2023 19th International Conference on Network and Service Management (CNSM)

Enhancing DeCrypto: Finding Cryptocurrency Miners based on Periodic Behavior

Josef Koumar[1,2], Richard Plný[2], and Tomáš Čejka[1]

[1]*CESNET, a.l.e.*, Prague, Czech republic
Email: {koumar, cejkat}@cesnet.cz
[2]*Czech Technical University in Prague*, Czech republic
Email: {koumajos, plnyrich}@fit.cvut.cz

***Abstract*—While the popularity of cryptocurrencies and the whole industry's value are rising, the number of threat actors who use illegal "coin miner malware" is increasing as well. The threat actors commonly use computational resources of companies, research and educational institutions, or end users. In this paper, we analyzed the long-term periodic behavior of the cryptocurrency miners communicating in computer networks. We propose a novel method for cryptominers detection using specially designed periodicity features. The detection algorithm is based on the mathematical detection of periodic *Flow time series (FTS)* and feature mining. Altogether with the Machine Learning technique, the resulting system achieves high-precision performance. Furthermore, our approach enhances a flow-based cryptominers detection system DeCrypto to further improve its reliability and feasibility for high-speed networks.**

***Index Terms*—cryptocurrencies, cryptocurrency miners, network traffic, network traffic analysis, periodicity, Lomb-Scargle periodogram, network traffic classification, Machine Learning**

I. Introduction

Cryptocurrencies firstly occurred in 2008, when Satoshi Nakamoto published Bitcoin's whitepaper [1]. Since then, many cryptocurrencies have been created. A decentralized system of money, where transactions are verified by individuals called miners and mutual trust is built on cryptography, quickly became very popular.

Many cryptocurrencies are based on Proof-of-Work (PoW) mechanisms, which consume a lot of electricity and processing power. Today, Proof-of-Stake (PoS) mechanisms, which do not require demanding computations, are becoming popular — Ethereum switched to the PoS mechanism in 2022[1]. However, PoW is still used dominantly[2]. Miners use electricity and processing power to verify transactions (mining), which is rewarded in the form of crypto coins[3].

However, mining can be easily exploited by attackers to gain money and therefore remains in high positions on lists of cybersecurity threats. The Cisco Umbrella Academy Report from 2021[4] showed that 69% of organizations experienced some level of unsolicited cryptomining. Attackers use "abusive cryptomining" to mine cryptocurrencies on the victims' computers. As a result, attackers get new crypto coins for free, and victims get high electricity bills and sometimes even broken devices. Furthermore, *Cryptojacking* [2] is a technique in which mining code (written in JavaScript) is embedded into a website and all who visit such a website will become victims of abusive mining. And they can never know! The second way to perform abusive mining is directly through *binary*-based mining malware [3]. Such binaries can be delivered in a phishing mail or as part of a regular downloaded file.

Abusive cryptomining is still an active and commonly used attack. Therefore, highly accurate and effective detection methods are still needed to protect users around the world. Existing detection methods utilize Machine Learning (ML) with basic statistical features derived from flow-based network telemetry. Nevertheless, actions performed by cryptocurrency miners repeat periodically and time-related features such as periodicity are essential in the network classification as shown by Koumar et al. [4] and MontazeriShatoori et al. [5]. However, we were unable to find a method to detect cryptocurrency miners based on their periodic behavior. Therefore, we propose and evaluate a novel approach based on the time series analysis of *Flow Time Series (FTS)* [6]. Our work focuses on the detection of periodic behavior, which is applied as an extension of DeCrypto system [7]; with the goal to make its reliability even higher.

Our contributions can be summarized as follows:

– We propose a novel technique for cryptominers detection using ML and periodic behavior feature set achieving 100 % precision
– We create a novel network traffic analysis approach for the detection of periodic behavior based on FTS, resulting in 43 input features for ML
– We present experiments with different lengths of FTS for the purpose of finding the best settings for deployment into high-speed ISP-level networks
– Our approach can be used as an extension of the DeCrypto system [7] and improve its accuracy by 2.95 %, recall by 7.74 %, and F1-score by 4.37 % enhancing its overall reliability
– We created datasets of FTS and periodic behavior features based on the CESNET-MINER22 datasets [8], which we

This research was funded by the Ministry of Interior of the Czech Republic, grant No. VJ02010024: Flow-Based Encrypted Traffic Analysis and also by the Grant Agency of the CTU in Prague, grant No. SGS23/207/OHK3/3T/18 funded by the MEYS of the Czech Republic.

[1] https://ethereum.org/en/developers/docs/consensus-mechanisms/pos/

[2] https://www.forbes.com/advisor/investing/cryptocurrency/proof-of-work/

[3] https://developer.bitcoin.org/devguide/mining.html

[4] https://learn-cloudsecurity.cisco.com/umbrella-library/2021-cyber-security-threat-trends-phishing-crypto-top-the-list

978-3-903176-59-1 ©2023 IFIP

made publicly available on Zenodo [9].

This paper is organized as follows: Section II summarizes the related works. Section III describes the periodic behavior of mining activity. Section IV provides information about time series analysis concepts and describes a novel detection approach to periodic cryptocurrency mining behavior. Section V provides a complete description of features used for detection. Section VI describes the entire classification pipeline and its results. Section VII describes the cooperation with the DeCrypto system and shows the achieved improvement of the system. Section VIII concludes this paper.

II. Related works

Basic principle for network detection are block-lists, for example [10] for cryptomining. However, each domain or IP address can host multiple services, making block-list-based detection unreliable [11], [12]. Moreover, block-lists can never be complete and suffer from a high false-positive rate. Therefore, multiple studies have proposed mining detection using various other approaches.

Swedan et al. [13] proposed Mining Detection and Prevention System (MDPS). Their system uses URL block lists, Man-In-The-Middle proxy, and Deep Packet Inspection (DPI). Decrypted payloads are inspected by mining code detectors and VirusTotal[5] is used for detection of malicious JavaScript libraries. Unfortunately, it is impossible to use MITM proxies and DPI on high-speed ISP-level networks due to technical difficulties. Moreover, concerns about users' privacy may rise.

Žádník et al. [14] proposed a detection solution that uses flow-based features and ML for primary cryptominers identification and active probing as a secondary verification to remove false positives. In addition, Žádník et al. [14] introduced an overview and statistical properties of cryptomining communication. Even though the proposed solution achieves good accuracy, utilization of active probing brings performance limitations and makes it unsuitable for high-speed networks.

Another flow-based ML detector was proposed by Muñoz et al. [15]. They generated a cryptocurrency miners traffic dataset and analyzed its characteristics. Used features were mainly describing the amount of data transferred and data throughput. However, real-world deployment was not considered and evaluated. The accuracy of their model in real deployment can be significantly lower due to usage of a small portion of lab-created cryptomining traffic that represents only 0.03% (approx. 700 flows) of the whole dataset.

Plný et al. [7] proposed DeCrypto — a system based on multiple data sources and classifiers, which achieves high accuracy and minimal false-positive rate. They utilize ML, keywords detection in TLS SNI[6] and detection of Stratum mining protocol. DeCrypto was deployed on a nation-wide ISP-level network and was thoroughly evaluated.

Compared to related works, DeCrypto is the only system which was deployed and evaluated on such large network. Our work enhances the DeCrypto system and provides another data source to further minimize false positives and support high accuracy and precision. The proposed classifier aims to detect the periodic behavior of cryptominers by utilizing FTS analysis.

III. Periodicity behavior of cryptocurrency miners

Functionality of cryptomining software is pretty straightforward. Firstly, a miner connects to a mining pool or a crypto network to fetch an unverified block. Then, the miner starts generating hashes of header of this block together with a nonce (number used only once). When the miner finds a hash string that starts with a certain number of leading zeros (difficulty), the hash is considered correct and verified, and the completed block is broadcasted back to the mining pool or crypto network. The whole process starts again — the miner is fetching a new unverified block. Furthermore, the miner can receive a notification that the currently processed block was already verified by someone else. Miner then has to fetch a new block.

We can notice many periodic activities in this rather elementary description. Moreover, the average time needed for mining a block for most cryptocurrencies is either known or can be derived from public Blockchain databases. For example, a new Bitcoin block is mined approximately every 10 minutes[7]. Therefore, we expect miners will fetch unverified blocks every couple of minutes. Mining pools usually have slightly lower difficulty used to verify that miners are really working — more hashes will satisfy rules and will be sent to the mining pool, even though the block can not be successfully mined. Some of these hashes will satisfy the network's difficulty as well. Therefore, communication will occur more often. However, it is enough for periodicity detection.

IV. Periodicity detection

We need the FTS with minimal noise for periodic behavior detection. Every FTS has to contain flows from the one process only; for example, YouTube video streaming, webpage loading, Keep-Alive communication or cryptomining communication. Because of this, we create FTS by sampling with *Network dependencies* [6]. A network dependency is a relationship between two IP addresses where one provides a service to the other. Furthermore, each network dependency has an ID created as `ip_addr:port-ip_addr`, where the port is well-known or registered and represents the service. Furthermore, some processes, like cryptocurrency miners, use non-registered ports. For such cases, it is necessary to use time series as `ip_addr:port-ip_addr:port` and apply detection of such ports in post-processing of FTS collection. For cryptominers from the CESNET-MINER22 dataset [8], such ports are 12433, 14433, 14444, and 20535.

The network dependencies were evaluated as the best choice for sampling [6], causing minimal noise and occurring with

[5] https://virustotal.com

[6] Server Name Indication of the TLS protocol

[7] https://thebitcoinmanual.com/articles/btc-block-time/

the best result of periodic behavior detection. For example, the sample only by relationship between two IP addresses ended significantly worse because multiple processes can share one public IP address.

In the FTS collection, we experimented with multiple time windows to evaluate deployment properties into real-time networks, for example, into ISP network CESNET2 side-by-side the DeCrypto system. We set the time windows, i.e., FTS length, from 10 minutes to 24 hours.

Because the FTS are *Unevenly Sampled Time Series (USTS)*, we use the Lomb-Scargle periodogram [16]–[18], P_{LS}, as a source of information for deciding whether the FTS contains periodic behavior.

The method for detection of periodic behavior is based on our previous work [4]. However, it is not feasible for high-speed ISP networks because it can process only a small number of time series per second, and thus optimizations would be required. A novel and faster periodicity detection pipeline presented in Fig. 1 is designed for high-speed networks. This pipeline is able to process the whole CESNET-MINER22 dataset (containing 3 million flows obtained during 3 months) in 5 seconds with proper precision based on two threshold values of *Scargle's Significance Test (SST)* [19].

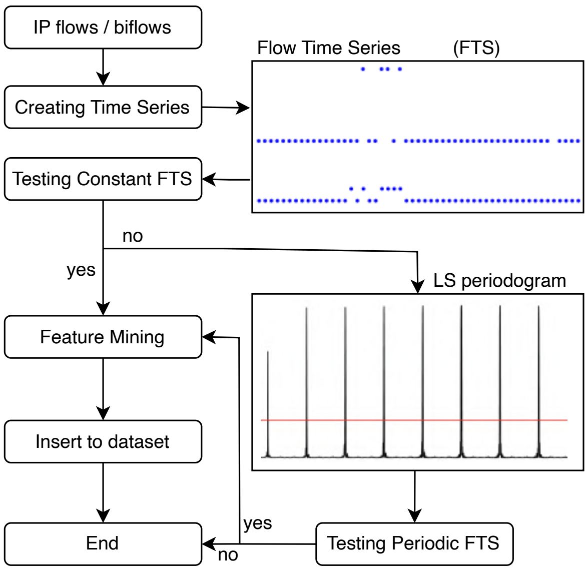

Fig. 1. Periodicity detection pipeline

The SST uses the *Cumulative Distribution Function (CDF)* of Lomb-Scargle periodogram powers (P_{LS}), which is defined as probability $P\left[P_{LS} > z\right]$. As the observed power z becomes larger, it becomes exponentially less likely produced by pure noise. Then the observed power level is more likely due to an original deterministic (i.e., non-noise) feature in the time series [19].

The SST is performed for threshold value *Significance level*, which defines the tested value:

$$z = max(P_{LS}) * \textit{Significance level} \qquad (1)$$

The probability that z is bigger then rest of periodogram powers P_{LS} is computed by the equation:

$$P\left[P_{LS} > z\right] = 1 - e^{-\frac{z}{\sigma^2_{P_{LS}}}} \qquad (2)$$

If $P\left[P_{LS} > z\right]$ is larger or equal to threshold value *Percentage level*, then the time series contains periodic behavior. In this work, we evaluate the classification results based on the settings of threshold values *Significance level* and *Percentage level*. Furthermore, the smaller *Significance level* is, the more strict is the periodic detection. Moreover, the bigger *Percentage level* is, the more strict is periodic detection.

Furthermore, for FTS that are constant, i.e., the same datapoint occurs periodically, periodic detection by LS periodogram and SST can fail. So, we test if FTS is constant and, in such case it is taken as periodic. Moreover, the periodic behavior detection pipeline can cause false positives for short FTS. Thus, we run the pipeline only for the FTS with at least ten datapoints.

V. Feature Mining

The FTS occur with two time series metrics: the number of packets and bytes within the IP flow. Furthermore, each datapoint, i.e., IP flow, has two time information: time of transmission of the first (tf_i) and the last (tl_i), packet. Moreover, from the time information, two more time series metrics can be created: the durations and time differences.

The duration for each flow is computed, $d = tl_i - tf_i$, and forms a time series metric. The time differences are calculated for neighboring flows as $dt = tf_i - tl_{i-1}$. These time series metrics and the Lomb-Scargle periodogram are input data for the computation of features for ML.

We identify two cases of periodic behavior in FTS, and we define specific features for their ideal representation. The first periodic behavior is *Clear*, it is defined as the same datapoint periodically occurring at the same time interval. Fig. 2 shows an example of Clear periodic behavior. The second periodic behavior is *Sinusoidal*, it is defined by occurring the datapoints like sinus function with some noise. Fig. 3 shows an example of Sinusoidal periodic behavior.

In total, we present 43 features. To describe both Clear and Sinusoidal periodic behavior, we define two groups of features. The first group of Clear periodic behavior features contains: *packet_value*, *bytes_value*, *duration_value*, and *difftimes_value*. The second group of Sinusoidal periodic behavior features contains two features for each metric named with suffixes: the interval's lower value (*_x*), and the upper value (*_y*). The list of these features contains: *packet_value_x*, *packet_value_y*, *bytes_value_x*, *bytes_value_y*, *duration_value_x*, *duration_value_y*, *difftimes_value_x*, and *difftimes_value_y*.

Moreover, we add basic statistic properties (*Mean*, *Standard deviation*, *Skewness*, and *Kurtosis*) for each time series metric resulting in 16 features. The *Skewness* is computed as $Sk =$

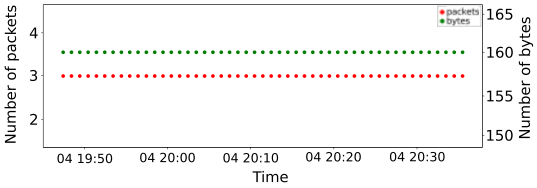

Fig. 2. Example of FTS with the Clear periodic behavior

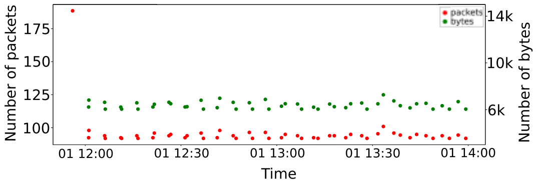

Fig. 3. Example of FTS with the Sinusoidal periodic behavior

$\frac{\sum_i (m_i - \mu_m)^3}{\sigma_m^3}$, where m_i is metric i-th value of metric m. The *Kurtosis* is computed as $K = \frac{\sum_i (m_i - \mu_m)^4}{\sigma_m^4}$.

Furthermore, we use the Lomb-Scargle periodogram to generate the set of well-known frequency-based features listed below:

- *Min power, Max power* — Represent the minimum and maximum power of the LS periodogram.
- *Frequency of min power, Frequency of max power* — Describe the frequency of the minimum and maximum power of the LS periodogram.
- *Spectral bandwidth* — Describes the difference between upper and lower frequencies at which spectral energy is half its maximum value.
- *Spectral centroid* — Indicates at which frequency the energy of a spectrum is centred upon.
- *Spectral energy* — Represents the total energy present at all frequencies in LS periodogram.
- *Spectral entropy* — The degree of randomness or disorder in the LS periodogram.
- *Spectral flatness* — Estimates the uniformity of signal energy distribution in the frequency domain.
- *Spectral flux* — The rate of change of periodogram power with increasing frequency.
- *Spectral kurtosis* — Indicates a nonstationary or non-Gaussian behavior in the power spectrum.
- *Spectral rolloff* — It is defined as frequency below at is concentrated 85% of the distribution power.
- *Spectral spread* — It is the difference between the highest and lowest frequency in the power spectrum.
- *Spectral skewness* — The measure of peakedness or flatness of power spectrum.
- *Spectral slope* — The slope of the power spectrum trend in a given frequency range.
- *Spectral zero crossing rate* — Refers to the rate of shift of the sign of a wave, which is the rate of change from negative to positive or the reverse.

VI. Classification based on periodic features

A. Creating datasets

As we mentioned before, we experiment with the length of FTS by defining the time interval in which FTS are collected. The chosen time intervals are: 24 h, 12 h, 6 h, 4 h, 2 h, 1 h, 30 m, 15 m, and 10 m. The IP flows were taken from the CESNET-MINER22 dataset, which was created by monitoring the CESNET2[8] network infrastructure by ipfixprobe[9] — open-source flow exporter. The computed FTS created from the CESNET-MINER22 dataset and the datasets of periodic behaviors have been published on Zenodo [9].

The number of created FTS depends on the size of the time interval. In the top left graph of Fig. 4, the number of FTS increases with the decreasing time interval. The number of constant FTS achieved the same trend as it can be seen on the left bottom graph of Fig. 4. However, the top right graph of Fig. 4 shows the number of tested FTS with at least ten datapoints, and it is not increasing with the same trend.

The number of periodic FTS depends on the settings of Scargle's Significance Test (SST), i.e., *Significance level* and *Percentage level*. The right bottom graph of Fig. 4 shows the numbers of periodic FTS for each tested setting. The number of periodic FTS with *Significance level* equals to 0.1 has the same trend as the number of tested FTS. The number of periodic FTS with *Significance level* equals to 0.01 has decreasing trend with decreasing time intervals. And the number of periodic FTS with *Significance level* equal to 0.001 has a slightly increasing trend with decreasing time intervals.

If we realize that the longer the time interval, the more accurate the periodicity test should be, and we observed the behavior just described, we can conclude that the correct SST setting should be *Significance level* equal to 0.01. Furthermore, we performed experiments with the settings of SST in classification to obtain the best classification results (the following sections provide further information).

The classification pipeline uses only flows from periodic FTS. The number of classified flows from non-periodic FTS is shown in Fig. 5. This figure shows results with SST *Significance level* set to 0.01 and *Percentage level* set to 0.99. The figure also shows that the number of flows inside non-periodic FTS increases with smaller time intervals. Furthermore, the non-periodic FTS usually contain small count of datapoints, only one in most cases.

B. Classification pipeline

The classification pipeline is a set of steps which create the best final model. Firstly, Plný et al. [8] split the CESNET-MINER22 dataset into the *Design* and *Evaluation* parts. The *Design* part is for creating, tuning and selecting the best model and the *Evaluation* part is for the one-time evaluation of the best model from the *Design* part. This default split avoids overfitting and allows comparison of results with other works without variation by random sampling.

[8] The Czech Educational and Science Network

[9] https://github.com/CESNET/ipfixprobe

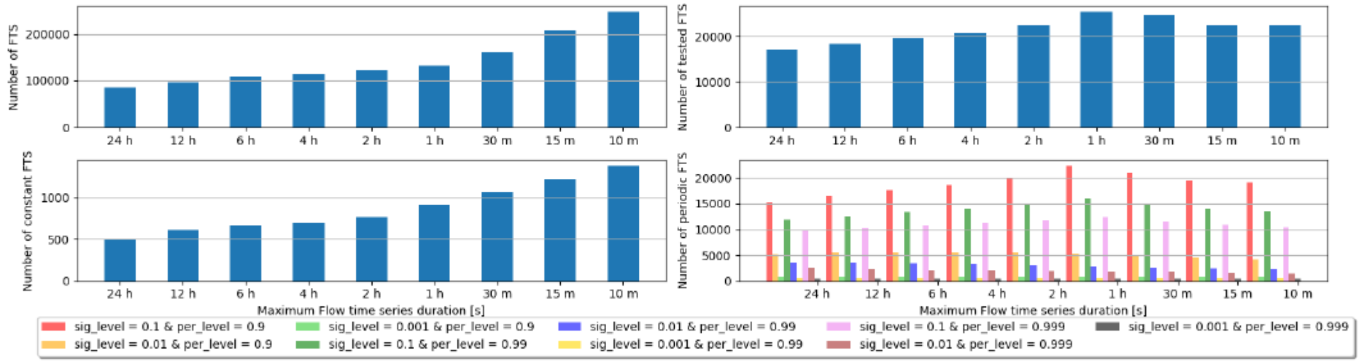

Fig. 4. The figure presents the number of FTS, the number of tested FTS, and the number of constant FTS and periodic FTS in datasets of periodic behavior created from the CESNET-MINER22 dataset. The number of periodic FTS depends on Scargle's Significance Test (SST) settings.

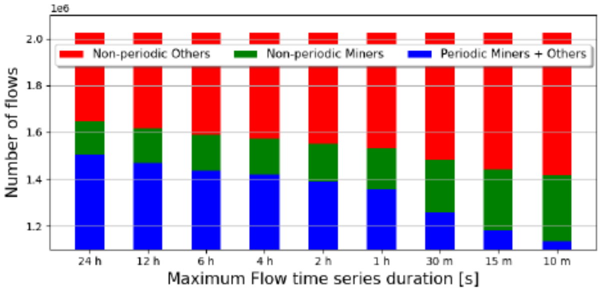

Fig. 5. The number of flows that are in periodic and non-periodic FTS.

TABLE I
SUMMARY OF BEST RESULTS WITH SETTINGS *Significance level* EQUAL TO 0.1 AND *Percentage level* EQUAL TO 0.9.

Time interval	Accuracy	Precision	Recall	F1-score
24 h	93.63	83.49	74.19	78.57
12 h	93.31	87.82	69.27	77.45
6 h	93.37	90.01	74.87	81.74
4 h	94.19	92.24	78.88	85.04
2 h	94.34	93.44	80.46	86.47
1 h	94.86	94.21	84.39	89.03
30 m	95.13	96.11	87.45	91.57
15 m	95.19	95.43	89.21	92.21
10 m	94.92	95.16	88.88	91.91

In the design phase, we start with selection of the optimal ML algorithm by testing 11 well-known ML algorithms from which XGBoost, LightGBM and CatBoost algorithms achieved similar results. We choose the XGBoost algorithm. Our experiments ended similarly on each dataset.

After selecting the algorithm, we optimized the hyperparameters to get a suitable set that performs well on the dataset, yet does not cause overfitting. We use the *hyperopt library* [20] to tune the following hyperparameters: *n_estimators*, *max_depth*, *gamma*, *reg_alpha*, *reg_lambda*, *min_child_weight*, and *colsample_bytree*. This tuning phase results in the best settings of hyperparameters of the XGBoost classifier. We picked the best models in the design phase. The best models for each time interval were selected based on the evaluation metrics: *Accuracy*, *Precision*, *Recall*, and *F1-score*.

After the design phase, we evaluate selected models on the *Evaluation* part of the CESNET-MINER22 dataset. Source code of the classification pipeline and experiments are published on GitHub[10].

C. Results of classification

The best results for settings *Significance level* equal to 0.1 and *Percentage level* equal to 0.9 are presented in Table I. As can be seen, the periodic features are more than suitable for cryptocurrency miners detection. Moreover, no False positives occur for each time interval.

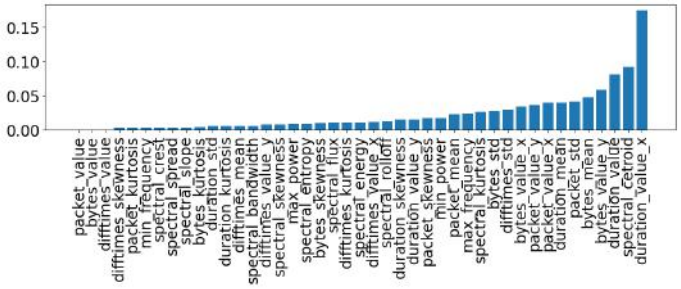

Fig. 6. Feature importance

The results of the feature importance investigation for the trained model are shown in Fig. 6. We can see that the Clear periodic behavior features are one of the least important, but they cause a low false-positive rate as described by Koumar et al. [4].

For the deployment of the augmented DeCrypto system into networks for real-time detection, it is crucial to choose a proper time interval for FTS sampling and proper settings of the SST test. However, the best achieved results, presented in Table I, can not be used for selecting proper time interval because most of them achieved similar results. Moreover, the results can be strongly influenced by variation caused by the randomness that naturally occurs in splitting, training and tuning of ML models [21]. We evaluated the classification pipeline for each time interval 100 times. The results of

[10] https://github.com/koumajos/EnhancedDeCrypto

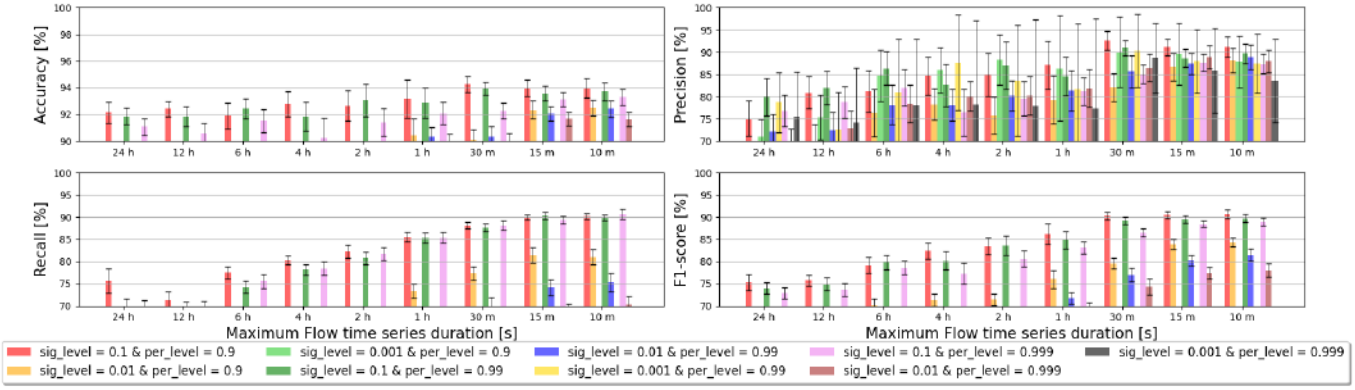

Fig. 7. The graph contains results of the Machine Learning pipeline for each dataset that was run 100 times to obtain significant statistical results. Therefore, the presented results are the mean of the classification metric with standard deviation. On the graph can be seen the dependency between ML results, the time window of creation time series and settings of periodicity detection.

this experiment enabled choosing the correct time interval for deployment. The resulting mean and standard deviation of each classification metric, for each time interval and SST setting, is shown in the Fig. 7.

Our experiments show that most of the best results are caused by variation because the mean values of each classification metric do not achieve the same or closely the same value. However, the mean and standard deviation helped to select the time interval equal to 30 m, *Significance level* equal to 0.1 and *Percentage level* equal to 0.9. This setting occurs with the best average results and the smallest standard deviation; thus the results are robust.

VII. Cooperation with DeCrypto system

A. Enhancing DeCrypto system

DeCrypto system consists of weak classifiers, which process different data sources. It is highly flexible, easily customizable and expandable. We extended the DeCrypto's Meta Classifier, which is now able to use *FTS proba* — the probability of a flow being a miner based on the time series analysis. The *FTS proba* is created by computing the mean of all previous `predict_proba()` output of the XGBoost classifier. The previous values are stored for each network dependency and time interval.

Flow time series analysis is highly accurate; therefore, we decided to use it for both marking miners and non-miners (class *other*). Two thresholds were used for classification — FTS_UPPER_BOUND and FTS_LOWER_BOUND. The flow is marked as the miner if the FTS proba is higher than FTS_UPPER_BOUND. If the FTS proba is lower than FTS_LOWER_BOUND, the flow is marked as the non-miner. Otherwise, the decision process of the original DeCrypto continues [7] (FTS proba is used after the Stratum classifier). We conducted the search for the optimal threshold settings. The FTS_UPPER_BOUND threshold was set to 90 %. The FTS_LOWER_BOUND threshold was set to 10 %.

B. Deployment of Enhanced DeCrypto

We use the *Dynamic Profile Processing Platform (DP^3)*, available at GitHub[11], for the deployment of our approach into a high-speed ISP network to enhance the DeCrypto system. The DP^3 is an open-source data processing platform for maintaining dynamically changing profiles of *entities* (network dependencies in our case) represented by sets of attributes of different data types, including various types of time series. It allows to apply custom processing functions over the profiles to enrich, correlate or otherwise analyze the data to derive new information or detect some events.

The proposed deployment of our approach with the DP^3 platform is shown in Fig. 8. The *Create time series* module creates FTS and using a *Datapoint sender* it sends them into a DP^3 instance, where they are stored into profiles of corresponding network dependencies. The processing engine of DP^3 then runs modules for the detection of periodic behavior, feature mining and ML-based classification. The DeCrypto system gets the alerts from our approach by calling DP^3 API.

C. Results of cooperation

The confusion matrix of the Enhanced DeCrypto system by classification based on periodic behavior is shown in Tab. II. The confusion matrix contains the absolute number of *True positive*, *False positive*, *True negative* and *False negative*. Furthermore, the relative number in percent and change (+ or −) in percent from original DeCrypto and Enhanced DeCrypto. The confusion matrix occurs with a significant decrease of *False negative*; in detail, the absolute change is from 61,300 to 29,543.

Moreover, the Enhanced DeCrypto system achieved the 97.25 % Accuracy, 99.99 % Precision, 92.47 % Recall, and 96.08 % F1-score. That is improvement by 2.95 % Accuracy, 0.001 % Precision, 7.74 % Recall, and 4.37 % F1-score.

[11] https://github.com/CESNET/dp3

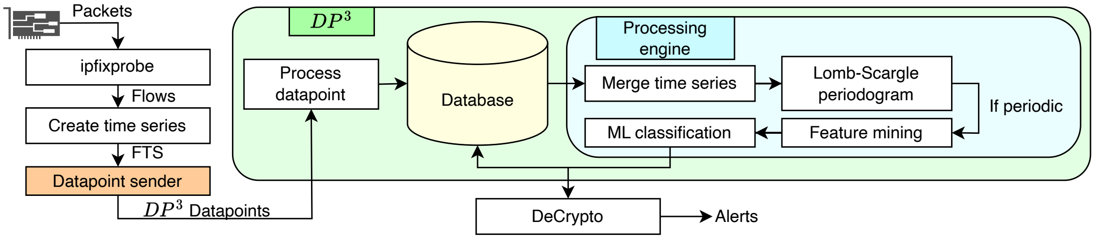

Fig. 8. Deployment into high-speed ISP network using DP^3 platform

TABLE II
RESULTS OF THE ENHANCED DECRYPTO SYSTEM

		Actual	
		Miner	Other
Predicted	Miner	33.75% (363,034)	0.0005% (6)
	Other	2.74% (29,543)	63.50% (682,993)

VIII. CONCLUSION

With the rising value of the cryptocurrency sector, people are more tempted to join the community and earn revenue by cryptomining. Even though many cryptocurrencies are by now based on Proof-of-Stake (PoS) mechanism, the Proof-of-Work (PoW) mechanism is still dominantly used. Thus, cryptomalware and abusive mining remain a critical threat to institutions' resources.

This paper proposed a novel approach for detection of cryptominers by monitoring network traffic. We create the *Flow Time Series (FTS)* by grouping IP flows by network dependencies with defined time intervals. We use the *Lomb-Scargle periodogram* and *Scargle's Significance Test (SST)* for detection of periodic behavior. We described 43 features for periodic and constant FTS that we experimentally evaluated. The feature set is an input vector to XGBoost algorithm. The correct settings of time interval, hyperparameters of SST, and hyperparameters of ML provide reliable detection.

The proposed cryptomining detection can be deployed as a stand-alone detector. Additionally, the cooperation with the DeCrypto system shows the feasible way to use a long-term view of communication as a weak indicator to increase the reliability of the final system. Furthermore, the cooperation also increases the ability to deploy the system into real-time networks and improves the trust in the produced alerts.

The DP^3 platform was evaluated as a suitable system for monitoring network traffic by Flow Time Series with excellent performance and scalability for the high-speed ISP network CESNET2. Furthermore, our approach for classification only needs classical IP flows or biflows as the input for computation of the Flow Time Series and related statistical metrics. It does not cause any increase in network telemetry like other classification methods which use extended IP flows. Moreover, the cooperation with the DeCrypto system occurs with almost zero percent of *False positives*. Based on these results, our approach is suitable for high-speed ISP networks and protect their users.

REFERENCES

[1] Satoshi Nakamoto. A peer-to-peer electronic cash system. *Bitcoin.org*, 4:2, 2008.

[2] Geng Hong et al. How you get shot in the back: A systematical study about cryptojacking in the real world. In *ACM SIGSAC*, 2018.

[3] Sergio Pastrana et al. A first look at the crypto-mining malware ecosystem: A decade of unrestricted wealth. In *Proceedings of the Internet Measurement Conference*, IMC '19, page 73–86, New York, NY, USA, 2019. Association for Computing Machinery.

[4] Josef Koumar et al. Network Traffic Classification Based on Periodic Behavior Detection. In *CNSM 2022*, pages 359–363. IEEE, 2022.

[5] Mohammadreza MontazeriShatoori et al. Detection of DoH Tunnels using Time-series Classification of Encrypted Traffic. In *DASC/PiCom/CBDCom/CyberSciTech 2020*, pages 63–70. IEEE, 2020.

[6] Josef Koumar et al. Unevenly Spaced Time Series from Network Traffic. *TMA*, 2023.

[7] Richard Plný et al. DeCrypto: Finding Cryptocurrency Miners on ISP Networks. In *NordSec 2022*. Springer, 2023.

[8] Richard Plný et al. *Datasets of Cryptomining Communication*. Zenodo, October 2022. https://doi.org/10.5281/zenodo.7189293.

[9] Josef Koumar et al. CESNET-MINER22-TS: Periodic Behavior Features of Cryptomining Communication, June 2023. https://doi.org/10.5281/zenodo.8033351.

[10] Eyal Webber Zvik. The crypto mining threat: The security risk posed by bitcoin and what you can do about it, Jan 2018.

[11] Abdurrahman Pektaş et al. Deep learning to detect botnet via network flow summaries. *Neural Comput. Appl.*, 2019.

[12] Karel Hynek et al. Evaluating bad hosts using adaptive blacklist filter. In *MECO 2020*, 2020.

[13] AbedAlqader Swedan et al. Detection and prevention of malicious cryptocurrency mining on internet-connected devices. In *International Conference on Future Networks and Distributed Systems*. ACM, 2018.

[14] Vladimír Veselý et al. How to detect cryptocurrency miners? by traffic forensics! *Digital Investigation*, 31:100884, 2019.

[15] Jordi Zayuelas Muñoz et al. Detecting cryptocurrency miners with netflow/ipfix network measurements. In *2019 IEEE International Symposium on Measurements Networking*, pages 1–6, 2019.

[16] Nicholas R Lomb. Least-squares frequency analysis of unequally spaced data. *Astrophysics and space science*, 39:447–462, 1976.

[17] Jeffrey Scargle. Studies in astronomical time series analysis. II - Statistical aspects of spectral analysis of unevenly spaced data. *The Astrophysical Journal*, 263, December 1982.

[18] Jacob T. VanderPlas. Understanding the lomb–scargle Periodogram. *The Astrophysical Journal Supplement Series*, 236(1), May 2018.

[19] Fabio Frescura et al. Significance tests for periodogram peaks. 2007.

[20] James Bergstra et al. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *International conference on machine learning*. PMLR, 2013.

[21] Hana Ahmed et al. Managing randomness to enable reproducible machine learning. In *Proceedings of the 5th International Workshop on Practical Reproducible Evaluation of Computer Systems*, P-RECS '22, Jun 2022.