Website Cryptojacking Detection Using Machine Learning

IEEE CNS 20 Poster

Venkata Sai Krishna Avinash Nukala
University of Cincinnati
nukalavh@mail.uc.edu

***Abstract*—Cryptojacking is the process of utilizing one's systems resources without their knowledge to mine cryptocurrency. This can be done by injecting a malicious javascript code into the website. The malicious javascript code injected utilizes system's resources in order to compute hashes.**

Most of the classic cryptocurrencies such as bitcoin, monero, webchain are built on proof-of-work(pow) algorithm called *CryptoNight*, which is CPU-bound. They make use of memory-bound functions for constructing computational puzzles [1], in order to maximize profit. This total process requires a lot of disc read and write operations. Hence, we have monitored the cache activity to detect whether cryptojacking exists or not. In addition to detection, our method can also detect the CPU percentage throttle set by the attacker making it a multiclass classification problem. We have leveraged a tool called *'perf'* [2] in order to measure the number of cache hits and misses. Using this methodology, we were able to detect cryptojacking corresponding to each class with an accuracy of 96.25%.

I. Introduction

Cryptocurrency mining is a process of validating transactions of various forms of cryptocurrency and adding them to the block chain ledger. It is the most essential process to keep the transaction chain running. In this process, the miner would be rewarded with some incentive in the form of cryptocurrency for validating the transaction. The amount of profit the miner gets if he uses his own system is not significant for the time he has invested [3], and hence in order to gain more profit the miner must run mining algorithm on multiple systems. This is the origin of the concept called "website cryptojacking." As explained above, cryptojacking is the concept of unauthorized usage of one's system recourses to mine cryptocurrency. The miner injects a malicious javascript code into the website, which when opened by a victim, utilizes the system recourses of the victim's computer without his knowledge. Hence the system is said to be cryptojacked as the victim's system is computing hashes for the miner.

There are many ways of being cryptojacked, through fraudulent email links, apps, browsers, browser extensions [4] etc... but the most popular technique is through cryptojacking a website. Over past few years, there are around 4000 newly created cryptocurrencies known as altcoins [5]. One of the altcoin that was popular for in-browser mining activity from the year 2018 is Webchain and was renamed recently as MintMe(MINTME) uses Cryptonight as its proof of work algorithm. This algorithm requires many disc read and write operations to be performed as it is said to be a memory-hard one. The traditional CryptoNight algorithm was released in 2013 and has three main components named Scratch pad initialization, Memory-hard loop and Final Result Calculation as shown in Fig. 1 [6].

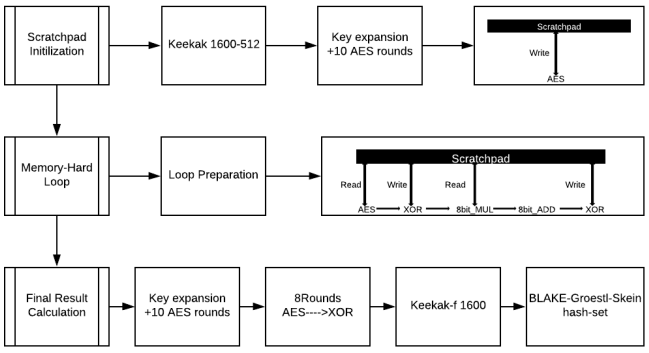

Fig. 1. Main Components of Crypto Night Algorithm [7]

The traditional CPU's are the main targets if the attacker is using this kind of algorithm because they have the desired amount of memory about 2MB readily available on the cache. The advent of Cryptonight algorithm had increased the website cryptojacking attacks as this algorithm can be executed on traditional CPU's which are the majorly available systems across the globe. In order to speed up the mining process, miners gather to form a group called mining pool. They share the profits as per the amount of work done by them. The workload in the pool is distributed as per the difficulty, the miners with high end machines gets difficult puzzles and viceversa.

We have leveraged a tool named *'perf'* developed by Brendan D. Gregg et al. [8] that is used to clearly monitor the cache activity. Perf is one of the powerful tool used to monitor kernel level activities such as CPU performance counters, tracepoints, kprobes etc... Using this tool, we can monitor both software and hardware level features as well.

II. Methodology

Data Collection: We have collected number of cache hits and cache misses for every 100msec interval for 60

978-1-7281-4760-4/20/$31.00 ©2020 IEEE

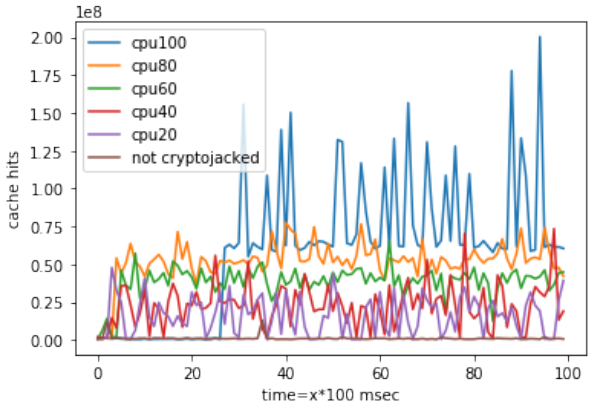

Fig. 2. Number of Cache-Hits

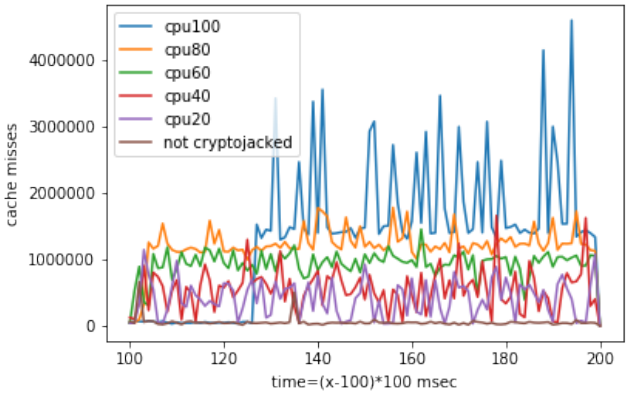

Fig. 3. Number of Cache-Misses

seconds. We have collected 100 traces for each class label enumerating the total number of entries per class to 60000 data points. Fig. 2, 3 shows the plot of cache hits, misses along with the time for each class. We can observe a clear difference between the number of hits and misses when the CPU percentage throttle is varied. The reason behind this is that when attacker set higher throttle, which means that the attacker is utilizing the CPU's maximum capacity for computing hashes and because of which the attacker could compute maximum number of hashes possible. Therefore, the number of hashes computed decreases in proportion with the CPU throttle. This methodology is especially useful in case of code obfuscation. We have conducted all the experiments on an Intel Core I5-7500 machine with Ubuntu 18.04 operating system.

III. Performance Evaluation

We have trained several machine learning models for performing multiclass classification on this time series data. The accuracy, precision, recall, f1-score parameters for each model are shown in Table I. The entire data is split into 70% of total for training and 30% of total for testing purpose.

Table I shows the performance metrics of various machine learning models obtained upon training and testing. The parameters shown in Table I are the best fit parameters obtained after finetuning, for example value of 'k' for k-nearest neighbors and the maximum depth for decision tree. The precision, recall parameters represented are the average values of precisions and recalls for each class. This is done by setting the average parameter of metrics to macro. We can say that our model can detect cryptojacking and to which class it belongs with very good accuracy for majority of the standard models. We got the best accuracy of 96.25% with SVM with precision, recall and F1-score of 0.9637, 0.9638, 0.9632. While decision tree gave lesser accuracy (85.55%) compared to other machine learning models, which is expected, as decision tree is not a best fit for multiclass classification problems. Naïve Bayes and Random Forest classifiers have shown performance on par with support vector machine with corresponding accuracies of 92.77% and 92.78% respectively. K-nn classifier had shown an accuracy of 88.88% which is the best accuracy obtained with a k-value of 7.

TABLE I
PERFORMANCE METRICS OF VARIOUS MACHINE LEARNING MODELS

Model	Accuracy	Precision	Recall	F1 Score
Knn, n=7	88.88%	90.78%	89.23%	90.00%
Random Forest	92.78%	93.29%	93.13%	93.21%
Decision Tree, max_depth=6	85.55%	86.34%	85.84%	86.09%
SVM	96.25%	96.37%	96.38%	96.32%
Naïve Bayes	92.77%	94.27%	92.65%	93.43%

IV. Future Work

We leave the scope of this methodology to future work for some cases like cryptojacking going on when there are other high performance activities such as gaming, etc... going on parallelly in the system, as there may be some more false positive cases involved in such cases.

References

[1] M.Marius, W.Christian, J.Martin and R.Konrad. Web-based Cryptojacking in the wild, *arXiv: 1808.09474v1, 2018.*
[2] [Online] Available: https://perf.wiki.kernel.org/index.php/Tutorial.
[3] G.Ankit, C.Mauro, Detecting Covert Cryptomining using HPC, *arXiv: 1909.00268v1, 2019.*
[4] E.Nicholas, "Cryptojacking-101," Nov, 2017. [Online]. Available: https://blog.authentic8.com/cryptojacking-101/ [Accessed April. 24, 2020].
[5] Hugo L.J.Bijmans, Tim M. Booij, and Christian Doerr, "Inadvertently Making Cyber Criminals Rich: A Comprehensive Study of Cryptojacking Campaigns at Internet Scale," *28th USENIX Security Symposium, Aug, 2019.* Available: https://www.usenix.org/conference/usenixsecurity19/presentation/bijmans.
[6] K.Radhesh Krishnan, V.Emanuele, M.Veelasha, L.Martina, K.Christopher, B.Herbert, V.Giovanni, "MineSweeper: An In-depth Look into Drive-by Cryptocurrency Mining and Its Defense," *CCS'18, Oct, 2018.*
[7] Seigen, M.Jameson, T.Nieminen, Neocortex and Antonio M. Juare, "CryptoNight Hash Function," March, 2013. [Online]. Available: https://cryptonote.org/cns/cns008.txt
[8] Brendan D. Gregg. [Online]. Available: http://www.brendangregg.com/