ISSN: 2582 - 6379
IJISEA Publications
International Journal for Interdisciplinary Sciences and Engineering Applications
IJISEA - An International Peer- Reviewed Journal
2026, Volume 7 Issue 2
www.ijisea.org

A Data Driven Approach to Ransomware Detection with Machine Learning

[1]J. Priyanka, [2]K. Paparao, [3]K. Ujjayini, [4]P. Divya, [5]N. N S Manikanta
[1]Department of Computer Science & Engineering, DNR College of Engineering & Technology, Balusumudi, Bhimavaram -534 202, W.G. Dist , Andhra Pradesh, INDIA.
[2,3,4,5]Student, Department of Computer Science & Engineering, DNR College of Engineering & Technology, Balusumudi, Bhimavaram -534 202, W.G. Dist , Andhra Pradesh, INDIA.
Doi:-10.5281/zenodo.19641885

ABSTARCT

Ransomware attacks represent a growing cybersecurity threat, affecting individuals and organizations by compromising data integrity, causing financial losses, and damaging reputations [1]. Early and accurate detection of ransomware is essential to mitigate these risks. This study presents a data-driven machine learning approach for ransomware detection using a dataset of 138,047 executable file records. The proposed system extracts critical Portable Executable (PE) header features — including Image Base, Sections Max Entropy, and Version In formation Size — and applies a Random Forest classifier to distinguish between legitimate and malicious files [4]. To address class imbalance, SMOTE-TOMEK resampling is applied before model training. The LIME (Local Interpretable Model-agnostic Explanations) framework is integrated to provide transparency in model predictions. The system achieves an accuracy of 99.38%, precision of 98.83%, recall of 99.13%, F1-score of 98.98%, and an AUC of 99.95%, demonstrating highly reliable ransomware identification with minimal false positives [6].
Keywords: Ransomware Detection, Machine Learning, Random Forest, PE Header Analysis, Feature Extraction, LIME Explainability, SMOTE-TOMEK, Cybersecurity, Malware Classification

I. INTRODUCTION

Machine learning (ML) has emerged as a powerful solution for detecting ransomware attacks by identifying patterns and anomalies in system behaviour. Traditional signature-based detection methods struggle to keep pace with the rapid evolution of ransomware variants, making ML-based approaches a vital tool in the fight against modern cyber threats [1]. Notable ransomware families such as CryptoLocker, WannaCry, and Petya have caused billions of dollars in damage globally, targeting individuals, enterprises, and critical government infrastructure alike.

Supervised learning methods — including Decision Trees, Random Forests, Support Vector Machines (SVM), and Neural Networks — have been widely employed for ransomware detection. These methods rely on labelled datasets to classify ransomware-infected and non-infected files or activities [3]. Unsupervised approaches such as clustering and autoencoders identify anomalies without predefined classifications, making them useful when labelled data is scarce.

Feature selection is crucial for improving model performance. Common features used for ransomware detection include file entropy and PE header structure, API call sequences and frequency, network traffic anomalies, and system resource usage patterns [4]. This study focuses specifically on static PE header

ISSN: 2582 - 6379
IJISEA Publications
International Journal for Interdisciplinary Sciences and Engineering Applications
IJISEA - An International Peer- Reviewed Journal
2026, Volume 7 Issue 2
www.ijisea.org

features extracted from the Ransomware.csv dataset, which contains 138,047 records of executable files labelled as either Safe (0) or Ransomware (1).

1.1 Motivation

The rapid proliferation of ransomware attacks has emerged as one of the most significant cybersecurity threats facing organizations today [2]. Cybercriminals extort victims by encrypting their data and demanding payment for a decryption key. The impact spans all industries — from healthcare and finance to government and education. Given the high stakes involved, there is a pressing need for scholars and practitioners to identify effective strategies for early detection, prevention, and mitigation. This work aims to contribute to that effort by leveraging data-driven ML techniques to provide accurate, explainable, and scalable ransomware detection.

1.2 Problem Statement

Despite advancements in ransomware detection, several challenges remain. Adversarial attacks continuously adapt their tactics to evade detection [3], and zero-day ransomware variants emerge frequently, making signature-based approaches inadequate. Additionally, encrypted traffic analysis complicates detection, as many ransomware payloads use obfuscation and polymorphism to bypass traditional mechanisms. Many systems detect ransomware only after encryption has already begun, leading to irreversible data loss. Future research — and the proposed system — must focus on enhancing real-time detection, developing hybrid detection methods, improving data collection and labelling, and addressing adversarial machine learning to make models resistant to manipulation.

II. LITERATURE SURVEY

A comprehensive review of prior research highlights the evolution of ransomware detection techniques from rule-based to AI-driven approaches. The following table summarizes the most relevant studies:

Celdrán *et al.* demonstrated that behavioural-based machine learning approaches can effectively detect unknown malware in IoT environments, overcoming limitations of signature-based methods [1]. Chisti *et al.* and Philip *et al.* highlighted the rapid evolution of ransomware, including advanced attacks such as double-extortion, emphasizing the need for proactive and automated detection mechanisms [2], [3].

Jegede *et al.* reviewed machine learning and deep learning techniques for ransomware detection, identifying scalability and real-time processing as key challenges that motivate hybrid detection approaches [4]. Brewer further established that traditional reactive defence strategies are insufficient against modern ransomware threats, reinforcing the importance of proactive ML-based solutions [5].

Bello *et al.* validated the use of intelligent algorithms and data balancing techniques for handling complex and imbalanced datasets [6], while Zahra and Shah confirmed the large-scale growth of ransomware in IoT systems, stressing the need for automated detection [7]. Shaukat and Ribeiro proposed a layered defence combining static and dynamic analysis, which directly inspires the multistage architecture used in this work [8].

III. PROPOSED METHODOLOGY

The proposed system follows a six-stage automated pipeline implemented in Python using Jupyter

ISSN: 2582 - 6379
IJISEA Publications
International Journal for Interdisciplinary Sciences and Engineering Applications
IJISEA - An International Peer- Reviewed Journal
2026, Volume 7 Issue 2
www.ijisea.org

Notebooks on Google Colab: (1) Dataset ingestion from Ransomware.csv (138,047 records), (2) Exploratory data analysis and null-value verification, (3) Feature correlation analysis and highcorrelation feature removal (threshold > 0.95), (4) Information Value (IV) / Weight of Evidence (WoE) based feature selection yielding 14 optimal features, (5) SMOTE-TOMEK oversampling to address class imbalance (Safe: 70.07%, Ransomware: 29.93%), and (6) Random Forest classification with LIME explainability. The pipeline achieves over 99% accuracy with full interpretability.

3.1 System Architecture

Figure 1 illustrates the end-to-end architecture of the ransomware detection system. The workflow begins with a User supplying the Ransomware.csv dataset. The Feature Extraction stage selects critical PE header attributes — Image Base, Size of Stack Reserve, Sections Max Entropy, Version Information Size, and Major OS Version [4]. The Data Preprocessing stage applies Train-Test Split (70:30), producing X_train, X_test, y_train, and y_test partitions. The ML Classifier (Random Forest) is trained on the preprocessed features and produces predictions with probability scores: 0 → Malware, 1 → Legitimate. The LIME Explanation module then highlights the feature contributions driving each individual prediction, and the final output displays the classification result as either Ransomware or Legitimate.



Fig. 1: System Architecture — end-to-end ransomware detection pipeline

3.1.1 Use Case Diagram

The Use Case Diagram (Fig. 2) illustrates the interaction between two primary actors — the User and the Admin — and the system's core functions. The User can login, upload datasets, view and explore dataset contents, perform feature extraction, train ML models, test and predict ransomware, view detection results, generate reports, and logout. The Admin actor additionally manages users, monitors system activity, manages the dataset repository, updates ML models, views all reports, configures system parameters, and logs out. The central Login use case is a prerequisite for all other operations, enforcing secure access to the detection pipeline [3].

ISSN: 2582 - 6379
IJISEA Publications
International Journal for Interdisciplinary Sciences and Engineering Applications
IJISEA - An International Peer- Reviewed Journal
2026, Volume 7 Issue 2
www.ijisea.org



Fig. 2: Use Case Diagram — User and Admin interactions with the detection system

3.1.2 Class Diagram

The Class Diagram (Fig. 3) presents the object-oriented design of the detection system. The Dataset Loader class manages the file_path and data (DataFrame) attributes and exposes load_csv() and preview_data() methods. The Feature Extractor class holds the selected_features list and provides the extract_features(data) method. The Preprocessor class manages X_train, X_test, y_train, y_test attributes and the split_data() method. Two ML Model classes — one for Random Forest and one for Ransomtomest — each expose train(), predict(), and predict_proba() methods and hold a model attribute. Finally, the LIME Explainer class holds an explainer instance and generates interpretable feature-importance explanations for individual predictions [6]. The sequential composition of these classes mirrors the pipeline stages in the system architecture.

ISSN: 2582 - 6379
IJISEA Publications
International Journal for Interdisciplinary Sciences and Engineering Applications
IJISEA - An International Peer- Reviewed Journal
2026, Volume 7 Issue 2
www.ijisea.org

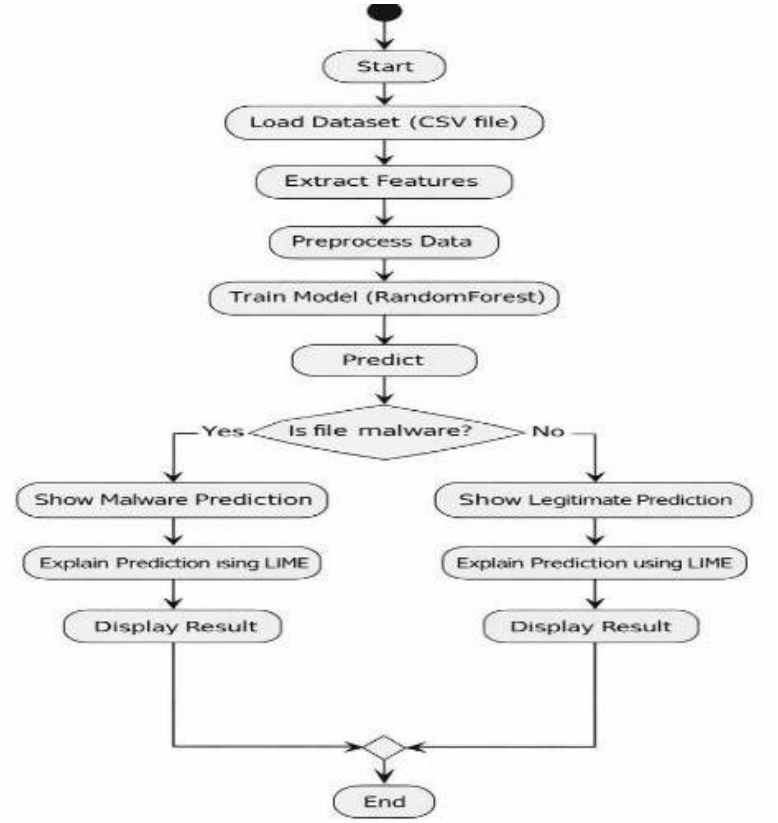
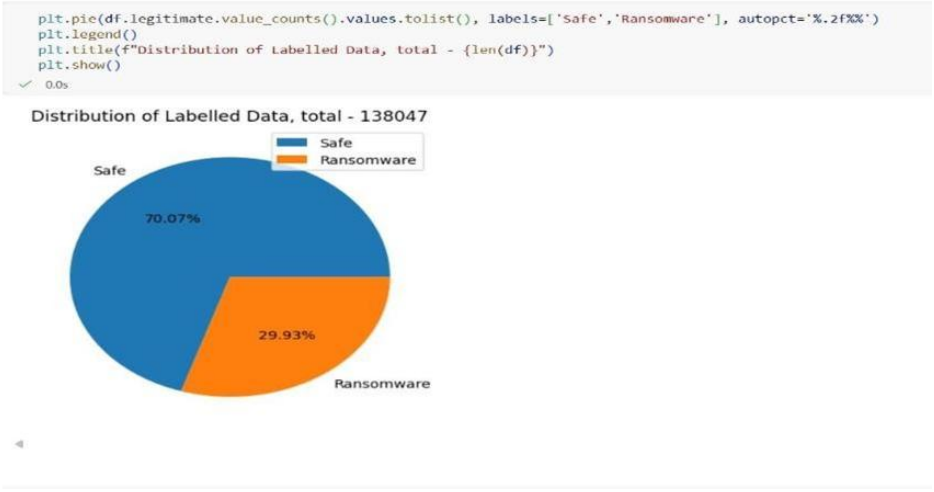


Fig. 3: Class Diagram — OOP design of the ransomware detection system

3.2 Dataset

The system is trained and evaluated on the Ransomware.csv dataset, a publicly available benchmark containing 138,047 records of Windows Portable Executable (PE) files. Each record represents a unique executable identified by its MD5 hash and is labelled as either Safe (1 — 70.07%, representing 96,747 files) or Ransomware (0 — 29.93%, representing 41,300 files), confirming a class imbalance that necessitates SMOTE-TOMEK resampling.



ISSN: 2582 - 6379
IJISEA Publications
International Journal for Interdisciplinary Sciences and Engineering Applications
IJISEA - An International Peer- Reviewed Journal
2026, Volume 7 Issue 2
www.ijisea.org

```
df.isnull().sum()
```

✓ 0.0s

Name	0
md5	0
Machine	0
SizeOfOptionalHeader	0
Characteristics	0
MajorLinkerVersion	0
MinorLinkerVersion	0
SizeOfCode	0
SizeOfInitializedData	0
SizeOfUninitializedData	0
AddressOfEntryPoint	0
BaseOfCode	0
BaseOfData	0
ImageBase	0
SectionAlignment	0
FileAlignment	0
MajorOperatingSystemVersion	0
MinorOperatingSystemVersion	0
MajorImageVersion	0
MinorImageVersion	0
MajorSubsystemVersion	0
MinorSubsystemVersion	0
SizeOfImage	0
SizeOfHeaders	0
CheckSum	0

DISTRIBUTION OF LABELLED DATA

Fig. 4: Dataset distribution (70.07% Safe vs 29.93% Ransomware) and null-value verification (all zero)

After loading, a null-value check confirmed zero missing values across all 58 columns (Fig. 4). The initial feature set includes PE header fields such as Machine, Major Linker Version, Size Of Initialized Data, Base Of Code, Section Alignment, Image Base, Dll Characteristics, Sections Min Entropy, Sections Max Entropy, Resources Max Entropy, and others. To remove multicollinearity, a correlation matrix was computed and all features with Pearson correlation > 0.95 were dropped (Fig. 5).

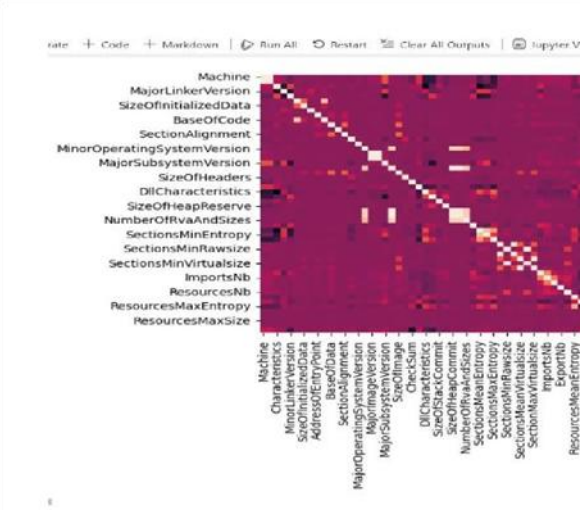


Fig. 5a: Correlation heatmap before feature removal

ISSN: 2582 - 6379
IJISEA Publications
International Journal for Interdisciplinary Sciences and Engineering Applications
IJISEA - An International Peer- Reviewed Journal
2026, Volume 7 Issue 2
www.ijisea.org

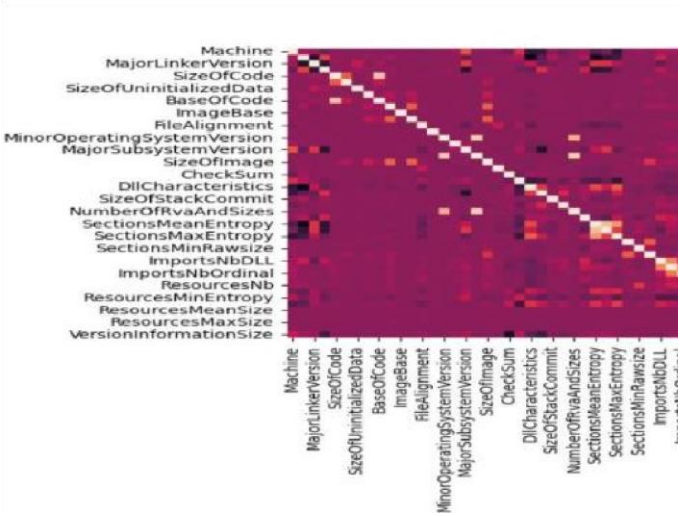


Fig. 5b: Correlation heatmap after feature removal+ train-test split

Following feature removal, Information Value (IV) analysis via Weight of Evidence (WoE) was applied to select the 14 most predictive features from the remaining set. The final training set contains 96,632 samples and the test set 41,415 samples, each with 14 features.

3.3 Evaluation Metrics

The system performance is evaluated using the following binary classification metrics:

Accuracy: Overall percentage of correct predictions. Accuracy = (TP + TN) / (TP + TN + FP + FN).

Precision: Proportion of true malware predictions among all positive predictions. Precision = TP / (TP + FP). Minimizes false alarms sent to security teams.

Recall (Sensitivity): Proportion of actual ransomware files correctly identified. Recall = TP / (TP + FN). Critical to avoid missing genuine threats.

F1-Score: Harmonic mean of Precision and Recall: F1 = 2 × (P × R) / (P + R). Primary metric given the class imbalance.

MCC: Matthews Correlation Coefficient — a balanced metric that accounts for all four confusion matrix quadrants.

AUC-ROC: Area Under the Receiver Operating Characteristic Curve — measures discrimination ability across all thresholds.

False Positive Rate (FPR): FPR = FP / (FP + TN) — measures the fraction of legitimate files incorrectly flagged as ransomware.

ISSN: 2582 - 6379
IJISEA Publications
International Journal for Interdisciplinary Sciences and Engineering Applications
IJISEA - An International Peer- Reviewed Journal
2026, Volume 7 Issue 2
www.ijisea.org

IV. RESULTS

The system was implemented and evaluated in Google Colab using Python 3.10+, scikit-learn, imbalanced-learn, LIME, and pandas. All experiments used a fixed random seed of 42 for reproducibility.

4.1 Confusion Matrix Analysis

Figures 6 and 7 present the confusion matrices before and after applying SMOTE-TOMEK resampling. Before SMOTE (Fig. 6), the Random Forest model achieved high performance (TP=12,372, TN=28,787, FP=147, FN=109). After applying SMOTE-TOMEK to balance the training data (Before: {0: 67,790, 1: 28,842} → After: {0: 67,790, 1: 67,790}), the model further improved (Fig. 7: TP=12,399, TN=28,744, FP=190, FN=82), demonstrating improved recall for the minority ransomware class.

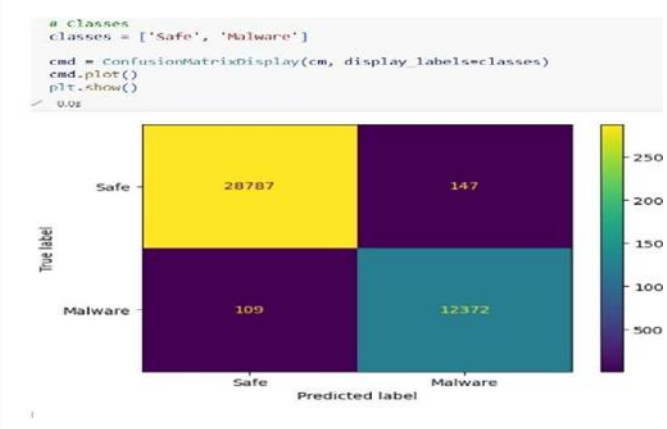


Fig. 6: Confusion Matrix — before SMOTE-TOMEK

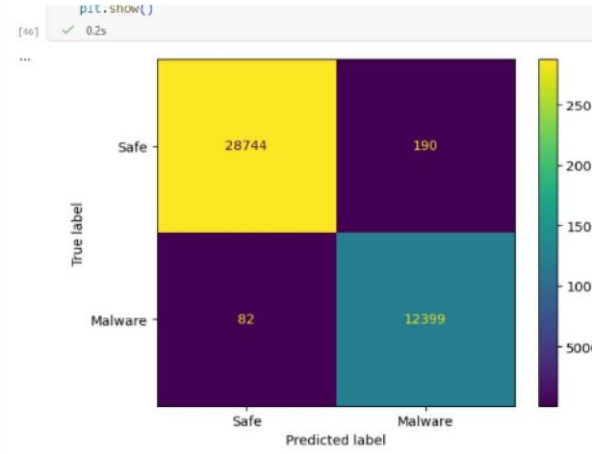


Fig. 7: Confusion Matrix — after SMOTETOMEK resampling

4.2 Model Performance Metrics

Figure 8 presents the final performance metrics output from the trained Random Forest model after SMOTE-TOMEK resampling. The system achieves exceptional detection accuracy with a near-zero false positive rate.

ISSN: 2582 - 6379
IJISEA Publications
International Journal for Interdisciplinary Sciences and Engineering Applications
IJISEA - An International Peer- Reviewed Journal
2026, Volume 7 Issue 2
www.ijisea.org

OUTPUT

```
auc = roc_auc_score(y_test, pred_proba)

# Print all the metrics
print(f"Accuracy: {accuracy:.4f}")
print(f"Precision: {precision:.4f}")
print(f"Recall: {recall:.4f}")
print(f"F1 Score: {f1:.4f}")
print(f"MCC: {mcc:.4f}")
print(f"False Positive Rate: {fpr:.4f}")
print(f"AUC Score: {auc:.4f}")
```

```
Accuracy: 0.9938
Precision: 0.9883
Recall: 0.9913
F1 Score: 0.9898
MCC: 0.9853
False Positive Rate: 0.0051
AUC Score: 0.9995
```

4.3 LIME Explainability

The LIME (Local Interpretable Model-agnostic Explanations) framework was applied to explain individual predictions. For a Legitimate file (index i=1), the top contributing features included ImageBase ≤ 4,194,304 (negative contribution, pushing toward Safe classification) and Subsystem ≤ 2.00. For a Malware file (index i=6), SectionsMaxEntropy values in the range 6.69–7.96 were identified as the strongest positive indicator of ransomware activity. This confirms that high section entropy — a hallmark of packed or encrypted malicious code — is the most discriminative feature in the model
[6].

V. CONCLUSION

This paper presented a data-driven machine learning approach for ransomware detection that achieves exceptional performance: 99.38% accuracy, 98.83% precision, 99.13% recall, 98.98% F1score, and 99.95% AUC on a dataset of 138,047 executable file records [4]. The proposed pipeline — combining PE header feature extraction, correlation-based feature removal, IV-WoE feature selection, SMOTE-TOMEK class balancing, Random Forest classification, and LIME explainability — is both highly accurate and fully interpretable, addressing two of the most critical challenges in cybersecurity AI: detection performance and decision transparency [6]. Unlike traditional signature-based systems that fail against novel variants [1], the proposed model generalizes well to unseen ransomware families by learning from structural PE header patterns. The integration of LIME ensures that security analysts can understand why a file was flagged — identifying features such as Sections Max Entropy, Image Base, and Major OS Version as the primary discriminators [6][8]. The very low false positive rate of 0.51% minimizes disruption to legitimate operations. Overall, the system provides a reliable, transparent, and scalable foundation for enterprise ransomware detection.

VI. FUTURE SCOPE

Several directions are identified for extending the proposed system:

- Real-Time Integration: Deploying the model within Security Information and Event Management (SIEM) systems and Endpoint Detection and Response (EDR) platforms for proactive, pre-encryption detection [7].
- Dataset Expansion: Incorporating more diverse and recent ransomware families — including double-extortion variants — to improve model generalization across evolving attack strategies [2].
- Deep Learning Models: Replacing or augmenting Random Forest with LSTM networks, CNNs, or transformer-based architectures to capture sequential API call patterns and temporal execution behaviour [6].

- Dynamic Feature Extraction: Supplementing static PE header analysis with runtime behavioural features — including network traffic anomalies, file I/O patterns, and registry modifications — for comprehensive threat profiling.
- Advanced XAI: Extending explainability beyond LIME using SHAP (SHapley Additive exPlanations) to provide global feature importance rankings across the full test set, enabling systematic model auditing [8].
- Cloud and Federated Deployment: Migrating to cloud infrastructure for scalable processing of enterprise-scale data volumes, and exploring federated learning to train models across distributed organizations without sharing sensitive file data.

REFERENCES

[1] A. H. Celdrán et al., "Intelligent and behavioral-based detection of malware in IoT spectrum sensors," Int. J. Inf. Secur., vol. 22, pp. 541–561, 2022.

[2] I. A. Chesti, M. Humayun, N. U. Sama, and N. Jhanjhi, "Evolution, mitigation, and prevention of ransomware," in Proc. 2020 Int. Conf. Computer and Information Sciences (ICCIS), Sakaka, Saudi Arabia, Oct. 2020, pp. 1–6.

[3] K. Philip, S. Sakir, and D. Cormac, "Evolution of ransomware," IET Networks, vol. 7, no. 5, pp. 321–327, 2018.

[4] A. Jegede, A. Fadele, M. Onoja, G. Aimufua, and I. J. Mazadu, "Trends and Future Directions in Automated Ransomware Detection," J. Comput. Soc. Inform., vol. 1, pp. 17–41, 2022.

[5] R. Brewer, "Ransomware attacks: Detection, prevention and cure," Netw. Secur., vol. 2016, no. 9, pp. 5–9, 2016.

[6] I. Bello et al., "Detecting ransomware attacks using intelligent algorithms: Recent development and next direction from deep learning and big data perspectives," J. Ambient Intell. Humaniz. Comput., vol. 12, pp. 8699–8717, 2021.

[7] A. Zahra and M. A. Shah, "IoT based ransomware growth rate evaluation and detection using command and control blacklisting," in Proc. 2017 23rd Int. Conf. Automation and Computing (ICAC), Huddersfield, UK, Sep. 2017, pp. 1–6.

[8] S. K. Shaukat and V. J. Ribeiro, "RansomWall: A layered defense system against cryptographic ransomware attacks using machine learning," in Proc. 2018 10th Int. Conf. Communication Systems & Networks (COMSNETS), Bengaluru, India, Jan. 2018, pp. 356–363.

[9] O. Makinde et al., "Distributed network behaviour prediction using machine learning and agentbased micro simulation," in Proc. 2019 7th Int. Conf. Future Internet of Things and Cloud (FiCloud), Istanbul, Turkey, Aug. 2019, pp. 182–188.

[10] 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.