Cryptomining cannot Change its Spots: Detecting Covert Cryptomining using Magnetic Side-channel

Ankit Gangwal and Mauro Conti, *Senior Member, IEEE*

***Abstract*—With new cryptocurrencies being frequently introduced to the market, the demand for cryptomining - a fundamental operation associated with most of the cryptocurrencies - has initiated a new stream of earning financial gains. The cost associated with the lucrative cryptomining has driven general masses to unethically mine cryptocurrencies using "plundered" resources in the public organizations (*e.g.,* universities) as well as in the corporate sector that follows Bring Your Own Device (BYOD) culture. Such exploitation of the resources causes financial detriment to the affected organizations, which often discover the abuse when the damage has already been done.**

In this paper, we present a novel approach that leverages magnetic side-channel to detect covert cryptomining. Our proposed approach works even when the examiner does not have login-access or root-privileges on the suspect device. It merely requires the physical proximity of the examiner and a magnetic sensor, which is often available on smartphones. The fundamental idea of our approach is to profile the magnetic field emission of a processor for the set of available mining algorithms. We built a complete implementation of our system using advanced machine learning techniques. In our experiments, we included all the cryptocurrencies supported by the top-10 mining pools, which collectively comprise the largest share (84% during Q3 2018) of the cryptomining market. Moreover, we tested our methodology primarily on two different laptops. By using the data recorded from the magnetometer of an ordinary smartphone, our classifier achieved an average precision of over 88% and an average F_1 score of 87%. Apart from our primary goal - which is to identify covert cryptomining - we also performed four additional experiments to further evaluate our approach. We found that due to its underlying design, our system is future-ready and can readily adapt even to zero-day cryptocurrencies.

***Index Terms*—Altcoin, Bitcoin, Cryptocurrency, Detection, Machine learning, Mining.**

I. Introduction

As opposed to the fiat currencies - which are typically issued by centralized financial institutions - cryptocurrencies are decentralized and generally rely on distributed public ledgers, in which transactions are validated by a process called cryptomining, or simply mining. Essentially, mining is a process of contributing computational resources to perform heavy computations, which continuously consumes electricity. For the effort of mining, successful miners receive newly generated cryptocoins as a reward. Numerous cryptocurrencies have emerged after the success of Bitcoin, and thus, the possibilities of mining and earning incentives. Cryptocurrencies such as Application-Specific Integrated Circuit (ASIC) resistant Monero enables mining feasible on web-browsers, which allows even naive users to mine.

Motivation: Apart from investment in hardware, the cost of electricity to power up mining hardware and cooling facilities is one of the significant expenses associated with mining [1]. Mining popular cryptocurrencies, such as Bitcoin, is not profitable using personal resources (mainly electricity) unless the mining is done using specialized hardware [2]. However, mining can be lucrative if it is done using "plundered" resources, *e.g.,* exploiting infrastructure at workplace. We can broadly categorize such plundering into two classes: conscious-mining and unconscious-mining. Conscious-miners exploit infrastructure allocated to them, *e.g.,* an unethical employee who mines at the workplace. On the other side, unconscious-miners mine unknowingly for a third party, *e.g.,* the visitors to a website that hosts cryptojacking scripts. In the market of cryptocurrencies, where new currencies emerge every day, miners have a wide variety of options to choose from. Thus, a solution to detect covert cryptomining that focuses on a particular cryptocurrency may not be adequate to tackle such an uncertain situation. Besides, unethical mining has lured all members of the modern society: government employees [3], corporate employees [4], students [5], teachers [6], researchers [7], nuclear scientists [8], and undoubtedly, hackers [9, 10]. Furthermore, extensive mining on an incompatible device may also damage the hardware of the device [11]. To summarize, we can say that covert cryptomining has financial consequences, such as monetary losses, as well as societal influences such as ethical and psychological impacts.

Contribution: The major contributions of this paper are listed as follows:

1) We propose a novel approach that leverages the magnetic side-channel to detect covert cryptomining. We target the core of the mining process, *i.e.,* the mining algorithms, and thus, the scope of our approach is broader.
2) We implemented our approach and built a complete system. We included twelve different cryptocurrencies in our experiments, which indeed are the most mined cryptocurrencies.
3) We designed and performed five different experiments to throughly assess the quality of our proposed approach.

Organization: The remainder of this paper is organized as follows. We discuss the essential concepts related to our work in Section II. Section III provides a comparative summary of the related works. We elucidate our system's architecture

Ankit Gangwal and Mauro Conti are with the Department of Mathematics, University of Padua, 35121, Italy. E-mail: ankit.gangwal@phd.unipd.it; conti@math.unipd.it.

Manuscript received MONTH DD, YYYY; revised MONTH DD, YYYY, MONTH DD, YYYY, and MONTH DD, YYYY; accepted MONTH DD, YYYY.

Digital Object Identifier XX.XXXX/TIFS.YYYY.XXXXXXX

in Section IV and its evaluation in Section V. Section VI addresses the potential limitations of our proposed approach. Finally, Section VII concludes the paper.

II. Preliminaries

In this section, we briefly explain the fundamental concepts related to our work. Section II-A introduces the magnetic field, Section II-B describes the magnetic field sensor of the smartphones, and Section II-C elucidates dynamic time warping that serves as the similarity measure for our classifier.

A. Magnetic field

The magnetic field at any point in space is represented by a vector quantity. It is specified by its magnitude as well as direction and measured in Tesla (T). In practice, magnetic fields are measured in the unit of millitesla (mT) or microtesla (μT). Electric current produces a magnetic field. The total magnetic field generated around an enclosed path is directly proportional to the current which passes through that path [12, 13]. In standard computers, the CPU is one of the biggest consumers of the electricity. The power consumption of modern energy efficient CPUs is dynamically affected by the workload [14]. In the most fundamental case, overloading the CPU with computations will draw more current, and hence, will generate a stronger magnetic field. To demonstrate this effect, we created a script that recursively engages all available CPUs for 2 seconds and then sleeps for 2 seconds. Fig. 1 shows the CPU usage and the generated magnetic field while executing this script.

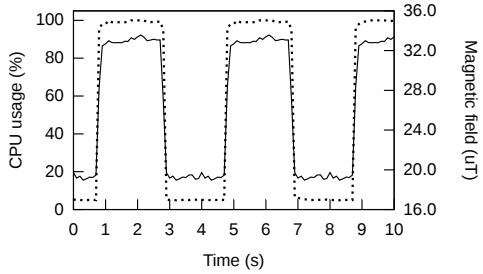


Fig. 1. The CPU usage (dashed line) and the generated magnetic field (plain line) while executing a script that recursively uses all available CPUs for 2 seconds and then sleeps for 2 seconds

It is clear from Fig. 1 that the magnetic field around a processor is directly affected by the workload of the processor. Hence, this side-channel can be exploited to infer a processor's activities.

B. Magnetic field sensor of the smartphones

Nowadays, smartphones are equipped with a variety of sensors. One of these sensors is the magnetic field sensor, which typically measures the strength (magnitude and direction) of the magnetic field along the three physical (x, y, z) axes. Using the three dimensional data from the magnetic sensor, we can calculate the total magnetic field (M_{total}) as

$$M_{total} = \sqrt{M_x^2 + M_y^2 + M_z^2}, \tag{1}$$

where M_x, M_y, and M_z represent the strength of the magnetic field along x, y, and z axis respectively.

C. Dynamic Time Warping

Dynamic Time Warping (DTW) is a widely adopted approach to find the optimal non-linear alignment between two time series that may vary in time as well as speed. DTW is also used as a distance measure to find similarity between time series [15]. Let us consider two discrete time series: $Q = (q_1, q_2, \ldots, q_i, \ldots, q_n)$ of length $n \in \mathbb{N}$ and $C = (c_1, c_2, \ldots, c_j, \ldots, c_m)$ of length $m \in \mathbb{N}$. DTW uses an $n \times m$ matrix, whose $(i,j)^{th}$ element represents the distance $d(q_i, c_j)$ between q_i and c_j (*i.e.,* $d(q_i, c_j) = (q_i - c_j)^2$). Each matrix element (i, j) corresponds to the alignment between the points q_i and c_j. A warping path $W = (w_1, w_2, \ldots, w_k, \ldots, w_K)$ is a contiguous set of matrix elements that defines a mapping between Q and C. The k^{th} element of W is defined as $w_k = (i,j)_k$ for $max(m,n) \leq K < m + n - 1$. The warping path is typically subject to the following constraints:

1) Boundary condition: $w_1 = (1,1)$ and $w_K = (m,n)$.
2) Continuity: Given $w_k = (a,b)$, then $w_{k-1} = (a',b')$, where $a - a' \leq 1$ and $b - b' \leq 1$.
3) Monotonicity: Given $w_k = (a,b)$, then $w_{k-1} = (a',b')$, where $a - a' \geq 0$ and $b - b' \geq 0$.

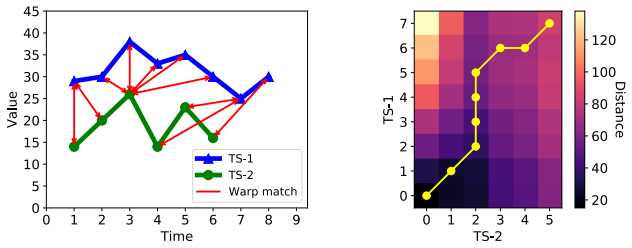
There could be several warping paths that satisfy the conditions reported above. However, our goal is to find the path that minimizes the warping cost:

$$DTW(Q,C) = min\left(\sqrt{\sum_{k=1}^{K} w_k}\right). \tag{2}$$

The optimal path can be found by computing the cumulative distance $\gamma(i,j)$ using the following recursive function:

$$\gamma(i,j) = d(q_i, c_j) + min(\gamma(i-1, j-1), \gamma(i-1, j), \gamma(i, j-1)). \tag{3}$$

Fig. 2(a) shows the alignment between two discrete time series TS-1 and TS-2. The arrows indicate the points that are matched by the DTW algorithm. Fig. 2(b) depicts the heat-matrix for these two time series. The color of a cell (i, j) depicts the minimum distances to reach the cell (i, j) from the cell (0, 0). The optimal warping path has been highlighted using a yellow line that starts from the cell (0, 0), which also satisfies boundary, continuity, and monotonicity constraints mentioned above.



(a) Alignment between TS-1 and TS-2 (b) Illustration of the optimal warping path

Fig. 2. A representative example of DTW algorithm applied to two discrete time series: TS-1 and TS-2

Standard DTW is optimal, but it is computationally expensive, both in space and time complexity. Standard DTW has a complexity of $O(nm)$ where n and m are the lengths of the first and second time series respectively. The performance of DTW can be improved by restricting the amount of warping allowed, which typically limits the number of cells computed in the DTW distance matrix. There are several proposals that claim to improve DTW such as PrunedDTW [16], SparseDTW [17], FastDTW [18], and MultiscaleDTW [19, 20]. These approaches [16–20] propose ways to speed-up exact DTW computation for two time series. Next, LB_Keogh lower bound [21] is a technique that allows efficient indexing and comparison of many time series via lower-bound approximation (less than exact DTW value). Moreover, Wang *et al.* [22] found that except for LB_Keogh lower bound [21], all other techniques are inefficient. Hence, in this work, we use LB_Keogh lower bound to speed up DTW, which is computed as shown in Eq. (4):

$$LB\_Keogh(Q,C) = \sqrt{\sum_{i=1}^{n} \begin{cases} (c_i - U_i)^2, & \text{if } c_i > U_i; \\ (c_i - L_i)^2, & \text{if } c_i < L_i; \\ 0, & otherwise. \end{cases}} \quad (4)$$

Here, L_i and U_i are lower and upper bounds for time series Q, which are defined as $L_i = min(q_{i-r} : q_{i+r})$ and $U_i = max(q_{i-r} : q_{i+r})$ for a reach r. LB_Keogh lower bound method has linear complexity, which makes it very useful for searching over large sets of time series.

III. Related works

The side-channel information leakage has been thoroughly studied over the previous years. Several research works focused on various side-channels to leak information from different type of systems. We will primarily discuss previous studies more relevant to our work.

Quisquater *et al.* [23] show that the electromagnetic attack on processors can reveal as much information as power consumption-based side-channel analysis. Mateos *et al.* [24] use specialized magnetic sensors to recover a secret key. Song *et al.* [25] investigate acoustic and magnetic side-channel attacks on 3D printers using smartphone's built-in sensors. Biedermann *et al.* [26] use the smartphone's magnetic sensors to fingerprint hard-drives. The authors are able to deduce ongoing system activities based on the emitted magnetic field due to movements of the hard drive's magnetic head. However, their methodology is inapplicable to the modern computers equipped with Solid State Drive (SSD) storage. Similarly, Matyunin *et al.* [27] establish a covert channel using the magnetic field between a laptop that is infected with a special program and a smartphone equipped with a magnetic sensor. The authors claim that their approach can decode the emanated signal up to a distance of roughly 12 cm from the laptop. ODINI [28] and MAGNETO [29] employ malware to control workload on the CPU, which, in turn, regulates the magnetic fields emitting from the target device. ODINI uses a high-precision magnetic sensor to receive data up to 150 cm while MAGNETO can receive data up to a maximum distance of 12 cm using a regular smartphone.

The literature on detecting the mining of cryptocurrencies is rather limited. Bonneau *et al.* [30] present a systematic study of various cryptocurrencies and discuss open research challenges. Huang *et al.* [31] in their analysis of Bitcoin mining malware have shown that modern botnets generate additional revenue via illicit mining. Eskandari *et al.* [32] present a security analysis of in-browser mining of cryptocurrencies. Recent works [33–37] focus specifically on browser-based mining. However, only a small subset of cryptocurrencies supports in-browser mining. MineGuard [38] focuses on mining operations in the cloud infrastructure. The authors utilize privileged hardware performance counters to fingerprint and detect covert mining.

Our work is different from the state-of-the-art on various dimensions: (1) Our proposed approach is a generalized approach that applies to all forms of cryptomining on computers; (2) Our approach does not require any specialized equipment, malware/software installation, user-/root-privileges, or even login-access to the monitored system. The only requirements are the physical proximity of the investigator and a magnetic sensor; (3) Our methodology does not impart any performance overhead on the monitored system; and (4) Our study includes all the cryptocurrencies supported by the top-10 mining pools, which covers the largest share of the market of cryptomining.

IV. System architecture

We explain the foundation of our work in Section IV-A, our data collection strategy in Section IV-B, our decision for selecting cryptocurrencies in Section IV-C, and the design of our classifier in Section IV-D.

A. Core concept

The task of mining is to execute a core Proof-of-Work (PoW[1]) algorithm repeatedly, which means that a robust signature can be constructed for a particular algorithm. Interestingly, there are a limited number of PoW algorithms. Therefore, we focus on the mining algorithms. The major benefit of our strategy is that the signature constructed for an algorithm would be able to detect even metamorphic and polymorphic implementations of that algorithm used by other cryptocurrencies. Notably, such signature-based detection of cryptomining can be partially deceived by restricted mining. But, it would directly affect the hashing rate and consequently the profits; making the task of mining less appealing.

Previous works [26–29] have shown the effectiveness of the magnetic side-channel to exfiltrate information from computers. We designed our system for detecting and classifying covert cryptomining using the magnetic side-channel for the following reasons: (1) The examiner may not have login-access[2] or root-privileges[3] on the device; (2) As discussed in Section II-A, magnetic field emission is a fundamental phenomenon associated with electronic circuits and it can

[1] We use "PoW" as the representative of various consensus algorithms.

[2] *E.g.,* an administrator in BYOD culture, who suspects an employee's machine.

[3] *E.g.,* an employee, who suspects an infection in the company-provided machine.

even penetrate air-gaps and a faraday-cage; and (3) Since the miners have to adhere to and repeatedly execute the core PoW algorithm used by a cryptocurrency, the pattern of the magnetic field emitted while mining a cryptocurrency is distinct as well as consistent. For the same reason, even a smaller signature database is sufficient for reasonable classification results.

B. Dataset collection

To better elucidate our work and to maintain the flow of reading, we first explain the data collection stage. We used two different systems in our experiments. These systems have the following configuration: (1) *S1*, a laptop with an Intel Core i5-7200U @ 2.50 GHz (1 socket x 2 cores x 2 threads = 4 logical compute resources) processor, 8 GB memory, 256 GB SSD storage, and Intel HD Graphics 620 mounted on Dell Inc. 0M60Y2 motherboard with Ubuntu 16.04 as the operating system; and (2) *S2*, a laptop with an Intel Core i7-8550U @ 1.80 GHz (1 socket x 2 cores x 4 threads = 8 logical compute resources) processor, 16 GB memory, 512 GB SSD storage, and Intel UHD Graphics 620 mounted on Dell Inc. 02PG84 motherboard with Ubuntu 18.04 as the operating system. To show the effectiveness of the proposed method, we used an ordinary smartphone (Samsung Galaxy S5 running Android 6.0.1), hereinafter referred to as *probe* device, to record the generated magnetic field. Fig. 3 depicts a representative demonstration of data collection on *S1* using the probe device. Section VI-B presents a detailed discussion on probe's orientation and position.

Fig. 3. A representative demonstration of data collection on *S1* using the probe device

We mined and profiled each cryptocurrency (discussed in Section IV-C) individually and collected a total of thirty samples per cryptocurrency per system, where each sample comprises measurements taken over a time interval of thirty seconds. The sampling rate of the probe's magnetic sensor was 10Hz. To obtain clean signatures of the core PoW algorithms, we profiled the miners in their stable stage, *i.e.,* omitting the bootstrapping phase. As representatives of non-mining tasks (negative-class), we chose: Internet browsing; PDF document scrolling; *Skype* test call; 3D benchmarking; solving N-queens problem (N=18); 4K video streaming; H.264 video encoding; network performance test using *iperf* tool; machine learning with *scikit-learn*; deep learning with *TensorFlow*; *stress-ng* [39] stress test with CPU-only workers; and *stress-ng* stress test with CPU, memory, I/O, and disk workers together. It is worth mentioning that these user-tasks represent low to high compute-intensive tasks and all belong to the same category (*i.e.,* non-mining) for classification purposes. Similar to the mining tasks, we profiled each non-mining task for the same time interval as well as number of samples.

Before any experiment was performed, we estimated the background magnetic field to calibrate subsequent measurements (triplets). We took 100 measurements (10sec @ 10Hz) along the three (x, y, z) axes and calculated the mean ($\bar{M}_x$, $\bar{M}_y$, $\bar{M}_z$) of the background noise. We calibrated each measurement, collected during the experiments, by eliminating the mean background noise from it. Finally, we computed M_{total} for each calibrated measurement using Eq. (1).

C. Cryptocurrencies and miners

In the context of cryptocurrency mining, miners pool their resources so they can generate blocks more quickly, and therefore earn a portion of the block reward on a consistent basis. Mining pools are characterized by their hashing power. TABLE I lists the cryptocurrencies supported by the top-10 mining pools [40], which collectively comprise the largest share (84% during Q3 2018) of the cryptomining market. See TABLE A.1 (in Appendix A) for acronyms and their corresponding cryptocurrency.

We included all the cryptocurrencies supported by these mining pools in our experiments. Additionally, we included QRK whose mining algorithm - unlike other currencies - consists of a combination of different hashing algorithms. To mine these currencies, we used open-source miner programs that are readily available online. Each miner was configured

TABLE I
CRYPTOCURRENCIES SUPPORTED BY THE TOP-10 MINING POOLS

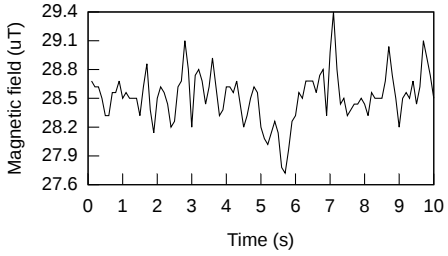
N.	Mining pool	Cryptocurrency															
		BCD	BCH	BTC	BTM	DASH	DCR	ETC	ETH	LTC	SBTC	SC	UBTC	XMC	XMR	XZC	ZEC
1	BTC.com	✗	✓	✓	✗	✗	✗	✗	✗	✗	✓	✗	✓	✗	✗	✗	✗
2	AntPool	✗	✓	✓	✓	✓	✗	✓	✓	✓	✗	✓	✗	✓	✗	✗	✓
3	ViaBTC	✗	✓	✓	✗	✓	✗	✓	✓	✓	✗	✗	✗	✗	✗	✗	✓
4	SlushPool	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓
5	F2Pool	✗	✗	✓	✗	✓	✓	✓	✓	✓	✗	✓	✗	✓	✓	✓	✓
6	BTC.top	✗	✓	✓	✗	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗
7	Bitclub.network	✗	✗	✓	✗	✓	✗	✓	✓	✗	✗	✗	✗	✗	✓	✗	✓
8	BTCC	✓	✓	✓	✓	✗	✗	✗	✗	✓	✓	✗	✗	✗	✗	✗	✗
9	BitFury	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
10	BW.com	✗	✗	✓	✗	✗	✗	✓	✓	✓	✗	✓	✓	✗	✗	✗	✗

to mine with public mining pools and to use all available CPUs on the machine, *i.e.,* 4 logical CPUs. TABLE II lists the mining algorithm of different cryptocurrencies as well as the CPU miners that we used to mine them.

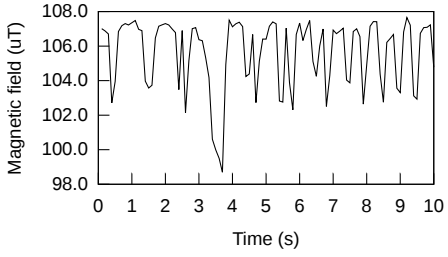
TABLE II
MINING ALGORITHM AND CPU MINER FOR DIFFERENT CRYPTOCURRENCIES

Cryptocurrency	Mining algorithm	CPU miner
BCD	X13	cpuminer-opt 3.8.8.1
BCH, BTC, SBTC, UBTC	SHA-256	cpuminer-multi 1.3.4
BTM	Tensority	bytom-wallet-desktop 1.0.2
DASH	X11	cpuminer-multi 1.3.4
DCR	Blake256-r14	cpuminer-multi 1.3.4
ETC, ETH	Ethash (Modified Dagger-Hashimoto)	geth 1.7.3
LTC	scrypt	cpuminer-multi 1.3.4
QRK	BLAKE + Grøstl + Blue Midnight Wish + JH + Keccak (SHA-3) + Skein	cpuminer-multi 1.3.4
SC	BLAKE2b	gominer 0.6
XMC, XMR	CryptoNight	cpuminer-multi 1.3.4
XZC	Lyra2z	cpuminer-opt 3.8.8.1
ZEC	Equihash	Nicehash nheqminer 0.3a

As explained in Section IV-A, different cryptocurrencies that employ the same mining algorithm exhibit the same signature. Therefore, it is sufficient to consider one currency for each mining algorithm listed in TABLE II. We excluded BCH, SBTC, UBTC, ETC, and XMC. In our study, we used only CPU-based miners as the proof-of-concept implementation. Nevertheless, our approach is also valid to distinguish GPU-based miners because a GPU operates differently than a CPU. For the same reason, GPU-based mining generates a distinct magnetic field in terms of magnitude as well as form. As a representative example, Fig. 4 shows the generated magnetic field while mining XMR on CPU and on GPU[4].



(a) On CPU



(b) On GPU

Fig. 4. Generated magnetic field while mining XMR

[4]Using ccminer v2.3 on 4 GB NVIDIA GeForce GTX 960M.

D. Classifier design

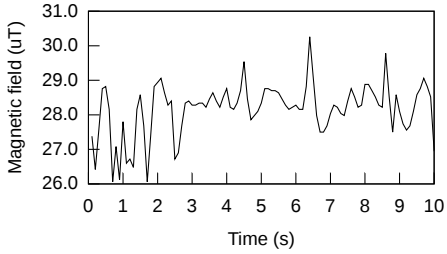
The data for the generated magnetic field can be represented as a time series. Thus, our problem converts to time series classification. In our scenario, we can identify the following two main objectives of the classification stage:

- Classify whether a given instance represents the mining activity or not;
- If so, then predict the specific currency (algorithm).

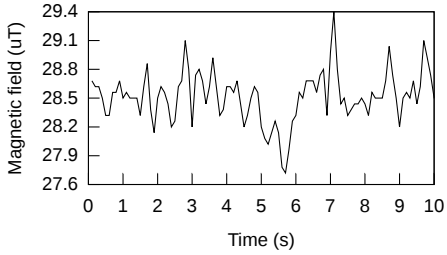
We designed our classifier to suffice these objectives. Now, we discuss our data preprocessing, machine learning model selection, training, and prediction phase.

1) Data preprocessing: The results of time series classification are affected by the quality of input data. All the time series data in our dataset are of equal length. Before starting the training, we employ a scaling function to normalize the input data. In particular, following the suggestion from the work [41], we use the Z-normalization technique. See Appendix B for further details on the scaling technique. Next, we *smooth* [42] the input data to remove noise.

2) Machine learning: Fig. 5 depicts the generated magnetic field while mining BTC and XMR. The time series graphs shown in Fig. 5 are different from each other because each of these cryptocurrencies uses a distinct PoW algorithm that performs a discrete task and has a different iteration cycle.



(a) BTC



(b) XMR

Fig. 5. A representative example of the generated magnetic field while mining BTC and XMR

We use the K-Nearest Neighbors (KNN) algorithm for the classification of our time series consisting of the values for M_{total} at each measurement, where DTW distance serves as the similarity measure. In particular, we use the KNN classifier with K = 1 because previous studies on the classification of time series data demonstrated that DTW-based 1NN classifier - which selects the first nearest neighbors - is "the best" [43, 44], "Nearest Neighbor DTW is very hard to beat" [45], and "1NN with DTW is exceptionally hard to beat" [46]. Nevertheless, in our experiments, we also performed stratified 4-fold cross-validation on training data - which we obtain from 80-20%

stratified partitioning of our dataset into training-test split - and we observed least error rate for K=1 among all the single digit odd values of K.

3) Training and prediction: Our classification model is instance-based. For every instance in the test-set, a search is performed through all the instances in the training-set to find the most similar time series. Given the quadratic complexity of DTW, we use LB_Keogh lower bound (see Section II-C) to speed up the classification stage. Given a new instance to classify, the prediction is made for both the objectives discussed at the beginning of this section.

V. Evaluation

Here, we describe the evaluation procedure used to thoroughly assess the quality of our proposed approach. For the objectives of the classification stage, mentioned in Section IV-D, we performed the following five experiments: (1) *binary* classification; (2) *currency* classification; (3) *full-stack* classification; (4) *unseen-miner-programs* classification; and (5) *cross-platform* classification. In order to increase the statistical significance of the results, we repeated each experiment five times with stratified 80-20% training-test partitioning. It is worth to state that even though the dataset has been collected in a controlled setup, our experiments fairly simulate the real-world scenario, where samples are gathered in real-time. TABLE III describes the sample distribution in our dataset for each system, *i.e., S1* and *S2*. Here, sub-classes of the mining task refer to the cryptocurrencies (discussed in Section IV-C) while sub-classes of the non-mining task refer to the actual user-tasks that belong to the negative class (mentioned in Section IV-B). It is important to mention that we created a single training-set where we kept the instances from both *S1* and *S2* together.

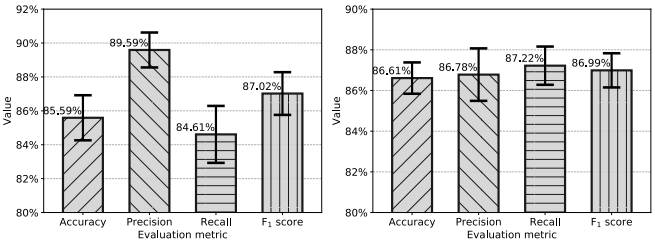
TABLE III
Dataset: name of the task, sub-classes per task, samples per sub-class, and total samples per task for each system

Task	Sub-classes per task	Samples per sub-class	Total samples per task
Mining	12	30	360
Non-mining	12	30	360

We evaluated our classifier using standard classification metrics: Accuracy, Precision, Recall, and F_1 score. For the statistical certitude of our results, we report the mean and the margin of error for the results with 95% confidence interval from five runs of each experiment for each of the evaluation metric. See Appendix B for details on the evaluation metrics and the related statistical terms. We use the notation $mean \pm margin\_of\_error$ to report our results.

A. Binary classification

In this setting, we consider our classification problem as a binary classification task for Mining (positive) class and non-mining (negative) class. All the instances of various cryptocurrencies are treated as the positive-class while all the instances of non-mining user-tasks fall in the negative-class. This assessment aims to evaluate our classifier's ability to detect the presence of cryptomining activities. Fig. 6 presents the results of the *binary* classification. Fig. 6(a) and Fig. 6(b) correspond to *S1* and *S2*, respectively.



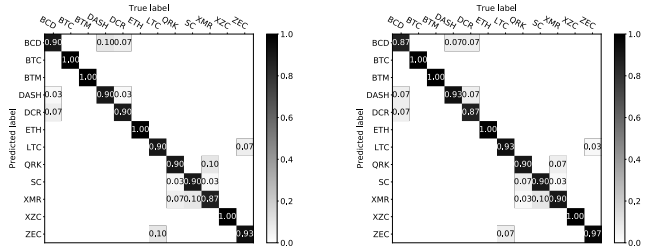
(a) *S1* (b) *S2*

Fig. 6. Results of the *binary* classification (whiskers represent the margin of error)

On *S1*, we achieved an average accuracy of 85.59%±1.33%, precision of 89.59%±1.03%, recall of 84.61%±1.68%, and F_1 score of 87.02%±1.26% while on *S2*, we attained an average accuracy of 86.61%±0.77%, precision of 86.78%±1.29%, recall of 87.22%±0.94%, and F_1 score of 86.99%±0.84%.

B. Currency classification

We designed this experiment to comprehend the level of difficulty in distinguishing various cryptocurrencies. Hence, the input data for this experiment comprised of instances belonging only to cryptocurrencies. Fig. 7 showcases the confusion matrix for classification among various cryptocurrencies. We drew the confusion matrices using the aggregate results from all the five runs. Fig. 7(a) and Fig. 7(b) correspond to *S1* and *S2*, respectively.



(a) *S1* (b) *S2*

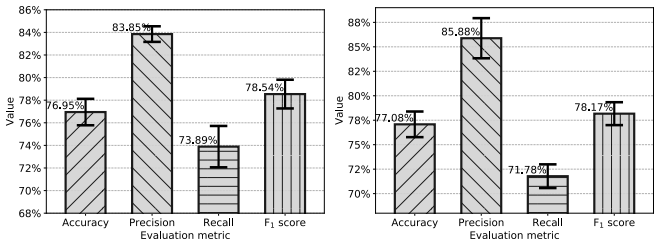
Fig. 7. Confusion matrix for *currency* classification

On both *S1* and *S2*, our classifier achieved an overall-average performance of over 93% for each of the evaluation metric. Furthermore, the results from this assessment also help us to better understand the outcomes of *full-stack* classification, which is discussed next.

C. Full-stack classification

This assessment aims to evaluate our classifier's ability to fulfill both our classification objectives, *i.e.,* first, identify whether the given instance represents a mining activity, and if so, then predict the specific currency. It is worth to mention

that an error in the primary stage of the full-stack classification influences the subsequent stage. Furthermore, given that our classifier makes a correct decision in the primary stage, the difficulty level of the subsequent stage (*i.e.,* classification among various cryptocurrencies, discussed in *currency* classification) affects the final results. Fig. 8 depicts the results of the *full-stack* classification. Fig. 8(a) and Fig. 8(b) correspond to *S1* and *S2*, respectively.



(a) *S1* (b) *S2*

Fig. 8. Results of the *full-stack* classification (whiskers represent the margin of error)

On *S1*, we attained an average accuracy of 76.95%±1.17%, precision of 83.85%±0.69%, recall of 73.89%±1.83%, and F_1 score of 78.54%±1.27% while on *S2*, we achieved an average accuracy of 77.08%±1.31%, precision of 85.88%±2.05%, recall of 71.78%±1.21%, and F_1 score of 78.17%±1.17%. Given the lenient requirements (mentioned in Section III) of our methodology, we believe that the results of the *full-stack* classification are justifiable. Nevertheless, our primary aim is to identify the presence of the covert cryptomining, for which, our *binary* classification has manifested promising results.

D. Unseen-miner-programs classification

Since there can be more than one miner programs for a cryptocurrency and training a classifier on every miner program might not be possible. Therefore, we designed this experiment to evaluate the proficiency of our approach in such circumstances. The goal of this experiment was to perform the *binary* classification of all mining and non-mining samples, as mentioned in Section V-A. However, we selected two additional miner programs for BTC, namely, BFGMiner 5.5 and cgminer 4.10. During the training, the classifier was exposed to samples from one of the three miner-programs for BTC. In contrast, during the testing phase, samples from one of the remaining two miner-programs for BTC were used. TABLE IV reports our results of classifying samples from the miner programs that were not seen in training.

The notation M_N means that for BTC, the classifier was trained with samples from M while samples from N were used for testing. Here, α = cpuminer-multi 1.3.4, β = BFGMiner 5.5, γ = cgminer 4.10. It is important to mention that even though we performed the classification with all the mining and non-mining sub-classes, TABLE IV presents the results only for BTC mining to preserve the goal of this experiment.

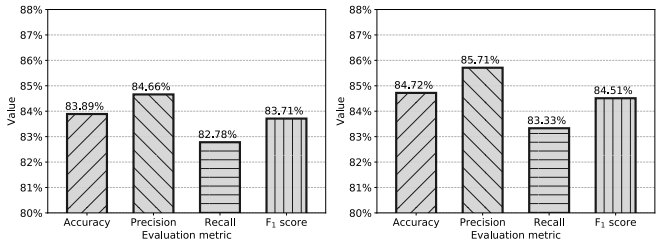
As mentioned in Section IV-A, the miners have to adhere to the core PoW algorithm used by a cryptocurrency. Our results presented in TABLE IV support our notion that the pattern of the magnetic field emitted while mining a given cryptocurrency is consistent across different miner programs.

TABLE IV
RESULTS OF THE *unseen-miner-programs* CLASSIFICATION

System	Set	Accuracy	Precision	Recall	F_1 score
S1	α_β	0.966 ± 0.015	0.969 ± 0.015	0.965 ± 0.016	0.967 ± 0.016
	α_γ	0.952 ± 0.024	0.957 ± 0.022	0.951 ± 0.023	0.953 ± 0.024
	β_α	0.966 ± 0.016	0.968 ± 0.016	0.964 ± 0.016	0.966 ± 0.016
	β_γ	0.969 ± 0.016	0.971 ± 0.015	0.967 ± 0.016	0.969 ± 0.016
	γ_α	0.955 ± 0.023	0.958 ± 0.020	0.954 ± 0.022	0.954 ± 0.022
	γ_β	0.966 ± 0.021	0.970 ± 0.019	0.964 ± 0.021	0.966 ± 0.021
S2	α_β	0.957 ± 0.010	0.961 ± 0.011	0.955 ± 0.010	0.957 ± 0.011
	α_γ	0.943 ± 0.018	0.949 ± 0.017	0.941 ± 0.016	0.943 ± 0.018
	β_α	0.951 ± 0.014	0.954 ± 0.014	0.950 ± 0.012	0.952 ± 0.013
	β_γ	0.954 ± 0.015	0.957 ± 0.015	0.953 ± 0.014	0.955 ± 0.015
	γ_α	0.941 ± 0.016	0.945 ± 0.015	0.941 ± 0.015	0.941 ± 0.015
	γ_β	0.953 ± 0.019	0.957 ± 0.017	0.951 ± 0.018	0.953 ± 0.019

E. Cross-platform classification

We designed this experiment considering one of our key motivations, *i.e.,* to build a system that can detect covert cryptomining in situations where the hardware is heterogeneous, *e.g.,* BYOD workplace. Here, we used two additional laptops, *S1'* and *S2'*, to collect a new test set. *S1'* and *S2'* has a distinct hardware configuration but identical processor as *S1* and *S2*, respectively. For each sub-class of both mining and non-mining tasks, we collected 15 separate samples on both *S1'* and *S2'*. The target of this experiment was to perform the *binary* classification of mining and non-mining samples, as mentioned in Section V-A. Here, we used our dataset collected previously on *S1* and *S2* as the training set, but for testing, we used samples obtained from *S1'* and *S2'*. Fig. 9 depicts the results of the *cross-platform* classification. Fig. 9(a) and Fig. 9(b) correspond to *S1'* and *S2'*, respectively.



(a) *S1'* (b) *S2'*

Fig. 9. Results of the *cross-platform* classification

Essentially, our proposed approach is to profile the magnetic field emission of a processor for the set of available mining algorithms. As shown in Fig. 9, the performance of the *binary* classifier on *S1'* and *S2'* is nearly at par (with a maximum performance degradation of 5.5%, which is in the precision) with its average performance on *S1* and *S2* (see Fig. 6), where the training set was originally collected.

Finally, we used the profile of one processor to classify samples from another processor. In these separate experiments, we used training-set from one device (*S1* | *S2*) and test-set from another device (*S2* or *S2'* | *S1* or *S1'*). We found that the profile of one processor may not be reliably used to classify instances from another processor. In fact, these results align with our fundamental idea to profile the magnetic

field emission of individual processors for the set of available mining algorithms.

VI. Discussion

In this section, we discuss the important aspects of our proposed approach and address its potential limitations.

A. Zero-day attack

In our context, a zero-day attack would be to mine a cryptocurrency that uses a brand new or custom PoW algorithm. However, for such a currency to have a value/worth in the real-world, its PoW algorithm must be mathematically robust as well as accepted by the crypto-community and its core network must be supported by moderate-to-large scale miners/pools. Hence, by the time a zero-day cryptocurrency becomes ready for mining, its algorithm would be public knowledge, providing us sufficient time to train our system for this new currency's signature.

On another side, conscious-miners as well as the actors behind unconscious-mining tend to mine more profitable currencies - whose mining algorithms are certainly public knowledge - to maximize their profit and avoid hashing the less valuable ones. In our experiments, we considered all the mining algorithms supported by the top-10 mining pools, which indeed are the most mined cryptocurrencies.

B. Probe's orientation and position

The orientation as well as the position of the probe with respect to the processor are the critical aspects of our work. Our approach relies on the total magnetic field (M_{total}), which is computed using Eq. (1). The magnetic sensor's reading - which can be positive as well as negative depending on the direction - for each component of the magnetic field is squared first, which eliminates the influence of the probe's orientation.

Since the magnetic fields decay over distance, the distance (position) of the probe from the processor can be seen as a limitation of our approach. In our scenario, the investigator has at least physical-access - if not login-access - to the system. Hence, one can place the probe near the processor simply by understanding the system's physical architecture. Any light-offset in positioning the probe would still perceive the same waveform of the generated magnetic field though with a different amplitude, which is neutralized during data normalization phase. Nevertheless, using high-precision magnetic sensors may help to manage this limitation to some extent.

C. Interference due to other processes

The miner programs tend to exploit all available compute resources and deprive other processes of these resources. However, minute interference due to the occasional scheduling of other processes can be handled by the very nature of our classification methodology, DTW in particular. Such interference would be minimum in the case of the conscious-miners, who would allocate all the resources to the mining process to maximize the profit. Whereas unconscious mining would interfere with its victim's tasks; this is the situation[5] where the victims can halt their tasks and use our system to detect covert cryptomining.

D. Scalability

The fundamental idea of our proposed approach is to profile the magnetic field emission of a processor for the set of available mining algorithms. Given the finite number of CPUs/GPUs, obtaining signatures is merely a data collection task. At the beginning, it might appear a tedious task. But, once completed, keeping it up-to-date is relatively easy because only a limited number of processors are released from time to time.

E. Restricted mining

A mining strategy to evade detection from our proposed methodology can be *restricted mining* that aims to change the pattern of the emitted magnetic field. Here, the miner can either throttle the mining down or perform arbitrary tasks during mining. But, both maneuvers would directly affect the hashing rate and consequently the profits; making the task of mining less appealing. Nevertheless, like any signature-based detection technique, it may be seen as a limitation of our work.

VII. Conclusions and future work

In this paper, we present a novel methodology to detect and classify covert cryptomining. Our proposed approach focuses on the core of cryptomining, *i.e.,* mining algorithms. Since it uses the magnetic side-channel, it works even if the examiner does not have login-access or root-privileges on the suspect machine. In our study, we considered a wide variety of cryptocurrencies and empirically demonstrated the effectiveness of our system.

In the future, we will further investigate the variations in the magnetic profiles of more processors. We will explore the possibility of creating a common profile across different processors for a given PoW algorithm. We will also evaluate the performance of our approach under different mining rate as well as in scenarios with varying magnetic field profiles, *e.g.,* server rooms. Finally, we hope to release a smartphone app for run-time identification of the covert cryptomining.

Acknowledgment

Ankit Gangwal is pursuing his Ph.D. with a fellowship for international students funded by Fondazione Cassa di Risparmio di Padova e Rovigo (CARIPARO). This work was supported in part by EU LOCARD Project under Grant H2020-SU-SEC-2018-832735, and in part by Huawei Project "Secure Remote OTA Updates for In-Vehicle Software Systems" under Grant HIRPO 2018040400359-2018.

[5]Modern malware can hide and circumvent standard detection approaches.

REFERENCES

[1] (2018) "Is Bitcoin Mining Profitable or Worth it in 2018?". https://tinyurl.com/ybnydb8g.

[2] (2018) "Revenues Down, Hashrates Up: 2018 Mining Outlook By The Numbers". https://tinyurl.com/yc586s9v.

[3] (2017) "NYC Government Employee Caught Mining Bitcoin On Work Computer". https://tinyurl.com/y7vb3nlj.

[4] (2018) "Rogue Employees Mine Cryptocurrency Using Company Hardware". https://tinyurl.com/y8u6s524.

[5] (2018) "College Students Use Free Electricity on Campus to Mine Bitcoin". https://tinyurl.com/ybue72el.

[6] (2018) "Ukrainian Professor Mines Bitcoin at School". https://tinyurl.com/ybglcusy.

[7] (2014) "US Government Bans Researcher for Supercomputer Bitcoin Mining". https://tinyurl.com/y9w8qq59.

[8] (2018) "Nuclear Engineers Arrested for Mining Cryptocurrency Using Government Supercomputer". https://tinyurl.com/y9kdf5ts.

[9] (2017) "An Italian bank's server was hijacked to mine bitcoin". https://tinyurl.com/yac8c8jq.

[10] (2018) "Tesla Hackers Hijacked Amazon Cloud Account to Mine Cryptocurrency". https://tinyurl.com/y9epv3do.

[11] (2017) "CryptoJacking Android Malware 'Loapi' Can Physically Damage Your Device". https://tinyurl.com/yazv85ay.

[12] V. P. Kodali and V. Prasad, *Engineering Electromagnetic Compatibility: Principles, Measurements, Technologies, and Computer Models*. IEEE, 2001.

[13] M. Sadiku, *Elements of Electromagnetics*. Oxford University Press, 2014.

[14] J. V. Kistowski *et al.*, "Variations in CPU Power Consumption," in *7th ACM/SPEC ICPE*, 2016, pp. 147–158.

[15] D. J. Berndt and J. Clifford, "Using Dynamic Time Warping to Find Patterns in Time Series," in *AAAI KDD Workshop*, 1994, pp. 359–370.

[16] D. Silva and G. Batista, "Speeding Up All-pairwise Dynamic Time Warping Matrix Calculation," in *SDM*, 2016, pp. 837–845.

[17] G. Al-Naymat *et al.*, "SparseDTW: A Novel Approach to Speed Up Dynamic Time Warping," in *8th AusDM*, 2009, pp. 117–127.

[18] S. Salvador and P. Chan, "FastDTW: Toward Accurate Dynamic Time Warping in Linear Time and Space," in *3rd KDD Workshop on Mining Temporal and Sequential Data*, 2004, pp. 70–80.

[19] M. Müller *et al.*, "An Efficient Multiscale Approach to Audio Synchronization," in *7th ISMIR*, 2006, pp. 192–197.

[20] T. Prätzlich *et al.*, "Memory-restricted Multiscale Dynamic Time Warping," in *41st IEEE ICASSP*, 2016, pp. 569–573.

[21] E. Keogh and C. A. Ratanamahatana, "Exact Indexing of Dynamic Time Warping," *Springer Knowledge and Information Systems*, vol. 7, no. 3, pp. 358–386, 2005.

[22] X. Wang *et al.*, "Experimental Comparison of Representation Methods and Distance Measures for Time Series Data," *Springer Data Mining and Knowledge Discovery*, vol. 26, no. 2, pp. 275–309, 2013.

[23] J.-J. Quisquater and D. Samyde, "ElectroMagnetic Analysis (EMA): Measures and Counter-measures for Smart Cards," in *Springer Smart Card Programming and Security (E-smart), LNCS*, 2001, vol. 2140, pp. 200–210.

[24] E. Mateos and C. H. Gebotys, "Side Channel Analysis using Giant Magneto Resistive (GMR) Sensors," in *2nd COSADE Workshop*, 2011, pp. 42–49.

[25] C. Song *et al.*, "My Smartphone Knows What You Print: Exploring Smartphone-based Side-channel Attacks Against 3D Printers," in *23rd ACM CCS*, 2016, pp. 895–907.

[26] S. Biedermann *et al.*, "Hard Drive Side-channel Attacks using Smartphone Magnetic Field Sensors," in *Springer Financial Cryptography and Data Security, LNCS*, vol. 8975, 2015, pp. 489–496.

[27] N. Matyunin *et al.*, "Covert Channels using Mobile Device's Magnetic Field Sensors," in *21st ASP-DAC*, 2016, pp. 525–532.

[28] M. Guri *et al.*, "ODINI: Escaping Sensitive Data from Faraday-caged, Air-gapped Computers via Magnetic Fields," *arXiv preprint: 1802.02700*, 2018.

[29] ——, "MAGNETO: Covert Channel between Air-gapped Systems and Nearby Smartphones via CPU Generated Magnetic Fields," *arXiv preprint: 1802.02317*, 2018.

[30] J. Bonneau *et al.*, "SoK: Research Perspectives and Challenges for Bitcoin and Cryptocurrencies," in *36th IEEE S&P*, 2015, pp. 104–121.

[31] D. Y. Huang *et al.*, "Botcoin: Monetizing Stolen Cycles," in *21st NDSS*, 2014, pp. 1–16.

[32] S. Eskandari *et al.*, "A First Look at Browser-based Cryptojacking," *arXiv preprint: 1803.02887*, 2018.

[33] R. K. Konoth *et al.*, "MineSweeper: An In-depth Look into Drive-by Cryptocurrency Mining and Its Defense," in *25th ACM CCS*, 2018, pp. 1714–1730.

[34] J. Liu *et al.*, "A Novel Approach for Detecting Browser-based Silent Miner," in *3rd IEEE DSC*, 2018, pp. 490–497.

[35] W. Wang *et al.*, "SEISMIC: SEcure In-lined Script Monitors for Interrupting Cryptojacks," in *23rd ESORICS*, 2018, pp. 1–20.

[36] J. Rauchberger *et al.*, "The Other Side of the Coin: A Framework for Detecting and Analyzing Web-based Cryptocurrency Mining Campaigns," in *13th ARES*, 2018, pp. 1–10.

[37] J. Rüth *et al.*, "Digging into Browser-based Crypto Mining," *arXiv preprint: 1808.00811*, 2018.

[38] R. Tahir *et al.*, "Mining on Someone Else's Dime: Mitigating Covert Mining Operations in Clouds and Enterprises," in *20th RAID*, 2017, pp. 287–310.

[39] (2013) "The *stress-ng* tool". https://tinyurl.com/my6ehnj.

[40] (2018) "Bitcoin Mining Pools". https://tinyurl.com/y8pdk922.

[41] T. Rakthanmanon *et al.*, "Searching and Mining Trillions of Time Series Subsequences under Dynamic Time Warping," in *18th ACM SIGKDD KDD*, 2012, pp. 262–270.

[42] (2017) "Smoothing of a 1D signal". https://tinyurl.com/ya9z9vgq.
[43] T. Mitsa, *Temporal Data Mining*. Chapman and Hall/CRC Press, 2010.
[44] V. Eruhimov *et al.*, "Constructing High Dimensional Feature Space for Time Series Classification," in *ECML/PKDD*, vol. 4702, 2007, pp. 414–421.
[45] A. Bagnall *et al.*, "The Great Time Series Classification Bake Off: A Review and Experimental Evaluation of Recent Algorithmic Advances," *Springer Data Mining and Knowledge Discovery*, vol. 31, no. 3, pp. 606–660, 2017.
[46] X. Xi *et al.*, "Fast Time Series Classification using Numerosity Reduction," in *23rd ACM ICML*, 2006, pp. 1033–1040.

Appendix A
Acronyms

TABLE A.1 lists the acronyms used for various cryptocurrencies.

TABLE A.1
Acronyms and their corresponding cryptocurrency

Acronym	Cryptocurrency
BCD	Bitcoin Diamond
BCH	Bitcoin Cash
BTC	Bitcoin
BTM	Bytom
DASH	Dash
DCR	Decred
ETC	Ethereum Classic
ETH	Ethereum
LTC	Litecoin
QRK	Quark
SBTC	SuperBitcoin
SC	Siacoin
UBTC	UnitedBitcoin
XMC	Monero-Classic
XMR	Monero
XZC	Zcoin
ZEC	Zcash

Appendix B
Standard definitions

Here, we present the definitions of some concepts that we used in our work.

Z-normalization transforms each feature in such a way that the mean becomes zero and standard deviation becomes one. Specifically, given a feature x and one of its value x_i, the following formula is applied:

$$Z(x_i) = \frac{x_i - \mu(x)}{\sigma(x)},$$

where $\mu(x)$ and $\sigma(x)$ are the mean and standard deviation of the variable x.

Standard error of a variable y is expressed as:

$$S_E(y) = \frac{\sigma(y)}{\sqrt{n}},$$

where n and $\sigma(y)$ are the number of observations and standard deviation of the variable y.

Margin of error is the range of values above and below the sample mean for a given confidence interval. It is calculated as:

$$z * S_E(y),$$

where z is the coefficient for the selected confidence level. *E.g.,* z is 1.96 for 95% confidence interval.

Accuracy measures how often the classifier makes the right prediction defined as the ratio between the number of hit and the number of predictions.

Precision quantifies the ability of a classifier to not label a negative example as positive. It is computed as the ratio of the number of true positives and the total number of instances labeled as positives.

Recall defines the probability that a positive prediction made by the classifier is actually positive. It is computed as the ratio of the number of true positives and the total number of positives in the set.

F_1 score is a single metric that combines both precision and recall via their harmonic mean:

$$F_1\ score = 2 \times \frac{precision \times recall}{precision + recall}$$

Ankit Gangwal received his BTech degree in Information Technology from RTU Kota, India in 2011 and his MTech degree in Computer Engineering from MNIT Jaipur, India in 2016. Currently, he is a Ph.D. student in the Department of Mathematics, University of Padua, Italy with a fellowship for international students funded by Fondazione Cassa di Risparmio di Padova e Rovigo (CARIPARO). His current research interest is in the area of security and privacy of the blockchain technology and novel network architectures.

Mauro Conti is Full Professor at the University of Padua, Italy, and Affiliate Professor at the University of Washington, Seattle, USA. He obtained his Ph.D. from Sapienza University of Rome, Italy, in 2009. After his Ph.D., he was a Post-Doc Researcher at Vrije Universiteit Amsterdam, The Netherlands. In 2011 he joined as Assistant Professor the University of Padua, where he became Associate Professor in 2015, and Full Professor in 2018. He has been awarded with a Marie Curie Fellowship (2012) by the European Commission, and with a Fellowship by the German DAAD (2013). His research is also funded by companies, including Cisco, Intel, and Huawei. His main research interest is in the area of security and privacy. In this area, he published more than 250 papers in topmost international peer-reviewed journals and conference. He is Area Editor-in-Chief for IEEE COMST, and Associate Editor for several journals, including IEEE COMST, IEEE TIFS, IEEE TDSC, and IEEE TNSM. He is Senior Member of the IEEE.