energies

MDPI

Article

Do Charging Stations Benefit from Cryptojacking? A Novel Framework for Its Financial Impact Analysis on Electric Vehicles

Asad Waqar Malik [1,2] and Zahid Anwar [2,*]

[1] Department of Computing, National University of Sciences and Technology (NUST), Islamabad 44000, Pakistan
[2] Department of Computer Science, North Dakota State University (NDSU), Fargo, ND 58105, USA
* Correspondence: zahid.anwar@ndsu.edu

Abstract: Electric vehicles (EVs) are becoming popular due to their efficiency, eco-friendliness, and the increasing cost of fossil fuel. EVs support a variety of apps because they house powerful processors and allow for increased connectivity. This makes them an attractive target of stealthy cryptomining malware. Recent incidents demonstrate that both the EV and its communication model are vulnerable to cryptojacking attacks. The goal of this research is to explore the extent to which cryptojacking impacts EVs in terms of recharging and cost. We assert that while cryptojacking provides a financial advantage to attackers, it can severely degrade efficiency and cause battery loss. In this paper we present a simulation model for connected EVs, the cryptomining software, and the road infrastructure. A novel framework is proposed that incorporates these models and allows an objective quantification of the extent of this economic damage and the advantage to the attacker. Our results indicate that batteries of infected cars drain more quickly than those of normal cars, forcing them to return more frequently to the charging station for a recharge. When just 10% of EVs are infected we observed 70.6% more refueling requests. Moreover, if the hacker infects a charging station then he can make a USD 436.4 profit per day from just 32 infected EVs. Overall, our results demonstrate that cryptojackers injected into EVs indirectly provide a financial advantage to the charging stations at the cost of an increased energy strain on society.

Keywords: connected vehicles; cryptojacking; battery life; financial impact

Citation: Malik, A.W.; Anwar, Z. Do Charging Stations Benefit from Cryptojacking? A Novel Framework for Its Financial Impact Analysis on Electric Vehicles. *Energies* **2022**, *15*, 5773. https://doi.org/10.3390/en15165773

Academic Editor: Adolfo Dannier

Received: 16 July 2022
Accepted: 6 August 2022
Published: 9 August 2022

1. Introduction

Congress recently passed the Infrastructure law allocating USD 15 billion for electrical vehicle (EV) charging stations, electric buses, and ferries. To reduce climate change, the government targets that 50% of vehicles sold be electric by 2030. EVs are becoming popular due to their fuel efficiency, eco-friendliness, and because of the increasing cost of fossil fuel. According to the US DoE, between 2015 and 2020, the number of charging stations more than doubled, and in 2021 alone, they grew by over 55%. Modern EVs house powerful processors that support a variety of apps to provide services such as turn-by-turn navigation, remote vehicle diagnostics, and infotainment. Moreover, they can efficiently run third-party applications such as cryptomining. In a recent incident [1], a user hacked his 2018 Tesla Model 3 to mine Ethereum to make USD 800 per month. Some EV companies themselves are designing cars to mine cryptocurrency when parked [2].

Problem statement—Cryptojacking is a malicious activity that entails the unauthorized use of a victim's device resources such as CPU and memory to mine cryptocurrency. This causes a drain on battery life and reduces system performance. Furthermore, if a large number of vehicles are infected, this energy drain would lead to frequent recharge demand and ultimately cause long queues at EV charging stations, thereby negatively impacting society. The goal of this research is to investigate the impact of cryptojacking attacks on EV energy, efficiency of the charging infrastructure, and attacker profit.

Motivation—Many apps on the official Microsoft store and Google Play Store [3] have cryptojacking software hidden inside them. Recently, Norton received furious comments regarding a cryptominer that it installed with its 360 antivirus product subscriptions without clear intimation to the end-user [4]. EVs provide a computing platform for running in-vehicle apps and services, which makes them an appropriate target for hackers looking to launch cryptomining attacks because of several reasons: (1) the advanced onboard computing hardware is ideally suited for this kind of workload, (2) the user interface makes it extremely difficult to detect for people having limited technical knowledge, and (3) currently, there exist limited preventive controls. Attackers have already started launching cryptojacking attacks on EV companies [5].

The EV ecosystem is shown in Figure 1. EVs are essentially mobile computers that can communicate, store, and process data. Hackers can therefore inject malware using a variety of attack vectors, including, but not limited to, vulnerable hotspot connections and OBD dongles, as well as EV communication. There are several examples of such vulnerabilities being exploited in the past, which are detailed next.

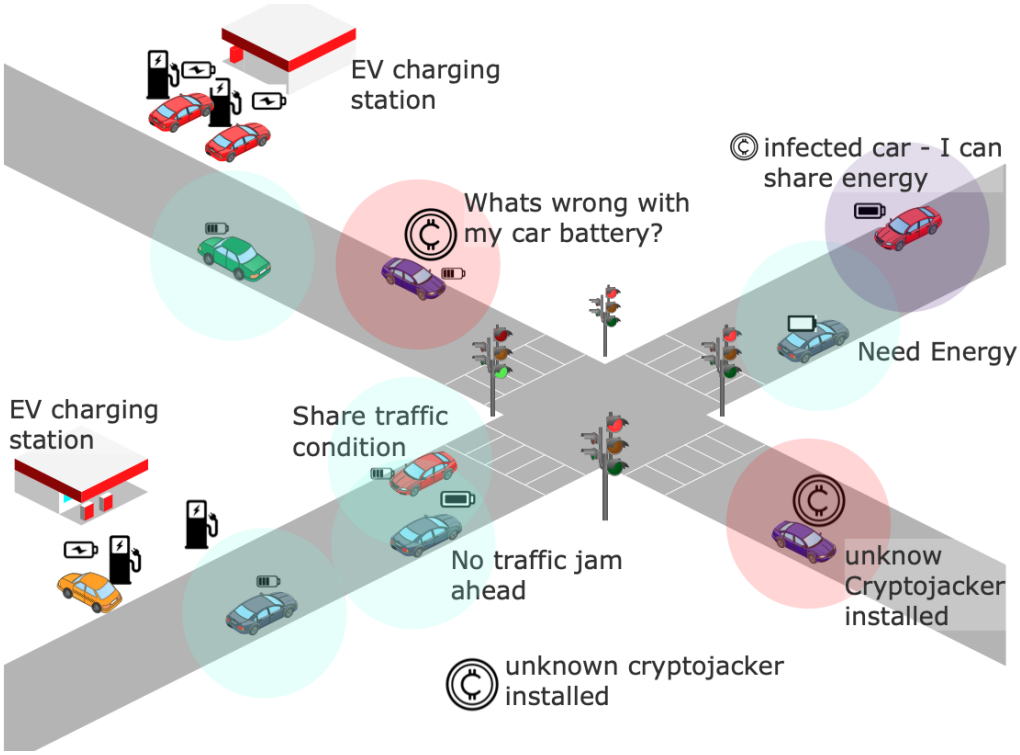


Figure 1. Road network showing cryptojacking infected EVs.

In the past, hackers have used the "TBONE" vulnerability to their advantage [6] in post-2018 Tesla vehicles, whereby these vehicles are configured to constantly scan for a wireless network called the "Tesla Service" using credentials hard-coded into the car's firmware. Since these credentials are shared widely on Twitter and other forums, hackers tricked the vehicle into connecting with rogue access points with the same names. This was achieved by flying a drone close to the car or leaving a router in the vicinity of a parked car. As a result, the hackers were able to remotely unlock the doors, open the charging port, and execute commands, just as a driver could, from the car's infotainment screen. The use of hard-coded, or otherwise called embedded, credentials by automakers makes these vehicles vulnerable to malware such as cryptojackers and opens doors for backdoor access.

Further, at the charging terminal, the charging process takes anywhere from 30 min to an hour. During this time, the driver is typically connected to the station's network for using amenities or downloading songs and apps. However, the network may be compromised. Researchers analyzed the management software as well as mobile and web

applications used by customers to interact with the charger at 16 charging stations. They found several web server vulnerabilities [7] for products by Schneider Electric and other companies. By exploiting these vulnerabilities, hackers can control the charging process, modify firmware settings, change the billing, access confidential information, and even utilize the system for mounting other attacks. Schneider Electric acknowledged these findings and prescribed 12 common vulnerabilities and exposures (CVE) identifiers to the vulnerabilities found for their products. Some stark examples of how charging station vulnerabilities can be used for hacktivism are as follows. During the Russia–Ukraine conflict, a Ukrainian hacker injected an abusive message against Putin [8] into a charging terminal display on a Russian motorway. In another incident, a team at the University of Oxford team demonstrated [9] how miscreants can use malicious radio signals to halt a fleet of electric ambulances from charging at charging terminals. A readily available software-defined radio placed within 50 yards of the charging station can jam the charging of nearby vehicles, as in the *Brokenwire* technique. Its reactivation involves unplugging and then resetting the process. As an example of multiple charging stations being attacked at the same time, consider the Isle of Wight Council incident in England. Obnoxious pornographic material was injected into the display screens of charging points across Quay Road, Ryde, Cross Street, Cowes, and Freshwater [10]. Additionally, the charging stations were made unavailable by hackers. The use of the charging station resulted in rebooting it every time. Furthermore, it is of note that charging station owners may not be very motivated to secure their networks against cryptojackers. Today, some charging stations use economical but insecure hardware such as Raspberry Pi devices to control the chargers which have limited support for secure boot, signed firmware, key storage, hardware encryption, USB port locking, and tamper resistance [11].

Contribution—We develop a novel framework that models the road network, charging station infrastructure, EV mobility, energy, and cryptojacker behavior. In our framework, EVs navigate between source to destination locations using predetermined paths taken from the Yellow Taxi trip [12] record dataset provided by the NYC government and use up energy. When the battery level is low, they park at the nearest charging station to recharge, and are infected based on the threat model assumptions described below. If EVs are unable to find a charging terminal, they try a different station or wait for their turn. The framework allows attackers to control the rate of cryptomining either remotely or through predefined configurations. The higher the throttling rate, the more the number hashes calculated per time unit, and the higher the battery drain. Hence, a cryptojacker running at full throttle rate runs the risk that it may be detected more easily by the EV owner/driver. We introduced a factor called *mining* to control this rate and observe the stealth behavior. A higher *mining* rate can help to detect cryptojacking attacks; however, a careful selection can provide a stealthy environment for longer persistence inside EVs. In this regard, we observed the impact of the *mining* rate on EV energy and hackers' profit.

We observe that the batteries of infected cars drain more quickly than those of normal cars, forcing them to return more frequently to the same charging station for a recharge. At a 10% arrival rate, we observed 70.6% more refueling requests. Moreover, if the hacker infects a charging station, then at this arrival rate he can make a USD 436.4 profit per day from 32 infected EVs. Thus, it is an advantage to charging station owners if they collude with the attackers to become points of infection. Our results are consistent with research works on browser-based cryptojacking [13] that show that mostly third-party websites are compromised by hackers to deliver cryptojackers. However, unlike browser cryptojacking, EV cryptojacking provides an additional incentive to the charging stations in that they receive an increased number of returning customers. While there is no evidence to suggest that charging station owners are unethical, our goal is to show that EV cryptojacking is a very real threat, and vulnerable charging stations may help spread infections. Ultimately, this may undermine the noble cause of reducing climate change that the initiative set out to accomplish.

Our contributions are as follows.

1. To the best of our knowledge, this is the first work to highlight EV cryptojacking as a real threat to the electric charging infrastructure and describe its behavior in detail.
2. A novel framework is proposed that incorporates an EV energy model to analyze cryptojacking behavior to allow researchers to study the financial impacts of attacks on different infrastructural configurations. Moreover, various scenarios with different configurations can be tested using the proposed framework.
3. Our results are significant in that they highlight how cryptojacking can severely impact the EV battery's residual energy, leading to increased trips to charging stations, incur financial burden on EV owners, and provide significant profits to the attacker. Furthermore, at low levels of mining rate (50% or less), the cryptojacking can proceed quite stealthily while still providing the attacker sizable profits.
4. A practical set of countermeasures outlined that can mitigate these threats and allow the vision of green energy in reducing climate change to materialize.

Paper organization—The rest of the paper is structured as follows: Section 2 covers the related work. The system model is presented in Section 3. Section 4 covers the proposed framework, while system evaluation is covered in Section 5. Finally, the discussion and conclusion are presented in Sections 6 and 7.

2. Related Work

Although we found no research related to an analysis of EV cryptojacking, our research benefits from works related to general cryptojacking techniques. The closest works in this area are summarized below.

In [14], they analyze in-browser mining trends based on Monero cryptocurrency. In this case, the user visiting a website pushes JavaScript code that executes stealthily in the browser to mine cryptocurrency. The authors outline an ethical framework to review it either as an attack or as a business opportunity. Hong et al. [15] propose a tracker to detect browser cryptojacker behavior. They discovered 2770 cryptojacking samples from 800k websites. This information is used to develop a comprehensive cryptojacking attack scenario that includes the attack and its distribution mechanism. However, to stay in stealth mode, they update their attack domains very frequently. Varlioglu et al. [16] explored cryptojacking after Coinhive discontinued its services. The authors manually examined the websites detected in [15] and concluded that 99% of the sites no longer continue cryptojacking. However, tracking the remaining 1% websites, the authors discovered that 600+ were unique. There also exist some works that have examined in-browser cryptocurrency mining in terms of their deployment, expansion, trends, and organized structures [13,17].

Tekiner et al. [18] presented a systematic study on emerging cryptojacking. The authors explored the challenges in cryptojacking detection and highlight the vulnerabilities in the system. Musch et al. [19] propose a three-phase analysis approach to identify mining scripts. The authors conclude that cryptojacking is common, with every 500th site hosting cryptojacking malware. Thus, the probability of falling victim to a mining attack is considerable for precautionary measurements that they conducted. Authors also show that cryptojacking drains a significant amount of energy.

Gomes et al. [20] proposed a machine-learning-based technique that monitors the CPU for cryptojacking detection. The authors argue that the CPU usage of content enriched sites may be higher; therefore, reliance on CPU monitoring alone is not an accurate solution for the detection process. However, combining it with machine learning techniques improves accuracy. In [21], authors proposed a framework termed Mininghunter to track mining scripts. The framework records all the web traffic that is later compared with others to identify the pattern based on the repeated keys, thus, classifying the mining camps. Similarly, in [22], the authors observed that cryptojacking execution establishes a bidirectional WebSocket with the remote server and this communication is easy to monitor through third-party software. The authors state that browser encryption makes it difficult to analyze this communication and identify cryptojacking.

Security for vehicles and IoT devices—Bajpai et al. [23] conducted a vulnerability analysis of the QNX platform which is an in-vehicle infotainment (IVI) system used in a variety of modern cars. They then performed experiments to exploit these vulnerabilities and were able to successfully conduct three specific types of attacks. The first was a proof-of-concept crypto-ransomware that encrypts data on the IVI system. The second was a resource exhaustion attack using a fork bomb, and finally they were able to execute code by exploiting an exposed service on the IVI system. The authors conclude that the same attacks are applicable to other IVI systems due to the lack of inherent protections against malicious activity in most real-time operating systems. They suggest that communication between the IVI system and the vehicle's controller area network (CAN) be kept to a minimum to prevent malware from crossing over. In [24], authors proposed the C4IoT framework that combines elements of security contracts and fog computing to handle security issues in IoT devices. The authors claim that IoT devices demand a detailed behavioral analysis in order to adopt a secure default configuration and models of self-configuration. Carlos et al. [25] proposed the BIoTS hardware design to introduce the blockchain architecture and security requirements in IoT devices. The main objective is to provide security to the sensor devices as they have limited processing and data storage capability which makes them vulnerable to data breaches. In [26], authors presented a secure IoT framework based on zero trust and blockchain. Zero trust architecture enhances the security aspect beyond the closed network, and the adoption of blockchain allows for improved device authentication, leading to more robust access control.

Discussion—Table 1 shows the comparison among proposed and existing work. The work highlighted is mostly focused on web-browser-based detection schemes, and, to a smaller extent, Android mobile devices are considered. However, the impact of cryptojacking on electric vehicles has not been explored. According to our literature review, this is the first work focused on cryptojacking over electric vehicles to observe its financial implications and impact on the efficiency of charging stations and the society as a whole. Electric vehicles and charging stations are becoming an important part of nations' cyber-physical systems. Thus, the vulnerability in any component can impact the entire society. In this work, we analyze the impact of cryptojacking in terms of demand–supply, financial loss, and overall system efficiency.

Table 1. Comparison showing recent advancement in cryptojacking. H: hackers, CS: charging stations.

Authors	Detection and Mitigation Technique	Focus Area	Financial Impact	Societal Impact	Beneficiary	Efficiency
Tekiner et al. [18]	Review work	System Browser	×	×	H	×
Stanislav et al. [27]	Dynamic metrics	Android devices	×	×	NA	×
Gomes et al. [20]	Machine learning	Web Browser	×	×	H	×
Rauchberger et al. [21]	Monitoring Web Traffic	Web Browser	×	×	H	×
Wang et al. [28]	Byte-code inspection	Web Browser	×	×	H	×
Yulianto et al. [29]	Taint analysis method	Web Browser	×	×	H	×
Nada et al. [30]	Throttling evasion tech.	App. agnostics	×	×	NA	×
Romano et al. [31]	WebAssembly, JavaScript	Web Browser	×	×	NA	×
Proposed Work	NA	Electric Vehicles	✓	✓	H/CS	✓

3. System Model

At a high level, our system model considers r charging stations (a set $\mathbb{F}$) placed at various locations to facilitate EVs. The energy and adversary model is described below.

3.1. Energy Model

A power-based EV energy consumption model is considered that does not require collecting vehicle-specific field data. It simply requires the instantaneous speed, acceleration levels, and the EV characteristics and outputs the energy consumption in kW h/km of the vehicle for a specific drive cycle, the instantaneous power consumed (in kW), and the state of charge (SOC) of the electric battery in percentage (%). Assuming there are n electric vehicles with c charging capacity each, N represents the total number of EVs such that $\{1 \dots n\} \in N$. EVs are equipped with an onboard computing system, which requires E_b energy to operate. E_m represents the energy consumed due to mileage. Thus, the total energy can be calculated as in Equation (1).

$$E_t = E_m + E_b \tag{1}$$

E_b depends on the available cores, and number of jobs required to execute per unit time. EVs use a multicore architecture (represented as $\{1 \dots m\} \in M$) designed for replication and redundancy. The process can utilize the m available cores for computation. Every core can execute q million instructions per second (MIPS). Therefore, the total execution time is [32]:

$$T = \sum_{k=0}^{m} \sum_{i=0}^{z} J_i / M_k \tag{2}$$

Here, z is the total number of jobs, and J_i represents the mips of the ith job. Here, M_k is the processing capacity of the kth core. The total energy per core including the idle state is computed as

$$E_b = \sum_{j=0}^{m} T_j * p + p' \times t_i \tag{3}$$

Here, p and p' represent the power consumption during peak and idle time and t_i represents the core idle time.

The power at wheels (P_w) can be computed using EV features as input; the output is the energy consumption and the battery's charge state (represented as SoC). The power at wheel at time t can be computed using Equation (4) [33]:

$$P_w(t) = [P_1(t) + P_2(t)].v(t) \tag{4}$$

Here, P_1 and P_2 are defined using Equations (5) and (6):

$$P_1(t) = \wp\alpha(t) + \wp g.cos(\theta).(\mathcal{R}_r/1000).(r_1 v(t) + r_2) \tag{5}$$

$$P_2(t) = \frac{1}{2}.\rho_{air}.\mathcal{A}_f C_D v^2(t) + \wp g.sin(\theta) \tag{6}$$

Here, the $\wp$ represents the EV mass, and $\alpha(t)$ is the acceleration and deceleration. The g is gravitational acceleration, θ is road grade, $\mathcal{R}_r$, r_1, and r_2 are the rolling resistance, with their values depending on road surface condition and tire type. ρ_{air} represents the air mass density, A_f denotes the EV's front area, C_D the drag coefficient, and $v(t)$ is the EV speed at instant t in m/s. Further, the battery state of charge (SoC) may be estimated via Equation (7).

$$SoC(t) = SoC_i - \sum_{i=1}^{N} \Delta SoC_i(t) \tag{7}$$

$$\Delta SoC_i(t) = SoC_{i-1}(t) - \frac{P_{EM}(t)}{3600.\mathbb{Q}_b} \tag{8}$$

$$P_{EM}(t) = [SoC_{i-1}(t) - \Delta SoC_i(t)] * 3600 * \mathbb{Q}_b \tag{9}$$

Here, P_{EM} is the consumed electric power that includes the power required by the auxiliary system, and $\mathbb{Q}$ is the battery capacity. Further, the EV operates so as to maintain the SoC at a minimum of 20% to ensure battery safety. Thus, using the SoC, the energy consumption for a distance (d in m/s) can be computed as (10):

$$E_m[\frac{kWh}{km}] = \frac{1}{3,600,000}.\int_0^t P_{EM}(t)dt.\frac{1}{d} \tag{10}$$

The P_w can be used to compute the maximum energy available to be recovered (E_r) during braking, computed as Equation (11).

$$E_r = \int^t P_w^{-1}(t).dt \tag{11}$$

As stated, P_w can be used to find the recoverable energy generated during braking the EVs. However, we are not considering it at this time to compute the total available energy.

3.2. Cryptojacking Adversary Model

The attack proceeds as follows: (1) A driver docks an EV at a charging terminal and connects to the network for amenities via an in-vehicle app. (2) The attacker records the vehicle information and then embeds a "cryptojacking" process (ç) into the requested app. (3) After charging is complete, the driver drives off, unaware that the EV has become a victim "zombie", allowing the miner to run stealthily as a daemon process. (4) The miner performs computationally complex tasks computing hashes while utilizing EV energy. (5) The hash results are later offloaded by the attacker based on the vehicle information collected in step 2. The battery drainage (B_d) due to cryptojacking can then be represented as $B_d = B_i - B_c$. Here, B_i is the initial SoC and B_c is energy consumed due to ç. However, if all the cores/GPU are utilized by the ç, the B_d factor increases such as $M \times B_c$. Let us assume that t_r is the time required to recover the charge, and $W = t_r \times B_c$. Here, W_i is the power consumed when cryptojacking is active in V_i. Assuming that $\mathcal{P}$ is the cost of electricity in kWh then the total battery loss—the additional money in USD that all the EV owners pay due to cryptojacking—can be computed as

$$Total\ Loss = \mathcal{P} \times \sum_{i=0}^{n} W_i \tag{12}$$

3.3. Efficiency

The efficiency is determined as the ratio of charging requests accepted to the overall number of requests received to serve from the nearby charging stations. It is an important matrix to measure the impact of cryptojacking on vicinity stations, which may result in higher price based on the demand factor. The efficiency is formulated as

$$Eff = \frac{\sum_{i=0}^{r} S_i}{\sum_{j=0}^{r} R_j} \times 100 \tag{13}$$

Here, S and R represent successful recharging requests and the total requests generated.

4. Simulation-Based Cryptojacking Framework

The proposed simulation framework is designed to simulate the cryptojacking attack on electric vehicles. The framework allows electric vehicles to move on the road; take a path to the destination based on the road traffic. Further, there are intersections on the road, where vehicles slow down momentarily before crossing. The framework is provisioned to execute the simulation with varying vehicle arrival rates in the simulation. It also facilitates adding the charging stations at particular locations along with defined charging spots.

The modules of the proposed framework are shown in Figure 2. The manager module is responsible for assigning resources to the applications, including storage and GPU/CPU. The storage module is used for data persistence. The EV battery is utilized by the engine module as well as for software execution. The cryptojacked application executes and mines hashes which are offloaded to the hackers' system. The framework allows EVs to communicate with nearby charging stations through dedicated short-range communication. This communication is used to identify available charging slots. Here, the charging stations act as passive entities that can share available slots information on request. The energy wasted due to cryptomining is estimated in terms of hashes computed. Each EV has a built-in GPU which is utilized to compute the hashes through the mining process. The framework facilitates the simulation study by allowing the user to vary the number of crypto-infected vehicles. The infection rate must be defined in the framework configuration file along with other parameters, such as mining rate, EV arrival rate (λ), etc.

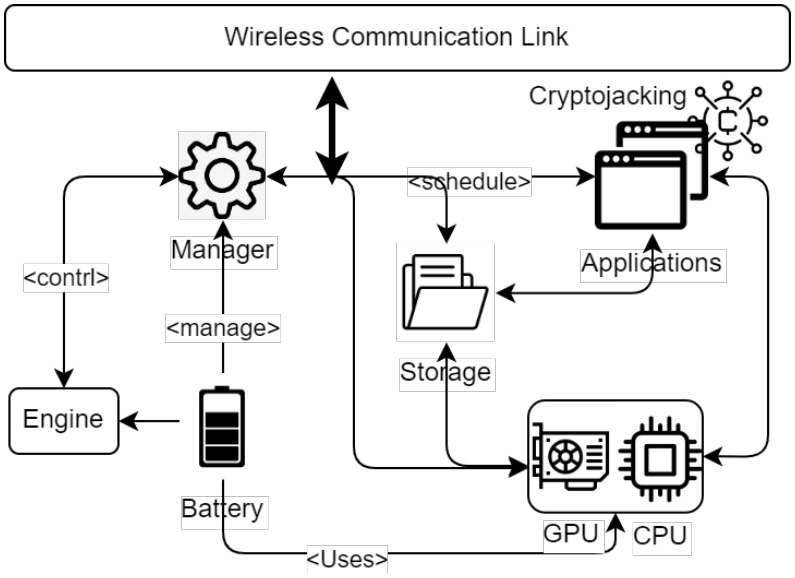


Figure 2. Architecture of the proposed EV framework illustrating module interactions.

The framework is developed using Anylogic, which is a discrete agent-based simulation framework. EVs are modeled as autonomous agents simulated using the New York Taxi dataset [12]. Using dedicated short-range communication, EVs can sense nearby vehicles via heartbeat messages as well as share data with nearby charging stations. The charging stations are placed near intersections with limited energy refueling spots. We assume that the recharging time is the same in all charging stations (shown in Table 2). The Manhattan road network is simulated, with bidirectional road traffic. The vehicle arrival rate is varied to generate the traffic workload. Further, the evaluation is performed with an initial 10–40% cryptojacking infected vehicles to observe the resulting increase in recharging demand, the additional loss incurred, system efficiency, and profit to the attacker.

Table 2. Simulation config and system specification.

Description	Value
Simulation area	3 by 3 km^2
Total simulation time	24 h
Simulation repetition	5 (five) times
Vehicle speed	10–60 km/h
Vehicle acceleration/deceleration	1.6/2.6 m/s^2
Vehicle arrival rate	100–400
Mobility model	New York Taxi dataset
Charging stations	9
Road network	Manhattan
GPU power	61.01 MH/s
Energy per MH/s	1.77 W
Charging spot per station	3–6
Charging time	30–40 time units
Ç̈rypto Infected vehicles	10–40%
Mining rate	0–100%
System	1.4 GHz Quad-Core Intel Core i5
RAM	8 GB
OS	macOS Catalina
Simulator	AnyLogic PLE v8.5

5. Evaluation

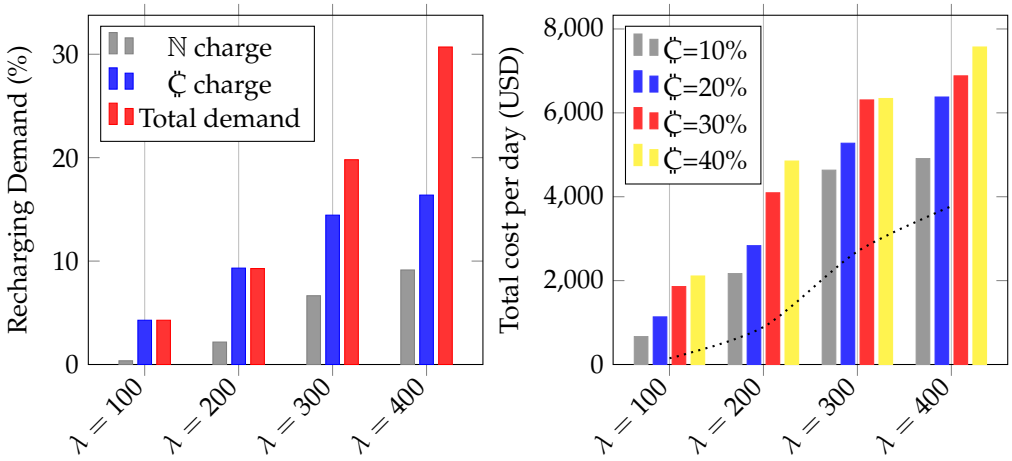
This section covers the evaluation results measured in terms of charging demand, cost, energy consumption due to cryptomining, and hackers' profit. The parameters used for the simulation are listed in Table 2.

5.1. Recharging Demand

Our results show that cryptojacking attacks decrease residual energy which increases the frequency at which infected EVs recharge, and thereby this increases the demand on charging stations. Figure 3a illustrates this increased demand in terms of recharging requests with increasing EVs arrival rate. The demand variation is computed as the sum of recharge requests, recharge successfully completed, and recharge rejected due to the unavailability of vacant spots at the requested time. Note that the demand is consistently lower for the normal scenario ($\mathbb{N}$) than it is when Ç̈ryptojacking is occurring. In this figure, consider the case where $\lambda = 400$. Initially, the charging rate noted for the normal case is approximately 9.14 (the gray bar), which increases to 16.38 when cryptojacking is employed (the blue bar). The rate difference is computed as the ratio of the difference to the new value i.e., $\frac{16.38-9.14}{16.38}$. Therefore, 44% more EVs request a recharge. Further, when the arrival rate increases, the recharge request rejection rate at the nearby charging stations swells to 46%.

5.2. Total Charging Cost

The impact of cryptojacking in terms of the additional cost paid due to recharging is shown in Figure 3b. Here, it is assumed that a complete charge incurs of USD 11.47 based on the current market rate for the mid-size Tesla 3.0 model. The impact of cost is measured with varying EV arrival rates in the simulation model. The dotted line shows the normal scenario where there is no cryptojacker installed, whereas the colored bars represent the impact with 10% to 40% cryptojacked EVs. A trend is clearly visible whereby as the EV arrival increases, the total charging cost increases as well. Thus, at $\lambda = 400$, the increase observed is 35% more than the normal scenario. Admittedly, at $\lambda = 300$, the difference between cost at 30% and 40% cryptojacked vehicles appears to be small (USD 300) and is hard to discern in the figure. We attribute this to the fact that the EV generation is based on the Poisson distribution, and that in this particular experiment some vehicles happened to randomly select shorter paths to the destination, masking the increase in recharging behavior.



(**a**) Avg. charging requirement per day (**b**) Total cost spend per day in recharging

Figure 3. Showing the demand fluctuation due to cryptojacking, the overall cost EV owners paid (loss), and system efficiency.

5.3. Efficiency

The system efficiency reflects the number of recharge requests successfully entertained by a nearby charging station. In Figure 4, the x-axis represents crypto-infected EVs. At $\lambda = 100$, due to the low number of EVs in the simulation, all recharge requests can be easily entertained by nearby stations, even when 50% of these vehicles are compromised. However, with an increase in the EV arrival rate, a clear dip in system efficiency can been observed. With $\lambda = 400$, the number of recharge requests increases manyfold, and in most cases, vacant spots are unavailable. This results in a significant financial impact on EV users.

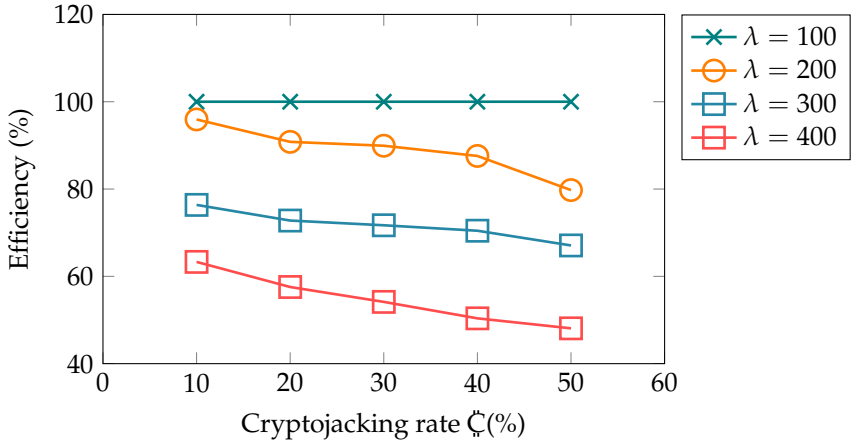


Figure 4. Showing the impact on system efficiency with varying cryptojacking rate.

5.4. Çrypto Energy Consumption

The main purpose of cryptojacking is to compute hashes using the energy and compute resources of electric vehicles. This waste of energy has a direct consequence on the charging stations, thus causing a congested situation at charging stations. Figure 5 shows the energy consumption due to cryptojacking with variable EVs arrival rate in the simulation. Here, we assumed that every EV has a graphical processing unit able to compute a million hashes per second. The GPU processing capability and energy consumed per million hashes are listed in Table 2. The figure shows the increasing trend in all cases, with λ = 400, and 40% initial crypto-infected vehicles causing an overall significant energy consumption.

Thus, with more energy consumption due to cryptomining, more recharging requests are generated on nearby stations.

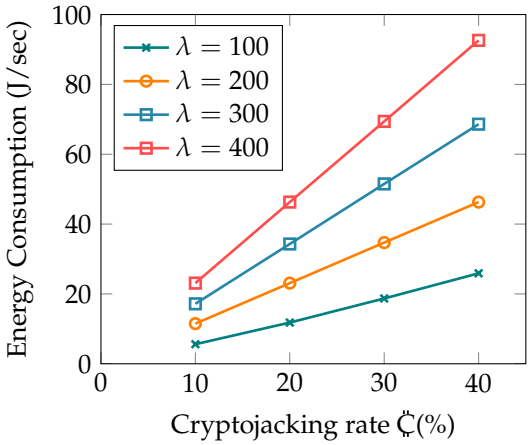


Figure 5. Cryptojacking energy consumption.

5.5. Cryptojacking Impact under Stealth Mode

This subsection covers the impact of the mining rate that may have a direct impact on detection strategies. Two main parameters are explored, namely, energy and hackers' profit. The results reported are computed with varying mining rates termed as α = [0,1]. Here, α = 1 means that cryptomining executes the entire time the vehicle is active in a simulation, whereas lower values of α limit the mining rate. The purpose of using α is to allow the malware to persist longer in the EV by introducing stealth capability in cryptojacking attacks. At higher α values, more energy is consumed; thus, it becomes relatively easy to detect such attacks on electric vehicles. A conservative selection negatively impacts the hackers' profit but the malware can maintain persistence for longer periods of time.

The baseline graph of energy usage at varying mining rates is shown in Figure 6, constructed using the estimated values given in [34]. The gray highlighted region depicts the baseline battery performance when the EV is driving normally. The green region above represents the increased drain when driving on a similar pattern but with a cryptojacker running at different mining rates. To compute the profit value, let P be the profit earned for a cryptojacking activity of Δt sec. Let h be the hash rate of the device in hashes/second. The marketplace pays USD 223.19 per XMR (where XMR is currency unit measured in terms of hashes; here we assumed 1 MH = 0.179). Therefore, the profit P earned in $\Delta t = 130$, where Δt represents the cryptomining time of a session, is computed as [35] $P(XMR) = 0.179 \times (\alpha \times hashes) \times \Delta t / 10^5$.

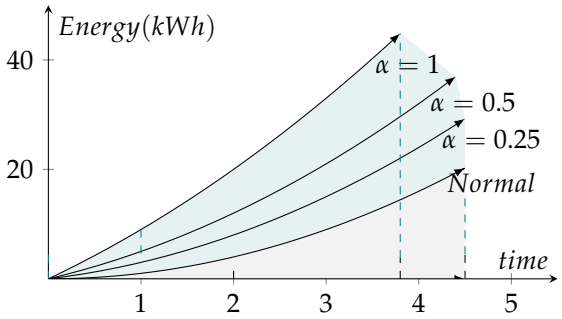
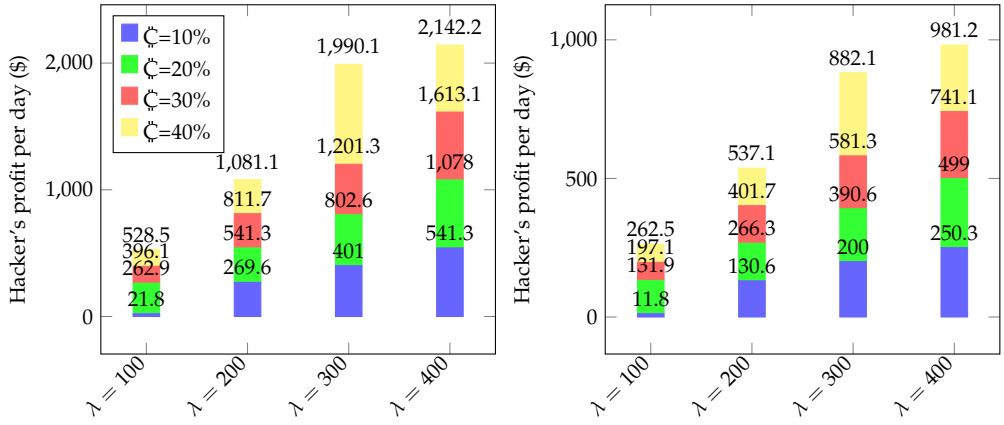


Figure 6. Impact of cryptojacking, showing energy consumption with cryptojacking at different throttles.

Figure 7 shows the hacker's profit per day with a variable arrival rate. The profit reported in Figure 7a uses α = 1, whereas Figure 7b shows the hacker's profit with α value

equal to 0.5. Infected vehicles with cryptojacking executing at a maximum mining rate of $\alpha = 1$ provides the maximum earnings. With 40% cryptojacked vehicles, and $\lambda = 100$, the earning reported is USD 528.5; which reaches USD 2142.2 at $\lambda = 400$. Similarly, the profit value reduces to USD 262.5 at $\lambda = 100$ with $\alpha = 0.5$. Thus, the tradeoff in profit is the stealth operation, which can help the malware to stay undetected for a longer period of time.



(**a**) Mining rate α=1 (**b**) Mining rate α=0.5

Figure 7. Earnings per day with variable cryptojacked infected vehicles with varying α values.

5.6. Ç̈ryptojacking Attack through Charging Station

Our experiments have so far assumed that at the start of simulation a certain proportion of EVs are already infected (determined by the infection rate) by having previously visited an infected charging station or otherwise. Here, we are interested in understanding the spread behavior in a congested urban environment by introducing a compromised charging station where EVs can dock and inadvertantly become infected during the course of the simulation.

Figure 8 shows the ratio of infected and uninfected vehicles. Initially, no vehicle is crypto-infected, at $\lambda = 100$, 173 vehicles are infected; whereas in a more congested environment with $\lambda = 400$, 670 vehicles are infected. It is worth noting that this high rate of spread depicts the infection through just a single compromised charging station. This can increase manyfold if charging stations decide to collude to infect EVs.

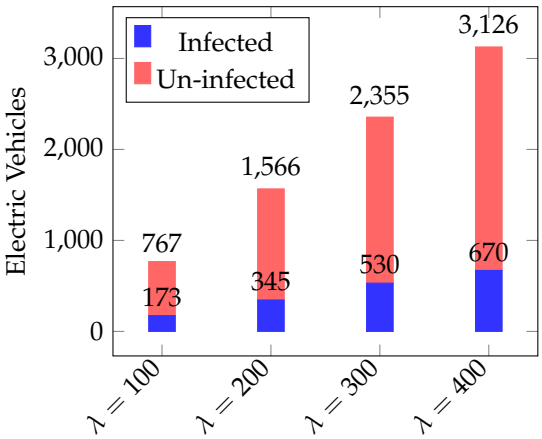


Figure 8. Infection spread through a compromised charging station.

6. Discussion

Our results indicate that infected EVs require frequent charging, which may cause price hikes with increased demand. Managing a higher influx with limited spots is challeng-

ing and leads to longer waiting times. This creates a societal imbalance where normal users are unable to obtain a timely recharge. The beneficiaries of cryptojacking are not only the hackers but also the charging stations, as they earn more profit. Thus, without proper regulations, there are chances that the charging stations may become the source of cryptojacking attacks. EV owners having limited technical knowledge are soft targets for cryptojacking attacks. An EV needs lower maintenance than a car that relies on mechanical moving parts, which reduces routine trips to the service station. This in turn reduces the chances that malicious activities would be detected by professional servicemen at a repair shop. An EV battery is designed to last for a certain number of complete charging and discharging cycles. A lithium-ion battery lasts for approximately 1000 to 1500 cycles. Cryptojacking will cause owners to charge the EV battery more frequently, placing stress on the battery, reducing its capacity by a fraction each time. Compounding over time, the battery's lifespan will reduce, forcing frequent battery replacement, and place extra financial burden on the EV owner.

There is a strong need for proactive defense, for which two strategies are suggested in Table 3. The first focuses on security, hardening the charging station design, and the other, the EV's resistance to cryptojacking. Charging stations should use hardware that relies on a secure root of trust such as a trusted platform module (TPM). This would allow secure boot and ensure that the operating system and apps are trusted. The hardware itself should be tamper-resistant so that hackers may not substitute their own components or conduct man-in-the-middle attacks by intercepting the communications. Since an infected charging station can communicate with an EV, there is a need for protocols that allow for mutual authentication and encryption of the information communicated. EVs themselves should not blindly rely on the app store providing them with high-integrity apps but rather should self-verify and monitor apps for malicious behavior. Apps with embedded cryptojacking functionality will use more CPU and memory than normal apps, which may be detected using machine learning algorithms. Code signing and verification should also be supported.

Table 3. Defensive strategies.

Category	Description
Charging Station Design	1. Utilize hardware in charging terminals that is resistant to physical attacks and allows for secure boot. 2. The procedure for charging an EV at a public charging point must allow for the identification, authentication, and safeguarding of information that passes between the charger and the vehicle. This will require cryptography.
EV Intrusion Detection Mechanisms	1. EVs should deploy their own intrusion detection systems that monitor apps for spike in CPU usage, decrease in performance, and overheating, using machine learning and AI techniques. 2. Provision of security features in EV apps that allow for overlay protection, root detection, and code integrity checks.

7. Conclusions

The proposed framework is designed to study the impact of cryptojacking attacks on electric vehicles. These attacks are analyzed to see the financial impact from a wide perspective. According to our literature review, this is the first contribution that not only covers cryptojacking attacks on EVs but also measures the impact from various societal aspects. The results indicate that EV cryptojacking attacks impact society in terms of sustainability, which can directly impact the demand and supply. Thus, a more organized attack can cause an extreme situation. The experimental analysis highlights the monetary benefit for the attackers. Further, the framework lays the foundation to build more sophisticated attacks on EVs. In the future, we will enhance our framework to include mitigation strategies and further attack scenarios. Moreover, we are also interested in

extending this work towards charging station spots scheduling and exploring the spread of infection via vehicle-to-vehicle communication.

Future Directions

In the future, we are planning to extend this work to propose various mitigation strategies which help identify cryptojacking behavior or block the malware injection. We believe it is an open area for researchers to explore. One such direction could be the use of multidimensional maps [36] that can emit more signals than the hacker can process, making it difficult to inject malware. Similarly, the role of genetic algorithms can be explored to identify the malicious processes running inside the electric vehicles as well as optimize energy [37].

Author Contributions: Both the authors contributed equally. All authors have read and agreed to the published version of the manuscript.

Funding: Funding provided by the Sheila and Robert Challey Institute for Global Innovation and Growth at North Dakota State University, USA.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Sigalos, M. This Tesla Owner Says He Mines Up to $800 a Month in Cryptocurrency with His Car. 2022. Available online: https://www.cnbc.com/2022/01/08/tesla-owner-mines-bitcoin-ethereum-with-his-car.html (accessed on 1 August 2022).
2. Robertson, H. An EV Company Is Planning a Crypto-Mining Car That Will Dig for Bitcoin While Parked. 2021. Available online: www.businessinsider.in/cryptocurrency/news/an-ev-company-is-planning-a-crypto-mining-car-that-will-dig-for-bitcoin-and-dogecoin-while-parked/articleshow/83171598.cms (accessed on 29 July 2022).
3. Google Play and Microsoft Stores Delete Suspected Compromised Apps. 2019. Available online: https://news.trendmicro.com/2019/04/15/google-play-and-microsoft-stores-delete-suspected-compromised-apps/ (accessed on 29 July 2022).
4. Clark, M. Here's the Truth about the Crypto Miner That Comes with Norton Antivirus. 2022. Available online: www.theverge.com/2022/1/7/22869528/norton-crypto-miner-security-software-reaction (accessed on 29 July 2022).
5. Newman, L. Hackers Enlisted Tesla's Public Cloud to Mine Cryptocurrency. 2018. Available online: www.wired.com/story/cryptojacking-tesla-amazon-cloud/ (accessed on 1 August 2022).
6. Stumpf, R. Researchers Used a Drone and a WiFi Dongle to Break into a Tesla. 2021. Available online: www.thedrive.com/tech/40438/researchers-used-a-drone-and-a-wifi-dongle-to-break-into-a-tesla (accessed on 1 August 2022).
7. Nasr, T.; Torabi, S.; Bou-Harb, E.; Fachkha, C.; Assi, C. Power jacking your station: In-depth security analysis of electric vehicle charging station management systems. *Comput. Secur.* **2022**, *112*, 102511. [CrossRef]
8. Knott, M. Russia's Great Firewall: As Putin Clamps Down, Activists Get Creative. 2022. Available online: www.smh.com.au/world/europe/russia-s-great-firewall-as-putin-clamps-down-activists-get-creative-20220308-p5a2n7.html (accessed on 15 July 2022).
9. Corfield, G. Security Flaws Leaves Electric Cars at Risk of Cyber Hacks. 2022. Available online: www.telegraph.co.uk/business/2022/03/29/security-flaws-leaves-electric-cars-risk-cyber-hacks/ (accessed on 28 July 2022).
10. David, B. Electric Vehicle Chargers Hacked to Show Porn. 2022. Available online: www.infosecurity-magazine.com/news/electric-vehicle-chargers-hacked/ (accessed on 28 July 2022).
11. Alamalhodaei, A. Security Flaws Found in Popular EV Chargers. 2021. Available online: https://techcrunch.com/2021/08/03/security-flaws-found-in-popular-ev-chargers/ (accessed on 20 July 2022).
12. City of New York. ETLC Trip Record Data. 2022. Available online: www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page (accessed on 30 June 2022).
13. Bijmans, H.L.; Booij, T.M.; Doerr, C. Inadvertently making cyber criminals rich: A comprehensive study of cryptojacking campaigns at internet scale. In Proceedings of the USENIX Security, Santa Clara, CA, USA, 14–16 August 2019.
14. Eskandari, S.; Leoutsarakos, A.; Mursch, T.; Clark, J. A first look at browser-based cryptojacking. In Proceedings of the European Symposium on Security & Privacy Workshops, London, UK, 23–27 April 2018.
15. Hong, G.; Yang, Z.; Yang, S.; Zhang, L.; Nan, Y.; Zhang, Z.; Yang, M.; Zhang, Y.; Qian, Z.; Duan, H. How you get shot in the back: A systematical study about cryptojacking in the real world. In Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, Toronto, ON, Canada, 15–19 October 2018; pp. 1701–1713.
16. Varlioglu, S.; Gonen, B.; Ozer, M.; Bastug, M. Is cryptojacking dead after coinhive shutdown? In Proceedings of the International Conference on Information and Computer Technologies, San Jose, CA, USA, 9–12 March 2020.
17. Marchetto, V.; Liu, X. An investigation of cryptojacking: Malware analysis and defense strategies. *J. Strateg. Innov. Sustain.* **2019**, *14*, 66–80.

18. Tekiner, E.; Acar, A.; Uluagac, A.S.; Kirda, E.; Selcuk, A.A. SoK: Cryptojacking Malware. In Proceedings of the 2021 IEEE European Symposium on Security and Privacy (EuroS&P), Vienna, Austria, 6–10 September 2021; pp. 120–139.
19. Musch, M.; Wressnegger, C.; Johns, M.; Rieck, K. Thieves in the browser: Web-based cryptojacking in the wild. In Proceedings of the 14th International Conference on Availability, Reliability and Security, Canterbury, UK, 26–29 August 2019; pp. 1–10.
20. Gomes, F.; Correia, M. Cryptojacking detection with cpu usage metrics. In Proceedings of the 2020 IEEE 19th International Symposium on Network Computing and Applications (NCA), Cambridge, MA, USA, 24–27 November 2020; pp. 1–10.
21. Rauchberger, J.; Schrittwieser, S.; Dam, T.; Luh, R.; Buhov, D.; Pötzelsberger, G.; Kim, H. The other side of the coin: A framework for detecting and analyzing web-based cryptocurrency mining campaigns. In Proceedings of the 13th International Conference on Availability, Reliability and Security, Hamburg, Germany, 27–30 August 2018; pp. 1–10.
22. Saad, M.; Khormali, A.; Mohaisen, A. End-to-end analysis of in-browser cryptojacking. *arXiv* **2018**, arXiv:1809.02152.
23. Bajpai, P.; Enbody, R.; Cheng, B.H. Ransomware targeting automobiles. In Proceedings of the Second ACM Workshop on Automotive and Aerial Vehicle Security, New Orleans, LA, USA, 18 March 2020; pp. 23–29.
24. Giaretta, A.; Dragoni, N.; Massacci, F. S× C4IoT: A Security-by-contract Framework for Dynamic Evolving IoT Devices. *ACM Trans. Sens. Netw.* **2021**, *18*, 1–51. [CrossRef]
25. Gonzalez-Amarillo, C.; Cardenas-Garcia, C.; Mendoza-Moreno, M.; Ramirez-Gonzalez, G.; Corrales, J.C. Blockchain-iot sensor (Biots): A solution to iot-ecosystems security issues. *Sensors* **2021**, *21*, 4388. [CrossRef] [PubMed]
26. Dhar, S.; Bose, I. Securing IoT devices using zero trust and blockchain. *J. Organ. Comput. Electron. Commer.* **2021**, *31*, 18–34. [CrossRef]
27. Dashevskyi, S.; Zhauniarovich, Y.; Gadyatskaya, O.; Pilgun, A.; Ouhssain, H. Dissecting android cryptocurrency miners. In Proceedings of the Tenth ACM Conference on Data and Application Security and Privacy, New Orleans, LA, USA, 16–18 March 2020; pp. 191–202.
28. Wang, W.; Ferrell, B.; Xu, X.; Hamlen, K.W.; Hao, S. Seismic: Secure in-lined script monitors for interrupting cryptojacks. In Proceedings of the European Symposium on Research in Computer Security, Barcelona, Spain, 3–7 September 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 122–142.
29. Yulianto, A.D.; Sukarno, P.; Warrdana, A.A.; Al Makky, M. Mitigation of cryptojacking attacks using taint analysis. In Proceedings of the 2019 4th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE), Yogyakarta, Indonesia, 20–21 November 2019; pp. 234–238.
30. Lachtar, N.; Elkhail, A.A.; Bacha, A.; Malik, H. An application agnostic defense against the dark arts of cryptojacking. In Proceedings of the 2021 51st Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Taipei, Taiwan, 21–24 June 2021; pp. 314–325.
31. Romano, A.; Zheng, Y.; Wang, W. Minerray: Semantics-aware analysis for ever-evolving cryptojacking detection. In Proceedings of the 2020 35th IEEE/ACM International Conference on Automated Software Engineering (ASE), Melbourne, VIC, Australia, 21–25 September 2020; pp. 1129–1140.
32. Iqbal, S.; Malik, A.W.; Rahman, A.U.; Noor, R.M. Blockchain-based reputation management for task offloading in micro-level vehicular fog network. *IEEE Access* **2020**, *8*, 52968–52980. [CrossRef]
33. Fiori, C.; Ahn, K.; Rakha, H.A. Power-based electric vehicle energy consumption model: Model development and validation. *Appl. Energy* **2016**, *168*, 257–268. [CrossRef]
34. Burlig, F.; Bushnell, J.B.; Rapson, D.S.; Wolfram, C. *Low Energy: Estimating Electric Vehicle Electricity Use*; Working Paper 28451; National Bureau of Economic Research: Cambridge, MA, USA, 2021.
35. Saad, M.; Khormali, A.; Mohaisen, A. Dine and dash: Static, dynamic, and economic analysis of in-browser cryptojacking. In Proceedings of the APWG Symposium on Electronic Crime Research, Pittsburgh, PA, USA, 13–15 November 2019.
36. Bucolo, M.; Buscarino, A.; Fortuna, L.; Gagliano, S. Multidimensional Discrete Chaotic Maps. *Front. Phys.* **2022**, *10*, 862376. [CrossRef]
37. Caponetto, R.; Fortuna, L.; Graziani, S.; Xibilia, M. Genetic algorithms and applications in system engineering: A survey. *Trans. Inst. Meas. Control* **1993**, *15*, 143–156. [CrossRef]