4-1-2023

Developing resilient cyber-physical systems: A review of state-of-the-art malware detection approaches, gaps, and future directions

M. Imran Malik
Edith Cowan University

Ahmed Ibrahim
Edith Cowan University

Peter Hannay
Edith Cowan University

Leslie F. Sikos
Edith Cowan University

10.3390/computers12040079
Malik, M. I., Ibrahim, A., Hannay, P., & Sikos, L. F. (2023). Developing resilient cyber-physical systems: A review of state-of-the-art malware detection approaches, gaps, and future directions. Computers, 12(4), 79. https://doi.org/10.3390/computers12040079

computers

MDPI

Review

Developing Resilient Cyber-Physical Systems: A Review of State-of-the-Art Malware Detection Approaches, Gaps, and Future Directions

M. Imran Malik [1,*], Ahmed Ibrahim [1,2], Peter Hannay [1,3] and Leslie F. Sikos [1,2]

[1] School of Science, Edith Cowan University, Perth, WA 6207, Australia
[2] Security Research Institute, Edith Cowan University, Perth, WA 6207, Australia
[3] NCC Group, Perth, WA 6207, Australia
* Correspondence: muhammad.malik@ecu.edu.au

Abstract: Cyber-physical systems (CPSes) are rapidly evolving in critical infrastructure (CI) domains such as smart grid, healthcare, the military, and telecommunication. These systems are continually threatened by malicious software (malware) attacks by adversaries due to their improvised tactics and attack methods. A minor configuration change in a CPS through malware has devastating effects, which the world has seen in Stuxnet, BlackEnergy, Industroyer, and Triton. This paper is a comprehensive review of malware analysis practices currently being used and their limitations and efficacy in securing CPSes. Using well-known real-world incidents, we have covered the significant impacts when a CPS is compromised. In particular, we have prepared exhaustive hypothetical scenarios to discuss the implications of false positives on CPSes. To improve the security of critical systems, we believe that nature-inspired metaheuristic algorithms can effectively counter the overwhelming malware threats geared toward CPSes. However, our detailed review shows that these algorithms have not been adapted to their full potential to counter malicious software. Finally, the gaps identified through this research have led us to propose future research directions using nature-inspired algorithms that would help in bringing optimization by reducing false positives, thereby increasing the security of such systems.

Keywords: critical infrastructures; cyber-physical systems; malware; metaheuristics; nature-inspired algorithms; optimization

Citation: Malik, M.I.; Ibrahim, A.; Hannay, P.; Sikos, L.F. Developing Resilient Cyber-Physical Systems: A Review of State-of-the-Art Malware Detection Approaches, Gaps, and Future Directions. *Computers* **2023**, *12*, 79. https://doi.org/10.3390/computers12040079

Academic Editor: Paolo Bellavista

Received: 21 February 2023
Revised: 29 March 2023
Accepted: 11 April 2023
Published: 14 April 2023

1. Introduction

Cyber threats have the potential to adversely impact an organization's operations via unauthorized access, destruction, disclosure, modification of information, and (distributed) denial-of-service attacks. These attacks can be physical or virtual, targeting internal or external components, either directed or non-directed in nature, with direct or indirect consequences. The increased dependence on technology enablers has transformed the cyberthreat landscape to be modular and multifaceted in which the adversaries steadily and successfully develop tools that defy the security controls used in critical infrastructures. As such, the tactics, techniques, and procedures (commonly known as TTP) [1] need to be dynamic and demand continuous review. The adoption of a proactive approach against the ever-growing cyber threat landscape is considered the only possible way for critical systems to be protected or defended from cyber-attacks.

Critical infrastructure (CI) embraces all the sectors that provide services and utilities in our daily lives, such as financial systems, healthcare, energy, water, and security. The Australian Cyber Security Centre (ACSC) defines CI as those "*physical facilities, supply chains, information technologies and communication networks which if destroyed, degraded or rendered unavailable for an extended period would significantly impact the social or economic well-being of the nation, or affect Australia's ability to conduct national defence and ensure national security*" [2].

In similar terms, the United States through the U.S. Patriot Act of 2001 defines CI as "*systems and assets, whether physical or virtual, so vital to the United States that the incapacity or destruction of such systems and assets would have a debilitating impact on security, national economic security, national public health or safety, or any combination of those matters*" [3].

The exponential growth of physical devices exchanging data over data networks during the last decade has enabled such devices to be controlled by computers, either with minimal or no human interaction. This technological shift is referred to as a cyber-physical system (CPS), and it redefines contemporary cybersecurity (information technology (IT) security) (e.g., personal computers, servers, firewalls, smartphones) with the addition of security aspects of physical resources and machines that process digital data in the physical world (operational technology (OT) security). Therefore, most, if not all, CIs are now in the realm of CPS.

Cyber-risks to CPS involve threats from both malicious actors (e.g., malicious software, distributed denial-of-service attacks) and non-malicious actors (e.g., vulnerabilities inherent in devices and software). Any success in defying the services provided by a CPS directly affects the national economy and its defense, thus posing a significant risk to the CI of any nation. Specific to malware, Stuxnet (2010), BlackEnergy (2015), Industroyer (2016), Triton (2017), the constant activities of Dragonfly 2.0 on the energy sector in many countries, the cyber-attack on Kudankulam Nuclear Power Plant, and, more recently, Pipedream (2022) by Chernovite group, are prime examples of CI compromises.

Artificial intelligence and machine learning have long been used to minimize the considerable impact caused by soaring numbers of malware attacks. However, with securing critical infrastructure, malware detection needs to be optimized and time intelligent as these systems are highly dynamic, complex, and distributed. Nature-inspired metaheuristic algorithms hold the potential to fulfil this much-needed requirement as they can provide effective and fast enough defenses [4]. These algorithms are inspired by the various phenomenon in nature such as ants, dragonfly, bat, firefly, and cuckoo search and are considered useful to solve complex problems without compromising the efficiency of the system being protected [5–7].

The novelty of this survey article is that it focuses on the security aspects of CIs, which are challenged by the unprecedented rise of malicious software specifically crafted to attack CIs. To answer the aforesaid research objective, we have comprehensively examined work from various authors on malware analysis during the last ten years. While reviewing these papers, we specifically looked at whether the researchers have analyzed CI-specific malware and whether nature-inspired algorithms were used or not.

The flow of the rest of the paper is as follows: The *critical infrastructures and cyber-physical systems* section cover details about CIs and draws its relationship with CPSes, discusses a typical CPS framework, threats, and attacks on CPSes followed by malware outbreaks on CPSes since Stuxnet made the headlines. The next section is on *cyber-physical system malware*, where we discuss CPS malware, followed by defining malware classes and its variants. The section further examines malware analysis techniques by first grouping them into static and dynamic, with further sub-grouping using basic and advanced for each group. This is followed by approaches that have been used for malware detection, features used by the researchers, and a summary of significant work undertaken during the last ten years. The benefits of the use of a metaheuristic that includes nature-inspired algorithms have then discussed. We have also developed six hypothetical scenarios for five CI sectors to show the severity of the impacts caused by false positives if it is not handled delicately in a CI setting. The *cyber-physical system malware countermeasures* section details the current countermeasures to counter malware attacks on CPSes. Here, we also discuss MITRE ATT&CK framework for ICS and its significance to improving the security posture of an organization entrusted with securing CPSes. *Conclusions and future directions* is our last section, where we sum up the whole paper and draw together the gaps in the literature and present future directions that we aim to address.

2. Critical Infrastructures and Cyber-Physical Systems

The following sections will give an overview of critical infrastructures and cyber-physical systems.

2.1. Critical Infrastructures

The US Department of Homeland Security (US-DHS) [8] mentions 16 sectors to fall under the ambit of CI. These are *chemical; commercial facilities; communication; critical manufacturing; dams; defense industrial base; emergency services; energy; financial services; food and agriculture; government facilities; healthcare and public health; information technology; nuclear reactors, materials, and waste; transportation systems; and water and wastewater systems.* This indicates the diversity and complexity of CIs, consisting of distributed networks that are interdependent and interrelated. This dependence makes the security and resilience of CI reasonably difficult from the perspective of adverse events, such as deliberate attacks, accidents, and natural disasters [9]. Therefore, a prolonged incapacity or the destruction of a CI leads to a devastating effect on the national security of a country.

2.2. Cyber-Physical Systems

The transformation from paper-based systems to computer-based systems paved the way for organizations to develop and deploy solutions at a faster pace, thereby making their workflows transparent and efficient. In parallel, the broad expansion of the Internet also attracted these organizations to connect their systems to the Internet. Since computers first developed, the combination of mechanical power and information processing capabilities has resulted in an explosive growth of the number of physical devices exchanging data over communication networks. This enabled such devices to be controlled by computers, either with minimal human interaction or entirely autonomously, in the form of so-called cyber-physical systems, which redefines conventional cybersecurity with the addition of security aspects of physical resources and machines that process digital data in the physical world [10–15]. In simple terms, a CPS typically consists of a digital device that monitors and controls a physical environment.

However, some believe that CPS is a somewhat vague term. There are other definitions as well, such as "cyber-physical systems (CPSes) are engineered systems that are built from and depend upon, the seamless integration of sensing, computation, control, and networking in physical objects and infrastructures" [16]. According to the National Institute of Standards and Technology (NIST), the term is often used in the context of the Internet of Things (IoTs), Industrial Internet, smart cities, smart grid, smart anything (e.g., cars, buildings, homes, manufacturing, hospitals, appliances). Some typical applications of CPS include smart cities, smart grids, medical devices, robotics, airplanes, dams, industrial systems, and trade [12–15,17,18]. These examples form part of the CI sectors discussed in the previous section. CPS can be divided into two broad categories:

- Infrastructural CPS: systems that operate factories, refineries, etc. (for example, electric power/smart grid);
- Personal CPS: systems that consist of end-user devices such as smartphones, home systems, appliances, etc. (for example, smart appliances/smartwatch).

2.3. Cyber-Physical Systems Framework

The privacy and security aspects of a CPS can be represented through a framework that uses three orthogonal coordinates: *systems*, *components*, and *security* [16]. The *components* coordinate include cyber, cyber-physical, and physical domains, whereas the *systems* include critical services such as smart cars, medical devices, smart grids, and industrial control systems (ICS). The authors have specifically mentioned these four components as they drew a comparison among these facilities in their paper. However, the *systems* coordinate can have all the potential services that fall under the CI sectors discussed earlier. The third coordinate covers the *security* aspect of a CPS and includes controls, attacks,

vulnerabilities, and threats. The graphical illustration of this framework is presented in Figure 1.

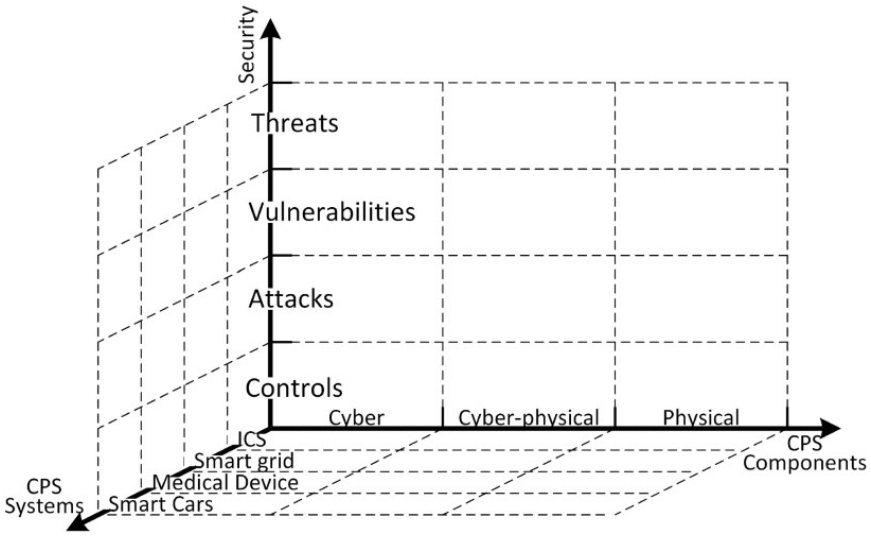

Figure 1. Abstract view of a CPS security framework [16].

While the CPS systems and security coordinates are self-sufficient, Humayed, Lin, Li and Luo [16] further clarified the CPS components by integrating each with the CPS abstract model. Figure 2 illustrates what constitutes a cyber, cyber-physical, and physical component in a CPS model.

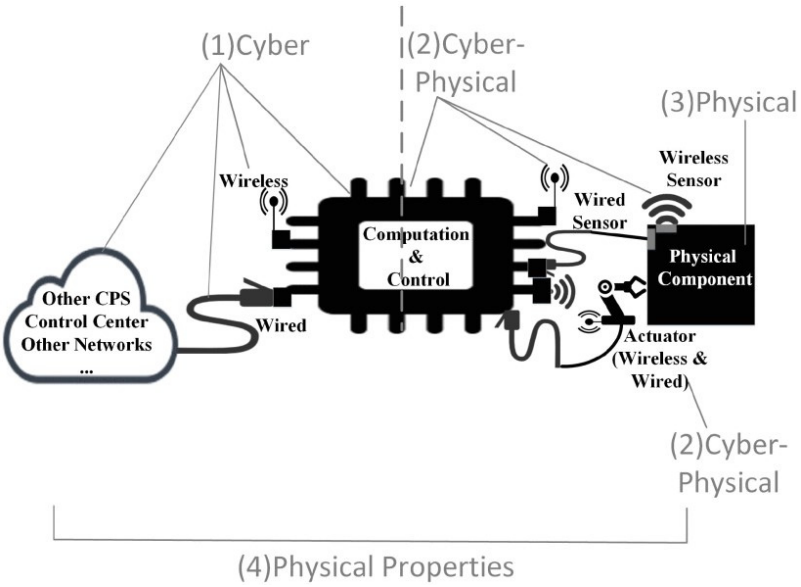

Figure 2. An abstract CPS model with integrated CPS components [16].

2.4. Threats to Cyber-Physical Systems

Security threats in a typical CPS ecosystem could occur at the interface between the devices, on the devices themselves, in the infrastructure that supports them, from the Internet, and from malicious users. Figure 3 demonstrates some of the security attack points in a CPS system.

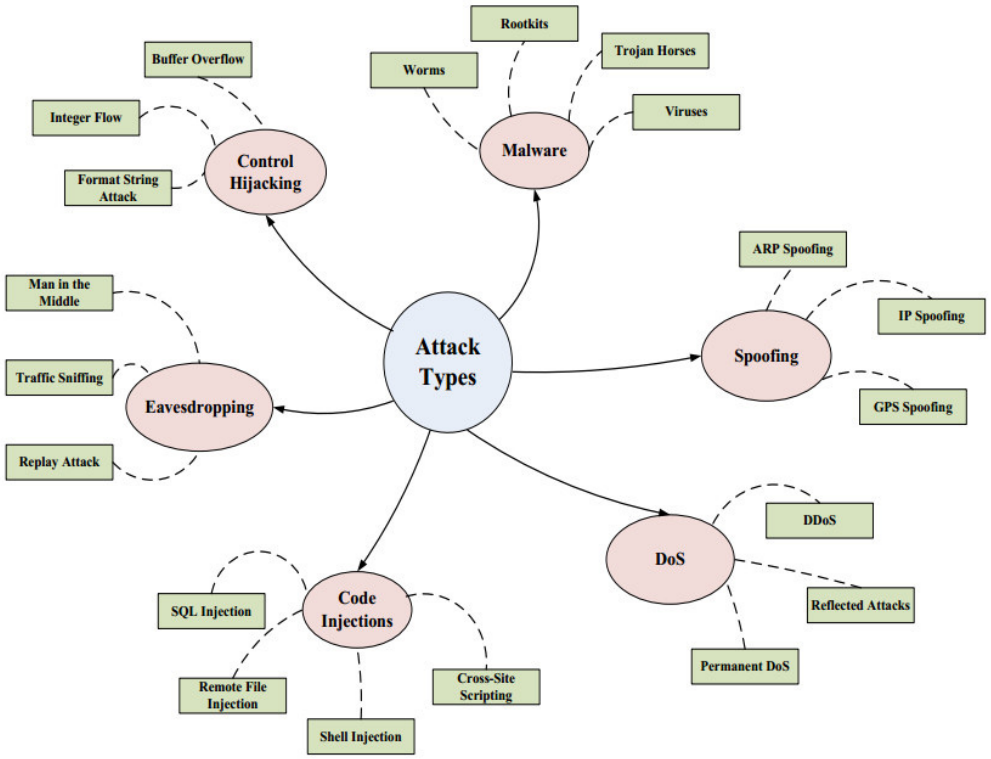

Figure 3. Attack types in a cyber-physical system [19].

Figure 3 illustrates multiple avenues where a malicious actor can leverage a flaw or flaws to compromise a CPS. These can be grouped as flaws related to software, network protocols, trust level between peer devices, end-users, and clients interacting with the systems [12,20,21]. Injection of malware at any of these vulnerabilities can nullify the trust relationship between physical and cyber-components, resulting in non-recoverable consequences [14]. Such a situation will undermine the societal benefits of utilizing a CI through compromising actions ranging from monetary gain to loss of human lives [22], unless such flaws are negated. A real-life example of this effect is the systematic control of a water treatment plant that remotely took control of the facility to poison households [23,24].

The above discussion signifies that exposure of CPSes to cyber threats has the potential to undermine national economies. Further, the continual lowering of sensor costs and the perpetual increase in the desire to connect Cis to the digital world further challenges the security of Cis [21]. This is supported by the Mordor Intelligence [25] report, where growth of USD 1386.06 billion in the IoT market by 2026 has been forecasted. Therefore, CPSes require a reliable and resilient security solution without significantly impacting their performance.

2.5. Attacks on Cyber-Physical Systems

While discussing the cumulative (physical and virtual) effect on the risks associated with CPSes, Scheuermann [26] argues that *if the non-cyber risk of fire or explosion at an oil refinery is X, then the risk that such a fire or explosion is caused by a cyber-attack becomes multiples of X*. Lloyd's, in their Emergency Risk Report 2015, presented hypothetical stress test scenarios with the title Business Blackout, underlining the consequences of a cyber-attack on the US power grid. The report presents that when a power plant is compromised through a piece of malware, it can cause sustained power outages, which in turn leads to substantial financial loss. In a particular example, such a power failure could cost the insurance industry up to USD 71.1 billion [27]. In another scenario, the report claims that an electricity outage in 15 US states affects 93 million people. This increases mortality

rates due to the failure of health and safety systems, causes a decline in trade owing to the shutdown of ports, and interrupts water supplies because of electric pump failures [27].

The cyber security strategy released by the Australian Government reflects how various critical services, both government and industry, have been targeted by malicious actors from July 2019 to June 2020, as shown in Figure 4. Considering the diversification in the attack vectors and the targeted approach adopted by the cyber criminals whereby the focus is to maintain the persistence in the compromised networks, the number of attacks is likely to be much higher.

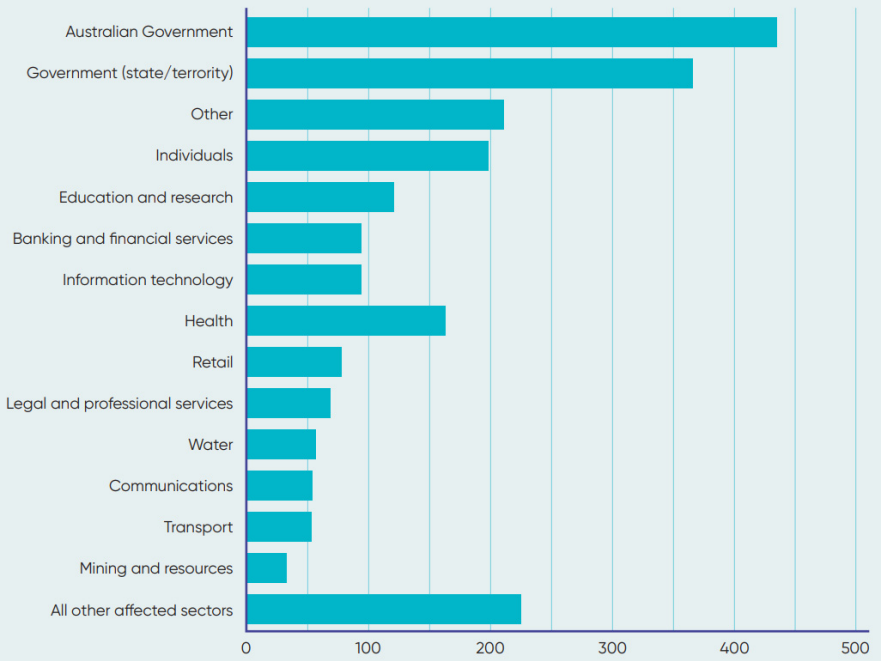

Figure 4. Cyber security incidents on Australia's critical services (January–June 2020) [28].

A report published by Malwarebytes Labs listed the top 10 industry sectors impacted by malware attacks from a variety of cyber criminals across the world using diverse attack vectors. Figure 5 shows that the services sector, which consists of a number of industries that fall under the ambit of CIs, and the education sector were the highest in terms of malware attacks. It is pertinent to mention that the graph in Figure 5 only depicts the attacks that were detected by the deployed scanners.

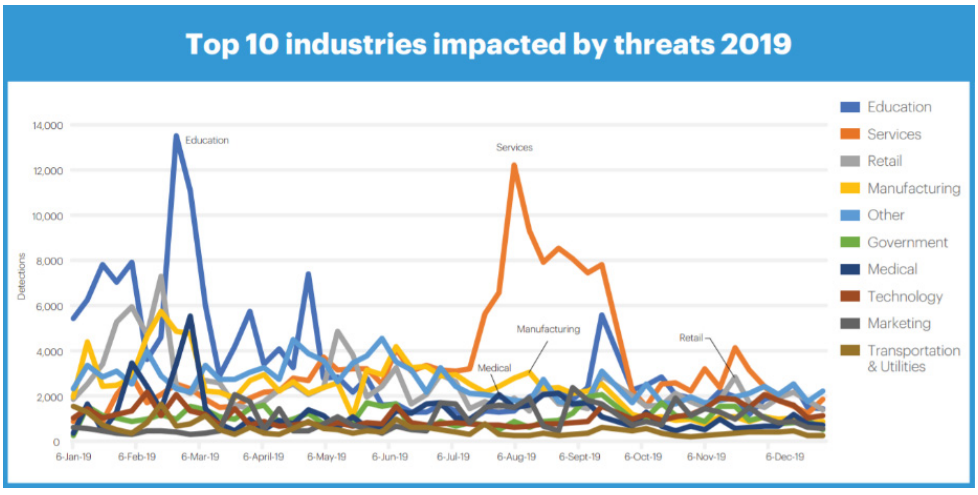

Figure 5. Malware attacks on critical infrastructure sectors in 2019 [29].

The year 2020 witnessed an unprecedented increase in ransomware attacks against the health sector worldwide. These attacks not only compromised patients' medical information but disrupted many critical care facilities, raising concerns among the community about a life-threatening situation. The ACSC identified two significant threats against the health sector in 2020 [30]. The first malicious activity was using the SDBBot malware, allowing remote access to the compromised computer. The compromise then enabled the successful deployment of the Clop ransomware. While the confirmation of successful execution of the two malware against the Australian health sector is unknown, one of the world's largest software companies, Software AG, fell victim to the Clop ransomware with a ransom demand of more than $20 million [31]. The 2021 global threat report released by CrowdStrike presents a graphical illustration of confirmed health sector compromises by a variety of ransomware families with infection counts, as shown in Figure 6.

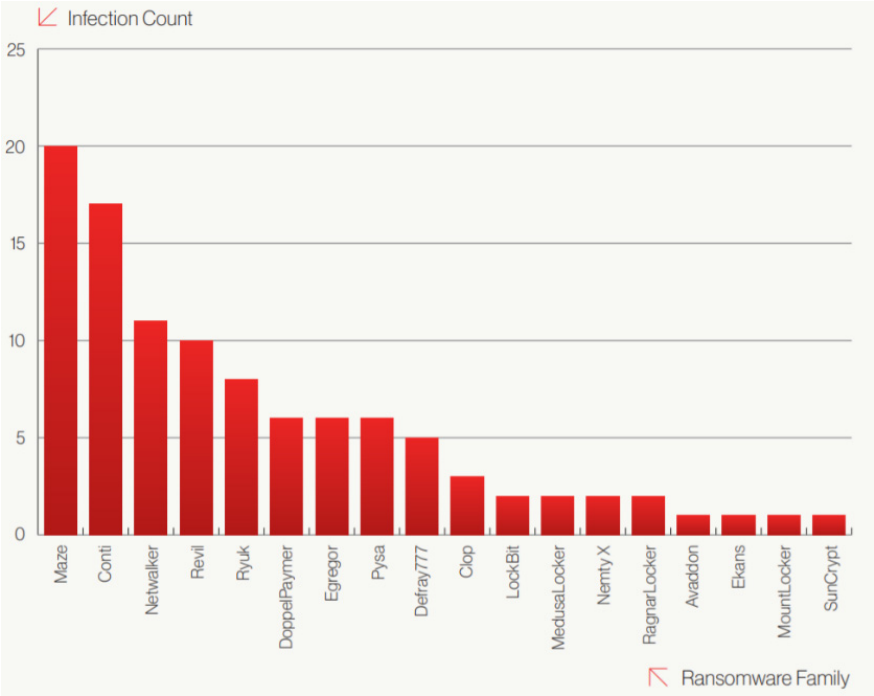

Figure 6. Ransomware compromises in health sector in 2020 [32].

2.6. Malware Outbreaks on Critical Infrastructures

The following section outlines some of the malware outbreaks that compromised CIs both physically and at the cyber front, thus supporting the claim made by Scheuermann [26].

- *Stuxnet*: First surfaced in 2010, Stuxnet (which is now termed a granddaddy of CI attacks) worm attacked Iran's Natanz nuclear facility with the motive to compromise her atomic program. The malware unusually infected the target system by exploiting a vulnerability in Siemens Programmable Logic Controller (PLC)—a piece of computer hardware commonly used in CIs. Security experts from Symantec claim that a thorough review of the Stuxnet source code revealed that the worm has 20 times more lines of code than average and is bug-free, which is very rare. Reportedly, the attack compromised one-thousand centrifuges deployed in the facility by enabling them to spin at a faster speed than usual, making them incapable of enriching uranium. The chief reason for this malware was persistence, enabling the attackers to remain informed about Iran's nuclear capabilities and slow down their uranium enrichment process. However, such a compromise has the potential to destroy any nuclear facility, when compromised, leading to catastrophe. While the worm predominantly compromised

the Natanz nuclear facility, various security firms claim that Stuxnet has subsequently attacked information systems all over the world.

- *German steel mill attack* (No formal malware name assigned): In 2014, a steel mill in Germany was compromised through a vulnerability in the support system for environmental control that was exploited by the attackers. To date, no official name has been given to this malware, and the relevant authorities have shared limited information about the attack to the public. The attack, however, massively damaged the productivity of the mill by not allowing the blast furnace to shut down causing substantial material damage. The compromise was realized through a systematic approach whereby the attackers gained control of the mill's industrial automation systems, disabling components that enabled the engaged workforce to view the status of the machines, making the blast furnace unable to stop in an organized way.
- *Energetic bear 2014 onwards*: Energetic Bear, often referred to as the name of the hacking group and the malware as well, was first spotted in 2014. The malware was found in 1000 energy firms (majority of them from the United States) in 84 countries. Though no actual damage had been reported then, security companies Symantec, F-Secure, and CrowdStrike claim that the developer of industrial control systems from three companies (FireEye, now Trellix, an intelligence-led security firm) claims four companies) was targeted, and their software was injected with this malware. When the control systems, more specifically PLCs, were updated/patched, the infections allowed hackers to monitor the activities of the infected companies. The traces of a similar attack were also later found in companies in the financial sector. The other name that referred to the malware is Havex, a Trojan used to create backdoor PLCs.
- *Ukraine power grid 2015/BlackEnergy*: BlackEnergy (BE) was first acknowledged in 2007 and has three variants to date, referred to as BE1, BE2, and BE3. Each time the malware gets more sophisticated and lethal compared to its predecessor regarding its features and capabilities. The central theme behind all these variants was to launch DDoS attacks. The US Department of Homeland Security exposed BE2 as compromising many CIs, such as nuclear sites, power grids, and water purification systems. However, the major disorder was reported in December 2015 when BE3 malware was used against Ukraine's power grids. The attack has been termed multisite and multistage, where supervisory control and data acquisition (SCADA) systems of three power distribution companies were compromised in a harmonized way. Through this, various substations were compromised resulting in power blackouts for a significant chunk of the country's population. Different reports suggest that the blackout remained for between three to six hours before being restored. Not only this, but BE3 was sophisticated enough that it used KillDisk malware that removed the attack traces and assisted the attackers to prolong power failure.
- *Ukraine power grid 2016/Industroyer (crashoverride)*: Termed by many independent security organizations as a continuation of the 2015 blackout but more intricate, systematized, and entirely independent of BE, this attack hit one-fifth of Kiev's (Ukrainian capital) population. Though the attack was not as prolonged as through BE malware, the consumers remained without power for more than an hour. While analyzing the samples, ESET named the malware as Industroyer and argued that this highly customizable malware has the potential to compromise other CIs as well. Effective against the power control products by ABB and Siemens SIPROTECT devices, the malware had the ability to control power substations and circuit breakers, causing catastrophic damage to the affected plant and to the consumers that also includes, but are not limited to, compromising the functioning of vital health services.
- *Triton*: Detected in 2017 and also named Trisis/Hatman, the malware attacked Safety Instrument Systems (SIS) in Middle Eastern countries. SIS controllers are aimed at monitoring the performance of critical systems and take corrective actions shifting the system into a safe state when it detects an unsafe condition. The attack targeted Triconex (installed in ~15,000 sites all over the world) by Schneider Electric. The com-

promise enabled the hackers to install a Trojan, allowing them to remotely manage the PLCs of the affected system and maintaining persistence, enhancing the ability of the system causing significant material and human damage. More specifically, Triton has affected the famous Saudi Arabian company Saudi Aramco—a petroleum and natural gas company. Since being examined, the details of the damage are still not available. The code manipulated the emergency shutdown protocols that caused the system to halt inadvertently. FireEye claims that the attacking entity intended to maintain persistence to allow them to cause damage more severe than shutting down the system. However, bringing the system to a halt gave the asset owners an opportunity to remediate the attack.

- *Pipedream*: Recently reported by the Cybersecurity & Infrastructure Security Agency [33] through an advisory, Pipedream is a purpose-built modular malware that actively scans for vulnerabilities in the CIs that have devices/components from Schneider Electric, OMRON Sysmac, and Open Platform Communications Unified Architecture (OPC UA) to establish initial access. Once the initial access to a CI is attained, the cyber criminals can open backdoors, maintain persistence, or change the device configurations, which could have a devastating effect. Although the real-life compromise from this malware has not yet been reported, a whitepaper published by Dragos [34] highlights the sophistication of Pipedream due to its capability of reconnaissance, brute-forcing passwords, and crashing the target device. The paper also highlights the extensive capability of the CHERNOVITE threat group behind Pipedream as the analysis of the malware shows the refined skills of this group in software development methods, ICS protocols, and securing funding.

3. Cyber-Physical System Malware

Malware (malicious software) is defined by NIST as "a program that is inserted into a system, usually covertly, with the intent of compromising the confidentiality, integrity, or availability of the victim's data, applications, or operating system or otherwise annoying or disrupting the victim" [35]. Stallings and Brown [36] and Souppaya and Scarfone [35] describe malware as one of the most significant and rapidly evolving threats to every information system. It can cause widespread destruction and disruption and requires extensive efforts for an organization to restore its normal operations. Symantec's security threat report of 2018 claims that between 2016 and 2017 there has been a 600% increase in malware attacks against IoTs [37]. This unprecedented growth is also supported by McAfee Labs, who recorded 57.6 million new malware samples at the end of 2017 [38].

Malware attacks on CPSes were first realized in 2010, when Iran's uranium enrichment plant at Natanz was compromised by Stuxnet. Since then, the proliferation of IoTs and their utilization in CIs have further intensified such threats. Others also support this notion and claim that attackers have the ability to successfully use malware, such as scareware and ransomware, to control drivers of (semi-)autonomous smart cars until they meet attacker demands, such as a payoff [39]. Bettany and Halsey [40] argue that future wars will be fought through organized cyberwarfare and will be targeted towards disrupting CPSes using malware. A few examples of such attacks include a German steel mill compromise in 2014, Ukraine's power grid compromise in 2015–2016 (using BlackEnergy and Industroyer), Saudi Arabia's oil company compromise in 2017 (through Triton), the suspected compromise of the United States nuclear power plant belonging to the Wolf Creek Nuclear Operating Corporation in 2017, the attack on India's Kudankulam nuclear power plant (KKNPP) in 2019, an attempted cyber-attack on Israel's water systems in April 2020, the compromise of one of the UK's electricity grids compromising its internal IT systems in May 2020, and the recent ransomware attack on Oil India Limited's (OIL) field headquarters. These examples support the findings of a report by the Kosciuszko Institute that claimed that from 2018 onwards, there will be an unprecedented rise in attacks on CI, primarily from malware [41].

3.1. Malware Classes

Malware is considered the most common threat that can cause widespread damage and disruption, and necessitates extensive recovery efforts while compromising the confidentiality, integrity, and availability of the attacked devices. Malware, therefore, is classified to gain a better understanding of the methods and purpose of compromised systems. Depending on its purpose, various classes of malware are summarized in this section. It is noteworthy that these classes are not mutually exclusive and may contain the attributes of more than one type at a given time [42].

- *Virus*: A program, when inserted, attempts to replicate itself by adding copies into system or data files [35,36]. These programs are mainly activated by user interaction;
- *Worm*: A worm, considered as an advanced form of the virus, has the features of being self-replicating and self-reliant as they spread without user interaction ranging from a single system to an entire network [35,36];
- *Trojan Horse*: A self-reliant, non-replicating program that appears to be legitimate and innocuous but has a hidden malicious objective of exploiting the system [35,36]. Once active, the Trojan opens a backdoor for the attacker to gain further control of the affected system or install a virus or worm to intensify the attack further;
- *Adware*: Largely, adware does not affect the system files nor the user data as they are aimed at occupying the user screen to display different advertisements. These programs are also integrated into other software that the user needs for their normal working. The program generates pop-ups and entices a user or the browser redirects to a commercial website [36]. In addition, this software has the potential to slow down the system by using considerable system resources;
- *Spyware*: A program installed on a system without the user's consent and transmits critical information to the attacker such as keystrokes, screen data, network traffic, and scrapes the user's files for sensitive information [36]. Spyware was considered as a companion to adware used to track a user's browsing interests and then selling it to the advertisers;
- *Rootkit*: A collection of files installed on the compromised system to escalate the permissions to the administrator level in a stealthy way that is incredibly difficult to detect [35,36]. The stealthiness is achieved due to a change in the system's configuration files that hides the rootkit from detection;
- *Backdoor*: Backdoor, also known as a trapdoor, is a program that executes the commands through TCP or UDP ports [35,36]. It can be considered as a secret entry for the attackers to maintain persistence into the compromised system. Consequently, the attacker attains the ability to acquire confidential information by executing arbitrary instructions. Backdoors also allow the attacker to install other malware on the compromised system;
- *Keystroke Logger/Keylogger*: As the name infers, keystroke loggers monitor and capture the keys of the keyboard being used [35,36]. Different variants of these loggers can either actively transfer the observed data to the attacker or through other means such as email or file transfer;
- *Scareware*: Ye, Li, Adjeroh and Iyengar [39] argue that scareware tricks the user to either buy or download software that is dangerous and designed for financial and privacy-related threats;
- *Ransomware*: Gaining popularity during the last five years, the Australian Cyber Security Centre (ACSC) [43], Connolly, et al. [44], and Hampton and Baig [45] define ransomware as a type of malware that locks the attacked system or network until the desired ransom is paid. After the initial foothold, the program can spread to other shared storage devices to encrypt data and make the systems inaccessible. Ransomware can even delete the data if the payment is not made within the given timeframe;
- *Bot*: Malicious programs that remotely control an already compromised system are referred to as Bots. This type of malware is a starting point that installs other types of

malware and has the ability to transform an already conceded system into a network of bots commonly known as Botnets [39].

3.2. Malware Variants

There are two main types of malware:

- *Polymorphic malware*: a program that changes its appearance each time it replicates but keeps its original code intact [36,39,46]. The change in appearance enables the program to hide from the malware scanners because, with every change in the appearance, the signature of the malware gets changed;
- *Metamorphic malware*: a program that mutates with every iteration.

The key difference between metamorphic and polymorphic is that the earlier rewrites itself entirely and does not maintain the original code [36,39,46]. This behavior makes it impossible for detectors to identify the threat and compromises the affected system.

4. Advanced Persistent Threat (APT)

NIST defines advanced persistent threat (APT) as "an adversary that possesses sophisticated levels of expertise and significant resources which allow it to create opportunities to achieve its objectives by using multiple attack vectors, including cyber, physical, and deception" [47]. Often incorrectly attributed as another malware class, an APT is a well-resourced group or organization, usually state-sponsored, that undertakes a cyber espionage operation using refined, coordinated, and purposeful techniques to compromise a high value selected target. According to the ACSC, APT groups' target are usually nation-states or private organizations [48], with their aim varying from data theft, disruption of operations, or destruction of infrastructure [36,49–51]. Malware outbreaks on the critical infrastructures discussed earlier in this paper are examples of attacks launched by different APT groups. Figure 7 highlights the attributes of a typical APT group.

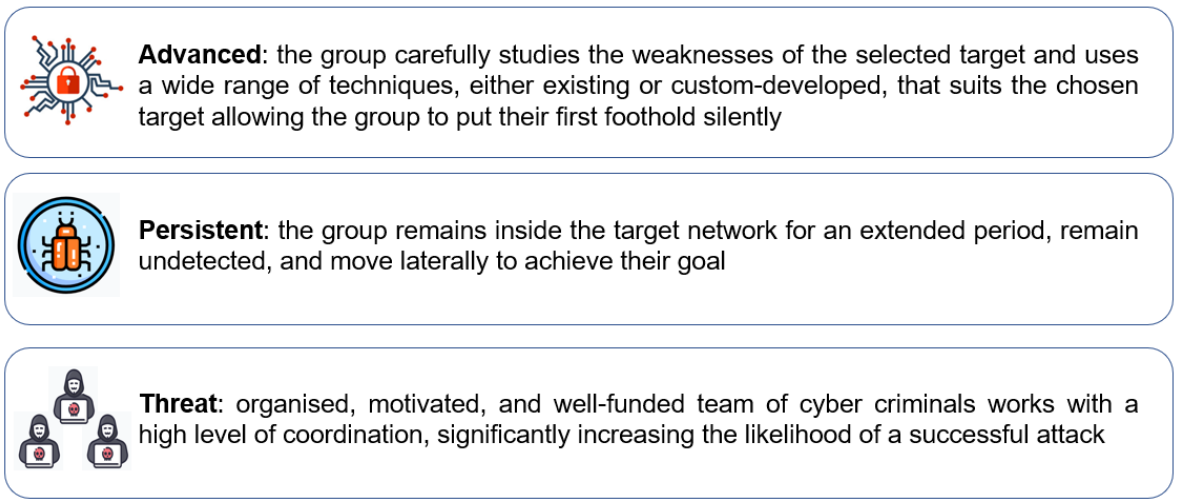

Figure 7. Typical characteristics of an APT group.

The MITRE ATT&CK (adversarial tactics, techniques, and common knowledge) maintains a list of 110 APT groups through open-source reporting. Table 1 presents a consolidated list of nine APT groups, highlighting their suspected attribution and the weapon of choice [52–54]. It is worth noting that each of the APT groups mentioned in the table uses malware against their targets.

Table 1. List of 9 APT groups with their capabilities.

APT Group	Attribution (Suspected)	Weapon of Choice
Lazarus (APT37)	North Korea	Ransomware
Equation	Unites States	Zero-day exploits, spyware
Fancy Bear (APT28)	Russia	Spear-phishing/malware
Dynamite Panda (APT18)	China	Trojan ransomware
Elfin (APT33)	Iran	Malware
OceanLotus (APT32)	Vietnam	Social engineering/malicious payloads
Zhenbao (APT21)	China	Spear-phishing/malicious attachments
APT5	Unknown	Malware with keylogging capabilities
CHERNOVITE	Unknown	Pipedream—a modular malware

5. Malware Analyzing Techniques

Malware analysis is defined as an "art of dissecting malware to understand how it works, how to identify it, and how to defeat or eliminate it" [55]. In general, the process for analyzing malware has two branches: static and dynamic. In static analysis, the malware is examined without execution, whereas the dynamic analysis scrutinizes malware through execution and observing its behavior in a virtual or emulated environment [51,56–62]. Static and dynamic techniques can be further grouped as shown in Figure 8.

The following table discusses the functionality for each of the subgroups presented in Figure 8.

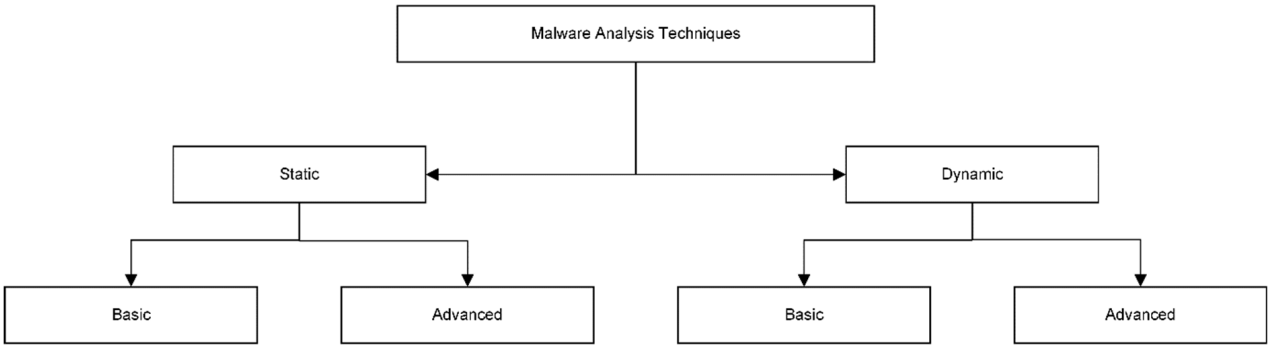

Figure 8. Malware analysis techniques.

A closer look at Table 2 indicates that while static malware analyses are more innocuous than dynamic analysis, they do not yield results that can be of much value in today's threat-evolving atmosphere. In contrast, dynamic analysis generates more information of interest, because it executes the malware in a controlled environment close to the real-world setting. This notion is supported by Ye, Li, Adjeroh and Iyengar [39], Ranveer and Hiray [51,63], Gaurav, Gupta and Panigrahi [62] and Yan, Ren, Wang, Sun, Zhang and Yu [61] as they argue that static analysis is ineffective as it has the predisposition of leaving away the key behaviors of malware that can be obtained through dynamic analysis and therefore are considered inadequate. While dynamic analysis is far better than static analysis, particularly in terms of accuracy, Gandotra, Bansal and Sofat [42] and Mathur and Hiranwal [46] argue that both static and dynamic analysis have limitations to achieve a high accuracy and a low false-positive ratio. They indicated the work of the researchers who have used a hybrid technique integrating the features obtained through both static and dynamic methods. A general flow of activity for hybrid analysis is presented in Figure 9. Results achieved from hybrid analysis bring better malware detection rates and reduce time requirements [39,42,58,61,64].

Table 2. Malware analysis techniques.

	Sub-Group	Functionality
Static	Basic	Uses tools to determine the nature of the file, and the range of operations an executable may perform. The tools used can also give technical information for the file being examined that can be used as signatures. While the process is time-efficient, it is not as effective as it may lead to false positives or false negatives. Elementary methods such as hash values and antivirus tools are used here.
	Advanced	Reverse-engineering of the malicious file is undertaken to understand its flow, and behavior of the program is observed. Tools such as IDA Pro are used for malware disassembly under this sub-category.
Dynamic	Basic	Malware is executed, and behavior observed on the system. The behavior enables the production of useful signatures that may assist in detection or eliminating the malicious files. Better than Basic Static, but vulnerable to bypass key malware attributes.
	Advanced	Allows the examiner to dig deep into the malicious file by using a debugger that enables the extraction of critical details that are otherwise not possible with other categories. The process is quite time-intensive, but the information obtained is far more effective in thwarting the malware as compared to the others. Debuggers are used in this subcategory and enable the examiner to acquire root level information about the executable.

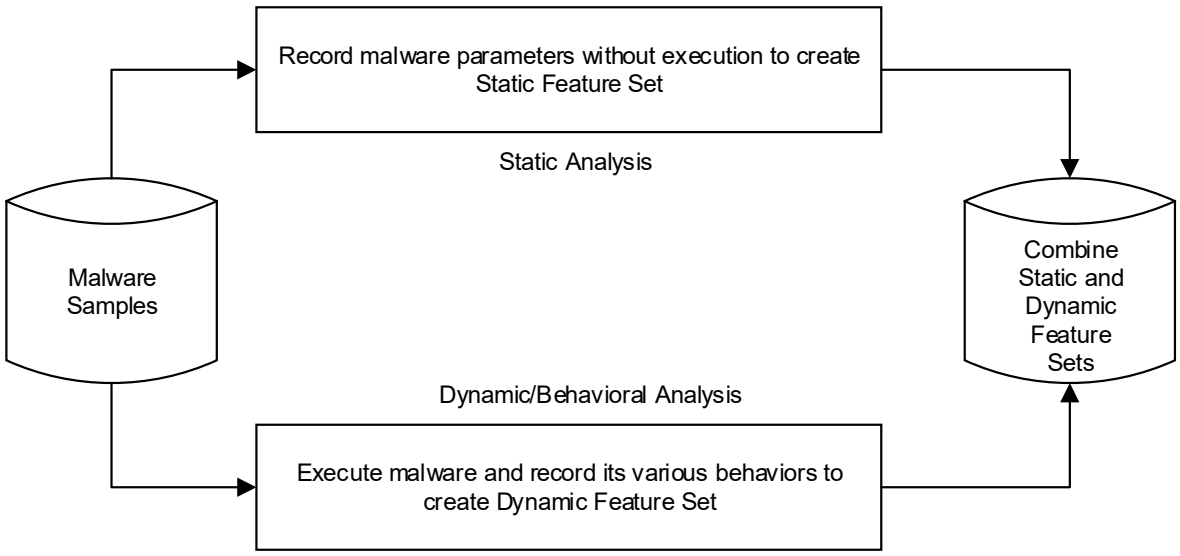

Figure 9. Malware analysis using hybrid approach.

6. Approaches to Malware Detection

Malware detection, as defined by [65], is a mechanism that analyzes a set of executables to detect its malicious or benign nature. The formal representation of the above definition can be represented as a function $D(P)$ [66], where D is the computational function, and P is the collection of programs. The function examines a program by either analysis or identification, and classifies them as malicious, benign, or undecidable. D continues to scrutinize the undecidable nature of P until the executable is classified as malicious or benign [66].

Signature-based and behavior-based are the two foremost classes for malware detection [57,67,68]. As the name implies, the signature-based detection solely relies on a predefined database of a short sequence of bytes [39]. A file is declared malicious when its characteristics match any of the signatures available in the database [69,70]. The process of the signature-based malware method is shown in Figure 10. While still being widely used, Bazrafshan, Hashemi, Fard and Hamzeh [69], Ye, Li, Adjeroh and Iyengar [39], and Damodaran, Troia, Visaggio, Austin and Stamp [57] argue that the flip side of this approach is that malware developers have advanced to executables that change its signatures each time they are launched. Such malware are known as polymorphic, as discussed earlier, and are unable to be detected by the signature-based method [59,70,71]. This swings

the malware detection focus to features related to the behavior of malicious executables. In addition to this, using the signature-based detection has the potential for malware to evade the security control and remain undetected for a long time.

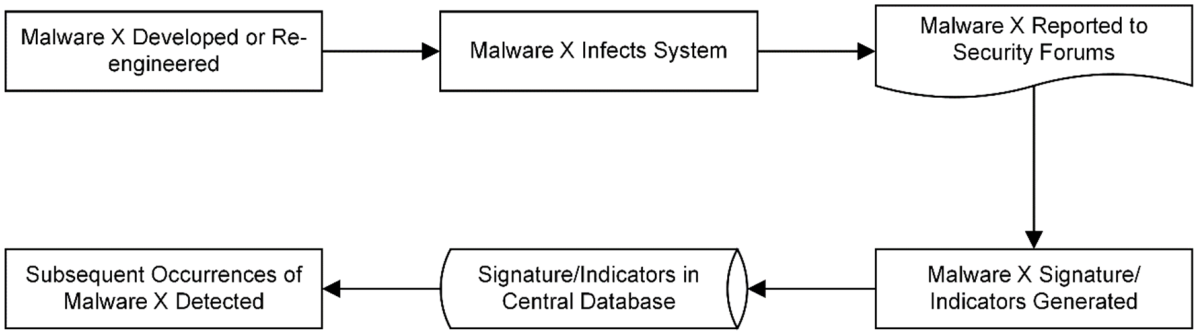

Figure 10. Signature-based malware detection process.

The literature indicates that some of the researchers take behavior-based and heuristics-based detection approaches under one umbrella, whereas others declare both approaches as independent. In his bachelor's thesis, Chumachenko [71] considers behavior-based and heuristics-based detection approaches as one, whereas Bazrafshan, Hashemi, Fard and Hamzeh [69] and Souri and Hosseini [70] treat them as separate. In another survey paper, Ye, Li, Adjeroh and Iyengar [39] consider heuristic-based malware detection method in similar terms as the behavior-based approach.

The inability of the signature-based approach in sensing evolving malware, including those that exploit zero-day vulnerabilities [56,66,69,71,72], resulted in behavior-based detection being introduced. In this approach, the artifacts of an executable are analyzed to classify them as malware or benign [63,65,69,71,72]. Bazrafshan, Hashemi, Fard and Hamzeh [69], Mohaisen, et al. [73], and Souri and Hosseini [70] further argue that this detection approach overcomes the weaknesses of signature-based detection techniques, such as detecting polymorphic malware and their variants. The executable is run in a sandboxed setting, which is a controlled environment, and various features are logged. While this approach has higher precision in distinguishing malware, Bazrafshan, Hashemi, Fard and Hamzeh [69], Souri and Hosseini [70], Chumachenko [71], and Mohaisen, Alrawi and Mohaisen [73] argue that time complexity, unpromising false positive ratio, and storage intricacy for behavioral artifacts are its weaknesses.

7. Malware Features

When the signature or behavior of an executable is examined, it represents a large set of attributes, called features, in the form of a matrix (referred to as a feature vector). Not all features are useful, because some are redundant or irrelevant. Selecting the most appropriate and pertinent set of values makes the extracted features more manageable, organized, and reduces the computational complexity, which in turn helps achieve a low false positive rate [56,61,63]. According to the literature, there is no standard set of features that can be employed universally for malware detection. The works studied indicate that the researchers used different features according to their needs and the issue being addressed. For this reason, it is difficult to make a valid comparison [74]. Further, some researchers have categorized malware features using detection approaches, i.e., signature or behavior-based, while others have represented using analysis techniques (static or dynamic). However, the use of different classes complicates malware examination, even though all researchers are essentially performing either static or dynamic analysis of executables. This belief has been supported by Ye, Li, Adjeroh and Iyengar [39] and Mathur and Hiranwal [46], who argue that static/dynamic/hybrid analysis are the applications of signature-based and behavior-based techniques. Further, Ye, Li, Adjeroh and Iyengar [39] argue that the hybrid technique for extracting a feature from an executable contributes to the efficient and accurate detection of malicious files. Table 3 presents a brief summary of

features using the previous categorization, including hybrid analysis. Figure 11 illustrates different features clustered into five groups [75].

Table 3. Malware features used by different researchers.

Category	Sub-Category		Features Used	References
Malware Detection	Signature-Based		Binary, Assembly	[70] [1]
	Behaviour-Based		API Calls, Assembly	
Malware Analysis	Static		Opcode n-gram, Byte Code n-gram, String, Portable Executables	[63]
	Dynamic		Function-based feature, API Calls, System Calls, Information Flow Tracking	
Malware Analysis	Static		Windows API Calls, byte n-grams, Strings, Opcodes, Control Flow Graphs (CFGs), File Property, File Resource Information, Export Table	[39] [2]
	Dynamic		No specific feature mentioned. Instead, the author discussed different execution environments (Debugger, Simulator, Emulator, and Virtual Machine)	
Malware Detection	Behaviour-Based		File System, Registry, Network	[73]
Malware Detection	Behaviour-Based		Files, Registry Keys, Mutexes, Processes, IP Addresses, and DNS Queries, API Calls	[71]
Malware Analysis	Static		n-grams	[68]
Malware Analysis	Static		API Calls, Opcodes	[57]
	Dynamic		API Calls	
Malware Analysis	Hybrid	Static	Function Length Frequency, Printable String Information	[59]
		Dynamic	API Calls	
Malware Analysis	Hybrid	Static	Opcodes	[76]
		Dynamic	System Calls, Operations, Raised Exceptions	
Malware Analysis	Hybrid	Static	2-g, opcodes [selective], Control Flow Graphs (CFGs)	[77]
		Dynamic	Instruction Traces, System Calls, Miscellaneous File Information Features	
Malware Analysis	Hybrid	Static	Control Flow Graphs (CFGs), Data- Flow Graphs (CFGs), System Calls	[58]
		Dynamic	API Calls	
Malware Analysis	Hybrid	Static	Printable String Information (PSI)	[64]
		Dynamic	API Calls Sequence (3-API-call-grams and 4-API-call-grams)	
Malware Detection	Hybrid	Heuristic	API Calls, Control Flow Graphs, n-grams, OpCodes, Hybrid features	[69] [2]

[1]: Survey paper summarizing work from different researcher; [2]: Used heuristic approach (discussed later) that is based on data mining and machine learning. A combination of signature and behaviour-based approach has been discussed by the authors when presenting the features.

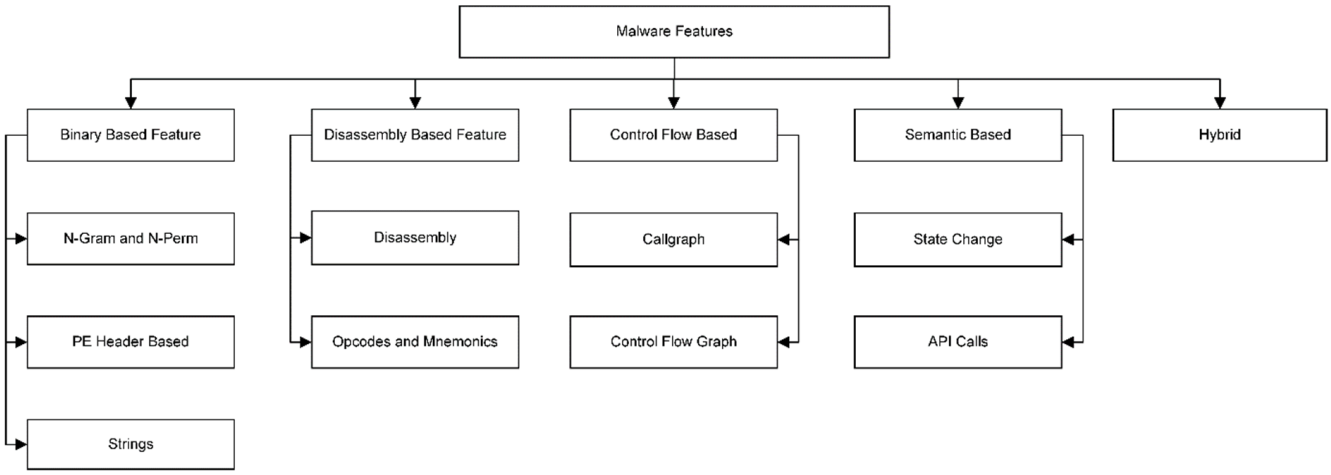

Figure 11. Categories of malware features clustered in five groups.

8. Artificial Intelligence and Machine Learning

Artificial intelligence (AI), the term coined by John McCarthy in 1955, holds the promise of enhancing computing power. The main idea behind AI is to enable computers to interpret data at a level adequate for making human-like decisions. Russell, et al. [78] argue that AI has a high potential for maximizing the societal benefits it has to offer by bringing robust and beneficial systems for the community. Through AI, computers are entrusted with the control of complex systems. Google Duplex is the recent example of an AI approach being extensively employed through which a robot talks to a human, initially, without providing a clue about itself. However, Google later confirmed that their robot would first introduce itself before carrying on the intended communication [79,80]. Machine learning (ML) is a field of AI aimed at building trust and bringing resilience in digital systems that are increasing exponentially around the world and have grown more dependence on computing machines than ever before [81,82]. The huge influx of new malware threats appearing daily attracted researchers to use ML capabilities for automated analysis. Compared to manual processing, this yields useful, fast, and reliable results.

9. Heuristic and Metaheuristic Techniques

The following section defines optimization, followed by the concepts behind heuristic and metaheuristic techniques. This will be followed by the significance and details of nature-inspired metaheuristic algorithms.

Optimization: The concept of optimization can be directly related to CI's security, based on its definition and criticality, as all such systems are aimed to provide efficient quality services with maximum profit. A compromised CI is detrimental to all these attributes. However, Luke [83], Talbi [84], Yang [85], and Yang [86] proclaim that real-world optimization problems are hard to solve as they are complex and intricate. Alternatively, approximate solutions or algorithms can be used. For malware detection, the disadvantages of the signature and behavior-based approaches make it necessary to use approximate solutions.

Approximate solutions, also known as stochastic algorithms [85], are further subdivided into heuristics and metaheuristics [83,84,86,87]. Ironically, the two terms have been used interchangeably, but there is a difference, though very subtle. Talbi [84] presented problem-specific heuristics and metaheuristic as two offshoots of heuristic algorithms. In their survey paper on malware analysis, Bazrafshan, Hashemi, Fard and Hamzeh [69] argue that approximate solutions utilize data mining and machine learning methods to study the behavior of the file being examined. The abstract view showing the general process for malware detection using an ML approach by either classification or clustering is represented in Figure 12.

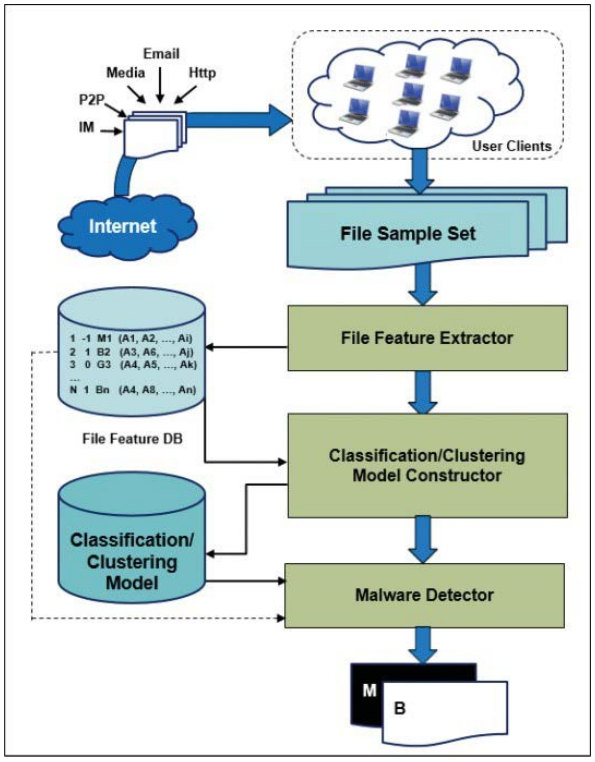

Figure 12. Malware detection using ML [39].

Heuristic Techniques: Heuristic, a Greek word meaning find or discover, is a technique that focuses on discovering a solution based on a trial-and-error method, i.e., finding a solution by thinking outside the box or by accident [84,86]. Examples of the heuristic-based algorithm are artificial neural networks (ANN) and support vector machines (SVM) that aim to minimize learning and prediction errors via iterative trial and error [86].

Metaheuristic Techniques: Metaheuristic, a major subfield of stochastic optimization [83,88], are self-learning algorithms aimed at efficiently solving complex optimization problems [89–92]. In contrast to heuristic algorithms that are problem dependent and are experience-based [89,90], metaheuristic attains optimization by guiding the design of heuristic algorithms with less computational cost [83,84,87]. Metaheuristic has three key attributes—the fast solution to the problem, solving large problems, and obtaining robust algorithms [84]—in addition to being very flexible and modest to design and implement. Therefore, these algorithms are also known as black box whereby only input and output information is required without needing to calculate the derivative of the search space. The first instance of metaheuristic algorithms dates back to between 1950 and 1955 when the pattern search and evolution process methods were introduced [89]. The two dominant classes for metaheuristic algorithms are evolutionary and swarm intelligence [88]. However, the aim behind both these classes is to find the most conducive solutions during optimization. These algorithms can also be classified into two categories: single-solution based and population-based metaheuristic algorithms. Single-solution metaheuristic (such as simulated annealing) are designed to give a single solution at a time, whereas population-based (such as genetic and firefly algorithms) are interactive and strive for an optimum solution [6,84,86,90].

Nature-Inspired Metaheuristic: Nature inspires the majority of the metaheuristic procedures, hence why they are called nature-inspired metaheuristic algorithms [6,84–86]. Researchers design these algorithms after drawing inspiration from nature, which comprises a habitat, an environment, and a collection of different species [93,94]. Therefore, the study of species living and movement style followed by its adaptation to algorithms is known as *nature-inspired algorithms* [94]. These nature-inspired algorithms have seen their influx in securing computing systems over the last decade because of their efficiency as they imitate the most exceptional features in nature as it has evolved over millions of

years [6,95,96]. However, this is still an open field of research as the cat-and-mouse game among the diverse attack vectors and security professionals are getting complicated day by day. Hence, optimization will remain a continuous struggle for security researchers. Figure 13 presents a typical high-level sequence of attack in nature using an aggressor-victim scenario. It draws its one-to-one relationship within the cyber domain using an attacker/malware scenario. This further supports the idea of using nature-inspired algorithms in securing CPSes from malware attacks. Table 4 presents various categories of nature-inspired algorithms with the example of specific algorithms within each.

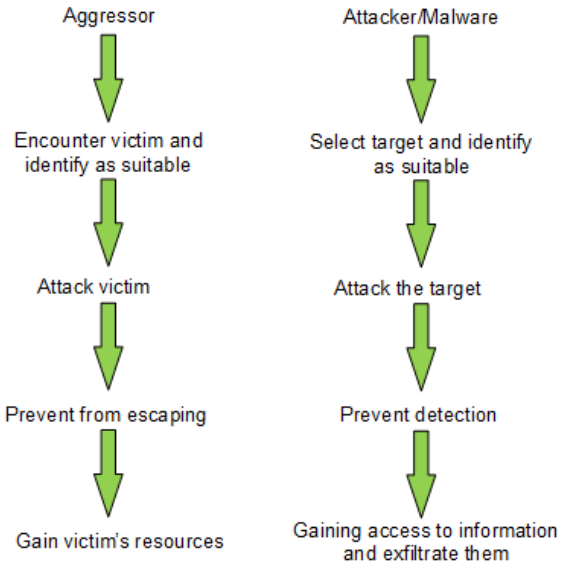

Figure 13. High-level sequence of attack: nature vs. digital world. Redrawn from [4].

Table 4. Classification of nature-inspired metaheuristic algorithms. Excerpt from [97].

	Examples
Evolutionary Algorithms	Genetic Algorithm (GA) Differential Evolution (DE) Genetic Programming (GP) Evolutionary Strategy (ES) Granular Agent Evolutionary Algorithm
Physical Algorithms	Simulated Annealing (SA) Memetic Algorithm (MA) Harmony Search (HS) Shuffled Frog-Leaping Algorithm (SFL)
Swarm Intelligence Algorithms	Ant Colony Optimization (ACO) Particle Swarm Optimization (PSO) Artificial Bee Colony (ABC) Fish Swarm Algorithm (FSA)
Bio-Inspired Algorithms	Artificial Immune System (AIS) Bacterial Foraging Optimization (BFO) Dendritic Cell Algorithm Krill Herd Algorithm
Other Nature-Inspired Algorithms	Cat Swarm Optimization (CSO) Cuckoo Search Algorithm [1] Firefly Algorithm [1] Invasive Weed Optimization Algorithm (IWO) Gravitational Search Algorithm River Formation Dynamics Bat Algorithm [1]

[1]: Also categorised as swarm intelligence algorithms.

The literature on metaheuristic algorithms studied for this research revealed that while the work on malware detection using nature-inspired methods has been performed, it is not substantial. In their survey paper, Rhmann and Ansari [90] presented that harmony search (HS), various flavors of artificial immune system (AIS), genetic algorithm (GA), and genetic programming-based models have been used for malware detection. Though the researchers did conclude that metaheuristic is effective against unknown malware and help in reducing false positives and false negatives, the accuracy attained using these approaches was not discussed. In another survey paper, GA, particle swarm optimization (PSO), and harmony search algorithms have been leveraged in malware exposure [70]. While the work presented by Souri and Hosseini [70] exhibits that use of nature-inspired metaheuristic algorithms is still at its infancy, the researchers did propose that use of these algorithms can speed up the processing time and improve the accuracy factor—a critical element in CI. In his research report, Abuelsamid [22] claims that heuristic-based algorithms have the potential of introducing false positives, which are acceptable in a general computing system, but not in a CI. The author further argues that the probability of false positives using a heuristic-based approach varies between 2% to 5%, which cannot be tolerated in a CI setting, because it increases the risk of fatality. Therefore, leveraging the real potential of nature-inspired algorithms is considered essential for the trustworthiness of CIs.

10. Impact of False Positives on CIs

A system receiving a positive signal and reacting oppositely is defined as a false positive. In a malware context, a FP is a condition when a file is wrongly classified as being malicious [39]. A well-known example of a FP is when an email system translates a legitimate incoming email as spam and forwards it to the spam folder. Such a scenario has the potential to miss essential and critical emails. In the previous sections, we mentioned why FPs are critical for correct classification of malicious and benign traffic considering the socio-economic conditions associated with CIs. Therefore, in CIs supported by CPS, such failures could be catastrophic and life-threatening. Table 5 describes the FP conditions and their impact on CIs. Table 5 describes the FP conditions and their impact on CIs using hypothetical scenarios.

Table 5. Impact of false positives on CIs.

	Sub-Domain	Impact of False Positive
Transportation Systems	Autonomous Vehicle	A high-speed car on a freeway suddenly applies brakes after receiving a non-life-threatening alert such as minor debris when it could have easily crossed them. Such a situation has the potential for severe accidents to the following vehicles. A similar situation could also exist when the automated system does not trigger an alarm about fuel status causing the car to stop abruptly.
	Autonomous Rail System	A moving train receives a stop (red) signal, but processes it as a moving (green) signal and does not stop, leading to fatal accidents.
Financial Services	Banking System	A malfunction on one automated teller machine (ATM) requires a system to shut it down. However, the system shuts all ATMs in that area affecting a broader community to use the services.
Defense and Industrial Base	Defense	A frigate while manoeuvring generates an alert of encountering an enemy ship which is otherwise a friendly ship. Such an alert has the potential of causing an additional activity on the frigate, thus diverting it from its real assignment.
		Missiles with the potential of being redirected once fired by providing new GPS coordinates can be miscued when the system involved wrongly translates the given parameters.
Energy	Power Grid	A smart grid receives an alert to shut down a few stations due to any natural cause but shuts down the entire network, or the networks that are not affected by the reason. Such false alerts deprive hundreds of thousands of people of energy supply, while also compromising vital healthcare services (a separate CI domain).
Information Technology	Automated Superstore Services	System installed in a smart home does not open gates/doors for a legitimate person while coming inside.
	Smart Home System	System generates an alert about the low stock level of certain items whereas the shelves still have enough quantity of items available. This scenario could impact the goodwill of the superstore in a competitive environment. Moreover, this false trigger could lead to increased manufacturing/production rate.

11. Cyber-Physical System Malware Countermeasures

Due to the targeted, sophisticated, and aggressive nature of cyber-attacks to disrupt the normal business operations of CPSes, a proactive approach is essential to be built and tested for a robust response. For resilient CPSes, an all-inclusive contingency plan (CP) consisting of a business impact analysis (BIA), a business continuity plan (BCP), an incident response plan (IRP), and a disaster recovery plan (DRP) is an essential element to anticipate, react, and recover from cyber intrusions. This is deemed necessary as disabling a CI generates fear and turmoil, in addition to crippling the nation's economy, as seen in the cases of Stuxnet, BlackEnergy, Industroyer, and Triton. To counter this unprecedented threat, the guardians of CPSes should comply with the five basic principles of security, namely layering, limiting, diversity, obscurity and simplicity. This approach is well known as 'Defense-in-Depth' (DiD), which is the concept of implementing multiple barriers to protect a digital network from the adversary such that if one mechanism fails, another will already be in place to prevent an attack [36]. Figure 14 presents a holistic view following the DiD approach that must be implemented to protect CPSes from an adversary's attacks. An ideal strategy to protect critical systems is to implement controls that deter the attacking entity from penetrating the target network. If this fails, the security mechanism should allow for detecting an infiltration, which in turn triggers prevention and correction mechanisms.

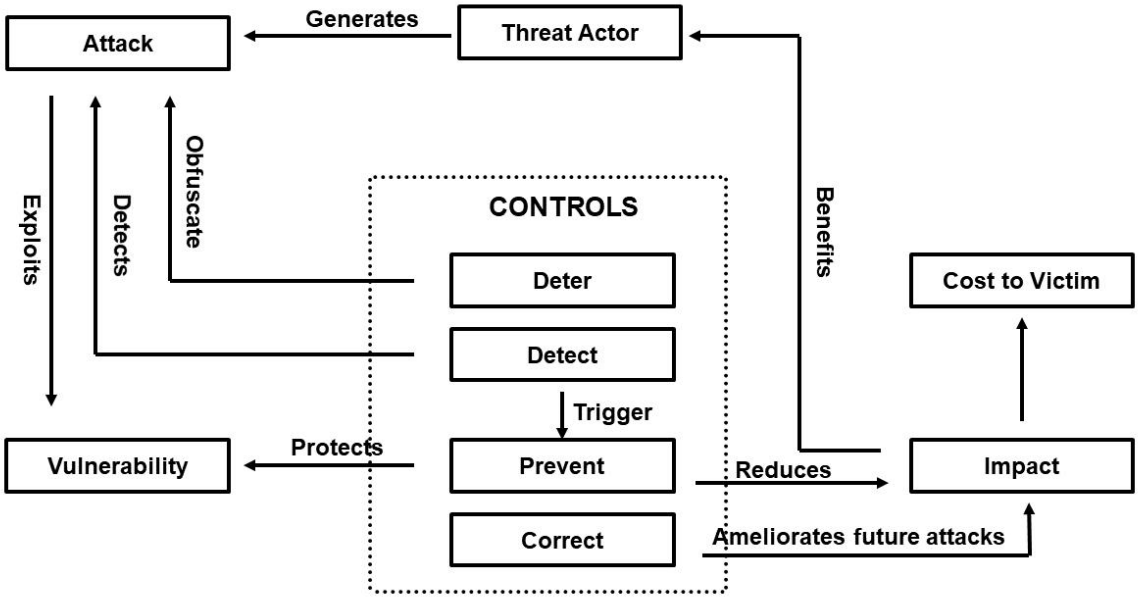

Figure 14. Holistic view to protect CPSes from cyber intrusions [98].

Developed nations have taken several initiatives in securing their CPSes through the development of best practice guides to ensure the security of critical systems from the risks of espionage and coercion, particularly from adversary nations. Australia's Critical Infrastructure Centre has developed six security principles through active involvement from key stakeholders that enables government and industry operators of CPSes to minimize the chances of a compromise. These principles include *pre-employment screening to minimize the occurrence of insider threats, sensitive data storage and protection, controlled physical access, effectively securing industrial control systems (ICS), comprehensive risk strategy to manage outsourced and offshore functions, and bringing security awareness and consciousness as an integral part of the organization* [99]. While the principles are developed for a specific CI sector (supply chain), these can be implemented for any other industry to halt malware propagation from spreading into the network before executing its payload. The United States [100] and the United Kingdom through a cyber assessment framework (CAF) [101] has a similar approach in securing their CPSes.

The MITRE ATT&CK framework describes how adversaries can infiltrate into a targeted network and move laterally to maintain persistence [102–104]. The framework was developed from an adversary's point of view, describing the tactics and techniques to compromise a system that they wish to attack, and what operating systems and applications are vulnerable to these techniques using various use cases. The framework also allows the owners with mitigation mechanisms of adversary's techniques to safeguard their networks and evaluate the strength of the currently deployed controls. MITRE recently released specific ATT&CK framework for industrial control systems (ICS) that allows the government and the industry to protect CPSes in a wholesome manner [105,106]. The ICS framework has eight use cases: *adversary emulation, behavioral analysis, cyber threat intelligence enrichment, defense gap assessment, red teaming, security operations centre (SOC) maturity assessment, failure scenario development, and educational resource*. The last two use cases have been included explicitly for industrial control systems to focus on alternative means if a CPS component fails and bridging the gap between people who operate CPSes and those who deal with cyber security issues in the CPS environment, respectively. Contrary to the enterprise ATT&CK framework, which has 14 tactics, the ICS framework has 12 tactics, each with several techniques. These are *initial access, execution, persistence, privilege escalation, evasion, discovery, lateral movement, collection, command and control, inhibit response function, impair process control, and impact*. Starting from the initial access (first tactic), security professionals entrusted with securing CPSes can look into their system to map the technique(s) that an adversary can employ against the given CPS. Following this, all tactics can be analyzed with the help of the current security controls, allowing the organization to know how the adversary can infiltrate into the system [106,107]. This would enable a comprehensive review of the security controls and their configurations and how they can be improved before the actual attack occurs.

12. Conclusions and Future Directions

CPSes are increasingly being used to improve CIs processes to deliver efficient and reliable services. Consumers and private sector industries are not the only beneficiaries of the inherent advantages of these systems; considering the 16 sectors, as introduced by the United States Department of Homeland Security (DHS), a government's infrastructure is also relying on CPSes. As cyber-attacks are becoming more and more sophisticated, a secure and robust mechanism is the need of the hour to protect physical operations from being sabotaged. This paper presented a comprehensive review of the state-of-the-art malware detection approaches that are currently being used and further discussed their limitations with a view that work on CI-specific malware is still in the early stages. This is verified by summarizing the work of various authors who have analyzed different malware less than those that have attacked CPSes. The survey draws the relationship between the CIs and CPSes and addressed their criticality, when compromised, with the help of real-world malware attacks, hypothetical scenarios, and the impact of false positives leading to catastrophic failures that could be devastating for any nation. The paper further presented the strong potential of nature-inspired metaheuristic algorithms to achieve optimization in malware detection processes in an efficient and time-intelligent manner. Further, our survey shows that the researchers in the cyber domain have barely utilized the benefits of nature-inspired algorithms.

Therefore, the next-generation defense systems would necessitate leveraging the use of nature-inspired techniques in attaining high accuracy and resilience with low false positives for the safe, continuous, reliable and secure industrial operations of CPSes. Adopting such an approach will allow CI service providers to identify malware threats facing their network in an automated, real-time manner, ultimately providing actionable intelligence to enable the mitigation of these threats. Thus, the paper has identified the following gaps, which the researchers aim to address in the subsequent phase:

1. Non-availability of benign and malware datasets that researchers have used thus far. Where available, it lacks the method used to create these datasets making it impossible to reproduce;
2. Lack of work on malware analysis directly relevant to CPSes. We believe that analysis of malware that have compromised CIs may allow us to classify features that distinguish general and CPS malware. A successful outcome has the potential for more reliable and robust CIs. Our analyses thus far also show that a very limited number of CPS bound malware executables/binaries (their variants) are publicly available. This presents a more daunting task as the analyses on a relatively small number of samples would give a smaller dataset;
3. Limited or almost negligible use of available nature-inspired metaheuristic algorithms that can be leveraged to bring optimization in malware-detection processes;
4. Restricted work in reducing the false positives specific to CPSes as it directly relates to the risk of fatality which could be seen as detrimental to bringing trust-level for the consumers.

Author Contributions: M.I.M.: conceptualization, methodology, data curation, formal analysis, investigation, writing—original draft. A.I.: methodology, validation, writing—reviewing and editing, supervision, project administration. P.H.: methodology, validation, supervision, writing—reviewing and editing. L.F.S.: validation, writing—reviewing and editing, supervision. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the Department of Jobs, Tourism, Science (DTSI), Western Australia, through the Science Industry PhD Fellowship Program (Grant No G1004105). We also thank NCC Group for providing industry mentor support as part of the Science Industry PhD Fellowship Program.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing is not applicable to this article.

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

References

1. United States Cyber Command. *Advanced Cyber Industrial Control System Tactics, Techniques, and Procedures (ACI TTP) for Department of Defense (DoD) Industrial Control Systems (ICS)*; United States Department of Defense: Arlington, VA, USA, 2017. Available online: https://apps.dtic.mil/dtic/tr/fulltext/u2/1040233.pdf (accessed on 27 March 2023).
2. Cyber and Infrastructure Security Centre; Australian Government Department of Home Affairs. *Defining Critical Infrastrucure*. Available online: https://www.cisc.gov.au/what-is-the-cyber-and-infrastructure-security-centre/defining-critical-infrastructure (accessed on 29 March 2023).
3. Barrett, M.P. *Framework for Improving Critical Infrastructure Cybersecurity*; National Institute of Standards and Technology (NIST): Gaithersburg, MD, USA, 2018. Available online: https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.04162018.pdf (accessed on 27 March 2023).
4. Rzeszutek, E.; Mazurczyk, W. Nature-inspired analogies and metaphors for cyber security. In *Nature-Inspired Cyber Security and Resiliency—Fundamentals, Techniques and Applications*; Institution of Engineering and Technology (IET): Stevenage, Herts, UK, 2019; pp. 1–28. [CrossRef]
5. Faris, H.; Aljarah, I.; Mirjalili, S.; Castillo, P.A.; Merelo, J.J. EvoloPy: An open-source nature-inspired optimization framework in Python. In Proceedings of the 8th International Joint Conference on Computational Intelligence—ECTA (IJCCI), Porto, Portugal, 11 November 2016. [CrossRef]
6. Yang, X.-S. *Optimization Techniques and Applications with Examples*; John Wiley & Sons: Hoboken, NJ, USA, 2018. [CrossRef]
7. Yang, X.-S.; Deb, S. Engineering optimisation by cuckoo search. *Int. J. Math. Model. Numer. Optim.* **2010**, *1*, 330–343. [CrossRef]
8. Cybersecurity & Infrastructure Security Agency (CISA). *Critical Infrastructure Sectors*; US Department of Homeland Security: Washington, DC, USA. Available online: https://www.cisa.gov/critical-infrastructure-sectors (accessed on 27 March 2023).
9. Fujita, H.; Gaeta, A.; Loia, V.; Orciuoli, F. Resilience analysis of critical infrastructures: A cognitive approach based on granular computing. *IEEE Trans. Cybern.* **2019**, *49*, 1835–1848. [CrossRef]
10. Russell, B.; Van Duren, D. *Practical Internet of Things Security: Design a Security Framework for an Internet Connected Ecosystem*, 2nd ed.; Packt Publishing Ltd.: Birmingham, UK, 2018.

11. Maynard, P.; McLaughlin, K.; Sezer, S. Decomposition and sequential-AND analysis of known cyber-attacks on critical infrastructure control systems. *J. Cybersecur.* **2020**, *6*, tyaa020. [CrossRef]
12. Yaacoub, J.-P.A.; Salman, O.; Noura, H.N.; Kaaniche, N.; Chehab, A.; Malli, M. Cyber-physical systems security: Limitations, issues and future trends. *Microprocess. Microsyst.* **2020**, *77*, 103201. [CrossRef]
13. Mamta; Gupta, B.B.; Li, K.C.; Leung, V.C.M.; Psannis, K.E.; Yamaguchi, S. Blockchain-Assisted Secure Fine-Grained Searchable Encryption for a Cloud-Based Healthcare Cyber-Physical System. *IEEE/CAA J. Autom. Sin.* **2021**, *8*, 1877–1890. [CrossRef]
14. Ch, R.; Srivastava, G.; Nagasree, Y.L.; Ponugumati, A.; Ramachandran, S. Robust Cyber-Physical System Enabled Smart Healthcare Unit Using Blockchain Technology. *Electronics* **2022**, *11*, 3070. [CrossRef]
15. Nguyen, G.N.; Viet, N.H.L.; Elhoseny, M.; Shankar, K.; Gupta, B.B.; El-Latif, A.A.A. Secure blockchain enabled Cyber–physical systems in healthcare using deep belief network with ResNet model. *J. Parallel Distrib. Comput.* **2021**, *153*, 150–160. [CrossRef]
16. Humayed, A.; Lin, J.; Li, F.; Luo, B. Cyber-physical systems security—A survey. *IEEE Internet Things J.* **2017**, *4*, 1802–1831. [CrossRef]
17. Jacobson, C. The importance of cyber-physical systems for industry. *ERCIM News* **2014**, *97*, 4.
18. Anthi, E.; Williams, L.; Burnap, P.; Jones, K. A three-tiered intrusion detection system for industrial control systems. *J. Cybersecur.* **2021**, *7*, tyab006. [CrossRef]
19. Nazarenko, A.A.; Safdar, G.A. Survey on security and privacy issues in cyber physical systems. *AIMS Electron. Electr. Eng.* **2019**, *3*, 111–143. [CrossRef]
20. Song, H.; Fink, G.; Jeschke, S. *Security and Privacy in Cyber-Physical Systems: Foundations, Principles, and Applications*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2018. [CrossRef]
21. Wang, Z.; Xie, W.; Wang, B.; Tao, J.; Wang, E. A Survey on Recent Advanced Research of CPS Security. *Appl. Sci.* **2021**, *11*, 3751. [CrossRef]
22. Abuelsamid, S. Autonomous Automotive Cybersecurity. Available online: https://karambasecurity.com/static/pdf/Autonomous-Automotive-Cybersecurity-Report.pdf (accessed on 27 March 2023).
23. Hassanzadeh, A.; Rasekh, A.; Galelli, S.; Aghashahi, M.; Taormina, R.; Ostfeld, A.; Banks, M.K. A review of cybersecurity incidents in the water sector. *J. Environ. Eng.* **2020**, *146*, 03120003–03120013. [CrossRef]
24. Hill, M. Water Treatment Plant Hit by Cyber-Attack. Infosecurity Group. 2016. Available online: https://www.infosecurity-magazine.com/news/water-treatment-plant-hit-by/ (accessed on 27 March 2023).
25. Mordor Intelligence. Internet of Things (IoT) Market—Growth, Trends, COVID-19 Impact, and Forecasts (2021–2026). 2021. Available online: https://www.reportlinker.com/p06067771/Internet-of-Things-IoT-Market-Growth-Trends-COVID-19-Impact-and-Forecasts.html (accessed on 29 March 2023).
26. Scheuermann, J.E. Cyber-Physical Attacks on Critical Infrastructure: What's Keeping Your Insurer Awake at Night? Legal Insight, Issue. K. L. Gates. 2017. Available online: https://files.klgates.com/files/publication/b54ead7b-7166-45a7-909a-e990c5ba85f8/presentation/publicationattachment/b186efde-3b30-4eda-86d1-ebdd8badd030/insurance_coverage_alert_01242017.pdf (accessed on 27 March 2023).
27. Lloyd's. Business Blackout–The Insurance Implications of a Cyber Attack on the US Power Grid. 2015. Available online: https://www.jbs.cam.ac.uk/wp-content/uploads/2020/08/crs-lloyds-business-blackout-scenario.pdf (accessed on 29 March 2023).
28. Department of Home Affairs. *Australia's Cyber Security Strategy 2020*; Australian Government Department of Home Affairs: Canberra, Australia, 2020. Available online: https://www.homeaffairs.gov.au/cyber-security-subsite/files/cyber-security-strategy-2020.pdf (accessed on 27 March 2023).
29. Malwarebytes Labs. *2020 State of Malware Report*; Malwarebytes Labs: Santa Clara, CA, USA, 2020. Available online: https://resources.malwarebytes.com/files/2020/02/2020_State-of-Malware-Report.pdf (accessed on 27 March 2023).
30. The Australian Cyber Security Centre (ACSC). *SDBBot Targeting Health Sector*; The Australian Cyber Security Centre (ACSC): Canberra, Australia, 2020. Available online: https://www.cyber.gov.au/about-us/alerts/sdbbot-targeting-health-sector (accessed on 29 March 2023).
31. Cimpanu, C. German tech giant Software AG down after ransomware attack. *ZDNet Security*. 2020. Available online: https://www.zdnet.com/article/german-tech-giant-software-ag-down-after-ransomware-attack/ (accessed on 27 March 2023).
32. CrowdStrike. 2021 Global Threat Report. 2021. Available online: https://go.crowdstrike.com/rs/281-OBQ-266/images/Report2021GTR.pdf (accessed on 27 March 2023).
33. Cybersecurity & Infrastructure Security Agency (CISA). *APT Cyber Tools Targeting ICS/SCADA Devices*; Cybersecurity & Infrastructure Security Agency: Arlington, VA, USA, 2022. Available online: https://www.cisa.gov/uscert/ncas/alerts/aa22-103a (accessed on 27 March 2023).
34. Dragos. *Pipedream: Chernovite's Emerging Malware Targeting Industrial Control Systems*; Dragos Inc.: Hanover, MD, USA, 2022. Available online: https://hub.dragos.com/hubfs/116-Whitepapers/Dragos_ChernoviteWP_v2b.pdf?hsLang=en (accessed on 27 March 2023).
35. Souppaya, M.; Scarfone, K. *Guide to Malware Incident Prevention and Handling for Desktops and Laptops*; National Institue of Standards and Technology (NIST): Gaithersburg, MD, USA, 2013. Available online: https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-83r1.pdf (accessed on 27 March 2023).
36. Stallings, W.; Brown, L. *Computer Security: Principles and Practice*, 4th ed.; Pearson Education: New York, NY, USA, 2018.

37. Symantec. *ISTR—Information Security Threat Report*; Symantec: Mountain View, CA, USA, 2018. Available online: https://www.phishingbox.com/assets/files/images/Symantec-Internet-Security-Threat-Report-2018.pdf (accessed on 27 March 2023).
38. McAfee Labs. McAfee Labs Threat Report. Available online: https://www.dailyhostnews.com/mcafee-labs-threat-report-q3-2017-identifies-57-6-million-new-malware-samples-increase-10-q2 (accessed on 29 March 2023).
39. Ye, Y.; Li, T.; Adjeroh, D.; Iyengar, S.S. A survey on malware detection using data mining techniques. *ACM Comput. Surv.* **2017**, *50*, 1–40. [CrossRef]
40. Bettany, A.; Halsey, M. *Windows Virus and Malware Troubleshooting*; Apress: Berkeley, CA, USA, 2017. [CrossRef]
41. The Kosciuszko Institute. The Kosciuszko Institute Cyber-Security Forecasts for 2018. *Cyber Defense Magazine*. 2018. Available online: https://www.cyberdefensemagazine.com/cyber-security-in-2018-the-kosciuszko-institute-publishes-experts-forecasts/ (accessed on 27 March 2023).
42. Gandotra, E.; Bansal, D.; Sofat, S. Malware analysis and classification: A survey. *J. Inf. Secur.* **2014**, *5*, 56–64. [CrossRef]
43. The Australian Cyber Security Centre (ACSC). *Ransomware*; The Australian Cyber Security Centre (ACSC): Canberra, Australia. Available online: https://www.cyber.gov.au/learn-basics/view-resources/glossary/r (accessed on 29 March 2023).
44. Connolly, L.Y.; Wall, D.S.; Lang, M.; Oddson, B. An empirical study of ransomware attacks on organizations: An assessment of severity and salient factors affecting vulnerability. *J. Cybersecur.* **2020**, *6*, tyaa023. [CrossRef]
45. Hampton, N.; Baig, Z.A. Ransomware: Emergence of the cyber-extortion menace. In Proceedings of the 13th Australian Information Security Management Conference, Perth, WA, Australia, 30 November–2 December 2015; SRI Security Research Institute, Edith Cowan University: Perth, Australia, 2015. [CrossRef]
46. Mathur, K.; Hiranwal, S. A survey on techniques in detection and analyzing malware executables. *Int. J. Adv. Res. Comput. Sci. Softw. Eng.* **2013**, *3*, 422–428.
47. NIST. *Security and Privacy Controls for Information Systems and Organizations*; National Institute of Standards and Technology (NIST): Gaithersburg, MD, USA, 2020. Available online: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-53r5.pdf (accessed on 27 March 2023).
48. The Australian Cyber Security Centre (ACSC). *Advanced persistent THREAT (APT)*; The Australian Cyber Security Centre (ACSC): Canberra, Australia. Available online: https://www.cyber.gov.au/learn-basics/view-resources/glossary/a (accessed on 29 March 2023).
49. Lockheed Martin. The Cyber Kill Chain®. Available online: https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html (accessed on 27 March 2023).
50. Andreasson, K. Is there a conclusion to cybersecurity? In *Cybersecurity: Public Sector Threats and Responses*, 1st ed.; Andreasson, K., Ed.; CRC Press, Taylor & Francis Group: Boca Raton, FL, USA, 2011; pp. 327–338. [CrossRef]
51. Li, S.; Zhang, Q.; Wu, X.; Han, W.; Tian, Z. Attribution Classification Method of APT Malware in IoT Using Machine Learning Techniques. *Secur. Commun. Netw.* **2021**, *2021*, 9396141. [CrossRef]
52. MITRE ATT&CK. MITRE ATT&CK Groups. 2023. Available online: https://attack.mitre.org/groups/ (accessed on 27 March 2023).
53. Waldman, J.; Cordona, E. Top 25 Threat Actors—2019 Edition. SBS CyberSecurity. Available online: https://sbscyber.com/resources/top-25-threat-actors-2019-edition (accessed on 27 March 2023).
54. Trellix. *The Threat Report*; Trellix: Milpitas, CL, USA, 2021. Available online: https://www.trellix.com/en-us/advanced-research-center/threat-reports/feb-2023.html (accessed on 29 March 2023).
55. Sikorski, M.; Honig, A. *Practical Malware Analysis: The Hands-on Guide to Dissecting Malicious Software*; No Starch Press: San Francisco, CA, USA, 2012.
56. Cui, Z.; Xue, F.; Cai, X.; Cao, Y.; Wang, G.; Chen, J. Detection of malicious code variants based on deep learning. *IEEE Trans. Ind. Inform.* **2018**, *14*, 3187–3196. [CrossRef]
57. Damodaran, A.; Troia, F.D.; Visaggio, C.A.; Austin, T.H.; Stamp, M. A comparison of static, dynamic, and hybrid analysis for malware detection. *J. Comput. Virol. Hacking Tech.* **2017**, *13*, 1–12. [CrossRef]
58. Elhadi AA, E.; Maarof, M.A.; Osman, A.H. Malware detection based on hybrid signature behaviour application programming interface call graph. *Am. J. Appl. Sci.* **2012**, *9*, 283–288.
59. Islam, R.; Tian, R.; Batten, L.M.; Versteeg, S. Classification of malware based on integrated static and dynamic features. *J. Netw. Comput. Appl.* **2013**, *36*, 646–656. [CrossRef]
60. Kaur, H.; Kalra, M. An approach for malware detection and predictive analysis using artificial neural networks. *Int. Ref. J. Rev. Res.* **2016**, *4*, 6–12.
61. Yan, S.; Ren, J.; Wang, W.; Sun, L.; Zhang, W.; Yu, Q. A Survey of Adversarial Attack and Defense Methods for Malware Classification in Cyber Security. *IEEE Commun. Surv. Tutor.* **2023**, *25*, 467–496. [CrossRef]
62. Gaurav, A.; Gupta, B.B.; Panigrahi, P.K. A comprehensive survey on machine learning approaches for malware detection in IoT-based enterprise information system. *Enterp. Inf. Syst.* **2023**, *17*, 2023764. [CrossRef]
63. Ranveer, S.; Hiray, S. Comparative analysis of feature extraction methods of malware detection. *Int. J. Comput. Appl.* **2015**, *120*, 1–7. [CrossRef]
64. Shijoa, P.V.; Salim, A. Integrated static and dynamic analysis for malware detection. *Procedia Comput. Sci.* **2015**, *46*, 804–811. [CrossRef]

65. Vinod, P.; Laxmi, V.; Gaur, M.S. Survey on malware detection methods. In Proceedings of the 3rd Hackers' Workshop on Computer and Internet Security (IITKHACK'09), Prabhu Goel Research Centre for Computer & Internet Security, Kanpur, India, 17–19 March 2009; Department of Computer Science and Engineering, Indian Institute of Technology Kanpur: Kanpur, India, 2009.
66. Saeed, I.A.; Selamat, A.; Abuagoub AM, A. A survey on malware and malware detection systems. *Int. J. Comput. Appl.* **2013**, *67*, 25–31. [CrossRef]
67. Cloonan, J. *Advanced Malware Detection—Signatures vs. Behavior Analysis*; Info Security Group, 2017. Available online: https://www.infosecurity-magazine.com/opinions/malware-detection-signatures/ (accessed on 27 March 2023).
68. EL Boujnouni, M.; Jedra, M.; Zahid, N. New malware detection framework based on N-grams and support vector domain description. In Proceedings of the 2015 11th International Conference on Information Assurance and Security (IAS), Marrakech, Morocco, 14–16 December 2015. [CrossRef]
69. Bazrafshan, Z.; Hashemi, H.; Fard, S.M.H.; Hamzeh, A. A survey on heuristic malware detection techniques. In Proceedings of the 5th Conference on Information and Knowledge Technology (IKT), Shiraz, Iran, 28–30 May 2013. [CrossRef]
70. Souri, A.; Hosseini, R. A state-of-the-art survey of malware detection approaches using data mining techniques. *Hum. -Cent. Comput. Inf. Sci.* **2018**, *8*, 3. [CrossRef]
71. Chumachenko, K. Machine Learning Methods for Malware Detection and Classification. Bachelor's Thesis, University of Applied Sciences, Kouvola, Finland, 2017. Available online: http://urn.fi/URN:NBN:fi:amk-201703103155 (accessed on 27 March 2023).
72. Wüchner, T.; Cisłak, A.; Ochoa, M.; Pretschner, A. Leveraging compression-based graph mining for behavior-based malware detection. *IEEE Trans. Dependable Secur. Comput.* **2017**, *16*, 99–112. [CrossRef]
73. Mohaisen, A.; Alrawi, O.; Mohaisen, M. AMAL: High-fidelity, behavior-based automated malware analysis and classification. *Comput. Secur.* **2015**, *52*, 251–266. [CrossRef]
74. Burnap, P.; French, R.; Turner, F.; Jones, K. Malware classification using self organising feature maps and machine activity data. *Comput. Secur.* **2018**, *73*, 399–410. [CrossRef]
75. LeDoux, C.; Lakhotia, A. Malware and machine learning. In *Intelligent Methods for Cyber Warfare*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 1–42.
76. Santos, I.; Devesa, J.; Brezo, F.; Nieves, J.; Bringas, P.G. *Opem: A Static-Dynamic Approach for Machine-Learning-Based Malware Detection*; Springer: Berlin/Heidelberg, Germany, 2013.
77. Anderson, B.; Storlie, C.; Lane, T. Improving malware classification: Bridging the static/dynamic gap. In Proceedings of the Proceedings of the 5th ACM Workshop on Security and Artificial Intelligence, Raleigh, NC, USA, 19 October 2012; pp. 3–14.
78. Russell, S.; Dewey, D.; Tegmark, M. Research priorities for robust and beneficial artificial intelligence. *AI Mag.* **2015**, *36*, 105–114. [CrossRef]
79. Nieva, R. Google Says It's Designing Duplex with 'Disclosure Built-in'. *C|Net*. 2018. Available online: https://www.cnet.com/news/google-says-its-designing-duplex-with-disclosure-built-in/ (accessed on 27 March 2023).
80. Goode, L. *How Google's Eerie Robot Phone Calls Hint at AI's Future*; Wired: New York, NY, USA, 2018. Available online: https://www.wired.com/story/google-duplex-phone-calls-ai-future/ (accessed on 27 March 2023).
81. Alpaydin, E. *Machine Learning: The New AI*; MIT Press: Cambridge, MA, USA, 2016.
82. Dua, S.; Du, X. *Data Mining and Machine Learning in Cybersecurity*, 1st ed.; Auerbach Publications, Taylor & Francis Group: New York, NY, USA, 2011. [CrossRef]
83. Luke, S. *Essentials of Metaheuristics*, 2nd ed.; Lulu: Morrisville, NC, USA, 2013. Available online: https://cs.gmu.edu/~sean/book/metaheuristics/ (accessed on 27 March 2023).
84. Talbi, E.-G. *Metaheuristics: From Design to Implementation*; John Wiley & Sons: Hoboken, NJ, USA, 2009; Volume 74.
85. Yang, X.-S. *Nature-Inspired Metaheuristic Algorithms*, 2nd ed.; Luniver Press: Frome, UK, 2010.
86. Yang, X.-S. *Nature-Inspired Optimization Algorithms*, 2nd ed.; Academic Press: Cambridge, MA, USA, 2021. [CrossRef]
87. Arora, T.; Gigras, Y. A survey of comparison between various metaheuristic techniques for path planning problem. *Int. J. Comput. Eng. Sci.* **2013**, *3*, 62–66.
88. Mirjalili, S.; Gandomi, A.H.; Mirjalili, S.Z.; Saremi, S.; Faris, H.; Mirjalili, S.M. Salp Swarm algorithm: A bio-inspired optimizer for engineering design problems. *Adv. Eng. Softw.* **2017**, *114*, 163–191. [CrossRef]
89. Gandomi, A.H.; Yang, X.-S.; Alavi, A.H. Cuckoo search algorithm: A metaheuristic approach to solve structural optimization problems. *Eng. Comput.* **2013**, *29*, 17–35. [CrossRef]
90. Rhmann, W.; Ansari, G.A. Use of metaheuristic algorithms in malware detection. *Int. J. Recent Innov. Trends Comput. Commun.* **2017**, *5*, 1370–1374. [CrossRef]
91. Suh, W.-J.; Park, C.-S.; Kim, D.-W. Heuristic vs. Meta-Heuristic Optimization for Energy Performance of a Post Office Building. In Proceedings of the Building Simulation 2011: 12th Conference of International Building Performance Simulation Association (IBPSA), Sydney, Australia, 14–16 November 2011. Available online: http://www.ibpsa.org/proceedings/BS2011/P_1313.pdf (accessed on 27 March 2023).
92. Almomani, A.; Alweshah, M.; Khalayleh, S.A.; Al-Refai, M.; Qashi, R. Metaheuristic algorithms-based feature selection approach for intrusion detection. In *Machine Learning for Computer and Cyber Security*, 1st ed.; Gupta, B.B., Sheng, M., Eds.; CRC Press: Boca Raton, FL, USA, 2019; pp. 184–208.

93. Fister, I., Jr.; Yang, X.-S.; Fister, I.; Brest, J.; Fister, D. A brief review of nature-inspired algorithms for optimization. *Electrotech. Rev.* **2013**, *80*, 116–122.
94. Luthra, I.; Chaturvedi, S.K.; Upadhyay, D.; Gupta, R. Comparative study on nature inspired algorithms for optimization problem. In Proceedings of the International conference of Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 20–22 April 2017. [CrossRef]
95. Sörensen, K. Metaheuristics—The metaphor exposed. *Int. Trans. Oper. Res.* **2015**, *22*, 3–18. [CrossRef]
96. Mthunzi, S.N.; Benkhelifa, E.; Bosakowski, T.; Hariri, S. *A bio-inspired approach to cyber security In Machine Learning for Computer and Cyber Security*, 1st ed.; Gupta, B.B., Sheng, M., Eds.; CRC Press, Taylor and Francis Group: Boca Raton, FL, USA, 2019; pp. 75–104.
97. Nanda, S.J.; Panda, G. A survey on nature inspired metaheuristic algorithms for partitional clustering. *Swarm Evol. Comput.* **2014**, *16*, 1–18. [CrossRef]
98. Malik, M.I.; McAteer, I.N.; Hannay, P.; Ibrahim, A.; Baig, Z.; Zheng, G. Cyber security for Network of Things (NoTs) in military systems: Challenges countermeasures. In *Security Analytics for the Internet of Everything*; Ahmed, M., Ullah, A.S.S.M.B., Pathan, A.-S.K., Eds.; CRC Press, Taylor & Francis Group: Boca Raton, FL, USA, 2020; pp. 231–250. [CrossRef]
99. Critical Infrastructure Centre. Protecting Your Critical Infrastructure Asset from Foreign Involvement Risk. Available online: https://www.homeaffairs.gov.au/nat-security/files/cic-best-practice-guidance-supply-chains.pdf (accessed on 27 March 2023).
100. Cybersecurity & Infrastructure Security Agency (CISA). *Critical Infrastructure Security and Resilience*; US Department of Homeland Security: Washington, DC, USA. Available online: https://www.dhs.gov/topic/critical-infrastructure-security (accessed on 27 March 2023).
101. National Cyber Security Centre. *Cyber Assessment Framework*; National Cyber Security Centre: London, UK. Available online: https://www.ncsc.gov.uk/collection/caf/cyber-assessment-framework (accessed on 27 March 2023).
102. Piazza, A. *ATT&CKing Threat Management: A Structured Methodology for Cyber Threat Analysis*; The SANS Institute: North Bethesda, MD, USA, 2019. Available online: https://www.sans.org/white-papers/39090/ (accessed on 29 March 2023).
103. Strom, B.E.; Applebaum, A.; Miller, D.P.; Nickels, K.C.; Pennington, A.G.; Thomas, C.B. *MITRE ATT&CK®: Design and Philosophy*; The MITRE Corporation: McLean, VA, USA, 2020. Available online: https://attack.mitre.org/docs/ATTACK_Design_and_Philosophy_March_2020.pdf (accessed on 27 March 2023).
104. The Mitre Corporation. MITRE ATT&CK™ Framework [Video]. *YouTube*. 28 February 2018. Available online: https://www.youtube.com/watch?v=0BEf6s1iu5g&t=207s (accessed on 27 March 2023).
105. CyberX. Addressing the MITRE ATT&CK for ICS Matrix. *CyberX*. Available online: https://scadahacker.com/library/Documents/White_Papers/CyberX%20-%20Addressing%20the%20MITRE%20ATTACK%20for%20ICS%20Matrix.pdf (accessed on 29 March 2023).
106. Alexander, O.; Belisle, M.; Steele, J. *MITRE ATT&CK® for Industrial Control Systems: Design and Philosophy*; The MITRE Corporation: McLean, VA, USA, 2020. Available online: https://attack.mitre.org/docs/ATTACK_for_ICS_Philosophy_March_2020.pdf (accessed on 29 March 2023).
107. Alexander, O.; Slowik, J. Introducing MITRE ATT&CK™ for ICS and Why It Matters [Video]. *YouTube*. 18 January 2020. Available online: https://www.youtube.com/watch?v=NARspb8QfFE (accessed on 27 March 2023).