Computer Science and Information Technologies
Vol. 6, No. 3, November 2025, pp. 283~293
ISSN: 2722-3221, DOI: 10.11591/csit.v6i3.p283-293

Characteristics ransomware stop/djvu remk and erqw variants with static-dinamic analysis

Dodon Turianto Nugrahadi[1], Friska Abadi[1], Rudy Herteno[1], Muliadi[1], Muhammad Alkaff[2,3], Muhammad Alvin Alfando[1]
[1]Department of Computer Science, Faculty of Science, Lambung Mangkurat University, Banjarbaru, Indonesia
[2]Department of Information Technology, Faculty of Engineering, Lambung Mangkurat University, Banjarmasin, Indonesia
[3]Department of Computer Science, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia

Article Info

Article history:

Received Jun 6, 2024
Revised Jun 9, 2025
Accepted Jun 13, 2025

Keywords:

Dynamic analysis
Ransomware
Static analysis
STOP/DJVU erqw
STOP/DJVU remk

ABSTRACT

Ransomware has developed into various new variants every year. One type of ransomware is STOP/DJVU, containing more than 240+ variants. This research to determine changes in differences characteristics and impact between ransomware variants STOP/DJVU remk, which is a variant from 2020, and the erqw variant from 2023, through a mixed-method research approach. Observation, simulation using mixing static and dynamic malware analysis methods. Both variants are from the Malware Bazaar site. The total characteristics based on dynamic analysis, the remk variant has 177, and the erqw variant has 190, which increased by 1.8%. The total characteristics based on static analysis, the remk variants have 586, and the erqw variants have 736, which increased by 5.7%. All characteristics from remk to erqw increasing in dynamic analysis, except the number of payloads that decreased about 20%. In static analysis, all characteristics from remk to erqw increase except the number of sections decreased about 1.5%. It can be the affected CPU performance, because the remk variant affects performance by increasing CPU work by 3.74%, while the erqw variant affects performance by reducing CPU work by 1.18%, both compared with normal CPU. which will affect the ransomware's destructive work and require changes in its handling.

Corresponding Author:

Dodon Turianto Nugrahadi
Department of Computer Science, Faculty of Science, Lambung Mangkurat University
Ahmad Yani Street Km 36.5, Banjarbaru 70714, Indonesia
Email: Dodonturianto@ulm.ac.id

1. INTRODUCTION

The Internet has become an important thing for world society. Widespread use of the internet creates security gaps that have the potential to endanger users. Attackers use various techniques to obtain information from victims [1]. Security gap attacks involving software created to steal information or commonly called malicious software (malware) [2]. The development of malware, including viruses and worms, has increased significantly with the increasing number of Internet users involved in daily email communications, and this cannot be separated from the existence of anti-malware software [3], [4]. The increasing number of malwares that commits crimes is a big challenge for digital forensic researchers to carry out malware analysis to identify, find out and develop techniques to detect this malware [3], [5]. Malware analysis as a multi-step process that provides insight into the structure and function of malware, determining its motives and functionality. Apart from that, it is also to get complete information about the capabilities of

malware so that can be aware of the impact of damage or data theft that can be carried out by malware [2], [6]–[8]. There are two ways that have been widely implemented to carry out malware analysis, namely static and dynamic analysis [8], [9].

Ransomware is considered one of the most dangerous malware variants [10], [11]. It is a type of malware that prevents users from accessing or restricts their access to a system or files, either by locking the screen or by encrypting files, to the point of demanding a ransom [12]–[14]. Ransomware STOP/DJVU is the most common family of ransomware viruses and has many variants, and every year many new variants emerge from this ransomware family. STOP/DJVU, a family of ransomware viruses containing more than 240+ variants [15]. This ransomware is most commonly injected into repackaged installers, and spreads via email with malicious attachments, misleading downloads, exploits, web injectors, and so on [16]. One of the STOP/DJVU variants is the remk and erqw variants on the Malware Bazaar site. Based on this site, the erqw variant is the newest variant of the STOP/DJVU variant compared to the remk variant.

The aim of this research was to analyze the Ransomware STOP/DJVU variants remk and erqw, to obtain differences in the characteristics of these two variants, as well as changes in characteristics from the previous variant to the latest variant. Knowledge of these characteristics is to obtain information in order to overcome attacks and describe exactly how both ransomware works. Apart from that, it is also to find out the impact of the ransomware remk and erqw variants on the victim's computer.

2. RESEARCH METHOD

This research is a mixed research, with a mixed-method research approach. It is a combination of qualitative methods and quantitative methods. The data collection technique used in this research is observation method (observation) with the malware analysis method, namely the dynamic analysis method to analyze behavior and the static analysis method to analyze internal structure.

2.1. Dynamic analysis

Dynamic analysis is the process of analyzing the behavior or actions carried out by an application when executing usually in a virtual environment [3], [17]. The static executable analyzer process can only reveal some information about the malware, but running the malware and examining its behavior at runtime provides more insight and improves the ability to identify malware [18], [19]. The dynamic analysis stage includes analysis using the hybrid analysis tool and running samples of both STOP/DJVU variants directly in the virtual lab, to obtain indicator data for the hybrid analysis tool, URL, payload, registry changes, and virtual lab CPU performance. An active approach is carried out by executing ransomware code. The ransomware code is executed under a controlled environment, and the features captured by the controlled environment.

2.2. Static analysis

Static analysis is analyzing software without executing it. These techniques can be applied to various parts of a program [20]. In this static analysis method, the malware file will not be activated directly but will instead be traced, researched, and analyzed against the source code written in the malware program. As a result, the obtained information is very complete. It can provide a very detailed picture of the overall working mechanism of the malware [21], [22]. The static analysis stage includes disassembling and unpacking samples of both STOP/DJVU variants and file-based heuristic analysis of the results of the disassembly and unpacking, to obtain data sections, DLLs, functions, signatures, and strings on the internal parts of the samples. The passive approach is carried out without executing the ransomware code.

2.3. Ingredients and stages

The stages carried out include studying and collecting various information related to the malware to be researched. This including the literature study and data collection from both samples of the STOP/DJVU variant, in the form of identification data and basic information about the ransomware, initial registry data and normal computer CPU performance data. In static analysis, we will trace the work of the ransomware and observe the source code using programs such as program analyzer, debugger, and disassembler [3]. Next, dynamic analysis of the virtual lab experimental environment, the ransomware is executed and traces what happens in the virtual lab environment [17], [23]. Set up virtual lab is the preparation of two research environments in the form of a virtual lab which includes virtual machines, tools, and research materials. Two virtual labs are dedicated to researching each STOP/DJVU variant. Both virtual labs are made isolated from the host computer (not connected to a network or shared folder) to prevent the STOP/DJVU ransomware from escaping the virtual lab and infecting the host computer. The flow of this research is shown in Figure 1.

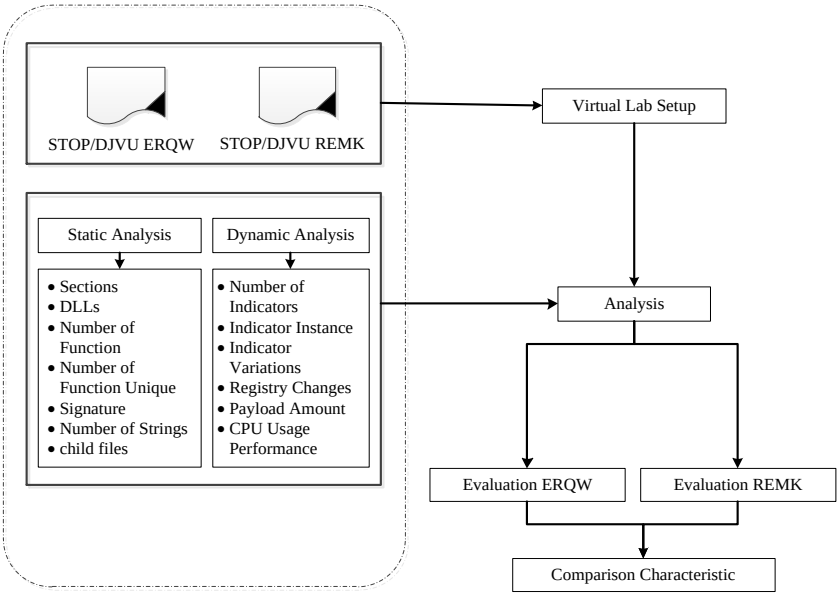

Figure 1. Research flow

Because the virtual lab is not connected to a network, this research focuses on the STOP/DJVU ransomware encryption attack offline [9], [15], [24]. Table 1 shows the software used in this research. The STOP/DJVU ransomware samples are the material used in this research, in the form of STOP/DJVU erqw and remk variants samples. The evaluation stage is a documentation and comparison stage between the results of the dynamic analysis stage and static analysis of the two STOP/DJVU variants. The information obtained from the evaluation then analyzed as a reference for drawing conclusions. These samples were selected based on the reporting date of the variant samples on the Malware Bazaar site:

a) The erqw variant is the latest variant reported in February 2023.
b) The remk variant is the oldest variant that can be downloaded from the Malware Bazaar site, which was reported in March 2020.

Table 1. Software used in research

Name	Function
Oracle VM virtual box	Virtual machine simulator
Malware bazaar	Source of STOP/DJVU ransomware samples
Hybrid analysis	Online and automated malware analysis tool
IDA pro	Disassembler/malware sample unloader
Process hacker	Displays the performance of the entire system on the computer
Regshot	Snapshot tool and analysis of registry changes
XPEViewer	Analysis of PE structure and components
PEId	Estimating the presence of packers
PE view	Shows the structure of PE components
Unpacme	Opens hidden files in PE files

3. RESULTS AND DISCUSSION

3.1. Dynamic analysis results

The results of the dynamic analysis process show that the erqw variant has more features and characteristics than the remk variant, including,

a) The results of the analysis using hybrid analysis tools are based on the number of indicator appearances in the observation. The remk variant has 28 types of indicators with 4 indicator variations and 121 indicator instances while the erqw variant has 29 indicator types with 5 indicator variations and 135 indicator instances, as shown in Table 2.

Table 2. Indicators, indicator variations, indicator instances on both STOP/DJVU variants

Tools results	Remk	Erqw
Number of indicators	28	29
Instance indicator	121	135
Indicator variations	4	5

Hybrid analysis tools also found that both STOP/DJVU variants access the same URL to check whether the victim's device is in the “whitelist” region, namely,

api.2ip.ua/geo.json

However, both STOP/DJVU variants access different URLs to download their payloads. Remk variants access,

nokd.top/ydtftysdtyftysdfsdpen3/get.php

Whereas erqw variant accesses,

bihsy.com/test1/get.php

b) Hybrid analysis tools also found that the remk variant uses 7 types of payloads, while the erqw variant uses 3 types of payloads, as shown in Table 3.

Table 3. Payload on both STOP/DJVU variants

Remk variant	Erqw variant	Function
icalcs.exe	icalcs.exe	Modify file permissions so that ransomware files cannot be deleted
updatewin.exe	-	Fake windows update window
updatewin1.exe	-	Disable windows defender & task manager
updatewin2.exe	-	Modify the hosts file to prevent access to security sites
3.exe	-	Remote accessto control the victim's PC
4.exe	-	Not known
5.exe	build2.exe	Data stealer trojan
-	build3.exe	Encrypts the victim's files

c) Analysis using the Regshot tool resulted in 17 registry changes caused by the remk variant and 18 registry changes caused by the erqw variant, as shown in Table 4.

Table 4. Registry changes in both STOP/DJVU variants

Tools results	Remk	Erqw
Registry	17	18

Registry changes due to these two variants occurred in the same 17 registers, only 1 more registry was different due to the erqw variant. One registry for SysHelper in the erqw variant, namely,

HKU\S-1-5-21-3996184357-4032267556-3958518104-500\Software\Microsoft\Windows\Current Version\
SysHelper: 0x00000001

d) Figure 2 is a graph of the progress a normal computer's CPU performance with the two virtual labs. Figure 2 shown, both samples of STOP/DJVU ransomware activate at random intervals. The difference from activation intervals: the remk variant operates within shorter intervals, whereas the erqw variant operates over relatively longer intervals compared to the remk variant. In the observed intervals, the process of the erqw variant results in CPU performance falling below that of the remk variant and normal CPU conditions. Conversely, the remk variant process causes CPU performance to exceed normal CPU conditions.

e) The analysis is using process hacker tools with observation of 60 data per second with normal scenarios, the remk variant infection, and the erqw variant infection. Observations found that the remk variant resulted in an average of 34.951% virtual lab CPU usage, while the erqw variant resulted in an average of 30.037% virtual lab CPU usage, compared to normal CPU usage performance of an average of 31.213%. This means that the remk variant results in an increase in CPU usage of up to 3.74%, while the erqw variant results in a decrease in CPU usage of up to 1.18%. As shown in Table 5.

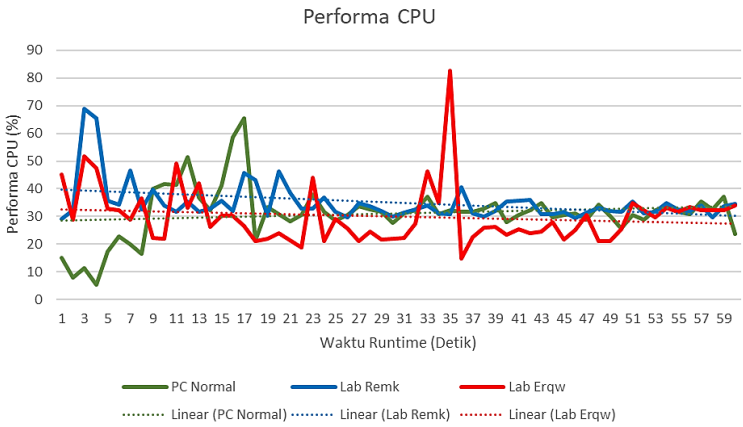

Figure 2. Graph of the progress of the normal computer CPU performance and the two virtual labs

Table 5. CPU performance changes in both virtual labs

Variant	Total CPU usage (%)	CPU average usage (%)
Normal	1,872.78	31.213
Remk	2,097.06	34.951
Erqw	1,802.22	30.037

3.2. Static analysis results

The results of the static analysis process show that,

a) The remk variant has a PE section of 8 sections, while the erqw variant has 5 sections, as shown by the XPEViewer tool for STOP/DJVU remk variant, in Table 6. As shown by the XPEViewer tool for STOP/DJVU erqw variant in Table 7, Figure 3 shows that the XPEViewer tool results show that four sections in the erqw variant only contain empty hex values.

Table 6. Section on STOP/DJVU remk variant

Virtual addresses	Memory map address	Sections
-	00400000	PE Header
00001000	00401000	.text
000a1000	004a1000	.rdata
000a5000	004a5000	.data
00156000	00556000	.gopawo
0015a000	0055a000	.pey
0015b000	0055b000	.yaxu
00164000	00564000	.kadaxu
00166000	00566000	.rsrc

Table 7. Section on STOP/DJVU erqw variant

Virtual addresses	Memory map address	Sections
-	00400000	PE Header
00001000	00401000	.text
0001a000	0041a000	.data
00117000	00517000	.rsrc
00119000	00519000	.reloc

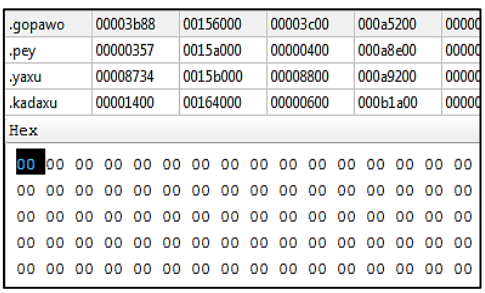

Figure 3. Section containing empty hex values

b) Both variants have the same number of DLLs, namely 2 DLLs in the form of kernel32.dll and user32.dll, as shown by the XPEViewer tool in Table 8.

Table 8. DLLs on STOP/DJVU remk and erqw variants

Variant	hash	etc
Remk	c31ebc12	kernel32.dll
	ebcb8781	user32.dll
Erqw	5b374031	kernel32.dll
	723077ec	user32.dll

c) The erqw variant has total 122 functions with 56 unique functions. The remk variant has total 101 functions with 36 unique functions. Table 9 is a comparison total functions and unique functions.

Table 9. Number of functions and uniq functions in both STOP/DJVU variants

Tools results	Remk	Erqw
Function	101	122
Function unique	36	56

d) Analysis of the XPEViewer tools shows that both variants have the same number of signatures, which is the two types of signatures in the form of the TEA encryption algorithm and anti-debug. Table 10 shows the signatures of the two STOP/DJVU variants.

Table 10. Signature on both STOP/DJVU variants

Variant	Address	Signature
Remk	0040102b	TEA encryption/decryption (0xc6ef3720 0x9e3779b9)
	004a4926	anti-debug: IsDebuggerPresent
Erqw	00404bf6	TEA encryption/decryption (0xc6ef3720 0x9e3779b9)
	419912	anti-debug: IsDebuggerPresent

e) The analysis of the XPEViewer tools also shows that the remk variant contains 437 strings while the erqw variant contains 549 strings. Both strings are filled with lots of information in the form of random strings, sections, functions, DLLs, runtime type identifiers (RTTI), and error messages. Table 11 shows a comparison of the number of strings in the two STOP/DJVU variants.

Table 11. Number of second strings of STOP/DJVU variants

Tools results	Remk	Erqw
Strings	437	549

f) The process of unpacking the two STOP/DJVU ransomware variants using the unpacme tool found child files hidden in the ransomware executable file as in Figure 4.

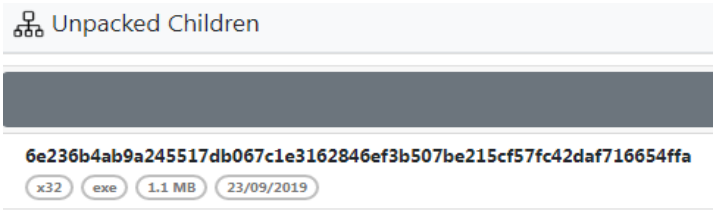

Figure 4. Unpacking child file STOP/DJVU ransomware

Regarding the child files, the two variants have completely the same characteristics. There are only differing in hash identification as shown in Table 12. These two child files have the following characteristics:

- 5 PE sections, which include PE header, text, rdata, data, and rsrc.
- 16 types of DLLs with 236 functions which have different functions as in Table 13.
- 58 signatures, which include 54 encryption algorithm hints, 3 encoding hints, and 1 anti-debug.
- 5974 strings, which contain random strings, application process hints, encryption algorithms, RTTI, error messages, functions, and DLLs.

Table 12. Identification of hash child files both STOP/DJVU variants

Child file variant	SHA256	MD5	SHA1
Remk	6e236b4ab9a245517db067c1e3162846ef3b507be215cf57fc42daf716654ffa	12f4252bae0fa860b95a38895a131d23	042ad1862c01338ed5778dbf7cbff6cda41646f5
Erqw	da98afa9307186b8c28507a3bb53f80ef8be98c9a9553e6748f320719fd188fd	37abc9bcf8951210db525f9ef601b0d664ab75ee	979b880e53e3da8371bdece613ed0553

Table 13. DLL in the second child file of the STOP/DJVU variant and its function

ETC	FUNCTION
RPCRT4	Generate UUID code
MPR	Collect information about internet resources
WININET	Access the internet to whitelist check & download payload
SHLWAPI	File path finding & checking
ADVAPI32	Encryption and registry modification
SHELL32	Executes ransomware files & their payloads
IPHLPAPI	Read internet adapter info
DNSAPI	Calculates and frees memory allocated for DNS
CRYPT32	Converts a string to a byte array
WINMM	Communication and control of multimedia devices (speakers, joysticks)
KERNEL32	Essential functions to run the program
USER32	Stores functions related to the user interface
ole32	Object linking & embedding
OLEAUT32	Installer setup settings
WS2_32	Provides TCP/IP networking
GDI32	Performs primitive drawing functions for output to video displays and printers

All data were generated from the analysis stage. An outline of the data resulting from dynamic analysis and static analysis is made. Once, all the information has been obtained, the percentage difference in each characteristic item between the remk variant and the erqw variant is calculated using the formula. These percentages areas as shown in Tables 14 and 15, as well as Figures 5 and 6.

$$\text{Percentage difference:} \frac{(a-b)}{(a+b)/2} \quad (1)$$

Information,

- a = ransomware data value a
- b = ransomware data value b

Table 14. Percentage characteristics of dynamic analysis results

Data	Variant		Percentage difference (%)
	Remk	Erqw	
Number of indicators	28	29	0.9
Indicator instance	121	135	2.7
Indicator variations	4	5	5.6
Registry changes	17	18	1.4
Payload amount	7	3	-20
Total characteristics of dynamic analysis	177	190	1.8

Table 15. Percentage characteristics of static analysis results

Data	Variant		Percentage difference (%)
	Remk	Erqw	
Sections	8	5	-1.5
DLLs	2	2	0
Function	101	122	4.7
Function unique	36	56	10.9
Signature	2	2	0
Strings	437	549	5.7
Total characteristics of static analysis	586	736	5.7

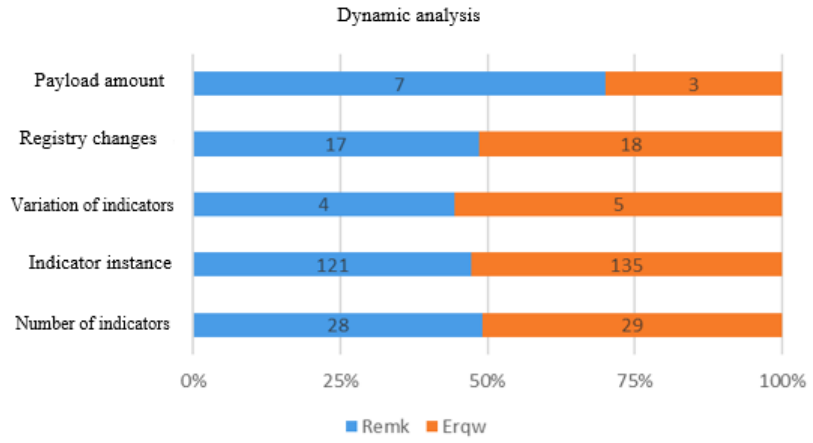

Figure 5. Graph of differences in dynamic analysis results

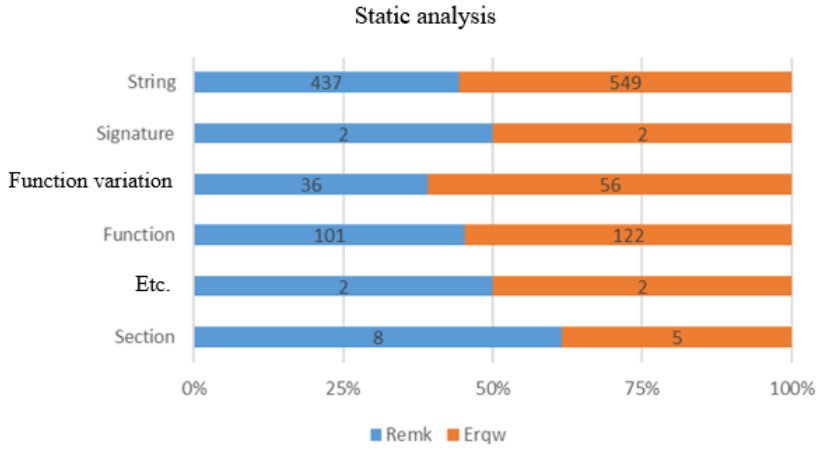

Figure 6. Graph of differences in static analysis results

Based on the percentage results above, on dynamic analysis, the characteristics of the latest variant of the erqw ransomware are more numerous than remk variant. Overall, from total characteristics in the dynamic analysis, the percentage comparison, remk 1.8% more than erqw, and in the static analysis, the percentage of comparison, remk 5.7% more than erqw. However, the dynamic analysis results found that the number of payloads in erqw variant was less than the remk variant, decreasing about 20%, and the static analysis results found that the number of sections in erqw variant was less than the remk variant, decreasing about 1.5%. Based on this, it is estimated that erqw will affect CPU usage performance on the victim's computer.

4. CONCLUSION

The results of dynamic analysis and static analysis show that the erqw variant has a greater percentage of characteristics than the remk variant. The difference in dynamic analysis is 1.8%, while the difference in static analysis is 5.7%. So, the change in the STOP/DJVU ransomware variant from remk in 2020 to erqw in 2023 will result in an increase in characteristics of up to 7.5% which will affect the ransomware's destructive work and require changes in its handling. Dynamic analysis increasing total of indicators, indicator instances, indicator variations, registry changes, but the number of erqw variant payloads decreased from remk variant. Static analysis increasing functions, function variations, number of strings. While the number of DLLs, the number of signatures remains the same. However, the number of sections in the erqw variant has decreased from remk variant. The remk variant resulted in an increase in CPU work of 3.74%, while the erqw variant resulted in a decrease in CPU work of 1.18% compared to CPU performance under normal conditions.

ACKNOWLEDGMENTS

The authors thank the Computer Science Study Program, Faculty of Mathematics and Natural Sciences, Universitas Lambung Mangkurat, for institutional support, and the Data Science Laboratory for providing research facilities. Appreciation is also extended to the anonymous reviewers for their valuable feedback and suggestions.

FUNDING INFORMATION

The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Dodon Turianto Nugrahadi	✓	✓							✓	✓				
Friska Abadi				✓						✓				
Rudy Herteno				✓						✓				
Muliadi				✓						✓				
Muhammad Alkaff				✓						✓				
Muhammad Alvin Alfando	✓	✓							✓	✓				

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

The authors declare that there is no conflict of interest regarding the publication of this paper.

INFORMED CONSENT

Not applicable. No human subjects were involved in this research.

ETHICAL APPROVAL

Ethical approval was not required for this study, as it did not involve experiments on humans, animals, or the use of identifiable personal data.

DATA AVAILABILITY

The malware samples in this study are publicly available from the MalwareBazaar project (https://bazaar.abuse.ch/), a platform maintained by abuse.ch for sharing malware samples with the research community. All data were accessed and downloaded in accordance with the platform's terms of use.

REFERENCES

[1] M. Hazri, "Analysis of PlasmaRAT malware using reverse engineering methods (in Bahasa: Analisis malware PlasmaRAT dengan metode reverse engineering)," *Jurnal Rekayasa Teknologi Informasi (JURTI)*, vol. 4, no. 2, 2020, doi: 10.30872/jurti.v4i2.4131.

[2] R. Adenansi and L. A. Novarina, "Malware dynamic," *JOEICT (Journal of Education and Information Communication Technology)*, vol. 1, no. 1, pp. 37–43, 2017.

[3] A. Amiruddin, P. N. H. Suryani, S. D. Santoso, and M. Y. B. Setiadji, "Utilizing reverse engineering technique for a malware analysis model," *Scientific Journal of Informatics*, vol. 8, no. 2, pp. 222–229, 2021, doi: 10.15294/sji.v8i2.24755.

[4] P. V Shijo and A. Salim, "Integrated static and dynamic analysis for malware detection," *Procedia Computer Science*, vol. 46, pp. 804–811, 2015, doi: 10.1016/j.procs.2015.02.149.
[5] R. Sihwail, K. Omar, and K. A. Z. Ariffin, "A survey on malware analysis techniques: static, dynamic, hybrid and memory analysis," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 8, no. 4–2, pp. 1662–1671, 2018, doi: 10.18517/ijaseit.8.4-2.6827.
[6] V. Patel and P. H. Bhathawala, "A literature review on anti virus and its analysis," *Think India*, vol. 22, no. 2, pp. 315–328, 2019, doi: 10.26643/think-india.v22i2.8732.
[7] N. Kaur and A. Kumar, "A complete dynamic malware analysis," *International Journal of Computer Applications*, vol. 135, no. 4, pp. 20–25, 2016, doi: 10.5120/ijca2016908283.
[8] R. Syahputra and Syaifudin, "Literature study of malware analysis using dynamic and static analysis methods (in Bahasa: Studi literatur analisis malware menggunakan metode analisis dinamis dan statis)," *Jurnal Jaringan Komputer dan Keamanan*, vol. 1, no. 1, pp. 14–24, 2020.
[9] A. Datta, K. A. Kumar, and A. D, "An emerging malware analysis techniques and tools: a comparative analysis," *International Journal of Engineering Research & Technology (IJERT)*, vol. 10, no. 4, 2021. [Online]. Available: https://www.ijert.org/research/an-emerging-malware-analysis-techniques-and-tools-a-comparative-analysis-IJERTV10IS040071.pdf
[10] A. Kapoor, A. Gupta, R. Gupta, S. Tanwar, G. Sharma, and I. E. Davidson, "Ransomware detection, avoidance, and mitigation scheme: a review and future directions," *Sustainability*, vol. 14, no. 1, 2022, doi: 10.3390/su14010008.
[11] C. C. Moreira, D. C. Moreira, and C. de S. de Sales Jr., "Improving ransomware detection based on portable executable header using xception convolutional neural network," *Computers & Security*, vol. 130, 2023, doi: 10.1016/j.cose.2023.103265.
[12] D. Arnold, C. David, and J. Saniie, "PowerShell malware analysis using a novel malware rating system," *IEEE International Conference on Electro Information Technology*, vol. 2022-May, pp. 182–187, 2022, doi: 10.1109/eIT53891.2022.9813771.
[13] B. I. Darmawan, "Simulation and analysis of encryption-based ransomware to map ransomware evolution (in Bahasa: Simulasi dan analisis encryption-based ransomware untuk memetakan evolusi ransomware)," *S.T. Thesis*, Computer Engineering Study Program, Bachelor's Degree Program, Faculty of Industrial Technology, Islamic University of Indonesia, 2019. [Online]. Available: https://dspace.uii.ac.id/bitstream/handle/123456789/18039/08.
[14] S. R. Davies, R. Macfarlane, and W. J. Buchanan, "Comparison of entropy calculation methods for ransomware encrypted file identification," *Entropy*, vol. 24, no. 10, 2022, doi: 10.3390/e24101503.
[15] C. Kapre, "A brief overview of DJVU ransomware family," *Bioscience Biotechnology Research Communications*, vol. 13, no. 14, pp. 249–252, 2020, doi: 10.21786/bbrc/13.14/58.
[16] C. Greinsmark, *Ransomware*. Kristianstad University Sweden, 2020. [Online]. Available: https://www.diva-portal.org/smash/get/diva2:1441210/FULLTEXT01.pdf
[17] M. Akbanov, V. G. Vassilakis, and M. D. Logothetis, "WannaCry ransomware: analysis of infection, persistence, recovery prevention and propagation mechanisms," *Journal of Telecommunications and Information Technology*, no. 1, pp. 113–124, 2019, doi: 10.26636/jtit.2019.130218.
[18] D. Vidyarthi, S. P. Choudhary, S. Rakshit, and C. R. S. Kumar, "Malware detection by static checking and dynamic analysis of executables," *International Journal of Information Security and Privacy*, vol. 11, no. 3, pp. 29–41, 2017, doi: 10.4018/IJISP.2017070103.
[19] S. N. A. Sherazi and A. Qureshi, "Hybrid analysis model for detecting fileless malware," *Electronics*, vol. 14, no. 15, 2025, doi: 10.3390/electronics14153134.
[20] M. I. Yousuf, I. Anwer, A. Riasat, K. T. Zia, and S. Kim, "Windows malware detection based on static analysis with multiple features," *PeerJ Computer Science*, vol. 9, 2023, doi: 10.7717/PEERJ-CS.1319.
[21] E. Bergenholtz, E. Casalicchio, D. Ilie, and A. Moss, "Detection of metamorphic malware packers using multilayered LSTM networks," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12282 LNCS, pp. 36–53, 2020, doi: 10.1007/978-3-030-61078-4_3.
[22] S. Madan, S. Sofat, and D. Bansal, "Tools and techniques for collection and analysis of internet-of-things malware: a systematic state-of-art review," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 10, pp. 9867–9888, 2022, doi: 10.1016/j.jksuci.2021.12.016.
[23] D. Uppal, V. Mehra, and V. Verma, "Basic survey on malware analysis, tools and techniques," *International Journal on Computational Science & Applications*, vol. 4, no. 1, pp. 103–112, 2014, doi: 10.5121/ijcsa.2014.4110.
[24] L. Albshaier, S. Almarri, and M. M. Rahman, "Earlier decision on detection of ransomware identification: a comprehensive systematic literature review," *Information*, vol. 15, no. 8, 2024, doi: 10.3390/info15080484.

BIOGRAPHIES OF AUTHORS

Dodon Turianto Nugrahadi is a lecturer in Computer Science Department, Lambung Mangkurat University. His research interest is centered on data science and computer networking. He completed his bachelor's degree in Informatics Engineering at UK. Petra, Surabaya in 2004. After that, he pursued a master's degree in Information Engineering at Gajah Mada University, Yogyakarta in 2009. His current area of research revolves around computer network, data science, medical informatics and, internet of things. He can be contacted at email: dodonturianto@ulm.ac.id.

Friska Abadi finished him bachelor's degree in Computer Science from Universitas Lambung Mangkurat in 2011. Subsequently, in 2016, he obtained her master's degree from the Department of Informatics at STIMIK Amikom, Yogyakarta. Following that, he joined Universitas Lambung Mangakurat as a lecturer in Computer Science. Currently, he holds the position of head of the software engineering laboratory. His current area of research revolves around software engineering. He can be contacted at email: friska.abadi@ulm.ac.id.

Rudy Herteno was born in Banjarmasin, South Kalimantan. After completing high school, he pursued his undergraduate studies in the Computer Science Department at Lambung Mangkurat University and graduated in 2011. Following his undergraduate program, he gained experience as a software developer for several years, particularly focusing on developing software for local governments. In 2017, he obtained his master's degree in Informatics from STMIK Amikom University. he is lecturer in the Faculty of Mathematics and Natural Sciences at Lambung Mangkurat University. His research interests encompass software engineering, software defect prediction, and deep learning. He can be contacted at email: Rudy.herteno@ulm.ac.id.

Muliadi is a lecturer in the Computer Science Department at Lambung Mangkurat University, where he specializes in artificial intelligence, decision support systems, and data science. His academic journey began with a bachelor's degree in Informatics Engineering from STMIK Akakomin 2004, followed by the attainment of a master's degree in Computer Science from Gadjah Mada University in 2009. With expertise in data science, he also brings valuable skills in start-up business development, digital entrepreneurship, and data management staff. He can be contacted at email: Muliadi@ulm.ac.id.

Muhammad Alkaff completed his undergraduate studies in Computer Science at Brawijaya University, Malang. He advanced his education with a master's degree from the Informatics Department at the Sepuluh Nopember Institute of Technology, Surabaya. Joining Universitas Lambung Mangkurat as a lecturer in 2015, he went on to pursue a Ph.D. at the Computer Science Department of King Abdulaziz University in Saudi Arabia in 2022. His research is primarily focused on the areas of machine learning and reinforcement learning. He can be contacted at email: m.alkaff@ulm.ac.id, malkaff0001@stu.kau.edu.sa.

Muhammad Alvin Alfando was born in Mataraman, Kab. Banjar on April 19 1999. He completed elementary school at SD Negeri 5 in 2011, then continued to SMP Negeri 1 Martapura until graduating in 2014. In 2017, he completed high school education at SMA Negeri 1 Martapura. In 2024, he finished his bachelor's degree in Computer Science from Universitas Lambung Mangkurat. During his studies, he was active in the HIMAKOM (Computer Science Student Association). He is current area of interest and research revolves around software engineering and computer security. He can be contacted at email: malvinalfando10013@gmail.com.