Anomaly Detection Technology Using Potential Difference Displacement Detection of Data Bus

Hye Lim Jeong[1], Sung Kyu Ahn[1], Sung Hoon Baek[2], and Ki-Woong Park[1*]
[1]Department of Information Security, Sejong University, Seoul, Korea
{hyello13, yiimfn}@gmail.com, woongbak@sejong.ac.kr
[2]Department of Computer System Engineering, Jungwon University, Chungbuk 28024, Korea
shbaek@jwu.ac.kr

Abstract

Ransomware attacks are constantly changing, and the damage they cause has increased. Detection and recovery researches and technologies that respond to variants of ransomware to prevent damage have high safety standards and sacrifices have to be made for the effective application of system re-sources. To overcome such problems,We proposed the determination of entropy using a measurement of the changes in voltages of capacitor microcurrent for distinguishing of the abnormal data accordingly. We hypothesized that both the unencrypted file and the encrypted file will affect voltage change that occur with the data from the I/O channels of the storage. The hypothesis was tested through an experiment that was conducted by implementing the storage and a capacitor circuit on a software level. Experimental results showed that the electrical properties could be detected by the capacitor, and the abnormal data could be detected by entropy calculations in the I/O channels of the storage.

Keywords: Ransomware Detection, Entropy, Computation-less, Abnormal Data Flow

1 Introduction

Ransomware is a type of malicious software designed to perform encryption on files stored on a victim's personal computer (PC) and demands a ransom payment before the victim can gain access to their files again. Ransomware attacks are constantly changing, and the damage they cause has increased. Detection and recovery researches and technologies that respond to variants of ransomware to prevent damage have high safety standards [1, 4, 14, 3, 8], and sacrifices have to be made for the effective application of system resources. Some variants of ransomware introduce a process that prevents their detection and recovery of the defense solution in the development phase. Thus, there is a great challenge in constantly supporting the existing detection and recovery solutions as ransomware is evolving into a dangerous and complicated attack.

To overcome such problems, this study, proposes a method of using the electrical property of capacitor microcurrent [12] to overcome the aforementioned challenges. This method is lighter than the solution suggested by previous research, and as it detects abnormal data flow from the I/O channels of the storage that attempts to achieve the goals of ransomware by continuous encryption calculations. When a PC is infected with ransomware, the normal files stored in the storage are converted into encrypted files, which is referred to as abnormal data flow in this paper. According to Shannon's information entropy theory that states that a normal unencrypted file and an encrypted file can be distinguished from each other by means of an entropy result value [11, 10, 9, 6, 2] we hypothesized that both the unencrypted file and the

Journal of Internet Services and Information Security (JISIS), volume: 9, number: 4 (November, 2019), pp. 68-77
*Corresponding author: Dept. of Computer and Information Security, Sejong University, Neungdonr-Ro 209, Gwangjin-Gu, Seoul, (05006), Tel: +82-2-6935-2453

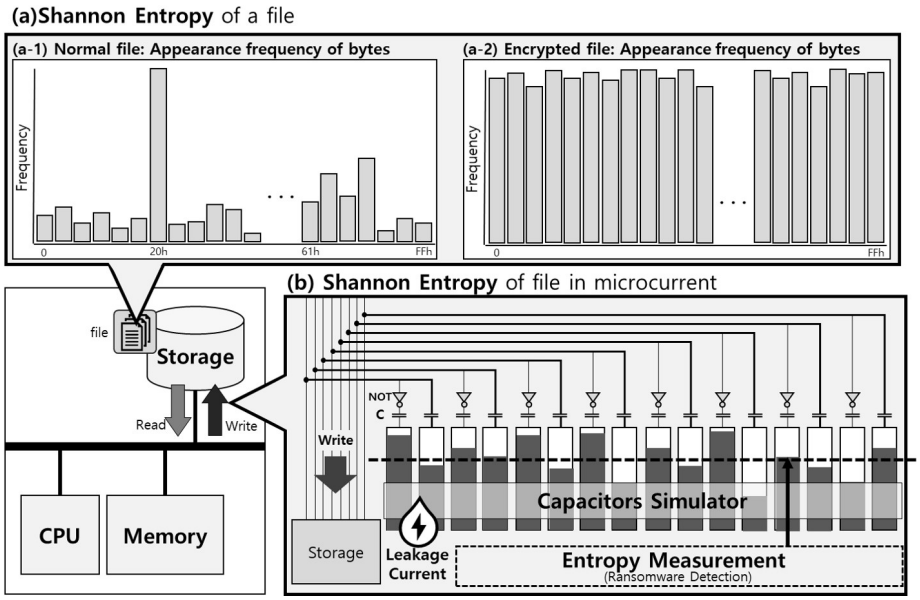

Figure 1: Overview of main contribution using Shannon entropy in microcurrent.

encrypted file will affect the voltage change that occur with the data flow from the I/O channels of the storage. We proposed the measurement of the changes in these voltages, determination of entropy, and distinguishing of the abnormal data flow accordingly. The hypothesis was tested through an experiment that was conducted by implementing the storage and a capacitor circuit on a software level. Experimental results showed that the electrical properties could be detected by the capacitor, and the abnormal data flow could be detected by entropy calculations in the I/O channels of the storage. When an abnormal data flow was detected, it determined whether a stored file was encrypted to assess if a PC was infected with ransomware.

The details of this experiment are illustrated in Figure 1. In this paper, eight data channels were assumed for entropy measurements and experimental verification. In Figure 1, (a-1) shows the entropy measurement value of the byte level of the normal file. Similarly, (a-2) shows the entropy measurement value of the byte level of the encrypted file. According to (a-1) and (a-2), the distinction between the unencrypted and encrypted files is possible using entropy measurements at the byte level. Based on this fact, we constructed a capacitor circuit module for data input and output in the unit of bit. In this experimental environment, eight capacitors were used to measure the voltage in each circuit of eight data channels. Figure 1 illustrates the capacitors (b) charged through the voltage flowing into the data bus circuit for the storage. As shown in Figure 1, abnormal data flow can be distinguished by calculating the amount of entropy charge of eight capacitors, and seeing different entropy values for each data. (The reason why there are 16 capacitors in Figure 1 is explained in Section 3.1).
In this study, we proposed a method to detect abnormal data flow by distinguishing the difference in entropy values between unencrypted files and encrypted files by monitoring the status monitoring of the capacitors. Our preliminary experiment described in Section 4 shows that a capacitor-based circuit can be constructed and used as an actual physical circuit to measure the abnormal data flow. Therefore, in this study, a detection scheme in a way of computationless has been proposed for the detection of abnormal data flow through a hardware-based capacitor module without using computation resources. In addition, we expected the ability of the proposed hardware-based detection method to be better at de-

tecting ransomware in comparison to the software-based detecting technology, which is avoided by the anti-ransomware technology.
Section 2 describes the related works of ransomware detection. This paper describes in detail the technique proposed for ransomware detection in Section 3, and in Section 4, we describe the experimental results. Section 5 is concludes this paper.

2 Related Work

This Section discusses three research studies related to the research presented in this paper and explains the contribution of this paper.
The first research [4] is a method to protect and recover data by ransomware in Windows OS environment, which can implement an add-on driver for the protection and recovery of data from ransomware attacks. It analyzes the I/O file system requests made by normal applications and detects ransomware by recognizing I/O requests generated from ransomware operations. The advantage of the driver system is that by protecting the data by ransomware, it can generate a copy of a specific file and use the rollback function to recover the original if the file has been tampered with by ransomware during an attack. The method proposed in this study has up to 3.8 times run-time overhead during copy creation for recovery, and up to 14.73 GB of additional space is required from the storage point of view in comparison to the Windows OS system.
The second research [14] proposes a solution for detecting ransomware and recovering files encrypted by ransomware. To improve the safety of the existing files or develop an application-based ransomware detection technology and to overcome the overhead, we proposed a ransomware detection method that operates using SSD internal firmware. This proposed method distinguishes ransomware by measuring the cumulative number of overwritten operations that are characteristic of ransomware. Moreover, based on the behavior of the GC inside the SSD erasing deleted or unused memory space, it tracks the previous version of data and delays deletion until the ransomware detection algorithm confirms that the data to be overwritten has not been attacked by ransomware. If it is determined that it is infected with ransomware, SSD Insider will use the data before it was overwritten to recover the data. This study assesses the detection process in the SSD firmware unit to overcome the safety of solutions such as file monitoring and driver, firmware, and kernel modification for conventional ransomware detection. The advantage of the method is that it is able to detect and recover data; however, its disadvantage is that it changes the existing SSD firmware by applying the firmware to the SSD.
In the third research, [3] if you use a PC synchronized with a backup system such as a cloud service, files infected with ransomware will be synchronized with the files on the backup system. This makes it challenging to restore the infected files. In this study , we proposed the restoration of infected backed up files by measuring the characteristics of the encrypted files using entropy and machine learning techniques. We also proposed a method for detecting files infected with ransomware using a machine learning model that measures the file entropy of the backup system. However, some image files and compressed files have high entropy; hence, there is a limit to detecting files infected using ransomware only by the entropy reference value. In this study , to overcome this limitation, we used an entropy-based machine learning based on various file formats to detect infected files. As a result of the experiment, the infected files were classified into high grade, and the false detection rate and false detection rate were very low. Files infected by ransomware were not synchronized to the backup system. Hence, even if your system was infected with ransomware, he original files could be recovered by restoring the files on the backup system.
However, by proceeding in synchronization with the client software ransomware detection module for each user, there is a disadvantage that an attack can be avoided by a software evasion technique, which

is one of the disadvantages of the existing ransomware.

3 Experimental Design

We expected that this experiment will detect abnormal data flows from the I/O channels of the storage. In this Section, we experimentally detect anomalies in files such as ransomware through the features of the Shannon entropy based on the hardware, a capacitor module. This section describes the experiments in detail to prove our proposal. The experiments were conducted in one structure as shown in Figure 1-(b). This structure is a model designed to validate the proposed abnormal data flow detection. The experimental setup included a capacitor simulator and related entropy measurements. As entropy measurement is modules for proving the operation of capacitor simulator, entropy measurement does not belong to structure in real world. The entropy measurement of the experiment includes the development of a hardware-type abnormal data flow detection technology by monitoring the charge of the capacitors. The capacitor was placed in the communication of data bus such that voltage in there was used to charge it.

3.1 Capacitor Simulator

A capacitor is a type of hardware element for voltage supplementation to the data input / output circuit and charges or supplies voltage to the circuit. It is also discharged through the leakage of current [12][5] as opposed to the charge. This allows it to check the amount of voltage flowing on circuit. The charge and discharge functions of the capacitor were used as follows: when the voltage comes from the capacitor 1, use the charge formula,

$$V_t = V_{pre} + V_{now}(1 - e^{\frac{-t}{RC}}) \tag{1}$$

When the voltage comes from the capacitor 0, use the discharge formula,.

$$V_t = V_{pre}(e^{\frac{-t}{RC}}) \tag{2}$$

In the formula above, V_t is the result when voltage is introduced, V_{pre} is the value of the previous voltage, and V_{now} is the value of the current voltage. R is the resistance of the resistor, C is the capacitance of the capacitor, and t is the time constant. Charge and discharge formula can be used to determine the check the amount of voltage in the capacitor. The capacitor simulator of this experiment was implemented using this formula. Therefore, the capacitor connected to the input/output circuit of the storage has a different charge voltage for each circuit due to a voltage generated randomly during data writing. We used these features to detect the abnormal data flow. As the amount of voltage generated from the input / output circuit of the storage becomes constant during the abnormal data flow, the charge of the capacitor becomes constant for each circuit, thereby confirming the encrypted file. The simulation was programmed by a software to run the experiment of the proposed method [7, 13]. The experiment was used as a pair consisting of a common capacitor and a capacitor with a inverter (a NOT gate). This allowed for the individual values to be reversed, which improved the distinction of entropy. The proposed method was tested through a software using the aforementioned equations to determine the abnormal data; the information must be checked to ensure that each capacitor has the same amount of voltage. Operations for measuring entropy are described in Section 3.2.

3.2 Entropy Measurement

Entropy measurement is modules for proving the operation of capacitor microcurrent measures, so that do not belong to structure in real SSD. In the process of writing data with storage, the charge and

discharge formula is calculated for each capacitor connected to each data bus circuit. As shown in Figure 2, (a) are capacitors charged through the voltage flowing in the circuit in a general file. (b) are capacitors charged through the voltage flowing in the circuit in the encrypted file. The following formula was used to check whether the charge voltage values of the 16 capacitors are regularly accumulated. The closer the result of the standard deviation is to zero, the more the numbers are not different between the samples.

$$\sqrt{\frac{\sum(x_n - \bar{x_n^2})}{n}} \tag{3}$$

In the standard deviation formula, x is the voltage value of capacitor. In the computationless method we proposed in this paper, had 16 samples in total because there was pair consisting of a capacitors and an inverter capacitor that were connected is connected to each data circuit. The x value of the 16 capacitors was calculated to the sigma (ς) to derive the difference value from each sample and the mean. If the difference value is closer to 0, the voltage on the data bus circuit is constant. We can use the method described above to detect abnormal data flow. The difference in values, which is the result of deviation, if it is close to 0, the entropy of the capacitor charging voltage is high, and thus, it may be determined that it is abnormal data flow.

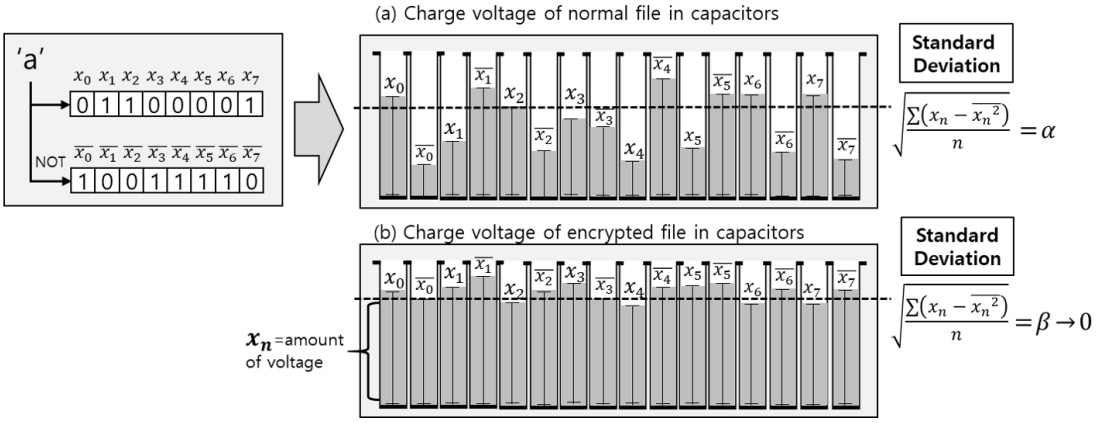

Figure 2: Comparison of Standard Deviation with Capacitor Charge Voltage Difference

4 Experiment and Result

Figure 3 shows the structure of the process to prove the process of detecting abnormal data flow by comparing the charge voltage of each capacitor. The structure of the process consisted of five modules i.e. the input translation module to perform input data conversion, channel queue module that functions as a data bus, capacitor simulator module for performing capacitor value calculation according to each data, states memory module that performs simultaneous measurement of capacitor's voltage value at unit time, entropy measurement module for computing a standard deviation. The two capacitors per I/O channel were used to detect input and output of the encrypted files. This corresponds to the leakage measurement presented in Figure 4. The leakage measurement consists of a capacitor x to which the bitstream generated in the data communication channel is input and a capacitor y to which the inverted bit is input through an inverter (NOT gate). The experiment consisted of 8 sets of leakage measurement. Therefore,

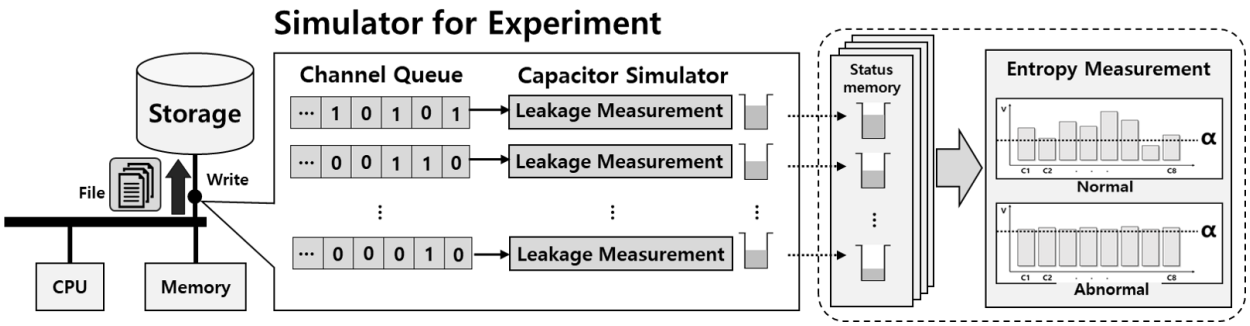

Figure 3: The operation flow in the experiment

the system consists of 16 capacitors, capacitors $x_1 \sim x_8$ and capacitors $y_1 \sim y_8$. In this structure, the unencrypted normal file had low entropy, so that the bitstream enters the capacitor y through the inverter during the input process. Thus, the result is that the charge amounts of the capacitors x and y were inversely proportional to each other. If the charge amount of the capacitors x and y were measured at each time set in the states memory, the standard deviation of the capacitors $x_1 \sim x_8$ and $y_1 \sim y_8$ increased. However, encrypted file had high entropy, and the charge voltage amount of capacitors x and y varies similarly. At this time, when the charge voltage amount of capacitor $x_1 \sim x_8$ and $y_1 \sim y_8$ are compared, the result of standard deviation is lowered. The input translation module received unencrypted general data or encrypted experimental data, converted it into bit units, and delivered them to the channel queue module, which is a data bus channel function. Inside the capacitor simulator, there was a leakage measurement module that calculated the values stored in the channel queue. The capacitor simulator module graphical values were comparable to the actual hardware capacitors through the leakage measurement result. The states memory module collected the values of the capacitors that changed as the capacitor value was continuously processed in the capacitor simulator module. The entropy measurement module calculated the standard deviation of the capacitor values stored by the timer and converted them to a fixed unit. The calculated value, showed that the flow of the input data was encrypted, that is, the operation to detect the encrypted data was active. In the experiment, the circuit voltage was $50V$, the capacitor was $2mF$, the register was $100k\Omega$, and the bit operation of the capacitor was performed every $50ms$.

4.1 Result

Figure 5 is The results of measuring the value of capacitors x, y in $20ms$ unit in states memory when inputting encrypted file and unencrypted file in the experiment are presented in Figure 5. For the dataset used in this study, a '.txt' of about $42MB$ was used and a '.docx' of about $18KB$. In Figure 5, Figure 5-(a) is graph of Frequency appearance of byte, and Figure 5-(b) is graph of capacitor voltage to byte and Figure 5-(c) is graph of standard deviation of capacitors in chronological order. The experiment was conducted on the general file of the '.txt' file, the encrypted file and the normal file of the '.docx' file, and the encrypted file. In Figure 5-(a), the frequency of occurrence of characters in each file was validated. The results presented in Figure 5-(a, b), different from the values of each capacitor in Figure 5-(b). The results show that the amount of voltage that entered the circuit was different. The measured standard deviation from the experiment in $20ms$ up to $13000ms$ in chronological order are given in Figure 5(c) including the result over time by calculating the standard deviation of the values of the capacitors in Figure 5-(b). The closer the result of the standard deviation was to zero, the more constant the charge voltage of the

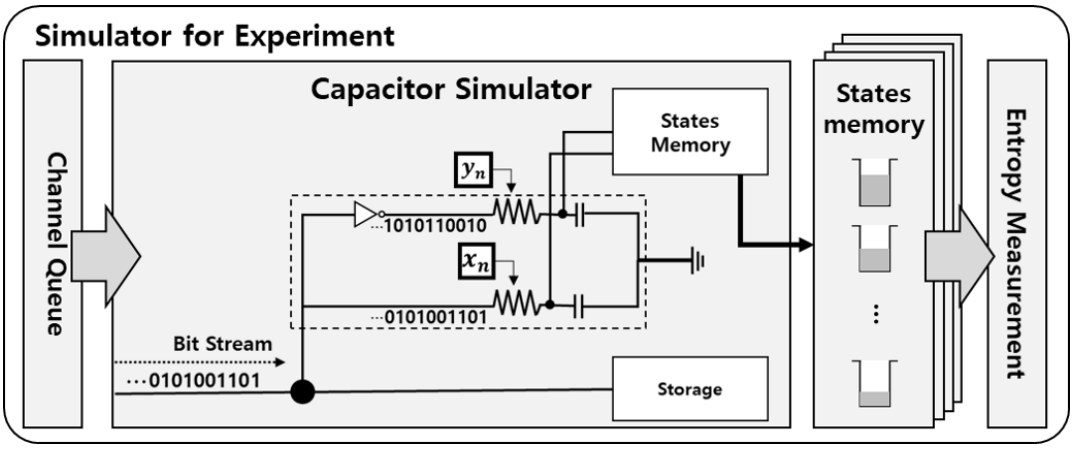

Figure 4: Describe the software structure of capacitor simulator as hardware method

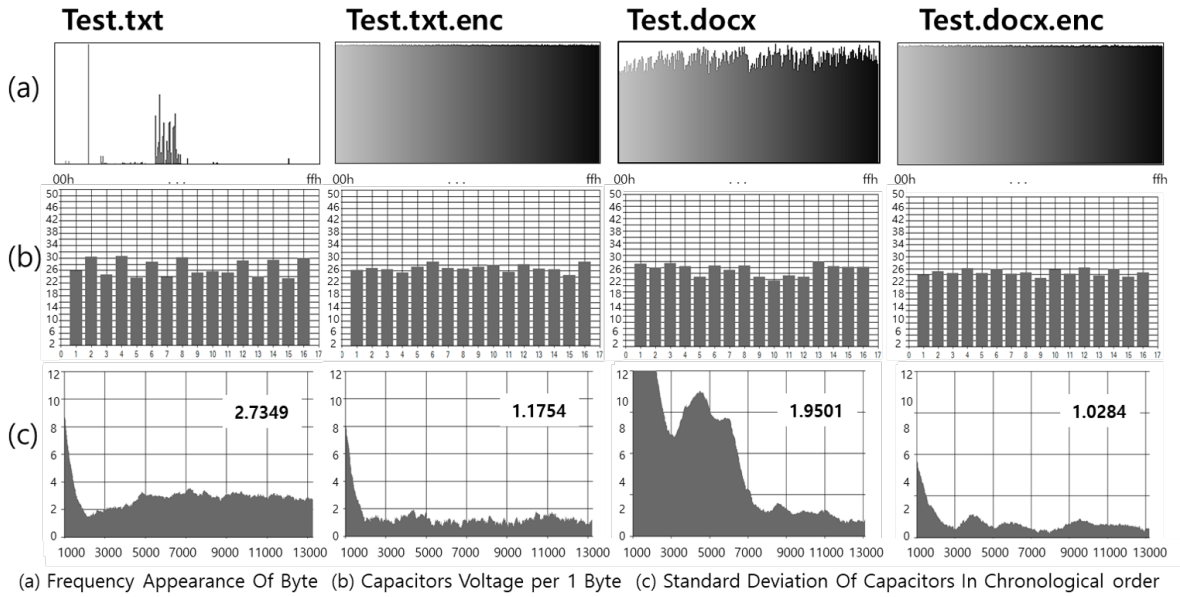

Figure 5: Unencrypted and Encrypted File Entropy Experiment Results

capacitors in Figure 5-(b). When the results of Figure 5-(c) were confirmed as shown in Figure 5-(a), the files in which the frequency of appearance of the characters in Figure 5-(a) were constant showed near zero results over time in Figure 5-(c).

Through this calculation process, the results showed high entropy values for encrypted file and low entropy value of unencrypted general file. This demonstrates that the Shannon entropy characteristic of the encrypted file were measurable through the charging and leakage current characteristics of the capacitor, which is a hardware module.. In addition, it is expected that because this approach is hardware-based, it will be able to detect ransomware relatively better than the software-based detecting technology which is avoided by anti-ransomware technology. Our future research work will focus on real storage devices such as SSDs and the construction of a safe storage environment that does not consume resources. The

final goal of this study is to warn and block users of devices from abnormal behavior in storage devices in a computationless environment on various devices.

5 Conclusion and Future Work

In this paper, we proposed a method for detecting abnormal data flow in storage files using capacitors. Abnormal data flow was detected during file storage by calculating the entropy of the charging information of the capacitors connected to the I/O channel. In addition, the process can be performed as a separate computing module, without consuming resources of the PC. The proposed method was validated through a simulation experiment using software. The experiment for proof uses the capacitor charge of the virtual circuit algorithm to construct the simulation environment. The results showed that the proposed method was able to distinguish between encrypted and unencrypted files. Our further research aims to experiment proposed this paper in real storage devices such as SSD.

Acknowledgments

This work was supported by the Institute for Information Communications Technology Promotion (IITP) of the Korea government (MSIT) [Grant No. 2019-0-00426, Grant No. 2018-0-00420] and by the National Research Foundation of Korea (NRF) [Grant No. NRF-2017R1C1B2003957].

References

[1] S. Baek, Y. Jung, A. Mohaisen, S. Lee, and D. Nyang. Ssd-insider: Internal defense of solid-state drive against ransomware with perfect data recovery. In *Proc. of the 2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS'18), Vienna, Austria*, pages 875–884. IEEE, July 2018.

[2] P. Bereziński, B. Jasiul, and M. Szpyrka. An entropy-based network anomaly detection method. *Entropy*, 17(4):2367–2408, April 2015.

[3] R. Brewer. Ransomware attacks: detection, prevention and cure. *Network Security*, 2016(9):5–9, September 2016.

[4] A. Continella, A. Guagnelli, G. Zingaro, G. De Pasquale, A. Barenghi, S. Zanero, and F. Maggi. Shieldfs: a self-healing, ransomware-aware filesystem. In *Proc. of the 32nd Annual Conference on Computer Security Applications (ACSAC'16), Los Angeles, California, USA*, pages 336–347. ACM, December 2016.

[5] X. Guo and X. Jia. Hardware-based cascaded topology and modulation strategy with leakage current reduction for transformerless pv systems. *IEEE Transactions on Industrial Electronics*, 63(12):7823–7832, September 2016.

[6] G. Jeong, E. Choo, J. Lee, M. Bat-Erdene, and H. Lee. Generic unpacking using entropy analysis. In *Proc. of the 2010 5th International Conference on Malicious and Unwanted Software (MALWARE'10), Nancy, Lorraine, France*, pages 98–105. IEEE, October 2010.

[7] Y. Kim, B. Tauras, A. Gupta, and B. Urgaonkar. Flashsim: A simulator for nand flash-based solid-state drives. In *Proc. of the 2009 First International Conference on Advances in System Simulation (SIMUL'09), Porto, Portugal*, pages 125–131. IEEE, September 2009.

[8] K. Lee, S.-Y. Lee, and K. Yim. Machine learning based file entropy analysis for ransomware detection in backup systems. *IEEE Access*, 7:110205–110215, July 2019.

[9] R. Lyda and J. Hamrock. Using entropy analysis to find encrypted and packed malware. *IEEE Security & Privacy*, 5(2):40–45, March 2007.

[10] C. B. Paul. Entropy based file type identification and partitioning. `https://calhoun.nps.edu/handle/10945/55513` [Online; Accessed on September 01, 2019], June 2017.

[11] C. E. Shannon. A mathematical theory of communication. *Bell system technical journal*, 27(3):379–423, July 1948.

[12] Y. Tang, W. Yao, P. C. Loh, and F. Blaabjerg. Highly reliable transformerless photovoltaic inverters with leakage current and pulsating power elimination. *IEEE Transactions on Industrial Electronics*, 63(2):1016–1026, September 2015.

[13] A. Tavakkol, J. Gómez-Luna, M. Sadrosadati, S. Ghose, and O. Mutlu. Mqsim: A framework for enabling realistic studies of modern multi-queue SSD devices. In *Proc. of the 16th USENIX Conference on File and Storage Technologies (FAST'18), Oakland, California, USA*, pages 49–66. USENIX Association, February 2018.

[14] M. Weckstén, J. Frick, A. Sjöström, and E. Järpe. A novel method for recovery from crypto ransomware infections. In *Proc. of the 2016 2nd IEEE International Conference on Computer and Communications (ICCC'16), Chengdu, China*, pages 1354–1358. IEEE, October 2016.

Author Biography

Hye-Lim Jeong received the B.S. degree in the department of information security from Daejeon University in 2015, and the M.S. degree in the department of information security from Daejeon University 2017. She is a Ph.D. Student of Sejong University. Her research interests include system security and secure storage system.

Sung-Kyu Ahn received the B.S. degree in the department of information security from Daejeon University in 2015, and the MS degrees in the department of information security from Daejeon University 2017. He is a Ph.D. Student of Sejong University. His research interests embedded system security and secure storage system.

Sung Hoon Baek received the B.S. degree in electronics engineering from Kyungpook National University, Korea, in 1997, the M.S. degree in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST) in 1999, the Ph.D. degree in electrical engineering from KAIST in 2008. He worked for Electronics Telecommunication Research Institute (ETRI) as an R&D staff from 1999 to 2005 and for Samsung Electronics as a senior R&D staff from 2008 to 2011. He has been an assistant professor in the department of computer system engineering at Jungwon University since 2011. His research interests include storage system, operating system, and parallel processing.

Ki-Woong Park received the B.S. degree in computer science from Yonsei University, South Korea, in 2005, the M.S. degree in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST) in 2007, and the Ph.D. degree in electrical engineering from KAIST in 2012. He received a 2009–2010 Microsoft Graduate Research Fellowship. He worked for National Security Research Institute as a senior researcher. He has been a professor in the department of computer and information security at Sejong University. His research interests include security issues for cloud and mobile computing systems as well as the actual system implementation and subsequent evaluation in a real computing system.