IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 22, 2022, accepted June 1, 2022, date of publication June 8, 2022, date of current version June 15, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3181278

Utilizing Cyber Threat Hunting Techniques to Find Ransomware Attacks: A Survey of the State of the Art

FATIMAH ALDAUIJI, OMAR BATARFI, AND MANAL BAYOUSEF
Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia
Corresponding author: Fatimah Aldauiji (faldauiji0001@stu.kau.edu.sa)

ABSTRACT Ransomware is one of the most harmful types of cyber attacks that cause major concerns on a global scale. It makes the victims' resources unusable by encrypting data or locking systems to extort ransom payments. Ransomware has variant families that continue to evolve. Moreover. cybercriminals use advanced techniques to develop ransomware, making it harder for anti-malware detection systems to detect them. Ransomware solutions need the capabilities of timely and effective detection and response to discover uncommon behavior before losing sensitive data. Cyber threat hunting (CTH) is a novel proactive malware detection approach that includes cyber threat intelligence (CTI) methods and data analysis methods. However, most present CTH solutions depend on internal data sources and reactive techniques to detect unusual activities. An effective CTI technique is required to obtain knowledge from external data sources and combine it with internal sources to enhance the hunting capabilities. Then, using the optimal data analysis technique is needed for the CTH approach to obtain valuable insights into abnormal patterns in running activities in the early stages. In this study, we investigate using a practical CTI approach and different CTH models. Subsequently, we discussed ransomware research directions to detect known and unknown ransomware attacks. Also, we discussed the available ransomware datasets used in present ransomware studies.

INDEX TERMS Ransomware, cyber threat hunting, cyber threat intelligence, malware analysis, machine learning, deep learning.

I. INTRODUCTION

In 2020, ransomware attacks against healthcare systems increased during the COVID-19 pandemic. Healthcare institutions face disruption of medical services and long-term consequences because of ransomware attacks [1]. Also, ransomware attacks affect individuals and organizations to gain more money [55]. In 2021, 66 percent of surveyed companies were attacked by ransomware, up from 37 percent in 2020 [2]. Ransomware is a form of malware that uses encryption methods to encrypt a user's files or locks the system. Ransomware attacks aim is to gain payments from the victim to unlock the system or decrypt the victim's data files [3]. In 1989, the first ransomware was created by Joseph Popp, when he initiated a ransomware attack called AIDS, also known as PC Cyborg. He shared floppy disks with several AIDS researchers containing the malicious scripts [4].

Subsequently, ransomware attacks have continued to evolve using different tactics and techniques. Crypto-ransomware and locker-ransomware are the two main types of ransomware. In a crypto-ransomware attack, the attacker encrypts the victim's valuable data using robust encryption methods, such as Rivest–Shamir–Adleman (RSA) or Advanced Encryption Standard (AES), and locks them until the victim pays a ransom. In contrast, instead of encrypting data files, locker ransomware locks the victim's system and requests a ransom payment to unlock it. Attackers primarily design ransomware attacks for money extortion from the victims. Pre-paid vouchers, premium rate SMS or calls, and online purchases are examples of early ransom payment techniques. Cryptocurrencies or virtual currencies, such as Bitcoin, are currently one of the most widely used ransom payment methods [50].

The associate editor coordinating the review of this manuscript and approving it for publication was Yizhang Jiang.

In recent years, security researchers have been investigating and tracking the evolution of various ransomware types. Famous ransomware families include TeslaCrypt, CryptoWall, Locky, Cerber, and WannaCry [5]. CryptoWall appeared in 2014 as locker ransomware that spreads by phishing emails, exploit kits, and infected attachments. In 2015, TeslaCrypt was distributed by exploit kits, and it used an AES encryption algorithm to encrypt all user data. Moreover, Cerber is frequently distributed by exploit kits and exchanged on hacker forums as ransomware-as-a-service (RaaS). Cerber starts by encrypting user data using the AES algorithm without connecting to the command and control (C&C) server. Locky ransomware came into view in 2016 and included embedded macros with Microsoft Office documents. As a custom method, encrypted communication is used by Locky ransomware for Tor and Bitcoin payments. WannaCry was one of the most severe ransomware attacks in 2017, affecting more than 300,000 computers in over 100 countries [6]. WannaCry employs the EternalBlue exploit tool set to exploit the SMB vulnerability in Microsoft Windows and uses the AES algorithm to encrypt data files [7].

Ransomware could attack various platforms, including PCs, mobile devices, and Internet of Things (IoT) devices. Ransomware attacks on mobile devices have increased since 2017, and they have a variety of impacts, such as stealing important data or locking mobile devices. Ransomware attacks on IoT devices have recently become a challenge [8]. Currently, adversaries do not need to develop their ransomware; instead, they can purchase it from another adversary, a practice known as Ransomware-as-a-Service (RaaS). RaaS makes it easier for inexperienced actors to create and launch ransomware attacks.

Most existing ransomware solutions are reactive. File hashes, IP addresses, and DNS records are examples of known indicators used in reactive approaches [9]. Adopting reactive methods to identify ransomware can result in data and system damages. However, employing a proactive defense strategy is the safest alternative for ransomware attacks. Proactive approaches use indicators and behavioral artifacts to identify malicious threats. Registry paths, system calls, user and authentication records, DNS queries and responses, and other run-time activities are captured by behavioral indicators [10], [11].

Cyber Threat Hunting (CTH) is a proactive approach utilized to secure critical assets. CTH is performed proactively in the environment, without any threat alerts. [12]. CTH's major purpose is to identify hidden threats, disable them, and establish policies to avoid them in the future. It integrates cyber threat intelligence and data analysis methods to find evidence of a threat in a network. Cyber Threat Intelligence (CTI) is the process of seeking and collecting information beyond what is readily available, such as event logs [13]. Evidence-based knowledge outside security logs is necessary to adopt a proactive step and help in the decision-making process.

Ransomware has received much attention recently due to the rise in ransomware attacks on individuals, businesses, and governments worldwide. Ransomware attacks are constantly changing and becoming more sophisticated than before. From this perspective, This study investigates the literature review of CTI and CTH for both malware and ransomware works with their limitations and gaps. Also, this study will investigate the current CTH techniques and the utilization of CTI techniques. Related studies and available datasets were reviewed to highlight the main trends. In addition, potential research directions of ransomware studies are described.

The remainder of this paper is organized as follows: section 2 summarizes ransomware studies and the existing CTI and CTH techniques. Section 3 provides an overview of cyber threat intelligence techniques. Section 4 presents a detailed overview of malware analysis approaches. Section 5 discusses cyber threat hunting techniques. Section 6 discusses the evolution of ransomware attacks and research directions. Section 7 discusses datasets of ransomware detection studies. Finally, Section 8 provides the conclusion of this study.

II. BACKGROUND

Ransomware targets computer, mobile, cloud-based, IoT, ICS, and other systems as extortion-based cyber threat [14], [15]. Researchers have developed several taxonomies to help understand how ransomware operates. Specific countermeasures should be implemented to secure different digital assets. Ransomware is classified into two categories based on confiscated resources: locker-ransomware and crypto-ransomware. In a Locker-ransomware attack, the victim will not be able to reach system services; however, data will not be compromised. Locker-ransomware is classified by the type of non-data resources it encrypts, such as operating systems, applications, services, user interfaces, and other utilities. Crypto-ransomware encrypts data resources and requests a ransom payment from users. Crypto-ransomware is classified into three types based on the encryption process: symmetric, asymmetric, and hybrid.

A deep understanding of the ransomware attack steps is required to discover an effective solution. Infection, installation, communication, execution, extortion, and emancipation are the most common ransomware attack phases. Figure 1 depicts the steps involved in a typical ransomware attack.

- *Infection phase:* This phase begins when the malicious ransomware code enters the victim's system. Different infection vectors for ransomware attacks include affiliate programs, exploit kits, and email-based malvertising campaigns [16].
- *Installation phase:* This phase begins following ransomware infection, when the ransomware installs itself on the system and takes control without attracting attention.
- *Communication phase:* This phase starts when the ransomware establishes an initial connection with the main adversary to carry out the following level of actions.

The ransomware initiates a connection with a command and control (C&C) server.

- *Execution phase:* This phase starts when the ransomware begins to carry out malicious operations on the victim's resources. These malicious ransomware actions include encrypting data, deleting files, accessing file systems, locking procedures, and modifying master boot records (MBRs) [17].
- *Extortion phase:* This phase starts when the ransomware notifies users that they have been attacked and must obey the attacker's instructions. A ransom note is shown to the victim, which uses social engineering techniques to persuade them to pay the ransom.
- *Emancipation phase:* This phase starts after receiving the ransom payment when the attacker unlocks system resources. Following ransom payment, attackers would send a link to infected victims that contains a specific decryption tool for some crypto-ransomware attacks.

Security researchers have investigated two defense approaches for ransomware attacks: signature-based and behavior-based approaches. Signature-based methods, often known as static analysis, refer to the process of examining a malicious file without its execution. Because of the growth of ransomware attacks and anti-forensic tactics such as packing and obfuscation, signature-based approaches have limitations. Behavior-based approaches, often known as dynamic analysis, refer to running a malicious program and observing its activities in the system. Behavior-based approaches can strive for the detailed characteristics of ransomware behavior. Their ability to strive for detailed characteristics makes using a defensive technique based on ransomware behavior much more effective. Thus, employing a behavior-based approach as a defensive strategy is more effective in preventing ransomware attacks from carrying out damaging actions.

III. LITERATURE REVIEW

Protecting data and systems from ransomware attacks requires a proactive solution. A proactive solution refers to the early recognition of the malware threat. CTI and CTH are novel techniques used to spot cyber threats in the environment.

A. CTI STUDIES

CTI is a proactive method that gathers valuable information from various sources to provide insight into the most recent cyber vulnerabilities and threats. Discovering and extracting such vital threat information is crucial for cybersecurity researchers and practitioners to improve awareness. Williams *et al.* [18] utilized a web crawling technique to find proactive cyber threat intelligence (CTI) in hacker forums. They implemented the Depth-First Search (DFS) technique, an incremental crawling method for collecting attachments while avoiding various popular anti-crawling measures.

Li *et al.* [19] relied on articles focusing on security event-related topics to build a proactive CTI. They collected 131 articles from the Internet and built an SVM model for data analytics. Samtani *et al.* [20] presented a methodology for implementing a more proactive CTI by mining hacker communities for source codes, tutorials, and attachments. The framework employs social network analysis methodologies and metrics to identify the key individuals behind discovered hacking assets. Ebrahimi *et al.* [21] focused on cyber threats hosted by the deep net market to avoid significant financial losses. They developed semi-supervised cyber threat identification, an integral part of the CTI, used to detect various types of threats and their primary data sources. They created a web crawler that used a combination of approaches to combat deep net marketplace anti-crawling mechanisms. Table 1 summarizes the proactive CTI strategies studied.

TABLE 1. A summary of CTI techniques.

Ref*	Data source	Analytics	Selected threats
[18]	10 hacker forums	LSTM	mobile, database, Web, system, network.
[19]	131 articles	SVM	Event-based security articles.
[20]	8 forums	SVM and LDA	DDoS, SQLi, Keyloggers, web exploits.
[21]	9 darknet markets	LSTM	Financial market threats.

* Ref refers to "Reference".

B. CTH STUDIES

Cyber threat hunting (CTH) is an approach that integrates CTI with data analysis methods to detect and respond to threats proactively. Homayoun *et al.* [23] developed sequential pattern mining as a ransomware hunting mechanism. They tried to hunt abnormal behavior within the first 10 seconds of ransomware execution by mining system logs of file system activities, registry, and Dynamic Link Libraries (DLL). Sequential pattern mining was implemented to discover Maximal Frequent Patterns (MFP) and combined with machine learning classification techniques to identify ransomware and benign samples and distinguish ransomware families.

Mavroeidis *et al.* [24] suggested a Sysmon log-based automated threat hunting system. Sysmon refers to a Windows system monitor service for monitoring and logging system activities. The proposed solution presents an automated threat assessment system that analyzes the continuous incoming feeds from Sysmon logs to classify the system processes to different threat levels. Detection was performed based on a predefined knowledge base.

Darabian *et al.* [25] developed an integrated multi-view learning approach that uses multiple features rather than a single feature view to detect malware behavior on diverse platforms. Weight is added to each view to enhance the hunting approach, including the header information, ByteCodes, API call, OpCodes, permission, and the attacker's intent. SVM model was used to assign weights to the obtained view. The proposed solution was employed on Windows, Android, and IoT platforms.

Naik *et al.* [26] developed triaging methods as hunting techniques to determine the similarity of two ransomware

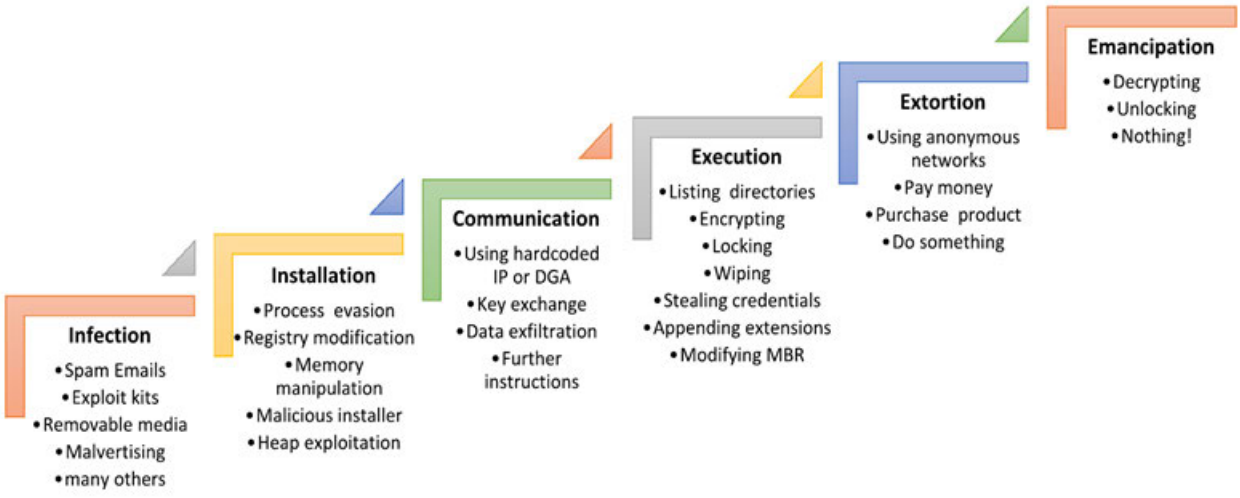

FIGURE 1. Ransomware attack steps [14].

samples. They applied four evaluation methods: import hashing method (IMPHASH), SSDEEP and SDHASH fuzzy hashing methods, and YARA rules. The performance results are described with the number of detected samples and a comparison between the four methods without showing performance results.

Jadidi *et al.* [27] proposed an industrial control system threat hunting framework (ICS-THF). The proposed framework focuses on detecting cyber threats against ICS devices. The proposed framework consists of three phases: threat hunting trigger, threat hunting, and cyber threat intelligence. The first phase includes events that could trigger the hunting phase. Then, the second phase uses a combination of the MITRE ATT&CK matrix and a diamond model of intrusion analysis to generate hunting hypotheses and predict future behavior. Finally, the third phase generates indicators of compromise (IoCs) for future threat hunting.

HaddadPajouh *et al.* [28] developed an IoT malware hunting method using a Long Short-Term Memory (LSTM) structure based on their OpCode sequences. Their findings demonstrated that stacked LSTM techniques could achieve high accuracy and handle input sequences of any length.

Jahromi *et al.* [29] developed an Extreme Learning Machine (ELM) approach that includes two hidden layers. They aimed to achieve an extremely fast learning speed, good generalization capability, straightforward implementation, and reduce the human intervention characteristics.

Homayoun *et al.* [30] developed a system for deep ransomware threat hunting in the fog layer. They used LSTM and CNN for classification to discover ransomware attacks within the first 10 seconds of program execution.

Al-rimy *et al.* [31] proposed two novel techniques, incremental bagging (iBagging) and enhanced semi-random subspace selection (ESRS), which are combined into an ensemble-based detection model. iBagging technique is used to build incremental subsets that show the evolution of crypto-ransomware behavior over various attack phases. ESRS technique is then used to construct feature spaces and exclude weak features.

Al-rimy *et al.* [32] proposed a novel Redundancy Coefficient Gradual Upweighting (RCGU) technique that improves redundancy–relevancy tradeoffs during feature selection. RCGU technique increases the redundancy term weight proportional to the number of selected features. The Enhanced MIFS (EMIFS) was developed by combining the RCGU technique with the Mutual Information Feature Selection (MIFS) technique. Moreover, MM-EMIFS was developed as an improvement that incorporates the MaxMin approximation with EMIFS to prevent redundancy overestimation. They mentioned that the limitation of the proposed work is the lack of consideration of the conditional redundancy term when calculating the feature importance.

Kok *et al.* [33] proposed a Pre-Encryption Detection Algorithm (PEDA) that aims to discover crypto-ransomware attacks at the phase of pre-encryption using two levels. The first level uses static analysis to compare the file signature with the known ransomware signature. The second level uses dynamic analysis with a learning algorithm model that analyzes the API generated in the pre-encryption stage.

Darem *et al.* [34] proposed an adaptive behavioral-based incremental batch learning malware variants detection model (AIBL-MVD) using concept drift detection and sequential deep learning.

Roy *et al.* [35] proposed a deep learning-based ransomware detector (DeepRans). The proposed model monitors the infected host's suspicious activity in the bare metal server network. DeepRans was developed using attention-based Bi-LSTM with Conditional Random Fields to classify the normal and infected host activities.

Pundir *et al.* [36] proposed a hardware-assisted ransomware detection technique using DL methods. They monitored micro-architectural events using a hardware performance counter to detect abnormal events. They showed that the proposed solution could detect ransomware in 2 milliseconds before encryption. However, hardware data processed in run-time could be corrupted.

Ullah *et al.* [37] proposed a ransomware detection model using online ML classifiers. The proposed model extracts the run-time features and performs ransomware detection. The model performs detection by tracing the ransomware behavior features during the execution, such as registry, network, and file systems API calls. Their proposed model used a modified decision tree, random forest, and AdaBoost classifiers.

Zhang *et al.* [38] proposed a deep learning-based model that uses a self-attention mechanism. The authors extracted contextual information to apply the static analysis. They used an N-gram of opcodes to identify ransomware fingerprints in the environment.

Khan *et al.* [39] proposed a DNAact-Ran system that uses digital DNA sequencing along with ML to detect ransomware. Naïve bayes, random forest, and Sequential minimal optimization classifiers were utilized in the proposed system. The proposed system was able to predict ransomware using DNA sequencing, which illustrates several numbers of features.

Poudyal *et al.* [40] proposed an AI-based ransomware detection framework (AIRaD). The proposed framework combines static and dynamic analysis to detect ransomware attacks. SVM, logistic regression, random forest, AdaBoost with J48, and J48 classifiers were used in the AIRaD framework. Multi-level analysis was performed on the assembly, DLL, and function calls.

Zuhair *et al.* [41] proposed a machine learning-based multi-layer ransomware detection system. The proposed solution consists of analysis, learning, and detection phases. The model utilized behavioral analysis to detect unknown ransomware variants. A decision tree and naïve bayes classifiers were used in the proposed solution. The first step is ransomware detection using a decision tree, and the second step is ransomware prediction using naïve bayes decisions. The limitation of the proposed system is the time of the fed samples analysis, which was done for 5 minutes.

Alrazib *et al.* [42] proposed DL-driven software-defined networking (SDN) intrusion detection system (IDS). They studied the presence of emerging cyber threats in the IoT environment. They utilized DNN and LSTM classifiers as a hybrid detection scheme. They used the CICIDS-2018 dataset, including 14 cyber threats such as brute-force, DDoS, and port scanning attacks.

Javeed *et al.* [43] proposed a novel SDN- enabled hybrid DL-driven cyber threat detection. They utilized LSTM and GRU classifiers as a hybrid detection scheme. The proposed solution detects cyber threats on the IoT platform. They used the CICDDOS-2019 dataset that includes network flow features for more than 78283 instances.

In summary, the limitations and gaps in current CTH solutions are as follows:

- *Disregarding the evolving nature of ransomware or malware attacks:* some previous studies have focused on detecting existing malware based on traditional analysis techniques that compare one or more static features that are insufficient to the evolving nature of attacks that utilize elusion and offense techniques. Some previous studies that detect malware based on static features, such us [25], [26], cannot detect unknown threats.
- *Relying on using classification methods based on static features:* some previous works relied on analyzing only static information that is ineffective against sophisticated malware. Static features involve analyzing a malware binary file without executing the code, such as determining the malware's signature and calculating the hash of the malware file. Known malware can be easily applied to other file formats in which previously collected data are useless. Darabian *et al.* [25] used only static features for their proposed solutions.
- *Small number of samples in the used dataset:* The growth in the complexity of ransomware or malware attacks requires the utilization of a large, diverse, and up-to-date dataset. Some studies used a small dataset that could affect the prediction and lead to overfitting issues. For a supervised deep learning algorithm, a rough rule of thumb indicates that using approximately 5000 labeled samples per category will mainly achieve acceptable performance [44]. Homayoun *et al.* [30] used a small dataset containing 660 ransomware samples and 219 benign samples. Moreover, HaddadPajouh *et al.* [28] applied the RNN model with a dataset that contained 281 malware samples and 270 benign samples to train the model, and then they used 100 malware samples to evaluate the model. Pundir *et al.* [36] applied their proposed RNN and LSTM solution using a dataset that contained 80 ransomware samples and 76 benign samples.
- *Imbalanced data:* Some studies include classification data with skewed class proportions that will make one class a majority class and another a minority class. Homayoun *et al.* [23] applied their proposed solution using a dataset that contains 1624 ransomware samples against 220 benign application samples. In addition, Al-rimy *et al.* [31] used a dataset that includes 8152 ransomware samples from 1000 benign application samples from another dataset. Darem *et al.* [34] used a dataset containing 19,076 malware samples and 3,994 benign application samples.
- *Hiding performance results:* Performance results, such as model accuracy, f-measure, and other measures, could help other researchers evaluate previous works and solve the current challenges. Some studies do not show performance results and findings that could affect the research field. Mavroeidis *et al.* [24] did not show any

performance results in their research to evaluate the proposed solution.

- *Some studies have not specified the source of the collected data samples:* Datasets are a significant part of scientific research. Some previous studies have not described the source of malware or ransomware samples. Homayoun *et al.* [30] did not mention the source of benign samples in their work.
- *Some studies have focused on finding a set of features that cannot be shown in other versions of ransomware samples:* Kok *et al.* [33] depended on finding Windows API calls that indicate pre-encryption processes. Other ransomware samples that used their native encryption codes will not be detected.

IV. CYBER THREAT INTELLIGENCE TECHNIQUES

Finding reliable intelligence regarding cyber threats helps defend against current attacks in a proactive manner [45]. Many CTI techniques have been proposed for obtaining timely information from trustworthy sources. CTI can provide detailed information related to anticipated cyber attacks. For example, an email designed for phishing attacks could include various vital features such as the attack technique used, attacker information, target information, software, and tools used to launch the attack [46].

The collection and analysis of massive amounts of online sources of threat data present a new area of challenges that enhance CTI abilities to mitigate or disable rising attacks [20]. Different capabilities are required to produce comprehensive CTI to find knowledge. To discover online sources, extensive data analysis, awareness of web crawling and anti-crawling mechanisms, understanding of foreign languages, knowledge of cyber world terms, and understanding of the complex structures of malicious assets are needed. Malicious assets can be found on different online platforms such as repositories, IRC channels, and hacker forums to exchange content and knowledge.

The web crawling mechanism is applied to search for web content as a computer program that systematically browses sources on the World Wide Web [47]. A web crawler is used for different purposes, such as searching for and extracting information or classifying web content. A crawler parses HTML tags and retrieves pages, extracts new hyperlinks from these tags, and stores HTML content. After collecting the data, the analysis technique is utilized to leverage the discovered information to understand the critical trends of malicious cyber assets.

V. MALWARE ANALYSIS

To detect malware, researchers used various techniques, including analyzing files with various tools, extracting static or dynamic features from the analyzed files, and categorizing features to distinguish between malicious and benign software. Malware analysis could be classified into static, dynamic, and hybrid [48]. Malware samples can be analyzed manually or automatically [49]. Automatic analysis requires advanced data science programming skills; however, domain expert knowledge is needed in manual analysis.

A. STATIC ANALYSIS

Static malware analysis is applied by reverse engineering, disassembling, or dissecting a malware binary file to analyze the different structural and semantic information found in the binary file. The structure of the malware sample is identified by static analysis without actually executing malicious code. File strings, header information, and functions are examined in fundamental static analysis. More details of the program commands are examined in the advanced static analysis.

B. DYNAMIC ANALYSIS

On the other hand, dynamic malware analysis is applied by observing or debugging a malware's program instructions to evaluate its behavior in an isolated environment. Isolated environments, such as virtual machines or sandboxes, are used to perform the dynamic analysis. API calls, memory and registry changes, parameters, information flows, and network activities are tested in dynamic analysis. There are two parts of dynamic malware analysis: basic and advanced. The fundamental dynamic analysis uses monitoring tools to examine malware's behavior. However, the advanced dynamic analysis uses debugging tools to execute each command individually to view command contents such as variables, parameters, and memory areas.

C. HYBRID ANALYSIS

In addition, hybrid malware analysis is a file analysis that combines both static and dynamic analysis aspects. It extracts the structural and semantic information of the binary file besides the run-time information.

Static analysis is easier and faster than dynamic analysis; however, it is impossible to analyze malicious software that utilizes obfuscation, packed, or polymorphic techniques using static analysis. For detecting unknown malware threats, dynamic analysis is more effective. Although dynamic analysis shows malware's actual functionality, some malware variants could be aware of being analyzed in isolated or closed environments, resulting in hiding their actual behavior.

VI. CYBER THREAT HUNTING TECHNIQUES

The concept of CTH describes combining an effective CTI method with a robust data analysis technique to detect cyber threats. Cyber attacks are evolving and becoming more sophisticated because of the advanced level of threat actors' skills [50]. Various solutions have been proposed as a data analysis technique to detect cyber threats. The application of machine learning-based techniques has a great majority of the current methods of CTH. The main development trends of the CTH are described in the following paragraphs.

A. TRADITIONAL MACHINE LEARNING APPROACHES

Machine learning (ML) is a part of artificial intelligence where machines learn from data or experience to automate the

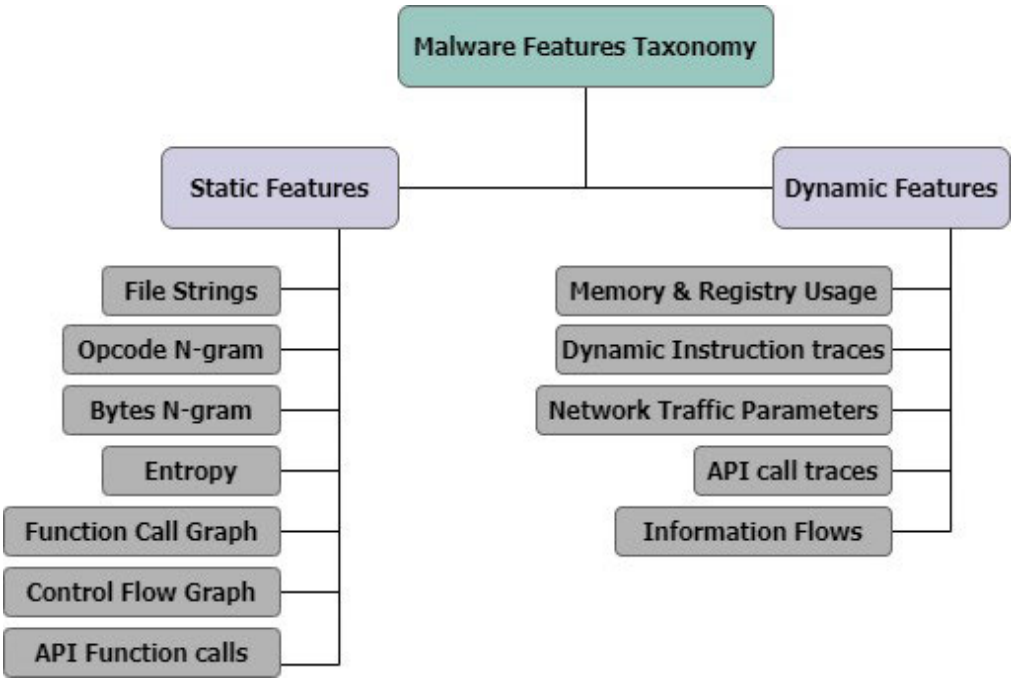

FIGURE 2. Malware Feature Taxonomy.

building of analytical models [51]. The efficiency of the ML model depends on the quality and performance of the chosen learning algorithm. Supervised learning, unsupervised, semi-supervised, and reinforcement learning are the four major categories of ML algorithms.

ML involves several steps such as data collection, data cleaning and preparation, model building, model evaluation, and model deployment. A set of software features is extracted during data preparation to describe it and classify it as benign or malware. The features were then used to train the model to solve the specified problem. Malware can be identified based on different features categorized according to the type of malware analysis approach: static and dynamic. Figure 2 shows the malware feature taxonomy.

Various ML classifiers have been employed in ransomware and malware detection models such as decision tree [23], [37], [41], ransom forest [23], [33], [37], [39], [40], naïve bayes [39], [40], Sequential minimal optimization (SMO) [39], logistic regression [40], J48 [23], [40], and SVM [25], [40]. Ensemble learning has also been employed in ransomware detection models [31], [37], [40]. Table 2 presents a summary of ML techniques used for CTH.

B. DEEP LEARNING APPROACHES

Deep learning (DL) is a subset of ML that learns from data, and the computation is performed through multilayer neural networks, and processing [52]. DL models require a large amount of data for each problem domain to construct a data-driven model. Moreover, DL algorithms require high computational capabilities to train models with a large amount of data. An essential characteristic of DL is that it decreases the time and effort required to construct the feature extractor. Supervised, unsupervised, and hybrid learning are the three major categories of DL algorithms.

Different DL architectures have been employed in ransomware and malware detection models such as MLP [23], [30], CNN [30], [38], Extreme Learning Machine (ELM) [29], sequential learning [34], RNN [36], and Long Short-term memory [28], [30], [35], [36]. Also, Hybrid DL models have been employed in IoT threat detection, such as [42] and [43]. Table 3 presents a summary of DL techniques used for CTH.

C. OTHER DATA ANALYSIS APPROACHES

Other data analysis approaches that do not include artificial intelligence methods have been utilized in CTH studies such as, [24], [26], and [27]. Table 4 presents a summary of other data analysis techniques for CTH.

VII. DIRECTION OF FUTURE RESEARCH ON RANSOMWARE

Most ransomware-related research works focus on different characteristics such as threat delivery, encryption algorithm and communication, associated IoCs, and behavior analysis [53]. Threat actors can change a malware's appearance to obfuscate its code; however, it is difficult to change its motivation and behavior. New ransomware variants are constantly being developed. Several detection and protection solutions rely on static analysis, which detects only earlier forms of ransomware samples. Cybercriminals apply advanced

TABLE 2. A summary of ML methods used for CTH techniques.

Ref*	Attack	Features	Method	Results
[23]	Ransomware	13 selected features	J48, Random Forest (RF), Bagging and MLP	Achieved F-Measure of more than 0.98 with FPR of less than 0.007.
[25]	Malware	OpCodes, ByteCodes, header information, permission, attacker's intent, and API calls	SVM	The accuracy of the proposed model is 99.6% on IoT dataset, 99.6% on Android dataset, and 98.01% on Windows dataset.
[31]	Ransomware	API calls	Ensemble-based learning	The detection accuracy ranges between 0.957838 to 0.97885 based on the number of the selected features.
[32]	Ransomware	API calls	SVM, Logistic Regression, Decision Tree, KNN, RF, AdaBoost, and MLP	The detection accuracy, detection rate and false positive of all feature sets per classifier ranges between 0.9573-0.9909, 0.9621-0.9940, and 0.0384-0.0068.
[33]	Ransomware	API calls	RF	The achieved recall rate was 100% based on 80:20 ratios of training and testing, and 99.9% recall rate with a 10-fold cross-verification test.
[37]	Ransomware	API calls of registry, network, and file system activities.	Modified DT, RF, and AdaBoost.	The achieved detection accuracy of Modified DT, RF, and AdaBoost are 99.56%, 99.24%, and 98.37%, respectively.
[39]	Ransomware	Generated DNA sequence of selected features.	Naïve Bayes (NB), RF, and sequential minimal optimization (SMO).	The achieved detection accuracy is 87.91%.
[40]	Ransomware	Multi-level of static and dynamic features.	SVM, LR, RF, AdaBoost with J48, and J48.	SVM and AdaBoost with J48 achieved the highest accuracy of 99.54%. J48 classifier achieved the second highest accuracy of 99.26%.
[41]	Ransomware	9 selected features.	DT and NB.	The model achieved an average accuracy of 96.27%.

* Ref refers to "Reference".

TABLE 3. A summary of DL methods used for CTH techniques.

Ref*	Attack	Features	Method	Results
[28]	Malware	OpCodes	LSTM	The proposed model achieved 98% detection accuracy against IoT malware samples not used in model training.
[29]	Malware	OpCode and system calls	TELM	TELM outperformed the original ELM models. Achieved accuracy rates are from 95.80% to 99.03%.
[30]	Ransomware	Sequence of actions	LSTM, CNN, and MLP	LSTM outperformed the other methods and achieved 0.996 F-measure of detecting ransomware in binary classification, and 0.972 TPR of identifying ransomware family in multi-class classification.
[34]	Malware	API calls	Sequential deep learning	The achieved F-measure results are higher than 99% and the average accuracy of detecting new malware is 99.41%.
[35]	Ransomware	Event ID of real-time bare metal logs	Attention-based bi-LSTM	Achieved 99.87% detection accuracy and 99.02% F-measure for early detection. Also, the model achieved 96.5% detection accuracy for classifying abnormal events.
[36]	Ransomware	Micro-architectural event traces	RNN and LSTM	Achieved an average of 97% detection accuracy.
[38]	Ransomware	N-gram of OpCode	Self attention-based CNN.	The achieved detection accuracy is 0.895 and the average F-measure is 0.873.
[42]	Cyber	Multiple features.	Hybrid DNN-LSTM	The achieved accuracy is 99.55% and the average F-measure is 99.42%.
[43]	Cyber	Network flow features	Hybrid LSTM-GRU	The achieved detection accuracy is 99.74% and the average F-measure is 99.79%.

* Ref refers to "Reference".

TABLE 4. A summary of other data analysis techniques used for CTH.

Ref*	Attack	Features	Method	Results
[24]	Malware	Features extracted from Sysmon logs	Dynamic analysis	The performance of the model has not been evaluated.
[26]	Ransomware	API names and order, Byte structure, textual, and binary patterns	IMPHASH, SSDEEP, SDHASH and YARA rules	SDHASH outperformed the other three methods based on the total number of samples matched.
[27]	Malware	events	MITRE ATT&CK matrix and a diamond model of intrusion analysis	The proposed framework was evaluated using three scenarios.

* Ref refers to "Reference".

TABLE 5. Some windows API calls categories and examples extracted from different ransomware samples.

#	Category	Windows API call	Purpose
1	registry	RegOpenKey	Open a registry key for editing and querying.
		RegCreateKey	Create the specified registry key.
		RegQueryValue	Retrieve the type and data for the specified value name.
		RegSetValue	Add a new value to the registry & sets its data.
		RegEnumValue	Enumerate the values for the specified open registry key.
2	fileSystem	CreateFile	Open document for reading and writing.
		ReadFile	Read data from document.
		WriteFile	Write data into document.
3	system	LoadLibrary	Load the specified module into the address space.
4	process	CreateRemoteThread	Create a thread of another process.
		CreateProcessInternal	Create a new process and its primary thread.
		ShellExecute	Perform an operation on a specified file.
		ExitProcess	End the calling process and all its threads.
5	memory	VirtualAlloc	Reserve, commit, or change the state of a region of memory within the virtual address space of a specified process.
6	synchronization	CreateMutex	Create or opens a named or unnamed mutex object.
		OpenMutex	Get a handle to another process's mutex.
7	services	OpenSCManager	Return a handle to the Service Control Manager.
		OpenService	Open an existing service.

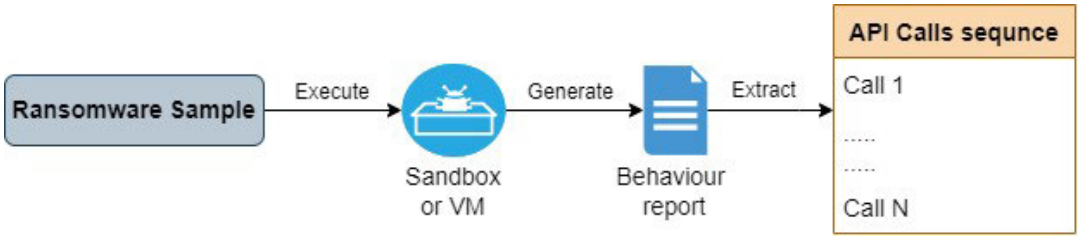

FIGURE 3. Extraction of API calls sequence from ransomware sample.

techniques to conceal the ransomware executable program intention to avoid detection.

Ransomware can appear as a standalone crypto-worm that replicates itself to other computers to maximize the impact on the network. In addition, ransomware can appear as a Ransomware-as-a-Service (RaaS), which is a distribution kit sold on the dark web. RaaS permits novel attackers with limited technical skills to launch ransomware attacks [55]. Moreover, ransomware can be deployed by threat actors who scan the Internet to find IT systems with soft protection to make them targets.

Ransomware can infect files on locally fixed, removable, or remotely shared drives. To minimize detection, attackers may attempt to sign their ransomware using code-signing technology by buying or stealing it. In addition, current ransomware utilizes exploits to abuse stolen administrator privileges and elevate their privileges. After that, the ransomware will start encrypting as many files as possible to ensure receiving ransom money from the victim. Files can be encrypted individually as a single thread or more than one at the same time as multiple threads. Moreover, ransomware can be programmed to start encrypting files with smaller file sizes or alphabetically [10].

A ransomware attack will have a tremendous financial impact on an organization when it encrypts its mapped network drives. Restoring multiple servers from backup data takes a long time, and data could not be up to date. Many organizations use only backup solutions as a critical defense against ransomware, which makes it a recovery solution rather than a detection solution. Ransomware attacks can also target backup files and folders, which can cause permanent damage and data loss [54].

The ransomware performs the file encryption process using two methods: overwrite (in-place) and copy. The overwrite method encrypts files by reading the original file, writing an encrypted version over the original file, and renaming the file. On the other hand, the copy method encrypts files by reading the original file, creating an encrypted copy, and deleting the original file. It is impossible to recover the original files using the overwrite method. However, ransomware that uses the copy method will use an additional wiping action to ensure that data files are not recoverable.

Ransomware behavior follows specific patterns that include the file identification process, file encryption, network command, and control communications [56]. In most ways, ransomware uses a Windows application programming interface (API) to make function calls. Windows API offers a collection of programming interfaces that simplify the software development process. Windows API calls can be used as behavioral features to identify abnormal patterns. Table 5 lists

TABLE 6. Datasets used by state-of-the-art ransomware detection works.

Ref*	Attack	Platform	Dataset classification	Dataset source	Number of samples
[23]	Ransomware	Windows	Ransomware	virustotal.com	1624
			Benign	portableapps.com	220
[26]	Ransomware	-	ransomware	hybrid-analysis.com malshare.com	200
[30]	Ransomware	Windows	Ransomware	virustotal.com	660
			Benign	portableapps.com	219
[31]	Ransomware	Windows	Ransomware	virustotal.com	8152
			Benign	informer.com	1000
[32]	Ransomware	Windows	Ransomware	virusshare.com	39378
			Benign	informer.com	16057
[33]	Ransomware	Windows	Ransomware	Sgandurra [58]	582
			Benign		942
[35]	Ransomware	Windows	Ransomware	PC host logs	4,820
			Benign		1,033,297
[36]	Ransomware	Windows	Ransomware	virusshare.com	80
			Benign	OpenSSC C programs	76
[37]	Ransomware	Windows	Ransomware	virustotal.com	35369
			Benign		43191
[38]	Ransomware	Windows	Ransomware	virustotal.com	1787
			Benign	System executable files	100
[39]	Ransomware	Windows	Ransomware	github	582
			Benign	System executable files	942
[40]	Ransomware	Windows	Ransomware	virustotal.com	550
			Benign	Windows 10 open-source software	540
[41]	Ransomware	Windows	Ransomware	virustotal.com malware blacklist	10000
			Benign	collected manually from websites	500

some Windows API call categories and examples extracted from different ransomware samples. Software API calls can be extracted from most modern devices [57]. Figure 3 shows the process of gathering API call sequences from ransomware samples.

VIII. RANSOMWARE DATASETS

Datasets are essential to foster the development of an effective ransomware detection solution. The outcome of a ransomware detection solution depends on the utilized dataset. Therefore, the accuracy of the solution is directly related to and dependent on the input dataset. Datasets contain several samples for benign and ransomware; however, one of the crucial challenges is a balanced dataset. Datasets for ML could either be privately collected or publicly available to anyone. Different ransomware studies used datasets from different repositories. Popular repositories that offer malware data include the following sources: VirusTotal, VirusShare, and theZoo. Table 6 shows a summary of openly available popular datasets and repositories used for ransomware detection studies.

IX. CONCLUSION

Ransomware is an evolving form of malware designed to block access to the system or encrypt its data. Various static and dynamic features of ransomware can be extracted and used to reveal its activities. This paper presents a systematic review of Cyber Threat hunting techniques for detecting ransomware attacks. The previous works of CTI and CTH have been investigated, and the limitations and gaps have been mentioned. Then, we explained the CTI technique. We provided an extensive overview of the malware analysis. CTH techniques are discussed based on the used data analysis method. Ransomware evolution and research directions are highlighted. The available ransomware datasets used in the previous works are mentioned with their data sources. In summary, ransomware attacks must be detected proactively, as shown in this study. Developing an effective ransomware CTH technique that can detect known and unknown ransomware is a concern. We provided a detailed review of ransomware research directions and the available ransomware datasets utilized with different data analysis methods. In our future work, we will adopt a CTI method to enhance the development of a CTH technique by collecting the latest shared information about ransomware attacks. Subsequently, the collected information will be incorporated into an effective new learning strategy model to enhance detection accuracy. A deep focus on dynamic features will be performed to hunt ransomware attacks based on behavior classification.

REFERENCES

[1] (2021). *Ransomware Attacks on Healthcare*. [Online]. Available: https://www.kaspersky.com/blog/ransomware-vs-healthcare/39635/

[2] S. Adam. (2022). *The State of Ransomware 2022*. Sophos News. [Online]. Available: https://news.sophos.com/en-us/2022/04/27/the-state-of-ransomware-2022/

[3] E. Berrueta, D. Morato, E. Magana, and M. Izal, "A survey on detection techniques for cryptographic ransomware," *IEEE Access*, vol. 7, pp. 144925–144944, 2019, doi: 10.1109/ACCESS.2019.2945839.

[4] R. Brewer, "Ransomware attacks: Detection, prevention and cure," *Netw. Secur.*, vol. 2016, no. 9, pp. 5–9, Sep. 2016, doi: 10.1016/S1353-4858(16)30086-1.

[5] P. O'Kane, S. Sezer, and D. Carlin, "Evolution of ransomware," *IET Netw.*, vol. 7, no. 5, pp. 321–327, 2018, doi: 10.1049/iet-net.2017.0207.

[6] O. Delgado-Mohatar, J. M. Sierra-Cámara, and E. Anguiano, "Blockchain-based semi-autonomous ransomware," *Future Gener. Comput. Syst.*, vol. 112, pp. 589–603, Nov. 2020, doi: 10.1016/j.future.2020.02.037.

[7] M. Akbanov, V. G. Vassilakis, and M. D. Logothetis, "Ransomware detection and mitigation using software-defined networking: The case of WannaCry," *Comput. Electr. Eng.*, vol. 76, pp. 111–121, Jun. 2019, doi: 10.1016/j.compeleceng.2019.03.012.

[8] M. Al-Hawawreh, F. D. Hartog, and E. Sitnikova, "Targeted ransomware: A new cyber threat to edge system of brownfield industrial Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 7137–7151, Aug. 2019, doi: 10.1109/JIOT.2019.2914390.

[9] E. Kolodenker, W. Koch, G. Stringhini, and M. Egele, "PayBreak: Defense against cryptographic ransomware," in *Proc. ACM Asia Conf. Comput. Commun. Secur.*, Abu Dhabi United Arab Emirates, Apr. 2017, pp. 599–611, doi: 10.1145/3052973.3053035.

[10] E. Berrueta, D. Morato, E. Magana, and M. Izal, "Open repository for the evaluation of ransomware detection tools," *IEEE Access*, vol. 8, pp. 65658–65669, 2020, doi: 10.1109/ACCESS.2020.2984187.

[11] S. Huda, J. Abawajy, B. Al-Rubaie, L. Pan, and M. M. Hassan, "Automatic extraction and integration of behavioural indicators of malware for protection of cyber–physical networks," *Future Gener. Comput. Syst.*, vol. 101, pp. 1247–1258, Dec. 2019, doi: 10.1016/j.future.2019.07.005.

[12] *Hunting for Hidden Threats–Cisco Cybersecurity 2019 Threat Report Series*. Cisco Umbrella. Accessed: Sep. 25, 2019. [Online]. Available: https://learn-umbrella.cisco.com/technical-papers/cisco-cybersecurity-series-2019-hunting-for-hidden-threats

[13] V. Mavroeidis and S. Bromander, "Cyber threat intelligence model: An evaluation of taxonomies, sharing standards, and ontologies within cyber threat intelligence," in *Proc. Eur. Intell. Secur. Informat. Conf. (EISIC)*, Athens, Greece, Sep. 2017, pp. 91–98, doi: 10.1109/EISIC.2017.20.

[14] M. Keshavarzi and H. R. Ghaffary, "I2CE3: A dedicated and separated attack chain for ransomware offenses as the most infamous cyber extortion," *Comput. Sci. Rev.*, vol. 36, May 2020, Art. no. 100233, doi: 10.1016/j.cosrev.2020.100233.

[15] A. Kharraz, W. Robertson, D. Balzarotti, L. Bilge, and E. Kirda, "Cutting the gordian knot: A look under the hood of ransomware attacks," in *Detection of Intrusions and Malware, and Vulnerability Assessment*, vol. 9148, M. Almgren, V. Gulisano, and F. Maggi, Eds. Cham, Switzerland: Springer, 2015, pp. 3–24.

[16] I. Nadir and T. Bakhshi, "Contemporary cybercrime: A taxonomy of ransomware threats & mitigation techniques," in *Proc. Int. Conf. Comput., Math. Eng. Technol. (iCoMET)*, Sukkur, Pakistan, Mar. 2018, pp. 1–7, doi: 10.1109/ICOMET.2018.8346329.

[17] I. Yaqoob, E. Ahmed, M. H. ur Rehman, and A. I. A. Ahmed, "The rise of ransomware and emerging security challenges in the Internet of Things," *Comput. Netw.*, vol. 129, pp. 444–458, Dec. 2017, doi: 10.1016/j.comnet.2017.09.003.

[18] R. Williams, S. Samtani, M. Patton, and H. Chen, "Incremental hacker forum exploit collection and classification for proactive cyber threat intelligence: An exploratory study," in *Proc. IEEE Int. Conf. Intell. Secur. Informat. (ISI)*, Miami, FL, USA, Nov. 2018, pp. 94–99, doi: 10.1109/ISI.2018.8587336.

[19] K. Li, H. Wen, H. Li, H. Zhu, and L. Sun, "Security OSIF: Toward automatic discovery and analysis of event based cyber threat intelligence," in *Proc. IEEE SmartWorld, Ubiquitous Intell. Comput., Adv. Trusted Comput., Scalable Comput. Commun., Cloud Big Data Comput., Internet People Smart City Innov. (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, Guangzhou, China, Oct. 2018, pp. 741–747, doi: 10.1109/SmartWorld.2018.00142.

[20] S. Samtani, R. Chinn, H. Chen, and J. F. Nunamaker, "Exploring emerging hacker assets and key hackers for proactive cyber threat intelligence," *J. Manage. Inf. Syst.*, vol. 34, no. 4, pp. 1023–1053, Oct. 2017, doi: 10.1080/07421222.2017.1394049.

[21] M. Ebrahimi, J. F. Nunamaker, and H. Chen, "Semi-supervised cyber threat identification in dark net markets: A transductive and deep learning approach," *J. Manage. Inf. Syst.*, vol. 37, no. 3, pp. 694–722, Jul. 2020, doi: 10.1080/07421222.2020.1790186.

[22] R. Vinayakumar, M. Alazab, K. Soman, P. Poornachandran, and S. Venkatraman, "Robust intelligent malware detection using deep learning," *IEEE Access*, vol. 7, pp. 46717–46738, 2019, doi: 10.1109/ACCESS.2019.2906934.

[23] S. Homayoun, A. Dehghantanha, M. Ahmadzadeh, S. Hashemi, and R. Khayami, "Know abnormal, find evil: Frequent pattern mining for ransomware threat hunting and intelligence," *IEEE Trans. Emerg. Topics Comput.*, vol. 8, no. 2, pp. 341–351, Apr. 2020, doi: 10.1109/TETC.2017.2756908.

[24] V. Mavroeidis and A. Jøsang, "Data-driven threat hunting using sysmon," in *Proc. 2nd Int. Conf. Cryptography, Secur. Privacy*, Guiyang, China, Mar. 2018, pp. 82–88, doi: 10.1145/3199478.3199490.

[25] H. Darabian, A. Dehghantanha, S. Hashemi, M. Taheri, A. Azmoodeh, S. Homayoun, K.-K.-R. Choo, and R. M. Parizi, "A multiview learning method for malware threat hunting: Windows, IoT and Android as case studies," *World Wide Web*, vol. 23, no. 2, pp. 1241–1260, Mar. 2020, doi: 10.1007/s11280-019-00755-0.

[26] N. Naik, P. Jenkins, N. Savage, and L. Yang, "Cyberthreat hunting–Part 1: Triaging ransomware using fuzzy hashing, import hashing and YARA rules," in *Proc. IEEE Int. Conf. Fuzzy Syst. (FUZZ-IEEE)*, New Orleans, LA, USA, Jun. 2019, pp. 1–6, doi: 10.1109/FUZZ-IEEE.2019.8858803.

[27] Z. Jadidi and Y. Lu, "A threat hunting framework for industrial control systems," *IEEE Access*, vol. 9, pp. 164118–164130, 2021, doi: 10.1109/ACCESS.2021.3133260.

[28] H. HaddadPajouh, A. Dehghantanha, R. Khayami, and K.-K. R. Choo, "A deep recurrent neural network based approach for Internet of Things malware threat hunting," *Future Gener. Comput. Syst.*, vol. 85, pp. 88–96, Aug. 2018, doi: 10.1016/j.future.2018.03.007.

[29] A. Namavar Jahromi, S. Hashemi, A. Dehghantanha, K.-K.-R. Choo, H. Karimipour, D. E. Newton, and R. M. Parizi, "An improved two-hidden-layer extreme learning machine for malware hunting," *Comput. Secur.*, vol. 89, Feb. 2020, Art. no. 101655, doi: 10.1016/j.cose.2019.101655.

[30] S. Homayoun, A. Dehghantanha, and M. Ahmadzadeh, "DRTHIS: Deep ransomware threat hunting and intelligence system at the fog layer," *Future Gener. Comput. Syst.*, vol. 90, pp. 94–104, Jan. 2019, doi: 10.1016/j.future.2018.07.045.

[31] B. A. S. Al-rimy, M. A. Maarof, and S. Z. M. Shaid, "Crypto-ransomware early detection model using novel incremental bagging with enhanced semi-random subspace selection," *Future Gener. Comput. Syst.*, vol. 101, pp. 476–491, Dec. 2019, doi: 10.1016/j.future.2019.06.005.

[32] B. A. S. Al-rimy, M. A. Maarof, M. Alazab, S. Z. M. Shaid, F. A. Ghaleb, A. Almalawi, A. M. Ali, and T. Al-Hadhrami, "Redundancy coefficient gradual up-weighting-based mutual information feature selection technique for crypto-ransomware early detection," *Future Gener. Comput. Syst.*, vol. 115, pp. 641–658, Feb. 2021, doi: 10.1016/j.future.2020.10.002.

[33] S. H. Kok, A. Abdullah, and N. Jhanjhi, "Early detection of crypto-ransomware using pre-encryption detection algorithm," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 5, pp. 1984–1999, May 2022, doi: 10.1016/j.jksuci.2020.06.012.

[34] A. A. Darem, F. A. Ghaleb, A. A. Al-Hashmi, J. H. Abawajy, S. M. Alanazi, and A. Y. Al-Rezami, "An adaptive behavioral-based incremental batch learning malware variants detection model using concept drift detection and sequential deep learning," *IEEE Access*, vol. 9, pp. 97180–97196, 2021, doi: 10.1109/ACCESS.2021.3093366.

[35] K. C. Roy and Q. Chen, "DeepRan: Attention-based BiLSTM and CRF for ransomware early detection and classification," *Inf. Syst. Frontiers*, vol. 23, no. 2, pp. 299–315, Apr. 2021, doi: 10.1007/s10796-020-10017-4.

[36] N. Pundir, M. Tehranipoor, and F. Rahman, "RanStop: A hardware-assisted runtime crypto-ransomware detection technique," 2020, *arXiv:2011.12248*.

[37] F. Ullah, Q. Javaid, A. Salam, M. Ahmad, N. Sarwar, D. Shah, and M. Abrar, "Modified decision tree technique for ransomware detection at runtime through API calls," *Sci. Program.*, vol. 2020, pp. 1–10, Aug. 2020, doi: 10.1155/2020/8845833.

[38] B. Zhang, W. Xiao, X. Xiao, A. K. Sangaiah, W. Zhang, and J. Zhang, "Ransomware classification using patch-based CNN and self-attention network on embedded N-grams of opcodes," *Future Gener. Comput. Syst.*, vol. 110, pp. 708–720, Sep. 2020, doi: 10.1016/j.future.2019.09.025.

[39] F. Khan, C. Ncube, L. K. Ramasamy, S. Kadry, and Y. Nam, "A digital DNA sequencing engine for ransomware detection using machine learning," *IEEE Access*, vol. 8, pp. 119710–119719, 2020, doi: 10.1109/ACCESS.2020.3003785.

[40] S. Poudyal and D. Dasgupta, "AI-powered ransomware detection framework," in *Proc. IEEE Symp. Ser. Comput. Intell. (SSCI)*, Canberra, ACT, Australia, Dec. 2020, pp. 1154–1161, doi: 10.1109/SSCI47803.2020.9308387.
[41] H. Zuhair and A. Selamat, *RANDS: A Machine Learning-Based Anti-Ransomware Tool for Windows Platforms*. Washington, DC, USA: IOS Press, Sep. 2019, doi: 10.3233/FAIA190081.
[42] M. A. Razib, D. Javeed, M. T. Khan, R. Alkanhel, and M. S. A. Muthanna, "Cyber threats detection in smart environments using SDN-enabled DNN-LSTM hybrid framework," *IEEE Access*, vol. 10, pp. 53015–53026, 2022, doi: 10.1109/ACCESS.2022.3172304.
[43] D. Javeed, T. Gao, and M. T. Khan, "SDN-enabled hybrid DL-driven framework for the detection of emerging cyber threats in IoT," *Electronics*, vol. 10, no. 8, p. 918, Apr. 2021, doi: 10.3390/electronics10080918.
[44] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
[45] M. F. Haque and R. Krishnan, "Toward automated cyber defense with secure sharing of structured cyber threat intelligence," *Inf. Syst. Frontiers*, vol. 23, no. 4, pp. 883–896, Aug. 2021, doi: 10.1007/s10796-020-10103-7.
[46] T. N. Jagatic, N. A. Johnson, M. Jakobsson, and F. Menczer, "Social phishing," *Commun. ACM*, vol. 50, no. 10, pp. 94–100, 2007, doi: 10.1145/1290958.1290968.
[47] E. Uzun, E. Serdar Güner, Y. Kiliçaslan, T. Yerlikaya, and H. V. Agun, "An effective and efficient web content extractor for optimizing the crawling process," *Softw., Pract. Exper.*, vol. 44, no. 10, pp. 1181–1199, Oct. 2014, doi: 10.1002/spe.2195.
[48] D. Gibert, C. Mateu, and J. Planes, "The rise of machine learning for detection and classification of malware: Research developments, trends and challenges," *J. Netw. Comput. Appl.*, vol. 153, Mar. 2020, Art. no. 102526, doi: 10.1016/j.jnca.2019.102526.
[49] A. D. Raju, I. Y. Abualhaol, R. S. Giagone, Y. Zhou, and S. Huang, "A survey on cross-architectural IoT malware threat hunting," *IEEE Access*, vol. 9, pp. 91686–91709, 2021, doi: 10.1109/ACCESS.2021.3091427.
[50] L. Caviglione, M. Choras, I. Corona, A. Janicki, W. Mazurczyk, M. Pawlicki, and K. Wasielewska, "Tight arms race: Overview of current malware threats and trends in their detection," *IEEE Access*, vol. 9, pp. 5371–5396, 2021, doi: 10.1109/ACCESS.2020.3048319.
[51] I. H. Sarker, "Machine learning: Algorithms, real-world applications and research directions," *Social Netw. Comput. Sci.*, vol. 2, no. 3, p. 160, May 2021, doi: 10.1007/s42979-021-00592-x.
[52] I. H. Sarker, "Deep learning: A comprehensive overview on techniques, taxonomy, applications and research directions," *Social Netw. Comput. Sci.*, vol. 2, no. 6, p. 420, Nov. 2021, doi: 10.1007/s42979-021-00815-1.
[53] (2021). *How Ransomware Attacks*. [Online]. Available: https://www.sophos.com/en-us/medialibrary/PDFs/technical-papers/sophoslabs-ransomware-behavior-report.pdf
[54] S. Sharmeen, Y. A. Ahmed, S. Huda, B. S. Kocer, and M. M. Hassan, "Avoiding future digital extortion through robust protection against ransomware threats using deep learning based adaptive approaches," *IEEE Access*, vol. 8, pp. 24522–24534, 2020, doi: 10.1109/ACCESS.2020.2970466.
[55] U. Urooj, B. A. S. Al-rimy, A. Zainal, F. A. Ghaleb, and M. A. Rassam, "Ransomware detection using the dynamic analysis and machine learning: A survey and research directions," *Appl. Sci.*, vol. 12, no. 1, p. 172, Dec. 2021, doi: 10.3390/app12010172.
[56] N. Hampton, Z. Baig, and S. Zeadally, "Ransomware behavioural analysis on Windows platforms," *J. Inf. Secur. Appl.*, vol. 40, pp. 44–51, Jun. 2018, doi: 10.1016/j.jisa.2018.02.008.
[57] Y. Ki, E. Kim, and H. K. Kim, "A novel approach to detect malware based on API call sequence analysis," *Int. J. Distrib. Sensor Netw.*, vol. 11, no. 6, Jun. 2015, Art. no. 659101, doi: 10.1155/2015/659101.
[58] D. Sgandurra, L. Muñoz-González, R. Mohsen, and E. C. Lupu, "Automated dynamic analysis of ransomware: Benefits, limitations and use for detection," 2016, *arXiv:1609.03020*.

FATIMAH ALDAUIJI received the B.S. degree in information technology from King Abdulaziz University, Saudi Arabia, in 2015, where she is currently pursuing the M.S. degree in information technology. Her main research interests include information security, data privacy, and networking.

OMAR BATARFI received the Ph.D. degree in network security from the University of Newcastle, in 2007. Since 1989, he has been working with King Abdulaziz University, where he is currently an Associate Professor with the Information Technology Department, Faculty of Computing and Information Technology. His main research interests include information security, data privacy, networking, and cloud computing.

MANAL BAYOUSEF received the B.S. degree in computer science from King Abdulaziz University, Saudi Arabia, in 2006, and the M.S. degree in information security and the Ph.D. degree in computer science from Florida State University, FL, USA, in 2013 and 2019, respectively. She is currently an Assistant Professor with the Department of Information Technology, Faculty of Computing and Information Technology, King Abdulaziz University. Her research interests include information security, network security, and quasi-Monte Carlo methods in high dimensional applications.