A 0-Day Aware Crypto-Ransomware Early Behavioral Detection Framework

Bander Ali Saleh Al-rimy(✉), Mohd Aizaini Maarof, and Syed Zainuddin Mohd Shaid

Faculty of Computing, Universiti Teknologi Malaysia (UTM), 81310 Johor Bahru, Johor, Malaysia
bnder32l@gmail.com, {aizaini, szainudeen}@utm.my

Abstract. Crypto-Ransomware exploits cryptography to hijack personal files and documents and hold them to ransom. Utilizing such technological leap, crypto-ransomware targets a wide range of systems, and platforms. Although many users, whether individuals or organizations, practice proactive security procedures like regular backup, advanced crypto-ransomware can bypass these countermeasures rendering the valuable data vulnerable to such extortion attack. Due to the irreversible nature of its damage, thwarting crypto-ransomware becomes challenging. Although several studies have been conducted to tackle crypto-ransomware detection problem, most of them dealt with it from malware perspective. Such approach has deemed ineffective given the unique characteristics that distinguish this attack which necessitate the early discovery before encryption takes place. To this end, this paper puts forward an efficient and effective framework for building crypto-ransomware early detection models that protect users, whether individuals or organizations, of being victimized by such attack.

Keywords: Crypto-ransomware · Locker-ransomware · Malware · Bitcoin · Cybercurrency · Cryptography · Scareware · Early detection

1 Introduction

The technological advancement has been accompanied with many problems to information security, privacy, and integrity. Malware is one of the security issues that threatens computer system [1, 2]. The arm race between malware writers and anti-virus vendors has motivated both parties to reinforce their fortifications which led to producing different types of malware such as viruses, Trojans, worms, rootkits, spyware, to name a few. Enabled by the autonomous peer-to-peer Cybercurrency like bitcoin, ransomware is one of the malware categories that attracted many adversaries recently [3–6].

As its name implies, ransomware is a malware category that attacks user's files and personal data rendering them inaccessible [7–10] and demands a ransom in exchange of the captivated resources. The first occurrence of ransomware was on 1989 when a Trojan called AIDS has been released [11]. The emergence of ransomware has introduced a new type of attacks called Denial-of-Resources (DoR) attack [12–14] that

F. Saeed et al. (eds.), *Recent Trends in Information and Communication Technology*, Lecture Notes on Data Engineering and Communications Technologies 5, DOI 10.1007/978-3-319-59427-9_78

targets any user's related resources using system utilities such as cryptography [15]. Typically, ransomware utilizes infection vectors similar to traditional malware like email attachments, drive-by downloads, and exploitation kits.

Motivated by the great revenue, ransomware becomes a lucrative business that attracts many people to develop more sophisticated strains hard to detect and difficult to content [3, 9]. In 1996, [16] presented the idea of potential employment of cryptography against users. Later, this idea has been evolved into ransomware which locks user's personal files and demands a ransom so as to emancipate them [3, 7, 10, 11]. Moreover, the arm race between the defenders and adversaries yields the emergence of different and more sophisticated types of ransomware that employ advanced cryptography as well as mutation techniques such as polymorphism and metamorphism.

Based on the attack approach, ransomware is classified into two main categories, Locking-Ransomware and Crypto-Ransomware. Locking-Ransomware hijacks one or more services on victim's system [17] such as desktop, or input devices and keeps the user from accessing these resources [18, 19]. The infected system is left with a very limited capability that only allows the victim to do simple activities related to the payment process. On the other hand, Crypto-Ransomware renders user data inaccessible by employing the cryptography against user's documents and personal files. The crypto-ransomware is further classified into Private Key Cryptosystem (PrCR) which uses one key for encryption and decryption (symmetric cryptography), Public Key Cryptosystem (PuCR) which uses two different keys for encryption and decryption (asymmetric cryptography), Hyper Key Cryptosystem (HCR) which integrates between the symmetric and asymmetric encryption methods [20].

Unlike other types of malware, the irreversible nature of its attack necessitates the early detection of crypto-ransomware. Thus, the definition of the phase that precedes encryption plays a vital role to effectively detect crypto-ransomware as early as possible. Although there are several studies recently tackled the early prediction of crypto-ransomware [21], such approaches neither effective nor efficient for crypto-ransomware detection as they merely focus on tracking individual ad-hoc events instead of the entire behavior. Given the diversity of crypto-ransomware attack approach [22], the existence of these ad-hoc events depends on the malicious code's family. Consequently, such approach suffers the low detection rate and high false alarms. Meanwhile, current research in ransomware detection does not clearly define the pre-encryption phase, hence, unable to effectively detect the crypto-ransomware attack early. In addition, existing solutions are misuse-based that depend on predefined signatures, structural or behavioral, which are incapable of detecting novel, zero-day arracks. To address these issues, this paper puts forward a framework for building 0-day aware crypto-ransomware early detection models that can effectively detect this type of attack as early as before the encryption takes place. The rest of this paper is organized as follows. Related work is introduced in Sect. 2. In Sect. 3 we give a description of the methods used for building the internal components of the proposed framework. Section 4 discusses the methodology followed as well as framework components, namely preprocessing, features engineering, and detection. In Sect. 5, the resulted framework is discussed while Sect. 6 concludes this paper with suggestions for future work.

2 Related Work

Several solutions have been introduced regarding ransomware attacks [5, 9, 10, 12–14, 20, 23–30]. Some of these studies are preventive [23, 24, 31] that tries to proactively or reactively prevent the damage from being inflicted against user data while others [5, 7, 10, 20, 30] try to detect the attack whenever it takes place. Although preventive procedures; like regular backup; could help users survive ransomware attacks, such approach is not effective as most of the attacks tend to target naïve and unsophisticated users who normally do not follow these security measures [30]. Besides, these procedures are likely to be bypassed by advanced ransomware strains. There are, for instance, some ransomware types delete shadow copies [8, 20]. As such, preventive countermeasures are not sufficient to protect users against this kind of attacks.

To address crypto-ransomware early detection, several studies have been conducted which look for specific, individual and ad-hoc events as indicators for imminent crypto-ransomware attacks. Ahmadian, Shahriari [20] proposed monitoring C&C communications, DGA requests and other information exchanged between the malicious code and the remote server. Likewise, Heldroid proposed by Andronio, Zanero [10] utilizes the dynamic approach to detect the threatening text embedded in the payload of crypto-ransomware. Similarly, Cabaj, Gregorczyk [32] leveraged the http messages sequences and their respective content sizes so as to detect crypto-ransomware attacks. However, since these techniques depend on certain individual events, they alone are not sufficient evidence for the program maliciousness. Furthermore, these events do not occur in all types and thus, generate high false alarms.

On the other hand, instead of observing specific events, few studies [5, 30] focus on the files subject to crypto-ransomware attacks. They have proposed data-centric crypto-ransomware detection solutions. These solutions are built based on continually inspecting user-related documents to detect any abnormal changes in their entropy as an indicator of potential malicious activities. This enables the victim to respond to the attack as early as before a major damage happens. Although these techniques have achieved reasonable success, they still lack the genericity to detect zero-day attacks as well as the accuracy to improve the early detection performance while maintaining false alarms as low as possible. Furthermore, data-centric approach tolerates encrypting part of the files before the detection. Having explored the research pertaining crypto-ransomware early detection, we conclude that users still highly vulnerable to crypto-ransomware attacks and there is a need to improve current detection models or build new ones that suit this type of ransomware. To sum up, potential models should consider the unique characteristics of crypto-ransomware [5, 30] in order to effectively detect their attacks as early as before a major damage takes place.

3 The Methods

To build the framework, several methods are leveraged, i.e. FCM, TF-IDF data-centric detection, and anomaly detection. In this section, and due to space limitation, a brief description for some of these techniques is provided.

FCM: Frequency Centric Model is a semantic-based feature extraction method proposed by Das, Liu [21] which groups all API calls that deal with the same resource into one feature set then calculate the frequency of similar sets. This method facilitates discovering the repetitive behavior that differentiates the actions carried out by crypto-ransomware from benign programs.
Data-Centric Detection: This technique is utilized by several studies [5, 30] and focus on user data instead of crypto-ransomware process so as to detect the changes that might happen to the data in these files. To do so, several measurements are used such as entropy and similarity.
Anomaly Detection: This method is built upon a normal baseline and detects the deviation from the normal behavior of the system. In this way, such method is able to detect novel attacks, i.e. 0-day attacks. Although this method is very popular in several domains like intrusion detection and malware detection, it has not been utilized for crypto-ransomware detection yet.

4 The Proposed Framework

Given the irreversible nature of crypto-ransomware, it is necessary to detect it early before the encryption is carried out. To this end, the framework proposed here tends to clearly define the pre-encryption data space as the main defense step against crypto-ransomware attacks. However, defining this area is not an easy task given the ever-changing and diversity nature of crypto-ransomware attack approaches [22]. For instance, some families tend to discover the running environment of the underlying system and locate all the targeted files before encryption whereas others start the encryption immediately. To cope with such dynamicity, the framework proposed in this paper starts with the preprocessing module that defines and extracts the pre-encryption features to be used by subsequent components, i.e. features engineering and detection module. Figure 1 illustrates the components of proposed framework.

The methodology used for developing the proposed framework contains several steps and starts with building the query vector space [33] that is used as stopping criteria for the extraction algorithm. This vector contains the cryptography relevant APIs and functions ranked based on TF-IDF [34] weighting method. Once defined, the query vector is ported to the pre-encryption data extraction algorithm that employs sliding window technique [35] to iterate the CRW dataset.

In features engineering module, two types of features are generated, behavioral and data-centric. The former is constructed by applying n-gram technique [36, 37] on the normalized API calls in the pre-encryption subset while the latter is constructed by applying FCM [21] on API-Parameter pairs which calculate the frequency of the API sets that share same parameters. Once constructed, Information Gain is adopted for feature selection due to its superb performance in dimensionality reduction [38].

As SVM has shown promising classification capabilities in text categorization [39], it is employed in the third module of the framework for behavioral detection model. Because of the wide usage of One-Class SVM (OC-SVM) for anomaly detection [40–42], it is adopted for the second part of the module, i.e. the CRW anomaly detection model. If the sample under observation is classified as malicious by the

behavioral model, then it is the final decision, otherwise, it undergoes further examination by the anomaly model. If there is a deviation from the normal baseline, then malicious, or benign otherwise.

5 Results and Discussion

As depicted in Fig. 1, the proposed framework consists of three modules, pre-processing module, features engineering module, and detection module. Following are the details of each component.

5.1 Pre-processing Module

As mentioned in the previous section, the behavioral variations between attack approaches among different crypto-ransomware families render extracting pre-encryption data a challenging task. So, there is a need to define the pre-encryption phase clearly and accurately. Unlike fixed runtime window of 30 s proposed by Sgandurra, Muñoz-González [43], pre-processing module starts by locating the ever-changing region of interest (ROI) whereby encryption starts taking place. ROI plays a crucial role as a stopping threshold in data extraction algorithm. These thresholds are defined by constructing the term weighting model, i.e. TF-IDF, on sliding window level. Having defined the stopping threshold, the extraction algorithms uses it to build the pre-encryption dataset. This algorithm dynamically samples the trace file into smaller parts called 'Sliding Window' and sequentially looks into each window for the stopping threshold. The iteration continues until it encounters this threshold. This model is boosted by several well-known ransomware specific events such as crypto API calls, function and encryption key generation. On the contrary to Sgandurra, Muñoz-González [43], our technique is effectively able to track the location of encryption starting point in the trace file and hence, accurately define the pre-encryption phase and extract the pre-encryption data space which extremely contributes to developing effective crypto-ransomware early detection solutions.

5.2 Features Engineering Module

Having built the pre-encryption dataset, features construction is taken place. The feature set encompasses two types of features, data-centric and semantic. In this module, we put forward Enhanced Frequency Centric Model (EFCM) technique to construct dynamic data-centric feature set based on the FCM technique [21]. EFCM resembles FCM in that it puts APIs that work on the same resource into same groups. Additionally, EFCM puts all APIs that work on one user-related resource into a specific action. Unlike static data-centric feature set constructed by Scaife, Carter [30], the features engineering module generates dynamic data-centric features suitable for the behavioral detection. On the other hand, the semantic features are constructed using

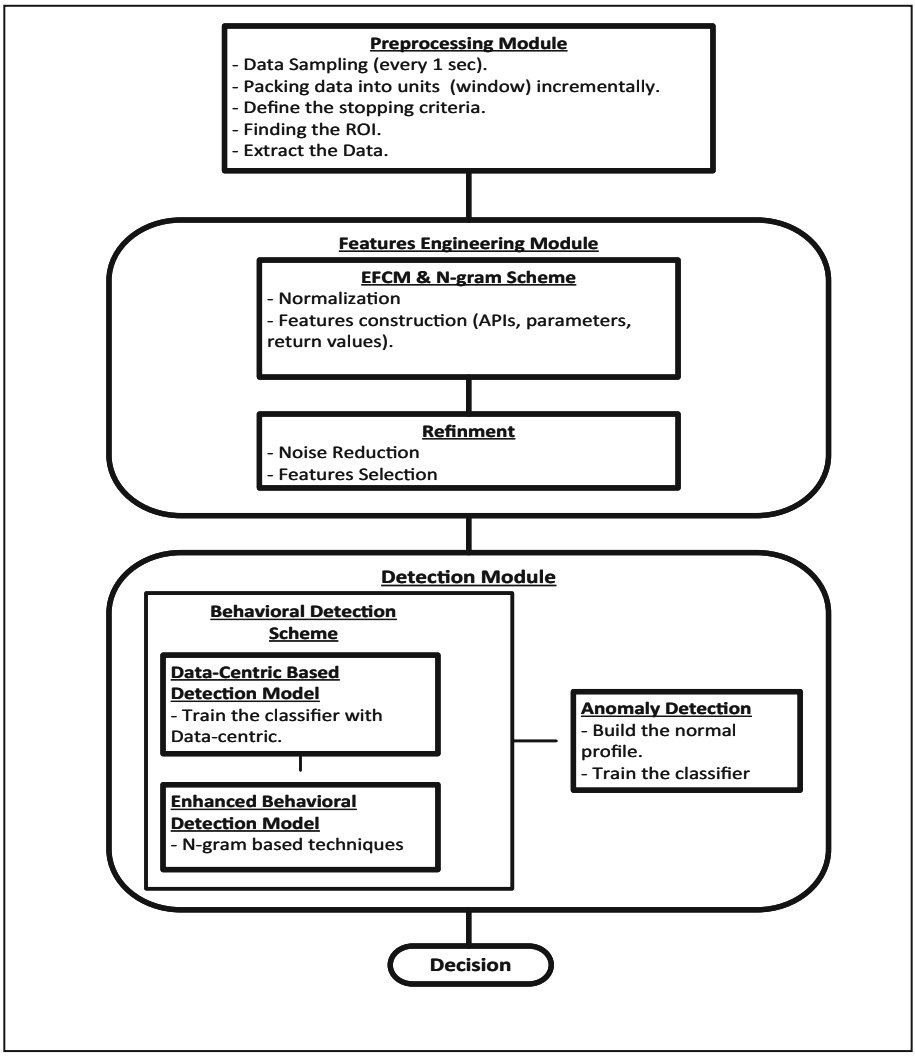

Fig. 1. The proposed 0-day aware early detection model.

n-gram technique that considers the relation between APIs. To accurately highlight the repeating actions of crypto-ransomware, APIs regularization is taken place before features construction. Moreover, regularization guarantees some sort of generalization by taking each API and changes it into its original root. Once constructed, the frequency of each action is calculated and compared against a threshold. Additionally, the frequency of crypto-ransomware critical APIs is calculated as well which helps to predict the imminent attack.

5.3 Detection Module

This module takes the extracted features as input and with the help of training set, it decides whether the sample is crypto-ransomware. To do that, detection module consists of two schemes, data-centric behavioral detection scheme built based on the dynamic data-centric features extracted in the previous module and enhanced by integrating the semantic features constructed using n-gram technique. Likewise, the second scheme is an anomaly-based detection that detects zero-day crypto-ransomware attacks. Moreover, the anomaly detection scheme builds the normal profile from the features extracted in the early stages of legitimate programs by leveraging one-class classifier approach. Figure 1 illustrates the process flow chart of the detection module.

6 Conclusion and Future Work

In this paper, we proposed an early detection framework for crypto-ransomware. This framework is one of few studies that employ machine learning techniques to detect crypto-ransomware. The proposed framework encompasses three modules, pre-processing module, features engineering module, and detection module. Sliding window technique was adopted for pre-encryption data acquisition from within the malicious code's runtime data. Based on these data, the features are constructed and fed into the detection module. To improve the detection accuracy, we proposed an adaptive anomaly detection that is able to cope with the dynamicity nature of the current systems and regularly updates the normal profile. For detection scheme to cope with the dynamicity of crypto-ransomware attack approach, we are currently working on further enhancement of the detection scheme by incorporating the ensemble classification principles with the incremental learning to produce an effective and efficient crypto-ransomware early detection solution.

References

1. Xue, L., Sun, G.: Design and implementation of a malware detection system based on network behavior. Secur. Commun. Netw. **8**(3), 459–470 (2015)
2. Naval, S., et al.: Employing program semantics for malware detection. IEEE Trans. Inf. Forensics Secur. **10**(12), 2591–2604 (2015)
3. Everett, C.: Ransomware: to pay or not to pay? Comput. Fraud Secur. **2016**(4), 8–12 (2016)
4. Spagnuolo, M., Maggi, F., Zanero, S.: BitIodine: extracting intelligence from the bitcoin network. In: Safavi-Naini, R., Christin, N. (eds.) 18th International Conference on Financial Cryptography and Data Security, FC 2014, pp. 457–468. Springer, Heidelberg (2014)
5. Kharraz, A., Arshad, S., Mulliner, C., Robertson, W., Kirda, E.: UNVEIL: a large-scale, automated approach to detecting ransomware. Paper presented at the 25th USENIX Security Symposium (USENIX Security 16), pp. 757–772 (2016)
6. Kharraz, A., et al.: Cutting the gordian knot: a look under the hood of ransomware attacks. In: Maggi, F., Almgren, M., Gulisano, V. (eds.) 12th International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, DIMVA 2015, pp. 3–24. Springer, Heidelberg (2015)

7. Song, S., Kim, B., Lee, S.: The effective ransomware prevention technique using process monitoring on android platform. Mobile Inf. Syst. **2016**, 1–8 (2016)
8. Mercaldo, F., et al.: Ransomware steals your phone. formal methods rescue it. In: Albert, E., Lanese, I. (eds.) Formal Techniques for Distributed Objects, Components, and Systems: 36th IFIP WG 6.1 International Conference, FORTE 2016, Held as Part of the 11th International Federated Conference on Distributed Computing Techniques, DisCoTec 2016, Heraklion, Crete, Greece, 6–9 June 2016, Proceedings, pp. 212–221. Springer, Cham (2016)
9. Yang, T., et al.: Automated Detection and Analysis for Android Ransomware, pp. 1338–1343 (2015)
10. Andronio, N., Zanero, S., Maggi, F.: HELDROID: dissecting and detecting mobile ransomware. In: Bos, H., Blanc, G., Monrose, F. (eds.) 18th International Symposium on Research in Attacks, Intrusions, and Defenses, RAID 2015, pp. 382–404. Springer, Heidelberg (2015)
11. Sittig, D.F., Singh, H.: A socio-technical approach to preventing, mitigating, and recovering from ransomware attacks. Appl. Clin. Inf. **7**(2), 624–632 (2016)
12. Young, A.L.: Cryptoviral extortion using Microsoft's crypto API. Int. J. Inf. Secur. **5**(2), 67–76 (2006)
13. Young, A.L.: Building a cryptovirus using Microsoft's cryptographic API. In: Zhou, J. et al. (eds.) Information Security: 8th International Conference, ISC 2005, Singapore, 20–23 September 2005, Proceedings, pp. 389–401. Springer, Heidelberg (2005)
14. Kumar, S.M., Kumar, M.R.: Cryptoviral extortion: a virus based approach. Int. J. Comput. Trends Technol. (IJCTT) **4**(5), 1149–1153 (2013)
15. Ahmadian, M.M., Shahriari, H.R., Ghaffarian, S.M.: Connection-monitor and connection-breaker: a novel approach for prevention and detection of high survivable ransomwares. In: 2015 12th International Iranian Society of Cryptology Conference on Information Security and Cryptology (ISCISC). IEEE (2015)
16. Young, A., Yung, M.: Cryptovirology: extortion-based security threats and countermeasures. In: Proceedings, 1996 IEEE Symposium on Security and Privacy. IEEE (1996)
17. Pathak, P., Nanded, Y.M.: A dangerous trend of cybercrime: ransomware growing challenge. Int. J. Adv. Res. Comput. Eng. Technol. (IJARCET) 5(2), 371–373 (2016)
18. Bhardwaj, A., et al.: Ransomware digital extortion: a rising new age threat. Indian J. Sci. Technol. **9**, 14 (2016)
19. Savage, K., Coogan, P., Lau, H.: The evolution of ransomware. In: Security Response. Symantec Corporation
20. Ahmadian, M.M., Shahriari, H.R., Ghaffarian, S.M.: Connection-monitor and connection-breaker: a novel approach for prevention and detection of high survivable ransomwares. In: 12th International ISC Conference on Information Security and Cryptology, ISCISC 2015. Institute of Electrical and Electronics Engineers Inc. (2015)
21. Das, S., et al.: Semantics-based online malware detection: towards efficient real-time protection against malware. IEEE Trans. Inf. Forensics Secur. **11**(2), 289–302 (2016)
22. Ahmadian, M.M., Shahriari. H.R.: 2entFOX: a framework for high survivable ransomwares detection. In: 2016 13th International Iranian Society of Cryptology Conference on Information Security and Cryptology (ISCISC) (2016)
23. Luo, X., Liao, Q.: Ransomware: a new cyber hijacking threat to enterprises. In: Handbook of Research on Information Security and Assurance, pp. 1–6. IGI Global (2008)
24. Bridges, L.: The changing face of malware. Netw. Secur. **2008**(1), 17–20 (2008)
25. Gazet, A.: Comparative analysis of various ransomware virii. J. Comput. Virol. **6**(1), 77–90 (2010)
26. Luo, X., Liao, Q.: Awareness education as the key to ransomware prevention. Inf. Syst. Secur. **16**(4), 195–202 (2007)

27. Kim, D., Soh, W., Kim, S.: Design of quantification model for prevent of cryptolocker. Indian J. Sci. Technol. **8**(19) (2015)
28. Cabaj, K., et al.: Network activity analysis of CryptoWall ransomware. Przeglad Elektrotechniczny **91**(11), 201–204 (2015)
29. Choi, K., Scott, T., LeClair, D.: Ransomware against police: diagnosis of risk factors via application of cyber-routine activities theory. Int. J. Forensic Sci. Pathol. **4**(7), 253–258 (2016)
30. Scaife, N., et al.: CryptoLock (and drop it): stopping ransomware attacks on user data (2016)
31. Mustaca, S.: Are your IT professionals prepared for the challenges to come? Comput. Fraud Secur. **2014**(3), 18–20 (2014)
32. Cabaj, K., Gregorczyk, M., Mazurczyk, W.: Software-defined networking-based crypto ransomware detection using HTTP traffic characteristics. arXiv preprint arXiv:1611.08294 (2016)
33. Singhal, A.: Modern information retrieval: a brief overview. IEEE Data Eng. Bull. **24**(4), 35–43 (2001)
34. Paltoglou, G., Thelwall, M.: A study of information retrieval weighting schemes for sentiment analysis. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics (2010)
35. Alam, S., et al.: Sliding window and control flow weight for metamorphic malware detection. J. Comput. Virol. Hacking Tech. **11**(2), 75–88 (2015)
36. O'Kane, P., Sezer, S., McLaughlin, K.: N-gram density based malware detection. In: 2014 World Symposium on Computer Applications and Research, WSCAR 2014 (2014)
37. Santhosh, S., Ranveer, S.: N-gram based malicious code detection using support vector machine learning approach. In: 4th International Conference on Advances in Recent Technologies in Communication and Computing, ARTCom 2012. Institution of Engineering and Technology (2012)
38. Yang, Y., Pedersen, J.O.: A comparative study on feature selection in text categorization. In: ICML (1997)
39. Joachims, T.: Text categorization with support vector machines: learning with many relevant features. In: Nédellec, C., Rouveirol, C. (eds.) 10th European Conference on Machine Learning Chemnitz Machine Learning: ECML 1998, Germany, 21–23 April 1998 Proceedings, pp. 137–142. Springer, Heidelberg (1998)
40. Zhang, M., Xu, B.Y., Wang, D.X.: An anomaly detection model for network intrusions using one-class SVM and scaling strategy. In: Guo, S., et al. (eds.) Collaborative Computing: Networking, Applications, and Worksharing, Collaboratecom 2015, pp. 267–278. Springer, New York (2016)
41. Shang, W.L., et al.: Intrusion detection algorithm based on OCSVM in industrial control system. Secur. Commun. Netw. **9**(10), 1040–1049 (2016)
42. Zhang, M., Xu, B., Gong, J.: An anomaly detection model based on one-class SVM to detect network intrusions. In: 11th International Conference on Mobile Ad-Hoc and Sensor Networks, MSN 2015. Institute of Electrical and Electronics Engineers Inc. (2015)
43. Sgandurra, D., et al.: Automated dynamic analysis of ransomware: benefits, limitations and use for detection. arXiv preprint arXiv:1609.03020 (2016)