ISFCR Experiential Learning Program - 2024

LITERATURE SURVEY
Cryptojacking detection System

No.	Initials	Paper	Publication	References	Objective	Attack	Method of Attack	Victim System	Solution Approach	Research Gaps & Future Works
1.	AU	Behavior-Based Detection of Cryptojacking Malware	2020	*Tanana, Dmitry. "Behavior-based detection of cryptojacking malware." 2020 Ural symposium on biomedical engineering, radioelectronics and information technology (USBEREIT). IEEE, 2020.* *https://ieeexplore.ieee.org/abstract/document/9117732*	detecting both browser-based and executable-type cryptojackers based on CPU usage heuristics.	In-browser & File based	40 browser-based and 10 executable-type from **VirusShare** as well as some other legitimate applications **(50 Samples)**	performed on virtual machine which was allocated 1 processor core with 3GHz frequency and 4 gigabytes of operative memory running Windows 7	Indicators: • CPU util. • RAM util. • Avg. Quadratic deviation Crucial focus on not having false positives.	expand proposed algorithm to detect cryptojacking malware with dynamic CPU load. **(Already done)**
2.	AU	Advanced Behavior-Based Technique for Cryptojacking Malware Detection	2020	*Tanana, Dmitry, and Galina Tanana. "Advanced behavior-based technique for cryptojacking malware detection." 2020 14th International Conference on Signal Processing and Communication Systems (ICSPCS). IEEE, 2020.* *https://ieeexplore.ieee.org/abstract/document/9310048*	modified prev. algorithm to deal with multicore processors and introduce a few additional detection techniques based on the internet connection usage and cryptographic libraries calls monitoring	In-browser & File based	**100** cryptojacking malware samples, 70 browser-based and 30 file-based from **VirusShare**.	virtual machine with Windows 7, 3GHz single-core CPU and 4GB RAM	Additional indicators: • Multi-core CPU usage • Network access • Cryptographic lib. calls	**1.** to study the difference in behavior between legitimate and malicious mining applications **2.** focus on detection of GPU-based malicious miners
3.	MR	Cryptojacking Detection in Cloud Infrastructure using Network Traffic	2023, IEEE	*Kwedza, Philip, and Stones Dalitso Chindipha. "Cryptojacking Detection in Cloud Infrastructure Using Network Traffic." 2023 International Conference on Electrical, Computer and Energy Technologies (ICECET). IEEE, 2023.* *https://ieeexplore.ieee.org/abstract/document/10389593*	Purpose of the model is to detect cryptojacking automatically in a cloud environment, utilizing network traffic	Host-based cryptojacking	**1.** Use of XMRig **2.** Network Traffic Generation **3.** Wireshark are used to collect packets from the VPS instances	cloud infrastructure-Virtual Private Servers (VPS)	**1.** created a dataset using ten VPS instances provided by DigitalOcean **2**.generated cryptomining traffic using XMRig to mine various cryptocurrencies **3.** Feature Engineering • Flow Aggregation • Flow Labeling • Statistic Calculation • Flow Pairing **4.** Machine learning models were classified as cryptomining and regular traffic used KNN,SVM,DT,RFT to detect cryptojacking and	**1.** To validate model it can be tested in a real cloud environment **2.** Can build a system that can detect cryptojacking in real time **3.** To improve the model deep learning techniques could also be done

									their performance evaluated using F1 score and Area Under the ROC Curve (AUC)	
4.	AU	Forensic Analysis of Cryptojacking in Host-based Docker Containers Using Honeypots	2023, IEEE	*Franco, Javier, et al. "Forensic Analysis of Cryptojacking in Host-Based Docker Containers Using Honeypots." ICC 2023-IEEE International Conference on Communications. IEEE, 2023.* *https://ieeexplore.ieee.org/abstract/document/10278764*	**1.** conduct a forensic analysis of host-based cryptojacking malware in Docker containers **2.** utilized honeypots to collect host resources and network data **3.** proposed an approach for monitoring host-based containers and alerting system administrators	Host-based cryptojacking	One of each of the Redis and Nginx containers was used to understand the baseline behavior and the other three of each were tested with three scripts mining Monero on Cudominer, Minexmr, and Xmrpool.eu mining pools	set up a high-interaction honeypot system isolated in a DMZ with ten Docker containers: 4 of these used Redis container images, 4 used Nginx container images, and 2 used Ubuntu container images	bbb—	planning to develop a cryptojacking detection framework for host-based Docker containers using honeypots and machine learning.
5.	AU	Detecting Covert Cryptomining Using HPC	2020, Springer	*Gangwal, Ankit, et al. "Detecting covert cryptomining using hpc." Cryptology and Network Security: 19th International Conference, CANS 2020, Vienna, Austria, December 14–16, 2020, Proceedings 19. Springer International Publishing, 2020.* *https://link.springer.com/chapter/10.1007/978-3-030-65411-5_17*	approach to detect covert cryptomining on users' machines. generic solution that detect covert cryptomining, which is not tailored to a specific cryptocurrency or a form of cryptomining	A generic model which detects all attacks in user machines	Created their own dataset of tasks in which half mining cryptocurrencies and other half not mining. (**1100 samples**)	User systems	Initailly data preprocessing is done along with Feature engineering. A supervised model consisting of Random Forest and SVM classification.	Investigation and monitoring of GPU dedicated events that can assist in creating unique signatures for GPUs experiments with a larger set of systems (CPUs) to observe the generalization of our approach desktop application for run-time identification of covert cryptomining
6.	MR	Cryptojacking Detection with CPU Usage Metrics	2020, IEEE	*Gomes, Fábio, and Miguel Correia. "Cryptojacking detection with cpu usage metrics." 2020 IEEE 19th International Symposium on Network Computing and Applications (NCA). IEEE, 2020.* *https://ieeexplore.ieee.org/abstract/document/9306696*	Cryptojacking detection mechanism based on monitoring the CPU usage of the visited web pages in real-time	In-browser	Cryptojacking Scripts in Web Pages		Used Weka Used signature-based intrusion detection system ● Webpage Crawler ● CPU Monitoring - -mpstat command line tool to monitor the CPU -used Amazon EC2 instances ● ARFF Parser ● Machine Learning Classifier -Two-Level Classification [TLC] -Multiple-Instance Support Vector Machine [MISVM] -Random Subspace Method - Sequential Minimal Optimization DID experiment on ● Training Dataset Composition ● 1 CPU core for 15 seconds ● 1 CPU core for 60 seconds ● Average of the cores for 60 seconds ● 2 CPU cores for 60 seconds ● 4 CPU cores 60 seconds	To work on their approach with non-browser based cryptojacking malware

7.	AU	Cryptomining Detection in Container Clouds Using System Calls and Explainable Machine Learning	2021, IEEE	*Karn, Rupesh Raj, et al. "Cryptomining detection in container clouds using system calls and explainable machine learning." IEEE transactions on parallel and distributed systems 32.3 (2020): 674-691.* [*https://ieeexplore.ieee.org/abstract/document/9215018*](https://ieeexplore.ieee.org/abstract/document/9215018)	use of explainable ML of syscalls to classify anomalous containers methodology for anomaly detection through system calls in the Kubernetes pods is proposed, designed & implemented	Host based cryptojacking	Eight different types of cryptomining containers are used.To enable an env. that supports CPU- intensive, healthy application pods, containers are created that are dedicated to it.	Kubernetes cluster	Developed various ML models and compared thier accuracy, precision, F! Score and other metrics. Models like Decision trees, Feed forward ANN, XGBoost EML, etc. are used and compared. Provides explainability for why the kubernetes pod was flagged to the administrator so an informed decision can be made on removal of pod.	—
8.	MR	Cryptojacking Malware Detection in Docker Images Using Supervised Machine Learning	2022	*Saide, Saide Manuel, Ednilson Luis Alfredo Sarmento, and Felermino DMA Ali. "Cryptojacking Malware Detection in Docker Images Using Supervised Machine Learning." International Conference on Intelligent and Innovative Computing Applications. 2022.* [*https://mauricon.org/conferences/index.php/iconic/article/view/6*](https://mauricon.org/conferences/index.php/iconic/article/view/6)	To develop and evaluate a machine learning-based model to detect cryptojacking malware in Docker images	Attackers insert cryptojacking scripts or malware into Docker images	**1.** Creation of Malicious Docker Images **2.** include cryptomining software (such as xmrig or cpuminer) in the Docker image. **3.** Simulation of Attack Scenarios **4.** Analysis of Detection Techniques	Docker Engine Host Systems Cloud Environment ● Iaas ● Caas	1. installed a VMware Workstation 16 Pro Virtual Machine (VM), with Ubuntu 18.04 LTS operating system where we installed Docker 20.10.7. - “docker pull” command 2. collected 800 Docker images from Docker hub- 400 images had instructions for cryptojacking. The remaining 400 images were free of malicious instructions. 3. “docker history” command for each Docker image to extract all the image instructions into a plain text file. 4. Data Preprocessing ● Lowercasing ● Noise cleaning ● Removal of punctuations ● Tokenization 5. Data Transformation Transformed the text dataset to numerical representation-with two approaches: Bag of Words and Term Frequency-Inverse Document Frequency - TF-IDF 4. trained and evaluated 10 classification algorithms, using K-Fold Cross Validation sampling -found that Stochastic Gradient Descent for Logistic Regression had the highest accuracy scores (97%), and K-Nearest Neighbors (KNN) algorithm had the lowest accuracy scores (89%).	Can use different Machine Learning algorithms to build a predictive model for detecting cryptojacking malware in Docker images.
9.	AU	DeCrypto Pro: Deep Learning Based Cryptomining Malware Detection Using Performance Counters	2020, IEEE	*Mani, Ganapathy, et al. "Decrypto pro: Deep learning based cryptomining malware detection using performance counters." 2020 IEEE International conference on autonomic computing and*	a detection system with a novel model selection framework containing a utility function that can select a classification model for behavior profiling from both the light-weight machine learning	Host based cryptojacking	Since we aim to capture the system status signature of PoW algorithm such as CryptoNight’s signature, we mainly focus on bitwise, cryptographic, and processor- specific encryption operations. Thus we consider	3 Windows machines with various configurations on processing frequency (2.40, 2.90, 2.30 GHz), memory size (16, 8, 8 GB), and number of processor cores (2, 4, 5).	DeCrypto Pro uses both environmental detection triggers such as high CPU usages or period detection (random / fixed intervals depending on user preference) of cryptomining malware. Once the sampling of data is completed, it is normalized through MinMax feature scaling and	DeCrypto Pro can also be extended to include specific types of APTs for profiling their algorithms. As a future study, in addition to training DeCrypto Pro with more APT classes, we

				self-organizing systems (ACSOS). IEEE, 2020. *https://ieeexplore.ieee.org/abstract/document/9196224*	models (Random Forest and k-Nearest Neighbors) and a deep learning model (LSTM), depending on available computing resources.		compression software (7Zip, SecureZip, PeaZip, WinRAR, WinZip, and Freemake) as our benign examples and cryptomining applications (XMRig, XMR-Stak, Coinhive, Computta, and GUIminer) as malicious example.	Each machine provided a unique signature of operating context since all of them had different applications and services installed, including various versions of drivers.	important features will be selected A model selection utility function considers the computational resources and previous 10-fold cross validation training accuracy and F1 Scores to determine the best model for profiling.	plan to expand on model selection framework to investigate more system setting use cases and integrate explainable AI framework with LSTM model to provide explainability with feature selection, training, and inference.
10.	MR	Detecting Cryptomining Malware: a Deep Learning Approach for Static and Dynamic Analysis	2020	*Darabian, Hamid, et al. "Detecting cryptomining malware: a deep learning approach for static and dynamic analysis." Journal of Grid Computing 18 (2020): 293-303.* *https://link.springer.com/article/10.1007/s10723-020-09510-6*	A deep learning approach to detect cryptomining malware through both static and dynamic analysis to Enhance Malware Detection Accuracy, Reduce False Positives,	Resource Hijacking Stealth Operations Frequent Use of Cryptographic Libraries	Cryptographic Library Usage System Calls Dataset • Dcrypto_sys_calls • Dbenign_sys_calls Opcode Dataset • Dcrypto_opcodes • Dbenign_opcodes	**1.** An Ubuntu 16.04 host system running Cuckoo Sandbox to manage and analyze malware behavior. **2.** A Windows 10 guest system running on VirtualBox for malware execution.	Dynamic Analysis using System Calls 1. Dcrypto_sys_calls 2. `Dbenign_sys_calls` is created using system calls from 220 portable applications using PyWinMonkey. 3. Deep Learning Models • LSTM (Long Short-Term Memory) • Attention-based LSTM (ATT-LSTM) • CNN (Convolutional Neural Network) 4. Evaluated using metrics accuracy, F-measure, MCC (Matthew Correlation Coefficient), and False Positive Rate (FPR). Static Analysis using Opcodes 1. Dcrypto_opcodes 2. Dbenign_opcodes 3. the same deep learning models (LSTM, ATT-LSTM, and CNN) are used for static analysis as for dynamic analysis. 4. Evaluated metrics -Used techniques such as Adam optimizer and dropout to enhance model training and to prevent overfitting. -Performed a 10 times 3-fold cross-validation to ensure robustness and reliability of the models.	these datasets might not cover the full spectrum of cryptomining malware behaviors Model Generalization Dynamic Behavior Variability use of obfuscation, packing, or encryption techniques that make static analysis less effective or more challenging -Expand the dataset to include more samples of cryptomining malware -Enhanced Feature Extraction -Can use Advanced Machine Learning Techniques like ensemble methods, reinforcement learning, and multi-view learning, to further improve the detection accuracy and reduce false positives. -Real-time Detection -evasion techniques used by advanced cryptomining malware to improve the sandbox environment to make it less detectable by malware.
11.	MR	Website Cryptojacking Detection Using Machine Learning	2020, IEEE	*Nukala, Venkata Sai Krishna Avinash. "Website cryptojacking detection using machine learning: IEEE CNS 20 poster." 2020 IEEE Conference on Communications and Network Security (CNS). IEEE, 2020.* *https://ieeexplore.ieee.org/abstract/document/9162342*	-for detecting website cryptojacking using machine learning techniques -monitored the cache activity to detect whether cryptojacking exists or not -to detect the CPU percentage throttle set by the attacker making it a multiclass classification problem	In-browser Website cryptojacking	website cryptojacking malicious JavaScript code into websites CPU Percentage Throttle Code Obfuscation False Positive Cases	the computers or devices of unsuspecting users who visit websites that have been compromised with malicious JavaScript code for the purpose of cryptojacking.	1. Cache Activity Monitoring • cache hits • cache misses -is collected at intervals of 100 milliseconds for a duration of 60 seconds. -100 traces are collected for each class label, resulting in a total of 60,000 data points for analysis. 2. Machine Learning Classification - Models include k-nearest neighbors (KNN), Random Forest, Decision Tree, Support Vector Machine (SVM), and Naïve Bayes. 3. evaluated based on metrics such as accuracy, precision, recall, and F1-score.	to explore cases where cryptojacking occurs alongside other high-performance activities, such as gaming, which may lead to false positives.

									4. The best-performing model, SVM, achieved an accuracy of 96.25%, with high precision, recall, and F1-score values.	
12.	MR	Detecting Illicit Cryptocurrency Mining Activity in Cloud Computing Platform	2022	*Ariffin, Muhammad Azizi Mohd, et al. "Detecting Illicit Cryptocurrency Mining Activity in Cloud Computing Platform." Journal of Positive School Psychology 6.3 (2022): 8611-8622.* *https://journalppw.com/index.php/jpsp/article/view/5126*	aim to address the increasing risk of unauthorized mining operations, which can lead to financial loss for organizations due to increased power consumption and resource utilization. By developing a detection algorithm and testing it on a cloud testbed, the paper seeks to provide a solution to mitigate the impact of illicit mining activities on cloud infrastructure	Illicit Cryptocurrency Mining Spread of Mining Malware Security Breaches	Exploiting Cloud Infrastructure Installing Mining Software Malware Infections Compromising Cloud Platform	the virtual machines (VMs) within the cloud environment	1. Algorithm Selection ● the Alternating Directions Dual Decomposition (AD3) algorithm for anomaly detection. This makes it suitable for analyzing system metrics and distinguishing between normal background processes and illicit mining activities. 2. Data Collection and Preprocessing ● metrics such as CPU utilization, memory usage, disk activity, and network traffic are collected from the cloud platform. These raw metrics undergo feature extraction and selection to identify relevant indicators of illicit mining activity. 3. Standardization and Anomaly Detection ● The collected metrics undergo pre-processing to standardize their values using z-scores. 4. Experimentation and Validation ● The proposed detection method is tested on a cloud testbed running OpenStack	-Lack of Effective Detection Methods -Limited Focus on Cloud Infrastructure -Insufficient Attention to Anomaly Detection Future Works ● Enhancing Detection Techniques ● Experimentation in Hybrid and Containerized Clouds ● Real-time Monitoring and Response ● Mitigating False Positives ● Investigating Novel Attack Vectors
13.	AU	Smart Analysis and Detection System for New Host-Based Cryptojacking Malware Dataset **(DATASET)**t	2023, JEAS	*Almurshid, Hadeel. "Smart Analysis and Detection System for New Host-Based Cryptojacking Malware Dataset." IN THE NAME OF ALLAH, THE MOST GRACIOUS, THE MOST MERCIFUL 10.1 (2023).* *https://m.mu.edu.sa/sites/default/files/2023-05/JEAS%20V%2010%20Issue%201.pdf#page=77*	**up-to-date datase**t consisting of 114,985 samples, with 57,948 categorized as benign and 57,037 as cryptojacking build a smart cryptojacking detection system, with 5 different convolutional neural network models trained and evaluated against a subset of the dataset	Host based cryptojacking	Monitoring Windows executables, which involves running Windows executables in an isolated environment and closely monitoring their CPU usage, provides a thorough understanding of cryptojacking malware behavior and enables detection of the malware	—	5 neural networks models were trained on the sub-set of the dataset to prove the effectiveness of the dataset. The models were evaluated using the basic evaluation metrics: Accuracy, Precision, Recall, etc.	• The dataset has been obtained from a single source, and the cryptojacking samples included in the dataset were restricted to those identified by antivi-rus software in VirusTotal. • Our models were trained on a portion of the dataset, rather than the complete dataset. • Restricting model training to only 5 models • The malware was not statically analyzed The following are potential areas for future investigation: • Increase the number of ML and DL models trained on the complete dataset • Collect host-based cryptojacking mal-ware datasets from multiple sources • Conduct research to comprehensively analyze the impact of cryptojacking malware on the cryptocurrency industry

										• Explore and analyze the different targets of cryptojacking malware. • Examine and analyze the existing techniques used by organizations to detect and prevent cryptojacking malware • Examine and analyze the existing techniques used by organizations to hinder the impact of the cryptojacking malware
14.	AU	MiNo: The Chrome Web Browser Add-on Application to Block the Hidden Cryptocurrency Mining Activities	2020, IEEE	*Aktepe, Safa, Cihan Varol, and Narasimha Shashidhar. "MiNo: The Chrome Web Browser Add-on Application to Block the Hidden Cryptocurrency Mining Activities." 2020 8th International Symposium on Digital Forensics and Security (ISDFS). IEEE, 2020.* *https://ieeexplore.ieee.org/abstract/document/9116443*	MiNo, a web browser add-on application to detect these malicious mining activities running without the user's permission or knowledge. This add-on provides security and efficiency for the computer resources of the internet users. MiNo designed and developed with double-layer protection.	browser -based cryptojacking		Chrome browsers of victim users.	The first control is on detection. It collects the malicious cryptominer scripts and URLs from an external file calls filters.txt inside the assets folder when it is installed into the Chrome browser. These filters in the file have been acquired from various resources including previous works and online resources. Once the MiNo has collected the filters, it can access the resources and requests of the web page with the functions provided by the Chrome API. In this case, MiNo has enough time to check the content before a web request is made by Chrome. Thereupon, MiNo scans the resources of the web page to check if there is any malicious script or URL.	adding a whitelist feature to create and improve a whitelist of trusted web pages which can be controlled by the user to prevent any possible false-positive results without disabling MiNo. The second one is using a longer control time for CPU usage detection function to prevent false-positives in clean websites due to instant high CPU usage.
15.	MR	Detecting Cryptojacking Web Threats: An Approach with Autoencoders and Deep Dense Neural Networks	2022	*Hernandez-Suarez, Aldo, et al. "Detecting cryptojacking web threats: An approach with autoencoders and deep dense neural networks." Applied Sciences 12.7 (2022): 3234.* *https://www.mdpi.com/2076-3417/12/7/3234*	to propose and validate a machine learning-based solution for detecting cryptojacking Combining both network and host-based features to effectively characterize and detect cryptojacking activities	malicious actors inject cryptojacking scripts into websites or use other network-based methods to deliver the payload Once the script or malware reaches the host (the victim's computer or device), it executes and begins utilizing the host's CPU and GPU resources to perform cryptocurrency mining operations.	**Network-Based Attacks**: Malicious actors can exploit vulnerabilities in websites or web applications to inject cryptojacking scripts. These scripts, often written in JavaScript, are designed to run silently in the background when a user visits an infected website. **Host-Based Attacks**: Once the cryptojacking script is executed on the victim's device (host), it starts using the device's resources for mining operations. This can lead to increased CPU/GPU usage, higher power consumption, and potentially slower performance. **Payload Delivery**: The methods used to deliver the cryptojacking payload to the victim's device can vary. It could involve exploiting vulnerabilities in web servers, using phishing emails with malicious attachments, or leveraging other attack vectors.	— —	**Holistic View of Cryptojacking:** The paper takes a comprehensive approach to cryptojacking, focusing on two main attack surfaces: the network (entry point of the threat) and the host (where the malware payload is executed and disseminated). **Feature Extraction and Selection**: A novel technique called Stacked Autoencoder (SAE) is employed for feature extraction and selection. SAE compresses and normalizes the dataset's outputs, retaining the best latent data for variant inputs. **Detection Algorithms Evaluated** Fuzzy C-Means Clustering Support Vector Machines (SVM) Multi-Layer Perceptron (MLP) Random Forest (RF) Multiple-Instance Support Vector Machines (MISVM) Random Subspace (RS) + Decision Tree (DT) Sequential Minimal Optimization (SMO) + SVM Logistic Regression (LR) Convolutional Neural Networks (CNN) and Long Short-Term Memory Networks (LSTM)	compiling the model to enable baseline detection with a sensing agent, capable of recognizing and classifying samples to trigger mitigation or remediation policies in minimal or restricted environments. enhancing the accuracy of cryptojacking detection by refining the detection algorithms and incorporating additional features or data sources. Understanding New Strains of Cryptojacking