UNIVERSITÀ
DI TRENTO

AICRYPTOJACKINGTRAP

A MACHINE LEARNING-BASED APPROACH FOR CRYPTOJACKING DETECTION

Thesis Presentation by Lorenzo Masè

Supervisor: Prof. Marco Roveri
Co-supervisor: Ph.D. Atefeh Zareh Chahoki

26/11/2024

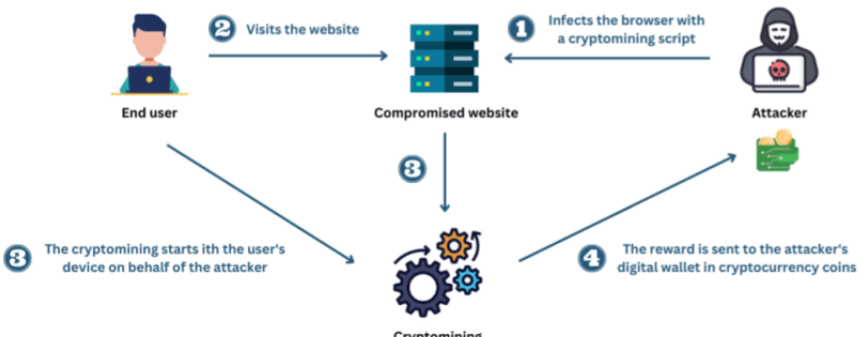

Figure. Monetize the stolen cycles for Botmasters [2]

- Mining: Mining is a computation-intensive process using CPUs or GPUs, to validate transactions and create new blocks in PoW-based blockchain networks like Bitcoin, Monero, and old Ethereum. Miners earn rewards for securing the network.
- Cryptojacking: Malicious behavior of cybercriminals that obtain control of the victim's computer to execute mining on their behalf. It can be executed both using executables injected inside the victim's computer or using JavaScript/HTML code in the browser.

UNIVERSITÀ DI TRENTO

Continent	Cryptojacking attacks in 2021 (millions)	2022 (millions)	Variation 2021-2022 (%)
North America	78.0	105.9	+36%
Asia	3.0	6.9	+129%
Europe	3.4	22.0	+548%

Table. Volumes of cryptojacking attacks for each continent (2021 vs. 2022) [4]

During 2023 cryptojacking attacks were reduced by 60% but for India, the attack's volume grew by 409% [5].

DETECTION

TAXONOMY

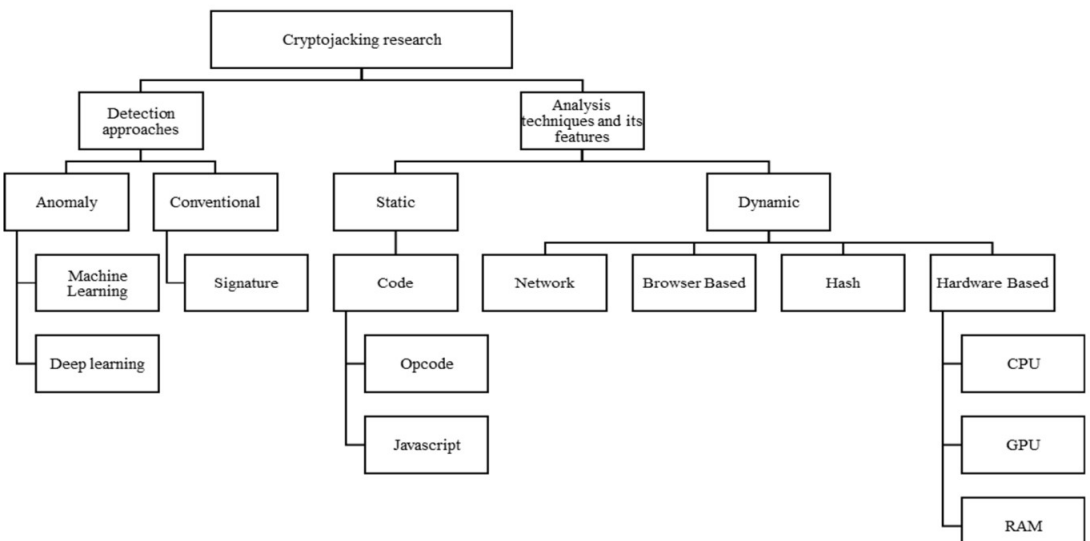

Figure. Taxonomy of cryptojacking [3]

UNIVERSITÀ DI TRENTO

Signatures: These need to be already known to detect the malware that is being injected.
Static Analysis: This can be evaded by using code obfuscation resulting in inefficiency, it can not detect polymorphic or sophisticated malware.
Dynamic Analysis:

- Encryption of network traffic.
- Delayed cryptojacking.
- Low-rate mining.

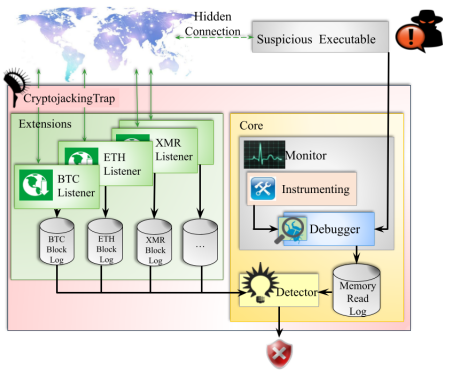

Figure. Architecture of CryptojackingTrap [1]

- Detection based on memory reads made from a suspicious process.
- Comprises a network-linked module (extensions) and a core module (monitor and detector).
- The algorithm checks whether a predictable value (hash), that is mandatory to mine, is retrieved from the memory, by the suspicious process, at different times in a fixed window of time.
- The solution is evasion resilient, but its algorithm has a high computational complexity.

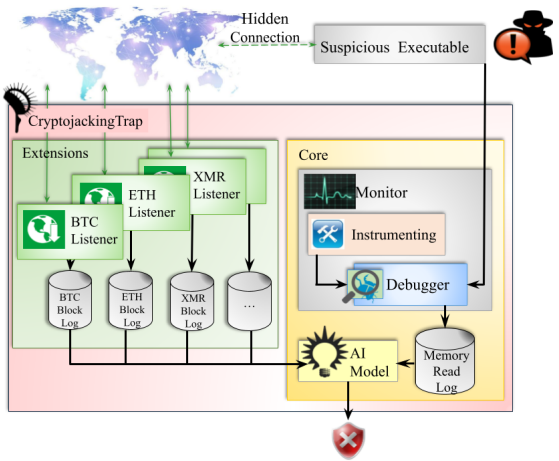

Figure. AICryptojackingTrap architecture

The proposed model's task is to take the place of the detector module of CryptojackingTrap work. Its task is to learn how to *classify* memory read log files that contain the predictable hash at different times, as mining cases. This task can be solved with supervised learning, in conjunction with the data set provided by CryptojackingTrap.

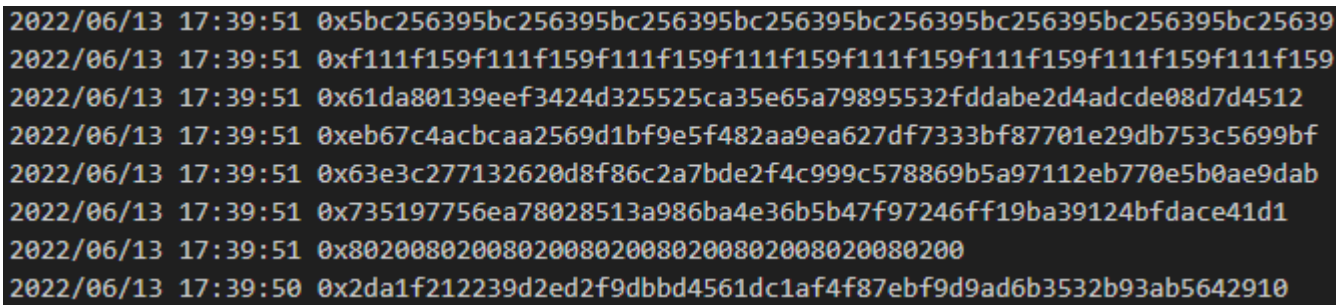

Figure. Example of a fraction of memory read log file

Memory-read log files are created from the CryptojackingTrap monitor module executed on real mining and non-mining applications. These log files contain a date followed by a timestamp for each read and the memory read content retrieved from the application's memory is represented as a hexadecimal number. The initial dataset is formed of different memory read log files of both mining and non-mining processes, each has its related hash value.

PRE-PROCESSING

UNIVERSITÀ DI TRENTO

The initial data set is pre-processed to create a new data set formed of different memory read log files and their hashes, following these steps:

- Data cleaning
- Encoding
- Normalization

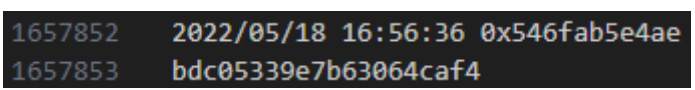

Figure. Erroneous \n inserted

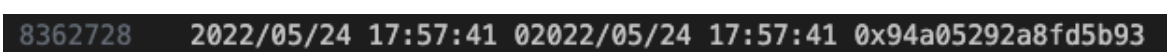

Figure. Multiple date and timestamp in the same line

During this phase, the memory read log files are checked, for possible formatting bugs related to concurrency problems, using regular expressions.

UNIVERSITÀ DI TRENTO

In this phase, the date and timestamp are removed from the lines. The proposed solution uses a byte encoding for both the hash value and the memory read log file payloads. That is, each two characters represented in hexadecimal form is represented by an integer. This technique is combined with a sliding-window algorithm to strengthen the learning.

$$(1A)_{16} = (1 \times 16^1) + (10 \times 16^0) = (26)_{10}$$

Example of encoding of two characters from hexadecimal to an integer

UNIVERSITÀ DI TRENTO

During this phase, the data set is normalized through a StandardScaler which computes the standard score for each sample x as:

$$z = (x - u)/s$$

- u is the mean of the training samples.
- s is the standard deviation of the training samples.

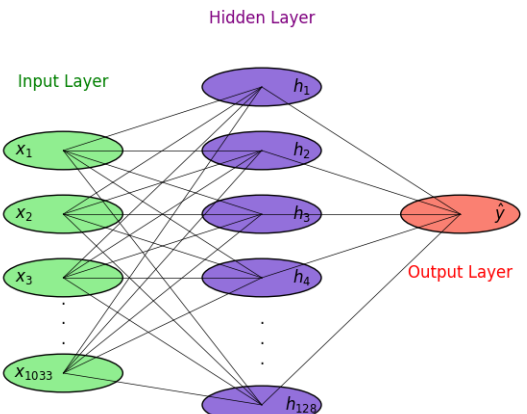

Figure. Neural Network created

- Developed with scikit-learn
- Layers: 'Dense'
- Activation function:'Sigmoid'
- Optimizer: 'Adam'

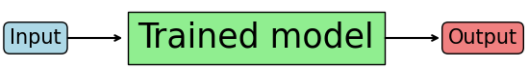

Figure. Testing execution

- Input: Two different datasets, one with miner data and another with non-miner data, were created through the same process for the training dataset.
- Trained model: Model after the execution of 25 epochs of the dataset
- Output: 1 or 0, 1 meaning mining activity and 0 meaning non mining activity

RESULTS

UNIVERSITÀ DI TRENTO

Phase	Accuracy	Loss
Training	54%	69%
Evaluation on non-mining data	100%	55%
Evaluation on mining data	0%	87%

Table. Training and evaluation results of the model

Conclusions and future works

The model did not understand the relation between the hash values and the log files used for the data set, this is mainly related to the fact that the memory read log files contain a lot of non-relevant data. This is the first research on the machine learning approach for memory read-based detection. The results and challenges overcome during this study will be the basis for further research on new solutions. Future works:

- New possible solutions using deep learning
- New approaches to the task

REFERENCES

[1] Atefeh Zareh Chahoki, Hamid Reza Shahriari, and Marco Roveri. **"CryptojackingTrap: An Evasion Resilient Nature-Inspired Algorithm to Detect Cryptojacking Malware".** In: *IEEE Transactions on Information Forensics and Security* (2024), pp. 1–1. DOI: 10.1109/TIFS.2024.3353072.

[2] datacamp. ***Monetize the stolen cycles for Botmasters*.** https://www.baeldung.com/cs/cryptojacking-attacks. 2024. (Visited on 11/25/2024).

[3] Laith M Kadhum et al. **"Features, Analysis Techniques, and Detection Methods of Cryptojacking Malware: A Survey".** In: *JOIV: International Journal on Informatics Visualization* 8.2 (2024), pp. 891–896.

[4] SonicWall. ***2023 SonicWall Cyber Threat Report*.** Tech. rep. SonicWall, 2023.

[5] SonicWall. ***2024 SonicWall Mid-Year Cyber Threat Report*.** Tech. rep. SonicWall, 2024.