Journal of Ambient Intelligence and Humanized Computing
https://doi.org/10.1007/s12652-020-02630-7

ORIGINAL RESEARCH

Detecting ransomware attacks using intelligent algorithms: recent development and next direction from deep learning and big data perspectives

Ibrahim Bello[1] · Haruna Chiroma[2] · Usman A. Abdullahi[2] · Abdulsalam Ya'u Gital[1] · Fatsuma Jauro[3] · Abdullah Khan[4,5] · Julius O. Okesola[6] · Shafi'i M. Abdulhamid[7]

Received: 20 March 2020 / Accepted: 24 October 2020
© Springer-Verlag GmbH Germany, part of Springer Nature 2020

Abstract

Recently, cybercriminals have infiltrated different sectors of the human venture to launch ransomware attacks against information technology infrastructure. They demand ransom from individuals and industries, thereby inflicting significant loss of data. The use of intelligent algorithms for ransomware attack detection began to gain popularity in recent times and proved feasible. However, no comprehensive dedicated literature review on the applications of intelligent machine learning algorithms to detect ransomware attacks on information technology infrastructure. Unlike the previous reviews on ransomware attacks, this paper aims to conduct a comprehensive survey on the detection of ransomware attacks using intelligent machine learning algorithms. The study analysed literature from different perspectives focusing on intelligent algorithms detection of ransomware. The survey shows that there is a growing interest in recent times (2016—date) on the application of intelligent algorithms for ransomware detection. Deep learning algorithms are gaining tremendous attention because of their ability to handle large scale datasets, prominence in the research community, and ability to solve problems better than the conventional intelligent algorithms. To date, the potentials of big data analytics are yet to be fully exploited for the smart detection of ransomware attacks. Future research opportunities from the perspective of deep learning and big data analytics to solve the challenges identified from the survey are outlined to give the research community a new direction in dealing with ransomware attacks.

Keywords Big data analytics · Decision tree · Deep learning · Machine learning algorithms · Random forest · Ransomware

✉ Haruna Chiroma
chiromaharun@fcetgombe.edu.ng; freedonchi@yahoo.com

Ibrahim Bello
ibrahimubello@gmail.com

Usman A. Abdullahi
danzazzau12@gmail.com

Abdulsalam Ya'u Gital
asgital@gmail.com

Fatsuma Jauro
fjauro@abu.edu.ng

Abdullah Khan
abdullahdirvi@gmail.com

Julius O. Okesola
olatunjiokesola@tech-U.edu.ng

Shafi'i M. Abdulhamid
shafii.abdulhamid@futminna.edu.ng

[1] Department of Mathematical Sciences, Abubakar Tafawa Balewa University, Bauchi, Nigeria

[2] Future Technlogy Research Center, National Yunlin University of Science and Technology, Yunlin, Taiwan

[3] Department of Computer Science, Ahmadu Bello University, Zaria, Nigeria

[4] Faculty of Computing and Information Technology, Information System Department, King AbdulAziz University, Jeddah, Saudi Arabia

[5] Institute of Computer Sciences and Information Technology, University of Agriculture Peshawar, Peshawar, Pakistan

[6] Department of Computer Science, First Technical University, Ibadan, Nigeria

[7] Department of Cyber Security and Information Technology, Community College Qatar, Doha, Qatar

Published online: 11 November 2020

1 Introduction

Undoubtedly, the increased reliance on digital technology solutions has not only affected our lifestyle and businesses; it has also brought several security threats. Malware is one of these threats that has dramatically grown in prevalence, striking cyberspace incessantly (Hansen et al. 2016), inflicting damage to individuals and organizations around the globe. Ransomware is among the recent malware trend that blocks or restricts access to resources in the infected computer unless money is paid as ransom, mostly in the form of Bitcoin to reverse the attack. Recently, ransomware attacks have penetrated different spheres of human endeavour, including education, health, business, research, and information technology. Contrasting traditional malware, eradicating ransomware is problematic, and the damage imposed is irreversible even when removed (Al-rimy et al. 2018). Thus, cybersecurity has become a critical concern that attracts many researchers and industries in finding an effective defensive solution (Pluskal 2015).

Recently, ransomware has grown equally in complexity, adversity, and multiplicity to turn into the most destructive among the malware trends (Shaukat and Ribeiro 2018). Moreover, Cisco annual security report reveals that ransomware is growing at a yearly rate of over 300% (King 2017). Even though ransomware has been in manifestation for years, its variants have increased gradually and advanced in capability for proliferation, detection evasion, scrambling files, and compelling victims into paying ransoms. Over 200 active ransomware families are in existence, such as Tescrypt, Crowti, Cerber, Locky, etc. (Lu et al. 2017). The semantic security report reveals that ransomware variants increased by 46% in 2017 (Symantec 2019).

The earliest known ransomware, AidsInfo, was discovered in 1989. Its lack of an enabling environment and untraceable payment methods have rendered ransomware repellent to many cybercriminals (Savage et al. 2015). Of course, the earlier ransomware attack was elementary in reality and had some flaws. Still, it sets a platform for the evolution of ransomware into the advanced and sophisticated attacks carried out nowadays. However, the first flood of modern ransomware got on track in 2005 (Savage et al. 2015). After that, ransomware advanced rapidly, and various novel families of ransomware have appeared in recent years (Zhang et al. 2019a, b). Thus, ransomware has increased fourfold in current time, with 4000 attacks arising daily, reaching an estimated $1 billion in 2016 (Druva 2017).

Ransomware attacks have imposed an adverse impact against businesses driving on information technology infrastructure. The effect of these attacks encompass data or information damage due to file encryption, downtime caused by system shutdown of most companies, financial cost incurred by businesses security for incident arrest and other security-related challenges, perhaps intellectual property theft and loss of life as a result of the sudden shutdown of some imperative health equipment (Andronio et al. 2015; Gómez-Hernández et al. 2018).

Many approaches have been put forward by researchers in different literature to detect and defend against the negative effect of these ransomware attacks to find a lasting solution (Chen et al. 2017a, b; Cusack et al. 2018; Daku et al. 2018). However, ransomware is engaging in a variety of proliferation and evasion methods to circumvent defensive mechanisms (Damshenas et al. 2013). To defend users from being maltreated by ransomware attacks, new protection techniques are paramount to detect and prevent these malicious programs before inflicting destruction.

Machine learning intelligent algorithms have been proven to solve real-world problems in different domains of applications. As such, it has attracted the attention of academia and the industry. Recently, intelligent algorithms have started penetrating the realm of ransomware to provide solutions to ransomware attacks. Any algorithm that can learn from data is referred to as the machine learning algorithm. Those machine learning algorithms are intelligent because of their ability to adapt to new situations. The suitability of the machine learning algorithms in solving the problem makes it possible to be applied in the detection of ransomware attacks. Many researchers have used intelligent algorithms to solve the problem of ransomware attacks, and successes were recorded.

Despite the successes recorded by the machine learning algorithms in detecting ransomware, no comprehensive dedicated survey is conducted on the applications of machine learning intelligent algorithms in the detection of ransomware to the best of the authors' knowledge. However, many surveys on ransomware exist in the literature; details can be found in Sect. 2. In this paper, we conducted a comprehensive dedicated study on the applications of machine learning defensive solutions to ransomware attacks. The survey is in three perspectives: (1) technical view of the machine learning algorithms found to be applied to detect ransomware attacks. (2) The applications of the machine learning intelligent algorithms in providing solutions to ransomware attacks. (3) Synthesis and analysis of the literature.

2 Previously published surveys on ransomware

In this section, the paper presents a survey on ransomware attacks that were published in the literature. This section provides an overview of published surveys as well as the

difference between the already published surveys and the present study. Recently, many papers on the survey of ransomware were published in the literature. For example, Yaqoob et al. (2017) surveyed the issue of the internet of things (IoT) ransomware. It converses the escalation of ransomware attacks. It also highlights the vulnerability of the IoT as well as their essential defensive measures requirement. Similarly, (Shakir and Jaber 2017) conducted a concise survey of the strength and weaknesses of the ransomware. The survey focuses on WannaCry ransomware. (Maigida et al. 2019) conducted a survey on ransomware attacks, including detection mechanisms, mainly conventional approaches.

Conti et al. (2018) presented an intensive survey on the economic impact of the ransomware from the Bitcoin payment perspective. Also, it provides a general view of each explored ransomware genesis, development, and mode of ransomware attacks operation. Kok et al. (2019) present a survey on a detailed ransomware attack lifecycle and its features. Sabharwal and Sharma (2020) presents a survey on the mode of propagation of ransomware. Drifts in criminology convictions were examined. Connolly and Wall (2019) present survey on crypto-ransomware in the cyber dynamics. Aurangzeb et al. (2017) conducts a survey of ransomware involving Windows-based ransomware families by forming a yardstick for evaluating ransomware attacking methods and payment modes.

Joseph and Norman (2020) focuses on memory forensics based on WannaCry ransomware that affected computers. Richardson and North (2017) duel on the ethics and legality of paying ransom and recovery mechanism in the event of ransomware attacks. Al-rimy et al. (2018) presents a survey on ransomware focusing on technology and loop halls that give room to effective ransomware attacks. Table 1 shows the summary of the survey already conducted with its corresponding focus. It indicates that the focus of our survey is different from the existing surveys.

3 Overview of ransomware: background, motivation, and target platforms

Ransomware is a devastating cyber threat with global damage costing individuals and organisations enormous forfeiture of assets. Ransomware is defined as the malware that denied user access to their devices or denied access to files. The access to the device or file is allowed after the victim pays a ransom. Some common examples of ransomware are as follows: Locky, Cryptolocker, CTB Locker, Cryptowall, Teslacrypt, Winlocker, Torrentlocker, among others (Verma et al. 2018). Ransomware attacks target various platforms, including PCs, mobile devices, IoT devices, wearable devices, and cloud productivity, to demand ransomware from individuals and organisations (Al-rimy et al. 2018). Recently, ransomware attacks have drastically increased to encompass IoT devices, mobile platforms including Android, and other internet-enabled devices (Chaudhary et al. 2018; Chen et al. 2017a, b; Lachtar et al. 2019; Muna et al. 2019; Villalba et al. 2018). Thus, ransomware dominated cybercrime reports in 2018, with its threat targeting both individuals and businesses (Berrueta et al. 2019). However, not only are individuals susceptible to ransomware attacks, organisations, and business entities are not spired regardless of the proactive countermeasures being practiced.

The motive for ransomware attacks is virtually always monetary. Unlike other types of malware attacks, ransomware-based attacks usually notified the victim that an exploit has occurred and is given instructions for how to recover from the attack. However, untraceable cryptocurrencies, like Bitcoin, Monero, etc. are the most popular ransom payment modes

Table 1 Summary of the surveys with a corresponding focus

References	Coverage area	Main focus
Yaqoob et al. (2017)	Ransomware related to security challenges in IoT	Internet of Things
Shakir and Jaber (2017)	Effects of WannaCry	WannaCry ransomware
Maigida et al. (2019)	The negative impact of ransomware and solutions	Information technology infrastructure
Conti et al. (2018)	The economic significance of ransomware campaigns	Bitcoin transactions
Kok et al. (2019)	Ransomware threat and detection techniques	Ransomware general perspective and conventional mechanism
Sabharwal and Sharma (2020)	Ransomware attack prevention awareness	Indian issues red alert
Aurangzeb et al. (2017)	Ransomware attack techniques and payment modes	Windows-based ransomware families
Joseph and Norman (2020)	Importance of memory forensics and tools	Memory forensics
Richardson and North (2017)	Ransomware evolution, mitigation, and prevention	Ransomware general perspective
Al-rimy et al. (2018) and Connolly and Wall (2019)	Ransomware threats, measures, and countermeasures The ransomware landscape	Different techniques and perspectives Crypto-ransomware
Our propose survey	Application of intelligent algorithms for detecting ransomware	Machine learning algorithms

required by cybercriminals to hide their identity. Generally, a time limit is assigned for the payment. If the deadline exceeds, the ransom demand multiplies, or files are damaged or permanently locked. Cybercrime has changed the landscape from a world of maverick attackers to a criminal business that generates massive revenue through extortion (Lee et al. 2019; O'Kane et al. 2018; Su et al. 2018). Thus, the time, data loss, and possible intellectual property theft that may be caused by the victim made ransomware attacks irreversible (Digital Guardian 2019).

Although ransomware extorts users and businesses for monetary benefit, however, the malicious program must gain access to the resources before holding it for ransom. This access happens through infection or attack vectors. Email attachments, email links messages, compromised websites, and online pop-ups are the most common deception used to distribute ransomware (Kok et al. 2019). Also, drive-by freeware apps, exploit kits, brute-force authentication credentials, Trojan botnet attacks, or social engineering techniques (Bhardwaj et al. 2016). Therefore, ransomware compromises the availability, confidentiality, and integrity of a victim's system (Javaheri et al. 2018).

In 2005, the notable trend of modern ransomware had grown in full swing (Savage et al. 2015). Various enablers, comprising undetectable payment methods, availability of cryptographic techniques, financial benefit, free development kits, and easy to use ransomware-as-a-service (RaaS) cloud services are the core contributors to the high rate of ransomware attacks. These enablers promote the advent of new advanced families of ransomware (Shukla et al. 2016).

Moreover, ransomware exploits system flaws such as remote code vulnerability, windows server message block to invade the system (National Vulnerability Databasa 2017). Many search techniques such as depth-first, file size, and file location in the tree hierarchy are often leveraged to trace user-related files in the victim's system (Scaife et al. 2016). Some ransomware families trace recently, access files, and encrypt them consecutively. While others render the entire drive inaccessible one time by only encrypting the master file table. Ahmadian and Shahriari (2016). Ransomware usually scrambled specific types of files such as.xls,.doc,.pdf,.jpg,.zip, and other critical business-related file types, like CAD files, database files, and website files (Lu et al. 2017). Ransomware has improved in complexity to hinders reverse engineering techniques by engaging emulation detection, advanced obfuscation, delayed dynamic code loading techniques (Martín et al. 2018; Min et al. 2018).

4 Intelligent algorithms applied for detecting ransomware

The devised taxonomy in Fig. 1 depicts the application of intelligent algorithms for the detection of ransomware. The taxonomy categorises the intelligent algorithms into random forest (RF), decision tree (DT), deep learning, and other algorithms. The RF shows capability in the detection of ransomware in Windows OS, virtual environment, PC, and Android OS (Bae et al. 2019; Cohen and Nissim 2018; Cusack et al. 2018; Scalas et al. 2019). The DT show capability in the detection of ransomware in Windows OS, real-time environment, and network (Alhawi et al. 2018; Daku et al. 2018; Wan et al. 2018). Deep learning algorithms show the ability to detect ransomware in Windows OS, Android OS, network, industrial internet of things, Twitter, etc. Other algorithms include the V-detector Negative Selection Algorithm with Mutation Optimization and Gradient Tree Boosting algorithm for the detection of ransomware in a virtual environment (Lu et al. 2017; Shaukat and Ribeiro 2018). Tree-Shaped Deep Neural Network (TSDNN) along with a (QDBP) and Random Tree autonomously with Bayes Net algorithm show capability in the detection of ransomware in the network environment (Almashhadani et al. 2019; Chen et al. 2017a, b).

Furthermore, the Softmax algorithm shows effectiveness in the detection of ransomware in an application (Homayoun et al. 2019). Complex Tree shows competence in the detection of ransomware in a real-time environment (Verma et al. 2018). Also, iBagging algorithm offers capability in the detection of ransomware in PC (Al-rimy et al. 2019). Similarly, Self-Attention Convolution Neural Network (SA-CNN) shows influence in the detection of ransomware in the Windows OS environment (Zhang et al. 2019a, b). Lastly, Naïve Bayes shows effectiveness in the detection of ransomware in the healthcare system (Maimo et al. 2019).

4.1 Deep learning

Deep learning is a branch of machine learning algorithms whose learning techniques are categorized into three: unsupervised, supervised, and semi-supervised (Amanullah et al. 2020). Supervised learning algorithms uses fully labeled data for training of the model, while the unsupervised learning techniques learn by extracting beneficial information from given unlabelled data. The semi-supervised learning techniques use a combination of both labeled and unlabelled training dataset. Deep learning has been further classified into; discriminative generative, and hybrid models. The discriminative models consist

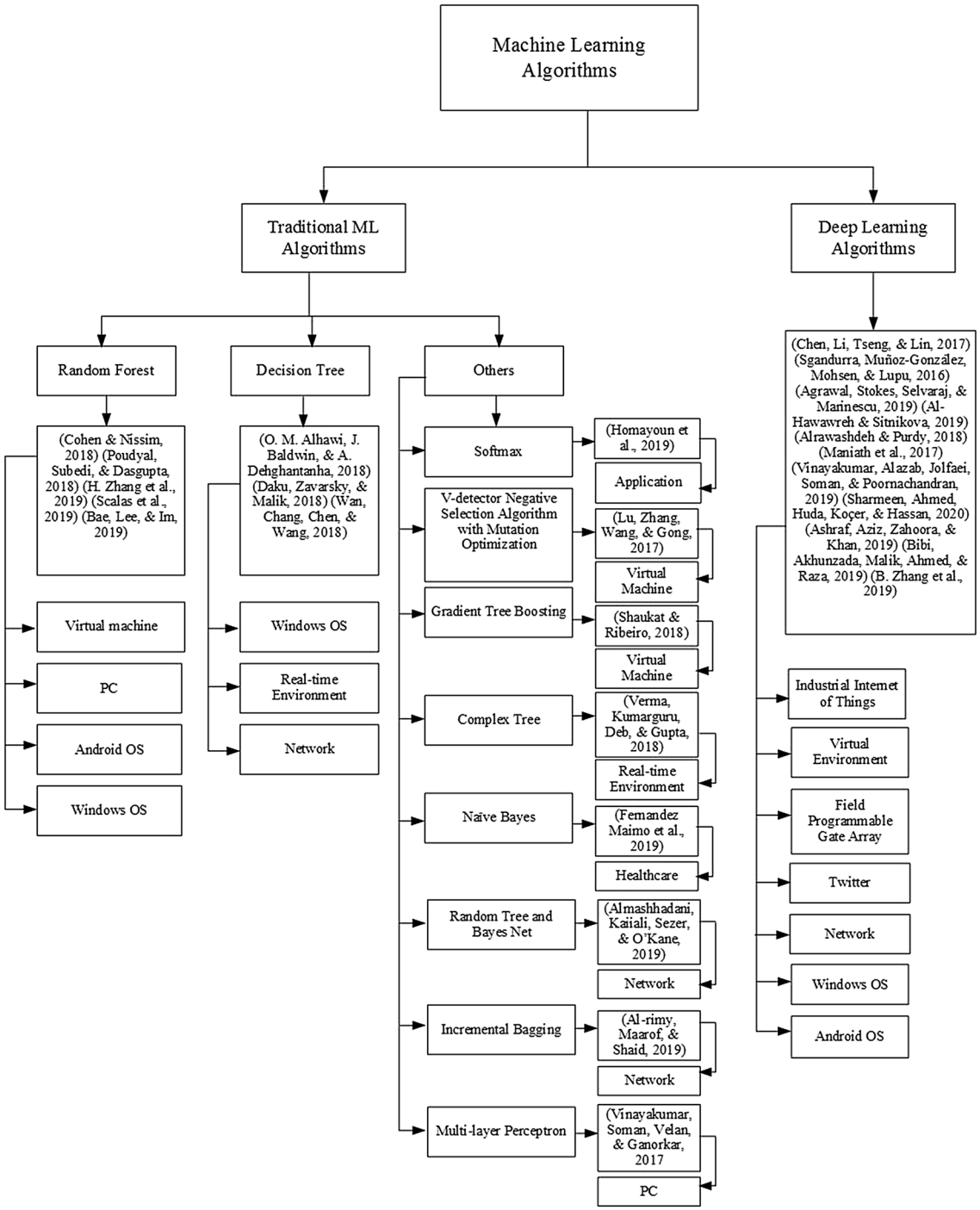

Fig. 1 Taxonomy of the applications of intelligent algorithms in detecting ransomware

of supervised learning methods, while the unsupervised learning methods are termed as the generative. The hybrid models benefit from a combination of the generative and discriminative models (Mohammadi et al. 2018).

Deep learning is a type of artificial neural network with multiple layers. Each descendant layer receives the output of its preceding layer as its input and further identifies more complex features (Mohammadi et al. 2018). It exploits multiple neurons and multiple layers to perform learning tasks such as auto encoding, clustering, classification, regression, and so on (Hatcher and Yu 2018). The process allows the models to uncover the hierarchical structure of data by learning both local and inter-relationships of the data (Haque and Neubert 2020). The models use an activation function to

determine the output of each neuron and a loss function for updating weights of the neurons (Hatcher and Yu 2018). A significant advantage of deep learning is that extracts meaningful information from raw data, and it barely requires manual feature engineering (LeCun et al. 2015).

4.2 Random forest

The random forest (RF) can be described as an intelligent ensemble algorithm, where a cluster of models combines to produce a robust model, and it is used in both regression and classification problems. However, RF generates results using multiple trees constructed via training processes. The RF combined tree predictors in such a way that each of the trees depends on the values of a random vector that is tested individually as well as with the same tree distribution in the forest (Breiman 2001).

4.3 Decision tree

The decision tree (DT) construction is a recursive problem. The structure of the DT starts by chosen the correct attribute that can be placed as the root node of the tree. Subsequently, a branch is made for each of the possible values. This process divides the set into a subset in such a way that each value of the attribute gets one subset. Afterward, the process is recursively repeated for each of the branches by using each of the instances that spread to the branch. The construction of the tree stops whenever the whole instances at the node of the tree possess the same classification (Frank et al. 2016). The DT is an intelligent algorithm commonly applied for solving machine learning problems such as classification and prediction. The DT has a tree structure; the structure of the tree represented major parameters as well as the condition for classification (Wan et al. 2018).

5 The detection of ransomware attacks via intelligent algorithms

This section presents the applications of intelligent algorithms in detecting ransomware attacks. It was found that many of the machine learning algorithms were applied to detect ransomware.

5.1 Detecting of ransomware using the random forest

RF is one of the intelligent algorithms applied to detect ransomware. For example, Cohen and Nissim (2018) proposes RF to detect unknown ransomware in virtual machines. A volatile memory dump taken from virtual machine trusted analysis was performed. Then, a general description of meta-features was extracted by applying the volatility framework. Consequently, leveraged these meta-features and RF is used to detect unknown ransomware in virtual machines. The results show that the RF out-performs Logit Boost (LB), AdaBoostM1 (AB), Logistic, Sequential Minimal Optimization (SMO), Naïve Bayes (NB), and Bagging classifiers. However, the system is exposed to malicious actions during the time between snapshots. Cusack et al. (2018) proposes RF detect and mitigate ransomware before encryption. The programmable forwarding engines (PFEs) utilizes switch hardware and dynamic memory cache to succeed high packet processing speeds while concurrently providing a rich flow of records.

Per-packet information is provided by these PEF-generated flow records, and allow the extraction of flow features for classification of ransomware at line rate in a decent fashion. The RF is applied to detect the ransomware. The results indicated that the RF outperforms the DT algorithm in detecting ransomware. However, the study did not consider user datagram protocol, and the RF can only detect a specific type of ransomware. Poudyal et al. (2018) proposes RF analyse and detects ransomware efficiently. The ransomware and regular binaries samples are pre-processed to extract features, and then different algorithms were applied for classification. The results indicated that the RF performs better than the NB, Bayesian Network (BN), Logistic Regression (LR), LB, SMO, Bagging, and AB algorithms. However, redesigned variants of new ransomware can decrease the rate of detection by the RF algorithm.

Zhang et al. (2019a, b) proposes RF classify ransomware families. The term frequency (TF) is computed for individual feature N-gram, which comprises the feature vector. The RF is subsequently applied to classify ransomware. The results indicated that the RF performs better than the DT, K-Nearest Neighbor (KNN), NB, and Gradient Boosting Decision Tree (GBDT). However, some ransomware families, exclusively locky, cryptowall, and reventon cannot be well-distinguished accordingly because the classification model is binary. Scalas et al. (2019) proposes RF to detect Android ransomware. The System API-based is static, and the Learning-based system extracts information (packages, classes, or methods) from the system API. The RF is applied to classify ransomware. The results indicated that the RF performs better than the Stochastic Gradient Descent (SGD) and SVM classifiers. The limitation is that replacing system-related entities with semantically equivalent or user-implemented ones can circumvent the system API-based techniques.

Bae et al. (2019) proposed RF to detect ransomware from malware and benign files. The class frequency, non-class frequency extracts windows API invocation sequences using the Intel PIN tool in a dynamic analysis environment. The sets of N-gram are generated from the extracted API sequences. The feature vectors are later generated from the

Table 2 Summary of the researches that use random forest for the detection of ransomware

References	Proposed algorithm	Evaluation algorithm	Contribution	Limitation
Cohen and Nissim (2018)	RF	Logistic, NB, BN, LB, SMO, Bagging, and AB	The results show that the RF performs better than the compared algorithms	The system is exposed to malicious actions during the time between snapshots
Cusack et al. (2018)	RF	DT	The results indicated that the RF out-performs the DT algorithm	The RF can only detect specific ransomware types. The study did not consider the UDP protocol
Poudyal et al. (2018)	RF	NB, BN, LR, LB, SMO, Bagging, and AB	The RF performs better than the compared algorithms	However, redesigned variants of new ransomware can decrease the rate of detection by RF algorithm
Zhang et al. (2019a, b)	RF	DT, KNN, NB, and GBDT	The results indicated that the RF performs better than the compared algorithms	Cryptowall, locky, and reveton cannot be well-distinguished because it is a binary classification
Scalas et al. (2019)	RF	SGD and SVM	The results indicated that the RF performs better than the SGM and the SVM	However, replacing System-related information with semantically equivalent, user-implemented ones may evade the system
Bae et al. (2019)	RF	LR, NB, SGD, KNN and SVM	The results indicated that the RF performs better than the compared algorithms	Only file-related APIs' are monitored, not the all system API invocations

sequences of N-gram. The RF is applied to detect ransomware. The results indicated that the RF performs better than the LR, NB, SGD, KNN, and SVM in detecting the ransomware. However, only file-related APIs' are monitored, not the whole system API invocations. Table 2 presents a summary of the studies that apply RF to detect ransomware attacks.

5.2 Detecting ransomware via decision tree

The DT is found to have been applied for the detection of ransomware. For example, Alhawi et al. (2018) proposed DT to detect windows ransomware. The NetConverse uses features extracted from the network traffic conversations when a host is infected. The DT is applied to detect the ransomware attacks. The results indicated that the DT algorithm performs better than the Bayesian network (BN), Multi-Layer Perceptron (MLP), K-Nearest Neighbour (KNN), RF, and Logistic Model Tree (LMT) algorithms in detecting Windows ransomware. However, the ransomware can behave differently upon detecting the control environment, thus deceiving the detection system. Daku et al. (2018) proposed DT to detect modified variants of ransomware attacks. The framework uses an iterative method to identify optimal behavioural attributes. Then, the DT is applied to determine the ransomware variants. The results show that the DT algorithm performs better than the NB and KNN algorithms in identifying the variants of ransomware. However, the method is unsuitable for real-time classification as it requires complete execution in the controlled environment.

Wan et al. (2018) proposed DT to detect a ransomware attack in a network. The framework applies a flow-based Biflow to substitute the packet-based data. Argus is used to convert these databases on open malicious traffic datasets into binary data representing flows of the network. The DT is used to detect the ransomware attacks. The results indicated that the DT algorithm performs better when combined with six feature selection algorithms. However, the study randomly classifies as abnormal (Cerber, Locky) and normal traffic. If the behaviors of the ransomware change, it could avert the classifier. Table 3 presents a summary of the applications of DT to detect ransomware.

5.3 Detecting of ransomware using the deep learning architecture

Al-Hawawreh and Sitnikova (2019) proposed a hybrid model based on classical auto-encoder (CAE), variational auto-encoder (VAE), and deep neural network with batch normalization (DNN-BN) to detect ransomware in the industrial internet of things (IIoT). The CEA and VEA are simultaneously used to reduce the dimension of data and extract features. The newly generated features are used to train DNN-BN. The DNN-BN is found to performs better than RF, DT, LR, SVM, and DNN. It does not address the problem of classifying multiple ransomware families. Agrawal et al. (2019) proposed an improved long short-term memory (LSTM) to detect ransomware in the

Table 3 The summary of the applications of the decision tree to detect ransomware

References	Proposed algorithm	Evaluation algorithm	Contribution	Limitation
Alhawi et al. (2018a, b)	DT	BN, KNN, MLP, RF, and LMT	The results indicated that DT performs better than BN, KNN, MLP, RF, and LMT algorithms in detecting Windows ransomware	The ransomware can behave differently upon detecting the control environment, thus deceiving the detection system
Daku et al. (2018)	DT	NB and KNN	The results show that DT performs better than NB and KNN algorithms in identifying variants of ransomware	The method is unsuitable for real-time classification as it requires complete execution in the controlled environment
Wan et al. (2018)	DT	Mixed with six feature selections algorithms for accurate precision	The results indicated that DT performs better than the compared algorithms	If the behaviors of the ransomware change, it could avert the classifier. Table 3 presents the summary of the applications of DT to detect ransomware

Windows environment. The attended recent input cell was incorporated with LSTM to integrate attention learning for ransomware sequences.

The ARI-LSTM performs better than the standard LSTM. Only a known target label and input event sequences are utilized for training the model in an end-to-end fashion. Alrawashdeh and Purdy (2018) proposed a four-layer deep belief network (DBN) model based on Restricted Boltzmann Machine (RBM) using memory-assisted-stochastic-dynamic-fixed-point arithmetic to detect ransomware in field programmable gate array (FPGA). The technique stores random bit-stream in memory to yield cross-correlation for the stochastic computation in FPGA. The memory technique trains the DBN for stochastic computation with dynamic fixed-point arithmetic. The memory-based cross-correlation reduction outperforms hybrid stochastic dynamic fixed-point (HSDFP) and the dynamic fixed-point methods. It can be difficult for the model to detect zero-day ransomware in FPGA. Maniath et al. (2017) proposed a model based on LSTM to detect ransomware behaviour for binary sequence classification of API calls. The method uses dynamic malware analysis of the ransomware to extract the API calls in the sequence. The LSTM uses the API sequences generated to classify the samples. The proposed model performs better than the RNN, DBN, auto-encoder (AE), RNN, and echo state networks (ESN). The malware may misbehave to hide its features in the execution environment, therefore bypassed the detection algorithm. Vinayakumar et al. (2019) proposed a model based on DNN to classify ransomware tweets to their respective families.

The method analyzes tweets from twitter posts to extract optimal features. The extracted features are then passed to the algorithm. The results show that the proposed model outperforms SVM and NB. The study is limited to twitter. Sharmeen et al. (2020) proposed a deep learning model for detecting ransomware threats. The method mines the intrinsic features from the different unlabelled ransomware samples. Then the unsupervised learned model is pooled with supervised classification to build an adaptive detection model. The actual ransomware data is leveraged to validate the framework with a dynamic analysis testbed. The results show that the proposed model outperforms SVM, RF, and multi-class classifiers. The ransomware may misbehave to hide its actual intent in the virtual environment. Bibi et al. (2019) proposed LSTM to detect Android ransomware through multi-factor feature infiltration. The method leverage eight different machine learning filtration technique to extract essential features. The deep learning-based model is used to detect the malicious behaviour of Android applications. The proposed model achieved 97.08% detection accuracy. There is no comparison made with other algorithms, so it is difficult to ascertain the advantage of the proposed algorithm over other algorithms.

Ashraf et al. (2019) proposed ransomware static and dynamic analysis (RanSD) for ransomware detection analysis using DNN. The method extracts feature from the collected samples. Then the extracted features are analysed to extract relevant features and sequences for classification. Finally, the effectiveness of the selected features is validated on the conventional learning model and deep learning based on transfer learning. The proposed model with a dynamic dataset performs better than the model with a static dataset. The proposed model only analyses the detection of ransomware using static features and dynamic features. Zhang et al. (2019a, b) proposed a self-attention convolution neural network (SA-CNN) to detect ransomware. The feature vectors were generated, and the self-attention captures valuable information from opcodes.

The sequence of N-gram is partitioned. CNN is combined with a bi-directional self-attention network. The SA-CNN is applied to detect the ransomware, and the result indicated that the proposed model outperformed the KNN, NB, and DT algorithms. The static analysis may not handle advanced packing techniques. Chen et al. (2017a, b) proposed a Tree-shaped DNN (TSDNN) with quantity dependent backpropagation (QDBP) algorithm to detect malicious flow, including ransomware in a network. The TSDNN model uses a behaviour-oriented approach to classify the data in a layer-wise manner. Subsequently, the QDBP incorporating the knowledge of the disparity among classes. The results show that the proposed algorithm outperformed the signature-based method in detecting the ransomware. The network behaviour of the malicious samples might not be well-captured within the threshold of 6 min.

5.4 The other class of the intelligent algorithms applied to detect ransomware

Apart from the main intelligent algorithms, namely, RF and DT that were found to be heavily used for the detection of ransomware attacks, there are other intelligent algorithms used for the detection of the ransomware discussed in this section. For instance, Homayoun et al. (2019) proposed the Softmax algorithm to detect ransomware and classify their families at the fog layer. The deep ransomware threat hunting and intelligence system (DRTHIS). The DRTHIS considers the set of activities performed by samples of the malware. The softmax algorithm is applied to detect ransomware and predicts its family. The results indicated that the Softmax algorithm performs better than traditional neural networks. However, some ransomware families can launch an attack at a time different from the threshold. Lu et al. (2017) proposed a V-detector Negative Selection Algorithm with mutation Optimization (op-RDVD) to improve ransomware detection. Behavioural features of ransomware were extracted. The op-RDVD is applied to detect ransomware. The results show that the op-RDVD performs better than the V-detector and real-valued negative selection (RNS). However, op-RDVD is unsuitable for real-time protection because it required complete execution within a controlled environment.

Shaukat and Ribeiro (2018) proposed gradient tree boosting (GTB) for the detection of cryptographic ransomware. The GTB is subsequently applied to detect ransomware. The results indicated that the GTB performs better than the LR, SVM, ANN, and RF algorithms. However, some families of ransomware can cripple systems in a time shorter than the threshold. Some ransomware families can cripple systems in a time shorter than the threshold. As such, escape detection. Verma et al. (2018) proposed complex tree (CT) to detect ransomware with new behaviour in real-time. The indicator of compromises (IOCs) observed traces of calls accomplished by all processes produced by the malware. The CT is applied to detect ransomware. It was found that the CT performs better than the linear determinant analysis (LDA), SVM, quadratic discriminant analysis (QDA), and KNN algorithms. However, the identified behaviors may not be sufficient to generally detect ransomware with different behaviour.

Almashhadani et al. (2019) proposed random tree for packet-level and BN for flow level to detect crypto-ransomware in a network autonomously. A thorough ransomware behavioural analysis was conducted using Wireshark. A network-based intrusion detection system is built using two autonomous classifiers executing in parallel. The results indicated that classifier C_1 performs better with the random tree algorithm, while classifier C_2 performs better with the BN algorithm. The system exclusively detects crypto-ransomware but does not considers locker ransomware. Al-rimy et al. (2019) proposes an ensemble-based model to detect ransomware. The ensemble model incorporated incremental bagging (iBagging) with an enhanced semi-random subspace selection (ESRS). The proposed iBagging is applied to detect ransomware. The results show that the proposed model outperformed AdaBoost, RF KNN, SVM, MLP, LR, and XGBoost algorithms. Evaluating each feature independently in each subspace can results in selecting redundant feature to other subspaces, which can decrease the accuracy of the classifier.

Maimo et al. (2019) proposed NB to detect and mitigate ransomware in an integrated clinical environment (ICE). The medical cyber-physical systems (MCPS) employs one-class support vector machine (OC-SVM) to detect anomaly in real-time through analysing the network flows generated during the ransomware spreading stage. Moreover, NB is applied to detect ransomware attacks. The results show that NB outperformed ANN and RF for ransomware classification. Not all the ransomware generated traffic patterns are distinguishable from the normal traffic patterns generated by the medical devices. Sgandurra et al. (2016) presents

EldeRan for detecting ransomware dynamically. EldeRan observes a series of activities executed by applications in their first stages of installation, inspecting for features signs of ransomware. The EldeRan outperformed SVM and NB. EldeRan does not correctly extract features of ransomware that are silent for some time or wait for user actions. Vinayakumar et al. (2017) proposed a model based on Multi-Layer Perceptron (MLP) to evaluate the effectiveness of shallow and deep networks for detection and classification of ransomware. The method passes the EXE files to the simulated environment and stores the detailed characteristics of ransomware samples in the sandbox logs. API calls are selected as features and passed as input to the MLP and some shallow models for learning to detect and classify ransomware. The results show that MLP outperformed NR, NB, DT, RF, KNN, and SVM. The ransomware may not reveal their actual intent in the simulated environment. Table 4 presents a summary of other intelligent algorithms in detecting ransomware.

6 Ransomware attacks dataset

To put it straight, "No data, No machine learning". This section presents the sources and type of data used in a different project to build the machine learning model for detecting ransomware attacks. Table 5 shows a summary of the sources and types of data used in various projects surveyed. The sources of the data can help researchers to obtain ransomware attacks data that are required for novel machine learning approaches.

We extracted the information about the various data from the project that revealed their source and types of data used for their work. However, the project that concealed the source of their data is not in Table 5 since the required information to fill the corresponding row is not available.

7 Analysis of the ransomware detection via intelligent algorithms

7.1 The intelligent algorithms that detect ransomware attacks

The use of intelligent algorithms for the detection of ransomware has been surveyed, as discussed in the preceding sections**.** Different types of intelligent algorithms were applied for the detection, and it has shown remarkable performance. The intelligent algorithm's performance in detecting ransomware has proven to be better than the conventional methods of detecting ransomware. This signifies that the intelligent algorithms have the potential for enhancing the accuracy of ransomware detection system when deployed in the real-world environment. Figure 1 shows the percentage of intelligent algorithms applied to detect ransomware attacks. The summary of the ransomware data sources is provided in Table 6.

The pie chart shows that 36% of the literature applied deep learning to detect ransomware activities or classify their families in windows OS, virtual environment, twitter, industrial internet of things, in FPGA, and android environment. Also, 21% of the literature applied RF to detect ransomware activities or classify their families in a virtual machine, PCs, Android OS, or window OS environment. On the other hand, 11% of the literature applied DT to detect ransomware attacks, including their variant in a network, window OS, or real-time environment. Finally, 32% applied other types of algorithms to detect ransomware attacks or classify their families in a virtual machine, network, application, PC, Windows OS, or real-time environment. It can be deduced from the descriptive statistics that the prominent algorithms that researchers heavily relied on to detect ransomware attacks are the deep learning followed by RF. As it can clearly be seen, deep learning algorithms is the state-of-the-art architecture used to detect ransomware attacks. Though, the idea of applying machine learning algorithms to detect ransomware attacks is still in an infant stage considering the time that the literature on the application of machine learning algorithms starts appearing (see Fig. 2).

7.2 The publication trend

Machine learning approaches have been prominent over the last decades for malware detection and analysis (Feizollah et al. 2015). However, machine learning has started finding its way into the detection of ransomware, as shown in Fig. 2. Figure 3 shows the literature based on publication year regarding ransomware detection using machine learning algorithms. In 2016, one literature had been published, then increased to four works of literature in 2017, later increased to nine and thirteen in 2018 and 2019, respectively. Finally, one literature has just been published in 2020 and expect to have many publications, especially in deep learning, because of the trend depicted in Figs. 2 and 3.

Figure 3 has shown that the researches in this domain are emerging with the possibility of witnessing a rapid increase in research outputs, especially towards the direction of deep learning because it is the core algorithms in machine learning in present times.

7.3 Domain of applying intelligent algorithms for the detection of ransomware attacks

Many works of literature have proposed different intelligent algorithms for the detection of ransomware in various domains. This domain includes network (Chen et al. 2017a,

Table 4 The summary of the applications of deep learning architecture in detecting ransomware

References	Proposed algorithm	Evaluation algorithm	Contribution	Limitation
Al-Hawawreh and Sitnikova (2019)	DNN-BN	RF, DT, LR, SVM and DNN	The DNN-BN performs better than RF, DT, LR, SVM, and DNN	It does not address the problem of classifying multiple ransomware families
Agrawal et al. (2019)	ARI-LSTM	LSTM	The ARI-LSTM performs better than the standard LSTM	Only a known target label and input event sequences are utilized for training the model in an end-to-end fashion
Alrawashdeh and Purdy (2018)	Memory-based DBN	HSDFP, DFP, and memory-based cross-correlation reduction	The memory-based DBL compared algorithms	It can find it challenging to detect zero-day ransomware in FPGA
Maniath et al. (2017)	LSTM	RNN, AE, DBN and RNN & ESN	The results show that the LSTM performs better than KNN, SVM, DBN, and RNN	The malware may misbehave to hide its features in the execution environment to escape detection
Vinayakumar et al. (2019)	DNN	SVM	The results show that the DNN performs better than the SVM	The study is limited to twitter
Sharmeen et al. (2020)	Deep learning-based model	SVM, RF, multi-class classifier	The results show that the proposed model outperforms SVM, RF, and multiclass classifiers	The ransomware may misbehave to hide its actual intent in the virtual environment
Bibi et al. (2019)	LSTM	NIL	It achieved 97.08%	The model only predicts the malware, but there is no comparison made with other algorithms
Ashraf et al. (2019)	DNN	SVM, RF, and ResNet-18	The proposed model outperforms with a dynamic dataset compared to a static dataset	The proposed model only analyses the detection of ransomware using static features and dynamic features
Chen et al. (2017a, b)	TSDN-QDBP	Signature-based method	The results show that the proposed method outperforms the signature-based method	The network behaviour of the malicious samples might not be well-captured within the threshold of six minutes
Zhang et al. (2019a, b)	SA-CNN	KNN, NB, and DT	The results indicated that DT performs better than the compared algorithms	Advanced packing techniques may not be handled by static analysis

Table 5 Summary of the used of variants of intelligent algorithms in detecting ransomware

References	Proposed algorithm	Evaluation algorithm	Contribution	Limitation
Homayoun et al. (2019)	Softmax	ANN	The results indicated that the Softmax algorithm performs better than the traditional ANN	Some ransomware families can launch an attack in a time different from the threshold
Lu et al. (2017)	op-RDVD	V-detector and RNS.s	The results show that op-RDVD performs better than V-detector and RNS	The op-RDVD is unsuitable for real-time protection
Shaukat and Ribeiro (2018)	GTB	LR, SVM, ANN, and RF	The results indicated that the GTB performs better than the compared algorithms	Some ransomware families can cripple systems in a time shorter than the threshold. As such, escape detection
Verma et al. (2018)	CT	SVM, LDA, QDA, and KNN	The results show that CT performs better than the compared algorithms	They identified behaviors may not be sufficient to detect ransomware generally
Fernandez Maimo et al. (2019)	NB	NN and RF	The results show that NB performs better than NN and RF	Not all the ransomware traffic patterns can be differentiated from the normal traffic patterns
Almashhadani et al. (2019)	Random Tree	LibSVM, and Random Tree	The results indicated that Classifier C_1 performs better with the Random Tree algorithm, while classifier C_2 performs better with Bayes Net algorithm	The system exclusively detects crypto-ransomware but does not considers locker ransomware
Al-rimy et al. (2019)	iBagging	AdaBoost, RF, SVM, KNN, multilayer perceptron (MLP), LR, XGBoost	The results show that the proposed model outperformed AdaBoost, RF, SVM, KNN, MLP, LR, XGBoost algorithms	Evaluating each feature independently in each subspace can result in selecting a redundant feature to other subspaces
Vinayakumar et al. (2017)	MLP	NR, NB, DT, RF, KNN, and SVM	The results show that MLP outperformed NR, NB, DT, RF, KNN, and SVM	The ransomware may not reveal their actual intent in the simulated environment

Table 6 Summary of the sources and type of the ransomware data used for modelling

References	Source	Type of data
Alhawi et al. (2018a, b)	ransomwaretracker.abuse.ch and virustotal.com	Network traffic captures
Cohen and Nissim (2018)	Virtual server snapshots	Meta- features created from volatile memory dumps
Cusack et al. (2018)	malwaretraffic-analysis.net	Network traffic signature
Daku et al. (2018)	virustotal.com	Behavioral attributes
Homayoun et al. (2019)	Ransomwaretracker.abuse.ch	system calls, the sequence of actions taken by an application
Lu et al. (2017)	virusshare.com	Application programming Interface function calls
Poudyal et al. (2018)	virusShare.com, virustotal.com, and https://github.com/ytisf/theZoo	Assembly instruction set and dlls extracted from binaries
Shaukat and Ribeiro (2018)	virusShare.com	Binary code
Verma et al. (2018)	malwr.com, virusShare.com, virustotal.com	Indicator of compromises
Zhang et al. (2019a, b)		N-grams extracted from the opcode
Wan et al. (2018)	malwaretrafficanalysis.net and wireshark.org	Network traffic captures
Chen et al. (2017a, b)	virustotal.com	Network traffic captures
Scalas et al. (2019)	https://www.virustotal.com https://github.com/necst/heldroid https://www.sec.cs.tubs.de/~danarp/drebin/ and Google Play store	System application programming interface based information
Fernandez Maimo et al. (2019)	https://perception.inf.um.es/ICE-datasets/	Network traffic captures
Almashhadani et al. (2019), (A-rimy et al. (2019; Zhang et al. (2019a, b) and Bae et al. (2019)	virusshare.com, malware-traffic-analysis.net, and virustotal.com virusshare.com, informer.com and virustotal.com virustotal.com, Windows(R) 10 professional edition Windows 7 system directories, virustotal.com	Behavioral and non-behavioral features Application Programming Interface calls opcode sequence System API invocations sequence
Al-Hawawreh and Sitnikova 2019)	virusShare.com, virustotal.com, and software.informer.com	API invocations, registry keys, file operations, file extensions, dropped file extensions, strings, and directory operations
Sgandurra et al. (2016)	virusShare.com, virustotal.com, and software.informer.com	API invocations, registry keys, file operations, file extensions, dropped file extensions, strings, and directory operations
Agrawal et al. (2019)	Microsoft Windows operating system	File events: createfile, virtualalloc, virtualalloc, getmodulehandle, and getmodulefilename
Alrawashdeh and Purdy (2018)	virusShare.com, virustotal.com, software.informer.com	File Extension, Extension Pattern, Encryption Algorithm, Registry Keys Operations, API Stats, Files Operations, Directory Operations, Dropped Files Extensions, Source File, Duration and HTTP Methods
Maniath et al. (2017)	Honeynets, Microsoft Windows, online software repositories	API calls, registry value changes, and file operations
Vinayakumar et al. (2019)	Tweeter posts	Tweets
Vinayakumar et al. (2017)	https://www.offensvecomputing.net/, https://contagiodump.blogspot.in/, https://malwr.com/, https://github.com.com/ytisf/theZoo/, https://virustotal.com/, and https://virusshare.com/	API invocations
Sharmeen et al. (2020)	virusShare, VirusTotal, and Software-informer	API calls
Bibi et al. (2019)	Smartphone executable App	API calls
Ashraf et al. (2019)	virusShare, VirusTotal, Windows 7 OS	API calls, Registry operations, File operation, Directory created, Network domains, Drop file extensions, DLL's, and Strings

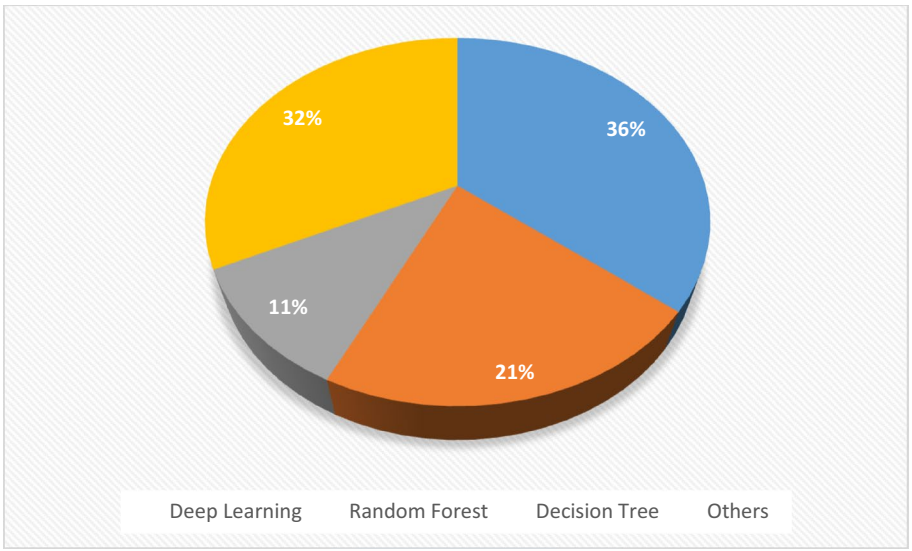

Fig. 2 The percentage of intelligent algorithms for the detection of ransomware attacks

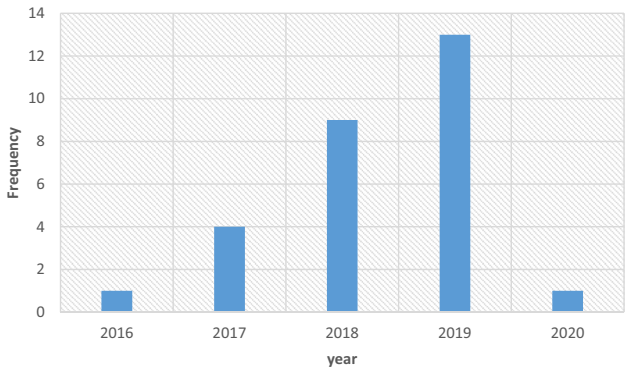

Fig. 3 Publications trend of ransomware attacks detection via intelligent algorithms

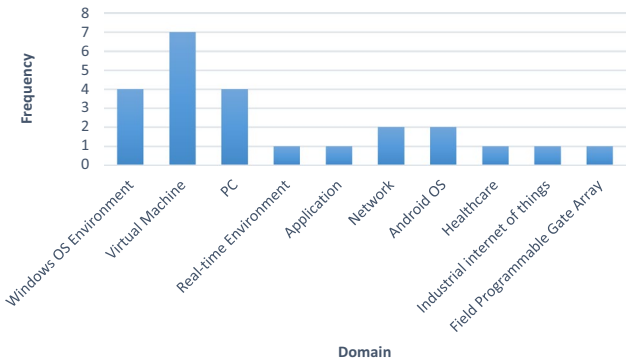

Fig. 4 Domain of applying the machine learning algorithm to detect ransomware

b; Wan et al. 2018), virtual machine (Ashraf et al. 2019; Cohen and Nissim 2018; Harikrishnan and Soman 2018; Lu et al. 2017; Maniath et al. 2017; Sharmeen et al. 2020; Shaukat and Ribeiro 2018; Verma et al. 2018), PCs (Cusack et al. 2018; Poudyal et al. 2018; Vinayakumar et al. 2017; Zhang et al. 2019a, b), healthcare (Maimo et al. 2019), application (Homayoun et al. 2019), Android (Bibi et al. 2019; Scalas et al. 2019), real-time environment (Daku et al. 2018), Twitter (Vinayakumar et al. 2017), industrial internet of things (Al-Hawawreh and Sitnikova 2019), FPGA (Alrawashdeh and Purdy 2018) and Microsoft Windows environment (Agrawal et al. 2019; Alhawi et al. 2018; Bae et al. 2019; Sgandurra et al. 2016).

Figure 3 depicts the frequency of application domains. The domain that has the highest patronage is a virtual environment with seven applications of machine learning algorithms—followed by PC and Windows OS, each having four applications of machine learning algorithms to detect ransomware. Furthermore, network and Android environments each have 2 applications of machine learning to detect ransomware. Finally, healthcare, application, real-time environment, Field Programmable Gate Array and industrial internet of things environments with one application each as shown in Fig. 4.

7.4 Family of ransomware attacks detected via the intelligent algorithms

The category of ransomware detected through the intelligent algorithms are presented in this section. We have two sub-set of ransomware: crypto and locker ransomware. Crypto denied users access to all or selected files using cryptography technology (Homayoun et al. 2019), such as advanced encryption standard (AES) or Riverest, Shamir, and Adleman (RSA) (Savage et al. 2015). The Filecoder, Teslacrypt, CryptoFortress are crypto strains that attack personal computers (Scaife et al. 2016). Resolving the attacks originate from crypto is difficult, and reversing the

damage made by the crypto may be irreversible. Crypto is arguably one of the most prominent ransomware used by hackers (Kok et al. 2019).

Locker ransomware hijacks services on the computer systems of the victim (Pathak and Nanded 2016). The services that are prompt to seize and denied access include input devices, applications, or desktop (Savage et al. 2015). The Urausy, Reventon, and VirLock are some PC-based ransomware (Kharraz et al. 2015). The infected system is left with the capability of performing activities related to payment (Al-rimy et al. 2018). Locker ransomware does not temper with the underlying OS or user files, and removing it resolves the trouble and returns the system to its safe state (Al-rimy et al. 2018).

Figure 5 depicts the families of ransomware as used in different works of literature extracted from the survey. Cryptowall and cryptolocker are the families of ransomware that has the highest frequency of 14 each as indicated by the longest bars—followed by Teslacrypt and Cerber having 13 and 12 frequencies of literature appearance, respectively. Then WannaCry having 11 frequency of literature appearance. Then Locky and Torrentlocker with each having ten frequency of literature appearance. Then Reventon having seven frequency of literature appearance. CTB-Locker and Petya followed them with each having six frequency of literature appearance. Next is Sage with five frequency of literature appearance. Then Hidden Tear, Jigsaw, Koler, Citroni, Kovter, Locker, Pgpcoder, Trojan ransom, and Ransomware with each having three frequency of literature appearance. They were followed by Filecryptor, Cryrar, CrptXXX, Fusob, Matsna, and Maktub, with each having two frequency of literature Appearance. Finally, a total of 48 different ransomware families are grouped as other families, with each having one frequency appearance in the literature.

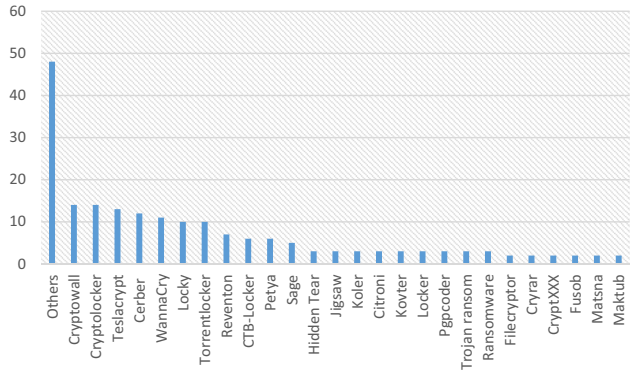

Fig. 5 families of ransomware and frequencies

8 Challenges and future research direction

Challenges found in the literature are discussed in this section. The possible methodology to solve the challenges pointed out in the survey is provided as a guide to researchers. The new research directions are provided from the perspective of deep learning and big data analytics.

8.1 Deep learning perspectives

The survey has shown that deep learning algorithms are gaining tremendous attention in detecting ransomware attacks. Many deep learning architectures that prove to be effective and efficient in solving real-world problems were not applied or under-exploit for detecting ransomware.

In some cases, it is challenging to differentiate traffic patterns produced by ransomware attacks from normal traffic patterns. Therefore, both scramble for a shared folder. Furthermore, the traffic pattern of both the files is similar for the application compressing the files (Maimo et al. 2019). Therefore, detecting the ransomware by the machine learning algorithms in this environment will be highly challenging. We propose multi-tasking learning as a possible solution to mitigate this challenge by exploiting their similarities and dissimilarities.

Detecting new states or differentiating between different machine states is a very challenging problem to tackle. The volatile memory is considered as a whole without knowing the running status of the system, safe or affected by malware, and unable to detect the exact process that characterizes the malware (Cohen and Nissim 2018). So, detecting ransomware attacks in this situation becomes challenging for machine learning algorithms. We suggest an intensive investigation of the factors that categorize the process that represent the malware. Subsequently, build a deep learning classifier for detecting the ransomware attacks.

It has been observed in the survey that the ransomware attack is irreversible, which further compounded the complex nature of ransomware attacks (Al-rimy et al. 2018). Therefore, the best approach is to detect and prevent it from the occurrence. We suggest the application of deep recurrent neural networks because it is the ability to memorize sequential events.

The ransomware attacks are of different types. To avoid the tedious process of building classifiers from scratch for the various ransomware attacks, waste of time, and computational cost, we propose the application of transfer learning for detecting ransomware with similar characteristics. This is because the transfer learning allowed a saved trained model to be used to solve a similar problem. The other deep learning models that can be exploited in

detecting ransomware to unravel their effectiveness and efficiency regarding ransomware detection include generative adversarial network, biologically plausible network, efficient inference, deep reinforcement learning, and explainable deep learning algorithms.

The ransomware attacks affect different platforms, e.g., network, PC, virtual machine, operating system, etc. detecting ransomware attacks on other platforms with a single model can be done through multi-task learning by exploiting the commonalities and differences among these platforms. Choosing the appropriate intelligent algorithm for detecting ransomware is mostly not a straightforward issue. It involves a lot of permutation (Kok et al. 2019). This is because of the different characteristics and nature of the ransomware attacks datasets. That is why an appropriate algorithm to handle the data set is not a straightforward issue. We propose researchers to work on a hybrid of transfer learning and multi-task learning that can take multiple families of ransomware attacks.

The shallow machine learning algorithms used in the detection of ransomware attacks require feature extraction techniques to extract the main features. Different feature extraction techniques need to be experiment with to determine the best method that can extract the best features. The best set of features are manually developed, which is tedious and time-consuming before feeding the best features into the algorithms for modeling. Feature extraction is an additional cost on the machine learning process of detecting the ransomware, and it can cause bias. To eliminate this tedious and time-consuming step, deep learning algorithms should be applied because deep learning algorithms do not require manual data engineering, including feature extraction. Acharya et al. (2017) proves in the literature that training of deep learning algorithm without feature extraction produce better performance than training with data with feature extraction. We suggest exploring deep learning algorithms for the detection of ransomware attacks to avoid manual feature engineering to improve detection accuracy.

In some instances, the input features required to develop ransomware attack detection mechanisms can be ambiguous. In such a situation, a deep Boltzmann machine is suggested because it can incorporate feedback in a top-down manner with ambiguous inputs. Ransomware attacks can be in the form of 2D, like images, the shallow machine learning algorithms are inadequate in handling images. However, CNN is under exploit in the detection of the ransomware, the CNN and it is variant such as the ResNet, Google Inception, DenseNet, etc. from the family of the deep learning algorithm is well suited for handling images, it can be used in the future to develop ransomware attacks detection system involving images.

8.2 Big data perspective

It was found from the survey that the use of intelligent algorithms in detecting ransomware in big data architecture is a virgin research area with a lot of challenges begging for machine learning solutions. The security challenges on big data architecture related to ransomware are discussed as follows:

Big data architectures are the main target of ransomware attackers because the attackers' motivation was to make money from their victims (Song et al. 2016). The big data outfits are mostly used for financial/business purposes, and data-intensive businesses become the likely lucrative targets of the attackers (Chong 2017). By it is nature, big data architecture mostly run on clusters of commodity hardware involving thousands of computer systems. Consequently, the data stored on the big data platform is housed in those clusters (Abdullahi et al. 2016). Thus, accommodating it is a huge volume. The ransomware can quickly attack one system in a big data platform and get spread fast to cover the entire architecture to pose the following challenges:

The ransomware attack may get spread to affect the entire big data platform because the platforms do not use confidential level access (CLA), or such access is made automatically. The reason for not using CLA is usually to allow seamless access to data kept in any part of the clusters hosting the big data platform. There may be a high possibility of compliance from the big data owners due to the massive volume of data that might be affected by the ransomware attack. As the recovery of such a large volume of data may be discouraging, cumbersome, and time-consuming. If the attack is made by symmetric crypto-malware, data recovery could be made through reverse engineering. However, running reverse engineering to recover the whole data stored on a big data platform may require a lot of computer time and high disk read and write activities, which may likely lead to the failure of some parts of the system. This can make some nodes to be unhealthy or running out of memory etc. We suggest researchers deploy a machine learning approach to build hybrid deep learning algorithms for detecting ransomware attacks on big data platforms.

9 Conclusions

The paper proposes to conduct a survey dedicated to the application of intelligent algorithms in detecting ransomware attacks, including synthesis and analysis. The survey presents an intelligent algorithm's solutions to ransomware attack detection. The survey examines the performance of machine learning defense mechanisms in detecting ransomware attacks. It is found that the applications of intelligent algorithms to detect ransomware is in an early stage

but is growing. The synthesis and analysis of the literature regarding machine learning applications in detecting ransomware are presented in the survey. The survey revealed new research directions from the perspective of deep learning and big data analytics for the future development of the research area.

References

Digital Guardian (2019) A history of ransomware attacks: the biggest and worst ransomware attacks of all time. https://digitalguardian.com/blog/history-ransomware-attacks-biggest-and-worst-ransomware-attacks-all-time. Accessed 17 Dec 2019

Abdullahi AU, Ahmad R, Zakaria NM (2016) Big data: performance profiling of meteorological and oceanographic data on hive. In: Paper presented at the 2016 3rd international conference on computer and information sciences (ICCOINS).

Acharya UR, Fujita H, Oh SL, Hagiwara Y, Tan JH, Adam M (2017) Application of deep convolutional neural network for automated detection of myocardial infarction using ECG signals. Inf Sci 415:190–198

Agrawal R, Stokes JW, Selvaraj K, Marinescu M (2019) Attention in recurrent neural networks for ransomware detection. In: Paper presented at the ICASSP 2019–2019 IEEE international conference on acoustics, speech and signal processing (ICASSP).

Ahmadian MM, Shahriari HR (2016) 2entFOX: a framework for high survivable ransomwares detection. In: 2016 13th international iranian society of cryptology conference on information security and cryptology (ISCISC), 7-8 Sept 2016. IEEE, Tehran, Iran, pp 79–84

Al-Hawawreh M, Sitnikova E (2019) Leveraging deep learning models for ransomware detection in the industrial internet of things environment. In: Paper presented at the 2019 military communications and information systems conference (MilCIS).

Alhawi OM, Baldwin J, Dehghantanha A (2018) Leveraging machine learning techniques for windows ransomware network traffic detection. Cyber Threat Intell 70:93–106

Almashhadani AO, Kaiiali M, Sezer S, O'Kane P (2019) A multi-classifier network-based crypto ransomware detection system: a case study of locky ransomware. IEEE Access 7:47053–47067

Alrawashdeh K, Purdy C (2018) Ransomware detection using limited precision deep learning structure in fpga. In: Paper presented at the NAECON 2018-IEEE national aerospace and electronics conference.

Al-rimy BAS, Maarof MA, Shaid SZM (2018) Ransomware threat success factors, taxonomy, and countermeasures: a survey and research directions. Comput Secur 74:144–166

Al-rimy BAS, Maarof MA, Shaid SZM (2019) Crypto-ransomware early detection model using novel incremental bagging with enhanced semi-random subspace selection. Future Gener Comput Syst 101:476–491

Amanullah MA, Habeeb RAA, Nasaruddin FH, Gani A, Ahmed E, Nainar ASM, Imran M (2020) Deep learning and big data technologies for IoT security. Comput Commun. https://doi.org/10.1016/j.comcom.2020.01.016

Andronio N, Zanero S, Maggi F (2015) Heldroid: dissecting and detecting mobile ransomware. In: Paper presented at the international symposium on recent advances in intrusion detection.

Ashraf A, Aziz A, Zahoora U, Khan A (2019) Ransomware analysis using feature engineering and deep neural networks. arXiv preprint. http://arxiv.org/abs/1910.00286

Aurangzeb S, Aleem M, Iqbal MA, Islam MA (2017) Ransomware: a survey and trends. J Inf Assur Secur 6(2):48–58

Bae SI, Lee GB, Im EG (2019) Ransomware detection using machine learning algorithms. Concurr Comput Pract Exp 32:e5422

Berrueta E, Morato D, Magaña E, Izal M (2019) A survey on detection techniques for cryptographic ransomware. IEEE Access 7:144925–144944

Bhardwaj A, Avasthi V, Sastry H, Subrahmanyam G (2016) Ransomware digital extortion: a rising new age threat. Indian J Sci Technol 9(14):1–5

Bibi I, Akhunzada A, Malik J, Ahmed G, Raza M (2019) An effective android ransomware detection through multi-factor feature filtration and recurrent neural network. In: Paper presented at the 2019 UK/China Emerging Technologies (UCET).

Breiman L (2001) Random forests. Mach Learn 45(1):5–32

Chaudhary R, Aujla GS, Kumar N, Zeadally S (2018) Lattice based public key cryptosystem for internet of things environment: challenges and solutions. IEEE Internet Things J 6:4897–4909

Chen J, Wang C, Zhao Z, Chen K, Du R, Ahn G-J (2017a) Uncovering the face of android ransomware: characterization and real-time detection. IEEE Trans Inf Forensics Secur 13(5):1286–1300

Chen Y-C, Li Y-J, Tseng A, Lin T (2017b) Deep learning for malicious flow detection. In: Paper presented at the 2017 IEEE 28th annual international symposium on personal, indoor, and mobile radio communications (PIMRC).

Chong H (2017) SeCBD: the application idea from study evaluation of ransomware attack method in big data architecture. Procedia Comput Sci 116:358–364

Cohen A, Nissim N (2018) Trusted detection of ransomware in a private cloud using machine learning methods leveraging meta-features from volatile memory. Expert Syst Appl 102:158–178

Connolly LY, Wall DS (2019) The rise of crypto-ransomware in a changing cybercrime landscape: taxonomising countermeasures. Comput Secur 87:101568

Conti M, Gangwal A, Ruj S (2018) On the economic significance of ransomware campaigns: a bitcoin transactions perspective. Comput Secur 79:162–189

Cusack G, Michel O, Keller E (2018) Machine learning-based detection of ransomware using SDN, pp 1–6. https://doi.org/10.1145/3180465.3180467. Accessed 17 Dec 2019

Daku H, Zavarsky P, Malik Y (2018) Behavioral-based classification and identification of ransomware variants using machine learning. In: Paper presented at the 2018 17th IEEE international conference on trust, security and privacy in computing and communications/12th IEEE international conference on big data science and engineering (TrustCom/BigDataSE).

Damshenas M, Dehghantanha A, Mahmoud R (2013) A survey on malware propagation, analysis, and detection. Int J Cyber Secur Digit Forensics 2(4):10–30

Druva (2017) Druva releases annual enterprise ransomware report. https://www.globenewswire.com/news-release/2017/06/28/1217348/0/en/Druva-Releases-Annual-Enterprise-Ransomware-Report.html. Accessed 17 Dec 2019

Feizollah A, Anuar NB, Salleh R, Wahab AWA (2015) A review on feature selection in mobile malware detection. Digit Investig 13:22–37

Fernandez Maimo L, Huertas Celdran A, Perales Gomez AL, Clemente G, Félix J, Weimer J, Lee I (2019) Intelligent and dynamic ransomware spread detection and mitigation in integrated clinical environments. Sensors 19(5):1114

Frank E, Hall MA, Witten IH (2016) The WEKA workbench. Morgan Kaufmann

Gómez-Hernández J, Álvarez-González L, García-Teodoro P (2018) R-Locker: thwarting ransomware action through a honeyfile-based approach. Comput Secur 73:389–398

Hansen SS, Larsen TMT, Stevanovic M, Pedersen JM (2016) An approach for detection and family classification of malware based on behavioral analysis. In: Paper presented at the 2016 international conference on computing, networking and communications (ICNC).

Haque IRI, Neubert J (2020) Deep learning approaches to biomedical image segmentation. Inform Med Unlocked 18:100297

Harikrishnan N, Soman K (2018) Detecting ransomware using GURLS. In: Paper presented at the 2018 second international conference on advances in electronics, computers and communications (ICAECC).

Hatcher WG, Yu W (2018) A survey of deep learning: platforms, applications and emerging research trends. IEEE Access 6:24411–24432

Homayoun S, Dehghantanha A, Ahmadzadeh M, Hashemi S, Khayami R, Choo K-KR, Newton DE (2019) DRTHIS: Deep ransomware threat hunting and intelligence system at the fog layer. Future Gener Comput Syst 90:94–104. https://doi.org/10.1016/j.future.2018.07.045

Javaheri D, Hosseinzadeh M, Rahmani AM (2018) Detection and elimination of spyware and ransomware by intercepting Kernel-Level system routines. IEEE Access 6:78321–78332

Joseph DP, Norman J (2020) A review and analysis of ransomware using memory forensics and its tools. Smart intelligent computing and applications. Springer, Berlin, pp 505–514

Kharraz A, Robertson W, Balzarotti D, Bilge L, Kirda E (2015) Cutting the gordian knot: a look under the hood of ransomware attacks. In: Paper presented at the international conference on detection of intrusions and malware, and vulnerability assessment.

King D (2017) Detect and protect. ITNOW 59(4):54–55

Kok S, Abdullah A, Jhanjhi N, Supramaniam M (2019) Ransomware, threat and detection techniques: a review. Int J Comput Sci Netw Secur 19(2):136

Lachtar N, Ibdah D, Bacha A (2019) The case for native instructions in the detection of mobile ransomware. IEEE Lett Comput Soc 2:16–196

LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521(7553):436–444

Lee S, Kim HK, Kim K (2019) Ransomware protection using the moving target defense perspective. Comput Electr Eng 78:288–299

Lu T, Zhang L, Wang S, Gong Q (2017) Ransomware detection based on v-detector negative selection algorithm. In: Paper presented at the 2017 international conference on security, pattern analysis, and cybernetics (SPAC).

Maigida AM, Olalere M, Alhassan JK, Chiroma H, Dada EG (2019) Systematic literature review and metadata analysis of ransomware attacks and detection mechanisms. J Reliab Intell Environ 5(2):67–89

Maniath S, Ashok A, Poornachandran P, Sujadevi V, Sankar AP, Jan S (2017) Deep learning LSTM based ransomware detection. In: Paper presented at the 2017 recent developments in control, automation and power engineering (RDCAPE).

Martín A, Hernandez-Castro J, Camacho D (2018) An in-depth study of the Jisut family of android ransomware. IEEE Access 6:57205–57218

Min D, Park D, Ahn J, Walker R, Lee J, Park S, Kim Y (2018) Amoeba: an autonomous backup and recovery SSD for ransomware attack defense. IEEE Comput Archit Lett 17(2):245–248

Mohammadi M, Al-Fuqaha A, Sorour S, Guizani M (2018) Deep learning for IoT big data and streaming analytics: a survey. IEEE Commun Surv Tutor 20(4):2923–2960

Muna A-H, den Hartog F, Sitnikova E (2019) Targeted ransomware: a new cyber threat to edge system of brownfield industrial internet of things. IEEE Internet Things J 6:7137–7151

National Vulnerability Databasa (2017) CVE-2017-0144 Detail. https://nvd.nist.gov/vuln/detail/CVE-2017-0144. Accessed 17 Dec 2019

O'Kane P, Sezer S, Carlin D (2018) Evolution of ransomware. IET Networks 7(5):321–327

Pathak P, Nanded YM (2016) A dangerous trend of cybercrime: ransomware growing challenge. Int J Adv Res Comput Eng Technol 5(2):371–373

Pluskal O (2015) Behavioural malware detection using efficient SVM implementation. In: Paper presented at the proceedings of the 2015 conference on research in adaptive and convergent systems.

Poudyal S, Subedi KP, Dasgupta D (2018) A framework for analyzing ransomware using machine learning. In: Paper presented at the 2018 IEEE symposium series on computational intelligence (SSCI).

Richardson R, North MM (2017) Ransomware: evolution, mitigation and prevention. Int Manag Rev 13(1):10

Sabharwal S, Sharma S (2020) Ransomware attack: India issues red alert. Emerging technology in modelling and graphics. Springer, Berlin, pp 471–484

Savage K, Coogan P, Lau H (2015) The evolution of ransomware. Symantec, Mountain View

Scaife N, Carter H, Traynor P, Butler KR (2016) Cryptolock (and drop it): stopping ransomware attacks on user data. In: Paper presented at the 2016 IEEE 36th international conference on distributed computing systems (ICDCS).

Scalas M, Maiorca D, Mercaldo F, Visaggio CA, Martinelli F, Giacinto G (2019) On the effectiveness of system API-related information for android ransomware detection. Comput Secur 86:168–182

Sgandurra D, Muñoz-González L, Mohsen R, Lupu EC (2016) Automated dynamic analysis of ransomware: benefits, limitations and use for detection. arXiv preprint. http://arxiv.org/abs/1609.03020

Shakir HA, Jaber AN (2017) A short review for ransomware: pros and cons. In: Paper presented at the international conference on P2P, parallel, grid, cloud and internet computing.

Sharmeen S, Ahmed YA, Huda S, Koçer B, Hassan MM (2020) Avoiding future digital extortion through robust protection against ransomware threats using deep learning based adaptive approaches. IEEE Access. 8:24522–24534

Shaukat SK, Ribeiro VJ (2018) RansomWall: a layered defense system against cryptographic ransomware attacks using machine learning. In: Paper presented at the 2018 10th international conference on communication systems and networks (COMSNETS).

Shukla M, Mondal S, Lodha S (2016) Poster: locally virtualized environment for mitigating ransomware threat. In: Paper presented at the proceedings of the 2016 ACM SIGSAC conference on computer and communications security.

Song S, Kim B, Lee S (2016) The effective ransomware prevention technique using process monitoring on android platform. Mobile Inf Syst 2016:9

Su D, Liu J, Wang X, Wang W (2018) Detecting android locker-ransomware on chinese social networks. IEEE Access 7:20381–20393

Symantec (2019) 2019 internet security threat report. https://www.symantec.com/en/uk/security-center/threat-report. Accessed 17 Dec 2019

Verma M, Kumarguru P, Deb SB, Gupta A (2018) Analysing indicator of compromises for ransomware: leveraging IOCs with machine learning techniques. In: Paper presented at the 2018 IEEE international conference on intelligence and security informatics (ISI).

Villalba LJG, Orozco ALS, Vivar AL, Vega EAA, Kim T-H (2018) Ransomware automatic data acquisition tool. IEEE Access 6:55043–55052

Vinayakumar R, Soman K, Velan KS, Ganorkar S (2017) Evaluating shallow and deep networks for ransomware detection and classification. In: Paper presented at the 2017 international conference

on advances in computing, communications and informatics (ICACCI).

Vinayakumar R, Alazab M, Jolfaei A, Soman K, Poornachandran P (2019) Ransomware triage using deep learning: twitter as a case study. In: Paper presented at the 2019 cybersecurity and cyberforensics conference (CCC).

Wan Y-L, Chang J-C, Chen R-J, Wang S-J (2018) Feature-selection-based ransomware detection with machine learning of data analysis. In: Paper presented at the 2018 3rd international conference on computer and communication systems (ICCCS).

Yaqoob I, Ahmed E, Rehman MH, Ahmed AIA, Al-garadi MA, Imran M, Guizani M (2017) The rise of ransomware and emerging security challenges in the internet of things. Comput Netw 129:444–458

Zhang B, Xiao W, Xiao X, Sangaiah AK, Zhang W, Zhang J (2019) Ransomware classification using patch-based CNN and self-attention network on embedded N-grams of opcodes. Future Gener Comput Syst 110:708–720

Zhang H, Xiao X, Mercaldo F, Ni S, Martinelli F, Sangaiah AK (2019) Classification of ransomware families with machine learning based on N-gram of opcodes. Future Gener Comput Syst 90:211–221. https://doi.org/10.1016/j.future.2018.07.052

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.