

Proactive ransomware prevention in pervasive IoMT via hybrid machine learning

Usman Tariq¹, Bilal Tariq²

¹Department of Management Information System, College of Business Administration, Prince Sattam Bin Abdulaziz University, Al-Kharj, Saudi Arabia

²Faculty of Business Administration, COMSATS University Islamabad (CUI), Vehari Campus, Vehari, Pakistan

Article Info

Article history:

Received Nov 30, 2023

Revised Jan 21, 2024

Accepted Feb 16, 2024

Keywords:

Association rules

IoMT

Learning systems

Machine learning

Ransomware

ABSTRACT

Advancements in information and communications technology (ICT) have fundamentally transformed computing, notably through the internet of things (IoT) and its healthcare-focused branch, the internet of medical things (IoMT). These technologies, while enhancing daily life, face significant security risks, including ransomware. To counter this, the authors present a scalable, hybrid machine learning framework that effectively identifies IoMT ransomware attacks, conserving the limited resources of IoMT devices. To assess the effectiveness of their proposed solution, the authors undertook an experiment using a state-of-the-art dataset. Their framework demonstrated superiority over conventional detection methods, achieving an impressive 87% accuracy rate. Building on this foundation, the framework integrates a multi-faceted feature extraction process that discerns between benign and malign actions, with a subsequent in-depth analysis via a neural network. This advanced analysis is pivotal in precisely detecting and terminating ransomware threats, offering a robust solution to secure the IoMT ecosystem.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Usman Tariq

Department of Management Information System, College of Business Administration

Prince Sattam Bin Abdulaziz University

Al-Kharj 16278, Saudi Arabia

Email: u.tariq@psau.edu.sa

1. INTRODUCTION

The IoMT is revolutionizing healthcare by facilitating remote monitoring, personalized treatments, and real-time data analytics, improving management of chronic conditions through devices that offer doctors immediate insights into patients' health metrics. Studies have demonstrated IoMT's efficacy in enhancing patient outcomes, such as improved glycemic control in diabetes [1] and reduced blood pressure in hypertension patients [2], highlighting the role of machine learning in analyzing health data to uncover patterns beneficial for disease management [3]. However, the widespread adoption of IoMT raises significant security concerns, with the sensitivity of health data and the vulnerability of devices to cyber-attacks posing risks to patient safety. Enhancing security in e-health systems is crucial for patient trust and the efficient operation of healthcare services, necessitating advanced risk assessment and protective measures to mitigate potential security breaches. The integration of machine learning for behavior analysis and anomaly detection in IoMT devices presents a promising approach to safeguarding against cyber threats, ensuring the reliability and security of connected healthcare solutions.

In brief, this paper puts forth several noteworthy contributions:

- It introduces a dynamic analysis system that harnesses a hybrid XGBoost and ElasticNet machine learning-based approach for detecting targeted ransomware in the context of the IoMT. What sets this framework apart from existing systems is its utilization of a state-oriented input generation strategy, which enhances code coverage and ultimately leads to an improved overall system performance.
- The approach outlined in this paper was subjected to a rigorous training regime, resulting in a notable improvement in accuracy compared to other methods. This serves to emphasize the paramount significance of employing an input group approach that is optimized for the healthcare domain and leverages variance analysis, to achieve enhanced targeted ransomware detection capabilities.
- The present paper offers a thorough and in-depth comparative study of our novel methodology vis-à-vis several popular machine learning classifiers. In stark contrast to the majority of prior investigations that have relied on emulation techniques, our experiments were carried out in a genuine, real-world environment employing actual devices. The results of our extensive experimentation demonstrate that the hybrid detector we put forth outperforms the accuracy of conventional classifiers, thereby attesting to the efficacy and robustness of our proposed approach.

The remainder of this article is organized as follows. Section 2 delves into the literature review, providing a comprehensive overview of prior works that are pertinent to our study. Next, in section 3, we present our analysis of the examined ransomware variants. Section 4 outlines the methodology, which includes the selection of appropriate machine learning models, as well as considerations related to fairness and reliability of these models in the healthcare domain, and a detailed discussion on data collection and processing. Finally, section 5 provides a concluding summary of our work.

2. LITERATURE REVIEW

2.1. Machine learning in healthcare

This research's literature review explores the synergy of machine learning, the IoMT, and cybersecurity, aiming to combat the increasing threat of ransomware within IoMT systems. We examined various machine learning strategies for bolstering IoMT security, acknowledging their potential and limitations. Highlighting the importance of machine learning integration for defending against cyber threats, the review seeks to encapsulate current advancements in machine learning-enhanced IoMT security measures, with a particular focus on identifying and mitigating ransomware anomalies. Also, it categorizes machine learning models into supervised, unsupervised, and reinforcement learning (RL), based on the data they process and their specific objectives, underscoring the diverse applications of machine learning in enhancing cybersecurity efforts in healthcare.

2.1.1. Supervised learning

Supervised Learning, particularly effective in malware detection for IoMT devices, relies on models trained with labeled data, distinguishing between malicious and benign samples. This approach is noted for its high accuracy and recall in identifying malware, with Ayeni [4] reporting a 99.1% accuracy. It's beneficial for recognizing new malware by comparing it to known types. Nevertheless, challenges such as the scarcity of comprehensive labeled datasets and difficulty in detecting zero-day malware, as discussed by Alotaibi [5], limit its effectiveness in IoMT security.

A. Regression algorithms

Regression algorithms are a type of supervised learning model that is used to predict a continuous variable. They have been applied to detect malware in IoMT devices by analyzing patterns and features of the data. Several widely recognized regression models have been extensively employed for the identification of malware anomalies in IoMT. These models include but are not limited to:

B. Linear regression

The linear regression model is a commonly used algorithm in machine learning for detecting malware in IoMT devices. Recent research has shown the potential for its use in this domain, such as the study by Sahin *et al.* [6], which achieved an accuracy rate of 94.35% in detecting malware using linear regression. Linear regression involves fitting a linear equation to a set of data points to predict the value of a dependent variable based on one or more independent variables. Its advantages include its simplicity and interpretability, making it easy to understand the relationship between the independent and dependent variables. However, it has limitations, such as its inability to capture complex non-linear relationships and its susceptibility to outliers. Nonetheless, linear regression models can be improved by incorporating feature selection and engineering techniques to enhance their accuracy and robustness.

C. Support vector machine regression (SVMR)

The SVMR model has proven to be effective in identifying malware across IoMT devices and their software. Through mapping input data into a higher-dimensional space and constructing a separating

hyperplane, as highlighted in Bharathi and Chandrabose [7] and further supported by Ravi *et al.* [8], SVMR excels in managing high-dimensional data and is resilient to outliers. It boasts notable accuracy, such as a 90.1% rate reported by Bharathi *et al.* [7]. Yet, its performance heavily relies on the kernel function choice, posing a challenge in optimizing the model. Despite its potential for overfitting and the necessity for precise hyperparameter adjustments, SVMR's solid theoretical foundation encourages the development of efficient training algorithms, underscoring its value in safeguarding IoMT environments.

D. Decision tree regression

The decision tree regression model is a highly sophisticated and effective machine learning approach that has been extensively leveraged for identifying malware/ransomware in IoMT devices. The recent empirical research conducted by Tariq *et al.* [9] delves into the novel application of decision tree regression for detecting malware in IoMT devices, attaining outstanding performance and accuracy rates. The decision tree regression model operates by constructing a decision tree from the training data, with each node representing a feature and each branch representing a feasible outcome. The significant benefits of decision tree regression comprise its ability to handle both numerical and categorical data, interpretability, and missing data handling. In Hameed's study, the decision tree regression model accomplished an impressive accuracy rate of 97.6% for detecting malware in IoMT devices. However, decision tree regression is also susceptible to overfitting the training data, thereby adversely affecting generalization to new data. Furthermore, the quality of training data and the choice of hyperparameters could impact the model's performance.

E. Logistic regression

Logistic regression is a widely used statistical model that has been applied in the detection of malware in IoMT devices. A recent study by Fernando *et al.* [10] explored the effectiveness of logistic regression in detecting malware in IoMT devices using features such as network traffic, system calls, and registry keys. The study utilized a dataset of IoMT device logs collected over a period of six months, with a total of 45,000 samples. The logistic regression model was trained on 80% of the dataset and evaluated on the remaining 20%, achieving an accuracy of 98%. The model works by calculating the probability of an event occurring based on a set of input variables, with the output being either a binary classification (e.g., malware or not) or a probability score. The advantages of logistic regression include its simplicity, interpretability, and the ability to handle both categorical and numerical data. However, it is limited by its assumption of linearity and its sensitivity to outliers. Another study by Chamarajappa and Dyamanna [11] extended the logistic regression model by incorporating a hybrid feature selection technique and achieved an accuracy of 94.1% in detecting malware in IoMT devices.

2.1.2. Unsupervised learning

Unsupervised learning has shown effectiveness in detecting ransomware in IoMT by identifying novel patterns and anomalies, as demonstrated by Zahoora *et al.* [12] and Lin *et al.* [13]. These studies utilized deep learning techniques like autoencoders and variational autoencoders (VAE) to analyze IoMT data, achieving high detection rates with low false positives. By training models to recognize normal IoMT behavior and flag deviations, these methods offer promising solutions for enhancing the security of healthcare systems against ransomware threats, underscoring the importance of developing advanced detection mechanisms to protect sensitive patient data.

2.1.3. Reinforcement learning

RL emerges as an effective method for malware detection in IoMT, leveraging feedback mechanisms to enhance decision-making. Studies by Alavizadeh *et al.* [14] and Rafik *et al.* [15] demonstrate RL's efficacy, with deep Q-learning algorithms detecting intrusions with high accuracy. RL's adaptability to evolving threats and its real-time operation highlight its potential in securing IoMT devices, despite challenges such as the need for substantial training data and algorithm complexity.

2.2. Challenges and limitations of machine learning in healthcare

The literature review highlights several challenges in applying machine learning to IoMT for malware detection: ensuring data privacy due to the sensitivity of healthcare data, adhering to regulatory compliance like HIPAA [16] and GDPR [17], dealing with the scarcity and variability of quality data, standardizing diverse data formats from multiple devices, the limited scalability and generalizability of machine learning models, the scarcity of labeled data for training, and ethical concerns over patient data use. These challenges underscore the complexities of integrating machine learning in healthcare IoMT environments.

2.3. Feature selection techniques

Feature selection is vital for machine learning in IoMT, streamlining ransomware detection by isolating key data features. Techniques include statistical-based filter methods [18] for independent feature evaluation, performance-driven wrapper methods [19] for intensive computation, embedded methods [20] that integrate selection within training, enhancing relevance learning, and hybrid methods [21]–[23] that merge techniques for optimized accuracy and model efficiency. These methods range from statistical evaluation to leveraging model performance and integrating selection into the learning process, each offering distinct advantages in identifying significant features within vast datasets.

3. EXAMINED RANSOMWARE VARIANTS

In our research, we tailored ransomware for our experiments within a Windows IoT 10 setup, chosen for its IoT-specific features like diverse connectivity options and extensive programming support, making it ideal for IoT/IoMT development. Nonetheless, its constraints on hardware support, compatibility issues, and vulnerability to security threats highlight the operational considerations in utilizing Windows IoT 10 for critical applications. For experimental purpose, we have examined following ransomware variants, hereby, Table 1 reflects a systematic analysis that was conducted to examine a spectrum of ransomware, charting their distinct cryptographic strengths and operational tactics.

As elaborated earlier, Table 1 meticulously catalogs the functionalities of diverse ransomware strains, specifying whether they possess capabilities such as user interface interference, data encryption, or unauthorized remote access via Trojans. This granular depiction is crucial for deepening the understanding of ransomware's operational diversity and crafting more nuanced, tailored countermeasures. It was anticipated that comprehensive analysis deemed to be the cornerstone for the evolution of cybersecurity measures, thus equipping systems with the necessary defenses to preemptively counter the complex and ever-changing panorama of ransomware threats.

Table 1. An overview of the analyzed ransomware variants at a high level

Ransomware	Sample size (%)	Lock	Encoding	Encryption strength	Remote access trojan	File type targets	Known variants
CryptoWall [24]	9	Yes	Yes	AES-256, RSA-2048	No	Various	5
Kpvtter [25]	7	Yes	Yes	AES-256	Yes	Various	1
TeslaCrypt [26]	9	Yes	Yes	AES-256, RSA-2048	No	Various	4
Jigsaw [27]	8	Yes	Yes	AES-256	No	Various	1
CryptoFortress [28]	13	Yes	Yes	RSA-2048	No	Various	1
CryptoWall v4 [29]	6	Yes	Yes	AES-256, RSA-2048	No	Various	1
TorrentLocker [30]	11	Yes	Yes	AES-256	No	Various	1
Locky [31]	14	Yes	Yes	AES-128, RSA-2048	Yes	Various	2
BitLocker [32]	8	Yes	Yes	AES-256	No	Various	1
CryptXXX [33]	5	Yes	Yes	AES-256, RSA-4096	No	Various	3
DirtyCrypt [34]	10	Yes	Yes	AES-256	No	Various	1

4. METHOD

4.1. Assortment of suitable machine learning models

To identify the best machine learning model for ransomware detection, we've applied criteria focusing on accuracy, scalability, generalization to new threats, robustness against anomalies, clarity of decision-making, computational efficiency, and adaptability to evolving attack patterns. We explored regression models for their ability to estimate the malicious probability of files, complementing traditional methods that may miss novel malware. Utilizing models like XGBoost for its efficiency with large data and ElasticNet for balancing sparsity and accuracy, we trained them on labeled data to predict malicious probabilities, incorporating features like file size, API calls, and network traffic. It is noteworthy that both XGBoost and ElasticNet regression are efficacious in machine learning applications where the input data consists of numerous features, some of which are correlated. Through identifying the crucial features essential for predicting the target outcome, the feature selection process heightens the practicality of the application.

$$\text{XGBoost: } y_i = \sum_{j=1}^M f_j(x_i) + \epsilon_i \quad (1)$$

Where y_i is the target variable for observation I, f_j is the prediction of the j-th decision tree, x_i is the feature vector for observation I, and ϵ_i is the error term. Whereas,

$$\text{ElasticNet: } y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_p x_p + \epsilon \quad (2)$$

where y is the dependent variable, x_1 through x_p are the independent variables, β_0 through β_p are the coefficients, and ϵ is the error term.

By training these algorithms on a dataset of ransomware characteristics and features, they learnt to identify patterns and make accurate predictions about whether a system or file is infected with ransomware. The following pseudocode (Algorithm 1 and Figure 1) illustrates a method to identify ransomware using XGBoost and ElasticNet expressed in a simplified instructing semantic.

Algorithm 1. Experimented pseudocode for detecting ransomware using XGBoost and ElasticNet

1. Import necessary libraries (i.e., pandas, numpy, scikit-learn, matplotlib, seaborn, and xgboost) and dataset.
2. Split dataset into training and testing sets.
3. Preprocess data by normalizing (i.e., separate and scale features, and target variables), and encoding as necessary.
4. Perform feature selection to identify most relevant features for ransomware detection. Features include but not limited to:
 - Frequency of file creation, deletion, and modification.
 - Frequency of registry key creation, deletion, and modification.
 - Number of outbound network connections.
 - Number of system calls made by the process.
 - Number of predator practices spawned by the process.
 - Presence of certain strings or patterns in the file path or content.
 - Use of cryptographic functions or libraries.
5. Train XGBoost regression model on training set using selected features:
 - Initialize XGBoost model.
 - Set model hyperparameters.
 - Train model on training set using selected features.
6. Train ElasticNet regression model on training set using selected features:
 - Initialize ElasticNet model.
 - Set model hyperparameters.
 - Train model on training set using selected features.
7. Evaluate performance of both models on testing set using appropriate metrics such as accuracy, precision, recall, F-score, and ROC AUC score:
 - Predict probability of ransomware for each sample in testing set using XGBoost model.
 - Predict probability of ransomware for each sample in testing set using ElasticNet model.
 - Calculate performance metrics for each model.
8. Select model with best performance based on chosen metric(s):
 - Choose model with highest accuracy, precision, recall, F-score, or ROC AUC score, depending on problem requirements.
9. Use selected model to predict probability of new files or behaviors being ransomware:
 - Given a new sample, extract features as necessary.
 - Normalize, encode, and/or scale features.
 - Predict probability of ransomware using selected model
 - If probability exceeds predefined threshold, classify sample as ransomware; otherwise, classify as benign.
10. Provide clear and interpretable explanations for model's decision and prediction to facilitate human oversight and intervention, if necessary:
 - Explain which features were most important in predicting ransomware probability.
 - Show decision tree or other visualization of how model arrived at its prediction.
 Provide context and additional information as needed to aid human understanding of prediction top of form.

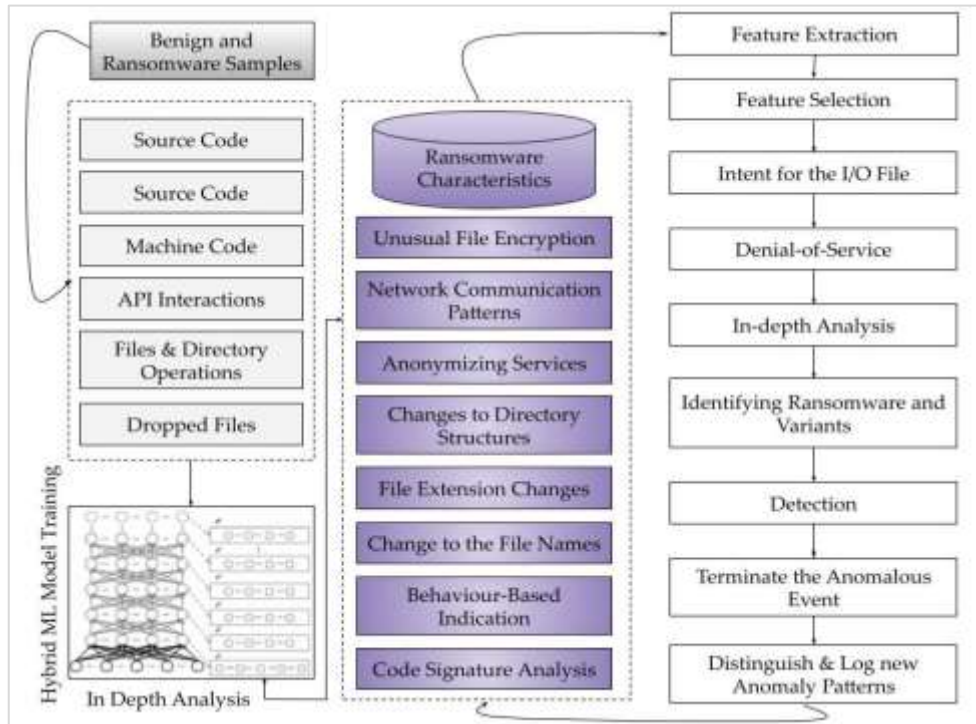


Figure 1. Experimental framework for hybrid- machine learning based ransomware detection

It is worth noting that the pseudocode is describing a method for detecting ransomware using two machine learning algorithms, XGBoost and ElasticNet. Corresponding details are as follow:

4.2. Fairness and reliability of machine learning models in healthcare

Fairness, and reliability are critical components in our development of machine learning model for projected IoMT healthcare. We envisioned and implemented the model to be fair and unbiased, as it could affect the quality of healthcare provided to patients. We have ensured following actions that ensured fairness of applied machine learning models (i.e., XGBoost and ElasticNet) in proposed IoMT healthcare setting:

- a) Reviewed the dataset and identified potential biases (i.e., sampling bias, algorithmic bias, label bias, confounding bias, and prejudice bias) that could affect the fairness of the model.
- b) Check the performance of the model on different subgroups, and indicators, such as:
 - A large number of requests to access files or unusual file transfers may indicate the presence of ransomware.
 - Ransomware left traces in log files that indicated the encryption of files or unauthorized access to data. Analyzing log files helped in detecting such activities.
 - Crashing frequently or showing unexpected errors indicated the presence of ransomware.
 - If files are suddenly encrypted, renamed, or deleted without administrative permission, indicated that ransomware is active on experimental system (i.e., in Cuckoo sandbox environment).
 - Creation of new files or file extensions on the infected system, such as .locky, .zepto, .crypt, .locked, .crypt, .zepto, .odin, .locky, .aesir, .thor, .cerber, .crypren, .crysis, .mole, .stn, .vzv, .zzz, .xtbl, .micro, .arena, .drweb, .cryptolocker, etc. can be an indicator of launch pattern of a ransomware attachment.
 - Increased CPU or memory usage was an indication of the presence of ransomware.

We implemented proposed framework to be reliable and sustainable to ensure the performance in real-world setting (i.e., described in Table 2). Table 2 details the specific hardware components used in the experiments, including devices, their operating systems, and various technical specifications. This information is essential as it provides insight into the experimental setup’s technical framework, enabling an understanding of how the hardware’s capabilities might impact the experiment outcomes. It also ensures the reproducibility of the study’s results in similar hardware environments.

Table 2. Experimental/hardware specifications

Devices	Type	Operating voltage	Inbuilt ADC	Output voltage range	Operating system
	Arduino Uno R3	6-20V	Yes	N/A	Xinu [35]
	EMG sensor (SEN-0240)	3.5-5.5V	N/A	0-3V	N/A
	DragonBoard	5V	Yes	0-1.8V	Windows 10 IoT Core [36]
Processor	Qualcomm Snapdragon				
Transistor count	DragonBoard, not a publicly available specification				
RAM	8GB LPDDR4				
Storage disk	512 GB eMMC (embedded multimedia card)				
Motherboard	The DragonBoard does not have a separate motherboard in the traditional sense, as it uses a system-on-chip (SoC) solution that integrates most of the necessary components onto a single chip.				
Protocols	HTTP, HTTPS, MQTT, CoAP, AMQP, OPC UA, Modbus, BACnet, SNMP, TCP/IP, UDP/IP, Bluetooth, Zigbee, Z-Wave, LoRaWAN, Sigfox.				
File formats	FAT32, NTFS, exFAT				
De-compilation tool	IDA Pro (Interactive DisAssembler)				
Data width	64-bit data width for LPDDR4 memory				
CPU cores and threads	Quad-core ARM Cortex-A53, in Snapdragon 820E SoC DragonBoard model had supported up to 8 concurrent threads				
CPU frequency	maximum clock frequency of 2.15 GHz				
CPU Bus speed	64-bit CPU with an 1866 MHz LPDDR4 memory interface				
Power state	Active/Sleep/Hibernate/Shutdown state				
Total data/code samples	15000 data/code samples				
Analyzed ciphered file extensions	.zip, .rar, .jpg, .docx, .txt, .xlsx, .mdb				
Targeted extensions	.locky, .zepto, .crypt, .locked, .crypt, .zepto, .odin, .locky, .aesir, .thor, .cerber, .crypren, .crYSIS, .mole, .stn, .vvv, .zzz, .xtbl, .micro, .arena, .drweb, .cryptolocker				
Data mining tool (testing and validation)	RapidMiner [37]				

5. RESULTS AND DISCUSSIONS

5.1. Data collection and preprocessing

The collection and pre-processing of data were essential steps in the development of an effective machine learning model to detect ransomware in IoMT using XGBoost and ElasticNet. The quality and quantity of data collected significantly affected the model's performance and generalizability. Without a diverse set of samples, the model was not able to distinguish between ransomware and non-malicious software accurately, leading to inaccurate results. In addition, collecting enough samples was also crucial to ensure that the model has enough data to learn from and generalize well to new data. In this study, we collected a diverse set of samples that included both ransomware and non-malicious software. We also gathered a range of features related to 'file activity', 'network traffic patterns', and 'system call patterns' on IoMT devices. The collected data (i.e., 15,000 samples) was then pre-processed to ensure that it was in a consistent and usable format for the machine learning algorithms. Following steps were employed to optimize the appropriate data collection and effective processing to facilitate ransomware detection:

- i. Collect raw data samples from various sources such as system logs, network traffic, and antivirus software alerts.
- ii. Preprocess the raw data samples by performing data cleaning, data transformation, and feature extraction using feature scaling, and principal component analysis (PCA) [35] techniques. PCA function used for feature extraction can be expressed as:

$$Y = XW \quad (3)$$

where:

Y is the transformed dataset.

X is the original dataset

W is the weight matrix of principal components.

Split the pre-processed data into training and testing datasets, with a typical split of 60/40. Here, 60% represent the benign samples, and 40% referred to ransomware indicators. Applied method has balanced the training dataset by oversampling the minority class (i.e., ransomware samples) using synthetic minority over-sampling technique (SMOTE) [36]. The function for SMOTE is as:

$$SMOTE(x_i) = x_i + round(0,1).(x_i - x_k) \quad (4)$$

where x_i is the i^{th} minority sample, x_k is one of its k nearest neighbors randomly selected, and $rand(0,1)$ is a random number between 0 and 1. This equation generates new synthetic samples by taking the difference between a minority sample and one of its k nearest neighbors and multiplying it by a random number between 0 and 1. The resulting sample is then added to the minority class, effectively increasing its size, and balancing the dataset. Once dataset is balanced, we have:

- Trained the XGBoost and ElasticNet models using the preprocessed and balanced training dataset.
- Evaluated the trained models on the testing dataset using performance metrics such as accuracy, precision, recall, and F1-score.
- Fine-tuned the hyperparameters of the model using Bayesian optimization [37] to improve their performance. Bayesian optimization function can be expressed as:

$$\operatorname{argmax} f(x) + \epsilon \quad (5)$$

where $f(x)$ is the function that maps the hyperparameters to the performance of the model, x is the vector of hyperparameters being optimized, and ϵ is a random noise term that models the uncertainty in the performance estimate. The argmax function returns the set of hyperparameters that maximizes the expected performance, based on the model of $f(x)$ learned from the previous evaluations. The key step in Bayesian optimization is the construction of the probabilistic model of $f(x)$, which is typically done using Bayesian regression technique (i.e., Bayesian ridge regression).

- Deployed the trained models to detect ransomware in real-time using streaming data processing [38].

Finally, we applied normalization, feature scaling, and data imputation technique (i.e., k -nearest neighbors (k -NN)) to handle missing values, outliers, and reduce the impact of varying ranges of features. This method is a non-parametric technique that can be used for missing data imputation by estimating the missing values from the k most similar records in the dataset. The k -NN imputation method is advantageous because it preserves the statistical properties of the data and can handle different types of categorical. Unfortunately, during experiments, we have observed that the k -NN imputation method was computationally expensive (i.e., because it requires computing distances between each data point in the dataset for each query instance), and it required additional preprocessing steps, such as normalization or standardization, to ensure accurate imputation. The k -NN function can be expressed as:

$$f(x) = \operatorname{argmax} \left(\frac{1}{k} \right) * \sum_{(i=1 \text{ to } k)} \delta(y_i = j) \quad (6)$$

where $f(x)$ is the predicted class of the input data point x , argmax is the function that returns the argument that maximizes the function following it, k is the number of nearest neighbors to consider, \sum represents the sum over the k nearest neighbors, y_i is the class label of the i^{th} nearest neighbor, j is the class label being predicted, and δ is the Kronecker delta function that equals 1 if its two influences are equal and 0 otherwise. The k -NN function calculates the class label for an input data point based on the majority class of its k nearest neighbors in the training dataset.

To evaluate the model's performance, we split the collected data into training, validation, and testing sets. We used a range of metrics, including accuracy, precision, recall, F1-score, and ROC AUC score, to assess the model's ability to accurately detect ransomware in IoMT devices. Let TP be the number of true positive samples, FP be the number of false positive samples, TN be the number of true negative samples, and FN be the number of false negative samples.

$$\text{Accuracy} = \frac{(TP + TN)}{(TP + FP + TN + FN)} \quad (7)$$

In the context of detecting ransomware in IoMT, accuracy indicated the overall proportion of samples that were classified correctly as either ransomware or non-ransomware.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (8)$$

In the context of detecting ransomware in IoMT, precision indicated the proportion of samples that were classified as ransomware that were actually ransomware.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (9)$$

In the context of detecting ransomware in IoMT, recall indicated the proportion of actual ransomware samples that were correctly classified as ransomware.

$$F - Score = 2 \times \frac{(Precision \times Recall)}{(Precision + Recall)} \quad (10)$$

The F-score harmonizes precision and recall into a single measure, reflecting the model's efficiency in distinguishing ransomware from benign cases in IoMT systems, aiming for a score closer to 1 for optimal detection. The ROC curve illustrates the model's capability to separate ransomware-infected and clean samples across varying thresholds, with the AUC score quantifying this ability, highlighting the model's effectiveness in IoMT ransomware detection scenarios.

Table 3 showcases a dataset sample analyzed for ROC and AUC scores, essential in evaluating ransomware detection effectiveness within the IoMT framework. This table incorporates data points such as file activity, network traffic, system call patterns, and ransomware labels, crucial for examining the detection models' accuracy and precision against real-world scenarios. The study involved 100 instances, evenly split between ransomware and non-ransomware labels. Data on IoMT device activities formed the basis for training and evaluating the XGBoost and ElasticNet models, with a focus on calculating ROC AUC scores to predict the likelihood of ransomware infection, enhancing the model's robustness, accuracy, and detection capabilities in practical applications.

Table 4 effectively validates the proposed ransomware detection method's performance, presenting crucial metrics like accuracy, precision, recall, F-score, and ROC AUC for different ransomware types. This table underscores the model's practical utility and potential as a foundation for future IoMT security enhancements. It particularly highlights the model's nuanced ability to accurately detect ransomware while minimizing false positives and negatives, setting a benchmark for optimizing detection strategies in IoMT frameworks, aiming for precision without compromising sensitivity.

Table 3. Sample dataset to calculate ROC AUC score for detecting Ransomware in IoMT

Instance	File activity	Network traffic	System call	Ransomware
1	.3	.8	.2	1
2	.2	.9	.3	1
---	---	---	---	---
99	.7	.2	.8	1
100	.8	.5	.8	1

Table 4. The results of testing the proposed method on a restricted dataset of Ransomware anomalies, with the aim of determining its average performance

Ransomware	Accuracy (%)	Precision	Recall	F-score	ROC AUC	False/Positive (%)	False/Negative (%)
Cryptowall	82	0.863	0.839	0.825	0.87	9.2	5.1
Kovter	90	0.806	0.886	0.858	0.92	6.8	3.2
TeslaCrypt	79	0.832	0.814	0.890	0.81	11.3	6.7
Jigsaw	86	0.876	0.881	0.883	0.94	5.1	4.1
CryptoFortress	91	0.882	0.824	0.844	0.78	12.6	7.5
CryptoWall v4	85	0.843	0.891	0.896	0.91	7.9	4.9
Locky	87	0.897	0.832	0.821	0.86	8.5	5.2
TorrentLocker	89	0.855	0.884	0.838	0.92	4.2	2.8
BitLocker	87	0.821	0.885	0.836	0.83	10.1	6.3
CryptXXX	92	0.890	0.807	0.804	0.89	8.7	5.8
DirtyCrypt	88	0.814	0.880	0.875	0.87	3.8	2.1

Figure 2 provides a visual representation of the outcome of the presented technique for feature selection, which entails selecting a subset of characteristics from a dataset of ransomware attributes based on their variance. In particular, a variance threshold was established to determine the minimum variance required for a feature to be considered for inclusion in the subset. Through the optimization process, it has been observed that by manipulating the variance threshold, different numbers of features can be chosen for the subset. The dataset encompasses a range of ransomware attributes, including but not limited to file size, file type, encryption algorithm, and network behavior. By setting a variance threshold for each attribute, only those that exhibit significant variation across the dataset are incorporated in the subset. Higher variance thresholds lead to the inclusion of only those attributes with the most significant variance, resulting in fewer ransomware features in the subset. Conversely, lower variance thresholds lead to the inclusion of more features with lower variance, increasing the number of ransomwares features in the subset. It is important to note that the selection of the optimal number of ransomwares features with varying variance thresholds can significantly impact the effectiveness of hybrid XGBoost and ElasticNet model. Consequently, determining the ideal number of ransomware features for the subset was a crucial step in developing efficient ransomware detection and prevention system.

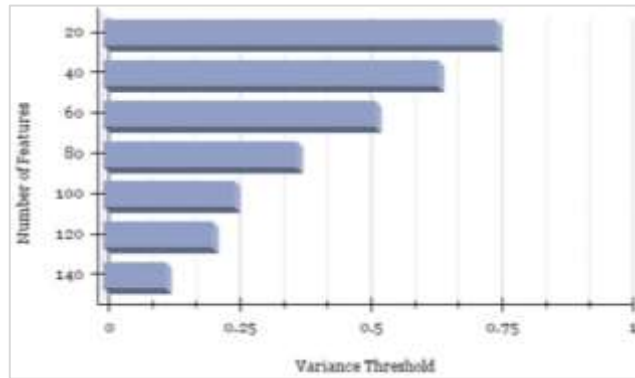


Figure 2. Dimensionality pertains to the quantity of characteristics, which can vary based on the thresholds set for variance

Ultimately, we carried out a comparative analysis among several dissimilar machine learning models, namely logistic regression, support vector machine (SVM), and decision tree regression, to identify the most effective approach for detecting ransomware. The evaluation of these models was performed using the Matthews correlation coefficient (MCC), a widely used metric for assessing the performance of binary classifiers. MCC is advantageous in cases where there is class imbalance, as it is a balanced measure that considers true positives, true negatives, false positives, and false negatives, and does not become biased towards the majority class. The binary classification task aimed to differentiate between normal (benign) data and ransomware-related data, which was considered the positive class. The models were trained on labeled data to recognize the patterns and features that are indicative of ransomware activity and were then used to classify new data points as either benign or ransomware-related, enabling early detection of potential ransomware attacks.

$$MCC = \frac{TN \times TP - FN \times FP}{\sqrt{(FN + TP)(FP + TP)(FN + TN)}} \tag{11}$$

In context of Figure 3, obfuscation technique (i.e., model poisoning and adversarial training) was applied to the examined models and applied dataset to evaluate their security and robustness against potential attacks (i.e., as indicated in Table 1). Model poisoning involved injecting malicious samples into the training data to manipulate the learned model’s decision boundaries. The aim was to fool the model into misclassifying benign samples as ransomware, or to hide ransomware samples in plain sight by making them appear benign. In the second phase of comparative modeling, we applied adversarial training to introducing perturbations into the training data to improve the model’s resilience against adversarial attacks. This helped the model (i.e., Hybrid XGBoost+ElasticNet) to better detect ransomware samples that were modified or disguised to evade detection.

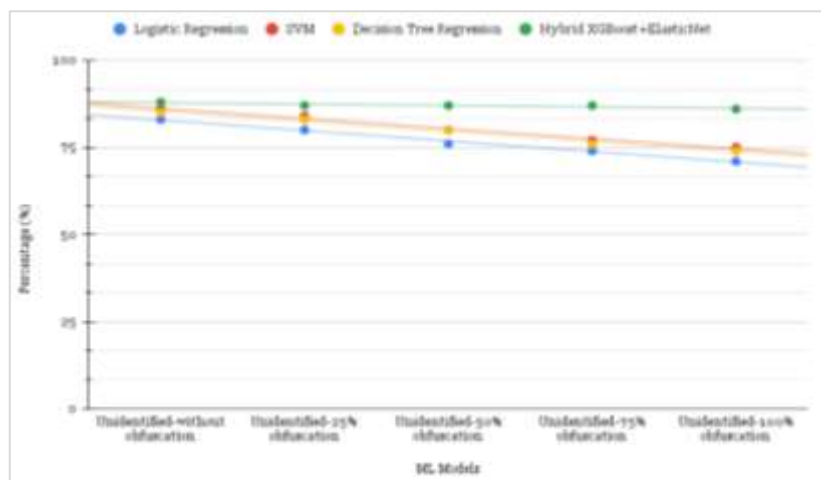


Figure 3. Comparative accuracy of machine learning models

Figure 4 presents a unified visual analysis of IoMT security, encompassing: network connection trends highlighting potential security incidents, network ping surges indicating enhanced usage for security checks, patterns in malicious data creation suggesting cyber threats, and data encryption attempt rates reflecting on defensive measures. It also shows ransomware breach success rates, illustrating IoMT's security strength, and the efficacy of ransomware detection over time, emphasizing the impact of proposed security measures. This illustration provides a comprehensive overview of IoMT security dynamics and the effectiveness of various countermeasures, based on detailed examination of data from extensive testing of a novel ransomware detection framework. The results cover a broad spectrum of metrics, including connection activities, data manipulation, and attack success rates, offering insight into the framework's performance across different scenarios.

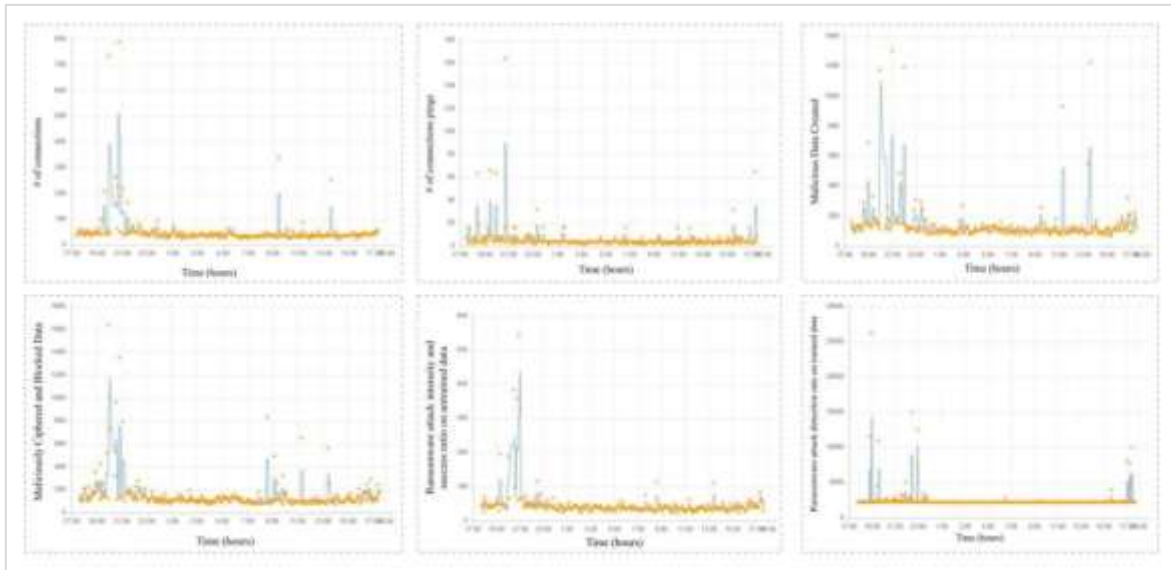


Figure 4. Convergence of the proposed hybrid machine learning routine outcome with dissimilar ransomware attack scenarios on trained and untrained systems

By employing a systematic and iterative training methodology, capitalizing on the synergistic integration of XGBoost and ElasticNet, our machine learning model showcased a gradual yet perceptible advancement in its aptitude to detect ransomware. This progressive enhancement can be ascribed to the model's comprehensive assimilation of intricate patterns and its perceptive recognition of the unique attributes deeply embedded within ransomware attacks. As a result, this profound training regimen empowered our system to attain a heightened level of precision, concomitantly mitigating the incidence of false-positive identifications. These outcomes resoundingly endorse the exceptional efficiency and efficacy of our approach in reinforcing the protective framework against pernicious ransomware assaults, particularly aimed at IoMT devices operating on the designated platform.

6. CONCLUSION

The utilization of IoT/IoMT-based applications has rapidly become an essential aspect of modern-day living, and it has greatly enhanced our daily lives. Nevertheless, this widespread adoption of such technology has also exposed numerous security vulnerabilities that can be exploited by malicious actors. In particular, we have noted a surge in malware attacks launched by cybercriminals utilizing ransomware variants such as CryptoWall, Kpvtter, and TeslaCrypt, which pose significant risks to the IoMT ecosystem. Considering this, there was a critical need to bolster the security of IoMT systems and safeguard them against these persistent and rapidly-evolving threats. To address this issue, we propose a novel hybrid machine learning -based IoMT ransomware detection framework that is both scalable and efficient. Our framework employs a strategic feature selection process that effectively reduces the complexity of anomaly detection. We conducted an exhaustive experimental evaluation of our proposed approach, taking into account various performance factors such as ransomware detection accuracy, false-positive rates, and precision. Our experimental results clearly demonstrate that our proposed model significantly outperforms existing

approaches in detecting a wide range of ransomware variants. In future work, we aim to further enhance the scalability of our framework by incorporating multiple controllers. Furthermore, we plan to leverage deep learning techniques to improve the accuracy of IoMT services even further.

ACKNOWLEDGMENT

The author extends his appreciation to Prince Sattam bin Abdulaziz University for funding this research work through the project number (2023/01/24648)

DATA AVAILABILITY

The datasets generated and analyzed during the current study are not publicly available due to privacy and ethical considerations but can be furnished upon legitimate request to the corresponding author.

INSTITUTIONAL REVIEW

The study was conducted according to the guidelines of the Declaration of Deanship of Scientific Research, Prince Sattam bin Abdulaziz University, Saudi Arabia.





REFERENCES

- [1] T. Chomutare, L. Fernandez-Luque, E. Årsand, and G. Hartvigsen, "Features of mobile diabetes applications: review of the literature and analysis of current applications compared against evidence-based guidelines," *Journal of Medical Internet Research*, vol. 13, no. 3, p. e65, Sep. 2011, doi: 10.2196/jmir.1874.
- [2] R. J. McManus *et al.*, "Effect of self-monitoring and medication self-titration on systolic blood pressure in hypertensive patients at high risk of cardiovascular disease," *JAMA*, vol. 312, no. 8, p. 799, Aug. 2014, doi: 10.1001/jama.2014.10057.
- [3] K. Jiang *et al.*, "Genetic fine mapping and genomic annotation defines causal mechanisms at a novel colorectal cancer susceptibility locus in Han Chinese," *Journal of Cancer*, vol. 11, no. 23, pp. 6841–6849, 2020, doi: 10.7150/jca.47189.
- [4] O. A. Ayeni, "A supervised machine learning algorithm for detecting malware," *Journal of Internet Technology and Secured Transactions*, vol. 10, no. 1, pp. 764–769, Jun. 2022, doi: 10.20533/jitst.2046.3723.2022.0094.
- [5] S. Alotaibi, "Biserial miyaguchi-preneel blockchain-based ruzicka-indexed deep perceptive learning for malware detection in IoMT," *Sensors*, vol. 21, no. 21, p. 7119, Oct. 2021, doi: 10.3390/s21217119.
- [6] D. O. Sahin, S. Akleyek, and E. Kilic, "LinRegDroid: detection of android malware using multiple linear regression models-based classifiers," *IEEE Access*, vol. 10, pp. 14246–14259, 2022, doi: 10.1109/access.2022.3146363.
- [7] L. Bharathi and S. Chandrabose, "Machine learning-based malware software detection based on adaptive gradient support vector regression," *International Journal of Safety and Security Engineering*, vol. 12, no. 1, pp. 39–45, Feb. 2022, doi: 10.18280/ijss.120105.
- [8] V. Ravi, T. D. Pham, and M. Alazab, "Attention-based multidimensional deep learning approach for cross-architecture IoMT malware detection and classification in healthcare cyber-physical systems," *IEEE Transactions on Computational Social Systems*, vol. 10, no. 4, pp. 1597–1606, Aug. 2023, doi: 10.1109/tcss.2022.3198123.
- [9] U. Tariq, I. Ullah, M. Y. Uddin, and S. J. Kwon, "An effective self-configurable ransomware prevention technique for IoMT," *Sensors*, vol. 22, no. 21, p. 8516, Nov. 2022, doi: 10.3390/s22218516.
- [10] D. W. Fernando, N. Komninos, and T. Chen, "A study on the evolution of ransomware detection using machine learning and deep learning techniques," *IoT*, vol. 1, no. 2, pp. 551–604, Dec. 2020, doi: 10.3390/iot1020030.
- [11] R. H. Chamarajappa and G. C. Dyamanna, "A novel and distributed three phase consensus based secured data sharing in internet of things environment," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 31, no. 2, p. 636, Aug. 2023, doi: 10.11591/ijeecs.v31.i2.pp636-646.
- [12] U. Zahoor, A. Khan, M. Rajarajan, S. H. Khan, M. Asam, and T. Jamal, "Ransomware detection using deep learning based unsupervised feature extraction and a cost sensitive Pareto Ensemble classifier," *Scientific Reports*, vol. 12, no. 1, Sep. 2022, doi: 10.1038/s41598-022-19443-7.
- [13] Y.-D. Lin, Z.-Q. Liu, R.-H. Hwang, V.-L. Nguyen, P.-C. Lin, and Y.-C. Lai, "Machine learning with variational AutoEncoder for imbalanced datasets in intrusion detection," *IEEE Access*, vol. 10, pp. 15247–15260, 2022, doi: 10.1109/access.2022.3149295.
- [14] H. Alavizadeh, H. Alavizadeh, and J. Jang-Jaccard, "Deep q-learning based reinforcement learning approach for network intrusion detection," *Computers*, vol. 11, no. 3, p. 41, Mar. 2022, doi: 10.3390/computers11030041.
- [15] H. Rafik, A. Maizate, and A. Ettaoufik, "Data security mechanisms, approaches, and challenges for e-health smart systems," *International Journal of Online and Biomedical Engineering (iJOE)*, vol. 19, no. 02, pp. 42–66, Feb. 2023, doi: 10.3991/ijoe.v19i02.37069.
- [16] A. Oakley, "HIPAA, HIPPA, or HIPPO: what really is the health insurance portability and accountability act?" *Biotechnology Law Report*, vol. 42, no. 6, pp. 306–318, Dec. 2023, doi: 10.1089/blr.2023.29329.aso.
- [17] S. Mulders, "Collective Damages for GDPR breaches: a feasible solution for the GDPR enforcement deficit?" *European Data Protection Law Review*, vol. 8, no. 4, pp. 493–506, 2022, doi: 10.21552/edpl/2022/4/8.
- [18] N. Divyashree and J. Nagaraja, "Review on malware classification with a hybrid deep learning," *Journal of IoT and Machine Learning*, vol. 24, no. 5, pp. 18–31, Jun. 2023, doi: 10.48001/joitml.2023.1118-21.
- [19] A. K. Kumar, K. Vadivukkarasi, R. Dayana, and P. Malarvezhi, "Botnet attacks detection using embedded feature selection methods for secure IOMT environment," *Pervasive Computing and Social Networking*, pp. 585–599, Sep. 2022, doi: 10.1007/978-981-19-2840-6_45.
- [20] U. Tariq, I. Ahmed, A. K. Bashir, and K. Shaukat, "A critical cybersecurity analysis and future research directions for the internet of things: a comprehensive review," *Sensors*, vol. 23, no. 8, p. 4117, Apr. 2023, doi: 10.3390/s23084117.
- [21] T. A. Ahanger, U. Tariq, F. Dahan, S. A. Chaudhry, and Y. Malik, "Securing IoT devices running PureOS from ransomware attacks: leveraging hybrid machine learning techniques," *Mathematics*, vol. 11, no. 11, p. 2481, May 2023, doi: 10.3390/math11112481.





- [22] Y. K. Saheed and M. O. Arowolo, "Efficient cyber-attack detection on the internet of medical things-smart environment based on deep recurrent neural network and machine learning algorithms," *IEEE Access*, vol. 9, pp. 161546–161554, 2021, doi: 10.1109/access.2021.3128837.
- [23] S. Khan and A. Akhuzada, "A hybrid DL-driven intelligent SDN-enabled malware detection framework for internet of medical things (IoMT)," *Computer Communications*, vol. 170, pp. 209–216, Mar. 2021, doi: 10.1016/j.comcom.2021.01.013.
- [24] Kara, M. Aydos, and A. S. Bozkir, "Zararlı Yazılımların Karekterislik Analizi: Cryptowall Fidyeye Yazılım Analizi," *European Journal of Science and Technology*, pp. 486–493, Apr. 2020, doi: 10.31590/ejosat.araconf63.
- [25] I. Kara, "Fileless malware threats: recent advances, analysis approach through memory forensics and research challenges," *Expert Systems with Applications*, vol. 214, p. 119133, Mar. 2023, doi: 10.1016/j.eswa.2022.119133.
- [26] F. Ghulam, M. Irfan, and F. Hassan, "A study of ransomware attacks on windows platform," *i-manager's Journal on Computer Science*, vol. 9, no. 4, p. 21, 2022, doi: 10.26634/jcom.9.4.18530.
- [27] G. McDonald, P. Papadopoulos, N. Pitropakis, J. Ahmad, and W. J. Buchanan, "Ransomware: analysing the Impact on Windows active directory domain services," *Sensors*, vol. 22, no. 3, p. 953, Jan. 2022, doi: 10.3390/s22030953.
- [28] A. Alqahtani and F. T. Sheldon, "A survey of crypto ransomware attack detection methodologies: an evolving outlook," *Sensors*, vol. 22, no. 5, p. 1837, Feb. 2022, doi: 10.3390/s22051837.
- [29] M. S. El-Genk and T. Schriener, "Simulated false data injections attacks on emulated and hardware programmable logic controllers of the pressurizer in a representative pressurized water reactor plant," *Journal of Cyber Security Technology*, vol. 6, no. 4, pp. 216–241, Sep. 2022, doi: 10.1080/23742917.2022.2123191.
- [30] A. O. Almashhadani, M. Kaiiali, S. Sezer, and P. O'Kane, "A multi-classifier network-based crypto ransomware detection system: a case study of locky ransomware," *IEEE Access*, vol. 7, pp. 47053–47067, 2019, doi: 10.1109/access.2019.2907485.
- [31] J. Du, S. H. Raza, M. Ahmad, I. Alam, S. H. Dar, and M. A. Habib, "Digital forensics as advanced ransomware pre-attack detection algorithm for endpoint data protection," *Security and Communication Networks*, vol. 2022, pp. 1–16, Jul. 2022, doi: 10.1155/2022/1424638.
- [32] M. Hirano, R. Hodota, and R. Kobayashi, "RanSAP: an open dataset of ransomware storage access patterns for training machine learning models," *Forensic Science International: Digital Investigation*, vol. 40, 2022, doi: 10.1016/j.fsidi.2021.301314.
- [33] T. Meurs, M. Junger, E. Tews, and A. Abhishta, "NAS-ransomware: hoe ransomware-aanvallen tegen NAS-apparaten verschillen van reguliere ransomware-aanvallen," *Tijdschrift voor Veiligheid*, vol. 21, no. 3, pp. 69–88, Dec. 2022, doi: 10.5553/tvv/000044.
- [34] P. Bajpai and R. Enbody, "Know thy ransomware response: a detailed framework for devising effective ransomware response strategies," *Digital Threats: Research and Practice*, vol. 4, no. 4, pp. 1–19, Oct. 2023, doi: 10.1145/3606022.
- [35] D. Arivudainambi, V. K. Kumar, S. Chakkaravarthy, and P. Visu, "Malware traffic classification using principal component analysis and artificial neural network for extreme surveillance," *Computer Communications*, vol. 147, pp. 50–57, Nov. 2019, doi: 10.1016/j.comcom.2019.08.003.
- [36] J. Manokaran and G. Vairavel, "GIWRF-SMOTE: gini impurity-based weighted random forest with SMOTE for effective malware attack and anomaly detection in IoT-Edge," *Smart Science*, vol. 11, no. 2, pp. 276–292, Dec. 2022, doi: 10.1080/23080477.2022.2152933.
- [37] F. T. ALGorain and J. A. Clark, "Bayesian hyper-parameter optimisation for malware detection," *Electronics*, vol. 11, no. 10, p. 1640, May 2022, doi: 10.3390/electronics11101640.
- [38] M. Katebi, A. RezaKhani, S. Joudaki, and M. E. Shiri, "RAPSAMS: robust affinity propagation clustering on static android malware stream," *Concurrency and Computation: Practice and Experience*, vol. 34, no. 15, pp. 1-15, Apr. 2022, doi: 10.1002/cpe.6980.

BIOGRAPHIES OF AUTHORS



Usman Tariq     currently an Associate Professor in the Department of Management Information Systems at the College of Business Administration, Prince Sattam Bin Abdulaziz University in Saudi Arabia, earned his doctorate from Ajou University in South Korea in 2010. With over a million dollars in research funding, Dr. Usman has tackled cross-disciplinary challenges in áreas, such as, wireless sensor networks, IoT/cyber-physical systems, cybersecurity, and intelligent infrastructures. His collaborative work has resulted in more than 200 influential publications. A recognized figure in his field, he has been at the helm of various international conferences and is a seasoned speaker, with over 50 keynotes and invited talks to his credit. He can be contacted at email: u.tariq@psau.edu.sa.



Bilal Tariq     Higher Education Commission of Pakistan (HEC) approved Ph.D. supervisor, currently working as an Assistant Professor at COMSATS University Islamabad (CUI), Vehari Campus. He holds a Ph.D. in Economics from the Universiti Malaysia Sarawak. He earned his Master in Economics from the Quaid-i-Azam University, Islamabad and Masters of Philosophy in Environmental Economics from Pakistan Institute of Development Economics, Islamabad. He is very keen to contribute towards the growth and development of an educational institute through optimum utilization of his personal abilities and knowledge attained during his studies. His research interests focus on processes of technological and institutional change. He is currently involved in a research project concerned with the process of catch-up by developing economies. He can be contacted at email: bilaltariq@cuivehari.edu.pk.