Volume (2) No (1), 2026
Journal of Global Research in Multidisciplinary Studies (JGRMS)
Research Paper
Available online at https://saanvipublications.com/journals/index.php/jgrms/index

Ransomware Attack Detection using Intelligent Machine Learning Algorithms: A Systematic Review of Challenges and Future Directions

Himani Pandya[1st]
Research Scholar
Computer Science
RK University
Joshi.himani1193@gmail.com

Snehal Sathwara[2nd]
Assistant Professor
Computer Science
RK University
ed.snehal@gmail.com

***Abstract*—Cybersecurity is a major concern in the IT sector these days. As a direct result of the increasing number of ransomware attacks and other types of cyberattacks, the protection of user data has become a primary focus. This systematic review explores the use of smart machine learning (ML) algorithms to identify ransomware. It demonstrates how these algorithms, towards their flexibility and precision, can be used for malware detection. A thorough survey of 30 primary studies from 2018 to 2025 reveals that advanced machine learning models like Random Forest, SVM, CNN, LSTM, XGBoost, and ensemble methods are the main contributors to achieving high detection accuracy results, very often beyond 99%, particularly when ransomware datasets are used. These models are able to detect the presence of ransomware in various platforms by utilizing static, dynamic, and hybrid features sets. The review also takes into account issues such as dataset imbalance, limitations of generalization, and detection strategies. The next goals are to create models that are resistant to adversarial attacks, employing generative augmentation (GANs), developing light machine learning architectures for real-time edge deployment, and using Explainable AI (XAI) to make AI more transparent. The findings demonstrate that ML models with high accuracy that are well-optimized and contextualized can significantly enhance the efficacy of cutting-edge ransomware defense systems. In sum, this study serves as a detailed and organized guide from the perspective of researchers and practitioners for the production of high-precision, robust, and scalable ransomware detection systems.**

***Keywords*—Cybersecurity, Malware, Ransomware Detection, Threat Intelligence, Machine Learning, Deep Learning (DL). Explainable AI (XAI).**

I. INTRODUCTION

Malware detection has become a significant research focus over the last few years, where the researchers investigate specifically ransomware, as the amount of private data that is kept on devices and in the cloud during network transfers has greatly increased [1][2]. To penetrate a system, a ransomware-like attack normally spreads and infects the system, then searches for the files that are most valuable, encrypts them, and asks for a ransom, in most cases, threatening to expose the victim's private data if the payment is not made. Such occurrences may lead to real physical and economic harm to people, businesses, and the whole society-infrastructure chain, e.g., through financial losses, downtime, data breaches, and brand damage [3][4]. Attackers are using phishing emails, malware files, or hacked websites to gain initial access and then rapidly infect networks and cloud environments to achieve their objectives. Numerous are the ways in which ransomware can be built, each subsequently more intricate in terms of detection and protection [5]. Encryption ransomware is the most typical one. It basically immobilizes the data of the victims and requests money in exchange for the decryption key. Scareware tricks people into buying fake solutions by sending them false alarms and using social engineering. Non-encrypting or locker ransomware, on the other hand, locks the user out of the whole computer without deleting any of the files [6][7][8]. It is very difficult to locate and remove ransomware by traditional means because the perpetrators can behave and attack in numerous different ways [9].

Most of the signature-based methods in traditional ransomware detection mainly locate threats through matching the code of malware that has been previously identified. It is often the case that new ransomware variants with different signatures are spun out, making static detection techniques obsolete very quickly [10]. Consequently, there is a demand for sophisticated techniques that not only inspect the static features of files but also monitor the real-time behavior of malicious code [11][12]. By integrating the static analysis, which inspects the malware code without execution, and the dynamic analysis, which observes how the ransomware operates in a simulated environment, security systems are capable of detecting anomalous activities such as rapid file encryption or unusual system changes prior to wreaking havoc [13]Considering these issues, ML has evolved to be an effective method for enhancing ransomware detection [14] [15]. ML models are capable of learning the patterns from the massive datasets of past attacks and are able to detect new ones by identifying slight variations in the system behavior. By employing behavior analysis, ML algorithms, and signature-based scanning concurrently, the present-day security solutions can accomplish the task of locating ransomware in a much shorter time and with less effort [16].

Most of the previous systematic reviews have mainly concentrated on the impacts of ransomware in certain sectors, like the healthcare industry, and thus have largely overlooked the dominant nature of ransomware across various platforms and environments [17][18]. By examining the evolution of a ransomware attack in detail, this work changes the way it looks at the issue. It also commits to the existing literature by extensively exploring the different methods of identification and interception, while emphasizing intelligent ML

techniques [19]. The paper evaluates in-depth the pros and cons of various models, namely static, dynamic, hybrid, and explainable AI-driven ones, over different datasets and platforms [20] [21]. Moreover, it proposes several methods to lessen the chances of a ransomware attack through the use of preventive measures and mitigation practices. This evaluation not only compiles existing knowledge but also creates a solid structure for further research and the establishment of dependable, scalable solutions for the ransomware problem.

A. Motivation and Contribution of Work

Ransomware assaults are evolving in terms of intricacy and frequency; thus, they represent a major challenge to worldwide cybersecurity. In addition, such threats are endangering not only the individuals, but the corporations and the national infrastructures as well [22]. The rapidly changing and polymorphic ransomware variants that can easily evade traditional filters are too difficult for traditional defense mechanisms, like signature-based and heuristic identification systems, to keep up with. The attack surface has increased dramatically as digital ecosystems spread across cloud platforms, IoT devices, and mobile systems, calling for proactive and astute detection techniques [23]. In such a scenario, ML emerges as a viable alternative since it equips the system with the capability to extract behavioral patterns and subsequently spot unknown threats. Consequently, it becomes imperative to continuously monitor the present ML-based ransomware detection techniques, to grasp their effectiveness and inefficiency, and to seek those aspects which are left uncharted in order to facilitate the development of cybersecurity frameworks that are stronger, more adaptable, and more capable of being scaled. This paper is further marked with the contributions that are listed below:

- **Comprehensive Literature Synthesis:** Systematically examines 30 primary studies that focus on the deployment of intelligent machine learning algorithms to identify ransomware in different settings.
- **Taxonomy of Detection Techniques:** Elaborate on how different types of ransomware detection techniques are structured, such as resourceful determination through static, dynamic, hybrid, heuristic, sandboxing, anomaly detection, deception-based, real-time, and also AI-driven explainable methods.
- **Feature Engineering Insights:** Points out and categorizes the most frequent feature sets (e.g., API calls, file behaviors, transaction logs, system events) and emphasizes their role in the development of ML-based ransomware detection frameworks.
- **Dataset Mapping and Challenges:** Reviews the availability of data sources (both publicly accessible and custom-built) that have been used in research, and pinpoints vital issues such as data imbalance, scarcity, lack of standard benchmarks, and difficulties in cross-platform applicability.
- **Research Gaps and Future Roadmap:** It suggests, among other things, the requirement for energy-efficient, edge-compatible models, GAN-based data augmentation, adversarial resilience, and scalable ML implementation in real-world situations as the most significant strategic directions for the future.

The findings of this comprehensive research emphasize that machine learning is the key factor in enhancing ransomware detection by means of analyzing behavioral patterns, static features, and hybrid techniques on different platforms. The paper serves as a map for making ransomware defense systems more intelligent, user-friendly, and resistant by specifying significant detection methods, datasets, issues, and next steps.

B. Structure of Paper

The findings of this comprehensive research emphasize that machine learning is the key factor in enhancing ransomware detection by means of analyzing behavioral patterns, static features, and hybrid techniques on different platforms. The paper serves as a map for making ransomware defense systems more intelligent, user-friendly, and resistant by specifying significant detection methods, datasets, issues, and next steps.

II. Overview of Malware

Malware comprises the malicious software that aims to undermine the security of a system and, in doing so, generate money illegally. Among the illegal activities committed by it are identity theft and data breaches; besides, it can be spread through various executable or software vectors. Malware is divided into three different categories, namely, viruses, worms, and trojans. A computer virus is a program or code that reproduces by making copies of itself in other programs. An important safety measure is that viruses need a software-housing program in which to carry on their harmful activities. On the other hand, a computer worm is an independent one that can make an unlimited number of its copies and run its code without the intervention of any other software. Viruses often seek to propagate via programs or data on an individual computer system.

A. Categories of Malware

Numerous malware types have been identified, and their numbers are rising annually. The following sections contain information on the behavior of various malware families.

- **Trojans:** Trojans are malicious applications that disguise themselves as legitimate or benign apps to deceive users [24]. Once installed, they secretly steal sensitive data such as browsing history, messages, contacts, device identifiers, and even banking credentials. Some Trojans, like SMS Trojans, exploit premium messaging services to cause financial loss by sending paid messages without the user's consent. Others mimic trusted apps (e.g., fake banking or streaming apps) to harvest login detail.
- **Backdoors:** In order to grant malware root access and facilitate their ability to avoid detection by antivirus software, backdoors employ root exploits [25]. The top three root exploits that take complete control of a device are Exploit, Rage against the cage (RATC), and Zim Perlich. Once inside, they give the malware root access, which enables it to evade detection by antivirus software and carry out tasks like installing new apps or altering system preferences without the user's awareness.
- **Worms:** These malicious programs make duplicates of it and disseminate them throughout the network[26]. For example, Bluetooth worms spread malware via the Bluetooth network by transferring copies of the malware to the connected devices. Android. Obad. OS is an example of a Bluetooth worm.

- **Spyware:** GPSSpy and Nickspy are two examples of spyware applications that appear innocuous but really track a user's location, contacts, messages, bank mTANs, and other sensitive information for unauthorized reasons [27]. Without the victim's awareness, personal spyware can install the harmful payload. It gives the attacker who installs the program victim's smartphone access to user's data, including contacts and text messages.
- **Botnets:** A botnet is a collection of smartphones that have been hacked. The C&C network is used by the botmaster, a distant server, to manage the botnet. A collection of compromised various devices managed remotely by a command-and-control (C&C) server is known as a botnet. Coordinated assaults including spam, data theft, and distributed denial-of-service (DDoS) attacks are frequently carried out via these devices.
- **Riskware's:** Riskware's are genuine software programs that criminal writers take advantage of in order to lower device performance or damage data, such as by deleting, altering, or copying it.
- **Ransomwares:** Until the ransom is paid, ransomware locks the device, preventing the user from accessing their data. One piece of malware that poses as an antivirus program is called Fake Defender. The victim's device is locked, and in order to unlock it, the user must pay a ransom.

Ransomware, sometimes referred to as malware, is a sort of malicious software that is always evolving. Its goal is to make a device's files and the systems that use them useless by encrypting them. The victims are forced to pay a ransom to either restore their data or save their reputation.

III. Ransomware and Its Types

IoT device ransomware attacks take use of a number of special features that set them apart from assaults on conventional IT systems. Ransomware has been a significant part of cybersecurity concerns since 1989 and has progressively evolved to be more complex and harmful. Basically, it refers to the act of locking or encrypting the victim's data and demanding ransom payments, which are most of the time in untraceable cryptocurrencies such as Bitcoin and Ethereum [28]. Since these are decentralized and anonymous payment methods, the attackers don't have to worry about being tracked by the authorities while using them. The situation is even worse in the case of the devices that are so spread among the people (in their homes, offices, factories, hospitals, etc.) that the attackers find it very difficult to be confronted with. The strategies used in ransomware operations are not the same as the past. Once upon a time, the attackers were targeting only individual users, while now they are targeting enterprises, financial institutions, healthcare providers, and government agencies. These entities, as a result, are attractive because they have connections to critical infrastructure and possess a large volume of sensitive data. The fallout of a ransomware assault aimed at these organizations can wreak havoc across entire industrial chains, cease services, and sabotage operations. Attackers take advantage of this flaw to demand astronomical ransoms and seriously harm businesses and finances. According to IBM's Security X-Force Threat Intelligence Index 2023, approximately 23% of cyberattacks on the manufacturing sector in 2022 were ransomware-related. Even when ransoms are not paid, the aftermath ranging from data loss to extensive recovery efforts can be devastating. As ransomware threats become more targeted and resilient, the need for proactive detection and response strategies has never been more critical [29].

A. Types of Ransomwares

This section provides the three primary types of ransomwares: crypto-ransomware, locker ransomware, and scareware.

1) Crypto-Ransomware

Amidst the several types of ransomwares that target computer systems and networks, crypto is the most prevalent instance. A combination of symmetric and asymmetric encryption methods is utilized by ransomware in order to encrypt data and files. The data that has been encrypted becomes unreadable even if the malicious software is removed from a computer that has been infected with it or if a storage device that has been hacked is introduced to another system. However, the device that has been hacked can still be used to pay the ransom since the virus does not always erase essential data that has been imported.

2) Locker Ransomware

Locker ransomware stops its owner from accessing a computer or other device by locking it and requesting money. Although the locker ransomware affects the workstation, stored data is not made unusable. The data remains unchanged once the malicious application has been removed. By connecting the compromised storage device like a hard drive to another computer, the data may frequently be recovered. Locker ransomware will not attract anyone who wish to demand money from victims of attack. Locker ransomware, a type of harmful software that is becoming more common in intrusions.

3) Scareware Ransomware

Scareware exploits its victims by telling them that their computers have been compromised and rendering to use a phoney antivirus product that the attacker has supported to remove the ransomware. Scareware notifications are so common that many gullible users purchase and install phoney antivirus software. Human-operated ransomware differs from traditional ransomware by targeting entire organizations rather than individual systems. Attackers break into networks, escalate privileges, and move laterally to exploit vulnerabilities, often due to poor security configurations. By gaining unauthorized access to privileged credentials, they can disrupt critical business operations.

Ransomware without files, on the other hand, launches assaults via a natural and dependable mechanism. Since no code needs to be installed on victim's computer for the attack to be effective, it is challenging to spot.

B. Ransomware Attacks and Its Impact

As Table I discusses, ransomware attacks may take many different forms, each of which disrupts the victim's ability to access their data. The primary categories are described below according to their techniques:

TABLE I. CLASSIFICATION OF RANSOMWARE BASED ON MALWARE CHARACTERISTICS, PLATFORMS, AND TARGETS

Category	Attacks	Description	Examples
Ransomware classification on malware characteristics	Encrypting Ransomware	leverages strong encryption to render the data belonging to the victim unreadable in the absence of a decryption key.	Cryptowall, WannaCry, and Cryptolocker
	Non-Encrypting Ransomware (Locker)	Locks the user out of the entire device; data remains intact but inaccessible.	CTB-Locker, Winlock
	Scareware (Fake Antivirus)	Uses fake warnings and social engineering to scare victims into paying for fake solutions.	Fake AV software, rogue security apps
Ransomware stratification based on platforms	PC/Workstation Ransomware	Focuses on workstations and personal PCs running Linux, macOS, or Windows.	WannaCry
	Mobile Ransomware	Infects mobile devices such as smartphones and tablets by means of malicious applications or websites that have been hacked.	Various Android locker apps
	IoT Ransomware	Targets Internet of Things devices with weak security, causing major disruption.	Smart home and industrial device attacks
Ransomware classification based on targets	Individual Users	Attacks home users with weaker security awareness, using phishing or malicious websites.	Generic phishing ransomware
	Enterprises	Targets businesses for valuable data and bigger payouts; often uses advanced intrusion methods.	Ryuk, Maze
	Government & Critical Infrastructure	Aims to disrupt vital systems like healthcare, energy, or national security services.	Not Petya, Conti
	Online Services	Attacks cloud services or online platforms to encrypt or steal bulk data.	Cloud account hijacks, service-wide ransomware

Ransomware infections impact both individual users and companies of any size or sector, resulting in service interruptions, monetary losses, and, in certain cases, the irreversible loss of important data. Potential effects of ransomware attacks include the following:

- **Financial loss:** Ransomware is a costly and unforeseen expense. In addition to the loss of revenue, an organization may incur other costs, some of which are evident and others of which are not. The ransom payment (if paid), the cost of cleanup (including hardware and software replacement), legal fees and litigation, insurance deductibles, incident response support, and public relations are more obvious expenses.
- **Recovery time:** The ransomware attack can hamper the functioning of a company considerably. Restoring the affected systems can be a time-consuming process, even in the case of a well-prepared organization with working backups. Additionally, it may take days or weeks for organizations that were ill-prepared or whose backups were compromised during the assault to fully resume operations, which suggests that their revenues would decline or stop completely during this time.
- **Data Loss:** The attacker encrypts various files and the systems reliant on them during a ransomware attack, rendering them inoperative. Failure to remit the ransom will result in the irreversible locking of these encrypted files, compelling the organization to rebuild the data.
- **Reputation damage:** Reputational harm is responsible for 20% of ransomware expenses. A ransomware attack or data leak might jeopardize an organization's reputation. Customers or clients of the business may believe that the company has extremely lax security procedures as a result of the data leak. The company may lose business as a result of the service interruption.

IV. RANSOMWARE DETECTION METHODS AND SOLUTIONS

As shown in Fig. 1, there are eleven different classes into which ransomware attack detection and categorization methods for IoT devices may be divided. The categories include: signature-based approaches, heuristics, behavioral analysis, anomaly detection, artificial intelligence, deception techniques, sandboxing, real-time protection systems, file integrity monitoring, cloud solutions, blockchain, and hybrid methods. Table II below provides a brief discussion of each of these detection strategies:

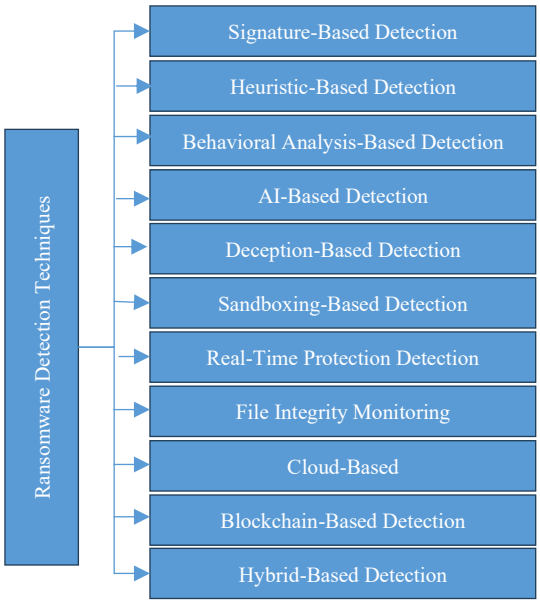

Fig. 1. Stratification of ransomware identification methods on IoT [30]

TABLE II. CLASSIFICATION OF RANSOMWARE DETECTION TECHNIQUES AND MITIGATION STRATEGIES

Category	Approach	How It Works	Example/Description	Limitations	Solutions
Signature-Based	Static signature matching	Compares incoming files or traffic to a database of known ransomware signatures (e.g., hashes, API patterns)	Locky ransomware detected via signature of encryption logic	Ineffective against new, obfuscated, or polymorphic ransomware	Frequent updates, behavioral signatures, integrate with anomaly detection

Heuristic-Based	Rule-based + ML	Detects based on suspicious code behavior or static features, using predefined rules and ML to detect deviations	96% detection accuracy using ML and heuristics on system behavior	May generate false positives/negatives; static rules may not adapt to new threats	Combine with dynamic analysis, use adaptive learning algorithms
Behavioral Analysis-Based	Dynamic pattern learning	Monitors real-time activity (file access, encryption, network traffic) and compares it to baseline behavior	Maze ransomware flagged via anomalous network behavior	Can be bypassed by slow-acting or stealthy ransomware	Use optimization (e.g., PSO, genetic algorithms), incorporate DL
AI-Based	ML/DL (CNN, RNN, RL)	Learns patterns in logs, traffic, or behavior using neural networks; CNNs for features, RL for adapting to evolving threats	Ryuk detected using CNNs on encryption logs; RL adapts to changing attacks	Requires large labeled datasets; high computational cost	Use transfer learning, model pruning, and lightweight architectures for IoT
Deception-Based	Honeypots/Honey files	Lures ransomware to interact with fake data (decoys), revealing its behavior and attack strategy	R-locker and honey files to log ransomware behavior	Resource intensive; risk of real system interaction if not isolated	Use dynamic, adaptive decoys; isolate decoys fully
Sandboxing-Based	Isolated execution	runs suspicious software in a virtual environment to watch for behaviour like as command-and-control communication or file encryption.	Used in APT detection (Flame, Red October), real-time pattern observation	Time-consuming; sandbox-aware malware may evade detection	Randomize environment, integrate ML to analyze behavior faster
Real-Time Protection	Continuous monitoring	Monitors system activities using behavioral/anomaly detection, alerts or blocks upon detecting ransomware indicators	ARDS detects anomalies; Rwguard blocks early-stage encryption	May miss stealthy or delayed attacks; can be noisy with false positives	Use unsupervised ML, enhance with contextual pattern learning
File Integrity Monitoring	Hash comparison	Tracks changes in file state using cryptographic hashes; alerts if files are modified/encrypted without authorization	BiLSTM+CRF for sequential pattern detection of file changes	Can't prevent ransomware if it's already active; limited to known file sets	Add predictive learning, extend monitoring to real-time events
Cloud-Based	Centralized detection	Aggregates IoT data logs in cloud; uses ML/DL to detect ransomware patterns; supports backup and recovery	RANSOMNET+ combines CNN + transformer on encrypted data; Cloud RPS with backups	Dependency on network/cloud; latency; privacy concerns	Use encrypted analytics, edge-cloud hybrid for faster local processing
Blockchain-Based	Immutable ledger check	Uses blockchain for storing hashes and event logs; detects unauthorized modifications; ensures decentralized, tamper-proof auditing	Cerber detected using hash mismatches on blockchain; IoT audit via blockchain	Scalability, high energy/resource consumption, and slower consensus mechanisms	Use lightweight blockchain models (e.g., DAG), selective logging strategies
Hybrid-Based	Multi-tech integration	Combines multiple techniques (e.g., AI + sandboxing, signature + behavior) for comprehensive detection and resilience	Integration of behavioral + ML + deception for high detection accuracy	Complexity, resource overhead, and potential integration conflicts	Modular design, edge intelligence, load balancing for IoT devices

V. Research Methodology for SLR

This section lays out the deliberate methodology adopted to identify, select, and scrutinize studies relevant to ransomware detection using machine learning. The methodology is structured according to widely accepted guidelines for conducting systematic literature reviews, ensuring transparency, reproducibility, and comprehensive coverage, as shown in Figure 1.

A. Research Objectives and Questions

The main objectives of this SLR are:

- To identify ML techniques applied for ransomware identification.
- To classify types of ransomware attacks and features used in detection.
- To scrutinize the effectiveness and potentiality of different ML-based approaches.

The guiding research questions (RQs) are:

- **RQ1:** What are common ML approaches used for ransomware detection?
- **RQ2:** What types of datasets are employed in these studies?
- **RQ3:** How do ML models perform in terms of recall, F1-score, accuracy, and precision?
- **RQ4:** What are trends, challenges, and gaps in current ransomware identification research?

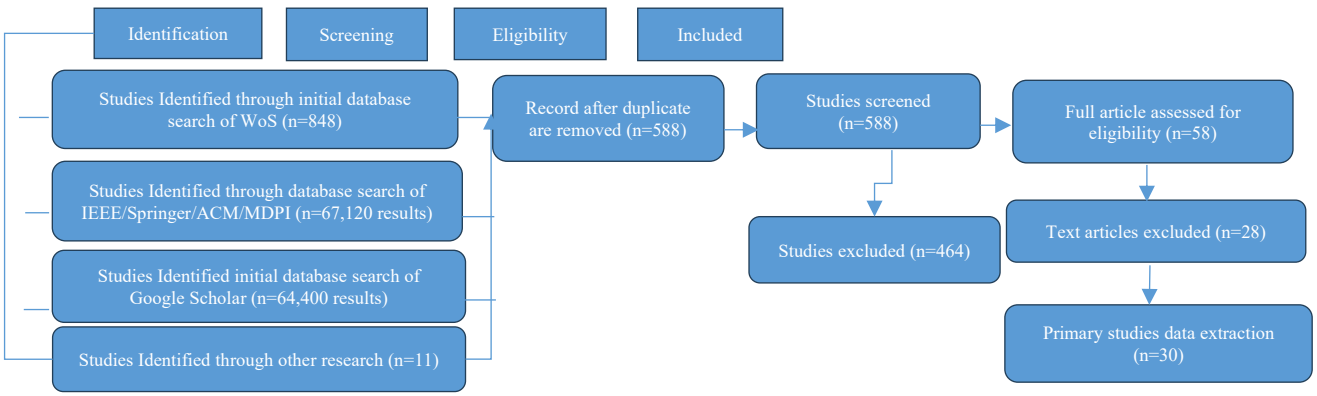

Fig. 2. Flowchart of SLR Review methodology

B. Source Material

To corroborate that primary research that would answer the research questions was retrieved, study employed a particular search engine and concentrated on inputting pertinent keywords. The chosen keywords were picked with care in order to maximize the production of pertinent results. The only boolean operators were AND OR. (Insert particular keywords here) were the search phrases used. (1) "ransom" OR "ransom-ware" OR "ransomware" OR "ransomware" OR "Mal-ware" OR "Malware" OR "ransomware attacks") AND "information security" ("ransomware" OR "ransom" OR "Malware AND" OR "cybersecurity" OR "cyber-security"). The first step is to do a thorough literature search to scrutinize the quality of study. As a result, a search for pertinent documents was carried out utilizing six distinct electronic libraries: IEEE Xplore, Science Direct, ACM, MDPI, Springer, Web of Science, and Google Scholar.

The four steps that make up the review technique are shown in Fig. 2. Depending on the platform, titles, keywords, and abstracts were employed in the search for pertinent papers. The selection/eligibility criteria listed below were used to include and filter all research published up to a specific point. Iterative forward and backward searches were conducted until no further publications that satisfied the selection criteria could be located. Both published and continuing research papers about ransomware attacks are included in this overview of the literature. To identify and select relevant publications for study, the process involves several steps and library searches.

1) Study Selection Process

The selection process followed a multi-step filtering procedure, summarized in the PRISMA flowchart below and detailed as follows:

Step 1. Identification

A total of 132,379 records were retrieved using keyword combinations like "ransomware detection," "machine learning," "malware classification," and "cybersecurity threat detection" from the following sources:

- IEEE Xplore, Springer, ACM, MDPI – 67,120 results
- Google Scholar – 64,400 results
- Web of Science (WoS) – 848 articles
- Other Research Sources – 11 articles

Step 2. Duplicate Removal

Duplicates records identified across the databases were removed, leaving 588 unique articles.

Step 3. Title and Abstract Screening

These 588 articles were evaluated for their relevance to ransomware detection and machine learning applications. 464 articles were discarded due to focus mismatch or duplicate themes.

Step 4. Full-Text Eligibility Review

The full text of the remaining 124 articles was reviewed. After 58 articles were evaluated for eligibility based on specific criteria, 28 of them were rejected due to lack of reproducibility, insufficient depth, or being not relevant.

Step 5. Final Inclusion

As a result of the comprehensive screening process, 30 high-quality primary studies have been decided to be included in the final synthesis and analysis.

C. Inclusion and Exclusion Criteria

In order to keep the rigor and reliability of SLR, the following criteria, which are shown in Tables III and IV, were used during the selection process:

TABLE III. INCLUSION CRITERIA (IC) WITH EXPLANATION

Inclusion Criteria	Explanation
Publication Period: 2018–2025	Focuses on recent advancements and emerging trends in ransomware detection using ML. Older studies may not reflect current attack vectors or ML capabilities.
Peer-Reviewed Journals or Conference Papers	Ensures the credibility, scientific quality, and research integrity of the included works. Peer-reviewed content is assumed to have undergone expert validation.
Explicit Use of ML Techniques for Ransomware Identification	Central to the research question. Only those studies that implement or evaluate ML models for detecting ransomware are considered.
English Language	Limits language bias and ensures that the research team can thoroughly evaluate and interpret the studies.

TABLE IV. EXCLUSION CRITERIA (IC) WITH EXPLANATION

Exclusion Criteria	Explanation
Studies not focused on ransomware or ML	Articles solely on other malware types or those not applying ML are out of scope.
Non-peer-reviewed Content (blogs, patents, Wikipedia, technical reports, etc.)	These sources may lack scientific validation and could introduce bias or inaccuracies.
Lack of Sufficient Experimental Results or Methodology	Studies that do not present clear evaluation metrics or lack methodological depth are excluded to ensure analytical rigor.
Duplicate or Redundant Publications	Repetitive data from the same study are removed to avoid skewing the synthesis.

D. Data Extraction and Synthesis

Data from the final 30 studies selected were extracted and synthesized in a structured and systematic manner. This step was vital in ensuring that the review would continuously collect data and yield insightful findings related to the study's objectives.

1) Data Extraction Process

To maintain consistency and reduce bias, a typical data extraction form was devised. All the selected pieces of research were thoroughly examined, and the necessary features of the studies, which are presented in Table V, were noted down:

TABLE V. DATA EXTRACTION WITH ATTRIBUTES AND DESCRIPTION

Attribute	Description
Authors and Publication Year	Identifies the origin and timeline of the research. Helps understand publication trends and prominent contributors in the field.
Dataset Description	Includes the name, source, size, nature (public/private), and characteristics of datasets used for training and testing ML models. Assessment of dataset diversity, relevance, and reusability is crucial.
Features Used	Details the feature selection method and the kinds of features (e.g., system calls, API usage, behavior logs, static code features) used in the ML model. It reveals which data representations turn out to be most effective for ransomware detection.

ML Algorithms Employed	Enumerate the ML or DL models employed, for instance, Decision Trees, Random Forest, SVM, CNNs, LSTM, XGBoost, etc. Understanding algorithmic diversity and figuring out which models demonstrate better performance is facilitated by this.
Evaluation Metrics	Concentrates on the performance measures that were used to evaluate the model's level of effectiveness. Precision, AUC-ROC, F1-score, recall, accuracy, and, in some cases, more specialized metrics like detection rate or false alarm rate are typical metrics that may be mentioned.
Key Findings	Summarizes the most important results, pointing out which models did the best and in what situations.
Limitations	Notes any shortcomings in the study such as dataset size, lack of generalizability, overfitting, or incomplete feature engineering.
Future Recommendations	The authors propose enhancements, future research trajectories, or application possibilities.

2) Data Synthesis Approach

In order to answer the research questions and provide a holistic picture of the current state of machine learning-based ransomware detection, the authors of the paper have come up with two complementary synthesis strategies:

a) Qualitative Synthesis

The qualitative examination included thematic coding and interpretation of the main findings from different studies.

Trends Identified:

- DL models such as CNN, RNN, and hybrid architectures are gaining popularity in the research community.
- Shifting from static to dynamic behavioral features for detection that is more resistant to attack.
- Use of ensemble models to improve accuracy and robustness.

Challenges Highlighted:

- Absence of standardized benchmark datasets across studies.
- Limited testing in real-time or real-world deployment scenarios.
- Imbalance in datasets causes biased classification.

Research Gaps Observed:

- Few studies choose Explainable AI (XAI) to improve interpretability.
- Most of the research works have not focused on lightweight models that can be used in resource-constrained environments.
- Lack of sufficient longitudinal research that evaluates the efficacy of models against progressively changing ransomware variants.

b) Quantitative Synthesis

Performance metrics such as F1-score, Accuracy, Precision, and Recall were gathered and merged whenever possible to evaluate:

- **Average Performance per Algorithm:** To find the best-performing models across all scenarios.
- **Metric Distribution Analysis:** To gauge variance and consistency of performance.
- **Dataset Usage Frequency:** Determining the most frequently employed datasets such as CICIDS, Virus Share, and custom-generated ransomware samples.

These findings were illustrated via bar charts for a more straightforward interpretation of the results.

VI. Literature Review

The section on the literature review discusses the use of machine learning (ML) for detecting ransomware. Various studies over the years 2018 through 2025 have examined static, dynamic, and hybrid analytic methods with different datasets, for instance, PE headers, API calls, transaction logs, and behavioural traces. Researchers have used an extensive array of ML techniques and DL models, often attaining elevated detection accuracy via improved features. A lot of recent work also focuses on making ML-based methods for fighting ransomware more understandable and able to detect threats in real time. Table VI gives a brief review of 30 important papers (2018–2025) on using ML to locate ransomware in different disciplines, including methodology, data description, features utilized, ML algorithms, assessment metrics, major results, limitations, and suggestions for the future.

TABLE VI. Comprehensive Summary of related work for ransomware detection using ML

S. No.	Authors and Year	Methodology	Dataset Description	Features Used	Ransomware Type	ML Algorithms	Evaluation Metrics	Key Findings	Limitations	Future Recommendations
1	Poudyal, Subedi & Dasgupta, 2018[6]	Reverse engineering + ML framework	Ransomware + benign binaries	ASM-level & DLL-level features	Windows binaries	8 supervised ML algorithms	Accuracy: 76–97%	Multi-level static analysis improves detection	Limited to static features; may fail under obfuscation	Include dynamic behavior for robustness
2	Hirano & Kobayashi, 2019[9]	Behavioral modeling via I/O storage patterns using Wayback Visor	Ransomware & benign app I/O logs	5-dimensional storage access features	Generic ransomware	Random Forest, SVM, KNN	F-measure: 98%	High behavioral detection accuracy	Focus on I/O only; storage-specific	Expand to include memory/network behaviors
3	Noorbehbahani & Saberi, 2020 [31]	Semi-supervised ML for ransomware	CIC And Mal 2017	OneR & Chi-squared selected features	Android ransomware	Semi-supervised Random Forest	Accuracy (not stated)	Effective with minimal labeled data	Performance not quantified clearly	Extend to other semi-supervised models
4	Almousa, Basavara	API-based ML detection for	Public ransomware &	API calls	Windows	Grid-optimized ML models	Accuracy: 99.1rand8 %	High accuracy for Windows systems	Windows-specific;	Apply to Linux/macO

	ju & Anwar, 2021 [32]	Windows ransomware	benign datasets		ransom ware				not cross-platform	S ransomware
5	Almomani, AlKhayer & Ahmed, 2021[14]	Android ransomware detection for v11	Custom dataset (1000 apps)	Permissions & API calls	Android ransom ware	Multiple ML classifiers	Accuracy: 98.3%	High accuracy after feature reduction	Dataset size limited	Scale with diverse Android versions
6	Usha et al. 2021 [33]	Comparative ML study with behavioral feature analysis	Not specified	API Calls, Registry, Network behavior	Windows ransom ware	KNN, Naïve Bayes, RF, DT	Accuracy, F1, Precision, Recall	Comparative analysis for best models	Dataset not detailed	Test on standard public datasets
7	Asaju et al. 2021[10]	ML-based detection/classification	Test/train dataset of ransomware	Static attributes of ransomware	Windows ransom ware	Naive Bayes, J48 Decision Tree	Accuracy: Naive Bayes (83.4%), J48 (97.6%)	High detection with J48	Static-only, lacks real-time behavior	Incorporate dynamic analysis
8	Almohaini et al. 2021[11]	Hybrid static + dynamic analysis of 100 ransomware samples across 10 families	API usage, code structure, behavioral traits	Ensemble methods combining static and dynamic approaches	Windows ransom ware	Accuracy: Static (40–55%), Dynamic (100%)	Dynamic analysis outperforms static significantly	Static fails under obfuscation; dynamic needs runtime execution	Static fails under obfuscation	Optimize runtime behavior extraction
9	Al-Haija & Alsulami, 2021 [28]	Used supervised learning (SNN and ODT) to classify ransomware Bitcoin transactions on a recent dataset	Bitcoin transaction metadata and behavioral patterns	Shallow Neural Networks, Optimizable Decision Trees	Crypto ransom ware	Accuracy, Confusion Matrix, PPV, TPR, Prediction Error	ODT achieved 99.9% (binary) and 99.4% (multiclass) accuracy	Dataset limited to Bitcoin; doesn't cover multi-currency ransomware trends	Doesn't cover altcoins	Extend to multi-currency frameworks
10	Hussain et al. 2022 [16]	Two-stage detection: binary (malicious/benign) and multi-class (family) using ML	BODMAS dataset (Aug 2019–Sept 2020)	Behavioral features from PE files	Windows ransom ware	Random Forest (binary), KNN, SVM, RF, DT, Gradient Boosting (ensemble)	99.48% (Stage 1), 92.49% (Stage 2)	High accuracy in both classification stages	May not adapt to new families	Include zero-day ransomware detection
11	Walker et al. 2022[34]	Time-series analysis of API call sequences	Custom API call dataset	Subset of API call sequences	Windows ransom ware	ELM, Online Sequential ELM (OS-ELM)	91% accuracy, 3s learning time	Higher accuracy and speed than LSTM	May miss complex long-term patterns	Compare with deep RNNs and Transformers
12	Molina et al. 2022 [17]	Behavioral modeling using paranoia API calls with NLP-based encoding	>3K samples from prominent ransomware families	Pre-attack evasion API calls (OoW features)	Generic ransom ware	Random Forest, NLP-based modeling	94.92% accuracy	Evasion behaviors are strong indicators of ransomware family	May not capture newer evasion tactics	Update feature sets regularly
13	Khurana 2023 [35]	Behavioral detection with SOM, RF, LSTM	System and network activity logs	Behavioral attributes	Generic ransom ware	SOM, Random Forest, LSTM	Accuracy: 93%, Precision: 97%	Adaptive learning, auto-response	Complex ensemble setup	Evaluate lightweight models for IoT
14	Herrera-Silva & Hernández-Álvarez 2023 [36]	Sandbox-based dynamic analysis with ML on ransomware & goodware; JSON logs with 50 features	Dynamic features (API calls, memory, file behavior)	Gradient Boosted Trees, Random Forest, Neural Networks	Windows ransom ware	Accuracy (avg >99%), 10-fold CV	Achieved >99% accuracy in distinguishing ransomware families	Limited sample size (20+20 artifacts); may not generalize to unknown variants	Limited generalizability	Use larger, real-world corpora

15	Zhuravchak & Dudykevych, 2023 [37]	Real-time detection using eBPF + ML + NLP	Mixed ransomware dataset	Packet and textual analysis features	Linux/ Windows ransomware	ML + NLP	Accuracy: 94.7%, Avg response: 2.3s	High detection with low false positives	Complex kernel-level integration	Integrate into cloud-native SOC tools
16	Kunku, Zaman & Roy, 2023 [38]	Behavioral ML detection/classification	Custom ransomware dataset	Behavior patterns, static indicators	Generic ransomware	XGBoost, Random Forest	Accuracy (not specified)	Effective family classification with ML	Accuracy not reported	Evaluate with standardized metrics
17	Sharma, Babbar & Vats, 2023 [39]	Empirical study using "BookMyShow" cybersecurity dataset for IoT ransomware detection	Extracted network activity features	Logistic Regression, KNN, XGBoost	IoT ransomware	Accuracy, Precision, Recall, F1-Score	XGBoost outperformed others with 96.62% accuracy	Dataset limited to specific scenario; may not generalize well	Case-specific dataset	Validate with diverse IoT datasets
18	Mohammed & Rezvy, 2024 [40]	XAI-based crypto ransomware detection	BitcoinHeist dataset	Transaction loops, volume patterns	Crypto ransomware	Random Forest + LIME	Accuracy, Precision, ROC-AUC	Interpretable model, high adaptability	Bitcoin-only focus	Extend to Ethereum and Monero chains
19	Ahmed et al. 2024 [29]	ML/DL-based detection using Kaggle dataset with 392,035 records covering 10 Android ransomware types	Full features and a reduced 19-feature subset	DT, SVM, KNN, FNN, TabNet, Ensemble of DT+SVM+KNN	Android ransomware	Accuracy, Precision, Recall, F1-score	DT had 97.24% accuracy, SVM achieved 100% recall	Feature selection may exclude subtle indicators; mobile-specific only	Mobile-specific, may exclude nuances	Broaden scope to hybrid OS threats
20	Li, Yang & Shao, 2024 [41]	Proposed ADC-TextGCN leveraging co-occurrence & PMI for API call sequences in ransomware detection	API co-occurrence, PMI, graph-based text features	Adaptive Diffusion Text GCN	Windows ransomware	Accuracy (>96.6%)	Outperforms CNN/RNN in modeling ransomware behavior sequences	Complex model architecture; limited interpretability	Complex graph structures	Simplify models for real-time use
21	Sharma, Ghosh & Singh, 2024 [42],	Transfer & ensemble learning using Xception	PE headers converted to images	Static PE header data as images	Windows ransomware	Xception (CNN + Transfer Learning)	Accuracy: 98.79%	Outperforms existing DL/ML models	Limited to static images	Combine with dynamic behaviors
22	Vaishnavi et al. 2024 [43]	VM-based monitoring with CNN	Harvard Dataverse V1 (VM traces)	CPU + Disk I/O under user loads	VM ransomware	2D CNN, others (5 total)	Accuracy: 98.66%	CNN outperforms state-of-art models	VM traces specific	Generalize to containerized systems
23	Rathina, Aadil & D, 2024 [44]	Hybrid static/dynamic detection for Android	Android apps (unspecified)	Threatening text, code, image	Android ransomware	Multiple ML models	Accuracy: 91%, low FN	Strong Android ransomware detection	Threat types not fully specified	Use multimodal learning
24	Rajululi et al. 2024 [45]	ELM + SVM with Seagull Optimization	Behavior-based dataset	Static & dynamic features	Windows ransomware	ELM, SVM, SGO	High detection accuracy	Fast and accurate threat detection	SGO optimization is complex	Assess other evolutionary methods
25	Rahman, Sabbir & Ghosh, 2024 [46]	Combines PCA for dimensionality reduction with SVM, Decision Tree, Random Forest	Behavioral patterns of malware	SVM, Decision Tree, Random Forest	Multiple ransomware	Accuracy: SVM (78.7%), DT (99.5%), RF (99.7%)	Decision Tree and RF show high detection accuracy	Behavior-only models can be bypassed with obfuscation	Behavior-based models bypassable	Integrate static + hybrid models
26	Alqarni et al. 2025 [22]	Static analysis with feature selection and comparative ML/DL models	Linux ELF files + 26 ransomware families (2015–2023)	Static features from ELF files	Generic ransomware	XGBoost, CatBoost, RF, Voting Ensemble, Bagging Ensemble, CNN	CNN: 97.78% accuracy	CNN and Bagging Ensemble perform best	Linux-specific focus	Extend to cross-platform malware

27	Zakaria et al. 2025 [23]	Detection of pre-encryption behavior using multiple classifiers	RENTAKA (pre-encryption ransomware activity dataset)	Pre-encryption behavioral features	Windows ransomware	RF, SVM, k-NN, Logistic Regression, Decision Tree	RF: 96.29%, SVM/LR: 94.98%; Precision, Recall, F1, AUC	RENTAKA framework shows early detection capability	Dataset scope limited	Combine with post-encryption analysis
28	Kipanga et al. 2025 [1]	Feature selection analysis for ML and DL on PE-header data	2 ransomware datasets: 2,157 PE-header + 29,807 malware samples	PE-header features; reduced via Chi-Square, PCA, MI	Linux ransomware	NB, LR, ET, SVM, RF, LSTM	Accuracy: 99.9% (ML), 98.9% (DL), RF + Chi-Square: 99.9%	Feature selection improves detection and reduces overhead	Limited generalization to new malware	Test with real-time threats
29	Adnan et al. 2025[20]	Interpretable ML with ETW logs using XAI (XRGuard)	Event Tracing for Windows (ETW) logs	File I/O activity patterns	Windows ransomware	SHAP, LIME, ML classifiers (unspecified)	99.69% accuracy, 0.5% FPR	Transparent, interpretable and high-performance detection	Model types not detailed	Clarify algorithms; validate across OS
30	Prajapati et al. 2025 [47]	Static + dynamic analysis using ML models	System and file activity logs	File/network activity and system calls	Windows ransomware	SVM, RF, XGBoost	97.333% accuracy	Early detection before encryption; strong classification performance	Dataset granularity unclears	Standardize benchmark datasets

VII. Findings with Machine Learning Models

The experiments that were looked at provide a complete process for applying ML to find ransomware, including everything from collecting data to testing the model. The goal of these studies is to improve the early and accurate detection of ransomware by employing a variety of machine learning techniques. Datasets include both publicly accessible corpora (e.g., BODMAS, CICAndMal2017, Bitcoin Heist, Harvard Data verse, RENTAHA) and bespoke collections, addressing Windows, Linux, Android, IoT, and cryptocurrency contexts. Raw data was refined using preprocessing methods such feature selection, dimensionality reduction (PCA, Chi-Square, Mutual Information), and normalization. PE headers, permissions, and API calls were examples of static features; I/O patterns, memory use, system logs, ETW traces, and hybrid combinations were examples of dynamic features. In addition to ensembles and interpretable AI approaches (SHAP, LIME), the ML models used include both conventional algorithms (SVM, RF, Decision Tree, KNN, Logistic Regression) and sophisticated models (CNN, LSTM, Extreme Learning Machine, TextGCN, Xception, TabNet). Accuracy is a typical evaluation criterion, and most research get an accuracy rate of more than 99.69%. Comparative study demonstrates that hybrid static-dynamic techniques and behavior-based models often surpass static-only solutions, whereas interpretable models provide transparency for practical implementation. The results show that combining rich behavioral data, hybrid ML pipelines, and explainable AI makes it much easier to locate and adapt to ransomware on a wide range of platforms.

A. Machine Learning Models

In this segment, it presents various ML models that have demonstrated high accuracy in ransomware identification. These models utilize diverse learning strategies, ranging from ensemble methods to DL, to effectively classify malicious activities. The following ML-DL techniques are:

- **Random Forest (RF):** This ensemble learning approach is used to build many DTs during training, and then outputs the individual trees' mean prediction (regression) or mode of classes (classification). It increases accuracy and decreases overfitting [16].
- **Support Vector Machine (SVM):** An method for supervised learning that determines the best hyperplane for grouping data into distinct groups. Using kernel methods, it performs well on both linear and non-linear data [1].
- **J48 Classifier:** The C4.5 decision tree technique is used for classification in this open-source Java implementation. It divides the dataset according to the characteristic that yields the most information [10].
- **Extreme Learning Machine (ELM):** A fast-learning technique that provides high-speed training for single-layer feedforward neural networks with analytically calculated output weights and input weights allocated at random [34].
- **Self-Organizing Map (SOM):** An unsupervised learning technique for dimensionality reduction and grouping. While maintaining topological characteristics, it projects high-dimensional data onto a low-dimensional (often 2D) grid [35].
- **Gradient Boosted Trees (GBT):** Sequential decision trees are constructed using an ensemble approach in which each tree fixes the mistakes of the one before it. It is frequently utilized in structured data activities and is renowned for its excellent predicted accuracy [36].
- **Neural Network (NN):** Repertoire of algorithms with layers of linked nodes (neurones) that are modelled after the structure of the human brain. used in huge and complicated datasets for tasks like pattern recognition, regression, and classification [36].
- **XGBoost (Extreme Gradient Boosting):** For improved performance and accuracy in classification and regression problems, gradient boosting is

optimized using regularization, parallel processing, and tree pruning [39].

- **TabNet:** A DL architecture for tabular data that selects which characteristics to employ for reasoning at each decision step by using sequential attention. It's interpretable and suitable for resource-constrained environments [29].
- **Graph Convolution Network (TextGCN):** A graph-based neural network designed to learn from structured data like text. TextGCN uses nodes for words and documents, applying graph convolutions for improved text classification [41].
- **Xception CNN:** A cutting-edge deep convolutional neural network architecture that utilizes depthwise separable convolutions. It notably elevates both the efficiency and performance, especially in the case of image classification tasks [42].
- **2D CNN:** A CNN that is purpose-built for 2D input data (e.g., images or trace matrices). The network extracts spatial features by the application of convolutional filters over the height and width of input matrices [43].
- **API-Based ML (Grid-Optimized):** A method that utilizes ML algorithms that are accessible via APIs. The model parameters are fine-tuned by grid search for best performance. It is a technique that is frequently employed in cloud-based or automated ML environments [32].
- **SNN (Simple Neural Network):** A simplistic structure of a neural network comprising a single or few layer, typically employed in a recognition task of a simple pattern. It does not have the complexity of deep networks but is quicker and more easily understood.
- **Optimized Decision Tree (ODT):** A traditional decision tree, enhanced by pruning, depth control, or other optimization methods to lessen overfitting and increase accuracy in prediction [28].

The detection accuracies of the subsequent models are summarized in Table VII, along with the performance of different machine learning methods in identifying ransomware. These models achieved high accuracy across various datasets and detection techniques. Accuracy in ML is a performance metric used to measure how well a model correctly predicts outcomes. It calculates as Equation (1):

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Where:

- **True Positives (TP):** These are malicious (ransomware) samples that model correctly identified as ransomware.
- **True Negatives (TN):** The model accurately identified these files as benign (harmless).
- **False Positives (FP):** These are benign files that the model wrongly classifies as ransomware.
- **False Negatives (FN):** These are ransomware files that the model fails to detect, wrongly classifying them as benign.

Accuracy stipulates how effectively model distinguishes between ransomware and benign files.

TABLE VII. ACCURACY PERFORMANCE OF ML MODELS FOR RANSOMWARE DETECTION

Model / Approach	Accuracy (%)
API-Based ML (Grid-Optimized) [32]	99.18
Android Detection (Multiple ML classifiers) [14]	98.3
J48 Classifier[10]	97.6
Hybrid Static + Dynamic Analysis[11]	100.0 (Dynamic)
Bitcoin Ransomware Classifier (SNN, ODT) [28]	99.9 (Binary)
Two-Stage Classifier (RF, SVM) [1]	99.48
ELM (Extreme Learning Machine) [34]	91.0
Behavioral + NLP Modeling[17]	94.92
SOM-Based Behavioral Detection [35]	93.0
GBT, RF, NN (Dynamic Analysis) [36]	>99.0
Real-time Detection (eBPF + NLP) [37]	94.7
XGBoost (IoT Dataset) [39]	96.62
TabNet (Android Detection)	97.24
ADC-TextGCN (Graph Neural Network) [41]	>96.6
Xception CNN[42]	98.79
2D CNN (VM Traces) [43]	98.66
Android Hybrid Detection[11]	91.0
Random Forest + PCA[16]	99.7
Voting Ensemble (CNN, RF, XGBoost) [22]	97.78
RENTAHA Dataset (Random Forest) [23]	96.29
Feature Selection + RF (Chi-Square) [1]	99.9
Explainable AI (ETW Logs, SHAP, LIME) [20]	99.69
Hybrid Logs + SVM/XGBoost[47]	97.33

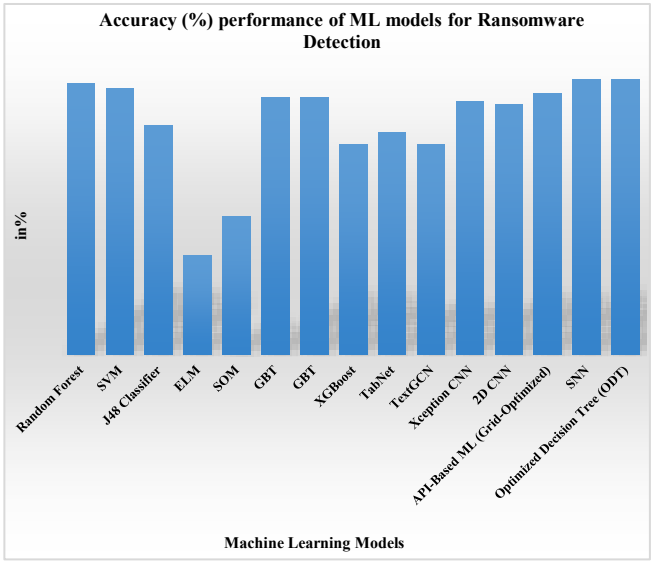

Fig. 3. Bar graph of accuracy comparison of ML models for ransomware detection

The comparative examination of ML models for ransomware detection demonstrates that several techniques attain very high performance, as seen in Fig. 3. Random Forest, Simple Neural Network (SNN), and Optimized Decision Tree (ODT) are all models that have an accuracy of 99.9%, which shows that they are very good at binary classification problems. Gradient Boosted Trees (GBT) and Neural Networks (NN) are likewise quite successful, with accuracy rates of over 99%. Support Vector Machine (SVM) is close behind with 99.48%. Other deep learning models, such as Xception CNN (98.79%) and 2D CNN (98.66%), illustrate how convolutional architectures may be used to understand complicated ransomware behavior. TabNet (97.24%), J48 Classifier (97.6%), and XGBoost (96.62%) all do well, especially when working with structured tabular or decision-based data. In the meantime, SOM and ELM perform moderately, with respective accuracies of 93.0% and 91.0%. Hyperparameter tuning is one of the main factors behind the 99.18% success rate of API-based ML with grid optimization.

In general, the results are indicative of the fact that both advanced DL methods and standard ML models are capable of yielding very high accuracy in the detection task, especially if they are further enhanced by feature selection and tuning strategies.

VIII. Challenges and Future Directions

Table VIII indicates that ML-based ransomware detection has various issues such ares evasions methods, unbalanced and small datasets, poor generalization, and a high number of false positives. To solve these problems, next research should focus on the development of robust, adversarial-resistant models, the application of data augmentation via GANs, and the support of cross-platform training. Real-time, low-latency detection can be enhanced by lightweight models such as TabNet. To contain the rapid uprising of ransomware and ensure that detection is stable, it is also necessary to use SHAP and LIME, open benchmark datasets, scalable cloud-based frameworks, and continual learning.

TABLE VIII. Challenges and Future Directions in ML-Based Ransomware Detection

Challenges	Future Directions
Evasion techniques by ransomware to bypass detection	Develop adversarial-resistant and robust detection models
Imbalanced datasets and limited ransomware samples	Use advanced data augmentation (e.g., GANs) and synthetic data generation
Limited generalization across environments and devices	Train on cross-platform, real-world, and diverse datasets
High false positives in behavioral models	Improve precision using hybrid models and contextual feature analysis
Real-time detection with minimal latency	Deploy lightweight, edge-compatible ML models (e.g., TabNet, TinyML)
Lack of labeled datasets and publicly available benchmarks	Promote open-source datasets and collaborative benchmarking initiatives
Scalability issues in enterprise-level environments	Employ distributed learning and cloud-based ML frameworks
Explainability and model transparency	Integrate Explainable AI (XAI) tools like SHAP and LIME
Rapid evolution of ransomware variants	Implement continual learning and self-updating AI models
Dataset contamination with benign/malicious overlaps	Apply refined preprocessing and robust validation protocols

IX. Conclusion

Ransomware attacks, which have been a major cause of harm to computer systems, have been the main reason for unauthorized access, data loss, and high costs to businesses, both in terms of money and their reputation. In response, researchers have developed a number of fast and accurate detection methods. Besides providing a timeline and background of ransomware attacks, this study communicates the problem's origin. This paper thoroughly and systematically reviews ML-based ransomware detection methods by referencing 30 primary research articles published in the period of 2018-2025. It categorizes the detection methods as static, dynamic, hybrid, and intelligent ML-based ones. Subsequently, it examines the common features such as API calls, system logs, permissions, and behavioral patterns across various platforms like Windows, Android, IoT, and Linux. The article also compares the performance and efficiency of different ML-DL models while pointing out critical research gaps, e.g., dataset imbalance, poor generalization, lack of real-time adaptability, and shortage of benchmark datasets. One of the primary advantages of this paper lies in the fact that it emphasizes XAI techniques integration to make models more transparent and reliable, which is a prerequisite for their application in the sensitive domain. Moreover, the study proposes future work directions, e.g., lightweight model design, attack-tolerant architectures, and data augmentation. These provide a clear route to achieving the goal of ransomware detection systems that are not only strong, scalable, but also adaptable. At some point in the future, it will be feasible to spot ransomware infections in their early stages, and ML algorithms may then be used to reduce the rates of false positives and false negatives that occur prior to encryption. Furthermore, heuristic-based methods might be utilized for ransomware detection that is new or remains unidentified.

References

[1] R. Kipanga and F. Khennou, "Leveraging Feature Selection and Deep Learning for Accurate Malware and Ransomware Detection in PE Files," in *2025 13th International Symposium on Digital Forensics and Security (ISDFS)*, IEEE, Apr. 2025, pp. 1–6. doi: 10.1109/ISDFS65363.2025.11012017.

[2] H. Mandaviya and S. Sathwara, "Cyber Terrorism-Related Multimedia Detection Using Deep Learning—A Survey," in *Proceedings of International Conference on Recent Trends in Machine Learning, IoT, Smart Cities and Applications: ICMISC 2020*, 2021, pp. 135–143. doi: 10.1007/978-981-15-7234-0_11.

[3] S. Amrale, "A Novel Generative AI-Based Approach for Robust Anomaly Identification in High-Dimensional Dataset," *Int. J. Adv. Res. Sci. Commun. Technol.*, vol. 4, no. 2, pp. 709–721, Oct. 2024, doi: 10.48175/IJARSCT-19900D.

[4] G. Sarraf, "Autonomous Ransomware Forensics: Advanced ML Techniques for Attack Attribution and Recovery," *Int. J. Adv. Res. Sci. Commun. Technol.*, vol. 3, no. 3, pp. 1377–1390, Jul. 2023, doi: 10.48175/IJARSCT-11978W.

[5] S. Sathwara, N. Dutta, and E. Pricop, "IoT Forensic A digital investigation framework for IoT systems," in *2018 10th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, IEEE, Jun. 2018, pp. 1–4. doi: 10.1109/ECAI.2018.8679017.

[6] S. Poudyal, K. P. Subedi, and D. Dasgupta, "A Framework for Analyzing Ransomware using Machine Learning," in *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, IEEE, Nov. 2018, pp. 1692–1699. doi: 10.1109/SSCI.2018.8628743.

[7] S. K. Chintagunta and S. Amrale, "Enhancing Cloud Database Security Through Intelligent Threat Detection and Risk Mitigation," *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, vol. 8, no. 3, pp. 756–768, Dec. 2022.

[8] S. Thangavel, "AI Enhanced Image Processing System For Cyber Security Threat Analysis," 2024

[9] M. Hirano and R. Kobayashi, "Machine Learning Based Ransomware Detection Using Storage Access Patterns Obtained From Live-forensic Hypervisor," in *2019 Sixth International Conference on Internet of Things: Systems, Management and Security (IOTSMS)*, IEEE, Oct. 2019, pp. 1–6. doi: 10.1109/IOTSMS48152.2019.8939214.

[10] C. B. Asaju, D. Otoo-Arthur, R. O. Orah, and F. Sekyi-Dadson, "Development of a Machine Learning Model for Detecting and Classifying Ransomware," in *2021 1st International Conference on Multidisciplinary Engineering and Applied Science (ICMEAS)*, IEEE, Jul. 2021, pp. 1–5. doi: 10.1109/ICMEAS52683.2021.9692402.

[11] R. Almohaini, I. Almomani, and A. AlKhayer, "Hybrid-Based Analysis Impact on Ransomware Detection for Android Systems," *Appl. Sci.*, vol. 11, no. 22, Nov. 2021, doi: 10.3390/app112210976.

[12] V. Shewale, "Beyond EDR: Exploring the rise of XDR for unified threat detection and response," *World J. Adv. Eng. Technol. Sci.*, vol. 15, no. 2, pp. 380–386, May 2025, doi: 10.30574/wjaets.2025.15.2.0551.

[13] S. Sathwara and C. Parekh, "Distributed Denial of Service Attacks – TCP Syn Flooding Attack Mitigation," *Int. J. Adv. Res. Comput. Sci.*, vol. 8, no. 5, pp. 1–5, 2017, doi: 10.26483/ijarcs.v8i5.3391.

[14] I. Almomani, A. AlKhayer, and M. Ahmed, "An Efficient Machine Learning-based Approach for Android v.11 Ransomware Detection," in *2021 1st International Conference on Artificial Intelligence and Data Analytics (CAIDA)*, IEEE, Apr. 2021, pp. 240–244. doi: 10.1109/CAIDA51941.2021.9425059.

[15] H. Cyril, "AI-Driven Anomaly Detection, Outage Prediction, and Self- Healing in Telecom Provisioning Systems," *Int. J. Appl. Math.*, vol. 38, no. 12s, pp. 2817–2832, Dec. 2025, doi: 10.12732/ijam.v38i12s.1589.

[16] M. J. Hussain, A. Shaoor, S. Baig, A. Hussain, and S. A. Muqurrab, "A hierarchical based ensemble classifier for behavioral malware detection using machine learning," in *2022 19th International Bhurban Conference on Applied Sciences and Technology (IBCAST)*, IEEE, Aug. 2022, pp. 702–706. doi: 10.1109/IBCAST54850.2022.9990203.

[17] R. M. A. Molina, S. Torabi, K. Sarieddine, E. Bou-Harb, N. Bouguila, and C. Assi, "On Ransomware Family Attribution Using Pre-Attack Paranoia Activities," *IEEE Trans. Netw. Serv. Manag.*, vol. 19, no. 1, pp. 19–36, Mar. 2022, doi: 10.1109/TNSM.2021.3112056.

[18] A. R. Bilipelli, "Forecasting the Evolution of Cyber Attacks in FinTech Using Transformer-Based Time Series Models," *Int. J. Res. Anal. Rev.*, vol. 10, no. 3, pp. 383–389, 2023.

[19] M. Pandya and S. Sathwara, "A Survey on Detection & Prevention Methods of Ransomware in Cyber Security," *Int. J. Eng. Dev. Res.*, vol. 10, no. 2, pp. 96–99, 2022.

[20] M. A. Alvi and Z. Jalil, "XRGuard: A Model-Agnostic Approach to Ransomware Detection Using Dynamic Analysis and Explainable AI," *IEEE Access*, vol. 13, pp. 53159–53170, 2025, doi: 10.1109/ACCESS.2025.3553562.

[21] S. Chatterjee, "Advanced Malware Detection in Operational Technology: Signature-Based Vs. Behaviour-Based Approaches," *ESP J. Eng. Technol. Adv.*, vol. 1, no. 2, pp. 272–279, 2021, doi: 10.56472/25832646/JETA-V1I2P128.

[22] S. I. Alqarni, A. O. Aljahdali, and S. Alsafari, "Linux Ransomware Analysis and Detection Using Machine Learning and Deep Learning Approaches," in *2025 International Conference on Innovation in Artificial Intelligence and Internet of Things (AIIT)*, IEEE, May 2025, pp. 1–8. doi: 10.1109/AIIT63112.2025.11082970.

[23] W. Z. A. Zakaria, M. F. Abdollah, O. Mohd, M. N. A. Aziz, S. M. W. M. S. M. M. Yassin, and S. R. M. Kassim, "RENTAKA: Detecting Ransomware at Pre-Attack Stage Using Machine Learning Approach," in *2025 IEEE 15th Symposium on Computer Applications & Industrial Electronics (ISCAIE)*, IEEE, May 2025, pp. 467–471. doi: 10.1109/ISCAIE64985.2025.11081161.

[24] V. Prajapati, "Enhancing Threat Intelligence and Cyber Defense through Big Data Analytics: A Review Study," *J. Glob. Res. Math. Arch.*, vol. 12, no. 4, pp. 1–6, 2025.

[25] H. P. C. Kapadia and K. C. Chittoor, "Quantum Computing Threats to Web Encryption in Banking," *Int. J. Nov. Trends Innov.*, vol. 2, no. 12, pp. a197–a204, 2024.

[26] S. Kumara, "AI-Driven Threat Identification and Response: Implications for Secure and Scalable Telecom Infrastructure," *Int. J. Adv. Res. Sci. Commun. Technol.*, vol. 5, no. 4, Dec. 2025, doi: 10.48175/IJARSCT-30567.

[27] R. Patel, "Automated Threat Detection and Risk Mitigation for ICS (Industrial Control Systems) Employing Deep Learning in Cybersecurity Defence," *Int. J. Curr. Eng. Technol.*, vol. 13, no. 06, pp. 584–591, Dec. 2023, doi: 10.14741/ijcet/v.13.6.11.

[28] Q. A. Al-Haija and A. A. Alsulami, "High Performance Classification Model to Identify Ransomware Payments for Heterogeneous Bitcoin Networks," *Electronics*, vol. 10, no. 17, Aug. 2021, doi: 10.3390/electronics10172113.

[29] A. A. Ahmed, A. Shaahid, F. Alnasser, S. Alfaddagh, S. Binagag, and D. Alqahtani, "Android Ransomware Detection Using Supervised Machine Learning Techniques Based on Traffic Analysis," *Sensors*, vol. 24, no. 1, Dec. 2023, doi: 10.3390/s24010189.

[30] P. Yan and T. T. Khoei, "Securing the internet of things: A comprehensive review of ransomware attacks, detection, countermeasures, and future prospects," *Franklin Open*, vol. 11, Jun. 2025, doi: 10.1016/j.fraope.2025.100256.

[31] F. Noorbehbahani and M. Saberi, "Ransomware Detection with Semi-Supervised Learning," in *2020 10th International Conference on Computer and Knowledge Engineering (ICCKE)*, IEEE, Oct. 2020, pp. 024–029. doi: 10.1109/ICCKE50421.2020.9303689.

[32] M. Almousa, S. Basavaraju, and M. Anwar, "API-Based Ransomware Detection Using Machine Learning-Based Threat Detection Models," in *2021 18th International Conference on Privacy, Security and Trust (PST)*, IEEE, Dec. 2021, pp. 1–7. doi: 10.1109/PST52912.2021.9647816.

[33] G. Usha, P. Madhavan, M. V. Cruz, N. A. S. Vinoth, Veena, and M. Nancy, "Enhanced Ransomware Detection Techniques using Machine Learning Algorithms," in *2021 4th International Conference on Computing and Communications Technologies (ICCCT)*, IEEE, Dec. 2021, pp. 52–58. doi: 10.1109/ICCCT53315.2021.9711906.

[34] A. Walker, R. M. Shukla, T. Das, and S. Sengupta, "Ohana Means Family: Malware Family Classification using Extreme Learning Machines," in *2022 IEEE 19th Annual Consumer Communications & Networking Conference (CCNC)*, IEEE, Jan. 2022, pp. 534–542. doi: 10.1109/CCNC49033.2022.9700583.

[35] S. Khurana, "Ransomware Threat Detection and Mitigation using Machine Learning Models," in *2023 IEEE International Conference on ICT in Business Industry & Government (ICTBIG)*, IEEE, Dec. 2023, pp. 1–6. doi: 10.1109/ICTBIG59752.2023.10456343.

[36] J. A. Herrera-Silva and M. Hernández-Álvarez, "Dynamic Feature Dataset for Ransomware Detection Using Machine Learning Algorithms," *Sensors*, vol. 23, no. 3, Jan. 2023, doi: 10.3390/s23031053.

[37] D. Zhuravchak and V. Dudykevych, "Real-Time Ransomware Detection by Using eBPF and Natural Language Processing and Machine Learning," in *2023 IEEE 5th International Conference on Advanced Information and Communication Technologies (AICT)*, IEEE, Nov. 2023, pp. 1–4. doi: 10.1109/AICT61584.2023.10452697.

[38] K. Kunku, A. Zaman, and K. Roy, "Ransomware Detection and Classification using Machine Learning," in *2023 IEEE Symposium Series on Computational Intelligence (SSCI)*, IEEE, Dec. 2023, pp. 862–866. doi: 10.1109/SSCI52147.2023.10371924.

[39] A. Sharma, H. Babbar, and A. K. Vats, "Ransomware Attack Detection in the Internet of Things using Machine Learning Approaches," in *2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC)*, IEEE, May 2023, pp. 553–559. doi: 10.1109/ICAAIC56838.2023.10141036.

[40] R. Mohammed and S. Rezvy, "Transparent Ransomware Detection in Bitcoin Transactions: Leveraging Machine Learning and Explainable AI," in *2024 7th International Conference on Algorithms, Computing and Artificial Intelligence (ACAI)*, IEEE, Dec. 2024, pp. 1–5. doi: 10.1109/ACAI63924.2024.10899728.

[41] J. Li, G. Yang, and Y. Shao, "Ransomware Detection Model Based on Adaptive Graph Neural Network Learning," *Appl. Sci.*, vol. 14, no. 11, May 2024, doi: 10.3390/app14114579.

[42] R. Sharma, M. Ghosh, and A. Singh, "Ensemble-Based Transfer Learning for Enhanced Ransomware Detection Using PE Header," in *2024 3rd Edition of IEEE Delhi Section Flagship Conference (DELCON)*, IEEE, Nov. 2024, pp. 1–6. doi: 10.1109/DELCON64804.2024.10866864.

[43] D. Vaishnavi, R. Rajakumar, S. RanjeethKumar, and M. Anuradha, "Optimal Machine Learning Model to Detect Ransomware on Processor and Disk," in *2024 International Conference on IoT Based Control Networks and Intelligent Systems (ICICNIS)*, IEEE, Dec. 2024, pp. 1296–1303. doi: 10.1109/ICICNIS64247.2024.10823178.

[44] X. A. Rathina, M. Aadil, and M. D, "Ransomware Detection in Android Using Machine Learning," in *2024 OITS International Conference on Information Technology (OCIT)*, IEEE, Dec. 2024, pp. 180–185. doi: 10.1109/OCIT65031.2024.00040.

[45] G. G. Rajululi, R. Geetha, J. Bebitha, and S. M. Basha, "Optimized Extreme Learning Machine Framework for Efficient Ransomware Detection Using Seagull Optimization-Based Feature Selection," in *2024 International Conference on Emerging Research in Computational Science (ICERCS)*, IEEE, Dec. 2024, pp. 1–5. doi: 10.1109/ICERCS63125.2024.10895166.

[46] M. S. Rahman, M. S. A. Sabbir, and S. Ghosh, "Ransomware Attack Detection using Machine Learning Approaches," in *2024 3rd International Conference for Innovation in Technology (INOCON)*, IEEE, Mar. 2024, pp. 1–7. doi: 10.1109/INOCON60754.2024.10512276.

[47] Y. Prajapati, O. P. Suthar, K. Gosai, and S. K. Singh, "Smart City Cybersecurity: Leveraging Machine Learning for Advanced Ransomware Detection and Prevention," in *2025 International Conference on Pervasive Computational Technologies (ICPCT)*, IEEE, Feb. 2025, pp. 808–813. doi: 10.1109/ICPCT64145.2025.10941048.

