2020 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT)

Behavior-Based Detection of Cryptojacking Malware

Dmitry Tanana
Laboratory of Combinatorial Algebra
Ural Federal University
Yekaterinburg, Russia
ddtanana@urfu.ru

***Abstract*—With rise of cryptocurrency popularity and value, more and more cybercriminals seek to profit using that new technology. Most common ways to obtain illegitimate profit using cryptocurrencies are ransomware and cryptojacking also known as malicious mining. And while ransomware is well-known and well-studied threat which is obvious by design, cryptojacking is often neglected because it's less harmful and much harder to detect. This article considers question of cryptojacking detection. Brief history and definition of cryptojacking are described as well as reasons for designing custom detection technique. We also propose complex detection technique based on CPU load by an application, which can be applied to both browser-based and executable-type cryptojacking samples. Prototype detection program based on our technique was designed using decision tree algorithm. The program was tested in a controlled virtual machine environment and achieved 82% success rate against selected number of cryptojacking samples. Finally, we'll discuss generalization of proposed technique for future work.**

Keywords—Cryptojacking, malicious mining, behavioral analysis, cryptocurrencies, cybercrime

I. Introduction

Nowadays, cryptocurrencies are a huge market with estimated value of 237.1 billion dollars in 2019 [1]. Cryptocurrency mining or, in other words, generating new cryptocoins usually requires extremely huge amount of processing power, with only Bitcoin network calculating 71 quintillion hashes every second, consuming 7 Mwh. Increasing popularity of cryptocurrencies motivates cybercriminals to use specific kind of malware – cryptojackers – which are used to mine cryptocurrencies on victims' computers for profit. Various companies, such as Kaspersky [2], Symantec [3], and Eset [4] report increasing amount of attacks which result in unauthorized mining on affected computers. In the first 6 months of 2019, more than 50 millions of cryptojacking attacks have been registered, which is 450% more comparing to same period of 2018 [5].

Originally, browser-based mining was proposed in 2013 in project called TidBit, which was essentially the alternative for regular ads. Instead of showing annoying advertisement towards users, the site owner would instead use a share of their CPU to mine cryptocurrencies. But cybercriminals heavily deployed this technology for their own goals starting with cryptocurrencies boom in 2017.

And while cryptojacking doesn't harm infected users directly, victims still suffer losses as their hardware wears out and their processing power works towards interests of cybercriminals instead of their own. The cryptojacking attack is even more harmful for businesses – if a cluster of computers is infected, the business would suffer huge financial losses associated with payment for additional consumed electricity aswell as replacing processing units which worked under huge load for a long time.

Therefore early detection of cryptojacking malware is extremely important, because it will help to minimize the losses for victims. In this paper we will propose a new method for detecting cryptojackers based on standart deviation of used CPU power and present a prototype tool which would point at potential cryptojacking processes in Windows OS.

II. Cryptojackers classification and previous work

A. Classification

Malware detection is one of the most relevant tasks in the world of information security. There are two main approaches to solve this task: statistical analysis and dynamical analysis. Statistical analysis tools usually look for specific signatures in files and compare them to the signatures of known malware within the tool's library. It is characterized by a high degree of reliability upon detection. Dynamical analysis studies the behavior of an already running program, therefore it is also called behavioral analysis. Both those approaches can be applied toward cryptojacking malware.

However, there're two main types of cryptojacking malware. First one is regular executable-type malware, which is launching as an application on affected computer and delivered as payload via some other method, second one, which is more prevalent, are browser-based javascript cryptojackers, they simply execute within the browser, when victim enters the infected web-site. Unfortunately, statistical signature analysis, which is employed by most antiviral tools is ineffective against fileless malware, such as browser-based cryptojackers and cannot be applied at all to zero-day attacks by both executable-type and browser-based cryptojackers. Therefore, we think that the detection of cryptojacking malware is better performed by behavioral analysis methods. In this paper we'll design and implement prototype program, which will detect both executable-type and browser-based cryptojackers in Windows OS based on one behavior cryptojacking malware must always perform – mine cryptocurrencies on infected processing unit.

B. Previous work

S. Eskandari et al. in their paper "A first look at Browser-Based Cryptojacking" [6] observed internet sites in order to distinguish those with presense of CoinHive, CryptoLoot and JSEcoin web-miners. Their work focuses on detecting scripts within web-pages by simply looking for script links, such as "coinhive.min.js". While it's very simple and allowed authors to detect large amount of malicious web-sites, it produced some number of false positives, mostly on security sites discussing cryptojackers, and a large number of false

This work was supported by the Ministry of Science and Higher Education of the Russian Federation, project no. FEUZ-2020-0016 .

negatives due to cybercriminals deploying code obfuscation or loading scripts via proxies.

D. Carlin et al. in their paper "Detecting Cryptomining Using Dynamical Analysis" [7] have used machine learning to dynamically analyse opcodes of non-executable subject files for browser-based cryptojacking malware. They report 99% detection rate on on 589 browser-based cryptojacker samples. And while their results are impressive, they don't detect executable-type cryptojackers.

Finally, W. Wang et al. in their paper "SEISMIC: SEcure in-lined script monitors for interrupting cryptojacks" [8] have proposed detecting and interrupting unauthorized, browser-based cryptomining based on semantic signature-matching. Unlike Eskandari's work, that approach is more robust than current static code analysis defenses, which are susceptible to code obfuscation attacks.

Unlike previous work, our detection program will deal with both browser-based and executable-type cryptojackers using same algorithm based on CPU usage heuristics.

III. Detection algorithm

A. Instruments

To operate successfully, the detection program should implement the following steps:

1. Gather data on analyzed process
2. Extract signs of infection from that data
3. Further analysis of the signs in order to make a decision whether the analyzed process is potential cryptojacker or not.

To do those steps, we need to find the signs of infection themselves, and in addition, we must offer an algorithm for making the final decision – infected or not.

In order to obtain data about running processes and application we suggest using Windows Management Instrumentation(WMI) [9].

It is a Windows management technology that operates at the kernel level of the OS, but it allows us to track not only events that occur in the kernel, but also events which were defined in user applications. Event parsing will allow us to obtain data on usage of any resource within Windows OS by any running process.

Also, since WMI operates at the kernel level, it can avoid some stealth techniques employed by cryptojackers. For example, some cryptojacker samples are known to go idle if user is launching Task Manager, preventing their behavioral detection by regular users. In order to determine indicators of cryptojackers activity within Windows we've conducted an experiment in controlled virtual machine enviroment during which we've analysed 25 samples were studied alongside with several legitimate resource-intensive applications, such as RenderMan, Adobe Photoshop and Blender in order to exclude indicators which are not exclusive for cryptojackers. It turned out to be possible to divide the set of cryptojackers samples into two categories: individual Windows executable files and cryptojackers implemented as a browser script.

B. Infection indicators

We've started determining of infection indicators by analyzing the use of CPU resources.

The experiment showed that usually cryptojackers use the resources of the central processor very intensively and the average share of CPU usage by cryptojackers was around 73%. Unfortunately, some legitimate resource-intensive applications have also actively used CPU resourses making cryptojacking malware detection based only on CPU usage impossible.

The second indicator chosen was the amount of RAM used by application. The experiment showed that Windows executables cryptojackers consume negligible amount of RAM during their work, in contrast to legitimate resource-intensive applications. However, browser-based cryptojackers are utilizing regular browser tab process to perform operations, which also consumes some RAM on its own. So it won't be possible to completely figure out the cryptojacker process based on those two metrics.

The third indicator chosen was the average quadratic deviation of the CPU utilization share.

For the purpose of studying we've chosen 20 browser-based cryptojackers as well as 5 executable-type cryptojackers from VirusShare site. All malware samples were uploaded in 2019. The 4:1 distribution between browser and executable types mimics the real-world one, according to S. Pastrana [10] around 17% of all cryptojacking malware samples are executable ones. All experiments were performed on virtual machine which was allocated 1 processor core with 3GHz frequency and 4 gigabytes of operative memory running Windows 7.

Analysis of experimental data showed that average quadratic deviation of the CPU utilization share for cryptojacking malware is no more than 2.5, while legitimate applications have more dynamic CPU utilization share with quadratic deviation of no less than 5. No browser-based cryptojacker consumed more than 300 megabytes of operative memory and no executable-type consumed more than 50 megabytes. And no cryptojacker used less than 30% of CPU power.

C. Algorithm overview

Based on those experiments we were able to deduce algorithm, as seen on Fig.1, which successfully detected all 25 cryptojacking samples, while providing 0 false positives on tested legitimate resource-intensive applications.

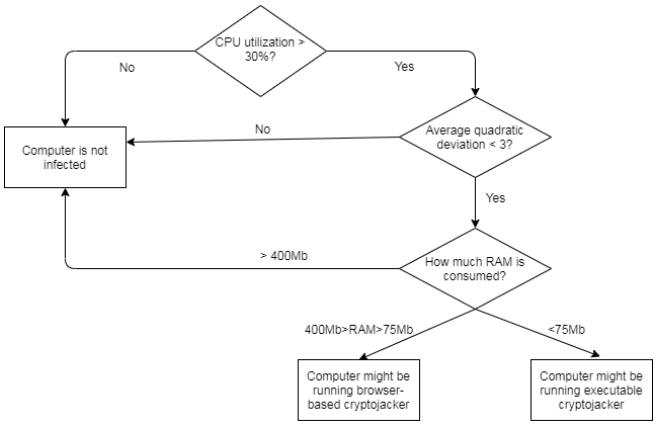

Fig. 1. Cryptojacking malware detection algorithm.

Special cryptojacking detection program was designed based on the experimental data and developed algorithm. It suggests to the user processes which might be mining

cryptocurrencies on his computer – both browser-based and Windows executable. And then it's up to user to make a decision, whether to terminate suspicious process using standart Windows tools or not.

IV. Results and Conclusion

In order to validate our cryptojacker detection program, we've tested it on 50 more cryptojacking malware samples, 40 browser-based and 10 executable-type from VirusShare aswell as some other legitimate applications.

Legitimate applications mostly didn't trigger our detection program with 2 exception types: obviously, our program detected legitimate mining programs, such as MoneroX CPU miner and synthetic CPU tests such as PCMark or CPU Z.

Out of 50 cryptojacking malware samples our program was able to detect 41. 2 out of the 9 undetected have consumed more than 400 megabytes of operative memory and 7 more were dynamically changing CPU load, which made them undetectable by our algorithm. Overall success of our prototype detection program on experimental malware set is 81%, as seen in table 1.

TABLE I. Results for Detection Program

Cryptojacking malware set	Number of samples	Number of successes
Test samples	20	20
Browser-based validation	40	32
Executable-type validation	10	9

Despite the fact that CoinHive was shut down in March 2019, it hasn't brought an end to the cryptojacking attacks – cybercriminals simply switched towards different services such as CryptoLoot and JSECoin with new surge in executable-type cryptojackers, therefore it's still important to keep research towards cryptojacking detection and counteraction. In future works we're planning to expand proposed algorithm to detect cryptojacking malware with dynamic CPU load.

References

[1] M. Szmigiera, "Market capitalization of cryptocurrencies from 2013 to 2019", Available at: https://www.statista.com/statistics/730876/cryptocurrency-maket-value/.

[2] M. Robot, "Rise of the cryptojackers", Available at: https://www.kaspersky.com/blog/cryptojacking-rsa2019/25938/.

[3] B. O'Gorman, "Internet Security Threat Report 2019", Available at: https://www.symantec.com/content/dam/symantec/docs/reports/istr-24-2019-en.pdf.

[4] A. Keeve, "Cryptojacking shows no signs of slowing down in 2019, says ESET", Available at: https://www.eset.com/us/about/newsroom/press-releases/cryptojacking-shows-no-signs-of-slowing-down-in-2019-says-eset/.

[5] J. Zorabedian, "Cryptojacking Rises 450 Percent as Cybercriminals Pivot From Ransomware to Stealthier Attacks", Available at: https://securityintelligence.com/cryptojacking-rises-450-percent-as-cybercriminals-pivot-from-ransomware-to-stealthier-attacks/

[6] S. Eskandari, A. Leoutsarakos, T. Mursch, J. Clark, "A First Look at Browser-Based Cryptojacking", Proceedings of 3rd IEEE European Symposium on Security and Privacy Workshops, April 2018, pp. 58-66.

[7] D. Carlin, P. Orkane, S. Sezer, J. Burgess, "Detecting Cryptomining Using Dynamic Analysis", Proceedings - 16th Annual Conference on Privacy, Security and Trust, August 2018.

[8] W. Wang, B. Ferrell, X. Xu, K.W. Hamlen, S. Hao, "SEISMIC: SEcure in-lined script monitors for interrupting cryptojacks", Computer Security. ESORICS 2018. Lecture Notes in Computer Science, vol 11099, August 2018, pp. 122-142

[9] "About Windows Management Instrumentation", Available at: https://docs.microsoft.com/en-us/windows/win32/wmisdk/about-wmi.

[10] S. Pastrana, G. Suarez-Tangil, "A first look at the crypto-mining malware ecosystem: A decade of unrestricted wealth", Proceedings of the ACM SIGCOMM Internet Measurement Conference, October 2019, pp. 73-86.