Industrial Internet of Things Based Ransomware Detection using Stacked Variational Neural Network

Muna AL-Hawawreh
University of New South Wales
Canberra, Australia
m.al-hawawreh@student.adfa.edu.au

Elena Sitnikova
University of New South Wales
Canberra, Australia
e.sitnikova@adfa.edu.au

ABSTRACT

To protect the Industrial Internet of Things (IIoT) systems against ransomware attacks, their host machines systems activities need to be efficiently monitored by an efficient detection model that is able to accurately detect ransomware behavior and trigger an alarm before its impact extends to the critical control systems. However, the detection models for these hosts' machines encounter significant challenges in dealing with a high dimension data, few numbers of trained observations, and the dynamic behavior of ransomware. Therefore, there is a need for an efficient detection model that can address these challenges. In this paper, we propose a detection model based on the stacked Variational Auto-Encoder (VAE) with a fully connected neural network that is able to learn the latent structure of system activities and reveal the ransomware behavior. Further, we also come up with a data augmentation method based on VAE for generating new data that can be used in training a fully connected network in order to improve the generalized capabilities of the proposed detection model. The results showed that our proposed model achieved considerable performance in detecting ransomware activities.

CCS Concepts

• Security and privacy→Intrusion/anomaly detection and malware mitigation→Intrusion detection systems

Keywords

IIoT; ransomware; LAN; API; windows; detection; deep learning.

1. INTRODUCTION

Nowadays, as Industrial Internet of Things (IIoT) technologies are becoming increasingly deployed within critical infrastructure such as smart grid, energy plants, water treatment systems, healthcare systems, transportation, and nuclear plants and among others, cybersecurity is becoming ever more essential, particularly, with expanding attacks surface and increasing the severity of its disruptive impact which may lead to threatening human life. With this deployment, advanced threats could easily find its way to the production systems through various cyber and physical vectors, one of the most recent threats is ransomware attack. It is a digital extortion attack that works in injecting malware in the system to deny access to the files or/ and system forcing the users into payment digital currency [1]. The IIoT systems are considered an attractive target for ransomware authors due to the significant financial values for denying or locking a critical data or/and system in this deployment [2, 3].

These IIoT systems expand over a large scale, and they have various devices, technologies that range from edge physical system, mobile, and cloud to the enterprise's devices, and each of them has its own vulnerabilities and threats leading to imposing significant challenges for IIoT systems security [4]. For example, Information Technology-Local Area Network (IT-LAN) at edge level (e.g., plant level), and Application Programming Interfaces (API) devices at the enterprise level are critical parts of these systems, and they add their own security challenges for IIoT systems [5, 6]. All simple services operations at the plant are performed in IT-LAN while the end users utilize the API to access the cloud storage and edge physical systems. A ransomware attack could succeed in accessing the deployed workstations in LAN or API devices which usually run over windows operating systems through the attached file, web link, shared folder, or a physical vector [5, 7]. In this situation, the ransomware attackers could easily proceed and spread ransomware infection to more effective and critical devices in the system such as control devices if no countermeasures are deployed [8], therefore, deploying a detection model that is able to monitor system activities, identify system behavior and raises an alarm, when there is a potentially malicious event relating to the ransomware, could considerably safeguard the critical control systems and minimize the ransomware impact.

A Rich literature exists for detecting and preventing ransomware attacks [9-11], however, detecting ransomware based on artificial intelligence and data mining techniques [12-15] are still less adopted and the existing proposed methods are also still suffering from various challenges such as high level of human intervention in identifying the latent features which could not be flexible and adaptive for the emerging ransomware variants, and they have less generalization capabilities which lead to imposing a burden on the early ransomware detection and producing high false alarms. So, there is a need for more efficient and accurate detection models that are able to reveal the complex versions of ransomware which mimic the normal user behavior. Based on these requirements, we utilize deep learning techniques to build an efficient solution for detecting ransomware as they are be able to define the characteristics of ransomware activities and deal automatically with high dimensional system data. Our proposed model is able to deal with continually evolving and changing of ransomware behavior by extracting complex and meaningful features from system data automatically and without the need to manual feature engineering techniques.

BDIOT 2019, August 22–24, 2019, Melbourne, VIC, Australia

DOI: https://doi.org/10.1145/3361758.3361763

In this paper, we utilize the VAE for unsupervised learning and stacked with a fully connected neural network for supervised learning and model tuning parameters. VAE can learn an efficient representation for collected data and reduce the dimension of data automatically, which in turn shows its capabilities in identifying the dynamic behavior of ransomware attacks in the host machines in IIoT systems. Further, we present a data augmentation method based on VAE to address the trained data paucity.

The remainder of the paper is organized as follow: Section 2 presents background and the main research problem, Section 3 provides a description for proposed model architecture, Section 4 presents the experimental results, and the paper is concluded in Section 5.

2. BACKGROUND AND PROBLEM

2.1 Previous works

Several related works have provided ransomware detection models. For example, the authors of [12] utilized mutual information criterion and regularized logistic regression for detecting crypto-ransomware attacks in windows system. The mutual information was utilized to reduce the dimensionality of data and choose the best features and then utilized classifier to identify the ransomware attack. Similarly, different machine learning methods were used by [14] to detect crypto-ransomware. The authors used power usage time series which was divided to set of subsamples, and each subsample was also divided into set of subsamples in order to overcome the distributed features problem. The experimental results showed that the Nearest Neighbor (KNN) achieved the best performance. Although the above mechanisms achieved reasonable performance to some extent, they need to perform a manual feature engineering process to guarantee a better performance for detection model.

The study of [16] proposed R-PackDroid detection system to classify ransomware and goodware based on information that was extracted from system API packages. The authors used the random forest as a classifier which demonstrated good performance. However, this model depended only on static analysis features which could fail in detecting ransomware that uses obscuration techniques. The authors of [13] utilized various machine learning techniques to detect ransomware based on the network traffic. The results showed that the detection model based on decision tree achieved the highest accuracy. In [15], the authors used different machine learning algorithms for detecting ransomware based on features extracted from registry events, DLL events, and file systems to registry transitions. Although the Multiple Layer Perceptron (MLP) demonstrated the best performance compared with other machine learning techniques, the proposed model has difficulty in identifying new ransomware pattern.

The authors of [17] used Deep Neural Network (DNN) to reveal a ransomware attack. Their proposed model was built based on features that were extracted from HTTP packet payload inspection. The results proved the efficiency of DNN in detecting ransomware with accuracy 93.9%. Furthermore, the study of [18] used a Long Short-Term Memory (LSTM) deep learning algorithm for detecting ransomware. The LSTM detection model was built using API calls which were collected during runtime, and it achieved an efficient performance in the classification process. The authors of [19] presented an intelligent detection system based on deep learning and One Class Support Vector Machine (OCSVM) for revealing the ransomware attacks. The LSTM and Convolutional Neural Network (CNN) were used in the first stage for classifying the collected API calls and then converting them to the numerical values to detect if this activity is goodware or ransomware. As a consequence, the best model of these algorithms was used to extract the final vector for each ransomware family and fed it to OCSVM. The authors used three OCSVM for each ransomware family; the output of these OCSVMs was fed into a final decision module for classifying ransomware samples. Although the above mechanisms achieved a considerable performance in detecting ransomware attacks, they had a high complexity in the data processing and classification processes which could delay the detection process.

Along this direction, we interest in deep learning techniques as they have shown their effectiveness in revealing malicious activities. We utilize the VAE for learning a latent structure of unlabeled data and then stacking it with a fully connected neural network for training based on a labeled data. The proposed model is able to deal with high dimension and unstructured collected data, and identify the ransomware activities.

2.2 Problem Statement

In summary, existing ransomware detection approaches based on artificial intelligence and data mining techniques, particularly classical machine learning algorithms suffer from high levels of human expert's intervention in processing data, heavy dependence on labeled data and relevant features selection, and unsatisfying detection performance. Further, existing approaches have a limited power for complex functions and limited generalization capabilities which can become challenge with the dynamic behavior of ransomware attacks. While our work utilizes both unlabeled and labeled data, performs an automatic features engineering, and provide generalization capabilities.

The problem statement of this work is as follows: we aim at ransomware detection model based on deep generative learning method. Specifically speaking, we utilize the VAE as a foundation for building our proposed detection model which takes the advantage of its significant parameters to control the latent variable learning processes through learning the underlying probability distribution of collected data. So, it combines the advantages of deep AE and statistical learning concepts in reducing the data dimension and learning the latent representation of trained data. Further, our model uses data augmentation which eliminates the need for new trained data and the stacking technique which has the capability to reduce the computational cost as discussed in [20].

2.3 Variational Auto-Encoder (VAE)

The Auto-Encoder (AE) is symmetric unsupervised neural network in which the input data is encoded using the hidden layer, approximates the minimum error and gains the best features representation. AE learns the identity function of input data and maintain it in the memory then it reconstructs the input data with high precision [21]. VAE inherits the behavior of AE (i.e, encoding and decoding) but with presence of statistical probability distribution model in the learning process. For ransomware detection, suppose A sets of collected data $\{x^{(1)}, x^{(2)}, x^{(3)},\ldots\ldots, x^{(n)}\}$ are used as input for VAE network, then the network model tries to learn its distribution by taking a sample of variables and mapping them through complicated functions. Mathematically speaking, the VAE tries to maximize the probability of each x in the data set using the following equation:

$$P(x) = \int P(x|z)P(z)dz \tag{1}$$

Where $P(x|z)$ is the probability function of x given to its latent variable z. The main clue of VAE is to try to sample values of z

that are likely to have x and compute its probability $P(x)$ from those values. Each x^i $(i = 1, 2, 3, \dots n)$ is encoded using the posterior distribution $q_\theta(z|x^i)$ which describes the distribution of latent variable z based on value of x and its parameters weight and bias are represented by $\theta\,\{w, b\}$, it helps in computing $E_{z \sim q} P_\emptyset(x^i|z)$ which represent the decoder process and its parameters $\emptyset$. The decoder takes the z as input and outputs the parameter to the probability distribution. Here, z is sampled from an arbitrary distribution and $q_\theta(z|x^i)$ is any distribution. For performing the learning process, VAE utilize negative log likelihood with Kullback-Leibler divergence for computing the loss function l^i for each observation x^i according to the following equation:

$$l^i(\theta, \emptyset) = -E_{z \sim q_{\theta(z|x^i)}}\left[log_{p_\emptyset}(x^i|z)\right] + KL(q_\theta(z|x^i)||p(z) \qquad (2)$$

The first term $-E_{z \sim q_{\theta(z|x^i}}\left[log_{p_\emptyset}(x^i|z)\right]$ represents the expected negative log-likelihood for observation x^i which is taken based on encoder's distribution, and it helps the decoder in reconstruction the observed data. While the second term $KL(q_\theta(z|x^i)||p(z)$ represents the Kullback-Leibler divergence as a regularizer. It is used to match the encoder's distribution $q_\theta(z|x^i)$ to $p(z)$. In other words, it measures of how much the q is close to p. The network model is trained using stochastic gradient descent with step size γ, so the encoder and the decoder parameters are updated using $\theta \leftarrow \theta - \gamma \frac{\partial l}{\partial \theta}$, $\emptyset \leftarrow \emptyset - \gamma \frac{\partial l}{\partial \emptyset}$ respectively.

3. PROPOSED METHEDOLOGY

The key steps of our proposed methodology are training VAE in unsupervised manner, and then stacking the VAE-encoder and latent space parts with a fully connected neural network in order to learn from labeled data and tune the final model parameters. The details can be explained as follows:

3.1 VAE training phase

The VAE steps discussed above are followed to build the proposed detection model. The VAE is trained using unlabeled dataset A to reduce the data dimensionality and learn the latent variables that can contribute to reconstructing the input data. The overall network structure is illustrated in Figure 1. Firstly, the encoder (q) turns the input observation x into two parameters in the latent space, namely, mean μ and standard deviation σ, and then a variables z is randomly sampled from the latent distribution via $\mu + \exp(\sigma) * \varepsilon$, where epsilon (ε) is a random normal number. Finally, the decoder (P) maps the latent space variables back to the original input data x.

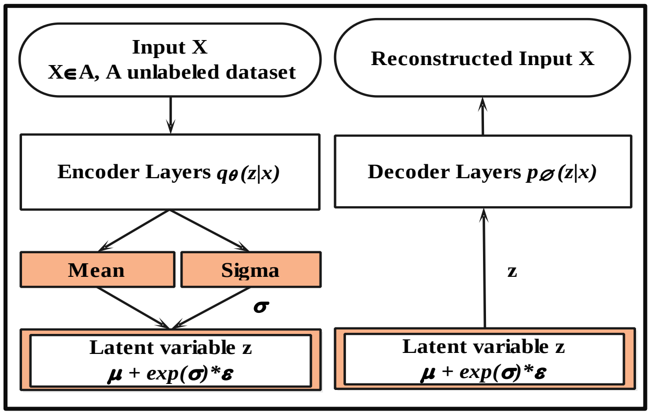

Figure 1. VAE network architecture

3.2 Data Augmentation

It is a method that artificially increases the number of observations without new data, it is commonly used in computer vision [22], however, we use it here to tackle the lack of sufficient trained observations which is usually a problem with ransomware attacks detection, and to build a generalized and efficient model is able to learn the main concept of underlying problem rather than memorizing the trained data. As VAE is a generative model, we utilize it to generate new data which can be used later for training a fully connected network. After trained VAE, the decoder part is only used to generate new dataset from dataset **A**, that is dataset **C**, to avoid getting samples too similar to the original ones **A**.

3.3 Fully connected network training phase

The first part of VAE, i.e., the encoder layers with latent space μ is stacked with new hidden layer and classifier output layer. During the training phase, the encoder layers with latent space part are frozen as they are already trained in the previous stage (i.e., VAE training phase) and only the fully connected neural network is trained in this phase. Given labeled dataset $B, (B \subset A)$, and new generated dataset C, the network is trained using the input data x and learn to figure out its label y (ransom or normal). This is systemically described in the left-hand side of Figure 2.

3.4 Network model parameters tuning phase

As depicted in the right-hand side of Figure 2, the VAE encoder and latent space part are active and the entire model is trained based on labeled data B in order to tune its parameters and build a more efficient trained model. The input data x and its label are passed through model layers and its parameters θ (i.e., weights and bias) are tuned based on stochastic gradient descent. It is worth mentioning here that any new real-time data set can be used only to tune this final model and there is no need to retrain it from scratch which indicates the possibility of deploying and training this model online based on mini-batch data.

3.5 Testing phase

The final trained model (see Figure 2) is represented as a decision engine to evaluate new observations set D $(D \not\subset A, B)$, and each new observation is classified as normal or ransom.

4. EXPERIMENTAL EVALUATION

4.1 Dataset Description

We used the dataset that was generated by the authors of [12] for windows ransomware research objective. It consists of 942 of well-known good application samples and 582 ransomware samples. These samples were analyzed dynamically and the data related to API invocations, registry keys, file operations, and directory operations were collected during the dynamic analysis. The total numbers of features of the available online dataset version are 14700 features including the binary label; we divided this data into two sets; 70% as training dataset and 30% as a testing dataset.

4.2 Experimental Results

The experiments were conducted using python and they were performed many times to choose the best parameters for our proposed model. The parameters of VAE consist of six hidden layers in addition to the three layers of latent space with neuron numbers 1500, 520, 128, 64, 64, 64, 128, 520, and 1500 respectively. Both SELU and Sigmoid were used as activation functions, while RMSprop as optimizer and the model was trained based on 250 batch size and 100 epochs using the loss function in

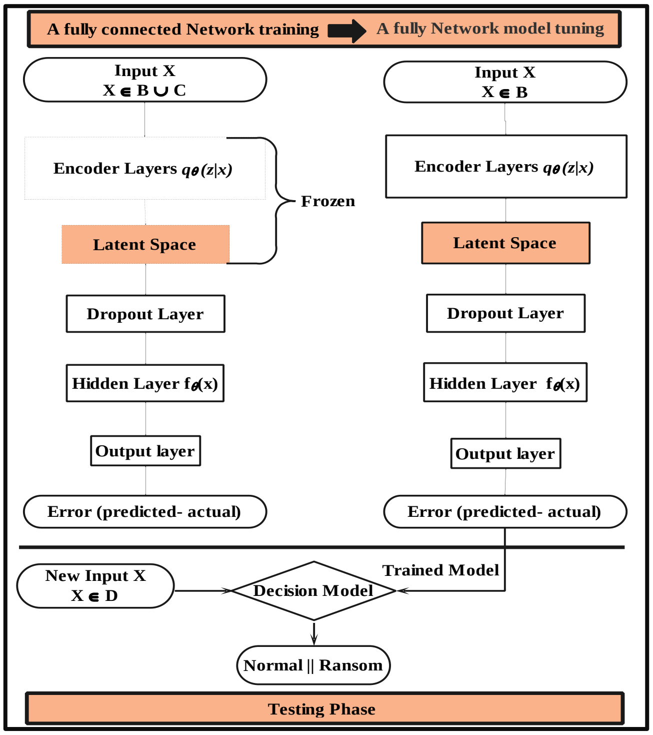

Figure 2. Fully connected network training and fully network model tuning architecture

equation 2, while a fully connected network has a new hidden layer (20 neurons), dropout layer (fraction= 0.5), and one neuron output layer, the binary cross-entropy as loss function, and adam as optimizer. In these experiments, we tested the capabilities of our proposed model to identify ransomware attacks. The performance evaluation metrics in Table 1 demonstrated that the system is able to distinguish between ransomware and legitimate samples with accuracy 92.81%, 99.47% for detection rate and 13.9 % for false positive rate.

Table 1. Our proposed model Performance metrics results

Performance Metric	Description	Value (%)
Accuracy	The total number of observations are correctly classified	92.81
Detection rate	The total number of actual attack observations that are correctly identified	99.47
False positive rate	The ratio of attack to normal observations that are incorrectly identified	13.9

To demonstrate the performance of our proposed model compared with existing approaches, we consider the same machine learning algorithms that were adopted by the authors of dataset [12] including Logistic Regression (LR), Support Vector Machine (SVM), and NaiveBase (NB). In addition, we consider other approaches that were presented in the state-of-art but for other datasets such as DNN [17] and random forest [16]. All these approaches are trained and tested using the same dataset that is used for evaluating our proposed model. As ransomware detection has the utmost importance, we used detection rate and false positive rate for performing comparison between our proposed model and existing approaches. Figure 3 shows the detection rates which are 89.08%, 89.66%, 90.23%, 91.38%, 94.83%, and 99.47% for LR, RF, SVM, DNN, NB, and our proposed model respectively. It can be noticed that our proposed model achieved the highest detection rate 99.47%. This indicates the capabilities of our proposed model in addressing the dynamic behavior of ransomware and its ability to learn the underlying features of such attacks.

The performance of existing approaches in Figure 4 shows that the false positive rate for LR, RF, SVM, DNN, our proposed model, and NB are 0.35%, 3.1%, 3.9%, 4.2%, 13.9 % and 53.71% respectively. Our proposed model achieved less performance compared with LR, RF, SVM, and DNN, but better than NB, this finds its cause that the data is unbalanced and apparently these existing approaches showed bias to the class with high observations (i.e., legitimate). This metric may not be very important in case of ransomware attack detection as classifying legitimate behavior as ransomware is not too significant compared with classifying ransomware as legitimate one. So, detecting ransomware attacks before encrypting data and extending for more critical storage and devices is much priority.

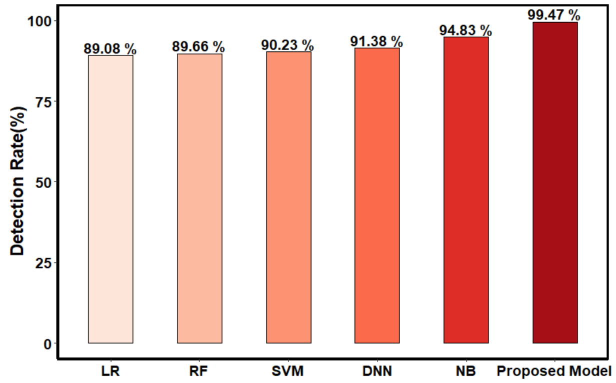

Figure 3. Comparison with existing approaches in term of detection rate

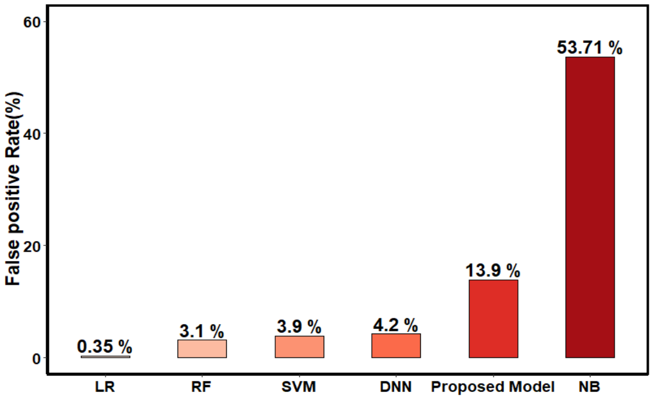

Figure 4. Comparison with existing approaches in term of false positive rate

Our proposed model is different from existing approaches that is used the VAE for learning the latent representation of data through its distribution, so it is able to figure out the underlying features of trained data. Using the latent space layer, the model learns the high-level features and reduced the dimension of data automatically. Further, it provided generalization capabilities with helping of data augmentation in the training a fully connected layer phase. Therefore, these traits emphasize that our proposed model is appropriate for dealing with the dynamic behavior of ransomware attacks and high dimension unlabeled and unbalanced data.

5. CONCLUSION

In this paper, we proposed a detection model based on stacked VAE and the fully connected neural network for the detecting ransomware attacks in the IIoT systems. The proposed approach automatically learnt the latent data structure and reduced the data dimension, additionally; it showed its efficiency in dealing with a few existing data observations and supporting the learning process from data augmentation process which was generated using VAE. It also showed its capability in dealing with unbalanced data and did not bias to the class with more observations compared with classical machine learning. At the ultimate, we showed that the proposed model is able to identify ransomware activities and it achieved a superior detection rate compared with the previously proposed models.

For further directions, we plan to test our model with multiple ransomware classes and improve its performance in distinguishing between legitimate and ransomware observations. We will involve the other data augmentation techniques in learning our model and to the possibility of involving other deep learning architecture and compare their performance. We will consider other unsupervised techniques to eliminate or reduce the labeled data use. Finally, we will study the detection model deployment and the cooperation between multiple hosts detection models in the IIoT system.

6. ACKNOWLEDGMENTS

Our thanks to Dr. Frank den Hartog for his valuable discussion.

7. REFERENCES

[1] Liska A, Gallo T. 2016. Ransomware: Defending against digital extortion*. " O'Reilly Media, Inc.".*

[2] Yaqoob I, Ahmed E, ur Rehman MH, Ahmed AI, Al-garadi MA, Imran M, Guizani M. 2017. The rise of ransomware and emerging security challenges in the Internet of Things. *Computer Networks*.

[3] Formby D, Durbha S, Beyah R. 2017. Out of control: Ransomware for industrial control systems. *InProc. RSA Conf*. pp. 8-16.

[4] Boye CA, Kearney P, Josephs M. 2018. Cyber-Risks in the Industrial Internet of Things (IIoT): Towards a Method for Continuous Assessment. *In International Conference on Information Security*. pp. 502-519. Springer, Cham.

[5] Kobara, K. 2016. Cyber physical security for industrial control systems and IoT. *IEICE TRANSACTIONS on Information and Systems*, pp. 787-795.

[6] Eliyahu R, Sadika O. 2018. System and method for identifying and preventing malicious api attacks. *United States patent application US 15/821*,124.

[7] Stouffer, K., J. Falco, and K. Scarfone. 2011. Guide to industrial control systems (ICS) security. *NIST special publication, 2011. **800**(82):* pp. 16-16.

[8] Al-Hawawreh. M, den Hartog . F, Sitnikova. E. 2019. TargetedRansomware: A New Cyber Threat to Edge System of Brownfield Industrial Internet of Things. IEEE Internet of Things. **DOI:** 10.1109/JIOT.2019.2914390.

[8] Zahra A, Shah MA. 2017. IoT based ransomware growth rate evaluation and detection using command and control blacklisting. *In 2017 23rd International Conference on Automation and Computing (ICAC)*. pp. 1-6. IEEE.

[9] Scaife N, Carter H, Traynor P, Butler KR. 2016. Cryptolock (and drop it): stopping ransomware attacks on user data. *In 2016 IEEE 36th International Conference on Distributed Computing Systems (ICDCS)* 2016. pp. 303-312. IEEE.

[10] Nieuwenhuizen D. 2017. A behavioural-based approach to ransomware detection. *MWR Labs Whitepaper*.

[11] Al-rimy B, Maarof M, Shaid S. 2018. Ransomware threat success factors, taxonomy, and countermeasures: A survey and research directions. *Computers & Security*. pp. 44-66.

[12] Sgandurra D, Muñoz-González L, Mohsen R, Lupu EC. 2016. Automated dynamic analysis of ransomware: Benefits, limitations and use for detection. ArXiv preprint arXiv:1609.03020.

[13] Alhawi M, Baldwin J, Dehghantanha A. 2018. Leveraging machine learning techniques for windows ransomware network traffic detection. *Cyber Threat Intelligence*. pp. 93-106.

[14] Azmoodeh A, Dehghantanha A, Conti M, Choo KK. 2017. Detecting crypto-ransomware in IoT networks based on energy consumption footprint. *Journal of Ambient Intelligence and Humanized Computing*.

[15] Homayoun S, Dehghantanha A, Ahmadzadeh M, Hashemi S, Khayami R. 2017. Know abnormal, find evil: frequent pattern mining for ransomware threat hunting and intelligence. *IEEE transactions on emerging topics in computing.*

[16] Maiorca D, Mercaldo F, Giacinto G, Visaggio CA, Martinelli F. 2017. R-PackDroid: API package-based characterization and detection of mobile ransomware. *In Proceedings of the symposium on applied computing*. pp. 1718-1723. ACM.

[17] Tseng A, Chen Y, Kao Y, Lin T. 2016. Deep learning for ransomware detection. *IEICE Tech. Rep*. 116(282). pp:87-92.

[18] Maniath S, Ashok A, Poornachandran P, Sujadevi VG, Sankar AP, Jan S. 2017. Deep learning LSTM based ransomware detection. *In 2017 Recent Developments in Control, Automation & Power Engineering (RDCAPE)*. pp. 442-446. IEEE.

[19] Homayoun S, Dehghantanha A, Ahmadzadeh M, Hashemi S, Khayami R, Choo KK, Newton DE. 2019. DRTHIS: Deep ransomware threat hunting and intelligence system at the fog layer. *Future Generation Computer Systems*. pp.94-104.

[20] Shone N, Ngoc TN, Phai VD, Shi Q. 2018. A deep learning approach to network intrusion detection. *IEEE Transactions on Emerging Topics in Computational Intelligence*. pp.41-50.

[21] Doersch C. 2016. Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908.*

[22] Shin HC, Tenenholtz NA, Rogers JK, Schwarz CG, Senjem ML, Gunter JL, Andriole KP, Michalski M. 2018. Medical image synthesis for data augmentation and anonymization using generative adversarial networks. *In International Workshop on Simulation and Synthesis in Medical Imaging*. pp. 1-11. Springer, Cham.