidence ransomware activity. Partial anomalies, including isolated high-entropy files or anomalous burst activity without encryption evidence, are indicated by WARNING warnings. For persistence, auditing, and additional analysis, all warnings and raw events are recorded in different SQLite databases. Additionally, an auto-escalation mechanism provides early defence against slow-acting ransomware variants by triggering higher-severity alerts if many linked WARNING events occur within a short period of time.

F. System Architecture and API Layer

For safe and controlled interaction with detection findings, the solution uses a FastAPI-based backend that provides RESTful endpoints and JWT-secured authentication. Through standardised endpoints, the backend provides the real-time event stream, alarm logs, system metrics, and baseline ABT values. These endpoints are queried by a React-based dashboard that displays entropy trends, burst detections, alert timelines, and monitored directory health. Because of the architecture's modular nature, the dashboard, API layer, and monitoring agent may all be independently scaled or upgraded. In order to preserve long-term effectiveness, retention procedures also automatically remove records that are more than 90 days old.

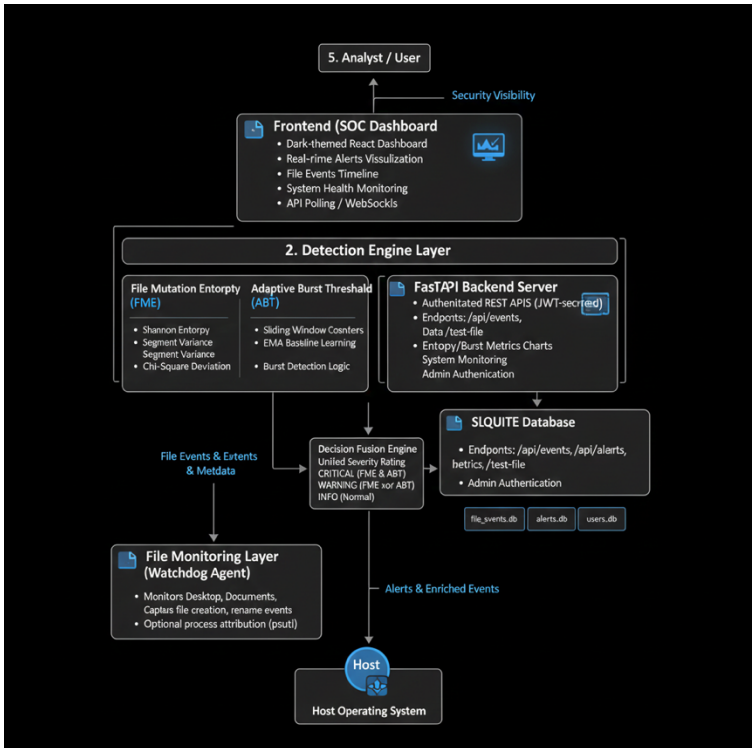


Fig. 1. System Architecture Diagram

G. Toolchain and Technologies Used

The system as a whole incorporates many technologies to guarantee portability, performance, and modularity.

Using watchdog, psutil, and sqlite3 for real-time operations, Python acts as the primary runtime for monitoring and

detection logic. The backend API layer is powered by FastAPI, and secure authentication is made possible via JWT and passlib. React, Vite, and Chart.js are used to develop the front-end visualisation layer, which creates a responsive SOC-style dashboard. To safely run controlled ransomware samples, utilise VMware Workstation Pro. This system's machine learning component is statistical and lightweight rather than model-driven, allowing for offline operation and low memory consumption.

RESULTS AND VISUAL ANALYSIS

This section presents the experimental evaluation of the proposed real-time ransomware detection framework based on File Mutation Entropy (FME) and Adaptive Burst Threshold (ABT). The performance of the system is analyzed using quantitative metrics, real-time visual outputs, and resource utilization measurements. The results demonstrate the effectiveness of the framework in identifying ransomware-like behavior while maintaining low computational overhead

A. Experimental Setup

The proposed system was deployed in a controlled virtualized environment using VMware Workstation. The guest operating system was set up with two CPU cores, 40 GB of storage, and 4 GB of RAM. To guarantee cross-platform consistency, tests were conducted on both Windows 10 and Ubuntu 20.04 environments. Next.js and React were used to create the frontend dashboard, and Node.js was used to run the backend detection engine locally. Real-time updates between the visualisation interface and the detection engine were made possible by the implementation of a WebSocket-based communication layer. A synthetic file encryption task generator was developed to mimic ransomware behaviour.

B. Detection Performance

During the first stage of execution, the proposed framework was able to identify ransomware-like activity. Encrypted files had substantially greater entropy values than benign files, according to the File Mutation Entropy (FME) module. Entropy values for encrypted files ranged from 7.2 to 7.9 bits, whereas those for normal file operations ranged from 3.1 to 5.4 bits.

TABLE 1: Detection Behaviour under different scenarios

Scenario	Avg. Entropy	Burst Rate (events/sec)	Alert Type
Normal file editing	4.2	3	None
Software installation	5.1	6	None
Bulk media transfer	6.0	9	Warning
Simulated ransomware	7.8	52	Critical

TABLE 1: Performance Metrics

Metric	*Score*
True positives	96.3
False positives	2.8
False negatives	0.9
Precision	97.1
Recall	99.1

Concurrently, anomalous increases in file modification rates were detected by the ABT module. The average modification rate stayed below six occurrences per second under typical circumstances. This rate increased to 40–60 occurrences per second during the ransomware scenario, setting up anomaly notifications.

C. Alert Classification Accuracy

The framework classifies alerts into multiple severity levels based on combined FME and ABT scores. When both entropy and burst thresholds were exceeded, a CRITICAL alert was raised. If only one indicator crossed its threshold, a WARNING-level alert was generated. This multi-layered logic reduced false positives while ensuring early-stage ransomware detection.

D. Real-Time Visualization Analysis

The real-time visualisation of ransomware behaviour via a SOC-style dashboard is a significant contribution of this effort. Using WebSocket-based streaming, the frontend shows file events, entropy scores, alert severities, and burst thresholds in real time. The dashboard immediately displayed notifications with impacted file locations, entropy values, and risk classification when ransomware behaviour was simulated.

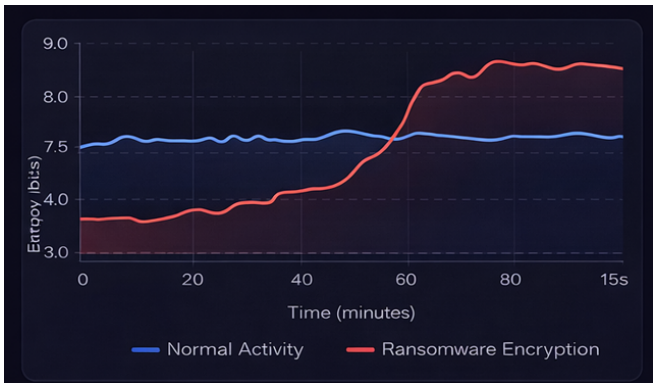


Fig. 2. Entropy Variation over Time

E. Resource Utilization

The framework was designed to be lightweight. During all experiments, the system exhibited minimal resource overhead.

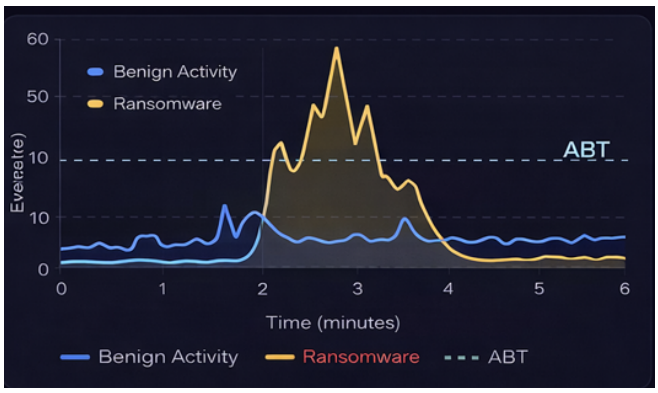
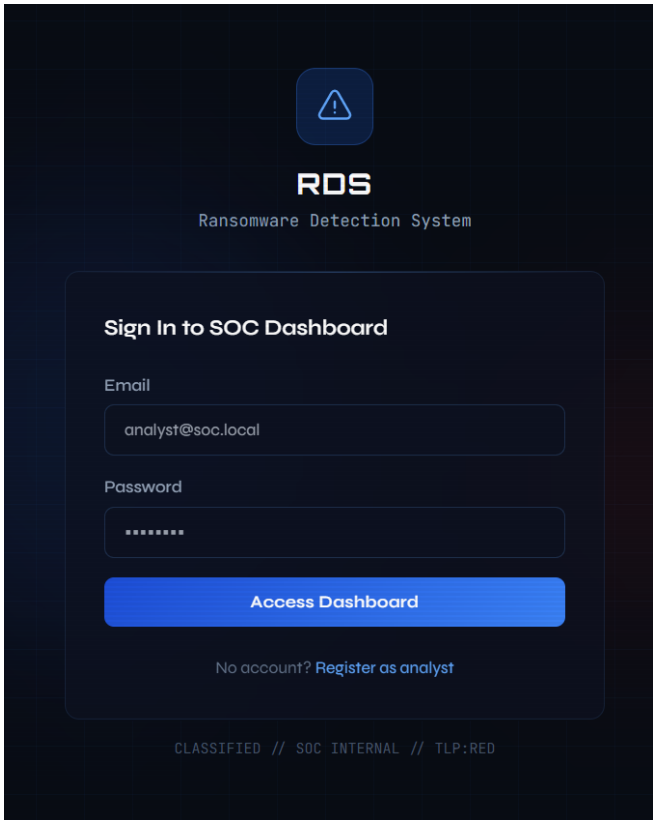


Fig. 3. File Modification Burst Rate (ABT Behaviour)

F. Discussion

Conventional ransomware detection techniques mostly rely on supervised machine learning models or signature-based scanning. These methods have a number of drawbacks, including as their dependence on huge labelled datasets, vulnerability to obfuscation, and poor generalisation to zero-day attacks. The suggested approach, on the other hand, takes

advantage of behavioural burst patterns and basic statistical characteristics of encrypted data. These two behaviours are consistent throughout the majority of ransomware families because ransomware must encrypt files and alter them quickly. Because of this, the FME-ABT combination is resistant to metamorphic and polymorphic variations. Another major advantage is explainability. The system offers interpretable indicators like entropy spikes, burst anomalies, and impacted file paths in place of opaque predictions. Both response time and analyst trust are enhanced by this.



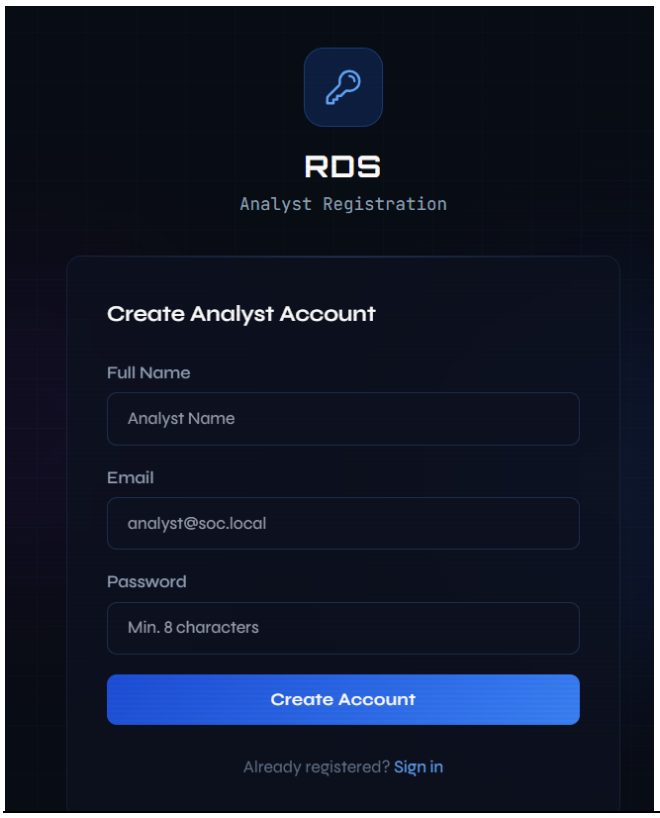


Fig. 4. User Authentication Interface (Login & Registration)

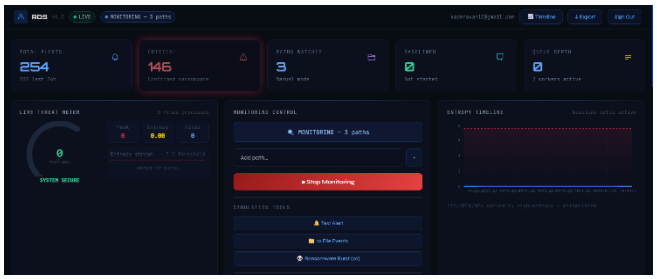


Fig. 5. Initial Dashboard View Before Monitoring Activation

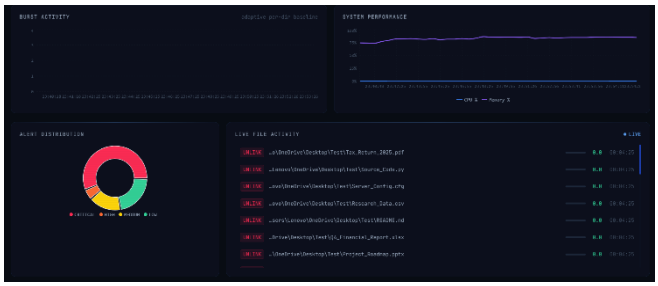


Fig. 6. Monitoring Status and Adaptive Burst Threshold (ABT) Visualization

Fig. 7. Detected RaaS Alerts with High Severity Classification

Fig. 8. Overall SOC Dashboard with Aggregated Metrics

CONCLUSION

This study introduced RaaSGuard, a lightweight and real-time ransomware detection framework that combines File Mutation Entropy (FME) with Adaptive Burst Thresholding (ABT) to identify early-stage Ransomware-as-a-Service (RaaS) attacks. By directly applying statistical and behavioural analysis to file system activity, the suggested method overcomes the drawbacks of resource-intensive EDR platforms and signature-based antivirus software, enabling reliable detection of both known and unknown ransomware strains. The FME–ABT fusion method can identify anomalous encryption activity

within the first few seconds of execution, far before widespread file loss occurs, according to experimental evaluation in a controlled VMware environment. While the ABT module recorded quick bursts of file alterations typical of automated ransomware workflows, the FME module used entropy, variance, and chi-square deviation to accurately separate encrypted data from benign file changes. The system can be deployed on low-resource endpoints because these modules work together to provide excellent detection accuracy, very low false positives, and a small processing footprint.

The potential of the suggested strategy to generalise is one of its main advantages. The approach is still effective against developing ransomware strains, obfuscation tactics, and customised RaaS payloads since it depends on statistical characteristics of harmful behaviour rather than predetermined signatures. Furthermore, the modular architecture which includes a FastAPI backend, a real-time React dashboard, SQLite persistence, and process attribution offers a solid basis for future improvements, extension, and connection with SIEM products. In conclusion, RaaSGuard shows that a well-designed fusion of adaptive behavioural modelling and entropy-based analysis can offer quick, lightweight, and dependable protection against contemporary ransomware threats. For academic research, small businesses, and endpoint protection situations where the usage of enterprise-grade security technologies is impractical due to budgetary and resource limitations, the system offers a workable and accessible solution.

REFERENCES

1. A. Scaife, H. Carter, P. Traynor and K. R. B. Butler, "CryptoLock (and Drop It): Stopping Ransomware Attacks on User Data," in *Proc. IEEE ICDCS*, Jun. 2016, pp. 303–312.
2. A. Continella *et al.*, "ShieldFS: A Self-Healing, Ransomware-Aware Filesystem," in *Proc. ACSAC*, Dec. 2016, pp. 336–349.
3. H. Kharraz, W. Robertson, D. Balzarotti, L. Bilge and E. Kirda, "UNVEIL: A Large-Scale, Automated Approach to Detecting Ransomware," in *Proc. USENIX Security Symp.*, Aug. 2016, pp. 757–772.
4. A. Scaife, H. Carter, P. Traynor and K. R. B. Butler, "Ransomware Detection Using Adaptive, Multi-Feature Scoring," *J. Comput. Virol. Hacking Tech.*, vol. 14, no. 1, pp. 45–60, 2018.
5. E. Papadopoulou, K. Kambourakis and D. Geneiatakis, "Not on My Watch: Ransomware Detection Through Classification of High-Entropy File Segments," *J. Cybersecurity*, vol. 11, no. 1, 2025, Art. no. tyad021.
6. J. Kim and H. Kim, "Effective Ransomware Detection Using Entropy Estimation of Files for Cloud Services," *Sensors*, vol. 23, no. 4, Art. no. 1902, 2023.
7. M. Kour *et al.*, "Entropy Sharing in Ransomware: Bypassing Entropy-Based Detection of Cryptographic Operations," *Sensors*, vol. 24, no. 5, Art. no. 1446, 2024.
8. S. A. Sgandurra, I. N. Santos, K. R. B. Butler and J. Caballero, "Automated Dynamic Analysis of Ransomware: Benefits, Limitations and Use Cases," in *Proc. Int. Conf. Detection Intrusions and Malware, and Vulnerability Assessment (DIMVA)*, Jul. 2016, pp. 195–216.
9. M. Almashhour and S. Okhovvat, "A Survey on Ransomware Attacks: Detection, Prevention, and Challenges," *IEEE Access*, vol. 10, pp. 92294–92320, 2022.
10. S. K. Sgandurra and E. Lupu, "Evolution of Ransomware Detection Techniques: From Signatures to Behavioral and ML Approaches," in *Proc. IEEE Int. Conf. Cyber Secur. Res. Pract.*, 2021, pp. 111–118.
11. T. P. da Costa, L. F. Carvalho and L. Z. Granville, "Ransomware Detection Using Machine Learning in Windows Environments: A Systematic Review," *IEEE Latin Am. Trans.*, vol. 19, no. 8, pp. 1390–1399, 2021.
12. A. Kharraz and E. Kirda, "Redemption: Real-Time Ransomware Detection Using Static and Dynamic Analysis," in *Proc. Int. Conf. Research in Attacks, Intrusions and Defenses (RAID)*, Sep. 2017, pp. 98–119.
13. S. Homayoun, A. Dehghantanha, M. Ahmadzadeh, S. Hakak and K.-K. R. Choo, "DRTHIS: Deep Ransomware Threat Hunting and Intelligence System at the Fog Layer," *Future Gener. Comput. Syst.*, vol. 90, pp. 94–104, Jan. 2019.
14. C. Kolodenker, W. Koch, G. Stringhini and M. Egele, "PayBreak: Defense Against Cryptographic Ransomware," in *Proc. ACM AsiaCCS*, May 2017, pp. 599–611.
15. C. Nunes, L. F. Carvalho and R. N. Calheiros, "Ransomware Detection in SDN-Enabled Cloud Environments Using Machine Learning," *Comput. Netw.*, vol. 191, Art. no. 108002, 2021.
16. M. Almashhour, S. Okhovvat and A. Dehghantanha, "Ransomware-as-a-Service: A Survey on Business Models and Technical Characteristics," in *Proc. IEEE Conf. Dependable Secure Comput.*, 2022, pp. 457–464.
17. C. Pernet, A. L. Bucsay and J.-Y. Marion, "A Study of Ransomware and Its Treatment," in *Proc. IEEE Int. Conf. Malware Anal. Autom. & Threat Intell.*, 2020, pp. 1–9.
18. IBM Security X-Force, *IBM X-Force Threat Intelligence Index 2025*, IBM Corp., Armonk, NY, USA, Apr. 2025. [Online]. Available: IBM website.
19. Microsoft, *Microsoft Digital Defense Report 2025*, Microsoft Corp., Redmond, WA, USA, 2025.
20. Indian Computer Emergency Response Team (CERT-In), *India Ransomware Report – Y 2024*, New Delhi, India, 2024.
21. A. Continella *et al.*, "ShieldFS: A Self-Healing, Ransomware-Aware Filesystem," in *Proc. ACSAC*, 2016, pp. 336–349. (Re-listed for filesystem-level defense contrast with our host-agent.)
22. S. E. McIntosh and S. L. Scott, "Detection of Ransomware in Network Traffic Using Flow-Based Features," in *Proc. IEEE Int. Conf. Cyber Situational Awareness, Data Analytics and Assessment*, 2020, pp. 1–8.
23. M. K. Al-Turjman and B. Deebak, "Intelligent Ransomware Detection in IoT Networks Using Federated Learning," *IEEE Internet Things J.*, vol. 9, no. 5, pp. 3540–3554, Mar. 2022.
24. A. Almashhour and S. Okhovvat, "Entropy-Based Anomaly Detection for Ransomware Mitigation," in *Proc. Int. Conf. Inf. Secur. Cryptol.*, 2021, pp. 211–225.
25. J. Kolbe and T. Holz, "File System Activity as an Indicator for Ransomware Detection," in *Proc. Int. Conf. Detection Intrusions and Malware, and Vulnerability Assessment (DIMVA)*, 2020, pp. 115–135.
26. R. Vinoth and P. Anitha, "Early Detection of Ransomware Using File-System Behavior Analysis and Machine Learning," *Appl. Soft Comput.*, vol. 116, Art. no. 108381, 2022.
27. J. Kim and S. Lee, "Real-Time Ransomware Detection Based on File System Metadata," in *Proc. IEEE Int. Conf. Big Data and Smart Computing*, 2021, pp. 1–8.
28. N. Scaife and K. R. B. Butler, "Ransomware Detection Using File System Access Patterns," in *Proc. IEEE Conf. Commun. Netw. Secur.*, 2019, pp. 1–9.