Journal of Social Science and Humanities, 6 (5):25-31, 2023
e-ISSN: 2600 - 9056

DOI: 10.26666/rmp.jssh.2023.5.4

Ransomware Behavior on Windows Endpoint: An Analysis

Wira Z. A. Zakaria[1], Mohd Faizal Abdollah[2], Othman Abdollah[2], S.M. Warusia Mohamed S.M.M[2]

[1]MyCERT, Cybersecurity Malaysia, Menara Cyber Axis, Jalan Impact, 63000 Cyberjaya, Selangor, Malaysia
[2]Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia
*Corresponding Author: wira@cybersecurity.my

Received: 30 June 2023; Revised: 20 July 2023 2023; Accepted: 10 August 2023; Published: 15 October 2023

Abstract: This paper provides a comprehensive examination of ransomware behavior on Windows endpoints, exploring the intrusion mechanisms, proliferation methods, and the mitigating strategies that can be employed. It provides a comparative analysis of several ransomware families and their effects on Windows systems, culminating with suggestions for future research directions in enhancing endpoint security against ransomware attacks. In the wake of a rising number of ransomware attacks worldwide, epitomized by the damaging disruptions to the Colonial Pipeline and the Irish Health Service Executive, the persistent threat of ransomware to critical infrastructure has never been more apparent. While Windows endpoints remain primary targets, these attacks have also highlighted a less explored but crucial aspect of ransomware behavior: the exploitation of Application Programming Interface (API) calls integral to the Windows operating system. This comprehensive study provides an exhaustive investigation into the interplay between ransomware and Windows APIs, emphasizing the patterns of invocation and manipulative misuse by various ransomware families. By investigating specific API calls, such as the CryptEncrypt function in the Cryptography API for encryption, and the CreateFile and WriteFile functions in the File API for file system interaction, we illuminate the mechanisms by which ransomware carry out their damaging actions. Further, using the real-world examples drawn from the Colonial Pipeline and Irish Health Service Executive incidents, among others, the study shows how these API calls were manipulated during actual ransomware attacks. In these cases, ransomware like DarkSide and Conti used Windows APIs not just for the primary tasks of encryption and file system manipulation, but also for achieving network communication, maintaining persistence, and even thwarting detection. By presenting a comparative analysis of API call sequences in both benign and ransomware-infected Windows environments, this study serves as a critical exploration into the behavior of these malicious entities. The different patterns observed provide us with valuable insights into their operational strategies and offer opportunities for the development of detection heuristics. The insights derived from this research contribute significantly to our understanding of the behavior patterns of recent, high-profile ransomware attacks. In turn, this work aims to guide the evolution of more sophisticated, behavior-based detection mechanisms, thus strengthening the security posture of Windows endpoints. Ultimately, this study underscores the need for continuous research into API call patterns, as the cybersecurity landscape continues to face dynamic and increasingly sophisticated threats.

Keywords: *Ransomware, Windows endpoint, API calls, Ransomware lifecycle, Ransomware behavior*

1. Introduction

Ransomware is a type of bad software that encrypts the victim's files and then asks for money in exchange for the key to unlock them [1]–[3]. The term "ransomware" wasn't used until the late 2000s, but the first known ransomware attack happened in the late 1980s with the AIDS Trojan [4]. But ransomware really came to the attention of the public around the middle of the 2010s, when digital payment methods like Bitcoin made it easier for cybercriminals to collect ransoms without being seen. Ransomware attacks can happen to

Corresponding Author: Wira Z. A. Zakaria, MyCERT, Cybersecurity Malaysia, Menara Cyber Axis, Jalan Impact, 63000 Cyberjaya, Selangor, Malaysia. Email: wira@cybersecurity.my

anyone, but the most well-known ones happen to businesses or public institutions, where the stakes are higher and there is more at risk. The cost of these attacks can be very high, from the direct cost of the ransom to the indirect costs of lost productivity, damage to a company's reputation, and the work that needs to be done to get back on track [5]. Ransomware can be roughly put into a few different types based on how it acts and how it works. The most common ways to divide things are:

Ransomware that encrypts files is the most common type [6], [7]. It works by putting a strong encryption algorithm on the files of the victim. The attackers then hold the decryption key hostage and ask for money in exchange for it. CryptoLocker, Locky, and WannaCry are all examples of ransomware that encrypts files.

Locker ransomware, also called "computer locker," doesn't encrypt files. Instead, it locks the user out of their device by making it impossible to use the user interface. The attackers then ask for a ransom to let the device work again. Ransomware with a police theme, like Reveton and Urausy, is one example [8]–[10].

Scareware, also called FakeAV, is a type of ransomware that looks like antivirus software. It often says that threats have been found on the victim's computer and wants money to get rid of them. Even though it isn't as bad as some other types of malware, it can still be annoying and can sometimes let other types of malware in [11].

Doxware or Leakware is a type of ransomware that threatens to make private information or data public if the victim doesn't pay a ransom. This method uses the fear of getting in trouble with the law or having your reputation hurt.

RaaS stands for "Ransomware as a Service." This is a type of cybercrime model in which people who make ransomware sell or rent it to other criminals, who then use it to attack other people. Most of the time, the creators get a share of the money. GandCrab and REVil are two examples [12].

Mobile Ransomware: This type of ransomware is designed to attack mobile devices, especially those that run Android. It can either lock the device or make the files on the device unreadable. Simplocker is an example of ransomware for mobile phones [13]–[15].

Wiper Ransomware: This type often looks like ransomware, but its main goal isn't to get money, but to destroy data. It encrypts or deletes the victim's data in a way that can't be undone. This kind of ransomware is shown by NotPetya.

2. Ransomware Infection Vectors

Threat actors use ransomware infection vectors to send ransomware to the computers of their victims. Ransomware exploits various intrusion mechanisms to penetrate Windows systems, such as phishing emails, malicious downloads, software vulnerabilities, and unsecured remote desktop protocols [16]–[19]. Phishing emails are the most common way for ransomware to get into a computer. In this case, attackers send emails that look like they came from legitimate sources and try to get the recipients to click on a link or open an attachment that installs ransomware.

Exploit kits are sets of tools that can be used to install ransomware by taking advantage of flaws in software. When a user goes to a website that has been hacked or has malicious ads, the exploit kit scans the user's system for weaknesses and uses them to install ransomware [20]–[22].

Malvertising is when online ads are used to spread malware without the user having to do much or anything. Malvertising can make a user infected with ransomware just by going to a website with malicious ads.

Drive-by Downloads: This happens when a user visits a compromised website that automatically downloads ransomware onto the user's system without the user's knowledge.

RDP (Remote Desktop Protocol) Attacks: In this type of attack, hackers look for systems with RDP access that can be reached from the internet. They then use methods like brute-force attacks to get in. Once inside, they can install the ransomware on their own.

Social engineering is the process of getting people to do things that let ransomware infect their computers. It can include scareware, bait-and-switch, and other ways to trick people [19], [23].

Ransomware can also be downloaded and installed on a system by other malware. For example, Emotet, which started out as a banking Trojan, changed over time to become a way for other types of malware, like ransomware, to spread [24].

Botnets, which are groups of infected computers, can be used to spread ransomware. Most of the time, this is done with the help of "command and control" servers that tell the infected systems what to do. Infected USB sticks and other removable media can be used to spread ransomware. When an infected device is plugged into a clean computer, ransomware is installed.

3. Ransomware Attack Stages

Like other types of cyberattacks, a ransomware attack has a structured process that can be broken up into different stages. Each stage has its own characteristics and needs a different way to defend against it. In the first stage, "infiltration," the malware gets onto the system of the victim. In the next stage, "installation," it plants itself deeply in the system, often changing system settings to make sure it stays there.

The next step is the exploitation phase, where the

ransomware shows how bad it really is by doing things like encrypting data or locking down the system. In most cases, it then starts talking to an attacker-controlled command and control server (C2) to do things like send information about the infected system or get encryption keys.

Then it moves on to the ransom demand stage, where it shows a message to the victim with payment information. Because cryptocurrencies like Bitcoin are anonymous, they are usually asked for as payment. If the victim gives in and pays the ransom, the next step, decryption, may or may not happen, depending on how "honest" the attackers are.

The lifecycle ends with the removal and recovery stage, where attempts are made to get rid of the ransomware and data recovery strategies are used, ideally from backups that haven't been infected and are up to date. Understanding each stage of ransomware's lifecycle is important for coming up with effective ways to stop, find, and get back from these annoying and potentially dangerous cyber threats.

4. Ransomware Behavior on Windows Endpoint

Ransomware operates in stages: delivery, infection, encryption, and ransom demand. This section offers a comparative analysis of various ransomware families including WannaCry, NotPetya, and Ryuk, focusing on their behaviors on Windows endpoints, such as file encryption methods, communication with command-and-control servers, persistence tactics, and self-propagation techniques.

4.1 Delivery and Infection

Ransomware is often delivered via malicious emails or downloads. For example, Emotet, a sophisticated Trojan often used as a downloader for ransomware, leverages email phishing campaigns for delivery. These emails often contain Word documents with macros that, when enabled, trigger the download and execution of the ransomware payload. In the case of the notorious WannaCry, the delivery was through a vulnerability in the Windows Server Message Block (SMB) protocol. Once a system was infected, the ransomware proliferated to other unpatched Windows systems on the same network.

4.2 Encryption and Persistence

Once delivered and executed, ransomware typically proceeds to enumerate and encrypt files. Ryuk ransomware provides a potent example of this process. It targets a broad range of file extensions, focusing on those likely to hold valuable data. It also attempts to disable Windows Volume Shadow Copy Service (VSS), thus inhibiting the system's native ability to recover previous versions of files, making the attack more potent .

Another ransomware, GandCrab, goes a step further by employing an anti-analysis technique. It checks the system's keyboard layout, and if it matches certain countries (mostly Eastern European), it terminates its execution, thereby eluding analysis from researchers in those countries.

4.3 Command and Control Communication

Many ransomware variants communicate with a command and control (C2) server to receive encryption keys and send back victim information. For instance, Cerber ransomware uses a domain generation algorithm (DGA) to generate multiple pseudo-random domain names to communicate with its C2 servers. This approach increases the resilience of the ransomware's communication infrastructure, making it harder for authorities to shut it down.

4.4 Ransom Demand

Following successful encryption, most ransomware will alter the desktop wallpaper, display a dialog, or create a text file detailing the ransom demand. NotPetya, despite its semblance to ransomware, is an exception to this rule. Although it displays a ransom note, it is designed more as a wiper, with its main goal being data destruction. The ransom note in this case acts more as a disguise rather than a true request for payment.

These examples underline the complex and evolving behaviors of ransomware on Windows endpoints. The sophistication level of modern ransomware requires defenders to employ advanced tools and strategies for protection, including real-time behavioral analysis, robust backup strategies, and prompt patching of known vulnerabilities.

5. Windows API Calls

Windows API (Application Programming Interface) calls are a set of routines, protocols, and tools that are used to build software applications on Windows operating systems. Basically, they are the building blocks that developers use to make programmes that interact with the Windows operating system and its services [25]–[28].

API calls show what the operating system can do and let applications ask it for services, like accessing hardware resources, managing memory, creating, and managing processes, or interacting with the filesystem [29]. Here are some examples of some of the things that Windows API calls can do:

File Management: You can work with files on the filesystem using functions like CreateFile, ReadFile, WriteFile, and CloseHandle. They make it possible for applications to open, read, write, and close files.

Process and Thread Management: Applications can create,

control, and end processes and threads by using functions like CreateProcess, ExitProcess, CreateThread, and ExitThread.

Memory Management: Applications can allocate and release memory using functions like GlobalAlloc, GlobalFree, VirtualAlloc, and VirtualFree.

Device Input and Output: Functions like ReadConsoleInput, WriteConsoleOutput, and DeviceIoControl let applications talk to devices and handle user input.

Network Services: Functions like socket, bind, connect, send, and recv let you connect to a network and send data over it.

Handling errors: Functions like GetLastError and SetLastError give applications ways to deal with errors.

Windows Registry: Applications can talk to the Windows Registry using functions like RegOpenKeyEx, RegQueryValueEx, and RegSetValueEx.

These API calls give applications the basic features they need to do their jobs. Malware, like ransomware, can use them to do bad things. For example, ransomware might use the CreateFile function to open files for encryption and the WriteFile function to write the encrypted data back to the files. So, one of the most important ways to find and stop malware is to keep an eye out for API call patterns that don't make sense.

6. Ransomware API Calls

Ransomware uses various Windows API calls to carry out their operations [28], [30]–[32]. The following are three examples of API calls utilized by different ransomware families:

CryptEncrypt (Cryptography API): Many ransomware variants use the CryptEncrypt function, part of the Windows Cryptography API (CryptoAPI), to encrypt the user's data. For instance, the Locky ransomware uses this API call to execute its encryption routine. After generating a symmetric AES-128 encryption key, Locky calls the CryptEncrypt function to encrypt the victim's files.

CreateFile (File API): Ransomware often needs to interact with the file system to identify and encrypt the victim's files. The CreateFile function, part of the Windows File API, is used for this purpose. For instance, the WannaCry ransomware uses the CreateFile API call to open the files it targets for encryption.

InternetOpenUrl (WinINet API): Ransomware often communicates with its command-and-control (C&C) server to send information about the victim and retrieve encryption keys. The InternetOpenUrl function, part of the Windows Internet (WinINet) API, is used to open a resource specified by a complete FTP or HTTP URL. For example, the Cerber ransomware uses the InternetOpenUrl API call to connect to its C&C server and report a successful infection.

DeleteFile (File API): The DeleteFile function is part of the Windows File API and is used to delete an existing file. Ransomware like Ryuk uses this API call to delete shadow copies of files, which are often used as a method for recovery after a ransomware attack. Deleting these copies makes it harder for victims to recover their data without paying the ransom.

RegOpenKeyEx and RegSetValueEx (Registry API): These two functions are part of the Windows Registry API and are often used by ransomware to gain persistence on the infected system. The RegOpenKeyEx function is used to open a registry key, and the RegSetValueEx function is used to set the data for a specified value in a registry key. The Locky ransomware, for instance, uses these functions to add itself to the list of programs to be launched at startup, ensuring that it can continue its operations after the system is rebooted.

FindFirstFile and FindNextFile (File API): These two functions, part of the Windows File API, are used to search a directory for a file or subdirectory with a name that matches a specific name. Ransomware like WannaCry uses these functions to enumerate files in a directory, so it can identify the files it wants to encrypt.

6.1 Wannacry

WannaCry, the ransomware that caused problems around the world in May 2017, uses several Windows API calls to do its bad things [22], [33], [34]. Depending on the version of WannaCry, the API calls used can be different, but some of the most common ones are:

CryptAcquireContext: This API call is used to acquire a handle to a particular key container within a cryptographic service provider (CSP). This is the first step in initializing the CryptoAPI, which WannaCry uses to generate the encryption keys for file encryption.

CryptGenKey: This API call generates a random cryptographic key. In the case of WannaCry, it uses this function to generate the symmetric key for the AES (Advanced Encryption Standard) encryption algorithm.

CryptImportKey: This API call imports a cryptographic key from a key BLOB (binary large object) into a CSP. WannaCry uses this function to import the RSA public key used to encrypt the AES key.

CryptEncrypt: This API call encrypts data. WannaCry uses this function twice: first, to encrypt the victim's files using the AES key, and then to encrypt the AES key itself using the RSA public key.

CryptReleaseContext: This API call is used to release the handle acquired by CryptAcquireContext.

FindFirstFile / FindNextFile: These API calls are used to enumerate the files in a directory. WannaCry uses these functions to find the files it needs to encrypt.

MoveFileEx: This API call is used to move a file, possibly replacing an existing file. WannaCry uses this function to replace the original files with their encrypted versions.

CreateService / StartService: These API calls are used to create and start a new service. WannaCry uses these functions to install itself as a service, thereby gaining persistence on the infected system.

6.2 Gandcrab

GandCrab was one of the most popular and active ransomware families from early 2018 until mid-2019, when it was said to have stopped being used. Like other types of ransomwares, GandCrab uses different Windows API calls to do its bad things [35], [36]. Some of the most common ones are:

CryptAcquireContext: This API call is used by GandCrab to gain access to the cryptographic service provider (CSP), which is part of the process for initializing cryptographic operations.

CryptCreateHash: This API call is used to initiate the hashing of the victim's system information, which GandCrab uses as part of its process to generate a unique identifier for each victim.

CryptHashData: This function is used by GandCrab to compute the hash of the victim's system information.

CryptDeriveKey: This API call is used to generate a cryptographic session key derived from the hashed system information.

CryptEncrypt: This API function is used by GandCrab to encrypt files on the victim's computer.

CreateFile: GandCrab uses this API call to open existing files for reading and writing, specifically for the encryption process.

WriteFile: This function is used to write the encrypted data back into the victim's files.

FindFirstFile / FindNextFile: These API calls are used by GandCrab to enumerate the files in a directory and subdirectories to identify potential files for encryption.

InternetOpen/InternetConnect/HttpOpenRequest/HttpSendRequest: These API calls are used to establish a connection and send a HTTP request to the Command & Control (C2) server for purposes such as sending victim information and receiving instructions.

ShellExecute: This API call is used by GandCrab to launch processes or open files, such as displaying the ransom note to the user.

7. Conclusion & Future Work

Any comments and suggestions are welcomed so that we can constantly improve this template to satisfy all authors' research needs. In conclusion, ransomware continues to be a major threat to cyber security around the world. Criminals are coming up with more and more sophisticated ways to avoid detection and shut down systems. Since Windows API calls are the building blocks for how applications interact with the Windows operating system, they are a natural way for ransomware to do its bad things. By looking at the API calls used by well-known ransomware like WannaCry and GandCrab, we can learn important things about how they work. This gives us a solid foundation for making effective defenses.

There is no one way to defend against this changing threat. So, it's important to keep a layered defense posture. This includes keeping systems and software up to date, backing up data often and well, teaching users how to spot and avoid ransomware attacks, and using advanced detection technologies that can spot and stop API call patterns that aren't normal.

In the end, fighting ransomware is an ongoing battle that requires both individuals and organizations to stay alert, come up with new ideas, and change. By working together like this, we can try to stay one step ahead of threat actors and make sure that our digital landscapes are safe.

API (Application Programming Interface) calls are needed for applications to talk to the operating system that runs them. The Windows operating system has several APIs that software developers can use to make Windows-compatible apps. Malware, like ransomware, can use these APIs to do bad things on a computer that is infected with it. By looking at the patterns of API calls, you can spot possible ransomware activity. The way different API calls work together can show more than any one call alone. For example, if a process creates a new file, writes data to it, deletes the original file, and then renames the new file to look like the original, it could be a sign of ransomware that encrypts files. Machine learning algorithms can be used to automatically look at these API call patterns and learn from them. Then, these algorithms can find possible ransomware

activity with high accuracy, which makes it possible to respond quickly and stop it. So, keeping an eye on and analyzing API calls can be a powerful weapon against ransomware.

REFERENCES

[1] U. Urooj, M. A. Maarof, and B. Ali Saleh Al-Rimy, "A proposed Adaptive Pre-Encryption Crypto-Ransomware Early Detection Model." 2021.

[2] S. H. Kok, A. Abdullah, N. Z. Jhanjhi, and M. Supramaniam, "Ransomware, Threat and Detection Techniques: A Review," *IJCSNS International Journal of Computer Science and Network Security*, vol. 19, no. 2, pp. 136–146, 2019.

[3] S. J. Lee, H. Y. Shim, Y. R. Lee, T. R. Park, S. H. Park, and I. G. Lee, "Study on Systematic Ransomware Detection Techniques," *International Conference on Advanced Communication Technology, ICACT*, vol. 2022-Febru, pp. 297–301, 2022, doi: 10.23919/ICACT53585.2022.9728909.

[4] J. de Groot, "A History of Ransomware Attacks: The Biggest and Worst Ransomware Attacks of All Time," *Digital Guardian*, 2019.

[5] A. Cartwright and E. Cartwright, "Ransomware and reputation," *Games (Basel)*, vol. 10, no. 2, 2019, doi: 10.3390/g10020026.

[6] E. P. Torres P. and S. G. Yoo, "Detecting and neutralizing encrypting Ransomware attacks by using machine-learning techniques: A literature review," *International Journal of Applied Engineering Research*, vol. 12, no. 18, pp. 7902–7911, 2017.

[7] J. Modi and B. Eng, "Detecting Ransomware in Encrypted Network Traffic Using Machine Learning," 2019.

[8] A. Adamov, A. Carlsson, and T. Surmacz, "An analysis of lockergoga ransomware," *2019 IEEE East-West Design and Test Symposium, EWDTS 2019*, pp. 1–5, 2019, doi: 10.1109/EWDTS.2019.8884472.

[9] J. A. Gómez-Hernández, L. Álvarez-González, and P. García-Teodoro, "R-Locker: Thwarting ransomware action through a honeyfile-based approach," *Comput Secur*, vol. 73, pp. 389–398, Mar. 2018, doi: 10.1016/j.cose.2017.11.019.

[10] J. A. Gómez-Hernández, R. Sánchez-Fernández, and P. García-Teodoro, "Inhibiting crypto-ransomware on windows platforms through a honeyfile-based approach with R-Locker," *IET Information Security*, vol. 16, no. 1. pp. 64–74, 2022. doi: 10.1049/ise2.12042.

[11] C. H. Malin, T. Gudaitis, T. J. Holt, and M. Kilger, "Phishing, Watering Holes, and Scareware," in *Deception in the Digital Age*, Elsevier, 2017, pp. 149–166. doi: 10.1016/b978-0-12-411630-6.00005-0.

[12] T. August, D. Dao, and M. Florin Niculescu, "Economics of Ransomware Attacks Technology Support and Demand for Cloud Infrastructure Services: The Role of Service Providers View project Economics of Ransomware Attacks." [Online]. Available: https://www.researchgate.net/publication/331688623

[13] A. Arbor, *Ransomware goes mobile: An analysis of the threats posed by emerging methods*. [Online]. Available: https://search.proquest.com/docview/1679446970?accountid=34984

[14] N. Andronio, S. Zanero, and F. Maggi, "Dissecting and Detecting Mobile," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2015, doi: 10.1007/978-3-319-26362-5_18.

[15] D. Venugopal and G. Hu, "Efficient signature based malware detection on mobile devices," *Mobile Information Systems*, vol. 4, no. 1, pp. 33–49, 2008, doi: 10.1155/2008/712353.

[16] A. Zimba, Z. Wang, and H. Chen, "Reasoning crypto ransomware infection vectors with Bayesian networks," *2017 IEEE International Conference on Intelligence and Security Informatics (ISI)*, pp. 149–151, 2017, doi: 10.1109/ISI.2017.8004894.

[17] A. Zimba, "Malware-Free Intrusion: A Novel Approach to Ransomware Infection Vectors." [Online]. Available: https://sites.google.com/site/ijcsis/

[18] A. Zimba, "Malware-Free Intrusion : A Novel Approach to Ransomware Infection Vectors," vol. 15, no. 2, pp. 317–325, 2017.

[19] P. L. Gallegos-Segovia, J. F. Bravo-Torres, V. M. Larios-Rosillo, P. E. Vintimilla-Tapia, I. F. Yuquilima-Albarado, and J. D. Jara-Saltos, "Social engineering as an attack vector for ransomware," *2017 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*, 2017, doi: 10.1109/CHILECON.2017.8229528.

[20] M. Hopkins and A. Dehghantanha, "Exploit Kits: The production line of the Cybercrime economy?," *2015 2nd International Conference on Information Security and Cyber Forensics, InfoSec 2015*, vol. 2015, no. January 2015, pp. 23–27, Nov. 2016, doi: 10.1109/InfoSec.2015.7435501.

[21] Z. Wang, "y RansomTracer : Exploiting Cyber Deception for Ransomware Tracing," no. 61702508, pp. 227–234, 2018, doi: 10.1109/DSC.2018.00040.

[22] D. Y. Kao, S. C. Hsiao, and R. Tso, "Analyzing WannaCry Ransomware Considering the Weapons and Exploits," *International Conference on Advanced Communication Technology, ICACT*, vol. 2019-Febru, no. 2, pp. 1098–1107, 2019, doi:

10.23919/ICACT.2019.8702049.

[23] M. Lansley, N. Polatidis, S. Kapetanakis, K. Amin, G. Samakovitis, and M. Petridis, "Seen the villains: Detecting Social Engineering Attacks using Case-based Reasoning and Deep Learning."

[24] C. Tankard and D. Pathways, "The threat of fileless trojans," *Network Security*, vol. 2018, no. 3, p. 20, 2018, doi: 10.1016/S1353-4858(18)30026-6.

[25] G. G. Sundarkumar, V. Ravi, I. Nwogu, and V. Govindaraju, "Malware detection via API calls, topic models and machine learning," *IEEE International Conference on Automation Science and Engineering*, vol. 2015-Octob, pp. 1212–1217, 2015, doi: 10.1109/CoASE.2015.7294263.

[26] V. Garg and R. K. Yadav, "Malware Detection based on API Calls Frequency," *2019 4th International Conference on Information Systems and Computer Networks, ISCON 2019*, pp. 400–404, 2019, doi: 10.1109/ISCON47742.2019.9036219.

[27] A. Sami, B. Yadegari, N. Peiravian, S. Hashemi, and A. Hamze, "Malware detection based on mining API calls," *SAC '10 Proceedings of the 2010 ACM Symposium on Applied Computing*, pp. 1020–1025, 2010, doi: 10.1145/1774088.1774303.

[28] S. Sheen, "Ransomware detection by mining API call usage," *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 983–987, 2018.

[29] S. Z. Mohd Shaid and M. A. Maarof, "In memory detection of Windows API call hooking technique," *I4CT 2015 - 2015 2nd International Conference on Computer, Communications, and Control Technology, Art Proceeding*, no. August, pp. 294–298, 2015, doi: 10.1109/I4CT.2015.7219584.

[30] M. Almousa, S. Basavaraju, and M. Anwar, "API-Based Ransomware Detection Using Machine Learning-Based Threat Detection Models," *2021 18th International Conference on Privacy, Security and Trust, PST 2021*, 2021, doi: 10.1109/PST52912.2021.9647816.

[31] M. Scalas, D. Maiorca, F. Mercaldo, C. A. Visaggio, F. Martinelli, and G. Giacinto, "On the effectiveness of system API-related information for Android ransomware detection," *Comput Secur*, vol. 86, pp. 168–182, Sep. 2019, doi: 10.1016/j.cose.2019.06.004.

[32] P. Bajpai and R. Enbody, "An Empirical Study of API Calls in Ransomware," *IEEE International Conference on Electro Information Technology*, vol. 2020-July, pp. 443–448, 2020, doi: 10.1109/EIT48999.2020.9208284.

[33] J. Fruhlinger, "What is WannaCry ransomware," *CSOonline*, 2018.

[34] M. Akbanov, V. G. Vassilakis, I. D. Moscholios, and M. D. Logothetis, "Static and Dynamic Analysis of WannaCry Ransomware." [Online]. Available: www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com

[35] A. Oktaviani and M. Syafrizal, "GandCrab Ransomware Analysis on Windows Using Static Method," vol. 3, no. 2, pp. 163–175, 2021, doi: 10.12928/biste.v3i2.4884.

[36] "Gandcrab ransomware analysis report".