

AI-DRIVEN DIGITAL FORENSIC READINESS FRAMEWORK FOR RANSOMWARE DETECTION IN WINDOWS-BASED OPERATIONAL TECHNOLOGY (OT) SYSTEMS WITHIN CRITICAL ENERGY INFRASTRUCTURE



Submitted by: Rekha Pambala

Supervisor: Dipo Dunsin

Student ID: 2428137

Date of submission: 07/06/2026

Declaration

I, Rekha Pambala declare that I am the sole author of this Project; that all references cited have been consulted; that I have conducted all work of which this is a record, and that the finished work lies within the prescribed word limits. This work has not previously been accepted as part of any other degree submission.

Signed: Rekha Pambala

Date: 07/06/2026

Form Of Consent

I, Rekha Pambala hereby consent that my Project, submitted in candidature for the MSc Cybersecurity and Digital Forensics, if successful, may be made available for inter-library loan or photocopying (subject to the law of copyright), and that the title and abstract may be made available to outside organizations.

Signed: Rekha Pambala

Date: 07/06/2026

Abstract

As cybercrime against critical infrastructure continues to grow in terms of both frequency and sophistication, the need for enhanced threat detection and digital forensic readiness capabilities has been emphasized. Conventional security measures are mostly concerned about threat detection and offer little assistance in preserving evidence or conducting a forensic investigation. This research introduces an AI-Driven Digital Forensic Readiness Framework that brings together ransomware detection, Operational Technology (OT) anomaly detection, automated forensic logging, and real-time security monitoring all into a single platform.

The framework has been developed with machine learning and deep learning techniques. A set of Windows Portable Executable (PE) headers was used to test Logistic Regression, Gradient Boosting, and Random Forest algorithms to detect ransomware. The results of the experiments demonstrated that the highest performance was obtained by Random Forest with accuracy of 99.66%, precision of 99.63%, recall of 99.76%, and f1 score of 99.70%. The HAI industrial control system dataset was used to implement the models of Isolation Forest and Neural Network for OT anomaly detection. 8,316 and 5,677 were the number of anomalies found by the Neural Network and the Isolation Forest, respectively.

For improving the readiness for forensic activities, a MySQL-based repository was created that is able to automatically store evidence related to threats. A Flask based AI Cyber Forensic Intelligence Dashboard was also integrated to provide real time monitoring, threat analytics and evidence management. The results show successful integration of intelligent threat detection, anomaly detection, forensic evidence preservation and security monitoring. The research provides a viable method to advance the cyber resilience readiness and digital forensic preparedness in CIs.

Table of Contents

CHAPTER 1: INTRODUCTION	10
1.1 Background:	10
1.2 Problem Statement:	11
1.3 Aim:	12
1.4 Objectives:	12
1.5 Research Questions:	13
CHAPTER 2: LITERATURE REVIEW.....	14
2.1 Ransomware Evolution: Opportunistic Attacks to Strategic Cyber Campaigns: 14	
2.2 Digital Forensic Preparedness: Beyond Response to Prevention:	15
2.3 Operational Technology (OT) Security and Industrial Anomaly Detection:	16
2.4 Windows Infrastructure and Ransomware Behaviour Modelling:	18
2.5 Fragmentation and Identified Research Gap:	19
2.6 Positioning of the Current Study:.....	21
CHAPTER 3: METHODOLOGY	26
3.1 Introduction:	26
3.2 Research Design:	27
3.2.1 Research Philosophy and Approach:	27
3.2.2 Proposed Framework Architecture:.....	27
3.2.3 Justification of Framework	29
3.3 Models and Algorithms:	29
3.3.1 Ransomware Detection Models:	30
3.3.2 OT Anomaly Detection Models:	32
3.4 Tools and Technologies:	33
3.5 Datasets:	34
3.6 Implementation Plan:	35
3.7 Ethical Considerations.....	36
3.8 Summary:	37
CHAPTER 4: IMPLEMENTATION AND RESULTS.....	38
4.1 Introduction:	38
4.2 Ransomware Detection Implementation and Results:	38
4.2.1 Dataset Exploration and Initial Analysis:	38

4.2.2 Data Preprocessing:	41
4.2.3 Feature Selection and Importance Analysis:	42
4.2.4 Hyperparameter Optimisation:	44
4.2.5 Model Training and Development:	44
4.2.6 Performance Evaluation:	46
4.2.7 ROC Curve Analysis:	47
4.2.8 Discussion of Ransomware Detection Results:	48
4.3 OT Anomaly Detection Implementation and Results:	49
4.3.1 Dataset Preparation:	49
4.3.2 Data Preprocessing:	50
4.3.3 Isolation Forest Implementation and Results:	51
4.3.4 Neural Network Implementation and Results:	54
4.3.5 Comparative Analysis & Discussion:	56
4.4 Digital Forensic Logging System Implementation:	58
4.4.1 Forensic Evidence Repository:	58
4.4.2 Automated Evidence Logging Process:	59
4.4.3 Contribution to Digital Forensic Readiness:	60
4.5 Flask Dashboard Implementation:	61
4.5.1 AI-Driven Cyber Forensic Intelligence Dashboard:	61
4.5.2 Ransomware Detection Module:	62
4.5.3 OT Anomaly Detection Module:	63
4.5.4 Live Threat Analytics:	63
4.5.5 Dashboard Functionality Assessment:	64
4.6 Evaluation Against Research Objectives:	65
4.7 Discussion of Findings:	65
4.8 Summary:	66
CHAPTER 5: CONCLUSION AND FUTURE WORK	67
5.1 Introduction:	67
5.2 Summary of Research Findings:	67
5.3 Achievement of Research Objectives:	68
5.4 Research Contributions:	68
5.5 Limitations of the Study:	69
5.6 Recommendations for Future Work:	69
5.7 Conclusion:	69
REFERENCES:	71

LIST OF TABLES

Table 1: Comparative Analysis of AI & ML Techniques in Digital Forensics for OT Ransomware Detection	24
Table 2: Comparison of Ransomware Detection Models	32
Table 3: Tools and Technologies Used	34
Table 4: Dataset Summary	35
Table 5: Top Features Selected for Ransomware Detection.....	43
Table 6: Optimal Random Forest Hyperparameters.....	44
Table 7: Performance Comparison of Ransomware Detection Models.....	46
Table 8: HAI Dataset Overview	50
Table 9: Isolation Forest Detection Results	52
Table 10: Neural Network Detection Results	55
Table 11: Comparison of OT Anomaly Detection Models	56
Table 12: Evaluation of Research Objectives	65

LIST OF FIGURES

Figure 1: Proposed AI-Driven Digital Forensic Readiness Framework	28
Figure 2: Class Distribution of Benign and Ransomware Samples.....	39
Figure 3: Correlation Heatmap of PE Header Features	40
Figure 4: Ransomware Data Preprocessing Workflow.....	42
Figure 5: Feature Importance Analysis.....	43
Figure 6: Comparison of Machine Learning Model Accuracy	47
Figure 7: Receiver Operating Characteristic (ROC) Curve	48
Figure 8: OT Data Preprocessing Workflow	51
Figure 9: Detected Anomalies Over Time.....	53
Figure 10: Isolation Forest Anomaly Scores	54
Figure 11: Neural Network Detected Anomalies	56
Figure 12: OT Model Comparison.....	57
Figure 13: MySQL Forensic Evidence Repository	59
Figure 14: Automated Forensic Logging Workflow	60
Figure 15: AI-Driven Cyber Forensic Intelligence Dashboard.....	61
Figure 16: Ransomware Detection Module	62
Figure 17: OT Anomaly Detection Module.....	63
Figure 18: Live Threat Analytics Interface	64

TABLE OF ABBREVIATIONS

Abbreviation	Full Form
AI	Artificial Intelligence
AUC	Area Under the Curve
CSS	Cascading Style Sheets
CSV	Comma-Separated Values
DSR	Design Science Research
F1-Score	Harmonic Mean of Precision and Recall
FN	False Negative
FP	False Positive
GB	Gradient Boosting
HAI	HIL-Based Augmented Industrial Control System
HTML	HyperText Markup Language
ICS	Industrial Control System
IoT	Internet of Things
IT	Information Technology
JS	JavaScript
ML	Machine Learning
MLP	Multi-Layer Perceptron
MySQL	My Structured Query Language
OT	Operational Technology
PE	Portable Executable
RF	Random Forest
ROC	Receiver Operating Characteristic
SOC	Security Operations Center
SQL	Structured Query Language
TN	True Negative
TP	True Positive
UI	User Interface
URL	Uniform Resource Locator

VS	Versus
XAI	Explainable Artificial Intelligence
Abbreviation	Full Form
API	Application Programming Interface
GBM	Gradient Boosting Machine
IDS	Intrusion Detection System
IPS	Intrusion Prevention System
LSTM	Long Short-Term Memory
MSE	Mean Squared Error
NN	Neural Network
PE Header	Portable Executable Header
ReLU	Rectified Linear Unit
REST	Representational State Transfer
SOC Dashboard	Security Operations Center Dashboard
UI	User Interface

CHAPTER 1: INTRODUCTION

1.1 Background:

In the modern digital world, which is changing fast, critical infrastructure especially in energy sector has changed drastically. Historically, Operational Technology (OT) systems were in isolation, and they were only thought of in terms of the control of physical processes, including power generation, transmission and distribution. Nevertheless, as digitalization is on the increase, such systems are becoming increasingly connected to Information Technology (IT) to provide more intelligent monitoring, automation and decision-making that is based on data.

Although this merging of IT and OT is undisputedly beneficial (in efficiency and operational intelligence) it also leads to a wider, and more difficult-to-defend, cyber-attack surface. Systems which were formerly secured by physical isolation have been exposed to network-based threats and thus are now easy targets by cybercriminals. Ransomware is one of the most disruptive and financially harmful cyberattacks of these threats.

The ransomware attacks in modern times cannot be considered simple. Attackers do not encrypt files at once but take a multi-step strategy, which involves initial access, reconnaissance of the system, subsequent network traversal and eventual implementation of encryption measures. Before initiating the attack, attackers also steal sensitive data in most instances putting extra pressure on the victims. Such change in attack strategy renders classic signature-based and reactive detection systems to be inadequate.

Artificial intelligence and machine learning methods have become a popular approach by researchers and practitioners to overcome these challenges. These methods can help detect the intricate behaviour patterns and detect any anomalies that could be the evidence of malicious activity. In the case of machine learning, Chen (2025) shows that subtle clues to malicious behaviour can be easily represented by machine learning models, which is a more proactive threat detection measure. Likewise, Yang (2026)

reminds of the increasing significance of AI in cybersecurity, noting its capacity to evolve to the new threats and enhance the accuracy of detection with time.

But in the Operational Technology setting, the use of these techniques can be even more complicated. OT systems are built with high reliability, availability and safety requirements. Any type of interruption or latency in such systems may have severe real-life effects, such as damage to equipment or service failures. Consequently, the security solutions within the realm of OT must be lightweight, efficient, and least intrusive. Wang et al. (2024) underline that the direct implementation of IT-based detection models to OT systems frequently causes the high false-positive rate and low effectiveness, and the domain-specific methods should be applied.

Besides detection, other important elements of cybersecurity that have become an issue of concern is digital forensic preparedness. Conventionally, digital forensics has been a reactive procedure, and this is initiated once an incidence has taken place. But, when the environment is complex, as in critical infrastructure, this is no longer adequate. The systems must be able to actively gather, retain and control evidence as a part of the routine of organizations. Garg et al. (2025) believe that combining AI-based detection with forensic preparedness could contribute to the better incident response and post-incident investigation.

Despite these developments, most of the existing solutions continue to consider detection and forensic operations as independent of each other. Detection systems aim to detect threats whereas forensic systems are intended to analyze the incident after it has taken place. This division hinders the effectiveness of the two forms and generates loopholes in the security system at large.

1.2 Problem Statement:

Despite the tremendous advancement in the domain of ransomware detection and anomaly detection, the existing solutions are still disjointed and cannot be used in real-life scenarios in critical infrastructural settings. Most of the existing literature aims at ransomware detection in IT systems, or anomalies in OT systems and lacks the discussion of the necessity of a combined method.

This disintegration leads to several major issues. First, detection systems are usually deployed in isolation and do not offer a global picture of threats in the systems both in

the IT and OT environment. Second, a significant number of detection models are more focused on the accuracy and do not pay enough attention to the need to preserve the evidence that can be used during the forensic investigation and adherence to the law. Third, forensic systems are usually reactive in nature, i.e. valuable evidence could be lost or destroyed before an investigation is even undertaken.

Moreover, the OT settings add further complexities because of its real-time nature of operation and its safety-related nature. Security systems should be developed cautiously not to interfere with the working of the system, and this restricts the relevance of conventional IT based solutions. This means that organizations are struggling to attain effective threat detection, as well as sound forensic preparedness.

The shortcomings of these constraints indicate an acute gap in current research and practice: the absence of a unifying framework bridging AI-assisted detection, as well as well-structured forensic preparedness of Windows-based OT systems. In the absence of this type of integration, organizations are also susceptible to advanced ransomware attacks and lack the tools to effectively react to such attacks.

1.3 Aim:

To design, develop and test an AI-based Cyber Forensic Intelligence Platform that combines ransomware detection, OT anomaly detection, automated logging of forensic evidence and real-time threat monitoring, improving the digital forensic readiness in Windows-based OT environments for critical energy infrastructure.

1.4 Objectives:

In this regard, the project follows the following objectives to attain this goal:

- To characterize ransomware behaviour with a dataset based on Windows and create a machine learning model that can identify malicious and benign activity.
- To use anomaly detectors to the industrial OT data, i.e., the HAI dataset, to detect an abnormal system behaviour.
- To combine ransomware detection and OT anomaly detection into a single system that offers an in-depth view of threats.
- To create and deploy a MySQL-based forensic logging system to organize storage of events, timestamps and risk indicators.

- To create a real-time visualization dashboard with the help of Flask to display system traffic, threat levels, and forensic logs in an easily readable format.

1.5 Research Questions:

The proposed research will provide the answer to the following research questions:

- How well can machine learning models be used to recognize ransomware behaviour in Windows environments?
- To what extent are anomaly detection methods effective at detecting abnormal behavior in OT systems?
- What is the way to introduce AI-based detection systems with digital forensic preparedness to the response and investigation of critical infrastructure?

CHAPTER 2: LITERATURE REVIEW

2.1 Ransomware Evolution: Opportunistic Attacks to Strategic Cyber Campaigns:

Ransomware is in numerous aspects a story of adaptation. What started as a rather unsophisticated malware to encrypt the user files has now turned out to be one of the most strategic and coordinated cyber threats. In its simplest model, ransomware worked on a straightforward concept that is to infect a system, encrypt files and require money. Such attacks were haphazard in nature and not thought through.

But, with the development of cybersecurity defenses, attackers evolved. Ransomware today is not dependent on real-time execution. Rather, it has a hierarchical, lifecycle. The first step of attackers is obtaining initial access, which can be via phishing emails or vulnerabilities in the system. When they gain access, they thoroughly search the network, find valuable resources and build up their control through lateral movement and privilege escalation.

This development is especially important due to the change in detection nature. The use of traditional signature-based systems, which rely on known patterns, can hardly detect these advanced and even new threats. As you mention in your literature, nowadays ransomware often operates in a pre-encryption stage, being undetected as it prepares to a significant scale of attack (Lee et al., 2022).

In the initial studies, an effort to combat ransomware was made by applying the techniques of a static analysis. Some of the models that included the analysis of file structures and especially the Portable Executable (PE) header, included the CNN-based approach by Manavi and Hamzeh (2020). Although these techniques had been showing promising results in controlled settings, they were limited in nature. The detection could be easily defeated by minor changes in the structure of the code or employing obfuscation.

Equally, entropy-based techniques were used to identify abnormal encryptions. These methods were conceptually sound, but failed in practice, with high false-positive rates. The legitimacy of the encryption processes would usually raise alarms which diminished their reliability.

Through the understanding of these difficulties, scientists directed their attention to behavioural detection. The behavioural models track system activities over time rather than emphasizing on the attributes of the system which remain constant. Urooj et al. (2021) proposed a pre-encryption detection model, which detects suspicious patterns prior to encryption, which significantly changes the detection approach towards proactive.

Enhanced developments were made in the form of ensemble learning. Albshaier et al. (2024) have shown that their models, including Random Forest and Gradient Boosting can reproduce various behavioural patterns and, therefore, are more robust against various ransomware families. The models offer a compromise between generalization and accuracy and thus they can be used in real world applications.

This evolution has also been aided by deep learning approaches. CNNs and Recurrent Neural Networks (RNNs) were pointed out by Kritika (2025) as the methods of modeling sequential behaviours. These are especially useful in modeling complicated temporal relationships. Nevertheless, they bring about new problems such as computational complexity and overfitting.

Even though this was done, there is one crucial limitation. Most of the detection models are trained with enterprise IT data, as pointed out by Mondal et al. (2025). This poses issues on their applicability in Operational Technology environments where the behaviour of the systems is different.

2.2 Digital Forensic Preparedness: Beyond Response to Prevention:

The identification of ransomware is important but not all the cybersecurity issue. It is also crucial to be able to comprehend what has occurred, retain evidence, aid investigation procedures. It is out of this requirement that Digital Forensic Readiness (DFR) has been created.

Digital forensics has traditionally been reactive. The investigations commence once an incident is noted and, in most cases, there is a lot of damage that has been done. Such a method has a few drawbacks, especially in the case of ransomware when attackers can actively seek to destroy or alter the evidence.

To solve the problems, Pilli et al. (2010) proposed the notion of forensic preparedness that focuses on evidence proactivity in gathering. Systems are made in such a way

that they constantly gather and archive pertinent data instead of waiting until an incident occurs. This will make sure that in case of an attack, the investigators will have access to credible evidence.

When it comes to ransomware, it is essential to gather evidence in time. According to the CONIT (2024) study, it is vital to capture volatile memory, system logs, and process artefacts. These elements give good insights into attack behaviour. Nevertheless, the study also discloses a major limitation as such evidence is usually only gathered once an attack has been detected.

To address this Mekala (2025) suggested lightweight forensic systems in the environment of the Industrial Internet of Things (IIoT). These frameworks seek to balance between evidence gathering and system operation to make sure that logging does not disrupt real time operations. It is especially crucial in OT systems, where there is too much logging, which can interfere with important processes.

Zawoad and Hasan (2015) expanded on the idea with the introduction of distributed forensic systems to IoT systems. Although such methods enhance the preservation of evidence, they are not combined with smart detection tools.

A more recent study by Garg et al. (2025) tries to close this gap and merge AI-based detection with forensic preparedness. This will allow automated evidence collection on detection events. Nevertheless, they are mostly oriented on enterprise settings and do not consider the specific needs of OT systems comprehensively.

Overall, although the concept of forensic preparedness has transformed, it has not yet been combined with real-time detection mechanisms.

2.3 Operational Technology (OT) Security and Industrial Anomaly Detection:

The environment of operational Technology systems is completely different than the traditional IT environment, and it is imperative to comprehend this fact prior to the discussion of security solutions. In contrast to IT systems, which choose to emphasize the processing of data and flexibility, OT systems are designed to be stable, predictable and operating continuously. Such systems are used in physical processes like power production, manufacturing lines and industrial automation where any slight failure can result in severe safety issues or financial damages.

Due to this fact, cybersecurity in OT cannot be easily duplicated using IT methods. According to Cosnek, Lutkus and Barragan, (2024), OT systems put availability and deterministic behaviour as a priority over other factors. This implies that any detection mechanism must work to the lowest latency and it cannot interfere with the normal operations of the system. Practically, this poses a fine line between security and reliability of operation.

Anomaly detection has proved to be one of the most appropriate methods in this limited space. Rather than using a set of pre-defined attack signatures, anomaly detection models attempt to be trained on what normal behaviour resembles and then detects abnormal behaviour. This comes in handy especially in OT systems where the processes tend to be repetitive and predictable.

Wang et al. (2024) have shown that lightweight anomaly detection models can effectively detect abnormal patterns in industrial settings. Their work demonstrates that anomalies in system behaviour like atypical sensor data or unanticipated actuator conditions- can serve as a warning of malignant activity. Nevertheless, their results also reveal a severe problem, models that are trained and tested on IT datasets do not work well in OT systems. High false-positive rates are the result of this mismatch, and they may overload operators and decrease confidence in the system.

To overcome this drawback, time-series modelling methods have gained more popularity as a method of research. The hierarchical clustering techniques proposed by Pekepok et al. (2025) have the potential to identify subtle sequential patterns in system behaviour. As OT data is time-dependent in nature, these methods give a better representation of system dynamics.

The other crucial feature is feature engineering. Mohamed (2025) underlines that domain-specific features that depict operational semantics are needed in OT environments. An example is sensor thresholds, actuator reactions, and control loop patterns, which are much more valuable than generic network metrics. In the absence of these domain-specific features, detection models can never be able to capture the actual nature of system behaviour.

These developments notwithstanding, there are still significant challenges. Afaq, Rahmani and Sikora (2025) note that there are no high-quality publicly available OT datasets. Most researches are based on simulated or enterprise-based data, which is

not a complete depiction of real industrial settings. This restricts the applicability of models suggested.

Moreover, although anomaly detection models are useful when it comes to detecting abnormal behaviour, they are not always capable of offering a contextual explanation or structured results. Alerts are created, although they do not necessarily have enough information to aid in forensic investigation. This gap is especially significant in case of critical infrastructure where it is of equal significance to tell how and why something abnormal is happening, as it is to trace it.

2.4 Windows Infrastructure and Ransomware Behaviour Modelling:

The windows-based systems remain to take centre stage even in the enterprise and industrial setting and as such, they have been a major target of ransomware attacks. Their pervasive nature, as well as extensive integration in organizational processes, gives them a big attack surface, which attackers actively capitalize on.

Active Directory is one of the most important elements of windows environments and it handles authentication, authorization and access control. McDonald and Pat (2022) have shown the ways in which ransomware attackers use the Active Directory to spread laterally across the networks and gain privileges. After having access to domain-level credentials, attackers can successfully gain control over various systems and thus, it becomes even harder to detect.

This act underscores a significant change in the ransomware tactics. Attackers are no longer interested in attacking single machines, but the whole networks. This necessitates that the detection systems should extend beyond analysis of single points and consider system wide interactions.

In reaction to these issues, scholars have endeavored to develop a few ways of modeling ransomware behaviour on windows systems. Conventional methods based on system logs and event-based analysis. The study by CONIT (2024) demonstrated the reconstruction of ransomware incidents with the help of Windows event logs, memory artefacts, and process tracking. These methods would be helpful in terms of attack schedule and impact of the system.

One of the main weaknesses of these methods is however, that they are reactive in nature. Most forensic studies are done once an attack has been perpetrated, and by

this time, a lot of damage could have been caused. There are also cases where attackers will use anti-forensic methods like deleting logs or obscuring them to conceal their actions.

To beat these challenges, an increased focus on behavioural modelling has been experienced. Modern methods do not just use logs but trace system events like process creation, file access patterns and registry changes. These behavioural indicators give previous indicators of malicious behaviour.

However, it is still hard to implement such detection systems in practice. Windows environments are also dynamic and produce huge amounts of data, it is not easy to tell the difference between normal and malicious behaviour. Furthermore, most detection systems are not a part of a forensic framework, which implies that even in cases where an attack is identified, corresponding evidence might not be stored well-organized.

This disastrous relationship between the detection and the forensic analysis underscores the importance of integrated systems to be able to detect and capture evidence simultaneously. In the absence of such integration, organizations stand the risk of losing important information that is necessary in investigating and responding.

2.5 Fragmentation and Identified Research Gap:

When the contemporary body of research is regarded as a unity, what comes out is an interesting trend not of totality but of disunity. Every field of research seems to be advancing separately, nearly like various parts of a greater puzzle are being created, yet never completely assembled.

On the one hand, a robust foundation of literature that concentrates on ransomware detection through machine learning methods exists. Such studies as the one conducted by Manavi and Hamzeh (2020), Urooj, Maarof and Al-rimy (2021) and Kritika (2025) have gone a long way in enhancing accuracy in the detection. They are far more interested in models that can properly classify malicious behaviour, and they may be impressive in controlled settings. Yet, in a bid to be precise, these methods are likely to ignore the post detection processes. The alert is created but the context of the attack the story is usually absent.

Concurrently, other body of research is on digital forensic preparedness. Pilli et al. (2010), Mekala (2025), and Garg et al. (2025) are some of the studies that have highlighted the significance of evidence collection, preservation, and governance. The aim of these frameworks is to make sure that, in case of an incident, investigators can access credible and admissible data. However, even though they are important, these approaches tend to be hypothetical. They are prescriptions of what needs to be done but do not necessarily work in harmony with real-time detection systems. Consequently, forensic processes are often instigated when an attack has already inflicted damage.

Meanwhile, a third line of research focuses on Operational Technology settings, especially the detection of anomalies in industrial systems. The works by Wang et al. (2024), Afaq, Rahmani and Sikora (2025) and Cosnek, Lutkus and Barragan (2024) emphasize the exceptional issues of ensuring the stability and uninterrupted work of OT systems. The studies are aimed at determining abnormal behaviour without interfering with the performance of the system. Nevertheless, they are more operational in nature- they are interested in ensuring that systems are running safely, most of the time without focusing on how the identified anomaly can be converted into structured forensic evidence.

When the three streams are laid side by side, the disconnect can be easily seen. Detection models can be very good at determining the existence of threats but seldom give the entire context in which to investigate. Forensic systems are evidence-based but do not have real-time alerts of smart detection systems. OT-based research guarantees that the systems are stable but does not pay much attention to explained and organized data.

There are several challenges that are brought about through this absence of integration. Detection alerts tend to be one-off events, with no structured contextual artefacts that provide an understanding of what occurred, how it occurred, and what systems were impacted. Forensic systems are not proactive and instead are mostly based on passive recording, which can overlook important evidence when a ransomware attack occurs rapidly. The use of anomaly detection models in OT environments produces alerts, which are not stored in relational structures that can facilitate systematic inquiry or be admitted in court.

The other critical problem is the lack of attention to the Windows-based OT systems. Although the use of windows platforms in the industrial setting especially in the energy infrastructure is common, there is little research that covers this hybrid space. Most of the research either dwells on enterprise IT systems or generic OT architecture, and there is a gap in knowledge on how ransomware acts and whether it should be countered within Windows-based industrial environments.

Combined, these gaps indicate a more profound problem than a lack of solutions, it is a lack of the integration of solutions. The contemporary cyber threats, and ransomware are not limited to separate spheres. They cross the IT and OT boundaries and take advantage of their vulnerabilities. However, the current research environment is not entirely representative of this interrelated fact.

This presents an evident and pressing research gap: an integrated, AI-based digital readiness framework, specifically targeting Windows-based OT systems in critical energy infrastructure, is currently lacking. This framework would require a way of integrating intelligent detection, OT-aware threat analysis, and organized evidence management in forensics into a unified, coherent system.

This is not merely an academic exercise, but a practical need to address this gap. Lack of integration will keep an organization in the reactive stance where it can only detect problems without the full understanding of the problems and respond to events without the evidence to learn. A comprehensive method, then, will be the next logical step of developing cybersecurity of critical infrastructure.

2.6 Positioning of the Current Study:

The identified gaps in the literature are not just theoretical because they represent practical challenges in trying to develop workable cybersecurity systems. During the process of the development of this project, it became clear that current solutions, though each of them is efficient per se, do not easily translate into a complete, end-to-end solution. The presented study thus is not only a conceptual framework, but it is a practical implementation that is able to integrate these fragmented parts into a system.

Instead of considering ransomware detection, OT anomaly detection, and forensic readiness as three distinct issues, this work discusses them as the constituent components of a single pipeline. The intelligence system that has been designed in

this research combines various layers of intelligence whereby each layer seeks to solve a particular weakness that is evident in earlier studies.

The implementation consists of a ransomware classification model, which is based on machine learning and trained on a Windows-based dataset. This model is concerned with the detection of malicious behaviour in the IT environments, patterns of process activity, file operations, and system interactions. Simultaneously, an anomaly detecting model is created based on the HAI ICS dataset that allows the system to track OT behaviour of the industry and to detect when it is not operated according to the common operational trends.

The uniqueness of this study lies in the fact that there are such models, but the way those models are united. Both models are not used separately and are incorporated into a single detection pipeline with their results being standardized to a common format. This enables the system to track both IT-level ransomware threats as well as OT-level anomalies in a single system giving a more holistic picture of system security.

The study based on this layer of detection establishes an automated forensic logging mechanism. In contrast to the traditional methods, where the collection of evidence is activated once an incident is detected, this system can capture the information on the artefact of interest when a detection is made. Every event either of ransomware model or of the anomaly detection model is logged with relevant metadata, including timestamps, source type (IT or OT), event classification and risk scores.

MySQL-based relational database is the forensic backbone of the system to provide a structured and reliable storage. This facilitates orderly storage, retrieval and analysis of events, which was found to be one of major gaps in the literature, absence of systematic evidence handling in detection systems. The system facilitates future investigation and auditing as well as monitoring by the organization of data in relational format.

The other important element of the implementation is the presence of explainable AI elements. The system is not meant to generate opaque predictions but rather to give interpretable predictions that enable users to know why a given event was considered suspicious. This is especially crucial in situations of critical infrastructure, where decision-making should be transparent and explainable.

To combine all these elements into a convenient way of use, a live dashboard monitoring is created with Flask. The dashboard displays the activity of the system, showing the detection of ransomware, OT anomalies, and the level of risk. It is also easy to access historical forensic logs and track down the trends and examine past occurrences. This converts raw detection outputs into useful insights, making the gap between technical models and operational usability.

The study transcends the shortcomings of past studies by incorporating these elements to form one operational system. Although the previous research tended to concentrate either in enhancing the detection accuracy or the development of conceptual forensic frameworks, this research indicates how the two can be operationalized in concert.

This integration is further enhanced using both windows ransomware dataset and the HAI ICS dataset, which enables the system to operate both in the IT and OT domain. This directly concerns the issue of cross-domain security which is frequently not taken into consideration in literature.

Overall, the research makes a disjointed research landscape into a unified implementation. Detection is not isolated anymore, forensic preparedness is not reactive anymore and OT security is no longer disconnected with evidence management. Rather, everything is integrated into a single architecture which accounts for the reality of modern critical infrastructure systems.

Thus, the study can be placed in the category of applied and integrated forensic preparedness, showing the way AI-powered detection, OT awareness, and systematic evidence management can be further integrated into practice. It does not just give us the contribution at the theoretical level but also it gives us a concrete system which can be extended and implemented practically in the energy infrastructure setting.

Table 1: Comparative Analysis of AI & ML Techniques in Digital Forensics for OT Ransomware Detection

Category	Research Work	Contribution	Benefits	Drawbacks	Integration & Impact
Ransomware Detection (Static Analysis)	Manavi & Hamzeh (2020)	CNN-based model using PE header features	High accuracy in controlled environments	Vulnerable to obfuscation and polymorphism	Low (limited real-world adaptability)
Behaviour-Based Ransomware Detection	Urooj et al. (2021)	Pre-encryption behavioural detection model	Early detection before encryption	Requires continuous monitoring and tuning	Medium (not integrated with OT or forensics)
Ensemble ML Detection	Albshaier et al. (2024)	Random Forest & Gradient Boosting for ransomware detection	Robust across multiple ransomware families	Dataset dependency and computational cost	Medium (focused only on detection)
Deep Learning Models	Kritika (2025)	CNN & RNN for sequential ransomware detection	Captures complex temporal patterns	High computational cost, overfitting risk	Medium (limited explainability and deployment)
ML-Based Ransomware Detection (Implemented)	This Study (IT Model)	Logistic Regression, Random Forest, Gradient Boosting on Windows dataset	High accuracy, balanced performance, interpretable results	Requires feature engineering and tuning	High (integrated with OT + forensic logging)
OT Anomaly Detection	Wang et al. (2024)	Lightweight anomaly detection for industrial systems	Suitable for real-time OT environments	High false positives with IT-trained models	Medium (no forensic integration)
Time-Series Pattern Detection	Pekepok et al. (2025)	Hierarchical clustering for sequential	Effective for time-series OT data	Complex implementation	Medium (not integrated with IT detection)

		anomaly detection			
Dataset Analysis in OT Security	Afaq et al. (2025)	Review of OT/IoT datasets	Identifies lack of quality datasets	Limited availability of real-world data	Low (no implementation focus)
OT Anomaly Detection (Implemented)	This Study (OT Model)	Isolation Forest on HAI dataset	Detects unknown anomalies, unsupervised learning	Requires threshold tuning (~13% anomaly rate)	High (combined with IT detection + logging)
Digital Forensic Readiness (DFR)	Pilli et al. (2010)	Introduced forensic readiness concept	Enables proactive evidence collection	Mostly theoretical	Medium (no ML integration)
Forensic Framework for IIoT	Mekala (2025)	Lightweight logging for industrial systems	Reduces performance impact	No intelligent detection integration	Medium
AI + Forensic Integration	Garg et al. (2025)	Combines AI detection with forensic readiness	Moves towards automation	Focus on enterprise systems	Medium
Windows Forensic Analysis	CONIT (2024)	Event logs and memory artefacts for reconstruction	Strong evidentiary support	Reactive approach	Medium
Active Directory Attack Analysis	McDonald & Pat (2022)	Analysis of ransomware in AD environments	Highlights privilege escalation	No real-time detection	Low
OT Security Constraints	Cosnek et al. (2024)	Analysis of OT cybersecurity challenges	Emphasizes safety and latency	Limited ML focus	Low
Integrated Forensic Framework (Final System)	This Study (Full Implementation)	ML detection + OT anomaly + MySQL logging + Flask dashboard	Unified detection, real-time monitoring, forensic-ready system	Implementation complexity	Very High (end-to-end system for real-world deployment)

CHAPTER 3: METHODOLOGY

3.1 Introduction:

As the energy sector becomes more connected and includes Information Technology (IT) and Operational Technology (OT) systems, the efficiency, automation, and real-time decision-making has improved. This convergence has also given industrial systems a new lane to be exposed to an increasingly high level of sophisticated cyber-attacks, including ransomware attacks. Today, ransomware attacks are no longer limited to encrypting files; they are multi-stage attacks that begin with initial compromise, spread laterally, gain privileges, steal data, and then encrypt it. As a result, traditional signature-based security approaches are not well suited to effectively identify new threats.

Meanwhile, organizations struggle to stay digital forensic ready. Threat detection is the priority of many cybersecurity solutions, and they don't usually maintain the necessary evidence for incident response, forensic investigation, or post-incident analysis. This leaves a gap between threat detection and forensic readiness, especially in Operational Technology (OT) where system availability and reliability are key requirements.

In response to this, this research presents an AI-Driven Digital Forensic Readiness Framework, which combines ransomware detection, OT anomaly detection, forensic evidence preservation and real-time monitoring in a unified platform. In this study, the methodology is intended to identify threats intelligently and aiding in forensic investigation by collecting and storing evidence automatically.

In this chapter, the methodological approach taken in the development of the proposed framework is presented. It outlines the research design, system architecture, machine learning models, and anomaly detection algorithms used in the system. The chapter also provides the rationale for these techniques as well as how they relate to the research objectives identified in Chapter 1 and the research gap identified in Chapter 2.

3.2 Research Design:

3.2.1 Research Philosophy and Approach:

This study follows a Design Science Research (DSR) method to design and test the proposed cybersecurity framework. Design Science Research is well suited for research efforts that aim to develop novel technological artefacts that have the potential to address real-world problems. DSR is more about designing, implement, and evaluate real solutions rather than theoretical-based research methodology aimed at explaining phenomena.

The main goal for this research is to create an integrated framework that can detect ransomware attacks, identify abnormal behaviour in an industrial environment, preserve forensic trace and aid real-time threat monitoring. Design Science Research is a suitable methodology for this study because it is a study that will result in a complete operational cybersecurity solution.

This study used the following stages for the DSR process. The first problem identified was from the literature review, which showed a lack of integration between ransomware detection, OT anomaly detection, and digital forensic readiness. Second, a framework architecture was developed that fills this void. Third, models for machine learning and anomaly detection were created and tested on public data. Lastly, the created framework was deployed and added with forensic logging and dashboard monitoring features.

The use of Design Science Research allows the study to go beyond conceptual analysis and bring to light the operationalisation of intelligent detection and forensic readiness in a real cybersecurity setting.

3.2.2 Proposed Framework Architecture:

The proposed framework brings various cyber functions together within a single framework that can support threat detection as well as preparedness efforts for security forensic investigations. The architecture is developed into four major layers, each of which plays an important role in the complete security monitoring and evidence management.

The first layer is the AI Detection Layer that takes data from ransomware and ICS datasets. In this layer, machine learning algorithms process the data that is received

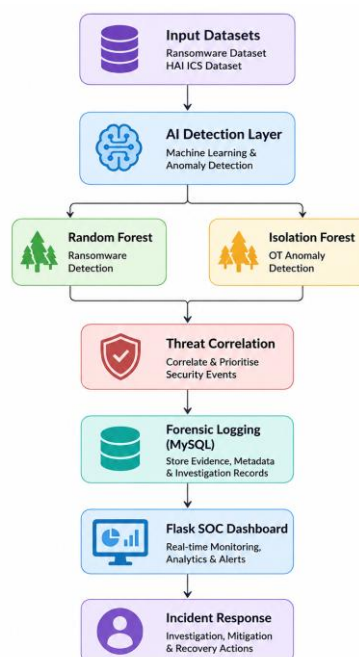
and make forecasts about the potential threats. A classification model is used for ransomware classification (Random Forest), and an anomaly detection model is used for OT (Isolation Forest).

The second layer is the Threat Correlation Layer. This part orchestrates the output of both the detection models and normalises the security events into a standardised format. The correlation process allows the framework to give a unified forward-looking picture of threats from both IT and OT.

The third layer is the Digital Forensic Readiness Layer. Upon detection of a threat, pertinent evidence is immediately captured in a forensic repository. Timestamps, threat classification, confidence level, severity and investigation notes are stored in this repository. The aim of this layer is to guarantee that important evidence is preserved in the initial stages upon the discovery of suspicious activity.

The last layer is a Flask-based Security Operations Center (SOC) Dashboard. This dashboard offers real-time visualization of forensic logs, security events, threat statistics and system status information. The dashboard allows security analysts to track activities, gain insight into incidents and view historical information via a single interface.

Figure 1: Proposed AI-Driven Digital Forensic Readiness Framework



Source: Developed by the Researcher

3.2.3 Justification of Framework

The framework that has been suggested in this research was created to tackle the fragmentation that was highlighted in the literature review. Current research has mostly concentrated on ransomware detection, anomaly detection or forensic readiness, in isolation. These methods are helpful for the detection, but not for forensic investigation.

By combining ransomware detection with OT anomaly detection, the framework can monitor traditional IT threats as well as industrial operational abnormalities in one environment. This is especially vital in the critical energy infrastructure sector, where cyber events often affect information systems and operational processes.

A digital forensic readiness element further sets the framework apart from the traditional detection systems. The frame allows for the automatic collecting of artefacts relevant to the incident now of detection, rather than afterwards. This method helps make evidence more available and aids future investigations.

Real-time visualisation is also another great feature of the framework, as you can do it on the Flask dashboard. Security staff can keep track of threat activity, see forensic records and analyse historical trends without having to use multiple independent tools.

Overall, the proposed architecture offers a single solution that integrates intelligent detection, anomaly analysis, evidence collection, and operational monitoring. The study tries to fill the gap in the research identified in chapter 2 and helps to enhance the cyber resilience and digital forensic readiness in the Windows-based Operational Technology environment by incorporating these capabilities in one framework.

3.3 Models and Algorithms:

The capability of the proposed framework relies on the accuracy of the detection mechanisms in identifying malicious activities and abnormal operational behaviour. For this purpose, several machine learning and anomaly detection models were tested. The models selected were based on their applicability in the context of cybersecurity, their predictive capabilities, computational efficiency, and their compatibility with the datasets used in this study.

It uses supervised machine learning algorithms to detect ransomware, and unsupervised anomaly detection algorithms to monitor OT. Comparative analysis was carried out to find the best possible models for incorporation to the final system.

3.3.1 Ransomware Detection Models:

Logistic Regression:

To serve as a baseline ransomware detection model, Logistic Regression was implemented. Logistic Regression is a supervised machine learning algorithm which is widely used for binary classification tasks. The algorithm determines the class distribution of a given observation according to a collection of input variables.

In this study, Logistic Regression was used for classification of executable files into benign and malicious. The simplicity, ease of computation and the interpretability of the model were the foremost reasons for the selection. It can serve as a benchmark or a baseline model against which more sophisticated machine learning techniques can be tested.

The logistic regression has one benefit in that it can produce a straightforward and easily interpreted probabilistic output. Moreover, the model is relatively low in computational resources, and it can be trained on many data sets. The Logistic Regression model, however, assumes about the relationship between the features and outcomes being linear. Ransomware behavior can be complex, non-linear, and may not be thoroughly represented within the model, as multiple factors of an executable file can contribute to the behavior.

Logistic Regression offers a solid baseline accuracy but is typically not as effective as ensemble learning methods in high dimensional cybersecurity data.

Random Forest:

Random Forest was chosen as the main ransomware detection model in the proposed framework. Random Forest is an ensemble learning method that builds several decision trees and takes the highest voted decision. Combining the algorithms together helps avoid overfitting and increases the accuracy of the prediction.

It works by building many decision trees, using random subsets of the training data and features. Each tree produces an independent classification result, and the final prediction is based on the aggregation of all the trees in the ensemble.

Random Forest is a great approach that makes it suitable for ransomware detection for the following properties: Firstly, it works well with high-dimensional data, meaning that it has many features. Second, the algorithm can capture the non-linear relationships that are common in malware sets. Thirdly, the ensemble structure is better at generalising and is more robust than individual decision trees when compared.

A key benefit is that the Random forest can produce the list of feature importance. These ranks indicate the most important variables in making classification decisions. This information can aid feature selection and help gain insight into the nature of ransomware.

The evaluation results of the various ransomware detection models tested in this study showed that the highest classification performance was obtained by the Random Forest model. The model had the highest accuracy, stability and predictive reliability and was the most suitable for inclusion in the resulting framework.

Gradient Boosting:

Gradient Boosting was also adopted as an extra ensemble learning model to be compared. Gradient Boosting builds trees one by one, while Random Forest builds trees individually. New tree corrects errors created by the previous trees, improving the performance of the model over time.

Gradient Boosting can learn sequentially and thereby gain high prediction accuracy and be able to see fine-grained patterns in complex data. The algorithm works well for non-linear feature interactions in classification problems.

However, Gradient Boosting has several drawbacks. Generally, training is more computationally intensive than Random Forest, because of the sequential nature of the learning process. Furthermore, hyperparameter tuning is crucial for the model's performance, and it can be challenging to avoid overfitting the model.

Gradient Boosting gave good classification results in the comparative evaluation. But, Random Forest had slightly better predictive performance with lesser computational complexity. Thus, Gradient Boosting was left in for comparison purposes, but it was not chosen for the final ransomware detection model.

Table 2: Comparison of Ransomware Detection Models

Model	Advantages	Limitations
Logistic Regression	Simple, interpretable, fast training	Limited ability to model non-linear relationships
Random Forest	High accuracy, robust, feature importance	Higher memory requirements
Gradient Boosting	Strong predictive capability	Computationally intensive and sensitive to tuning

Source: Developed by the Researcher

3.3.2 OT Anomaly Detection Models:

Isolation Forest

Isolation Forest was chosen as the main anomaly detection model in OT environments. The Isolation Forest algorithm is an unsupervised machine learning technique for anomaly detection. The algorithm does not need a set of labelled attack data and instead detects an anomaly by setting off observations that are far away from normal behavior.

The algorithm works by recursively splitting data with random features and split values. Anomalies are also more quickly isolated than regular data points because they are rare and significantly different from those in the regular data stream. This property makes it possible to detect abnormal operational behaviour efficiently.

Industrial Control Systems are well suited to Isolation Forest, since labeled OT data is typically sparse. Moreover, it is also a light-weight computation with efficient processing of huge amount of industrial sensor data. Its properties are ideal for real-time cybersecurity monitoring in critical infrastructures.

Neural Network:

The anomaly detection process also included a Neural Network model. Neural Networks can learn very complex patterns by using a bunch of layers of interconnected artificial neurons. The model can detect subtle relationships in big data and is commonly applied in cybersecurity applications.

In this study the Neural Network was used as a comparative deep learning approach. The model exhibited good anomaly detection performance, but it used much more computational resources and training time than the Isolation Forest. Further, the complexity of neural networks can make them less interpretable than simpler anomaly detection algorithms.

Model Selection Justification:

After experimentation, Random Forest and Isolation Forest were chosen to be incorporated into the final framework. Among the ransomware detection models, Random Forest performed best and was robust and interpretable. Likewise, Isolation Forest showed promising anomaly detection performance with no need for labelled attack data, and its computational properties made it suitable for deployment at OT sites.

Together, these two models make it possible to tackle both IT-centric ransomware attacks and OT-centric operational anomalies. Therefore, the proposed architecture offers robust threat detection capabilities while ensuring efficiency and scalability in critical infrastructure environments.

3.4 Tools and Technologies:

The proposed framework has been implemented in various programming languages, machine learning libraries, database technologies, and web development tools. Python is chosen as the main programming language because of its comprehensive support for A.I., machine learning, and cybersecurity use. Jupyter Notebook was used for the development and experiments, as it offers an interactive environment for data analysis, model training, and visualisation of results.

During implementation, many Python libraries were used. Data manipulations and preprocessing were performed using pandas and numerical computations using NumPy. Scikit-Learn offered implementations of the machine learning and anomaly

detection algorithms used within the framework: Logistic Regression, Random Forest, Gradient Boosting and Isolation Forest. Joblib was used to store trained models for use in the final system.

To enable digital forensic readiness, threat events, timings, severity, confidence level and investigation records were stored in a MySQL database. The database provides structured evidence storage and enables future forensic investigations.

The monitoring and visualisation part of the framework was created with Flask. For real-time updates on the dashboard, Flask-SocketIO was integrated, and HTML, CSS, and JavaScript were used to design the user interface and interactive features of the dashboard.

Table 3: Tools and Technologies Used

Technology	Purpose
Python	Machine learning and system development
Jupyter Notebook	Development and experimentation
Pandas	Data preprocessing
NumPy	Numerical computation
Scikit-Learn	Machine learning implementation
Joblib	Model storage and deployment
MySQL	Forensic evidence storage
Flask	Web application development
Flask-SocketIO	Real-time communication
HTML/CSS/JavaScript	Dashboard interface

Source: Developed by the Researcher

3.5 Datasets:

This study applied two public data sets. The first dataset is a windows ransomware dataset from Kaggle, which was used for ransomware classification. The following dataset consists of around 62,485 executable samples (Portable Executable (PE) header features). DebugSize, DebugRVA, ResourceSize, DllCharacteristics, NumberOfSections, and ExportSize are important attributes that can be used to identify malicious activities.

Irrelevant data like FileName, md5Hash were stripped off before training. The data was normalised using feature scaling, and feature importance ranking using Random Forest was performed. The dataset was then split in 80:20 ratio of training and testing.

The second dataset was from the HIL Based Augmented Industrial Control System (HAI) dataset for OT anomaly detection. There are around 216,001 training records and 43,201 test records from industrial sensors, actuators and process. Data preprocessing involved removing the time stamp, converting to numeric, filling missing values, and standardisation with 'StandardScaler'.

The ransomware dataset was chosen because it includes representative windows executable features for malware classification, and the HAI dataset contains realistic Operational Technology data for anomaly detection research. But both data sets are incomplete: They don't represent future threats and industrial environments well.

Table 4: Dataset Summary

Dataset	Records	Features	Purpose
Windows Ransomware Dataset	62,485	18	Ransomware Classification
HAI Dataset	259,202	84	OT Anomaly Detection

Source: Developed by the Researcher

3.6 Implementation Plan:

Integrating ransomware detection, anomaly detection, preserving forensic evidence, and real-time monitoring required a logical progression of stages to ensure seamless integration.

The first stage involved dataset acquisition and preparation. To determine attributes applicable to machine learning analysis, both the ransomware and the HAI datasets were collected and analyzed. The data were then cleaned, preprocessed, scaled, and selected, to ensure the quality and boost the model performance.

In the second stage, model development was done. Logistic Regression, Random Forest, and Gradient Boosting algorithms were trained and tested on the Windows ransomware dataset for ransomware detection. A comparative analysis of the models

was performed to identify the best classification model. To build and test models for Operational Technology anomaly detection, two models, Isolation Forest and Neural Network models were developed and tested with the HAI dataset.

The third stage involved model evaluation and selection. Analysis of performance metrics like accuracy, precision, recall, F1 and confusion matrices and anomaly detection statistics were performed. The results were used to choose the final models that were integrated into the framework: Random Forest and Isolation Forest.

The fourth stage focused on forensic readiness implementation. A MySQL database was created to hold security events created by the detection models. Suspected activity was recorded with automated logging systems, which captured a log of time, threat type, confidence or severity level, and notes on the investigation.

The fifth stage involved the development of a Flask-based Security Operations Center dashboard. The dashboard was created to provide a visual representation of ransomware detections, OT anomalies, the severity of threats, forensic records and system status information. Flask-SocketIO has been added to allow real-time updates.

System integration and testing was the final step. All elements have been integrated into one architecture to support detection models, forensic logging and dashboard interfaces as one cybersecurity platform.

3.7 Ethical Considerations:

There were several ethical issues raised in this research. Publicly available datasets were used, and no personal, confidential or sensitive information was collected from individuals. Thus, the threat to privacy of human subjects was reduced.

The framework was created only for defensive cybersecurity purposes. The goal of the research is to enhance the capability of detecting threats, maturing cyber forensic readiness, and mitigating cyber resilience in critical infrastructure settings. The system developed is not designed to be able to support offensive cyber activities or to be able to be used for unauthorised access to computer systems.

Preservation of evidence and data integrity were also taken into account. Structured forensic logging guarantees accurate and uniform security event logging. This will help with transparency, accountability and future forensic investigations.

The possibility of biases in machine learning models was recognized. Selected datasets may not be complete for the ransomware variants or industrial environments. Consequently, the effectiveness of models can vary in different use settings. Thus, the results presented should be viewed in the context of the data sets employed.

Lastly, the research was carried out in a manner that respected academic integrity and institutional ethical standards for studies involving cybersecurity and AI.

3.8 Summary:

The methodology used to design the proposed AI-Driven Digital Forensic Readiness Framework was outlined in this chapter. The design of the framework was done using the Design Science Research approach, which are the approaches for guiding in the design, implementation and evaluation of the framework. The selected models for machine learning and anomaly detection were presented in the chapter: Logistic Regression, Random Forest, Gradient Boosting, Isolation Forest and Neural Networks. The tools and technologies used for development were discussed, as well as the ransomware and HAI datasets used for training and evaluation.

The implementation plan outlined the stages involved in developing and integrating the framework, while ethical considerations highlighted issues related to privacy, responsible AI usage, and data integrity. Together, the methodology creates a framework for future implementation and evaluation activities described in the next chapter. The implementation of the framework, experimental results, performance evaluation and analysis are presented in Chapter 4.

CHAPTER 4: IMPLEMENTATION AND RESULTS

4.1 Introduction:

This chapter describes the implementation, experimental testing, and performance evaluation of the proposed AI-Driven Digital Forensic Readiness Framework for ransomware detection and Operational Technology (OT) anomaly detection in critical infrastructure environment. The framework aimed to combine intelligent threat detection, automated forensic evidence preservation and monitoring capabilities into a single cybersecurity platform.

The implementation was realized by applying machine learning techniques with Python; the MySQL forensic evidence repository and a Flask based Security Operations Center (SOC) Dashboard. The ransomware detection component was built on top of supervised machine learning algorithms such as Logistic Regression, Random Forest, and Gradient Boosting algorithms and the OT anomaly detection component used models of Isolation Forest and Neural Network to find the OT system behaviour that was abnormal. The results of these models were merged with a forensic logging system that could automatically capture evidence of threats and aid in future investigations.

The chapter starts with the implementation and evaluation of the ransomware detection module that encompasses data preprocessing, feature selection, model training and evaluation. Then, the implementation of the OT anomaly detection module is discussed and then the deployment of the forensic logging repository and the flask application-based dashboard. Finally, the effectiveness of the proposed framework was tested with the research aim and the main findings from the experimental results were discussed.

4.2 Ransomware Detection Implementation and Results:

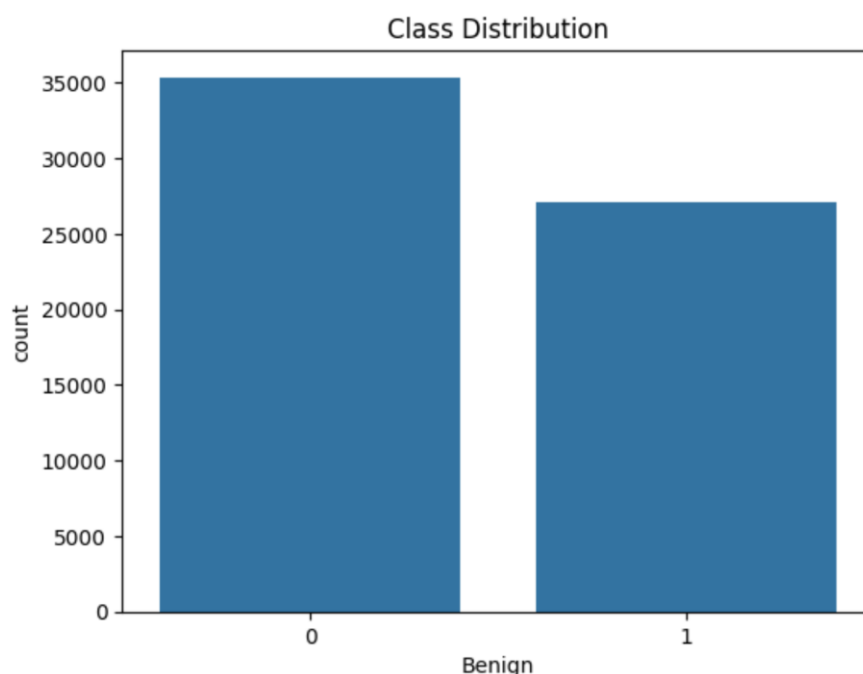
4.2.1 Dataset Exploration and Initial Analysis:

The ransomware detection module was created by combining benign and ransomware Windows Portable Executable (PE) headers. The dataset was composed of about 62,485 executable records, comprising a variety of structural properties from Windows applications. They contain details about the executable's behaviour, memory usage,

usage of resources, and linking properties, which are valuable for distinguishing malicious software from valid applications.

Exploratory Data analysis was performed before developing the model to learn about the characteristics of the data and its structure. This was a significant step to determine class distributions, analyze feature relationships and evaluate the quality of the data. By grasping these attributes, they were better able to train the machine learning models and handle potential challenges like class imbalance and feature redundancy, if they arose during the pre-processing phase.

Figure 2: Class Distribution of Benign and Ransomware Samples



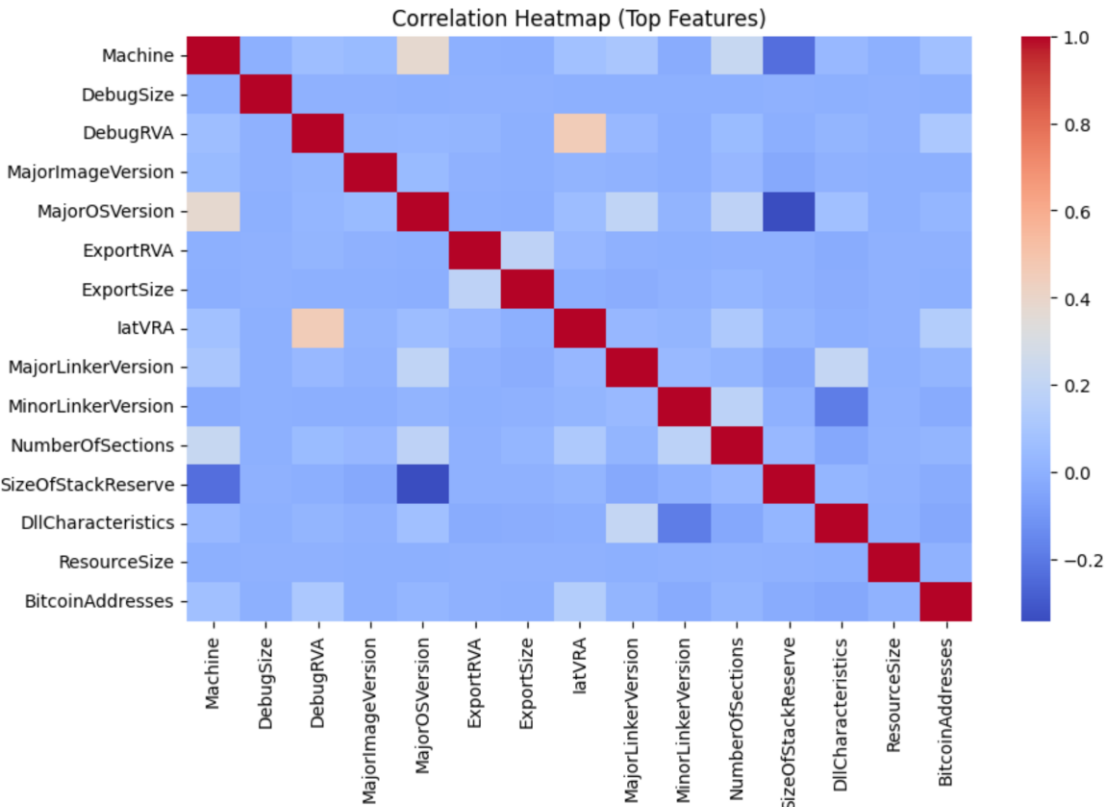
Source: Developed by the Researcher using Jupyter Notebook Output

The class distribution analysis showed that the data set was comprised of an adequate number of benign and ransomware samples. This even distribution avoided the risk of classification bias and set up an adequate ground for supervised machine learning. In the context of cybersecurity applications, a balanced dataset is even more crucial since an imbalanced class distribution can lead to models being biased towards the majority class and overlooking malicious samples.

A class distribution evaluation was conducted after the class and a correlation analysis was conducted to examine relationships between the extracted PE header

features. The purpose of correlation analysis is to get help in determining which variables are closely correlated and to gain a general idea of the structure of the data. Knowing about feature interactions can help in feature selection as highly correlated attributes can contain redundant information.

Figure 3: Correlation Heatmap of PE Header Features



Source: Developed by the Researcher using Jupyter Notebook Output

Correlation heatmap showed different levels of association between the different executable features. The relationships of some of the variables were moderate, but most of the features added distinct information which could be useful to boost the classification accuracy. The exploratory analysis thus verified that the dataset was suitable for ransomware detection and was aiding in the subsequent preprocessing and feature engineering steps.

4.2.2 Data Preprocessing:

The data went through preprocessing to enhance data quality, eliminate irrelevant data, and prepare them for machine learning analysis. Preprocessing is crucial for enhancing the predictive accuracy and reliability of machine learning models, given that their performance is greatly dependent on the input data quality.

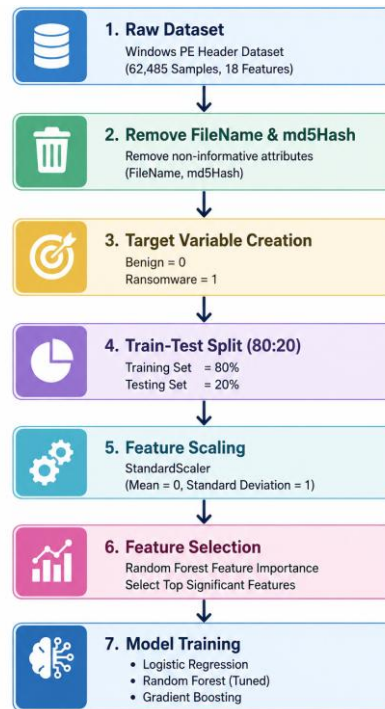
The first phase was the elimination of "trivially" non-informative attributes. FileName and md5Hash are not part of this analysis, as they are mostly used to identify the files and are not used to classify the behaviour. The presence of such variables could cause noise in the learning process and could diminish the capacity of the model to generalize.

The attributes were filtered out, and the target variable was ready for supervised classification. Samples were classified as either benign or ransomware, allowing the machine learning algorithms to learn the distinguishing features from the extracted PE header features.

The dataset was then split into training and test sets with a ratio of 80:20. This training set was used in the model development, and the testing set was used for independent performance evaluation. This separation allowed for the model to be evaluated on new data that had not been used before, which helps to prevent overfitting.

The technique for feature scaling used was the StandardScaler method to ensure consistency of numerical values across the variables. The features were standardised to create a common scale by adjusting their mean and standard deviation. This process prevented that larger number variables from having an undue impact on the learning process.

Figure 4: Ransomware Data Preprocessing Workflow



Source: Developed by the Researcher

4.2.3 Feature Selection and Importance Analysis:

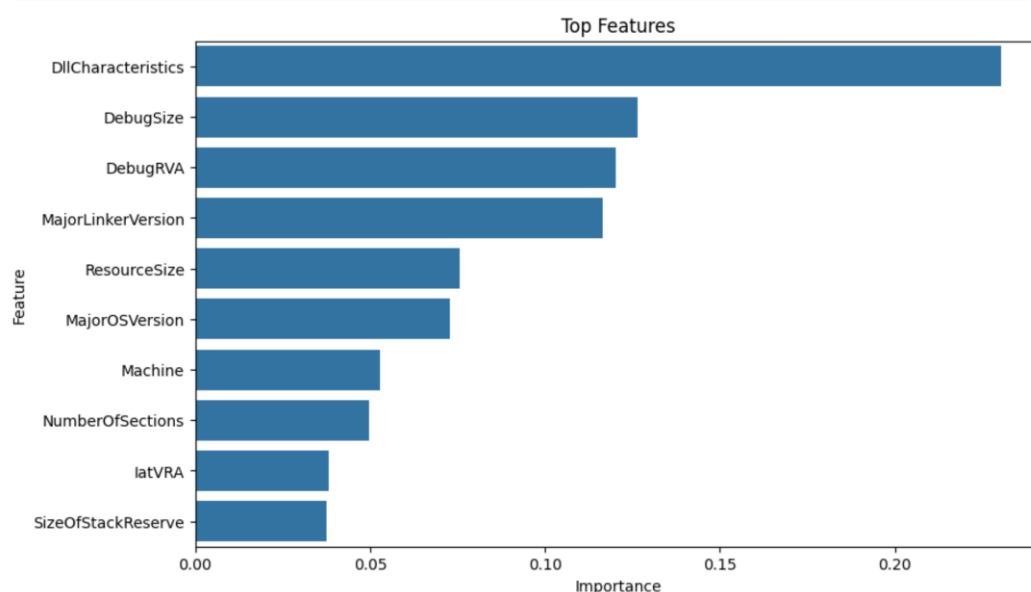
The most influential attributes for the classification of ransomware were identified through feature selection. A smaller number of features also makes the computational process more efficient, the model less complex, and more interpretable, without compromising the predictive power of the model.

Random Forest feature importance analysis was performed to determine the order of importance of the attributes in terms of their contribution to the classification accuracy. The analysis showed that several characteristics of the PE header were significant in determining the difference between ransomware and non-malicious applications.

Figure 5: Feature Importance Analysis

feature importance

```
In [22]: plt.figure(figsize=(10,6))
sns.barplot(x='Importance', y='Feature', data=feature_df.head(10))
plt.title("Top Features")
plt.savefig(f"{output_path}/feature_importance.png")
plt.show()
```



Source: Developed by the Researcher using Jupyter Notebook Output

Based on the analysis, the most influential features detected were:

Table 5: Top Features Selected for Ransomware Detection

Feature	Description
DllCharacteristics	DLL configuration and security characteristics
DebugSize	Size of debugging information
DebugRVA	Relative virtual address of debugging information
MajorLinkerVersion	Linker version information
ResourceSize	Size of executable resources
NumberOfSections	Number of sections within the PE file
latVRA	Import Address Table information

Source: Developed by the Researcher using Jupyter Notebook Output

This is the output from the Jupyter Notebook used by the Researcher to develop this document. This is the output from the Jupyter Notebook that the Researcher used to create this document.

The results show that the PE header attributes offer good means of identifying malicious actions. The features relating to the executable structure, linking behaviour and memory allocation proved to be most effective in distinguishing ransomware samples from actual software.

4.2.4 Hyperparameter Optimisation:

GridSearchCV was used to perform hyperparameter optimisation for the Random Forest model to maximize classification performance. Differing parameters were tried to determine the one that would give the best prediction.

Optimisation process aimed at parameters that determine number of decision trees, tree depth, node splitting parameters. The obtained configuration was then used to train the final ransomware detection model.

Table 6: Optimal Random Forest Hyperparameters

Parameter	Optimal Value
n_estimators	150
max_depth	None
min_samples_split	2
min_samples_leaf	1
bootstrap	True

Source: Developed by the Researcher using Jupyter Notebook Output

The optimized configuration for Random Forest yielded a good compromise between prediction accuracy and computational complexity, ensuring that the model can be implemented in the proposed setting.

4.2.5 Model Training and Development:

After data preprocessing, feature selection and hyperparameter optimization, three supervised machine learning algorithms were trained and tested to classify ransomware. The aim of this step was to find the best model to be deployed in the proposed framework. The widely used Logistic Regression, Random Forest, and

Gradient Boosting algorithms were chosen because they can be used to classify data in binary classification problems and they have been used in many cybersecurity studies.

Logistic Regression

A baseline classification model Logistic Regression was used. The selected PE header features were used to train the algorithm, and the testing set was used for evaluation. Logistic Regression was a linear classification technique and was used to act as a benchmark to compare the performance of more sophisticated ensemble learning techniques.

The model was able to achieve satisfactory classification performance, but it failed to capture the complex non-linear relationship between the attributes of ransomware completely. However, the results served as a baseline to evaluate the performance of the ensemble-based methods.

Random Forest

Random Forest was used as the main ransomware detection model. Optimized hyperparameters found through the GridSearchCV were used by the algorithm and predictions were made with an ensemble of decision trees. Subsets of features and observations were randomly selected to train each tree, ensuring better generalisation and avoiding overfitting.

The model exhibited good prediction ability and was able to detect a ransomware pattern in the PE header dataset. The ensemble structure facilitated the ability of the algorithm to detect higher-order feature interactions and remain robust and stable.

Gradient Boosting

To benchmark the performance of Gradient Boosting, it was applied as an ensemble learning method. Random Forest is trained independently of each other and Gradient Boosting builds trees one by one and corrects the error of the previous tree.

The model performed well in terms of classification accuracy and was able to learn subtle behavioral trends in the data. The complexity of the training, however, and the computational costs were greater than in the case of Random Forest.

The results of each model were tested with several performance metrics to decide which of them is best suited for deployment in the overall system.

4.2.6 Performance Evaluation:

The accuracy, precision, recall and F1-score were used to evaluate the model. These are used to measure how well classification works and are typically employed in cybersecurity applications where false positive minimization and detection capability play crucial roles.

The accuracy indicates the percentage of correctly classified samples, whereas the precision indicates the percentage of the ransomware samples that have been correctly identified as malicious. Recall is used to measure the capacity of the model to detect the instances of ransomware successfully, while the F1-score is used to measure precision and recall in a balanced way.

Table 7: Performance Comparison of Ransomware Detection Models

Model	Accuracy	Precision	Recall	F1 Score
Logistic Regression	86.72%	85.07%	92.82%	88.78%
Gradient Boosting	99.22%	99.21%	99.41%	99.31%
Random Forest	99.66%	99.63%	99.76%	99.70%

Source: Developed by the Researcher using Jupyter Notebook Output

The results show significant performance variations for the models considered. Both ensemble learning algorithms performed much better than Logistic Regression in classification performance, which was acceptable. This is due to the complexity of the ransomware datasets and the limitations of linear classifiers.

Gradient Boosting outperformed all other algorithms, with an accuracy of more than 99%. The model performed well with high accuracy and recall rates, demonstrating its effectiveness in correctly identifying ransomware samples and reducing false predictions.

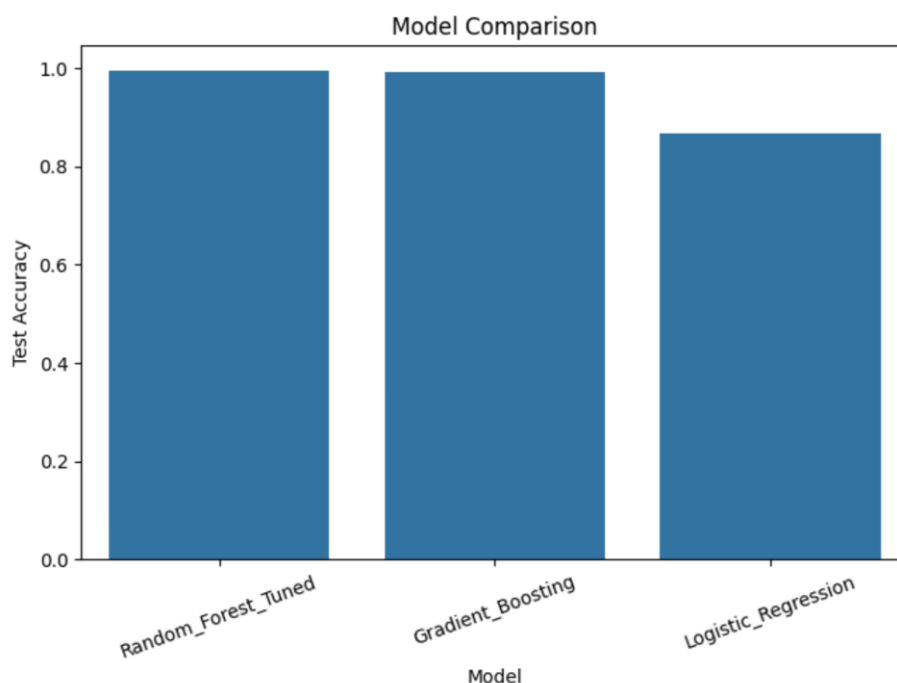
Random Forest showed the best performance with respect to all the evaluation metrics. The model produced an accuracy of 99.66%, precision of 99.63%, recall of 99.76%, and an F1-score of 99.70%. The results reflect the excellent classification

performance and prove the feasibility of using Random Forest in critical infrastructure for ransomware detection.

Figure 6: Comparison of Machine Learning Model Accuracy

comparison plot

```
In [20]: plt.figure(figsize=(8,5))
sns.barplot(x="Model", y="Test Accuracy", data=results_df)
plt.title("Model Comparison")
plt.xticks(rotation=20)
plt.savefig(f"{output_path}/advanced_comparison.png")
plt.show()
```



Source: Developed by the Researcher using Jupyter Notebook Output

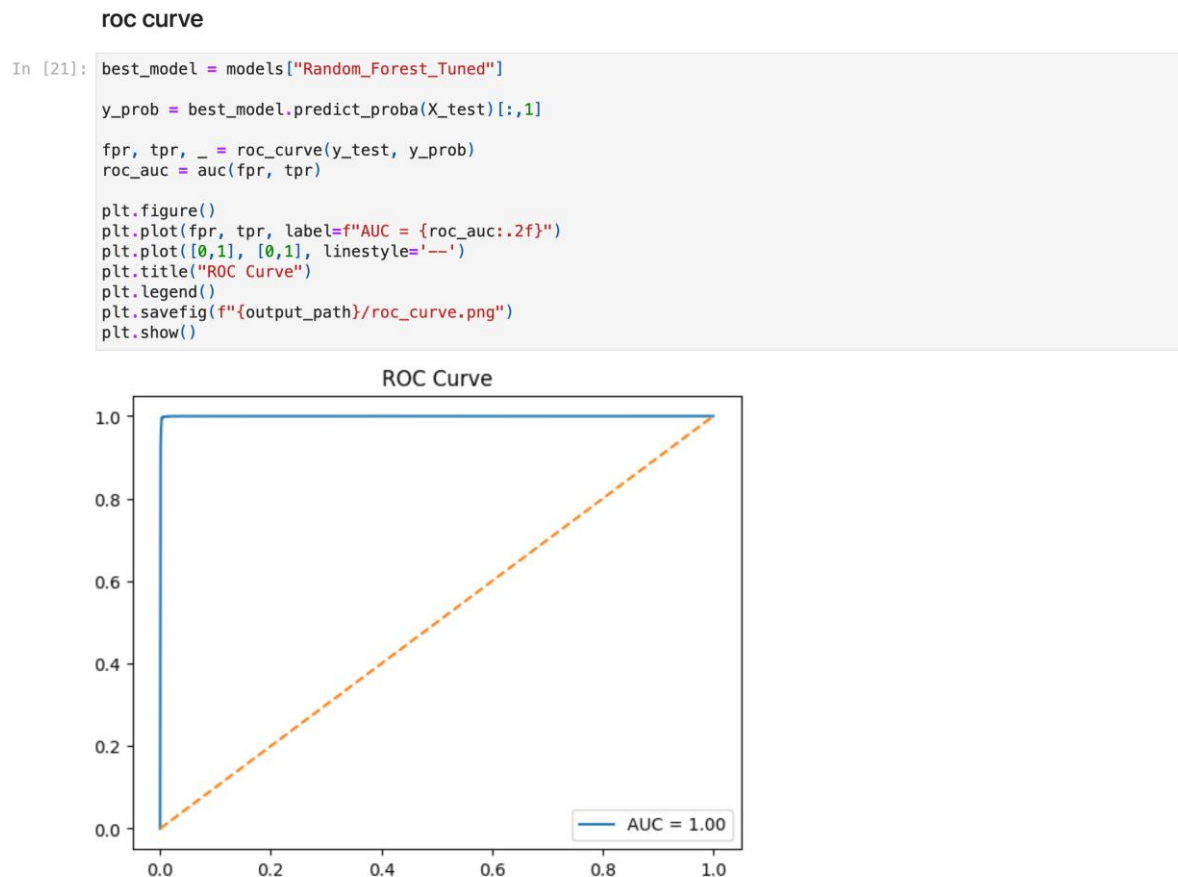
The comparison graph again demonstrates the supremacy of the Random Forest model over the other models. Overall, the results demonstrate the benefits of ensemble learning methods over the conventional linear classification methods for analysing ransomware datasets.

4.2.7 ROC Curve Analysis:

Receiver Operating Characteristic (ROC) analysis was also performed to assess the classification performance. The ROC curve graphically depicts the relationship between the True Positive Rate (TPR) and False Positive Rate (FPR) over a range of classification thresholds. The analysis is especially relevant to cybersecurity applications where it can be used to evaluate how well a model can discriminate between malicious and benign samples.

The Area Under the Curve (AUC) is a quantitative measure of the effectiveness of classification. A value close to 1.0 is excellent discrimination, whereas lower values may be indicative of less discrimination.

Figure 7: Receiver Operating Characteristic (ROC) Curve



Source: Developed by the Researcher using Jupyter Notebook Output

An AUC value of around 1.00 was obtained in ROC analysis, demonstrating that the classification capability is close to perfect. This finding proves that the features and machine learning method are effective to differentiate ransomware samples from legitimate applications. The high AUC value further indicates the reliability of the Random Forest model and justifies its use as the final ransomware detection model in the proposed system.

4.2.8 Discussion of Ransomware Detection Results:

Experimental results validate the capabilities of machine learning methods for the ransomware classification from the PE header features. Data preprocessing, feature

selection, hyperparameter optimisation and ensemble learning resulted in very accurate classification results.

The feature importance analysis showed that the attributes DllCharacteristics, DebugSize, DebugRVA, ResourceSize and NumberOfSections were important in detection of ransomware. The results indicate that structural features of the executable files are useful and can be used effectively in automated threat identification.

Random Forest consistently outperformed all the models on all the evaluation measures. The model was found to be more effective in terms of classification accuracy, and it achieved high levels of precision and recall. This means that the algorithm was able to accurately detect ransomware samples while reducing both false positives and false negatives, both essential during operations in a cybersecurity environment.

Gradient Boosting gave outstanding results as well and validated the power of ensemble learning when it comes to malware analysis. However, the performance of Random Forest was slightly higher and lower in complexity of implementation. While Logistic Regression was useful as an initial model it was found to be less effective as it assumed linear decision boundaries.

The ransomware detection module was able to accomplish the first major goal of the research by providing a very accurate and reliable means of detecting ransomware threats. The results validate the feasibility of using the Random Forest for integration in the proposed AI-DDrF and serve as a foundation for the remainder of the forensic logging and dashboard monitoring features of the system.

4.3 OT Anomaly Detection Implementation and Results:

4.3.1 Dataset Preparation:

The Operational Technology (OT) anomaly detection module of the proposed framework has been designed with the help of HIL-Based Augmented Industrial Control System (HAI) dataset. The data set was chosen since it is realistic for an industrial control system and includes sensor values, actuator states, and process values typical in critical infrastructure. Unlike enterprise datasets, the HAI dataset is an operational dataset of industrial systems and thus is appropriate for anomaly detection technique evaluation.

The data set contained 216,001 training records, 43,201 testing records, and a total of 259,202 observations of 84 variables. These are various operational aspects of industrial processes and are a complete representation of normal and abnormal system behaviour. The number of observations allowed for a comprehensive model training and evaluation and helped to identify complex operational patterns.

Table 8: HAI Dataset Overview

Dataset Component	Records
Training Dataset	216,001
Testing Dataset	43,201
Total Records	259,202
Features	84

Source: Developed by the Researcher using HAI Dataset Processing Results

The dataset was examined before building the model to ensure the quality of the data and to identify the attributes that need preprocessing. The dataset structure was verified to have enough operational data to enable anomaly detection and the efficacy of machine learning methods in an industrial environment.

4.3.2 Data Preprocessing:

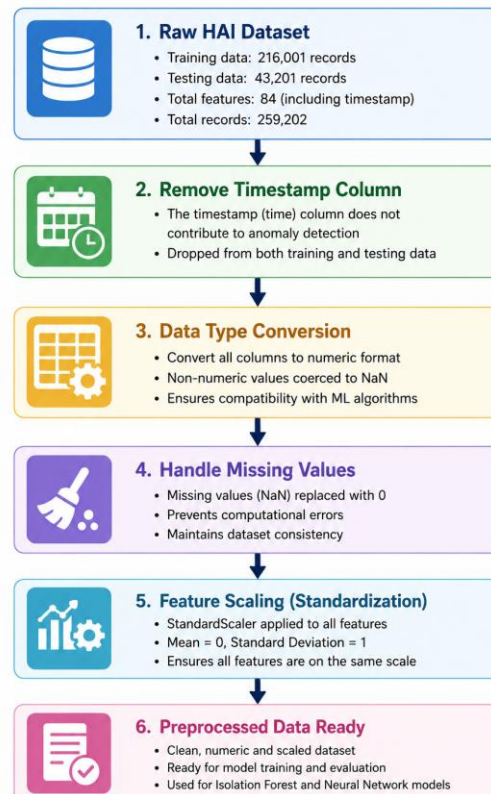
Data was preprocessed to make it suitable for anomaly detection analysis. In many industrial datasets, attributes are measured on different numerical scales, making it important to preprocess the data to enhance model performance and consistency between attributes.

The first step was to drop the timestamp attribute as it was not directly used for anomaly classification. After removing the timestamps, all other variables were transformed into machine learning variable formats. The missing values that were discovered during the conversion were also set to zero to avoid computational errors and inconsistencies in the data set.

The StandardScaler technique was then applied to make the features of the dataset comparable. All variables were standardised to make them comparable by using the mean and standard deviation. This eliminated high numerical values of variables from

dominating the training of the model and allowed for equal comparison between all the measurements made by the sensors.

Figure 8: OT Data Preprocessing Workflow



Source: Developed by the Researcher

The finished preprocessing process yielded a clear, standardized data set that could be used for detecting anomalies. This processed data was then fed into the Isolation Forest and Neural Network models to train and test them.

4.3.3 Isolation Forest Implementation and Results:

The proposed framework has adopted Isolation Forest as the major unsupervised anomaly detection algorithm. The choice of the algorithm is because it is specifically developed for anomaly detection and does not need labelled attack data. This is very useful in Operational Technology environments where it is often difficult to gather accurately labelled sets of anomalies.

With 200 estimators, 20% contamination level and random state of 42 for consistency in experimental results, it was configured. The contamination parameter is the expected value of the proportion of anomalies in the data and directly affects the sensitivity of the model.

The model was trained and then applied to the test data set for the purpose of identifying the abnormal observations. The generated predictions were based on the isolation characteristics of the records and were used to determine whether a record was normal or anomalous. The results showed that the model identified 5677 anomalous records of the 43201 testing records, with an anomaly detection rate of 13.14%.

Table 9: Isolation Forest Detection Results

Metric	Value
Test Samples Analysed	43,201
Detected Anomalies	5,677
Anomaly Percentage	13.14%
Number of Trees	200
Contamination Level	0.02

Source: Developed by the Researcher using Jupyter Notebook Output

Figure 9: Detected Anomalies Over Time

Anomaly Count

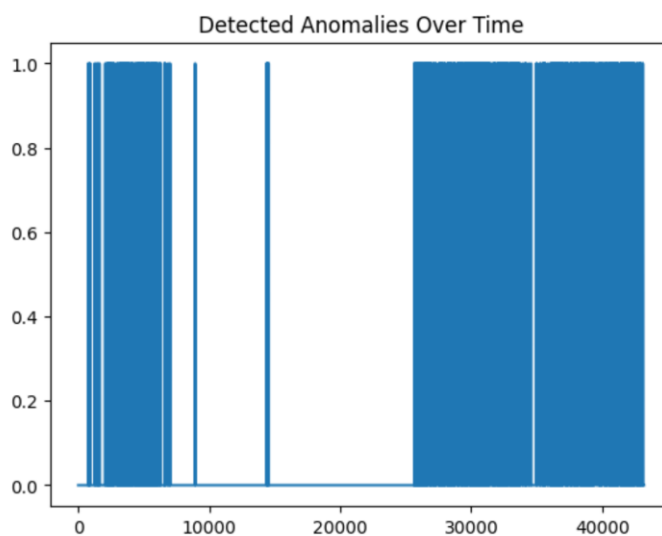
```
In [57]: anomaly_count = np.sum(preds)
total = len(preds)

print("Total Samples:", total)
print("Detected Anomalies:", anomaly_count)
print("Anomaly Percentage:", (anomaly_count / total) * 100)
```

```
Total Samples: 43201
Detected Anomalies: 5677
Anomaly Percentage: 13.140899516214901
```

```
In [58]: ### plt.figure(figsize=(12,5))
plt.plot(preds)

plt.title("Detected Anomalies Over Time")
plt.savefig(f"{output_path}/anomaly_plot.png")
plt.show()
```

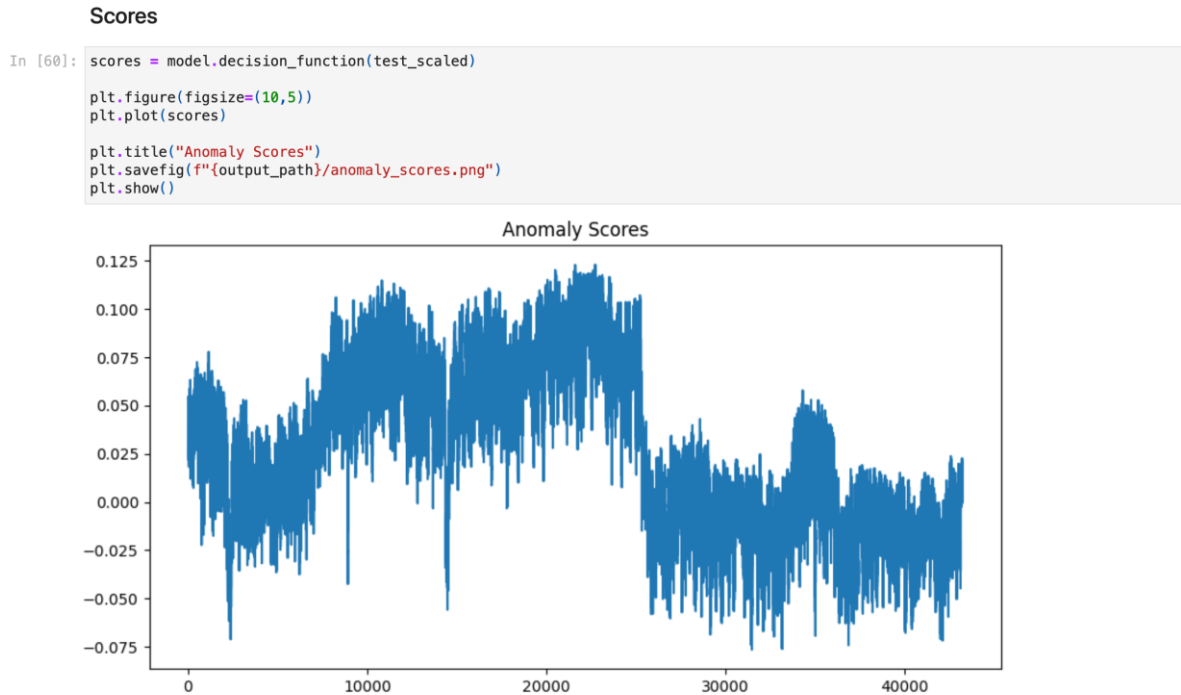


Source: Developed by the Researcher using Jupyter Notebook Output

Anomaly detection visualisation showed up some periods of abnormal operational behaviour in the testing data set. Groups of abnormally recorded observations have been found at several operational intervals, indicating that there is variation from industrial process behaviour. These deviations could be a result of cyberattacks, sensor issues, process anomalies, or equipment problems that need further investigation.

Anomaly scores obtained by the Isolation Forest decision function were further examined to assess model behaviour.

Figure 10: Isolation Forest Anomaly Scores



Source: Developed by the Researcher using Jupyter Notebook Output

A clear distinction between normal and abnormal observations was observed in the anomaly score distribution. The lower the anomaly score the more unusual the behaviour and the higher the anomaly score the more normal the operation. The separation seen in the score distribution validates the effective identification of abnormal events in industrial environments by the model.

Further experiments with several contamination levels showed that contamination value increased as the number of anomaly detections increased. This discovery underscores the need to carefully choose contamination parameters for deployment to ensure high sensitivity and low false positives.

4.3.4 Neural Network Implementation and Results:

A Neural Network model was created with a Multi-Layer Perceptron (MLP) architecture as an alternative deep learning approach. Neural Networks can learn very complex and non-linear relations between variables and can be used to detect subtle anomalies in industrial process data.

The architecture implemented is two hidden layers with 64 and 32 neurons, respectively. To add non-linearity to the learning, Rectified Linear Unit (ReLU) activation function was used, and the model was trained for 10 maximum iterations. The testing dataset was then used for the second stage of testing the Neural Network, which was to detect unusual operational behaviour.

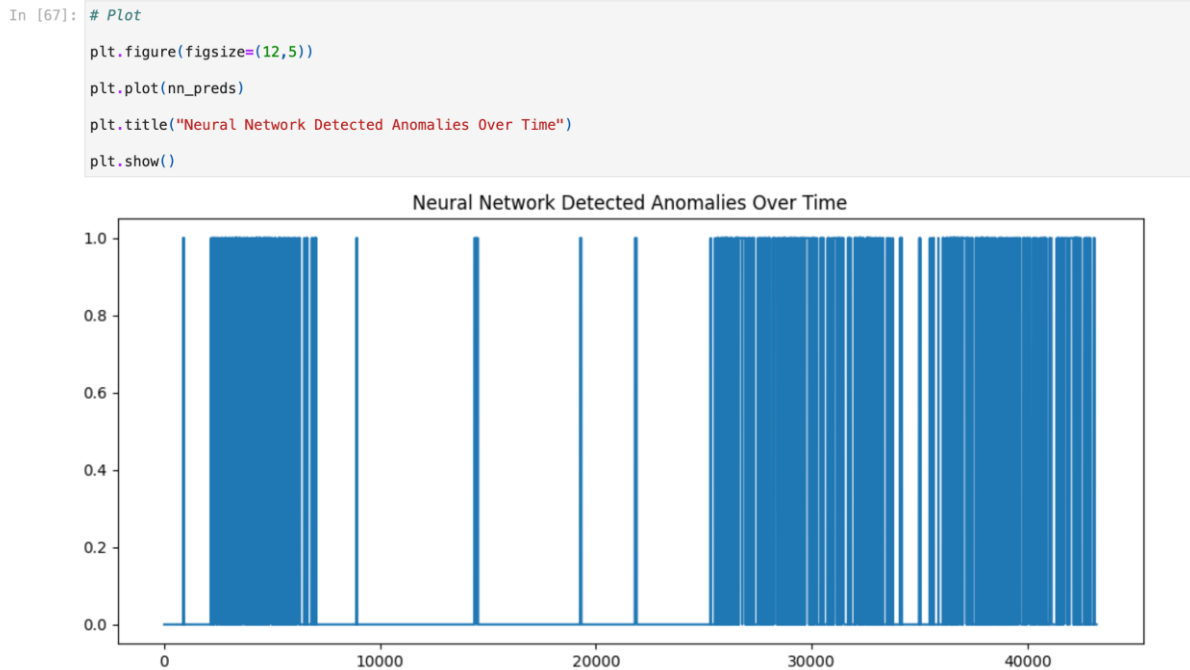
The results showed that the model identified 8,316 anomalous observations, which is 19.25% of the testing data. This anomaly rate was significantly higher than the one obtained by the Isolation Forest model.

Table 10: Neural Network Detection Results

Metric	Value
Test Samples Analysed	43,201
Detected Anomalies	8,316
Anomaly Percentage	19.25%
Hidden Layer 1	64 Neurons
Hidden Layer 2	32 Neurons
Activation Function	ReLU

Source: Developed by the Researcher using Jupyter Notebook Output

Figure 11: Neural Network Detected Anomalies



Source: Developed by the Researcher using Jupyter Notebook Output

The visualisation shows that the Neural Network detected more behavioural deviations throughout the test period. The improvement in the number of anomalies indicates that the deep learning model was able to identify certain subtle operational anomalies that the Isolation Forest approach could not.

The higher sensitivity of the Neural Network could provide better threat visibility in industrial environments. But more sensitivity can also lead to more false positive alerts and so more workload to investigate and operational overhead.

4.3.5 Comparative Analysis & Discussion:

A comparative analysis was carried out, based on anomaly detection rates and operational suitability, to assess the effectiveness of the anomaly detection approaches.

Table 11: Comparison of OT Anomaly Detection Models

Model	Detected Anomalies	Percentage
Isolation Forest	5,677	13.14%
Neural Network	8,316	19.25%

Source: Developed by the Researcher using Experimental Results

Figure 12: OT Model Comparison

Model Comparison

Isolation Forest

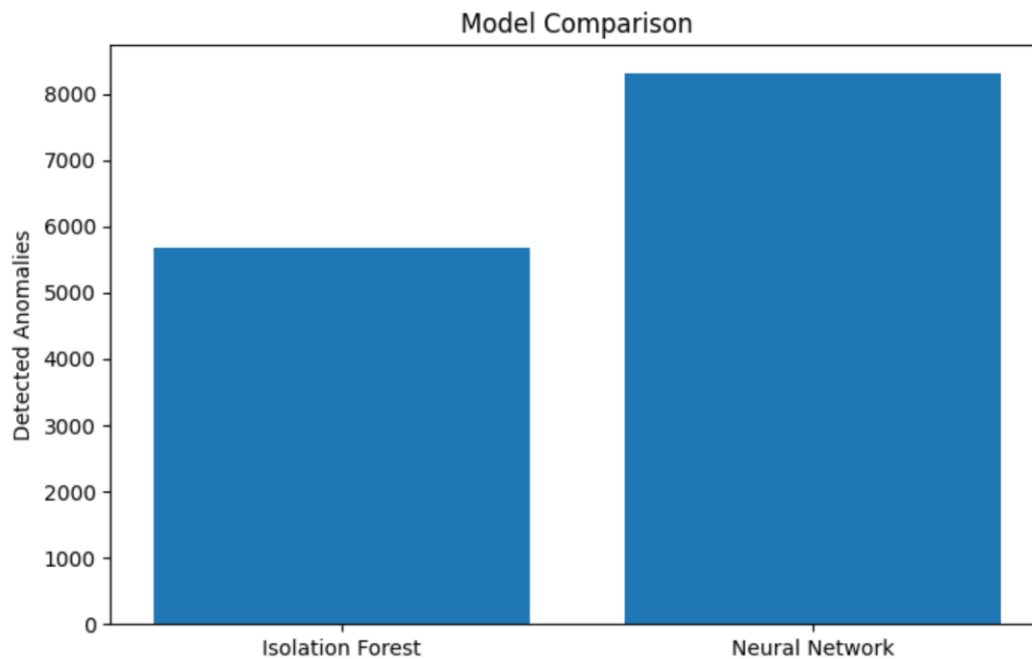
Detected Anomalies: 5677

Anomaly Percentage: 13.140899516214901

Neural Network

Detected Anomalies: 8316

Anomaly Percentage: 19.249554408462767



Source: Developed by the Researcher using Jupyter Notebook Output

The comparison shows that the Neural Network detected 2,639 more anomalous events than Isolation Forest, which is about 46.5% more anomalous events. This means that the Neural Network was more sensitive to out-of-normal operational conditions and was able to detect more deviations in the industrial process environment.

The Neural Network was more capable of detecting the fraud, but Isolation Forest was able to offer some practical benefits. The model was less resource intensive, training times were reduced and the model was easier to implement. Moreover, interpretable anomaly scores were produced by Isolation Forest, which can be helpful for investigation and forensic analysis.

The results indicate that both models were effective in identifying abnormal industrial behaviour. Isolation Forest was lightweight and efficient; the Neural Network was better at picking up on complex behaviour patterns and was more suited for operational use.

Overall, the anomaly detection experiments met the second research objective of determining the feasibility of applying machine learning and deep learning techniques to identify anomalous behaviour in OT environments. The results indicate the proposed framework could be a viable solution for the security monitoring of OT systems and play a role in enhancing the cyber resilience of critical infrastructure systems.

4.4 Digital Forensic Logging System Implementation:

4.4.1 Forensic Evidence Repository:

One of the primary goals of the proposed framework was to improve the readiness of the digital forensic by automatically logging and storing security events for future analysis. For this purpose, a MySQL database-based forensic evidence repository was designed and coupled with the machine learning detection modules.

The repository is a centralized storage solution for security and/or OT related information created during ransomware detection and/or OT anomaly detection activities. Each time a threat or anomaly is detected, the automated system can capture all needed information within the database automatically. This method not only saves time in collecting evidence manually, but it also minimizes the risk of losing vital evidence in incident response efforts.

The database includes data such as threat classification, threat time, threat confidence, severity, affected modules, investigation notes. These logs give investigators a comprehensive, organized place to find evidence to analyze security incidents and reconstruct the timeline of attacks.

Figure 13: MySQL Forensic Evidence Repository

id	timestamp	threat_type	severity	prediction	status	records_processed	notes
1	2026-05-11 20:35:05	Ransomware Detection	MEDIUM	56.61	RANSOMWARE DETECTED	62485	36375 malicious records detected
2	2026-05-12 18:40:57	Ransomware Detection	MEDIUM	56.61	RANSOMWARE DETECTED	62485	36375 malicious records detected
3	2026-05-12 18:41:22	OT Anomaly Detection	MEDIUM	31.8	OT ANOMALIES DETECTED	43201	13650 anomalies identified
4	2026-05-12 18:48:33	Ransomware Detection	LOW	28.67	SYSTEM SAFE	10	0 malicious records detected
5	2026-05-12 18:48:36	Ransomware Detection	LOW	28.67	SYSTEM SAFE	10	0 malicious records detected
6	2026-05-12 18:48:46	Ransomware Detection	LOW	28.67	SYSTEM SAFE	10	0 malicious records detected
7	2026-05-19 22:21:31	Ransomware Detection	MEDIUM	56.61	RANSOMWARE DETECTED	62485	36375 malicious records detected
8	2026-05-19 22:22:00	Ransomware Detection	MEDIUM	56.61	RANSOMWARE DETECTED	62485	36375 malicious records detected
9	2026-05-19 22:23:27	Ransomware Detection	LOW	28.67	SYSTEM SAFE	10	0 malicious records detected
10	2026-05-19 22:23:42	Ransomware Detection	LOW	28.67	SYSTEM SAFE	10	0 malicious records detected
11	2026-05-19 22:23:51	Ransomware Detection	LOW	28.67	SYSTEM SAFE	10	0 malicious records detected
12	2026-05-19 22:24:11	Ransomware Detection	LOW	28.67	SYSTEM SAFE	10	0 malicious records detected
13	2026-05-19 22:25:03	Ransomware Detection	MEDIUM	56.61	RANSOMWARE DETECTED	62485	36375 malicious records detected
14	2026-05-19 22:25:42	Ransomware Detection	MEDIUM	56.61	RANSOMWARE DETECTED	62485	36375 malicious records detected
15	2026-05-19 22:25:44	Ransomware Detection	MEDIUM	56.61	RANSOMWARE DETECTED	62485	36375 malicious records detected
16	2026-05-19 22:25:45	Ransomware Detection	MEDIUM	56.61	RANSOMWARE DETECTED	62485	36375 malicious records detected
17	2026-05-19 23:12:03	Ransomware Detection	LOW	28.67	SYSTEM SAFE	10	0 malicious records detected
18	2026-05-19 23:05:29	OT Anomaly Detection	LOW	5	OT ANOMALIES DETECTED	216001	10800 anomalies identified
19	2026-05-19 23:52:59	Ransomware Detection	MEDIUM	56.61	RANSOMWARE DETECTED	62485	36375 malicious records detected
20	2026-05-19 23:53:33	Ransomware Detection	MEDIUM	56.61	RANSOMWARE DETECTED	62485	36375 malicious records detected

Source: Developed by the Researcher

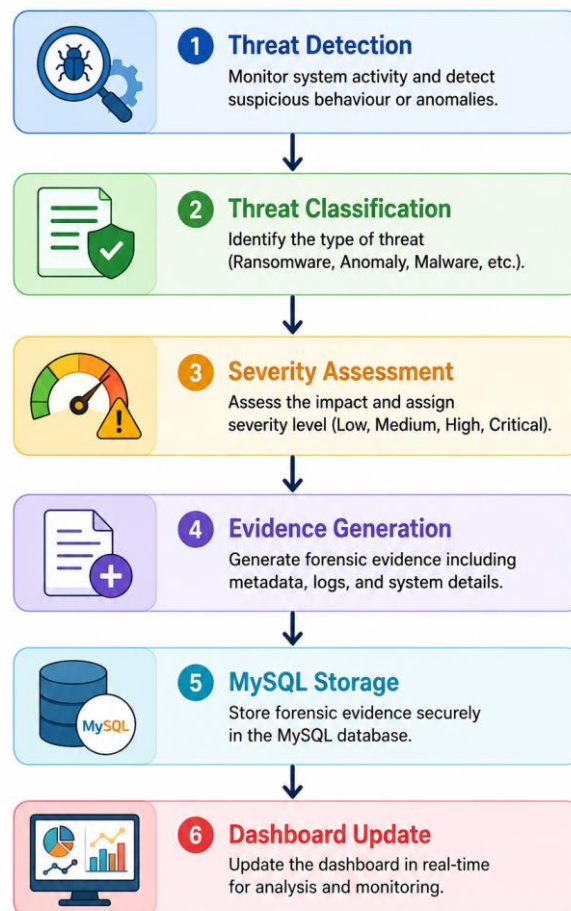
A separate forensic repository will provide immediate preservation of evidence, aiding digital forensic investigations and organisational incident response procedures upon threat detection.

4.4.2 Automated Evidence Logging Process:

The forensic logging process was designed to automatically run whenever the detection models detected suspicious activity. After a prediction, the framework categorises the threat, determines the confidence factor, gives the threat a level of severity and updates the information in the forensic database.

The automated logging mechanism guarantees consistency in evidence collection, thereby decreasing manual involvement. This is especially relevant in critical infrastructure where a significant number of security events could be generated in a very short time.

Figure 14: Automated Forensic Logging Workflow



Source: Developed by the Researcher

The workflow shows how threat information is submitted from detection modules to the forensic repository. The framework automatically generates evidence records, which help to foster forensic readiness and increase the information available for future investigations.

4.4.3 Contribution to Digital Forensic Readiness:

Compared with just detection systems, automated forensic logging offers several advantages. First, it is evidence is conserved at the time of detection, thereby minimising the chances of information loss. Secondly, investigators are given structured and searchable records which can assist in incident reconstruction. Third, the repository means that information on security events can be stored for the long term and trends can be analysed across multiple incidents.

The implementation proves machine learning-based detection can be effectively integrated with forensic evidence management in one package. This integration helps to fill one of the important research gaps identified in Chapter 2, which is the absence of solutions involving intelligent threat detection with digital forensic readiness.

Overall, the forensic logging subsystem successfully provides for the preservation, management and retrieval of digital evidence and is an important element of the proposed framework.

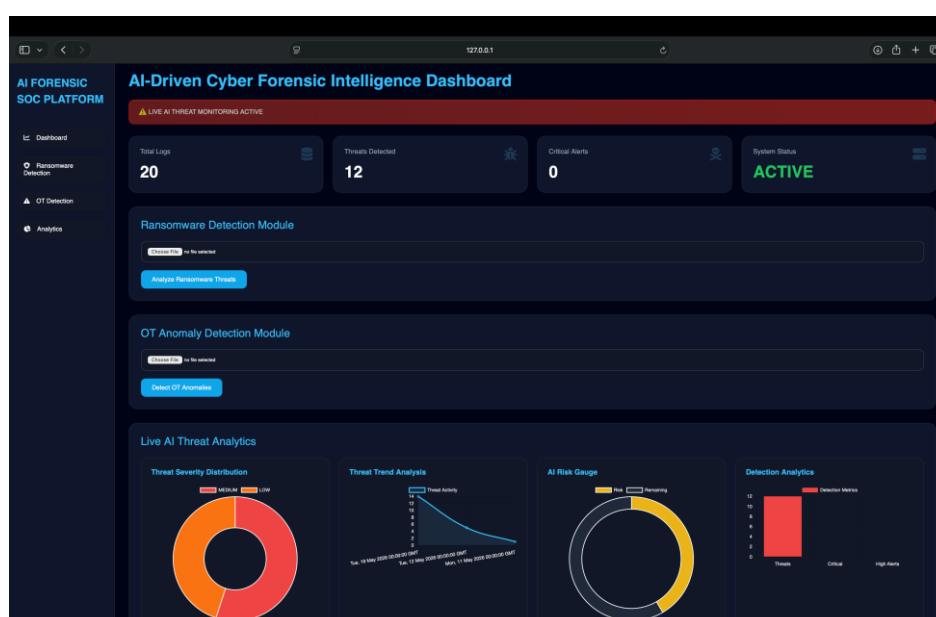
4.5 Flask Dashboard Implementation:

4.5.1 AI-Driven Cyber Forensic Intelligence Dashboard:

The last part of the proposed framework was creating an AI-Driven Cyber Forensic Intelligence Dashboard in Flask. The dashboard serves as a Security Operations Center (SOC) interface, allowing security analysts to monitor threats, review forensic information, and evaluate system conditions from a single interface.

Outputs from the ransomware detection module, OT anomaly detection module and forensic logging repository are all integrated into the dashboard. As the information produced by these components, it is presented in real-time, giving users a more unified perspective of the cybersecurity landscape.

Figure 15: AI-Driven Cyber Forensic Intelligence Dashboard



Source: Developed by the Researcher

The dashboard provides important operational data such as:

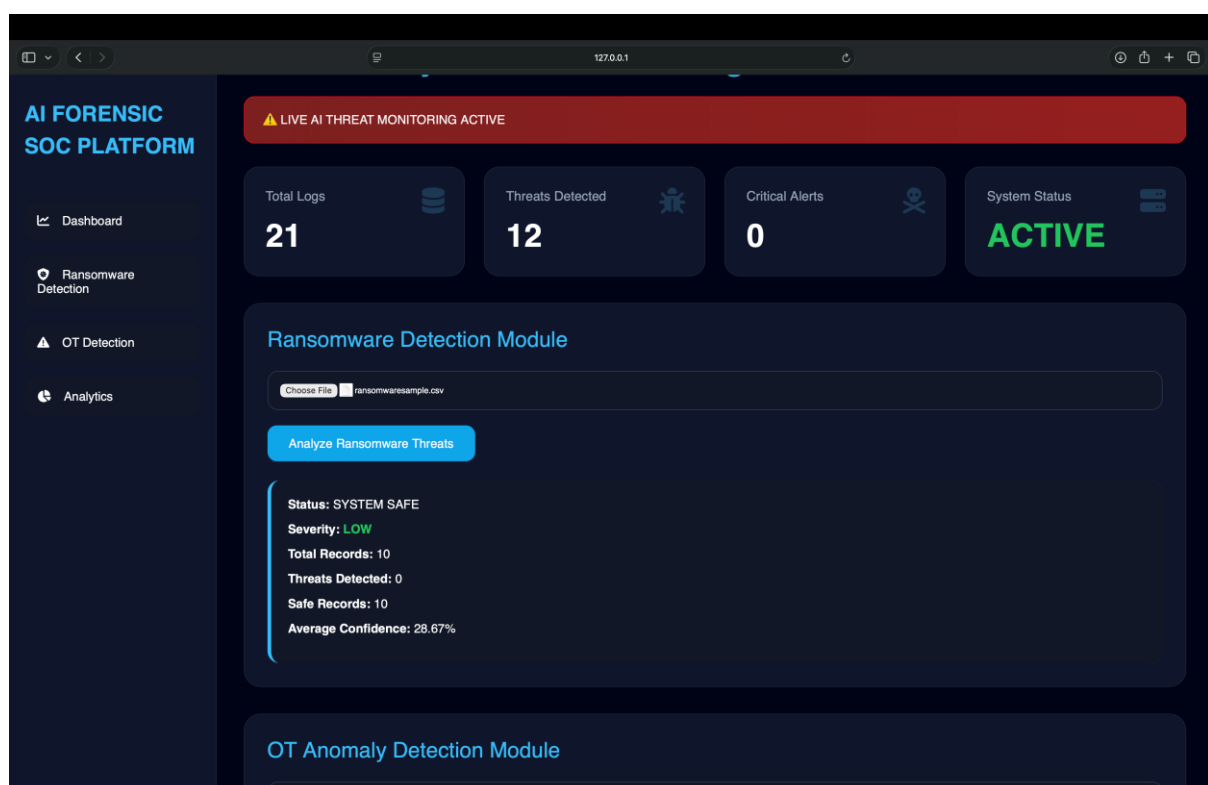
- Total Logs
- Threats Detected
- Critical Alerts
- System Status

During evaluation the Dashboard showed 20 recorded security events, 12 threats detected, no critical alerts, and an active system status. These metrics offer on-the-spot awareness and help security staff recognize new threats.

4.5.2 Ransomware Detection Module:

The module of ransomware detection was added as a part of the dashboard interface. Users can upload ransomware datasets and run an analysis via the machine learning detection engine.

Figure 16: Ransomware Detection Module



Source: Developed by the Researcher

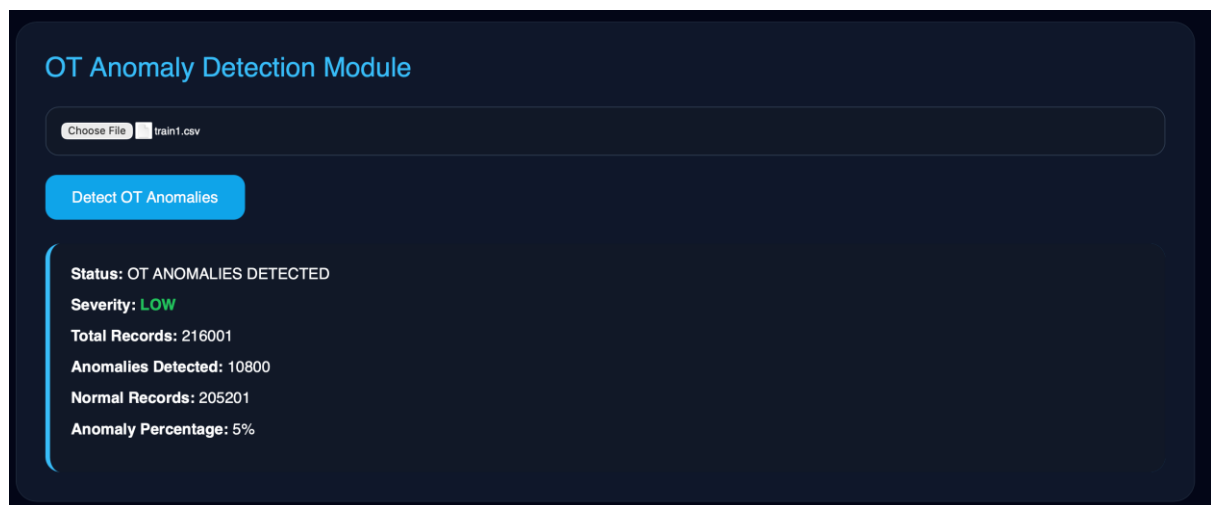
After upload of the files, the Random Forest model processes the data and provides classification results. Detected threats are automatically noted in the forensic

repository and presented in the dashboard statistics. This functionality offers an easy and effective way to do ransomware analysis and preserve forensic evidence.

4.5.3 OT Anomaly Detection Module:

The dashboard was also enriched with a dedicated OT anomaly detection module. This component allows for upload of industrial process datasets and for conducting anomaly analysis based on the trained anomaly detection models.

Figure 17: OT Anomaly Detection Module



Source: Developed by the Researcher

The module analyses uploaded data, detects anomalies and sends summary results to security analysts. By integrating OT monitoring capabilities into the dashboard, the framework can also address traditional IT security risks and Operational Technology anomalies from a single point of access.

4.5.4 Live Threat Analytics:

The dashboard features a live threat analytics section that lets you automatically see security events in real-time in the monitored environment.

Figure 18: Live Threat Analytics Interface



Source: Developed by the Researcher

The analytics component is responsible for consolidating data from both detection modules and visualising threat activity on the dashboard metrics and status indicators. This capability allows for better visibility and helps to quickly identify when something is amiss and needs further investigation

4.5.5 Dashboard Functionality Assessment:

This Flask dashboard met its goals of offering a centralized monitoring solution for cybersecurity operations. The interface integrates threat detection, anomaly monitoring, preservation of forensic evidence and security analytics in one place.

The usability of the dashboard was effective, its responsive performance was okay, and it was easy to present threat information. Moreover, integration with the forensic logging repository guaranteed that all events that are detected are kept for future analysis. These capabilities help to enhance situational awareness and provide informed decision-making when cybersecurity incidents occur.

4.6 Evaluation Against Research Objectives:

The outcomes of the proposed framework were compared with the research objectives presented in Chapter 1 to determine effectiveness.

The results support that all the research objectives were met. The framework provided successful ransomware classification, detection of anomalies in the OT, automatic preservation of forensic evidence and monitoring of threats via dashboard.

Table 12: Evaluation of Research Objectives

Research Objective	Outcome
Develop a ransomware detection model	Achieved
Develop an OT anomaly detection model	Achieved
Implement automated forensic logging	Achieved
Develop a real-time monitoring dashboard	Achieved
Improve digital forensic readiness	Achieved

Source: Developed by the Researcher

4.7 Discussion of Findings:

The results from the experimental assessment confirm that a combined approach of machine learning, anomaly detection, forensic preparedness, and dashboard monitoring in a unified cybersecurity framework is effective.

The Random Forest model performed best for ransomware classification, with an accuracy of 99.66%, which was higher than Logistic Regression and Gradient Boosting. The result shows that ensemble learning techniques work very well to detect malicious executable files by analyzing PE header features.

In the case of OT anomaly detection, both models, Isolation Forest and Neural Network, were able to classify the industrial behaviour as abnormal. The Neural Network was more sensitive in detecting the anomalies with a score of 8316 anomalies detected, while the Isolation Forest detected 5677 anomalies. But Isolation Forest has lower computational complexity and deployment efficiency.

One of the main drawbacks noted in previous studies was the lack of automated forensic logging. Instead of just threat detection, the framework preserved evidence

automatically at the place and time of detection, to facilitate future evidence collection and incident response.

Real-time visibility of detected threats, forensic records, system status information etc. added to the Flask dashboard making operations more user-friendly. This integration changed the nature of individual security components into a combined Security Operations Center platform.

These are the strengths, but it's also important to identify some of the limitations. The ransomware detection was based mainly on the static PE header features and did not feature any dynamic behavioural analysis. Additionally, the evaluation was performed on publicly available datasets and not in real industrial environments. Researchers might thus wish to investigate real-time deployment and behavioural ransomware analysis to further enhance ransomware detection.

The results overall validate the proposed approach of combining threat detection, anomaly detection, forensic application of evidence, and operational monitoring into one architecture.

4.8 Summary:

In this chapter, the implementation and evaluation of the proposed AI-Driven Digital Forensic Readiness Framework were introduced. The accuracy of the ransomware detection module was good, with the highest accuracy being 99.66% for Random Forest. Both models, Isolation Forest and a Neural Network, were able to successfully detect abnormal industrial behaviour in the OT anomaly detection module. Security events were saved in a MySQL repository with automated forensic logging, so they could be captured for future investigation.

The Flask dashboard was created to integrate all the components of the framework; it offered real-time monitoring, threat visualisation, forensic evidence management features. The evaluation showed that all the research goals were met and that the framework is effective in assisting with cybersecurity monitoring and digital forensic preparedness.

The findings of the research, the limitation of the research and recommendations for future research are presented in the next chapter.

CHAPTER 5: CONCLUSION AND FUTURE WORK

5.1 Introduction:

The conclusion of the study is presented in this chapter, and the achievement of the proposed objectives is discussed. The study was mainly about creating an AI-Driven Digital Forensic Readiness Framework that can identify ransomware threats, uncover anomalies in OT environments, capture digital forensic evidence, and monitor the environments in real time using a Security Operations Center (SOC) dashboard. The chapter reviews the main results, the contribution of the research, the limitations found and suggestions for further research.

5.2 Summary of Research Findings:

The main goal of this work was to create a cybersecurity framework that integrates the technologies of AI, anomaly detection, digital forensic readiness and threat visualization in a single application. To this end, machine learning and deep learning algorithms were applied and analyzed with publicly available ransomware and industrial control system datasets.

Three machine learning algorithms, namely logistic regression, gradient boosting, and random forest were tested for ransomware detection. The experimental results showed that the highest classification accuracy was obtained by the Random Forest with an accuracy of 99.66%, precision of 99.63%, recall of 99.76% and F1 score of 99.70%. The results of these experiments proved that ensemble learning techniques have great potential to recognize ransomware from Portable Executable (PE) header features.

In the context of Operational Technology anomaly detection, two models were used: Isolation Forest and Neural Network with the HAI dataset of industrial control systems. The Isolation Forest detected 5,677 anomalies and the Neural Network detected 8,316 anomalies. The results revealed that both models could detect an abnormal industrial behavior, the Neural Network having a higher detection sensitivity and the Isolation Forest a better computational efficiency.

Additionally, the study achieved a highly effective MySQL based forensic logging repository that could automatically store evidence relating to threats. The functionality helps with digital forensic investigations by allowing security events to be logged at

the point of detection. Furthermore, a Flask-based AI Cyber Forensic Intelligence Dashboard was created to offer real-time monitoring of threats, management of forensic evidence and operational visibility

5.3 Achievement of Research Objectives:

The research questions formulated at the start of the study were answered. The primary aim, that is, the creation of a ransomware detection system based on the AI algorithms, was achieved by implementing the Random Forest classifier. To achieve the primary aim, the development of an AI-based ransomware detection system, the Random Forest classifier was implemented. To accomplish the second aim, which was the detection of OT anomalies, the models used were Isolation Forest and Neural Network.

The third goal was to enhance digital forensic readiness by automated evidence preserving. This was done with the help of the MySQL forensic logging repository. The final goal, real time monitoring and threat visibility, was accomplished by creating the Flask-based dashboard.

Together, these components created a comprehensive solution that could be used for both monitoring and forensics in critical infrastructure.

5.4 Research Contributions:

This research makes significant contributions to the realms of cybersecurity and digital forensics by suggesting and developing a holistic framework that incorporates threat identification, anomaly detection, evidence collection, and dashboard monitoring. Many studies have concentrated on the individual cybersecurity tasks. This research shows a more different approach; by showing the multiple security capabilities can be integrated into a single operational platform.

The framework offers a practical solution to strengthen cyber resilience and digital forensic capabilities in an IT and OT co-existence scenario. Moreover, the study highlights the capabilities of AI methods in assisting cyber security operations.

5.5 Limitations of the Study:

Although the framework has been successfully implemented, there are some limitations to note. The first is that the ransomware detection was based on mostly static PE header features, with no dynamic behavioural analysis capabilities. Secondly, the experiments were performed on publicly available datasets not in live industrial environments. Therefore, the outcomes should not necessarily reflect actual operating conditions.

Furthermore, the sensitivity of the Neural Network in detecting anomalies was higher, but further studies would be needed to determine possible false positive rates. The framework was also tested in controlled research setting and large-scale deployment was not part of the research.

5.6 Recommendations for Future Work:

The framework could be further expanded with the addition of dynamic malware analysis techniques and behavioural monitoring to enhance the ransomware detection capabilities. Other machine learning/deep learning architectures like Long Short-Term Memory (LSTM) networks, Autoencoders, Transformer based networks could also be considered for OT anomaly detection.

The inclusion of threat intelligence feeds, automated incident response capabilities, and cloud-based forensic repositories could further augment the capabilities of the framework. The framework can also be tested using real industrial environments in future research to ensure that it is scalable, robust, and effective in practical applications.

5.7 Conclusion:

Finally, based on the research findings, an AI-Driven Digital Forensic Readiness Framework was designed and assessed that incorporates ransomware detection, OT anomaly detection, automated forensic evidence preservation, and real-time monitoring. The experimental results showed excellent ransomware classification performance and the ability to identify abnormal industrial behaviour. Forensic logging and dashboard monitoring further enhanced the usefulness of the framework.

In conclusion, the study highlights the potential of AI to contribute to the advancement of cyber resilience, security investigations, and security operations in critical infrastructure settings. The suggested framework offers a basis for future studies and implementation of the intelligent cyber forensic systems.

REFERENCES:

Albshaier, L., et al. (2024) 'Earlier Decision on Detection of Ransomware Identification: A Systematic Review', Information, 15(8). Available at: <https://www.mdpi.com/2078-2489/15/8/484>

Digital Forensics in Ransomware Analysis for Windows-Based Computer Systems (2024) ResearchGate Publication. Available at: https://www.researchgate.net/publication/396531008_Digital_Forensics_in_Ransomware_Analysis_for_Windows-Based_Computer_Systems

European Union Agency for Cybersecurity (ENISA) (2024) ENISA Threat Landscape 2024. Available at: <https://www.enisa.europa.eu/publications/enisa-threat-landscape-2024>

HAI Dataset (2023) HIL-Based Augmented ICS Security Dataset. GitHub. Available at: <https://github.com/icsdataset/hai>

IBM (2024) IBM X-Force Threat Intelligence Index 2024. Available at: <https://www.ibm.com/reports/threat-intelligence>

Kritika (2025) 'A Comprehensive Literature Review on Ransomware Detection Using Deep Learning', Cyber Security and Applications. Available at: <https://www.sciencedirect.com/science/article/pii/S2772918424000444>

Mekala, S.H. (2025) 'A Comprehensive Digital Forensic Readiness Framework for IIoT Environments', IEEE IT Professional. Available at: <https://www.computer.org/csdl/magazine/it/2025/03/11029707>

Mohamed, N. (2025) 'Artificial Intelligence and Machine Learning in Cybersecurity', Knowledge and Information Systems. Available at: <https://link.springer.com/article/10.1007/s10115-025-02429-y>

Mondal, B., et al. (2025) 'Machine Learning for Early Detection of Ransomware within Networks', Saudi Journal of Engineering and Technology, 10(4), pp. 159-168. Available at: https://saudijournals.com/media/articles/SJEAT_104_159-168.pdf

Pekepok, A., et al. (2025) 'Hierarchical Pattern Decryption Methodology for Ransomware Detection', arXiv preprint. Available at: <https://arxiv.org/abs/2501.15084>

Ransomware Detection Dataset (2023) Kaggle Dataset. Available at: <https://www.kaggle.com/datasets/amdj3dax/ransomware-detection-data-set>

Wang, X., et al. (2024) 'CanCal: Real-Time Lightweight Ransomware Detection for Industrial Environments', arXiv preprint. Available at: <https://arxiv.org/abs/2408.16515>

Khowla, K., Rahim, N.Z.A., Hamid, K., Ibrar, M., Ahmad, U. and Ullah, M.U. (2024) 'Windows forensic analysis and detection of ransomware attacks using event logs and tools', Proceedings of the 4th International Conference on Intelligent Technologies (CONIT 2024). IEEE. Available at: <https://ieeexplore.ieee.org/document/10626973>

McDonald, G. and Pat, P. (2022) 'Ransomware: Analysing the impact on Windows Active Directory domain services', Sensors, 22(3), 953. Available at: <https://www.mdpi.com/1424-8220/22/3/953>

Garg, M., Singh, S., Tiwari, S., Kaushik, K., Kumawat, R. and Soni, M. (2025) 'Adaptive cybersecurity frameworks: Leveraging artificial intelligence for proactive threat detection and forensic readiness', Proceedings of the 2nd International Conference on Intelligent Systems for Cybersecurity (ISCS 2025). IEEE. Available at: <https://ieeexplore.ieee.org/document/11386404>

Urooj, U., Maarof, M.A.B. and Al-rimy, B.A.S. (2021) 'A proposed adaptive pre-encryption crypto-ransomware early detection model', Proceedings of the 3rd International Cyber Resilience Conference (CRC 2021). IEEE. Available at: <https://ieeexplore.ieee.org/document/9392548>

Afaq, M., Rahmani, J. and Sikora, A. (2025) 'A comprehensive review of public datasets for machine learning-based intrusion detection in IoT and OT networks', Proceedings of the 13th International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS 2025). IEEE. Available at: <https://ieeexplore.ieee.org/document/11321996>

Khaliq, K., Rahim, N.Z.A., Hamid, K., Ibrar, M., Ahmad, U. and Ullah, M.U. (2024) 'Ransomware attacks: Tools and techniques for detection', Proceedings of the 2nd International Conference on Intelligent Systems. IEEE. Available at: <https://ieeexplore.ieee.org/document/10532926>

Manavi, F. and Hamzeh, A. (2020) 'A new method for ransomware detection based on PE header using convolutional neural networks', Proceedings of the 17th International ISC Conference on Information Security and Cryptology (ISCISC 2020). IEEE. Available at: <https://ieeexplore.ieee.org/document/9261903>

Lee, S.-J., Shim, H.-Y., Lee, Y.-R., Park, T.-R., Park, S.-H. and Lee, I.-G. (2022) 'Study on systematic ransomware detection techniques', Proceedings of the 24th International Conference on Advanced Communication Technology (ICACT 2022). IEEE. Available at: <https://ieeexplore.ieee.org/document/9728909>

Cosnek, M., Lutkus, C. and Barragan, D.E. (2024) 'The crucial role of cybersecurity in operational technology', Proceedings of the IEEE IAS Pulp and Paper Industry Conference (PPIC 2024). IEEE. Available at: <https://ieeexplore.ieee.org/document/10902235>

Chen, T. (2025) Recent progress on financial fraud detection using machine learning. Journal of Risk and Financial Management. Available at: <https://www.mdpi.com/1911-8074/18/1/12>

Yang, H. (2026) A review of artificial intelligence for financial fraud detection. Applied Sciences. Available at: <https://www.mdpi.com/2076-3417/16/2/455>

Wang, X., et al. (2024) CanCal: Real-time lightweight ransomware detection for industrial environments. arXiv. Available at: <https://arxiv.org/abs/2408.16515>

Garg, M., Singh, S., Tiwari, S., Kaushik, K., Kumawat, R. and Soni, M. (2025) Adaptive cybersecurity frameworks: Leveraging artificial intelligence for proactive threat detection and forensic readiness. IEEE Conference Proceedings. Available at: <https://ieeexplore.ieee.org/document/11386404>