IEEE Access
The journal for rapid open access publishing

IEEE Xplore Digital Library
http://ieeexplore.ieee.org/Xplore

Industrial and Critical Infrastructure Security: Technical Analysis of Real-Life Security Incidents

GEORGIOS MICHAIL MAKRAKIS[1], CONSTANTINOS KOLIAS[1], GEORGIOS KAMBOURAKIS[2], CRAIG RIEGER[3], JACOB BENJAMIN[4]

[1]Department of Computer Science, University of Idaho, Idaho Falls, ID, 83402 USA, (e-mail: gmakrakis@uidaho.edu, kolias@uidaho.edu)
[2]European Commission, Joint Research Centre (JRC), 21027 Ispra, Italy, (e-mail: georgios.kampourakis@ec.europa.eu)
[3]Idaho National Laboratory (INL), Idaho Falls, ID, 83402, USA, (e-mail: craig.rieger@inl.gov)
[4]Dragos, Hanover, MD 21076 USA, e-mail: (jbenjamin@dragos.com)

Corresponding author: Georgios Kambourakis (e-mail: georgios.kampourakis@ec.europa.eu).

Prepared as part of the Laboratory Directed Research and Development program under DOE Idaho Operations Office Contract DE-AC07-05ID14517. We also acknowledge the Resilient Control and Instrumentation Systems (ReCIS) program.

ABSTRACT Critical infrastructures and industrial organizations aggressively move towards integrating elements of modern Information Technology (IT) into their monolithic Operational Technology (OT) architectures. Yet, as OT systems progressively become more and more interconnected, they silently have turned into alluring targets for diverse groups of adversaries. Meanwhile, the inherent complexity of these systems, along with their advanced-in-age nature, prevents defenders from fully applying contemporary security controls in a timely manner. Forsooth, the combination of these hindering factors has led to some of the most severe cybersecurity incidents of the past years. This work contributes a full-fledged and up-to-date survey of the most prominent threats and attacks against Industrial Control Systems and critical infrastructures, along with the communication protocols and devices adopted in these environments. Our study highlights that threats against critical infrastructure follow an upward spiral due to the mushrooming of commodity tools and techniques that can facilitate either the early or late stages of attacks. Furthermore, our survey exposes that existing vulnerabilities in the design and implementation of several of the OT-specific network protocols and devices may easily grant adversaries the ability to decisively impact physical processes. We provide a categorization of such threats and the corresponding vulnerabilities based on various criteria. The selection of the discussed incidents and identified vulnerabilities aims to provide a holistic view of the specific threats that target Industrial Control Systems and critical infrastructures. As far as we are aware, this is the first time an exhaustive and detailed survey of this kind is attempted.

INDEX TERMS OT, ICS, IIoT, critical infrastructure, cybersecurity, network protocols, security.

I. INTRODUCTION

CRITICAL infrastructures (CI) are comprised of systems and assets so indispensable for the proper function of society that their deterioration will surely prove detrimental to public health, national security, and economic well-being. Such systems cover multiple facets of our everyday lives, but water, energy, communications, and transportation, are considered among the most vital sectors. Security of CI has always been in the epicenter of thorough assessments. Yet until today, security was mainly geared to prevent random accidents and man-made physical assaults. Today, due to the increasingly more significant role of IT systems in the operation of CI, such environments have also become the subject of cyber threats.

An Industrial Control System (ICS) can conceptually be subdivided into the IT and Operational Technology (OT) domains. The IT portion is providing all services that support the business operations. It is comprised of workstations, servers, and databases, all of which are interconnected using IP-based networks. The OT portion focuses on the operational aspects of machinery. The main components of OT systems are domain-specific devices such as Programmable Logic Controllers (PLCs) and Variable-Frequency Drives (VFDs).

Digital Object Identifier: 10.1109/ACCESS.2021.3133348

In their majority, OT consist of components for which their hardware, software, and networking elements, are optimized to have prolonged life of many decades. Interestingly, despite their critical nature, many OT devices do not inherently support any cybersecurity mechanisms.

It can be argued that today, the vast majority of cybersecurity practitioners have only a superficial knowledge of ICS. Yet, due to its pivotal role in CI, it is worth putting ICS security under the magnifying lens. Unlike modern IT systems, OT operate beyond the boundaries of cyberspace and are rather entangled with the physical domain. For this reason, an anomaly caused by a security breach may not only inflict significant economic losses or loss of privacy, which are the typical worst-case scenarios in pure IT systems. In the most dreadful scenarios against OT, such violations may result in wide-reaching environmental destruction or put the safety of citizens at risk.

Naturally, the potential of large-scale and high-profile impact makes ICS inside the CI alluring targets for various adversaries. These actors bear different characteristics than the stereotypical “IT hacker”. They usually have many more resources at their disposal and are driven by motives that range from the mere pursuit of profit but may expand to applying geopolitical pressure.

At the same time, the security of the CI is a daunting task for multiple reasons. Besides the existence of a large number of legacy equipment and the insecurity of the communications in ICS, the complexity of the systems requires operators with a deep understanding of the multiple domains that constitute CI. However, at this point, a dichotomy exists as IT and OT personnel appear to have disjoint training backgrounds. This results in certain aspects of the system, including critical security functions, being viewed by their operators as black-boxes. Therefore, it does not come as a surprise that errors caused by the human factor are still the primary reason behind the majority of the incidents observed in real life.

On top of the aforementioned reasons, one should also take into account the integration of IT and OT realms. This is a tendency that is observed lately across virtually all CI sectors. Driven by the desire to evolve the production processes to fit into the broader context of the fourth industrial revolution (Industry 4.0), the paradigm of the Industrial Internet of Things (IIoT) is introduced to assist such evolution with the extensive use of Big Data and Data Mining techniques. Along with this development comes the need for better security, as the systems become more complex, and the attack surface broadens. Therefore, the organizations should examine the additional risks introduced by this new class of technology integration, how the requirements in the existing and planned standards will be impacted, and how potential cybersecurity solutions can fit at the very beginning of IIoT implementation.

Through the analysis of real-life incidents, several other factors have been identified and are outlined in subsequent portions of the paper. Altogether, the purpose of the work at hand is to offer a full-scale survey around the current state of play of ICS and CI security. After examining the related work and defining an adversarial model, we meticulously examine and categorize the vulnerabilities that originate from the potential insecurities of the integrated cyber systems, including the relevant networking protocols and devices. We perform a deep analysis of the most well-known security incidents against such systems, based on the most preeminent information acquired both from academic work and reports/whitepapers created by the overall security industry. We classify the vulnerabilities and attacks based on the adversary’s methodology, potential damage, attack impact, and available countermeasures. Outside the scope of this paper is any work that discusses in depth the aspects of defense tools, ICS testbeds, and human factor-related threats. For all of the above, we refer the concerned reader to the following [1], [2].

Specifically, the *key contributions* of this work are as follows:

- We offer a comprehensive analysis and discussion of the hitherto major ICS and CI security incidents. This enables a comprehensive view of the attackers’ tactics, techniques, and procedures. The incidents are further taxonomized based on the type of vulnerabilities that leverage the affected level of the ICS, their outcomes, and the possible mitigation strategies.
- A review of the security characteristics of all prominent communication protocols employed in the context of ICS and CI. This line of discussion also elaborates on protocols’ vulnerabilities as pinpointed by the relevant literature, and therefore results in common attack types and major challenges towards providing a better security posture.
- An analysis and discussion of the vulnerabilities that exist in ICS-specific devices that have been discovered in academia and how these vulnerabilities are employed against the control process of ICS and CI.

Given the above, vis-à-vis the relevant literature, the current work is the first to our knowledge to not only provide a extensive, and contemporary analysis of the major security incidents against ICS and CI, but also to blend this analysis with both the practical and theoretical security shortcomings pertaining to all key operational levels of the ICS. Particular focus is given to the vulnerabilities that affect the levels that are closer to the physical process. This choice is made since IT-related vulnerabilities have been examined thoroughly in the past, and that ICS-specific issues present unique characteristics worth investigating.

The remainder of this paper is structured as follows. The next section II provides background information about ICS and CI. Section III addresses the related work. The adversarial model is given in section IV. Section V details on major ICS and CI cybersecurity incidents reported over the last few years. The analysis of the incidents also focuses on the reasons why each attack was prosperous. Section VI concentrates on prominent ICS protocols and elaborates

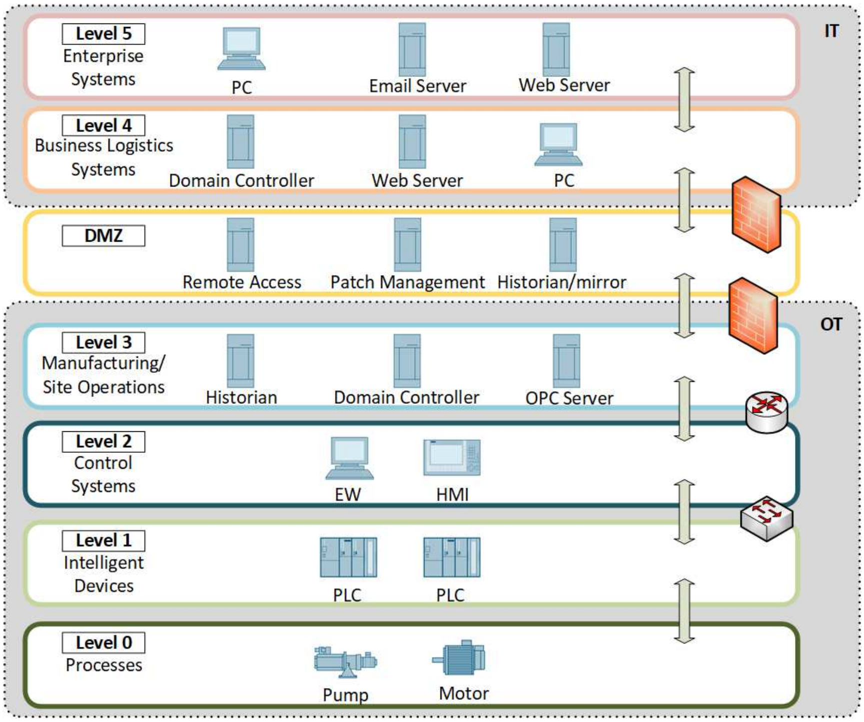


FIGURE 1. An adaptation of the Purdue Enterprise Reference Architecture by ISA-95.

on their potential weaknesses as identified by the relevant literature. Section VII discusses the case of vulnerable ICS devices and the repercussions that they can have to the controlled processes. The last section concludes and offers pointers to future work.

II. BACKGROUND

This section provides brief background information regarding the prevailing terms seen in ICS environments. A level of familiarity with all these concepts is necessary to better comprehend the discussions included in the main sections of the paper.

A. ICS ARCHITECTURE

The Purdue Enterprise Reference Architecture (PERA), or simply Purdue Model [3], is usually adopted when attempting to describe ICS architectures. The model represents the systems that may be tracked in typical ICS into levels. Each one of these represents a distinct section of functionality offered to the ICS.

A brief explanation of each level of the model follows:

- Level 0 — Sensors motors, pumps, and valves, that is, instruments whose main purpose is to provide sensing or actuating capabilities to the system.
- Level 1 — Intelligent devices that sense, monitor, and control the physical processes. Such devices are the Programmable Logic Controllers (PLCs), Proportional-Integral-Derivative (PID) controllers, and the Safety Instrumented System (SIS) controllers.
- Level 2 — Control systems used for supervising and monitoring the physical processes. Among others, this level includes Human-Machine Interface (HMIs) and Engineering Workstations (EWs).
- Level 3 — Manufacturing/Site operations systems used to manage the production workflow for plant-wide control. Devices typically found in this level are the Data Historians, Microsoft Active Directory Domain Controllers, and file servers.
- Industrial Demilitarized Zone (DMZ) — Created to prevent the direct communication between IT and OT environments by installing “broker” services. Proxy servers, database replication servers, and remote access servers are typical entities at this extra level.
- Level 4 - Business/Planning logistics systems used to oversee the IT-related activities of the site operations that support the production process. Some of the systems in this level are application servers, e-mail clients and servers, and Enterprise Resource Planning (ERP) systems.
- Level 5 - The enterprise network used for production and resource data exchange for business-to-business, and business-to-customer purpose services.

A high-level adaptation of the Purdue Model and the main elements of this architecture are illustrated in Figure 1. Based on the above adaptation of the Purdue Model, a typical environment can be subdivided into IT and OT networks. The former comprises conventional PCs, application servers, e-mail servers, and ERP systems. The latter consists of more domain-specific devices (and their accompanying software) that have low hardware specifications, run simple but well-defined tasks, and are seldom updated/replaced.

Nowadays, we observe an apparent convergence of OT and IT network divisions. Therefore, standard IT components are found in the OT realm, such as desktop PCs and industrial devices communicating via either standard protocols such as TCP/UDP or via industrial ones as detailed in Section VI.

Naturally, by moving downwards in the model, different levels of trust for the underlying devices are established. For example, devices that reside inside the enterprise and business levels have lower trust due to their exposure to untrusted networks. The DMZ entities have medium trust, and levels 0 to 3 have high trust. All these are based on the restrictions in terms of the installed equipment and software, as well as the physical access to these systems. Naturally, this is also subject to the particular requirements of each sector and facility of interest. In this work we have tried to follow the Purdue Model as close as possible when describing real-life incidents. However, for reasons that will become apparent, a completely faithful adoption of the model was not possible in all cases.

B. ICS HARDWARE

Level 1 of an ICS typically includes PLCs, Remote Terminal Units (RTUs), Intelligent Electronic Devices (IEDs), and SIS controllers. HMIs belong to level 2, EWs in levels 2 to 3, Data Historians in level 3, remote access servers (or "jump servers") in DMZ, and common network management devices such as layer 3 and layer 2 switches, routers, and firewalls placed in various levels. Other devices can also be present, depending on the requirements and the utilities or products the industrial facility provides. In terms of hardware, the devices in the upper levels of the model (levels 2-3) resemble typical IT devices, e.g., PCs that run Microsoft (MS) Windows OS and multicore processor servers with surplus memory. These devices become more common even in advanced-in-age installations. While HMIs were once separate devices, they are now frequently implemented as desktop applications. A detailed description of the aforementioned types of devices remains out of the scope of this paper; however, the interested reader can obtain further information from the work in [4].

Industrial devices in the lower levels of the model, namely levels 1 or 2 have (a) much lower hardware specifications, say, a few MHz CPU cycles, a few kilobytes or megabytes of memory, (b) run real-time operating systems (RTOS) created for deterministic performance, i.e., the system guarantees a specific amount of CPU cycles between actions, (c) are modular and easy to expand with additional components, and (d) are rugged and designed for 24/7 operation under harsh environmental conditions, say, high temperatures and humidity, (e) are replaced after many years of continuous operation, mainly because they are constantly connected and interact directly with physical equipment. Actually, from real-life observations, one may notice that ICS devices in general, even at higher levels often rely on deprecated and sometimes unpatched OS and applications.

C. ICS PROTOCOLS

The most widespread protocols used in ICS are Modbus, DNP3, IEC-104, IEC 61850, PROFIBUS/PROFINET, EtherNet/IP, OPC, WirelessHART and ZigBee. These protocols were specifically designed to deal with the complexity and the special requirements of the ICS. The operations inside ICS are real-time (deterministic), reliable, safety-critical, ruggedized, and sometimes remote. In the most typical cases, the use of serial buses is widespread, and the protocols that are used in levels 0-1 are referred to as Fieldbus protocols. However, nowadays, protocols that utilize directly Ethernet or TCP/IP stacks depending on the particular use-case, are rather common. Several of the traditional serial protocols, including Modbus and PROFIBUS, have a corresponding TCP/IP variant, in this case, Modbus TCP and PROFINET, while others, like EtherNet/IP and ZigBee, were designed to work directly over Ethernet and TCP/IP.

It is not the intention of this work to describe the operation of each of the above-mentioned protocols in depth [5], [4], but only focus on its security aspects. To this end, by referring to well-studied, real-life major incidents, Section V details on how these protocols can sometimes be exploited by attackers. Moreover, security shortcomings and vulnerabilities of the protocols as identified by the so-far published academic work are outlined in Section VI.

D. ICS SECURITY

Security practices in ICS can be either mandated by regulatory bodies, such as North American Electric Reliability Corporation (NERC) or recommended by entities such as the U.S. Cybersecurity and Infrastructure Security Agency (CISA) and the U.S. National Institute of Standards and Technology (NIST). Common key "best practices" are as follows [6]:

- Identify the critical assets that need to be protected.
- Separate the systems into logical and functional groups.
- Implement access control into and between each group.
- Monitor activities.
- Implement a defense-in-depth strategy.
- Limit the actions that can be executed within and between groups.

When first commissioned, many, if not all, of the ICS were kept isolated from other systems, forming a separate OT network. However, due to the rapid convergence of OT and IT, ICS are nowadays exposed to adversaries that aim either at financial gain, espionage, or sabotage through process disruption or physical destruction. In addition, many of the aforementioned devices and protocols used in ICS lack security features vis-à-vis their IT counterparts. They instead were created with a focus on assuring the timeliness and availability of the data used for monitoring and controlling critical industrial processes, rather than preserving security services like authentication, data confidentiality, and integrity. Often, there are certain misconceptions that peripheral network security measures, including firewalls and "air gaps" can protect from all sorts of cyber threats. Sections V, VI, and VII further stress on the fact that such security measures cannot be conceived as a "silver-bullet" defense for the ICS.

Furthermore, ICS security requires a deep knowledge of the system's specific operations. In some industries, typical operations are the combination and mix of chemicals, refining of oil, and the generation, transmission, and distribution of electricity or energy in general. These processes are usually automated and called *control loops* in industrial terminology. Consider, for example, a tank filled with liquid chemicals that get mixed. The level and composition of the liquid in the tank is indicated by sensors. When the combination of these chemicals reaches a specific density, the liquid is removed from the tank using pumps, and more chemicals are poured back into the tank. These are parameters that need to be well-understood given that: (a) the implemented security measures should not disrupt the process in any way, and (b) ICS and CI are an attractive target, especially for competent and well-equipped adversaries.

E. CRITICAL INFRASTRUCTURES

A CI can be defined as the physical and cyber systems and assets that are essential for the uninterrupted functioning of a nation's society and economy. According to the U.S. Department of Homeland Security (DHS), there are 16 CI sectors [7], namely chemical, energy, nuclear reactors, materials and waste, water and wastewater systems, healthcare and public health, transportation systems, financial services, critical manufacturing, dams, commercial facilities, communications, emergency services, defense industrial base, food and agriculture, government facilities, and information technology.

Besides the U.S., similar critical sectors have been identified by the European Union (EU) and individual countries around the globe. For acquiring more information on this matter, the interested reader can refer to [8], [9], [10], [11].

The threats against CI can be associated with either physical phenomena such as extreme weather, earthquakes, floods, and epidemics or pandemics, or human-related phenomena, including accidents, espionage, acts of terrorism, and cyberattacks. Therefore, the aspects of security and resiliency based on potential threats, are cardinal to the risk management process per CI sector. Since CI is complex, multi-layered, and involves a plethora of stakeholders, more attention is paid to the remediation of risks that are more probable and are also estimated to have a higher impact. This often results in cybersecurity-related measures being neglected since other measures, including physical security and protection from physical phenomena, are often considered of higher priority.

Furthermore, cooperation and communication via information sharing, namely cyberthreat intelligence, is essential, not only in an inter-CI fashion, but also across different CI sectors, as many of them are obviously interdependent. From a cybersecurity viewpoint, CISA [12], and the Information Sharing and Analysis Centers (ISACs) such as E-ISAC [13] are well-known organizations that promote and assist this effort.

III. RELATED WORK

ICS, CI, and their security issues and challenges have been investigated for several years. However, no particular focus has been given on the technical aspects and the root causes of the described incidents.

The rest of this section elaborates on the most pertinent surveys on this topic and discusses the key differences between them and the work at hand. Our study spans ten years, i.e., from 2012 to 2020, and the various works are presented chronologically, from the most current to the oldest. Nevertheless, for the sake of completion, and as summarized in Table 1, we do provide references to either significant but outdated work [14], or others devoted to more specific areas of ICS [1]. We choose the categories based on the information gathered from the related work and the information we provide in this work.

In [15] McLaughlin et al. survey the ICS cybersecurity landscape and discuss both offensive and defensive mechanisms for various levels of the ICS, including hardware, firmware, software, network and process. The authors focus on vulnerability assessment methodologies, ICS testbeds, attack vectors, say, payload construction and false data injection, vulnerability remedies and a number of secure control architectures. However, the paper does not offer a full-scale analysis of vulnerabilities and real-life incidents in ICS and CI.

The work of Xu et al. [16] reviews the vulnerabilities of common ICS protocols and elaborates on relevant attacks. On top of it, the authors detail on proposed countermeasures and current testbed implementations that can be used to perform both offensive and defensive research. Nevertheless, their work completely neglects wireless protocols exploited in the context of IIoT.

Hemsley and Fisher [17] present a study of publicized security incidents against various CI sectors and elaborate on the diverse types of adversaries. Similar to our work, they focus on the most significant incidents in an attempt to provide a complete view of the vulnerable components per type of CI. However, the incidents included in their work lack a detailed analysis. Furthermore, no discussion is made on the impact of vulnerabilities that affect specific ICS protocols and devices.

Volkova et al. [18] survey the several ICS communication protocols, namely Modbus, OPC-UA, TASE.2, DNP3, IEC 60870-5-101, IEC 60870-5-104, and IEC 61850. Some of these protocols, along with their vulnerabilities, are also discussed in this paper. The authors categorize the various protocols based on whether they cater for confidentiality, integrity, and availability. Potential security breaches in control system

TABLE 1. Related Work.

Contribution	Year	Incidents	ICS protocols	ICS Devices	Taxonomy included	Testbeds	ICS Security Framework	Novel Approaches
[14]	2012	●						
[1]	2015					●		
[15]	2016	●		●		●	●	●
[16]	2017		●			●		
[17]	2018	●						
[18]	2019		●		●		●	
[19]	2020							●
[20]	2020	●			●			
[21]	2020	●						
[22]	2020	●	●			●		●
This work	2021	●	●	●	●		●	

communication protocols based on the vulnerable protocols along with real-case scenarios and security recommendations are also put forward. Pliatsos et al. [22] discuss the security of Supervisory Control and Data Acquisition (SCADA) communication protocols. Additionally, they present a number of security incidents against ICS and CI along with relevant objectives and threats. They examine various proposals that aim at enhancing the security of SCADA systems in terms of attack detection. Moreover, they detail on the most common attack types against SCADA systems and offer an extensive presentation of SCADA security testbeds. On the downside, both the above-mentioned works do not provide a fully-fledged analysis of ICS security incidents from a technical viewpoint and lack of an analysis of wireless protocols used in ICS.

The work from Bhamare et al. [19] discusses the general state of play of cybersecurity in ICS. The key topics presented in this work are the integration of ICS with cloud-based environments and the use of machine learning techniques in aid of ICS cybersecurity. Moreover, a thorough categorization of approaches for ICS cybersecurity is offered. However, neither a detailed investigation of major real-life incidents nor the specific vulnerabilities pertaining to ICS equipment and protocols are presented.

Ahmadian et al. [20] perform a survey around cybersecurity incidents in ICS. They group these incidents into attack and non-attack related based on specific characteristics that govern them. They present information about the diverse attack sources, the entry points that may leave room for realizing such an incident, as well as its direct impact. The authors abstractly analyze some of the considered incidents, without however presenting adequate technical details that would allow the reader to grasp the precise nature of these events. The work from Alladi et al. [21] analyses on attacks against ICS and CI. Yet, the technical details provided are once more limited, and no discussion about ICS protocols and relevant devices is included.

Given the above discussion, the current work not only contributes an extensive and state-of-the-art analysis of the various major security incidents against ICS and CI, but also puts forward practical and theoretical security shortcomings that specifically affect diverse operational levels of the Purdue model. In this respect, the work at hand is not only full-featured, thus complementing the hitherto literature on the specific subject, but also is anticipated to provide a spherical view regarding the ICS and CI security state of affairs.

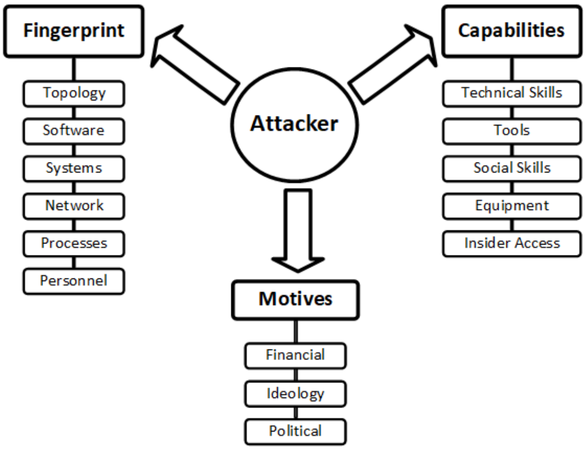


FIGURE 2. Taxonomy of characteristics of CI attackers based on three main criteria.

IV. ADVERSARIAL MODEL

Adversaries are individuals, groups, or organizations who attempt to compromise the security of CI, and possibly disrupt its operation. This section elaborates on their types, namely outsiders, insiders, (cyber)criminals, industrial espionage actors, terrorists, and nation-state actors, and discusses their characteristics and motives. Specifically, as shown in Figure 2, the segregation between the diverse types of adversaries is made based on three factors; their fingerprinting they perform against the target, their motives, and their capabilities.

Fingerprinting: An adversary that targets ICS and CI should possess adequate information to accomplish its goals. Deep knowledge of the physical location, the type and configuration of systems, and the running processes can provide them with a high advantage. In addition, information about the network topology and personnel can assist in the design of the attack.

Capabilities: The capabilities of an adversary can vary in terms of their skill set and the access they have to tools, vulnerabilities, and exploits. In addition, the use or deployment of human assets inside the target environment increases the chances of successful attack. If a precise replication of the targeted systems is possible, more accurate results can be achieved.

Motives: Adversaries have different motives when choosing an ICS or CI as the target. These threat actors may be driven by financial gain, political reasons, military objectives, or simply by their emotions. Based on these motives, the attacks may be carried out by nation-states, terrorists, competitors, and even ordinary cyber-criminals and hacktivists.

Based on all the above, the adversaries can be further categorized in:

- **Outsiders**: They are the most common adversaries in IT and OT environments. They exist outside the physical and network locations of the ICS environment. Depending on their resources and skills, they may possess prior knowledge of the ICS-specific assets.
- **Insiders**: Malicious insiders can cause harm to the systems by leveraging their access. Often, these are disgruntled employees with access both to the facility and the network. Mainly, they leverage their knowledge and level of access as tools to perform their actions.
- **Criminals / Hacktivists / Script kiddies**: These actors usually perform their actions for financial gain or hacktivism. They use common and sometimes widely available tools that are not drastically modified. Although their capabilities are limited in terms of resources and knowledge of the process that runs in ICS and CI, they can cause damage by exploiting commonly used systems.
- **Industrial espionage actors**: Industrial espionage has as a primary goal the exfiltration of information about the inner workings of ICS and CI. These actors have skills that allow them to acquire a great amount of information (such as screenshots, blueprints, application logic) and often collaborate with insiders, but at the same time, they wish to remain as stealthy as possible. Especially with the arrival of IIoT, the collected big data can provide crucial information to this type of adversary.
- **Cyber-terrorists**: This category also includes extremists, hacktivists, and other organized cyber-criminals. They target ICS and CI with the purpose of creating havoc and possibly spreading their ideology. They may be familiar with the physical premises of the targeted ICS and CI, and they persistently attempt to gain access to the network. They can acquire tools from resources that are not widely accessible to other actors. The use of the human element to deliver or initiate the exploits is another characteristic of their tactics.
- **Nation-state actors**: They are considered the most powerful, well-equipped, and skilled outsiders. Having, by definition extensive and sometimes unrestricted resources, they can target and damage a diverse set of CI. The attacks that originate from this type of actors can be performed as a means to test their capabilities, apply pressure to other nations or organizations for political reasons, polarize public opinion on controversial or other key matters, cause, and even harm to the administration and citizens. Their tactics are often performed under high secrecy with the ultimate goal to maintain a foothold in the targeted network. Their arsenal comprises a mix of legacy and specially crafted, highly sophisticated tools that might also include zero-day exploits. They have the capability to replicate the OT network, partially or in its entirety.

We should also mention that based on the analysis of Caltagirone et al. [23], often there is not only one actor, but rather an activity group that usually operates in a specific geographical area, verticals, or mission These adversarial groups can be combined or split based on their motivations and intent.

V. INDUSTRIAL CONTROL SYSTEMS AND CRITICAL INFRASTRUCTURE INCIDENTS

This section details some of the most prominent incidents that targeted ICS and/or CI. The description of each incident is conceptually split into six parts/axes, namely, infection, spreading, payload effects, command-and-control (C&C) (if any), variants, and key factors that enabled the attack. The chronologically arranged Figure 3 summarizes the relevant information. The selection of the specific incidents is based on the fact that the pieces of information collected were adequate to provide a complete view driven by the above axes.

A. STUXNET

Stuxnet [24] is considered the first malware specifically designed to inflict damage against equipment residing in an ICS. The malware is also supplemented with industrial espionage capabilities. Evidence indicates that the malware's primary target was the Natanz nuclear enrichment plant in Iran. The whole malware behavior is depicted in Figure 4. It should be noted that there is no report that clearly indicates all the devices that have existed in the targeted environment. Therefore, the separation on Purdue Levels is added only as supplement information for better comprehension of the incident.

The Stuxnet binaries consist of driver files that where digitally signed with compromised certificates. This was used as a means to avoid suspicion. From that point on, it attempts to spread to other workstations in the target network via multitude of alternative zero-day vulnerabilities ①, including (a) USB flash drives (CVE-2010-2568), (b) the Windows Print Spooler service (CVE-2010-2729), (c) network shares or the Server Service (CVE-2008-4250), (d) local privilege escalation (CVE-2010-2743), and (e) WinCC and PCS 7 SCADA system (CVE-2010-2772). Interestingly, it is programmed to only infect up to three victims, and then it erases itself from the infected media ②. Moreover, after

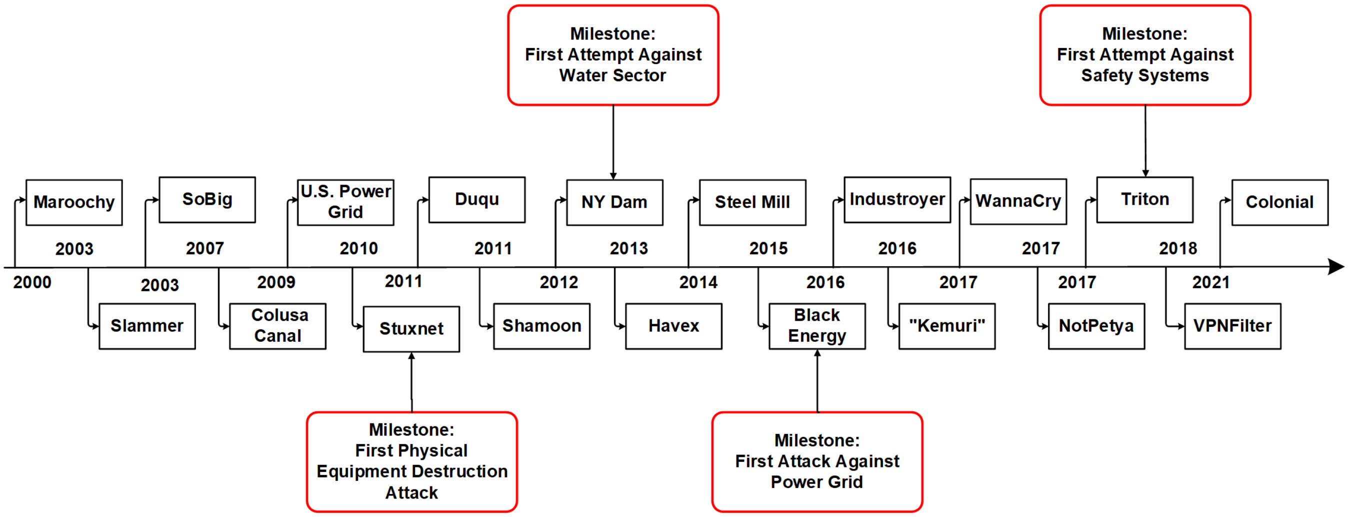


FIGURE 3. Timeline of the discussed incidents.

a specific, hardcoded date (in the discovered cases, June 24, 2012), it ceases any infection attempts.

The malicious code includes functionality that allows the attackers to control and update Stuxnet through a C&C server ③. However, since the "Field PGs" i.e., the specific type of EW made by Siemens, are expected to operate in an isolated network, the malware does not aim in conventional, direct ways of communication, e.g., the Internet. Rather, it aims to compromise and then use the external contractor companies as proxies. Commands and updates are first pushed to these naturally less secure networks, in hopes that they will eventually penetrate the siloed networks of the facility via conventional means, e.g., USB flash drives.

On a second stage, Stuxnet redirects its focus on spreading to "Field PGs". To this end, the malware infects the WinCC, a Siemens software designed to monitor and write data to the PLCs. The malware takes advantage of hardcoded credentials embedded in the software ④. As soon as an infected Field PGs connects to a Siemens S7-315 PLC for programming it, the malicious payload gets uploaded. The payload itself alters the control logic of the PLCs. The payload includes a rootkit destined to hide all the malicious actions performed ⑤. In the case of a successful PLC infection, Stuxnet monitors the PROFIBUS connections for 13 days [24]. Then, it alters the operational speeds of two frequency converter drives ⑥. Firstly, all functions related to the graceful shutdown of the system in case of a malfunction are disabled. In parallel, a sequence of actions that affect the centrifuges is performed within 27 days. Initially, the malware records the benign process events to infer the active operating frequency and then increases the rotating frequency to 1410Hz for 15 min. Then, normal operations are resumed for 27 more days. On the subsequent cycle, it forces the frequency to rapidly drop to 2Hz, followed by an extremely rapid increase to 1064Hz. The changes in rotating frequency creates damages to the inner walls of the containers [25]. It is believed that the malware aims to simply accelerate the degradation rate of the equipment, which in turn leads to higher operational costs.

An alternative, earlier version of Stuxnet (version 0.5) was found and analyzed in 2013 [26]. The main difference of this newer version is that it aims to control the centrifuge valves that are handled by an S7-417 PLC, instead of the frequency converter drives. Specifically, it monitors the pressure inside the centrifuge via the infected PLC, and as soon the pressure reaches a specific level, it closes the valves.

Stuxnet was designed taking into account the detailed information about the specifics of the target environment. Mainly, the malware relies on a multitude of zero-days to increase the probability of penetration to the target environment. The manipulation of the I/O process image is used to intercept the benign values and ensure that are not written to the process image output, to deceive the operators. This was a common design flaw in ICS and can be easily exploited, as indicated by Langner [27]. Finally, the compromise of the digital certificates indicates a powerful adversary with high determination, capabilities and resources behind this attack. From all the above, we conclude that Stuxnet affected the Levels 5, 4, 2, 1, and 0 of the adapted Purdue model shown in Figure 1.

B. DUQU

Duqu [28] is a malware discovered in Hungary by the Laboratory of Cryptography and System Security (CrySyS Lab). This malware shares behavioral similarities with Stuxnet. For instance, it hinges on compromised digital certificates that are used to sign device drivers and exploits zero days as part of its offensive repertoire. Yet, unlike Stuxnet, Duqu's main purpose lies only in cyber-espionage, i.e., the leakage of valuable information from ICS and CI. There is no publicly available information regarding the organizations that were impacted, although the malware samples analyzed by Symantec were obtained from ICS entities [29].

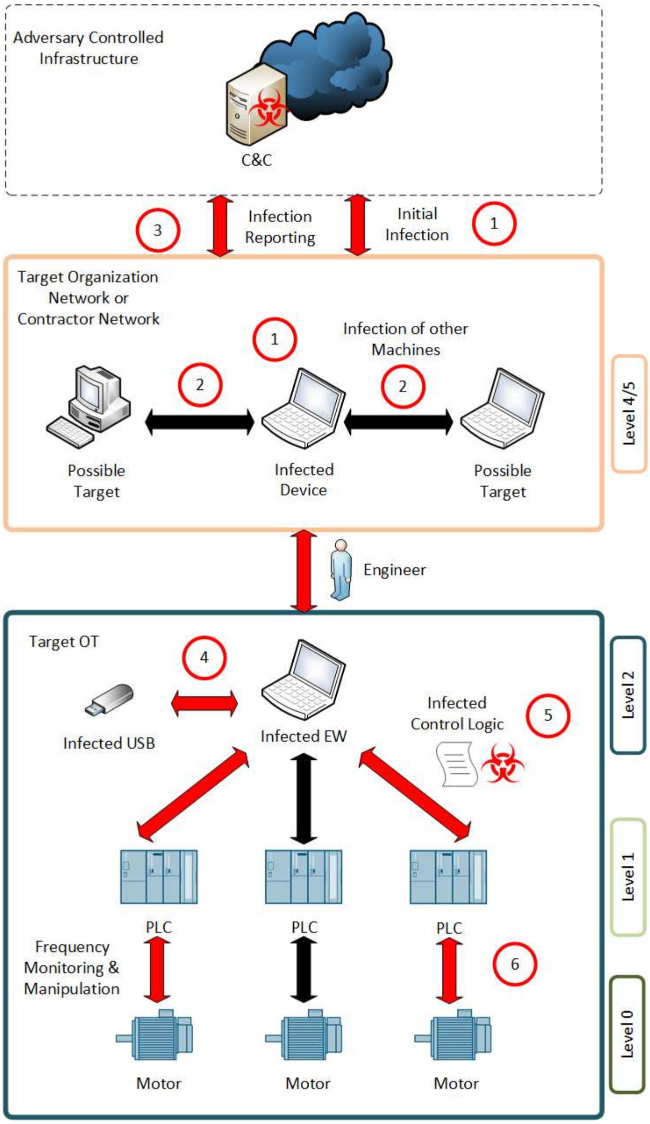


FIGURE 4. Stuxnet attack against the Natanz nuclear facility.

Existing studies indicate that a zero-day vulnerability in MS Windows TrueType font (CVE-2011-3402) is responsible for the initial infection of the target machine. This vulnerability enables remote code execution (RCE), allowing the attacker to perform arbitrary commands through the Internet. Presumably, the malware reaches the target host and exploits that vulnerability via MS Word documents. Then a driver is used to inject the malicious payload at system boot. This driver can be either singed with a compromised digital certificate or be unsigned. The actual payload is encrypted, and its decryption happens after the driver's initialization, only when the malware has verified that the Safe and Debug modes on Windows OS are disabled. The payload is a *.dll* file masqueraded as a *.pnf* file (the particular file type contains setup information and facilitates the installation process of programs in MS Windows), that is loaded into the *services.exe* system process.

After the original infection, Duqu may either (a) download a keylogger from the C&C server and use it to steal the administrator's credentials of critical servers in the targeted network, or (b) copy itself in corporate shared storage folders. The malware spawns a Remote Procedure Call (RPC) component to communicate with an external malicious C&C server from within the compromised network. This allows transfer of stolen information, and the receival of commands or even the download of upgrades that extend its capabilities. To remain stealthy, it employs a number of external compromised servers as communication proxies. For hosts unable to directly communicate with the C&C, i.e., those deprived of Internet access, Duqu is equipped with P2P communication capabilities.

After the infection phase, as a first step, the malware attempts to bypass antivirus programs installed in the system, if any. As a second step, the payload gets downloaded from the C&C server. It contains two *.dll* files aiming to steal data, namely, by recording keystrokes, taking screenshots, enumerating files from all drives, and storing them into temporary locations in the system after compressing and encrypting them. Finally, the captured information is later transmitted to the C&C.

In 2015 Kaspersky detected a variant that they coined as Duqu 2.0 [30]. This newer version is also a cyber-espionage tool that is very modular and aims to the extensive collection of system and user information. Moreover, it is believed that Duqu 2.0 targeted the UN Security Council's five permanent members (P5+1) events regarding the negotiations of Iran's nuclear program, cybersecurity, and telecommunications companies.

Duqu was able to perform all its actions due to the use of a zero-day vulnerability and social engineering techniques that deceit the employees to download the malicious attachment and initiate the assault. In addition, similarly to Stuxnet, the adversaries demonstrate access to considerable resources and high determination, e.g., the capacity to compromise the digital certificates. We can conclude that the malware affected directly only the Level 4 of the Purdue model.

C. SHAMOON

Shamoon [31] is a malware that aims in rendering the computers inside target organizations unusable by wiping their hard drives. Among the well-known victims of the malware are the Saudi Arabian oil companies Saudi Aramco and RasGas in 2012. In the case of the former, it is believed that 30K to 55K hosts inside the company's business network were affected, resulting in downtime that span 10 days. Although the ICS network was not directly affected, this incident is an example of how the demise of the IT network may cause an indirect disturbance in OT activities.

Originally, it was believed that the infection took place using a phishing attack. However, further investigation indicates that potentially an insider might have been involved [32], [33]. As soon as the malware is installed, its dropper component disables the User Account Control (UAC) in the

Windows registry. Then, it creates either a persistent service with name “ntssrv” or a scheduled task that executes the payload at a specific time [34]. Subsequently, as detailed in the following, two more components are unpacked, the “reporter” and the “wiper”.

The malware spreads to other hosts in the network via network shares [35], installs the wiper module and waits for instructions from the C&C server [36].

The wiper module, that is responsible for the files deletion, may attach itself to a standard MS Windows process in an attempt to better masquerade itself [36]. The wiper includes a signed driver extracted from a third-party disk utility developed by Eldos, namely the Rawdisk [37]. Normally, the installation of disk drivers as well as raw disk access requires administrator privileges. Yet, by first disabling the UAC, and by using special access features provided by the utility, modification to the disk can be performed even in user-mode. Once the wiper module executes, it enumerates all files and appends their names into *.inf* files. The enumerated files are then filled with fragments of a *.jpeg* image. The disk overwriting is performed recursively, which may corrupt the Master Boot Record (MBR). Another interesting point is that additional actions, namely, encryption of files, are also supported [34].

The last standard module of the malware, namely the reporter, forwards details regarding the number of deleted files per targeted host back to the C&C server. Interestingly, this module also includes modes for receiving new executable files from the C&C. However, due to coding errors, this module was not functional.

Newer versions of Shamoon were observed in 2016 [38] and in 2018 [39]. Interestingly, the 2018 version also includes an updated wiper component, to perform a deep erase of the disks, rendering them non-recoverable even with the use of forensics techniques.

The malware is able to perform its actions due to the interconnectivity of all computers in the business network, stolen credentials, and the use of a legitimate driver. Since the exchanged data between the IT and OT are used to determine the business’s needs and procedures, such catastrophic attacks to the IT network may deprive the ICS of the high-level site operations that support the production process in the OT. Shamoon affected primarily the operations of the Level 4 of the Purdue model.

D. HAVEX

Havex [40] (also known as Backdoor.Oldrea) is a backdoor malware used by the Dragonfly group to perform espionage against CI mainly in Europe and the U.S. Well-known targets involve companies in power and pharmaceutical sectors [41]. After Stuxnet, Havex is the first malware designed to impact critical infrastructures by targeting ICS communication protocols.

The malware infects the target systems using a triad of tactics [42]: (a) phishing campaigns, (b) “watering hole” attacks, and (c) compromised vendors. The first infection method is based on delivering the malware through malicious PDF documents, dropped to the victim via e-mail attachments. In the case of “watering hole” attacks, the evil-doers first compromise websites frequently visited by the victim company. Relevant discoveries point to the use of, an *iframe* to force the automatic download of the *LightsOut* exploit kit. The third tactic replaces legitimate software distributed by the websites of third-party vendors with a Trojan as a form of supply chain attack. Examples of such software are VPN clients or PLC drivers [41].

After infecting the victim’s system, the malware modifies the *Temp* and *System* folders of the Windows OS along with the system’s registry. Havex tries to collect information about the infected system, including available drives, generic files, e-mail addresses, and ICS configuration files.

The most distinguishable feature of Havex is its ability to discover networked devices connected to typical PLC-related ports, namely, TCP ports 44818 (Rockwell), 102 (Siemens) and 502 (Schneider Electric). This process is facilitated by Distributed Component Object Model-based (DCOM) OPC technology that is normally used to interconnect equipment from different vendors, along with the MS Windows networking (WNet) service, which is used to expose networking functions to Windows applications (see Section VI).

As a first step, the malware collects information about each OPC server’s version, vendor information and bandwidth [40] [43]. Next, the Havex payload enumerates the OPC tags provided by each server. An OPC tag is a structure that contains information about the data transmitted by an OPC server to any component in the OT network (e.g., a PLC) in a self-explanatory and human readable format. Therefore, observing these tags may give the attackers knowledge about the physical processes. The malware also has the ability to make a distinction between real OPC tags and other ones that are provided by honeypots [44]. Both the data collected from the IT and OT environments are sent to the C&C servers, using a custom encryption scheme to protect this transmission [44].

The malware was able to infect the facilities based on the fact that the spam campaigns were successful. In other words, the main diode of infection was once again the IT network. In addition, there was a successful compromise of websites of interest and the replacement of legitimate software from the vendors with malicious ones. The OPC scanning module achieved its goals due to the improper isolation of the IT and OT networks in the targeted facilities as well as the open nature of OPC. All in all, in the case of Havex, the affected operations were those that existed in the Purdue Levels 5, 4, and 3.

E. BLACKENERGY/2015 UKRAINE POWERGRID CYBERATTACK

BlackEnergy (BE) [45] is predominantly a botnet/DDoS tool, which through the years evolved into a malware suite with additional sophisticated features. Particularly, the third iteration of the malware (BE3) is of special interest as it was responsible for an illustrious attack campaign against Ukrainian power distribution companies in 2015. In these major incidents, BE3 was used only during the early stages of the attack to deliver the payload to the targeted networks and grant remote access to the ICS to the perpetrators. As a result, power outages started occurring, which affected approximately a population of 225K in the regions of Ivano-Frankivsk, Chernivtsi, and Kiev. The process that inflicted damage to the systems can be found in Figure 5. Similar to Figure 4, the separation on levels is created abstractly for the reader to better understand the attackers' approach against the target utilities.

BE is created with a very modular architecture in mind [46]. The BE3 version, can be delivered via the use of MS Word documents embedded with malicious macros [47]. If macros execution is enabled in MS Word, a malicious VBA script attaches the payload to the startup folder of the system ①. Then, the payload initiates a connection with the C&C server ②. It is worth noting that all communication is done over HTTP and is encrypted via RC4 [48].

Prior to the attacks that caused the power outages, a thorough reconnaissance stage took place using BE3, which is believed that it may have lasted up to six months [49]. During this period, the attackers gathered all the necessary information regarding both the IT and OT environments. To do so, several external tools specifically designed for credentials theft, network discovery and scan, remote access, screen capturing, and key logging were used ③ [50]. The captured credentials provided the attackers with access to the ICS network via VPNs ⑤. Moreover, devices, which under normal conditions cater for power supply redundancy to communication and data servers (UPS), were also discovered and were re-configured so the attackers could disconnect them at will. During the reconnaissance phase, the attackers also installed the *KillDisk* component in a network share ④. When executed at a later stage, this component overwrote the MBR of of IT PCs, and deleted logs and system events, making any subsequent investigation of the attack much harder [51].

The last stage of the attack took place on Dec. 23 2015. The adversaries exploited two different approaches to wreak havoc. In the first approach, Remote Access Tools (RATs) were used by the attacker to connect to the HMIs ⑥. Additionally, the operators were locked out of their workstations, unable to perform any actions. The second approach was more stealthy, as the attackers issued commands directly to the Distribution Management System (DMS) server using the VPN connections ⑦, ⑧. As a result, the attackers were able to access the HMIs, to open the circuit breakers, and to cause power outages to at least 57 substations ⑨.

After causing the outage, the adversaries proceeded to additional actions to amplify the inflicted damage: (a) pushed a malicious firmware update to corrupt the Moxa and IRZ Serial-to-Ethernet adapters [52]; in this way, they effectively reduced all monitoring and control capabilities of the operators, (b) the installed KillDisk was executed and wiped the operators' PCs ⑩ but also, due to poor network configuration, affected the HMIs connected to Remote Terminal Units (RTUs) [45], (c) disabled the UPS from the communications server to cause further confusion to the operators ⑪, and (d) to make matters worse, a DoS was performed against the telephone center.

To restore power, all operations were switched to manual mode [53]. The restoration process required approximately six hours.

As described in detail in [54], the BE malware was also used against numerous CI targets in a campaign that took place one and a half year before the described incident. Variants of BE have also been identified in alternative campaigns against U.S. CI sectors [55].

The malware succeeded in its goals due to the lack of security awareness on the operators' side, the detailed acquisition information about the equipment used in the facility, the lack of two-factor authentication for the VPN services, the improper configuration of the firewalls, and the deficiency of security mechanisms in the Serial-to-Ethernet adapter devices. The Purdue Levels from 4 to 2 where impacted directly in this attack.

F. INDUSTROYER/CRASHOVERRIDE/2016 UKRAINE POWERGRID CYBERATTACK

Industroyer [56] (or CrashOverride) is a malware that targeted the Ukrainian power grid on the Dec. 2016 attacks. This assault comes just a year after the BE3 attack (see subsection V-E) but it is much more sophisticated in comparison. Similarly to BE3, the malware follows a highly modular design that allows it to directly access ICS equipment, however this time at the transmission substations. During the attacks, it caused power outages that lasted almost one hour, affecting one-fifth of the Kiev region.

A report from Dragos [57] indicates that the intrusion took place during a phishing campaign that occurred a few weeks after the successful 2015 attack. Once the malware is installed in the victim's PC it starts to scan for legitimate credentials of remote access (or VPN) tools that may provide a direct connection to the ICS networks. The adversaries created users with administrator privileges in the access server so they could subsequently access a database server, namely the *Data Historian* [57]. A historian concentrates all the data from the ICS environment to provide information to the business network. By default, Data Historians should support unidirectional data flow only from the ICS to the IT network. A misconfiguration that allowed bidirectional data flow was exploited by the attackers to gain a foothold to the ICS network.

The attackers leveraged the tool Mimikatz [58] as a way to capture and reuse credentials inside the ICS environment. Subsequently, they accessed multiple hosts and attempted to create a link between servers. *Visual Basic* and *BAT* scripts were used to move masqueraded *.exe* files as *.txt* files and

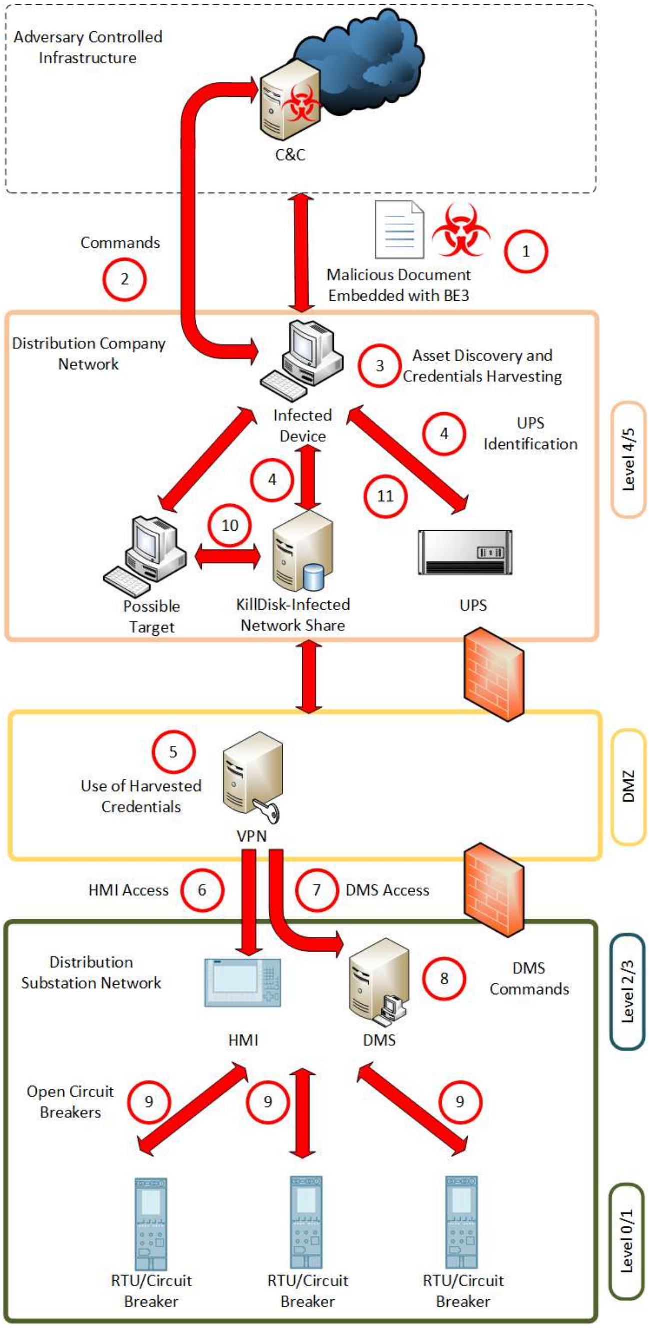

Adversary Controlled Infrastructure
C&C
Commands
2
1
Malicious Document Embedded with BE3
Distribution Company Network
3
Asset Discovery and Credentials Harvesting
Infected Device
4
UPS Identification
Level 4/5
4
11
10
Possible Target
KillDisk-Infected Network Share
UPS
5
Use of Harvested Credentials
VPN
DMZ
HMI Access
6
7
DMS Access
Distribution Substation Network
8
DMS Commands
Level 2/3
HMI
DMS
Open Circuit Breakers
9
9
9
RTU/Circuit Breaker
RTU/Circuit Breaker
RTU/Circuit Breaker
Level 0/1

FIGURE 5. The 2015 cyberattack against the Ukrainian power grid.

execute PowerShell commands.

The malware then tries to install and start itself as a system service in order to execute the payload components. The components included fall into the following categories [56]: Backdoor (one primary and one alternative), launcher, wiper, port scanner, ICS protocol-specific malicious payload modules (IEC-101, 104, 61850, OPC-DA), and DoS module for Siemens SIPROTEC protective relays.

The primary backdoor communicates with C&C servers via the Tor anonymity network and activates only in a specific hour of a day. The alternative backdoor is a "trojanized" MS Windows Notepad application that once executed it can run a shellcode downloaded from the C&C server; However, as described in [57] these backdoors are not vital, and only play an auxiliary role to the attack. The launcher module was configured to be triggered at specific dates. The data wiper module is executed as the final stage of the attack. It alters the Registry Keys by making them point to an empty string path and rewrites the standard filepath that is used by every ABB software.

The IEC-101 malicious component implements serial communication according to the IEC 60870-5 standard. This component controls COM ports that communicate with RTUs, which are connected to physical circuit breakers to modify their status from closed to open. The IEC-104 component is similar to the IEC-101, but utilizes TCP/IP communication (see Section VI). The IEC-61850 component probes and enumerates devices that use the protocols under this specific standard. If such a device exists, it requests data using the Manufacturing Message Specification (MMS) protocol and searches if there are any matching tags to these messages such as *CSW*, that will indicate the presence of switches and circuit breakers [56], [59]. The port scanning tool is custom-made, probably to evade detection.

The Industroyer's OPC-DA payload scans and lists all the OPC servers that are provided by ABB software, and also attempts to change the state of devices connected to these OPC servers. A similar behavior has also been observed by Havex (see Section V-D).

The DoS tool leverages a vulnerability found in the Siemens SIPROTEC protective relays (CVE-2015-5374), that allows an attacker to send hand-crafted packets to the device in port 50,000 rendering it unresponsive. It should be noted that according to Slowik [57], the authors of this component made a mistake in the byte conversion of IP addresses, and since these IPs were hardcoded, this component of the attack did not execute. If this component was properly implemented, the disruption event could have been transformed into a physical destruction attack [60].

Industroyer was successful because of the attackers' knowledge of the grid operations and network communications, the infection via spear-phishing campaigns, the fundamental lack of security mechanisms for the ICS protocols, and the exploitation of a vulnerability of the Siemens SIPROTEC devices. It is observed that this malware, has managed to affect a multitude of Purdue Levels, namely levels 4, 3, 2, 1 and 0.

G. TRITON/TRISIS/HATMAN

Triton [61] (also known as Trisis or HatMan) malware is created to interact with Triconex SIS controllers (made by Schneider Electric) and more specifically the Triconex 3008 processor. Such controllers independently monitor the status of the controlled processes. The malware's intention is to disrupt the safety mechanisms of the controllers in the target facility. However, as FireEye later discovered, due to incomplete implementation the malware unintentionally triggered the forceful shutdown of the controllers. Figure 6 demonstrates the actions taken by Triton. Once again, the partition of the environment into levels, acts supplementary to the understanding of the incidents, as there is no clear indication of all the possible devices that might have existed in the targeted facility.

Evidence indicate that the attackers may have gained access to the OT network almost a year before the actual incident ①. Due to a misconfigured firewall [62] ②, the attackers got foothold to a SIS EW and through it, they delivered the payload to the target controller using a custom-made TRITON attack framework.

The two main components of the Triton module that infected the SIS Engineering Workstation ③ are: (a) an executable, namely, *trilog.exe*, which aims to deliver the payload, and (b) a *library.zip* file that contains all the libraries required to communicate with the Triconex SIS controllers. The trilog.exe was developed in Python, but was compiled using Py2EXE to be able to execute in the SIS EW where a Python environment is not usually installed. To establish communication with the SIS controller, the TriStation protocol had to be reverse-engineered by the attackers [63].

The authors of the malware were counting on that eventually the physical four-position key switch of the SIS controller would be set on PROGRAM mode by the engineers [64]. In this mode, where changes are allowed to be performed to the controller, the trilog.exe was able to deliver the file *inject.bin* to the controller ④.

The inject.bin exploits a zero-day vulnerability (CVE-2018-7522), to elevate its privileges, add another file and restore expected permissions. When finished, a dummy program (initiated by trilog.exe), overwrites the part of the memory segment on the controller that stores the inject.bin. In practice, part of the malicious OT payload namely, the *imain.bin* is uploaded either to firmware or application area of the controller's memory region by inject.bin. This provides an attacker with full access of read/write/execute functionality to the controller irrespective of the Triconex key switch position [64], [65].

Furthermore, four modules inside the *library.zip* are used to deliver inject.bin and imain.bin to the SIS controller, via the reverse-engineered TriStation protocol. The module *TsHi* exports functions used for input and code signing, while the *TsBase* translates those functions into specific codes and formats the data. The underlying UDP protocol is implemented by *TsLow*, where the appropriate function code is chosen, and the serialization and send of the payload to the controller is performed. The last module, *TS_cnames.py*, contains all the function and response codes, as well as the key switch and control program states.

The code for the Triton malware was leaked and can be found in a GitHub repository [66].

Technically, the possible malicious outcomes of Triton's capabilities may be: (a) shutdown of the process through operational uncertainty, (b) forcing the SIS controller to an unsafe state by maliciously altering the SIS logic, (c) removing all the fail-safes that exist to prevent damage, thus creating an unsafe physical condition [67]. In the studied real-life example, only shutting down of the controllers was observed possibly due to attack implementation errors.

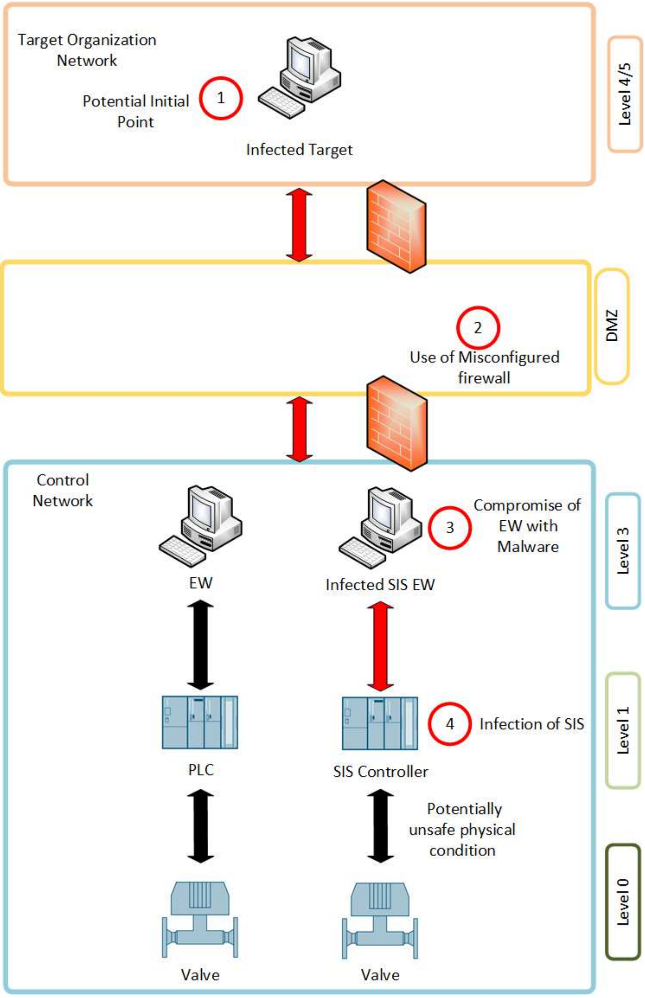


FIGURE 6. The Triton attack against a petrochemical facility.

The overlook of alarms from the anti-malware system, a misconfiguration of a firewall, the hardware key set to PROGRAM mode [62], and the relevant zero-day vulnerability, made it possible for the attacker to gain access to the EW and the SIS controller. In this case, the devices at Purdue Levels 2 and 1 were the ones exploited by the Triton malware.

H. VPNFILTER

VPNFilter is a modular malware that incorporates both reconnaissance and destructive features. Its scanning and infection activity was first observed in May 2018 by the Cisco's Talos Intelligence Group [68]. According to their analysis, the scans performed by the malware targeted primarily routers and Network Attached Storage (NAS) in more than 100 countries. The binary analysis performed by Talos shows that the MIPS and x86 are the targeted architectures.

It is also estimated that the malware infected more than 500K devices worldwide. While VPNFilter contains ICS monitoring capabilities, it may also affect other types of environments.

During the infection stage, VPNFilter installs binaries that attempt to connect to a C&C server to target devices that run the well-known Linux BusyBox. The infection method is very resilient, as it adds the binary to the Linux task scheduler configuration service *crontab* to persist across device reboots.

As a first step, the installed malware attempts to retrieve images from Photobucket and *toknowall.com*. These contain the active URLs of the C&C server in their meta-data portion. This is done as a way of obfuscation. If that practice fails, the malware tries to directly establish a connection to a hardcoded, public IP address. Interestingly, the RC4 implementation that is used to encrypt the communications contains a similar bug to the one observed in the BlackEnergy malware (see Section V-E).

The most notable instruction that can be received from the C&C is the "kill" function that can be triggered via the *dstr* module, which can wipe the device's storage [68].

Two additional modules of VPNFilter include a packet sniffer (*ps*) and a Tor network plugin. The former is used to extract website credentials and log Modbus TCP/IP packets. The Tor plugin is used to communicate with the C&C anonymously. Frequently, this communication involves the downloading of new modules that extend the malware with capabilities such as data exfiltration and device management. Newly discovered modules such as *ssler*, are able to intercept and manipulate all traffic from port 80 [69], [70]. Another module namely *tcpvpn* can establish VPN connections on compromised devices, thus enabling the adversaries to access the internal networks of the infected devices.

Common countermeasures are the hard reset, applying patches, and change of the default login credentials of the devices [71], [72].

VPNFilter was able to achieve its goals by infecting some of the most critical components of a network, i.e., routers and NAS servers, that many applications use by exploiting unpatched vulnerabilities and default credentials. Moreover, the use of some of this equipment is intended to be used in SOHO applications and not ICS installations. This poor choice of insufficiently protected equipment provided leverage to the attackers, as they managed to mainly penetrate the levels 4 and 3 of the Purdue model.

I. WANNACRY

WannaCry [73], is a cryptoworm-based attack that affects MS Windows computers. The worm encrypts files in the OS and demands Bitcoin as ransom. Some mission-critical organizations that were affected by WannaCry in 2017 include the National Health Service (NHS) of the United Kingdom, the Spanish telecommunications company Telefonica, and the U.S. delivery service, FedEx. In the case of NHS, it is estimated that the cost from the WannaCry damages is £19 million. [74].

The infection stage uses an existing exploit known as *EternalBlue* that can achieve Remote Code Execution (RCE). In turn, the exploit is based on a vulnerability of the SMBv1 protocol (CVE-2017-0145), which is specific to MS Windows 7, Windows Server 2008, and earlier versions. It should be pointed out that Microsoft addressed the vulnerability even before the original WannaCry-based attack, in security bulletin MS17-010. Nevertheless, the targeted systems remained vulnerable due to negligence in installing the particular patch.

Once the vulnerable computer is infected, another tool namely, DoublePulsar is used to deliver the ransomware part. The malware spawns services and tries to connect to a specific domain [73]. In parallel, it runs a persistent service that scans both the internal and external networks. The two threads of the service, check if computers have the SMB port 445 open. The analysis of Malwarebytes Labs [75] indicates that the "wormable" part of WannaCry maybe still be effective on computers behind a NAT or proxy and not just Internet-facing computers.

After the end of the scans, the malware creates several files, such as images, and README files that are used to display messages in various languages. It also searches for files with specific extensions in all disk drives, including networks shares and removable drives [76]. Then, it uses a combination of RSA and AES algorithms to encrypt the files and changes their extension to *.WNCRY*. Some variants of the malware also delete every shadow copy volume that exists in the system. An additional process called *@Wanadecryptor@.exe* displays the ransom message on the screen and alters the wallpaper.

The malware also includes a hardcoded unregistered domain that is checked only during the primary stages of the infection. This domain acts either as a "kill-switch", or as an anti-sandbox technique that evades rudimentary malware detection and dissection procedures. In the original incident, Marcus Hutchins, a security researcher, identified this domain and proceeded to register that domain himself in an effort to better study the malware [77]. Although there was no impact on already affected systems, it made it possible to stop the spreading of the malware. In subsequent versions of the malware, two more domains were included, but both also became registered quickly by security researchers [78], [79].

WannaCry has a modular architecture that allows it to possibly drop and execute different payloads to its targets. Furthermore, its network traffic is encrypted through a custom Transport Layer Security (TLS)-like protocol. Interestingly, a similar technique was used in the attacks against Sony Pictures in 2014 [80].

As we observe, the malware achieved its goals by using a disclosed vulnerability and the negligence of applying updates that can prevent the spread of the malware. The operations that rely on devices of the Level 4 of the Purdue model where severely impacted.

J. NOTPETYA

NotPetya [81] is a cryptoworm attack against MS Windows based-hosts. It started spreading in Ukraine one month after the WannaCry attack. Among the victims of this malware are numerous Ukrainian ministries, banks and metro systems, the Heritage Valley Health System, and the logistics-shipping company Maersk. In the case of Maersk, the estimated loss in revenue from the damages was over $200 million [82]. The steps taken by NotPetya can be seen in Figure 7. The indication of the Purdue Levels is supportive to the description of the whole incident.

The main delivery mechanism in Ukraine was the tax application system, M.E.Doc. More specifically, a deficiency to the patch update policies of the company, allowed the attackers to compromise the particular servers ① [83]. According to Cisco's Talos Intelligence Group [84], the attackers identified the SSH credentials of administrator accounts and injected a backdoor into the M.E.Doc's software update mechanism. The backdoor can establish a connection with a proxy, and from there, it enables the downloading of malware or the uploading of information extracted by the victim.

Once the targeted host systems update M.E.Doc, the malware is also delivered ②. As a next step, NotPetya, drops the files for the ransomware message, the *.dll* file that contains the ransomware, the masqueraded version of the *PsExec* utility (a telnet-replacement for remote execution of processes) along with the tool Mimikatz [58] in order to perform credentials harvesting. Then, the malware decides its next steps based on the antivirus present on the infected system [85], if any. If a Kaspersky antivirus is present, the module will not proceed to encrypting any files on the victim. If one of the Norton or Symantec antiviruses are installed, the included EternalBlue exploit will not be used to spread the malware to other hosts. Moreover, it checks its execution privileges to decide whether it is going to use the credentials theft module.

The malware employs numerous alternative ways for its proliferation ③: (a) network enumeration to discover any DHCP services that will allow it to scan for the SMB ports 445 and 139 [86], (b) through the SMB copy and execution, leveraging the stolen credentials, (c) via the EternalBlue or EternalRomance exploits with the purpose of launching a shellcode and injecting the malware to the target. Targets were also accessed via the NTLM protocol that is typically used for authentication against Active Directory ④ [87].

After that, NotPetya triggers its encryption capabilities. Precisely, it reads the MBR and installs a custom bootloader in its first sector, adds the Bitcoin wallet address for the ransom, and reboots the machine. Once the machine reboots, the malware encrypts the MTF as well as all the files in the computer using a combination of RSA and AES encryption algorithms.

Moreover, the malware proceeds into several anti-forensics actions [86], [88]. Once it executes, it deletes itself and its associated tools and modules from the disk, thus running only in memory. It then rewrites that part of the disk with zeros. Finally, it deletes all security, setup, system, and application logs.

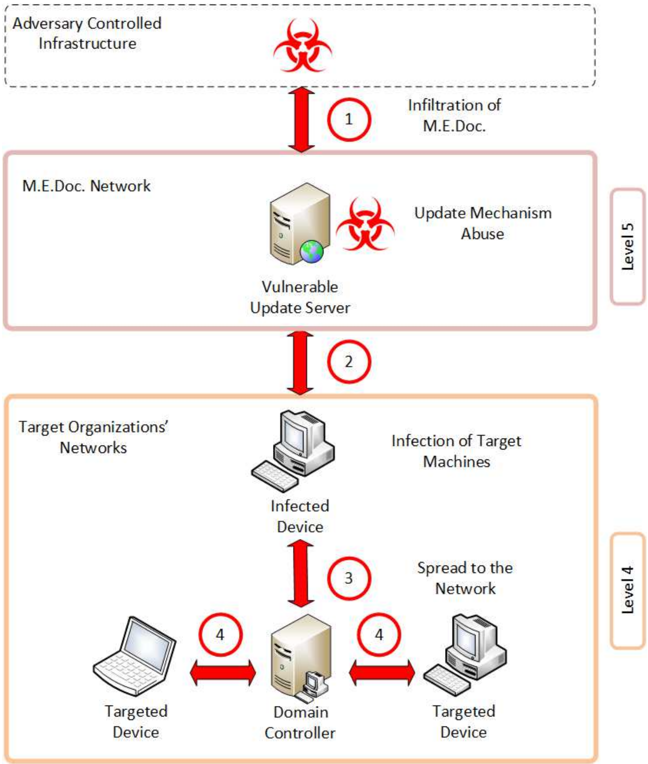


FIGURE 7. The NotPetya attack process.

NotPetya was able to spread due to the infiltration of the M.E.Doc's update system, the credentials harvesting, and the use of unpatched and outdated Windows machines. To achieve its goals, it exploited the Purdue Levels 5 and 4.

K. COLONIAL PIPELINE

On May 7, 2021, Colonial Pipeline was hit by a cyberattack that forced the company to proactively shutdown its OT network and stop all of its IT processes [89]. Colonial Pipeline, provides almost 45% of the U.S. East Coast's fuel. The DarkSide ransomware targeted Colonial's corporate IT networks, and it required the company to pay ransom in order to provide them with the decryption keys that would restore their systems. The outage lasted almost six days until the company managed to slowly restart its operations [90].

According to Mandiant/FireEye [91], the initial entry point was a VPN account that was not believed to be still active. The password of this account may have been used on another website that was compromised beforehand. It has been noted that the password was complex in terms of length and special characters. However, no information exists on how the VPN username was obtained. Notably, the particular VPN account did not have any multi-factor authentication protection enabled.

Once the DarkSide ransomware gains foothold to the network, it tries to move laterally by using network shares to install itself to other connected MS Windows machines. The ransomware is capable of encrypting all the drives that are reachable from the infected machines, but it does not include any self-propagating mechanisms [92].

Once dropped to the target machine, it checks a list with files, directory paths, and file extensions that will be skipped during the encryption process. It then checks if it is running under administrator privileges. If that is not true, it attempts to bypass the MS Windows UAC. Before encrypting the files, the DarkSide ransomware checks the language of the OS and skips the encryption if this language is included in a hardcoded list. It also has the capability to send information about the files in the infected machine back to the C&C server.

After dropping the ransom note and changing the machine's wallpaper, the malware encrypts all files using a combination of Salsa20 and RSA-1024 encryption [92]. As the encryption takes place, it sends progress updates to its C&C. Finally, it executes an encoded PowerShell command that deletes all the volume shadow copies in the targeted machine. If PowerShell is unavailable, the ransomware accesses the Windows Management Instrumentation (WMI) and performs this process manually.

To prevent detection from anti-malware software, the ransomware performs dynamic resolution of MS Windows API calls using hashed and encrypted names [93].

The use of compromised credentials enabled the attackers to penetrate the company's network. The malware follows double-extortion techniques where the threat actors first exfiltrate information from the victims' systems and then launch the encryption routine. The aggressors seemed to have a money-orientated approach as they have targeted IT systems (Purdue Levels 5 and 4) and not OT systems [89]. However, with no IT systems, Colonial could not perform the business operations needed to drive its pipeline storage and refining tasks.

L. OTHER INCIDENTS

1) German Steel Mill

The attack against a German steel facility in 2014 has been reported by Bundesamt für Sicherheit in der Informationstechnik (BSI) in their annual report [94]. The adversaries managed to gain access to the OT network of the steel plant and cause severe damage to furnace equipment. Due to the subtlety of the issue, BSI never disclosed technical details or specifics regarding the attack.

According to the BSI report, the attackers used spear-phishing and social engineering tactics to establish access to the business network. From there, they assumed access to the OT network and were able to connect to individual control systems. Then, they changed the logic of components that prevent systems failures, e.g., the non-controlled shutdown of the furnace. The attackers demonstrated familiarity not only with the systems inside IT and OT environments but also with the steel production process. According to Lee et al. [95], the components that were possibly impacted by this attack were, PLCs, HMIs, SIS controllers, and alarm systems. The analysts also believe that the attackers' goal was to cause intentional damage directly to the steel production process.

2) Maroochy Water Services

The 2000 Maroochy Water Services incident [96] was a targeted attack from a former employee having special knowledge of the internal procedures that typically take place in the specific installation. Using special equipment, the attacker had the capability to issue remote commands to the system. The infiltration and actions against the water systems in the Maroochy area in Queensland, Australia, caused 800K liters of sewage to be emptied into local parks, rivers, and the grounds of the Hyatt Regency hotel.

According to Abrams and Weiss [96], the installed SCADA system consisted of 142 sewage pumping stations, each of which had two monitoring computers. The latter were equipped with PDS Compact 500 radio transmitters that were acting as RTUs/PLCs to receive instructions from the control center. Due to the wide area of the installation, several repeater stations were also deployed to assist the communication.

Due to the attack, the systems lost communication, and the pumps could not perform their normal operations, thus releasing sewage. The contractor company initiated an audit to investigate the root cause of the issue. Despite altering the identifier of a station, the operators noticed that the old identifier was still used in some of the remotely issued commands. Initially, these prevented the remote commands from being executed, but soon after, the perpetrator suspected that alteration and initiated a brute force to discover the new identifiers.

In subsequent incidents, the adversary disabled the alarms at four pumping stations. This time, the contractor company, in coordination with the police, suspected that an ex-employ could be behind the attack. Therefore, the attacker was physically located by the authorities and found in the possession of a laptop with a stolen software for SCADA reconfiguration installed, along with Motorola M120 two-way radio and PDS control devices. Evidence retrieved from the laptop also indicated that commands from the system program run at least 31 times, which matched the behavior observed in the company's logs.

At that time, the radio communications used in SCADA systems lack security features or had improper configuration. Furthermore, there were no security requirements from the contractor, the logging mechanisms were not tuned with a security mindset, and the incident response procedures were insufficient.

3) New York Dam

The intrusion of the Bowman Dam in Rye, New York occurred in 2013 [97]. The target was a small dam with insignificant reservoir volume to cause large-scale damage. Yet, the demonstrated technical capabilities of attackers are alarming.

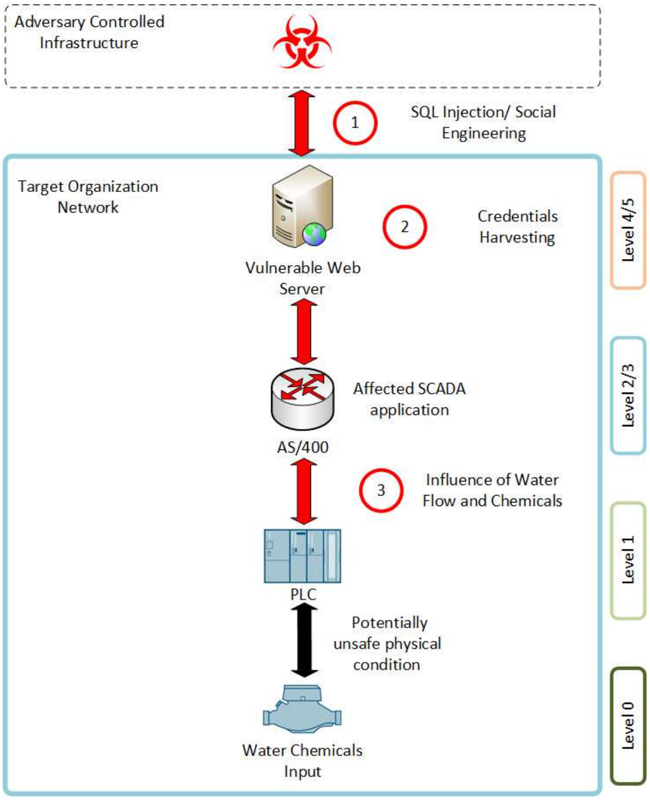

Adversary Controlled Infrastructure
1
SQL Injection/ Social Engineering
Target Organization Network
Vulnerable Web Server
2
Credentials Harvesting
Level 4/5
Affected SCADA application
AS/400
Level 2/3
3
Influence of Water Flow and Chemicals
PLC
Level 1
Potentially unsafe physical condition
Water Chemicals Input
Level 0

FIGURE 8. The attack against the "Kemuri" company.

According to the Department of Justice report, this assault is also linked to a larger attack campaign against various U.S. financial institutions, including Bank of America, JP Morgan Chase, and Wells Fargo [98].

The attackers managed to gain access to the system via a cellular modem [99]. Precisely, six remote access attempts took place between Aug. and Sept. 2013. Information about the water levels, temperature, and status of the sluice gate was also obtained. The sluice gate of the dam was not operational at the time of the attack according to city officials [100]. No additional technical details exist in the public domain.

4) "Kemuri" Water Company

In 2016 Verizon performed a security assessment for a water company (simply mentioned with the pseudonym "Kemuri") [101]. The assessment took place after the employees became suspicious of an intrusion due to irregular valve and duct behavior. The malicious actors managed to access the SCADA system and influence the PLCs that regulate the water flow as well as the chemicals blended in. Fortunately, an alert system that was already in place notified the operators in a timely manner, and more disastrous consequences were prevented. This incident is illustrated in Figure 8, although no clear information exists about the rest of the devices that might have been placed in the environment. Therefore, the incorporation of levels acts as supplementary information.

During the security audit, Verizon identified Internet-facing applications associated with critical operations. Moreover, the equipment that existed in the OT network was antiquated, thus unable to receive any updates. All the network connections from customers' applications, i.e., a payment portal, and PLCs were going into a single router, namely, an obsolete IBM AS/400 produced back in 1988. In addition, the AS/400 was managed by a single employee that possessed the required knowledge of the system, thus creating a single point of failure.

Verizon uncovered unauthorized access to both the business and controls networks. Vericlave [102] stated that an SQL injection attack in combination with social engineering might have been the most probable method of exploitation ①. From there, the attackers gained access to the Web server that hosted the payment portal ② and managed to leak 2.5 million customer records.

The attackers retrieved a list of credentials from a configuration file stored in plaintext form in the Web server's filesystem ②. Interestingly, the credentials were also reused in the SCADA applications. Therefore, they were able to manipulate the industrial process ③.

The incident occurred mainly because of the inadequate ICS network segmentation, the improper configuration of the services (Internet exposed, access to AS/400 from external IPs), the use of outdated hardware and software, as well as the lack of cybersecurity awareness that could prevent social engineering attacks.

5) Slammer Worm

The Slammer worm in 2003 [103] managed to disable the monitoring system of a nuclear power plant in the Ohio Davis-Besse [104]. The worm was based on a vulnerability in the Microsoft SQL Server 2000 (CVE-2002-0649), and it penetrated the nuclear power plant's network via a contractor's laptop that was connected to the business network of the facility.

The worm managed to reach the monitoring system by leveraging the improper network isolation and made it inaccessible due to the excessive amount of traffic that was created. There were not any hazardous physical consequences or data theft from this incident since the plant was offline for maintenance. Therefore, the impact of this incident was minimal.

6) SoBig Virus

In 2003, a shutdown of systems that manage train signals in Florida, U.S. is attributed to the SoBig virus [105]. It infected the SCADA systems via e-mail attachments and propagated quickly. However by this infection, neither major problems were caused in the control process nor data exfiltration. Thus, this incident is omitted from our subsequent discussions.

7) Tehama Colusa Canal

A former employee in 2007 [106], installed malicious software on the Tehama Colusa Canal Authority SCADA system

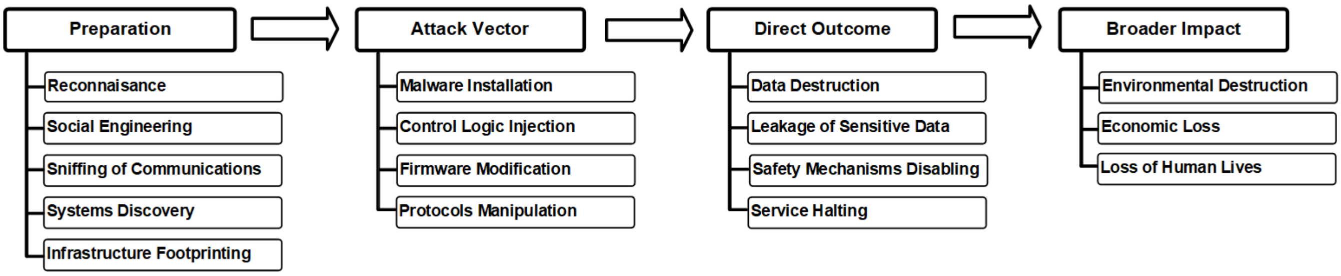


FIGURE 9. Alternative actions taken during the major phases of the studied ICS and CI incidents.

that was used to divert water from the Sacramento River to provide various services to the local area. Nevertheless, no further details for this incident were published.

8) U.S. power grid intrusion

Foreign nation-states were reported to have accessed U.S. power grid utilities in 2009 [107]. Allegedly, the adversaries gathered information about the infrastructure. However, technical details that identify the compromised systems and the adopted methods, do not exist in the public domain.

M. DISCUSSION

A longitudinal analysis of known incidents against CI, attests that the malware reconnaissance and exploitation phases tend to evolve towards much simpler methodologies. We have seen that initially, malware was designed to infect specific devices, inside specific infrastructures. Modern incidents rely on much more generic malware and methodologies. As an example of this trend, we have incidents like Stuxnet on the one side of the spectrum. Being released in 2009-2010, the malware relies on sophisticated self-propagating functions and zero-day vulnerabilities, but it is custom-tailored to impact only Siemens PLCs inside a specific network. Similarly, the Maroochy Water Services incident is a case where the attacker has full knowledge of hardware, software and the corresponding configuration for the particular installation. On the other side of the spectrum, the 2015 Ukrainian power grid attack, demonstrated that even without a custom-tailored malware in place and by solely relying on well-known vulnerabilities of the IT and OT systems, an attacker could penetrate the ICS. Such incidents may impact physical processes causing considerable disruption for a non-negligible amount of time. This tendency has also been observed in more recent incidents that involved the WannaCry, NotPetya, and DarkSide malware. The effects of the described cyberattacks against CI are recapitulated in Figure 9.

1) Common tools and approaches

Adversaries rely increasingly on the use of commodity tools for the reconnaissance and attack phases rather than developing them from scratch. As indicative example of this trend is the use of Mimikatz for credentials harvesting by Industroyer and NotPetya. Early incidents like those related to Duqu, leveraged custom keyloggers for that purpose. However, the adversaries abandon these tactics due to the effort and the time that has to be invested in achieving the desired results.

Another trend is the installation of malware, such as Stuxnet that it uses legitimate drivers signed with a valid (but stolen) private key. Despite being stealthy, this approach is also discarded, as the process to steal digital certificates, to sign the drivers, is very laborious.

Actually, a trend observed during the last few years with instances like BlackEnergy, Industroyer and VPNFilter is for malware to adopt a modular architecture. This allows the adversaries to extend their attacking repertoire on-the-fly, by relying on existing modern and possibly more effective components.

An interesting discovery unveiled in our study, revolves around the use of wiping software (Shamoon, 2015 and 2016 Ukraine powergrid attacks). This is a commonly adopted technique by adversaries to cover their tracks and to make the recovery of the impacted systems cumbersome. This provides an indicator that well-tested techniques are adopted by numerous actors despite targeting different sectors. Therefore, the goal of causing loss of view (LOV), loss of control (LOC), and potentially loss of safety (LOS) can be achieved with minimum innovation in terms of tactics.

We have also seen that even benign tools that inherently exist in these environments, such as PowerShell (Industroyer, Havex and NotPetya), and OPC (Havex, Industroyer) to be used against the targeted organizations. For decreasing the chances of malicious abuse, such organizations should harden, monitor, and especially protect their special-purpose tools.

2) Vulnerabilities categorization

By dissecting the attack methodologies observed in the described incidents, one could categorize the exploited vulnerabilities as:

- Type 0, zero-day vulnerabilities.
- Type 1, known vulnerabilities.
- Type 2, vulnerabilities stemming from inherently insecure services, protocols.
- Type 3, vulnerabilities relevant to insecure configuration of networks and equipment.
- Type 4, social engineering.

TABLE 2. Categorization of incidents based on their Actions, Targeted systems, Initial Infection points and Highlights.

	Action	Target	Initial Infection	Highlights
Stuxnet ([24])	• Lateral Movement • Infection Reporting • Infection of PLC • Centrifuges Destruction	• PCs • EWs • PLCs	• USBs • PGs • Network Shares	• Use of Zero-Days • PLC Manipulation • "Air Gap" Penetration
Duqu ([28])	• Reconnaissance • Data Exfiltration	• PCs	• Phishing • Malicious Documents	• Use of Zero-Day • P2P communication
Shamoon ([31])	• Reconnaissance • Data Exfiltration • Wiping of Drives	• PCs	• Phishing	• Drive Wiping
Havex ([40])	• Reconnaissance • Data Exfiltration	• PCs • OPC Servers	• Phishing • Compromised Websites • Compromised Vendors	• Multiple Infection Modes • Protocols Reconnaissance • OPC Targeting
BlackEnergy ([45])	• Reconnaissance • Credentials Harvesting • Wiping of Drives • Closing of Breakers	• PCs • UPS • VPN Servers • HMI • DMS • Circuit Breakers	• Phishing • Malicious Documents	• Multiple Attack Vectors
Industroyer ([56])	• Reconnaissance • Credentials Harvesting • Persistence • Issue of Commands • Closing of Breakers	• PCs • VPN Servers • Historians • OPC Servers • Circuit Breakers • Protective Relays	• Phishing • Malicious Documents	• Automated Infections • Multitude of Protocols
Triton ([61])	• Protocol Manipulation • Infection of SIS • Potential Unsafe Conditions	• EW • SIS Controller	• Misconfigured Firewall	• Manipulation of Protocol • Zero-Day
VPNFilter ([68])	• Reconnaissance • Wiping of Drives	• Routers • NAS	• Default Credentials • Unpatched Systems	• Covert Channels • Tor Usage • Protocol Reconnaissance
WannyCry ([73])	• Lateral Movement • Encryption of Drives	• PCs	• Unpatched Systems	• Common Exploit • Covert Channels
NotPetya ([81])	• Lateral Movement • Encryption of Drives	• PCs • Network Share	• Accounting System	• External Entity Infection • Common Exploit
Colonial ([89])	• Encryption of Drives	• PCs	• Compromised Credentials	• Reuse of Old Credentials
Steel Mill ([94])	• Blast Furnace Destruction	• Blast Furnace	• Phishing	• Blast Furnace Destruction
Maroochy ([96])	• SCADA Commands • Sewage Release	• RTUs • Sewage Pumps	• Knowledge of System • Owned Equipment	• Insider
NY Dam ([97])	• Reconnaissance	• Cellular Modem	• Unprotected Modem	• Unprotected Network
"Kemuri" ([101])	• Credentials Harvesting • Data Exfiltration • Issue of Commands	• Web Server • Router • SCADA	• Social Engineering • Unpatched System	• Network Misconfiguration
Slammer ([103])	• Denial of Service	• Monitoring System	• PC	• Nuclear Facility Impact

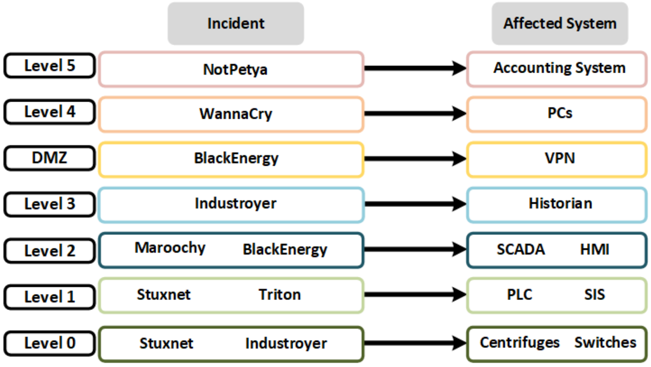


FIGURE 10. Correspondence of incidents to affected systems; For a holistic view, see Table 2.

TABLE 3. Taxonomy of incidents based on the Types vulnerabilities exploited.

	Type 0	Type 1	Type 2	Type 3	Type 4
Stuxnet ([24])	•	•			
Duqu ([28])	•				•
Shamoon ([31])				•	•
Havex ([40])			•	•	•
BlackEnergy ([45])				•	•
Industroyer ([56])	•		•	•	•
Triton ([61])	•		•	•	
VPNFilter ([68])		•	•		
WannyCry ([73])		•	•		
Colonial ([89])				•	•
NotPetya ([81])		•	•	•	
Steel Mill ([94])					•
Maroochy ([96])			•	•	
NY Dam ([97])				•	
“Kemuri” ([101])		•		•	•
Slammer ([103])		•		•	

† Type 0: zero-day vulnerabilities, Type 1: known vulnerabilities, Type 2: insecure protocols, Type 3 : insecure configuration, Type 4: social engineering

Based on the above categorization, we observe that a form of social engineering (**Type 4**) is omnipresent, especially during the early reconnaissance stages of the attacks (Duqu, Havex) or as a way of gaining a foothold in the systems of the targeted organizations (BlackEnergy, Industroyer, German Steel Mill, ”Kemuri” Water Company). Social engineering attacks can be prevented mainly by educating the employees, frequent training sessions that simulate social engineering attempts, as well as with properly documented policies in place.

Type 3 vulnerabilities are exploited by adversaries to access the OT environment directly (VPN) or indirectly, via the IT systems. This can be achieved with credentials harvesting and/or leverage of misconfigurations. Equipped with this knowledge, the attackers create and test the appropriate payload, and in the end, deliver it to their target. In the Triton incident, the misconfiguration of a firewall allowed the adversaries to gain access to the EW that was used to communicate with the SIS controller. Naturally, regular security assessments and the use of multi-factor authentication can mitigate these issues.

Type 2 vulnerabilities derive from pre-existing security flaws in the adopted protocols. For example, the Industroyer malware issued the commands to the switches and circuit breakers, without the need for any authentication. Once the adversaries reach this level, their tasks become easier (but occasionally time-consuming), even when the asset owners used proprietary protocols. In these cases, the use of modern and updated protocols, that provide better security is needed, although the update process is not trivial in many of the ICS. We expand on this in Section VI.

It is also observed that existing unpatched vulnerabilities (**Type 1**) can have devastating effects on the ICS and CI. WannaCry and NotPetya leveraged the negligence of update from the organizations and managed to infect numerous devices. In the “Kemuri” Water Company incident, a SQL injection vulnerability that was unaddressed provided a window of opportunity to the adversaries. The early discovery of those vulnerabilities and the update of the systems when possible reinforce the security of the ICS.

Zero-day vulnerabilities, i.e., **Type 0** are not common, but dedicated attackers (Stuxnet, Duqu, Industroyer, Triton) can use them against what they consider vital targets. Once again, the earlier those vulnerabilities are discovered by the vendors and the asset owners’ systems are patched, the lower are the chances of exploitation. Other countermeasures, if patching is not possible, may include network segregation, anomaly detection mechanisms, or use of equipment from different vendors (security-through-diversity) that can increase the overall security of the environment.

To provide a holistic view of the factors that enabled each incident, we categorize the observed vulnerabilities of the discussed incidents in Table 3. Moreover, the reader can perceive a categorization of each incident and some of the affected Purdue levels in Figure 10. A more detailed one, along with the rest of the related information discussed in this section, is presented in Table 2.

3) Affected Purdue Levels

Most of the discussed incidents span across multiple levels. For example, Stuxnet infects the PCs and the EWs (levels 4 and 2) that will transfer the project files to the PLC (level 1). While performing its malicious actions against the centrifuges (level 0), it also affects the view of the operators (level 2). Duqu infects only IT systems, however, the information retrieved (such as credentials) can be used in subsequent attacks that target systems across multiple Purdue levels. Shamoon affects IT systems as well, but its wiping capabilities can have repercussions indirectly to the OT side of an organization. A similar behavior is observed in NotPetya and Colonial pipeline incidents (levels 5 and 4). Havex exfiltrates information from the IT systems and also moves to level 3 of the OT environments, when the conditions allow to do so.

BlackEnergy is used to equip the adversaries with the toolset to perform reconnaissance in the IT and access software such as VPN and remote access tools. Having this type of access, they can connect directly to the OT and perform their malicious actions. The included wiper component can also affect equipment in both IT and OT. From a bird’s view,

TABLE 4. Mitigation Strategies.

	Secure Remote Access	Patch Management	Credential Management	Network Segmentation	Software Restriction Policies	Outbound Traffic Detection	Execution of Explicitly Allowed Software	Audit Network Hosts for Suspicious Files	Secure Configuration Management	Incident Planning and Response	Awareness and Training
Stuxnet ([24])			•		•	•	•	•			
Duqu ([108])					•	•	•	•			•
Shamoon ([31])			•		•	•	•	•			•
Havex ([40])				•		•	•	•			•
BlackEnergy ([45])	•		•		•	•	•	•			•
Industroyer ([56])					•	•	•	•			•
Triton ([61])				•	•	•	•	•	•		
VPNFilter ([68])		•				•					•
WannyCry ([73])		•	•	•	•	•	•	•			
NotPetya ([81])		•	•	•	•	•	•	•			
Colonial ([89])	•		•	•	•	•	•	•			•
Steel Mill ([94])											•
Maroochy ([96])	•									•	
NY Dam ([97])	•		•								
“Kemuri” ([101])			•	•		•					•
Slammer ([103])		•		•					•		•

[1] blank is not applicable or unknown

Industroyer follows a similar approach. However, it has the additional capability of accessing the equipment that controls switches, circuit breakers at the lowest of the Purdue levels. Triton performs its actions against SIS controllers (level 1). However, for doing so, it requires prior access to an EW (level 2).

VPNFilter infects routers that exist in numerous Purdue levels, and gathers information from the traditionally IT systems (levels 5 and 4) as well as from routable industrial protocols (level 3). In the German Steel Mill incident, the furnace equipment was damaged, something that requires prior adversarial access to some or all of the above mentioned Purdue levels. In the case of the Maroochy Water Services incident, the attacker gained access to the SCADA system (level 2) and issued the commands to the RTUs (level 1). This enabled the pumps to open and release sewage (level 0).

The attackers in the New York Dam intrusion, accessed the SCADA system in level 2 of the Purdue model and retrieved information about the conditions of the dam. In the “Kemuri” Water Company incident the assailants started by accessing the IT system (levels 5 and 4), and due to misconfigurations, they were able to issue commands to the equipment that regulates the water flow and blends the chemicals via the SCADA system (levels 3 and 2).

4) Mitigation

Having all the above details about the so far occurred major incidents, this section attempts to provide mitigation strategies that could have prevented incidents from happening or could have minimized their impact. The mitigation strategies follow those provided by organizations such as the CISA [109], [110] and NIST [6]. The reader should take into account that all the below recommendations are formed based on the description of the above incidents and the affected Purdue levels. Therefore, every mitigation strategy does not apply universally across every ICS installation and CI sector. The different risks should be identified and the corresponding safeguards must be installed for protecting the critical systems. A holistic view of incidents and their respective mitigations can also be found in Table 4.

Level 5: It has been demonstrated that this level accounts for the main entry in numerous attack incidents. Either a collaborator of the affected infrastructure (Havex) or the infrastructure itself (“Kemuri”) can provide unintentionally access to the assailant. From there, the adversary can inflict damage to the organization directly or use it as a stepping stone to move to other levels, as demonstrated in the “Kemuri” Water Company incident. By having secure remote access, adequate patch management procedures, secure credentials management, proper networks segmentation, and security awareness culture, similar attacks can be prevented.

Level 4: This level can be considered once again the low-hanging fruit for accessing CI environments. There is a large number of adversaries that find level 4 attractive to cause direct havoc. As an example, Shamoon and the DarkSide ransomware attacked the equipment that resided in this level to disrupt the operations of the targeted organizations. This level can also be used indirectly to deliver other malicious payloads to the lower levels, e.g., Stuxnet and Industroyer. Similarly to level 5, securing remote access, having good patch management procedures and software restriction policies, managing credentials securely, and providing trained employees, lessens the impact of attacks that involve this level.

Level 3: For the organizations that employ OT systems, level 3 is considered particularly important as it provides a connecting point between site operations and business management. This is mainly due to the data collection and the management of the OT devices. An evildoer that reaches this level can acquire a plethora of information and establish persistence to the OT network. Such persistence can be used to launch further attacks as seen in Havex, and the 2015 and 2016 Ukraine powergrid attacks. Therefore, the applied patch management, the secure configuration management, the segmentation of networks wherever this is possible, and the detection of outbound traffic, can put the defender in an advantageous position. This can also be the level where effective and efficient countermeasures can be deployed in a fully or semiautomated way if an intrusion is detected [111].

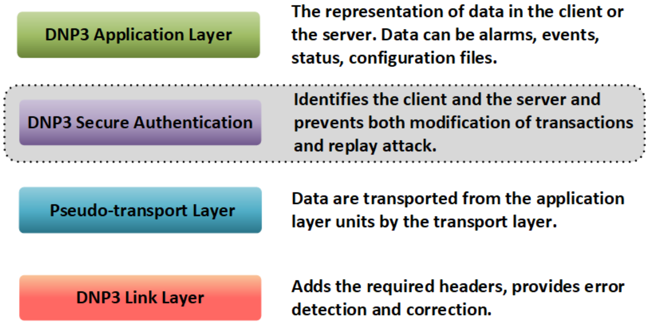
Level 2: Based on the degree of trust that exists at this Purdue level, if the attacker manages to access the devices that lie there, they can download new software to the controllers or issue commands to protective relays, as demonstrated in the Triton and the Industroyer malware correspondingly. This level also provides important information to them with regard to the exact management and control of the physical process. Important mitigations at this level include, the execution of explicitly allowed software, networks segmentation, patching and the auditing of the devices whenever that is possible, monitoring of the outbound traffic, and performing secure configuration management of the devices.

Level 1: This level is the one of most challenging in terms of protection via cyber means. Devices such as PLCs, SIS controllers, and digital protective relays have sometimes minimal security, run outdated and proprietary software, while also communicate over protocols that cannot safeguard the transfer of information. Paradigms of such weaknesses have been exposed by Stuxnet, Industroyer and Triton. The need for continuous operation and their proximity to the controlled process, makes it inadvisable to update such devices' software even in the cases that patches are available. Therefore, the mitigations that can take place at level 1 are network segmentation, outbound traffic detection, and secure configuration management. Anomaly detection especially in such ways that minimal or even zero alterations occur to the environment, is another tool that can provide early responses of compromise [112].

Level 0: At this level, the main issue is that erroneous sensor data can lead to incorrect control-level decisions. Furthermore, from an actuator viewpoint, there is no authentication of the source and the integrity of commands, and such commands can lead to catastrophic actions against the equipment. A straightforward approach is to create a separate level 0 monitoring network and compare the sensor data sent to Level 2 with the data received on the original level 0 monitoring network. Other more sophisticated mitigations include the use of autonomous and/or external defenses that can estimate the state of sensors/actuators based on physics-based models [113], [114]. For the actuation decisions, all the commands should be authenticated using mechanisms of the industrial protocols [115] (see Section VI).

Attacks that source from insiders should be given the same attention as external attacks. However, most insider attacks cannot be prevented, and therefore, there is a need for rapid detection. More traditional countermeasures include the exit interview of employees who leave the organization, the installation of proper access controls, and the immediate decommissioning of expired credentials.

Additionally, enhancing resiliency aspects is crucial for any organization that employs ICS and CI to reduce the impact of adversarial tactics. For example, incidents that used wiping malware like the 2015 Ukrainian powergrid attack and NotPetya can severely affect the recovery process. Therefore, it is of high importance that operations, security, and C-suite level personnel to have a deep understanding of the physical process and various risk indicators. With that in place, the organizations can focus on building secure and resilient mechanisms around their "crown-jewels" and then move towards assets that can tolerate greater disturbance.



The representation of data in the client or the server. Data can be alarms, events, status, configuration files.

Identifies the client and the server and prevents both modification of transactions and replay attack.

Data are transported from the application layer units by the transport layer.

Adds the required headers, provides error detection and correction.

FIGURE 11. The DNP3 protocol stack. The grey area is an additional layer introduced in later versions.

VI. ICS PROTOCOLS VULNERABILITIES

This section discusses known vulnerabilities of the well-established ICS protocols DNP3, Modbus, IEC-104, IEC-61850, PROFINET, WirelessHART, ZigBee, EtherNet/IP and OPC. Such protocols can reside in various levels of the Purdue model. It is worth noting that vulnerabilities may exist in other legacy IT protocols that are also used in ICS, including HTTP, ARP, and even Telnet. This section however, intentionally focuses on protocols destined specifically to ICS.

A. DNP3

The work by East et al. [116] offered a taxonomy of nearly thirty attacks that can be performed against DNP3-oriented systems. The attacks were categorized based on four criteria, namely the target, the threat category, the layers of the protocol that are exploited, and the impact that they cause to the implemented systems. In [117] Jin et al. advocated that a SCADA network consisting of DNP3 devices is vulnerable to flooding attacks that can take place due to a surge of fake unsolicited responses, one of the main distinguishable features of DNP3.

Darwish et al. [118] scrutinized the behavior of the DNP3 protocol in smart-grid installations by verifying some of the vulnerabilities mentioned in [116]. Another work by Darwish et al. [119] presented an approach that can be used to model DNP3 attacks against the smart-grid realm. Their setup comprised a virtual environment where the attacker was able to drop and manipulate packets through a MitM attack.

Rodofile et al. [120] implemented an extension for the Scapy Python library with the purpose of crafting DNP3 packets having as ultimate goal to validate some of the attacks presented by East et al. [116]. Crain and Bratus [121], demonstrated the use of a fuzzing approach that identifies vulnerabilities in the DNP3 protocol [122], namely the lack of message confidentiality, integrity, authentication, and authorization. The possibility for unauthorized message modification, replay, and spoofing attacks has been identified in DNP3-Secure Authentication (SA) by Amoah et al. [123]. The vulnerability that leads to these attacks stems from the Aggressive Mode (AGM) mechanism of DNP3-SA.

Given the above discussion and as illustrated in Figure 11, DPN3 lacks the support of basic security features such as confidently, integrity, availability, and authenticity. Simply put, anyone who is able to reach devices at that level of the ICS, can straightforwardly inspect and manipulate the exchanged messages. Implementation and design vulnerabilities exist also in DNP3-SA, although to a lesser extent.

B. MODBUS

Possible vulnerabilities in the Modbus specification and major implementations of the protocol were investigated by Huitsi [124]. Such weaknesses can be exploited to perform spoofing, message replay, and flooding attacks. Morris et al. [125] detailed theoretical data injection and DoS attacks against industrial equipment that relies on Modbus. Such attacks stem from the protocol's insufficient security mechanisms for data integrity and availability. Latter, the work by Gao and Morris [126] described and tested reconnaissance, response injection, command injection, and DoS attacks, and also elaborated on several standalone and stateful IDS rules in an effort to deter such incidents.

After confirming that Modbus is prone to flooding attacks, Bhatia et al. [127] devised and assessed anomaly and signature-based detection as a means of mitigating them. Nardone et al. [128] formally assessed and evaluated the security of the Modbus protocol in terms of the security features each variant provides. The work by Tsalis et al. [129] demonstrated that even in the presence of encryption, side-channel attacks might reveal information on Modbus protocol messages.

Using a testbed comprising of virtual machines running on Linux, Parian et al. [130] detailed on two attacks, namely manipulation of packets via malware-infected hosts and classic MitM attacks using ARP poisoning. A very similar MitM attack on Modbus has also been demonstrated by Chen et al. [131].

Even if Modbus is one of the oldest and most well-established industrial protocols stacks, it has not until recently included any mechanisms that target the provision of fundamental security services. Actually, the protection of the confidentiality, integrity and authenticity of messages, have been introduced with Modbus/TCP Security in 2018. However, these enhancements have not penetrated the market yet.

C. PROFINET

Baud and Fester [132] were the first to investigate the potential of mounting MitM attacks against a PROFINET network. Akeberg and Bjorkman [133] elaborated on the feasibility of attacking and gaining control over a PROFINET node using two attacks, a MitM one and an assault based on a race condition bug. Hui and McLaughlin [134] investigated the security issues that possibly exist in newer Siemens PLCs and uncovered vulnerabilities of the PROFINET's discovery protocol (DCP). They also detailed on a weakness in the protocol's anti-replay mechanism, based on the lack of integrity check in the acknowledgment packets.

Pfrang and Meier [135] exploited vulnerabilities of PROFINET to conduct two attacks against systems that rely on it. They leveraged vulnerabilities of (a) switch port stealing and (b) the lack of any authentication measures in DCP. Lately, by exploiting a vulnerability in DCP, Mehner and Konig [136] explored a DoS attack that interrupts the Application Relationship (AR) between a PROFINET controller and a device, and subsequently obstructs the repair of the system.

By summarizing the above, it can be discerned that PROFINET is vulnerable to simple attacking techniques. Specifically, violation of availability, and message authenticity and integrity, can be achieved, allowing DoS, MitM, and replay attacks, even against modern devices.

D. OTHER PROTOCOLS

The work from Yang et al. [137] provided suggestions on potential ways to exploit IEC-104 vulnerabilities that stem from the lack of data integrity, availability, and authentication mechanisms to perform attacks against power systems. In [138], Maynard et al. have also focused on specific IEC-104 vulnerabilities that can lead to MitM and replay assaults.

IEC-104 is popular in the electrical sector, and therefore it appears to be a lucrative diode for attackers that wish to inflict damage to this critical ecosystem. Indeed, the lack of authentication and anti-replay mechanisms in this protocol can make it possible for adversaries to inject malicious or even issue unauthorized commands in the network.

With the help of a custom testbed environment, Yang et al. [139] performed fuzzing in order to evaluate the security of the protocols that fall under the IEC-61850 umbrella, and identified poor implementations in certain protocol stacks. Kabir et al. [140] scrutinized the GOOSE protocol of the IEC-61850 standard using a properly configured testbed, and mounted attacks that are possible due to the lack of authentication and encryption. The work by Silveira and Franco [141] also presented a handful of attacks that originate from vulnerabilities of the IEC-61850's GOOSE protocol.

IEC-61850 is a relatively new protocol and favorably comes with integrated security features to remedy the issues of its predecessors. Nevertheless, researchers have identified flaws mainly in the way the protocol is implemented, which in turn may leave room for message spoofing and replay attacks.

The work by Raza et al. [142] pinpointed several vulnerabilities in the WirelessHART protocol. These lead to packet flooding, gateway spoofing, traffic analysis, resource exhaustion, and desynchronization attacks. Samaddar et al. [143] introduced timing attacks in WirelessHART networks. They elaborated on how such an attack can aid an aggressor in analyzing the eavesdropped traces of the real-time data flows to infer the schedule of the exchanged data.

Since WirelessHART transmits all the information over the air, a lot of attention is given to secure the protocol from eavesdroppers. As presented in the relevant works, the disruption of the exchanged data, can be used to cause commotion or interruption to the industrial process.

In [144] Wright presented a methodology and a corresponding tool for manipulating the distribution of keys in Zigbee protocol with the purpose of decryption or injection of messages. Kennedy and Hunt [145], detailed on an association flooding attack that may occur if the coordinator of the Zigbee network does not limit the number of association requests.

Replay attacks in ZigBee are possible if the participating devices in the network use the same network-wide key as presented by Farha and Chen [146], and other review works [147]. This vulnerability is rooted in the incorporation of the frame counter, which was introduced to defend against replay attacks.

ZigBee vulnerabilities mainly relate to the acquisition of the network-wide encryption key by adversaries. Typically, for protocols with encryption capabilities, the secret keys comprise the most critical asset of the system, and careful consideration must be made regarding their creation, exchange, and storage.

Grandgenett et al. [148] performed an analysis of Allen-Bradley's implementation of EtherNet/IP [149], and identified that DoS attacks are feasible. The work of Urbina et al. [150] demonstrated how MitM attacks in EtherNet/IP protocol and related topologies such as the ring topology, can be used to modify sensor measurements and influence actuators in a water treatment testbed. A fuzzing tool coined *ENIP Fuzz* destined to EtherNet/IP and parts of the Common Industrial Protocol (CIP) was created by Tacliad et al. [151], allowing the authors to identify a vulnerability in the File Type value of the protocol that can lead to DoS.

As it is summarized from all the above, the widely used EtherNet/IP protocol and its implementations include vulnerabilities that can lead to data manipulation and DoS. These are two of the most severe attacks that can be triggered against an ICS environment.

The work of Qi et al. [152] extends the work of Wang et al. [153] that detected flaws in OPC, Distributed Component Object Model (DCOM), and Remote Procedure Call (RPC) parts of the ecosystem by leveraging a custom fuzzer called *OPCMFuzzer*. Puys et al. [154] have formally analyzed OPC-UA protocol and identified flaws in the cryptographic signing of messages, as well as in the authentication mechanisms. Finally, the work by Roepert et al. [155], demonstrates various methods that can be used to discover vulnerabilities in OPC-UA servers, mainly authentication bypass and DoS.

It is concluded that OPC is prone to common attacks that are met in legacy IT environments. Insecure RPCs and insufficient validation along with specific product implementation errors can provide the adversaries with a handful of knowledge of exchanged data, including "secret recipe", since the nature of OPC is to interconnect diverse ICS components.

VII. ICS DEVICE VULNERABILITIES

This section is dedicated to describing vulnerabilities specific to ICS devices. These devices operate primarily at the lower levels of the Purdue model, as illustrated in Figure 1. Thus, this section intentionally focuses on academic works that discover weaknesses and prove the feasibility of possible assaults.

A. REVERSE ENGINEERING

The methodology introduced in [156] and subsequently the tool in [157], aim to automate the malicious payload construction process for PLCs. The presented tool coined *SABOT* receives a high-level specification of the device's behavior as input. The tool then retrieves the benign control logic bytecode from the target PLC and automatically identifies mappings of PLC memory locations to physical ones, and modifies a generic malicious payload into one capable of infecting the target PLC.

In the same context, Keliris and Maniatakos [158] present a framework for automatically reverse-engineering full PLC binaries with the aim of reconstructing the complete Control Flow Graph (CFG) of the control logic. A decompiler for the ladder logic called *Laddis* is presented in [159]. Laddis can decompile a program on the fly by observing packets that contain control code transmitted during the PLC configuration cycles, and decompiles it into a human-readable ASCII format. As part of the control logic attack presented in [160], the authors contributed a decompiler referred to as *Eupheus* that produces the Instruction List (IL) source code from binaries specific to the RX630 platform.

From the above it can be concluded that besides its application for benign purposes, reverse engineering can be considered as the first step to perform malicious actions against an ICS, as it can reveal information about the target hardware or software.

B. CONTROL LOGIC INJECTION & MODIFICATION ATTACKS

The control logic programs that run in a PLC specify how that device will control aspects of the physical process. Malicious alteration of the control logic of PLCs using three attacks presented in [159], are capable of degrading the integrity and availability of the system. These three attacks are: (a) a MitM that hides the change of control logic, (b) a MitM to replace a selected number of control logic instructions with arbitrary instructions, and (c) a form of DoS against the EW in the case the latter attempts to obtain a maliciously manipulated control logic from the PLC.

Kalle et al. [160] present the so-called "CLIK" attack that consists of four main steps: (a) direct compromise of the PLC to acquire the control logic, (b) decompilation of the binary and injection of malicious instructions, (c) "download" of the altered version of the control logic back to the PLC, and (d) concealment of all the actions from the EW with the use of a virtual-PLC. Yoo and Ahmed [161] explored various methods of injecting malicious logic to a PLC directly through the network, without requiring physical access to a PLC. They describe two alternative attack methodologies, namely Data Execution, and Fragmentation with Noise Padding.

The malicious manipulation of control logic through the means described in this subsection, can cause serious problems to the physical process and the controlled equipment. Particularly for legacy devices, modern protection mechanisms such a data execution prevention (DEP) are simply non-existent.

C. LADDER LOGIC BASED ATTACKS

Govil et al. [162] introduced the concept of ladder logic bombs (LLBs), i.e., malicious snippets of ladder logic that may be implanted in the benign logic by a malicious engineer with direct access to the EW. LLBs can lead to (a) DoS, (b) manipulation of sensor readings and commands, and (c) stealthy logging of data. The work of Serhane et al. [163] focuses on ladder logic code vulnerabilities or simply bad code practices that may become the root cause of bugs and subsequently be exploited by attackers.

Ladder logic is one of the IEC61131-3 compatible languages for programming control logic in PLCs. As a visual programming language it can be sometimes challenging to identify differences between malicious and benign versions, especially to the inexperienced eye, as indicated by the aforementioned works.

D. NATIVE ICS MALWARE

Spenneberg et al. [164] demonstrated the first Proof of Concept (PoC) worm written in structured text that propagates among PLCs without the involvement of an EW. This can be achieved due to inadequate security measures such as the lack of integrity protection in the PLC and the default (turned off) settings of the access protection. The work from Garcia et al. [165] presented HARVEY, a rootkit that once it is installed in the device's firmware, has the capability to inspect the control logic and then modify its instructions. The rootkit is also aware of the control process that the PLC handles and can intercept the measurement inputs that are used by this process. Yet, firmware level modification is assumed not trivial, since most of the time the firmware can be updated only through direct physical access.

The majority of the PLC infecting malware capitalizes on vulnerabilities of the EW or other platforms that are based on commodity hardware and software, say, PCs running MS Windows OS. Several works, however, proved that it is possible to create malware that operates directly at the PLC side.

E. UNAUTHORIZED ACCESS

Beresford [166] identified several vulnerabilities regarding the Siemens Simatic S7 PLCs. These vulnerabilities can be used by malicious actors to perform replay attacks, authentication bypass, DoS, remote memory dumps, and access via remote shell. Klick et al. [167] demonstrated how an attacker could extend access to all PLCs in the production network and, depending on the circumstances, the corporate IT network by leveraging injection of Statement List (STL) code in an Internet-facing PLC, SNNP scanning, and SOCKS proxy installation. To automate the steps of the attack, the authors provided a tool called *PLCinject*.

The work from Wardak et al. [168] investigated some issues existing in the access control mechanisms of S7-400 PLCs, and more particularly: no protection, write protection, and read/write protection. Keliris et al. [169] discussed a vulnerability discovered in the authentication mechanism of several protective relays of the General Electric (GE) Multilin protection and control family of products that stems from a weak, custom encryption algorithm for protecting passwords.

A study regarding the authentication protocols used by Schneider Electric, Allen-Bradley, Automation Direct, and Siemens PLCs, has been presented by Ayub et al. [170]. Among others, the researchers unveiled vulnerabilities rooted in the small-sized encryption key, the weak client-side authentication process, and the improper session management.

As presented in this subsection, unauthorized access can be one of the most severe vulnerabilities due to the fact that it can provide the attacker with full control of the compromised devices.

F. SIDE CHANNEL ANALYSIS

Krishnamurthy et al. [171] described the possibility for malware to rely on acoustic emissions of actuators, e.g., that of a motor controlling a valve as part of a closed-loop process, towards creating a covert channel that can ultimately retrieve a 128-bit key in little over four minutes. Tychalas and Maniatakos [172], examined the applicability of cache timing side-channel attacks, including Spectre and Evict-and-Reload. Theoretically, such assaults can be used to leak data from PLCs that utilize the Codesys framework [173].

Blinkware [174] is an attack that can achieve information leakage among embedded systems through the use of an optical side-channel. In the described example, sensitive information were transmitted via memory-mapped peripherals such as LED by copying data from arbitrary memory locations via the DMA controller. Similarly, the Waterleakabe malware [175] relied on the optical side-channel to achieve transmission of the sensor readings from lamps. Note that the lamps should be connected to the digital output of the PLC and the compromised video recorder camera should be placed one meter away.

Side-channel attacks can capitalize either or both electromagnetic, thermal, or acoustic signals that get involuntarily transmitted during the regular operational cycles of devices. Such channels can possibly reveal valuable information to passive observers, a situation which may progressively lead to leakage of sensitive information such as cryptographic keys.

VIII. CONCLUSION

This work explores the so far most important ICS and CI security incidents and scrutinizes their key aspects. The article also elaborates on the common factors and vulnerabilities that enabled these incidents and suggests potential mitigation measures that could possibly have prevented the unfolding of the corresponding attacks.

From our analysis, several important conclusions were extracted. For example:

- Social engineering practices are the first step of the majority of attack campaigns mentioned in this work. Real-life incidents, such as the 2015 and 2016 Ukraine powergrid attacks, were all based on such methodologies to provide the attacker with initial access to the target environment.
- Several incidents attest that attackers can easily penetrate OT environments after breaking into IT networks. From empirical observations, this is primarily due to the insecure configuration of these systems. For example, poor patch management and insufficient network segregation may expose OT to even "commodity" malware. Such is the case of the NotPetya and “Kemuri” Water Company incidents.
- Despite their critical mission, ICS devices are still susceptible to zero-days and exploits. Such resources tend to be part of the arsenal of highly-skilled attackers and nation-state actors that aim to inflict surgical strikes. Both Stuxnet and Triton were based on unknown but highly targeted attacks.

Moreover, within this work, we included an early-stage study on the relevant network protocols and key infrastructure components typically met in such realms. This study was conducted upon prestigious academic works, and it may act as an indicator for the characteristics of future incidents in the ICS arena. The most important conclusions extracted are:

- Antiquated network automation protocols did not consider security as a design tenet but rather introduced it as an afterthought. Nowadays, most of these protocols have shifted from paradigms based on serial bus communication towards IP-based models. Inadvertently, these protocols have become directly susceptible to the still-vast ocean of TCP/IP-based attacks.
- Many of the design and implementation inefficiencies in ICS devices are naive (e.g., rudimentary authentication mechanisms) and beget vulnerabilities. In turn, this leaves room for information theft, LOV, LOC, and LOS. Given the long system lifecycles, the required engineering resources, and vulnerabilities for which patches are unlikely to be applied, contemporary security mechanisms cannot always be straightforwardly administered.

Today, the once isolated and monolithic CI systems (for example, electricity, water, gas, manufacturing, and transportation) have evolved into increasingly complex and interlinked systems-of-systems. Provably this complexity has turned these infrastructures into a very fertile attack ground. Furthermore, novel communication paradigms such as IIoT along with the 5G and beyond communication technology are already culminating in a tighter union of systems. This underlines the requirement from modern security analysts to not isolate themselves in silos by solely concentrating on threats and vulnerabilities of specific sectors. Instead, cross-sector cyber-thinking and a comprehensive defense strategy are desired to fight off modern threats.

The quantification of the cyber risk posture is another challenging aspect in ICS and CI since these complex systems include a plethora of uncontrollable risk states. New approaches are needed to analyze the miscellaneous points of failure in the current risk assessment methods. Such approaches can effectively be used by practitioners and regulators to advise the various organizations on how to create plans for addressing potential uncontrollable risks.

Without any doubt, the ICS and CI play a vital role in the well-functioning of modern society. Therefore, instead of focusing on ad-hoc solutions to combat specific malware, organizations should concentrate on deploying generic countermeasures. Well orchestrated mitigation strategies could deter or prevent the early stages of any potential attack, known or unknown, and improve the control processes’ reliability and resiliency.

Future extensions of this work will further investigate bleeding-edge vulnerabilities and exploitation strategies that have been described in academia. Furthermore, we will attempt to identify the most promising mitigation measures that have been proposed in the past years and outline open research issues.

REFERENCES

[1] H. Holm, M. Karresand, A. Vidström, and E. Westring, “A survey of industrial control system testbeds,” in *Nordic Conference on Secure IT Systems*. Springer, 2015, pp. 11–26.

[2] I. Ghafir, J. Saleem, M. Hammoudeh, H. Faour, V. Prenosil, S. Jaf, S. Jabbar, and T. Baker, “Security threats to critical infrastructure: the human factor,” *The Journal of Supercomputing*, vol. 74, no. 10, pp. 4986–5002, 2018.

[3] T. J. Williams, “The purdue enterprise reference architecture,” *Computers in industry*, vol. 24, no. 2-3, pp. 141–158, 1994.

[4] E. D. Knapp and J. T. Langill, *Industrial Network Security: Securing critical infrastructure networks for smart grid, SCADA, and other Industrial Control Systems*. Syngress, 2014.

[5] P. Ackerman, *Industrial Cybersecurity: Efficiently secure critical infrastructure systems*. Packt Publishing Ltd, 2017.

[6] K. Stouffer, J. Falco, and K. Scarfone, “Guide to industrial control systems (ics) security,” *NIST special publication*, vol. 800, no. 82r2, 2015.

[7] DHS. (2013) Nipp 2013 partnering for critical infrastructure security and resilience. [Online]. Available: https://www.cisa.gov/sites/default/files/publications/national-infrastructure-protection-plan-2013-508.pdf

[8] BMI. (2009) National strategy for critical infrastructure protection (cip strategy). [Online]. Available: https://www.bmi.bund.de/SharedDocs/downloads/EN/publikationen/2009/kritis_englisch.pdf?__blob=publicationFile&v=1

[9] Australian Government Department of Home Affairs - Critical Infrastructure Center. (2018) Critical infrastructure resilience strategy: plan. [Online]. Available: https://cicentre.gov.au/document/P50S021

[10] Cabinet Office. (2017) Public summary of sector security and resilience plans. [Online]. Available: https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/678927/Public_Summary_of_Sector_Security_and_Resilience_Plans_2017__FINAL_pdf___002_.pdf

[11] COMMISSION OF THE EUROPEAN COMMUNITIES. (2004) Critical infrastructure protection in the fight against terrorism. [Online]. Available: https://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=COM:2004:0702:FIN:EN:PDF

[12] Cybersecurity and Infrastructure Security Agency. (2020) Industrial control systems. [Online]. Available: https://us-cert.cisa.gov/ics

[13] E-ISAC. (2020) Electricity information sharing and analysis center. [Online]. Available: https://www.eisac.com/

[14] B. Miller and D. Rowe, "A survey scada of and critical infrastructure incidents," in *Proceedings of the 1st Annual conference on Research in information technology*, 2012, pp. 51–56.

[15] S. McLaughlin, C. Konstantinou, X. Wang, L. Davi, A.-R. Sadeghi, M. Maniatakos, and R. Karri, "The cybersecurity landscape in industrial control systems," vol. 104, no. 5, pp. 1039–1057. [Online]. Available: http://ieeexplore.ieee.org/document/7434576/

[16] Y. Xu, Y. Yang, T. Li, J. Ju, and Q. Wang, "Review on cyber vulnerabilities of communication protocols in industrial control systems," in *2017 IEEE Conference on Energy Internet and Energy System Integration (EI2)*, pp. 1–6.

[17] K. E. Hemsley, E. Fisher *et al.*, "History of industrial control system cyber incidents," Idaho National Lab.(INL), Idaho Falls, ID (United States), Tech. Rep., 2018.

[18] A. Volkova, M. Niedermeier, R. Basmadjian, and H. de Meer, "Security challenges in control network protocols: A survey," vol. 21, no. 1, pp. 619–639.

[19] D. Bhamare, M. Zolanvari, A. Erbad, R. Jain, K. Khan, and N. Meskin, "Cybersecurity for industrial control systems: A survey," vol. 89, p. 101677. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0167404819302172

[20] M. M. Ahmadian, M. Shajari, and M. A. Shafiee, "Industrial control system security taxonomic framework with application to a comprehensive incidents survey," vol. 29, p. 100356. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1874548220300202

[21] T. Alladi, V. Chamola, and S. Zeadally, "Industrial control systems: Cyberattack trends and countermeasures," vol. 155, pp. 1–8. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0140366419319991

[22] D. Pliatsios, P. Sarigiannidis, T. Lagkas, and A. G. Sarigiannidis, "A survey on scada systems: Secure protocols, incidents, threats and tactics," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1942–1976, 2020.

[23] S. Caltagirone, A. Pendergast, and C. Betz, "The diamond model of intrusion analysis," Center For Cyber Intelligence Analysis and Threat Research Hanover Md, Tech. Rep., 2013.

[24] N. Falliere, L. O. Murchu, and E. Chien, "W32. stuxnet dossier," *White paper, Symantec Corp., Security Response*, vol. 5, no. 6, p. 29, 2011.

[25] R. Langner, "A technical analysis of what stuxnet's creators tried to achieve," p. 37.

[26] G. McDonald, L. O. Murchu, S. Doherty, and E. Chien, "Stuxnet 0.5: The missing link," p. 18.

[27] S. Events. Langner's stuxnet deep dive. [Online]. Available: https://www.youtube.com/watchv=zBjmm48zwQU

[28] E. Chien, L. OMurchu, and N. Falliere, "W32.duqu: The precursor to the next stuxnet," p. 2.

[29] Symantec. Duqu Updated targeting information. [Online]. Available: https://community.broadcom.com/symantecenterprise/communities/community-home/librarydocuments/viewdocument?DocumentKey=f6ae6c26-2641-4f53-a4d1-17d8a51f9e4d&CommunityKey=1ecf5f55-9545-44d6-b0f4-4e4a7f5f5e68&tab=librarydocuments

[30] Kaspersky. (2015) The duqu 2.0. [Online]. Available: https://media.kasperskycontenthub.com/wp-content/uploads/sites/43/2018/03/07205202/The_Mystery_of_Duqu_2_0_a_sophisticated_cyberespionage_actor_returns.pdf

[31] Symantec. The shamoon attacks. [Online]. Available: http://www.symantec.com/connect/blogs/shamoon-attacks

[32] H. MacKenzie. Shamoon malware and SCADA security – what are the impacts? | tofino industrial security solution. [Online]. Available: https://www.tofinosecurity.com/blog/shamoon-malware-and-scada-security-%E2%80%93-what-are-impacts

[33] J. Carr. Qas iran responsible for saudi aramco's network attack? [Online]. Available: https://jeffreycarr.blogspot.com/2012/08/was-iran-responsible-for-saudi-aramcos.html

[34] R. Falcone. Shamoon 2: Return of the disttrack wiper. [Online]. Available: https://unit42.paloaltonetworks.com/unit42-shamoon-2-return-disttrack-wiper/

[35] D. of Homeland Security. Shamoon/disttrack malware (update b). [Online]. Available: https://www.us-cert.gov/ics/jsar/JSAR-12-241-01B

[36] Shamoon the wiper: Further details (part II). [Online]. Available: https://securelist.com/shamoon-the-wiper-further-details-part-ii/57784/

[37] M. Edwards. Eldos provides raw disk access for vista and xp. [Online]. Available: https://www.itprotoday.com/windows-78/eldos-provides-raw-disk-access-vista-and-xp

[38] FireEye. FireEye responds to wave of destructive cyber attacks in gulf region. [Online]. Available: https://www.fireeye.com/blog/threat-research/2016/11/fireeye_respondsto.html

[39] Symantec. Shamoon: Destructive threat re-emerges with new sting in its tail. [Online]. Available: https://www.symantec.com/blogs/threat-intelligence/shamoon-destructive-threat-re-emerges-new-sting-its-tail

[40] ICS focused malware | CISA. [Online]. Available: https://www.us-cert.gov/ics/advisories/ICSA-14-178-01

[41] N. Nelson, "The impact of dragonfly malware on industrial control systems," p. 27.

[42] S. I. Response, "Dragonfly: Cyberespionage attacks against energy suppliers," *Rapp. tecn*, vol. 7, 2014.

[43] J. Rrushi, H. Farhangi, C. Howey, K. Carmichael, and J. Dabell, "A quantitative evaluation of the target selection of havex ICS malware plugin," p. 5.

[44] Havex, it's down with OPC. Library Catalog: www.fireeye.com. [Online]. Available: https://www.fireeye.com/blog/threat-research/2014/07/havex-its-down-with-opc.html

[45] D. of Homeland Security. Cyber-attack against ukrainian critical infrastructure | CISA. [Online]. Available: https://www.us-cert.gov/ics/alerts/IR-ALERT-H-16-056-01

[46] M. G. Kurt Baumgartner. Be2 custom plugins, router abuse, and target profiles. [Online]. Available: https://securelist.com/be2-extraordinary-plugins-siemens-targeting-dev-fails/68838/

[47] GReAT. Blackenergy apt attacks in ukraine employ spearphishing with word documents. [Online]. Available: https://securelist.com/blackenergy-apt-attacks-in-ukraine-employ-spearphishing-with-word-documents/73440/

[48] A. Cherepanov, "A new threat for industrial control systems," p. 17, 2016.

[49] P. Polityuk. Ukraine sees russian hand in cyber attacks on power grid. [Online]. Available: https://www.reuters.com/article/us-ukraine-cybersecurity-idUSKCN0VL18E

[50] A. Cherepanov and R. Lipovsky, "Blackenergy–what we really know about the notorious cyber attacks," *Virus Bulletin October*, 2016.

[51] A. Cherepanov. Blackenergy by the sshbeardoor: attacks against ukrainian news media and electric industry. [Online]. Available: https://www.welivesecurity.com/2016/01/03/blackenergy-sshbeardoor-details-2015-attacks-ukrainian-news-media-electric-industry/

[52] F. iSIGHT Intelligence. (2016) Overload: Critical lessons from 15 years of ics vulnerabilities. [Online]. Available: https://www.fireeye.com/blog/threatresearch/2016/08/overload-critical-lessons-from-15-years-of-icsvulnerabilities.html

[53] D. U. Case, "Analysis of the cyber attack on the ukrainian power grid," *Electricity Information Sharing and Analysis Center (E-ISAC)*, vol. 388, 2016.

[54] B. A. Hamilton, "Industrial cybersecurity threat briefing," p. 82, 2016.

[55] P. Behr and B. Sobczak. Grid hack exposes troubling security gaps for local utilities. [Online]. Available: https://www.eenews.net/stories/1060040519

[56] A. Cherepanov, "A new threat for industrial control systems," p. 17.

[57] J. Slowik. Anatomy of an attack: Detecting and defeating crashoverride. [Online]. Available: https://informationsecurity.report/Resources/Whitepapers/7a8d5f1e-8ac2-46c4-92f4-dfdc5615a65d_Anatomy_wp.pdf

[58] gentilkiwi. Mimikatz. [Online]. Available: https://github.com/gentilkiwi/mimikatz

[59] R. M. Lee, M. Assante, and T. Conway, "Crashoverride: Analysis of the threat to electric grid operations," *Dragos Inc., March*, 2017.

[60] I. Dragos, "CRASHOVERRIDE: Reassessing the 2016 ukraine electric power event as a protection-focused attack," p. 16.

[61] M. K. Blake Johnson, Dan Caban. (2017) Attackers deploy new ics attack framework triton and cause operational disruption to critical infrastructure. [Online]. Available: https://www.fireeye.com/blog/threat-research/2017/12/attackers-deploy-new-ics-attack-framework-triton.html

[62] M. Giles. (2019) Triton is the world's most murderous malware, and it's spreading. [Online]. Available: https://www.technologyreview.com/2019/03/05/103328/cybersecurity-critical-infrastructure-triton-malware/

[63] A. Di Pinto, Y. Dragoni, and A. Carcano, "Triton: The first ics cyber attack on safety instrument systems," in *Proc. Black Hat USA*, 2018, pp. 1–26.

[64] M. B. Labs. (2018) Analyzing the triton industrial malware. [Online]. Available: https://www.midnightbluelabs.com/blog/2018/1/16/analyzing-the-triton-industrial-malware

[65] S. Events. (2018) Triton - schneider electric analysis and disclosure. [Online]. Available: https://www.youtube.com/watch?v=f09E75bWvkk

[66] MDudek-ICS. (2017) Trisis-triton-hatman. [Online]. Available: https://github.com/MDudek-ICS/TRISIS-TRITON-HATMAN

[67] Dragos. (2018) Trisis malware analysis of safety system targeted malware. [Online]. Available: https://dragos.com/wp-content/uploads/TRISIS-01.pdf

[68] W. Largent. New VPNFilter malware targets at least 500k networking devices worldwide. [Online]. Available: http://blog.talosintelligence.com/2018/05/VPNFilter.html

[69] ——. VPNFilter update - VPNFilter exploits endpoints, targets new devices. [Online]. Available: http://blog.talosintelligence.com/2018/06/vpnfilter-update.html

[70] VPNFilter: New router malware with destructive capabilities. [Online]. Available: https://www.symantec.com/blogs/threat-intelligence/vpnfilter-iot-malware

[71] QNAP. Security advisory for vpnfilter malware. [Online]. Available: https://www.qnap.com/en/security-advisory/nas-201805-24

[72] FBI. Foreign cyber actors target home and office routers and networked devices worldwide. [Online]. Available: https://www.ic3.gov/media/2018/180525.aspx

[73] M. Security. (2017, May) Wannacrypt ransomware worm targets out-of-date systems. [Online]. Available: https://www.microsoft.com/security/blog/2017/05/12/wannacrypt-ransomware-worm-targets-out-of-date-systems/

[74] M. Field. (2018, Oct) Wannacry cyber attack cost the nhs £92m as 19,000 appointments cancelled. [Online]. Available: https://www.telegraph.co.uk/technology/2018/10/11/wannacry-cyber-attack-cost-nhs-92m-19000-appointments-cancelled/

[75] Z. Clark. (2017, May) The worm that spreads wanacrypt0r. [Online]. Available: https://blog.malwarebytes.com/threat-analysis/2017/05/the-worm-that-spreads-wanacrypt0r/

[76] GReAT. (2017, May) Wannacry ransomware used in widespread attacks all over the world. [Online]. Available: https://securelist.com/wannacry-ransomware-used-in-widespread-attacks-all-over-the-world/78351/

[77] MalwareTech. (2017, May) How to accidentally stop global cyber attacks. [Online]. Available: https://www.malwaretech.com/2017/05/how-to-accidentally-stop-a-global-cyber-attacks.html

[78] A. McBride. (2017, May) The hours of wannacry. [Online]. Available: https://umbrella.cisco.com/blog/the-hours-of-wannacry

[79] CheckPoint. (2017, May) Wannacry – new kill-switch, new sinkhole. [Online]. Available: https://blog.checkpoint.com/2017/05/15/wannacry-new-kill-switch-new-sinkhole/

[80] K. Logic. (2017, May) Wannacry: Two weeks and 16 million averted ransoms later. [Online]. Available: https://www.kryptoslogic.com/blog/2017/05/wannacry-two-weeks-and-16-million-averted-ransoms-later/

[81] Great. (2017, June) Schroedinger's pet(ya). [Online]. Available: https://securelist.com/schroedingers-petya/78870/

[82] L. Mathews. (2017, August) Notpetya ransomware attack cost shipping giant maersk over $200 million. [Online]. Available: https://www.forbes.com/sites/leemathews/2017/08/16/notpetya-ransomware-attack-cost-shipping-giant-maersk-over-200-million/#48a445fc4f9a

[83] ABC. (2017, July) Cyber attack: Ukrainian software company will face charges over security neglect, police suggest. [Online]. Available: https://www.abc.net.au/news/2017-07-03/cyber-attack-charge-ukarine/8675006

[84] M. O. David Maynor, Aleksandar Nikolic. (2017, July) The medoc connection. [Online]. Available: https://blog.talosintelligence.com/2017/07/the-medoc-connection.html

[85] L. Labs. (2017, July) Notpetya technical analysis. [Online]. Available: https://gallery.logrhythm.com/threat-intelligence-reports/notpetya-technical-analysis-logrhythm-labs-threat-intelligence-report.pd

[86] M. D. A. R. Team. (2017, June) New ransomware, old techniques petya adds worm capabilities. [Online]. Available: https://www.microsoft.com/security/blog/2017/06/27/new-ransomware-old-techniques-petya-adds-worm-capabilities/

[87] I. Gofman. (2017, October) Advanced threat analytics security research network technical analysis: Notpetya. [Online]. Available: https://www.microsoft.com/security/blog/2017/10/03/advanced-threat-analytics-security-research-network-technical-analysis-notpetya/

[88] K. Sood and S. Hurley. (2017, June) Notpetya technical analysis – a triple threat: File encryption, mft encryption, credential theft. [Online]. Available: https://www.crowdstrike.com/blog/petrwrap-ransomware-technical-analysis-triple-threat-file-encryption-mft-encryption-credential-theft/

[89] C. Osborne. (2021, May) Colonial pipeline attack: Everything you need to know. [Online]. Available: https://www.zdnet.com/article/colonial-pipeline-ransomware-attack-everything-you-need-to-know/

[90] C. Pipeline. (2021, May) Media statement update: Colonial pipeline system disruption. [Online]. Available: https://www.colpipe.com/news/press-releases/media-statement-colonial-pipeline-system-disruption

[91] C. on Homeland Security. (2021, June) Cyber threats in the pipeline: Using lessons from the colonial ransomware attack to defend critical infrastructure. [Online]. Available: https://homeland.house.gov/activities/hearings/cyber-threats-in-the-pipeline-using-lessons-from-the-colonial-ransomware-attack-to-defend-critical-infrastructure

[92] I. X.-F. Megan Roddie. (2021, May) Darkside malware profile. [Online]. Available: https://exchange.xforce.ibmcloud.com/collection/06d0917405c36ca91f5db1fe0c01d1ad

[93] A. Kleymenov. (2021, May) Colonial pipeline ransomware attack: Revealing how darkside works. [Online]. Available: https://www.nozominetworks.com/blog/colonial-pipeline-ransomware-attack-revealing-how-darkside-works/

[94] BSI. (2014, December) Die lage der it-sicherheit in deutschland 2014. [Online]. Available: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/Lageberichte/Lagebericht2014.pdf__blob=publicationFile

[95] R. M. Lee, M. J. Assante, and T. Conway, "German steel mill cyber attack," *Industrial Control Systems*, vol. 30, p. 62, 2014.

[96] M. D. Abrams and J. Weiss. (2008, August) Malicious control system cyber security attack case study: Maroochy water services, australia. [Online]. Available: https://www.mitre.org/publications/technical-papers/malicious-control-system-cyber-security-attack-case-study-maroochy-water-services-australia

[97] D. of Justice. (2016, March) Seven iranians working for islamic revolutionary guard corps-affiliated entities charged for conducting coordinated campaign of cyber attacks against u.s. financial sector. [Online]. Available: https://www.justice.gov/opa/pr/seven-iranians-working-islamic-revolutionary-guard-corps-affiliated-entities-charged

[98] (2012, September) Denial of service attacks against u.s. banks in 2012–2013. [Online]. Available: https://www.cfr.org/cyber-operations/denial-service-attacks-against-us-banks-2012-2013

[99] D. Yadron. (2015, Dec) Iranian hackers infiltrated new york dam in 2013. [Online]. Available: https://www.wsj.com/articles/iranian-hackers-infiltrated-new-york-dam-in-2013-1450662559

[100] S. G. Tracy Connor, Tom Winter. (2015, December) Iranian hackers claim cyberattack new york dam. [Online]. Available: https://www.nbcnews.com/news/us-news/iranian-hackers-claim-cyber-attack-new-york-dam-n484611

[101] Verizon. (2016) Data breach digest. scenarios from the field. [Online]. Available: https://www.ndia.org/-/media/sites/ndia/meetings-and-events/divisions/cfam/past-events/2016-august/cfam-forum-slides---verizon-data-breach-digest.ashx?mod=article_inline

[102] Vericlave. (2018) Vericlave – the kemuri water company hack. [Online]. Available: https://www.vericlave.com/wp-content/uploads/2018/10/Vericlave_WhitePaper_KemuriWater_1018_F.pdf

[103] D. Moore, V. Paxson, S. Savage, C. Shannon, S. Staniford, and N. Weaver, "Inside the slammer worm," *IEEE Security & Privacy*, vol. 1, no. 4, pp. 33–39, 2003.

[104] K. Poulsen, "Slammer worm crashed ohio nuke plant net," *The Register*, vol. 20, 2003.

[105] E. Levy, "The making of a spam zombie army. dissecting the sobig worms," *IEEE security & privacy*, vol. 1, no. 4, pp. 58–59, 2003.

[106] D. Goodin, "Elecrical supe charged with damaging california canal system," *The Register, November*, vol. 30, 2007.

[107] S. Gorman, "Electricity grid in us penetrated by spies," *The wall street journal*, vol. 8, 2009. [Online]. Available: https://www.wsj.com/articles/SB123914805204099085

[108] B. Bencsáth, G. Pék, L. Buttyán, and M. Félegyházi, "Duqu: A stuxnet-like malware found in the wild," *CrySyS Lab Technical Report*, vol. 14, pp. 1–60.

[109] DHS. (2013) Targeted cyber intrusion detection and mitigation strategies. [Online]. Available: https://www.us-cert.gov/ics/tips/ICS-TIP-12-146-01B

[110] ——. Recommended practices. [Online]. Available: https://us-cert.cisa.gov/ics/Recommended-Practices

[111] P. Nespoli, D. Papamartzivanos, F. G. Mármol, and G. Kambourakis, "Optimal countermeasures selection against cyber attacks: A comprehensive survey on reaction frameworks," *IEEE Commun. Surv. Tutorials*, vol. 20, no. 2, pp. 1361–1396, 2018. [Online]. Available: https://doi.org/10.1109/COMST.2017.2781126

[112] T. K. Das, S. Adepu, and J. Zhou, "Anomaly detection in industrial control systems using logical analysis of data," *Computers Security*, vol. 96, p. 101935, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167404820302121

[113] H. R. Ghaeini, M. Chan, R. Bahmani, F. Brasser, L. Garcia, J. Zhou, A.-R. Sadeghi, N. O. Tippenhauer, and S. Zonouz, "Patt: Physics-based attestation of control systems," in *22nd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2019)*. Chaoyang District, Beijing: USENIX Association, Sep. 2019, pp. 165–180. [Online]. Available: https://www.usenix.org/conference/raid2019/presentation/ghaeini

[114] S. Potluri, C. Diedrich, and G. K. R. Sangala, "Identifying false data injection attacks in industrial control systems using artificial neural networks," in *2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2017, pp. 1–8.

[115] J. H. Castellanos, D. Antonioli, N. O. Tippenhauer, and M. Ochoa, "Legacy-compliant data authentication for industrial control system traffic," in *International Conference on Applied Cryptography and Network Security*. Springer, 2017, pp. 665–685.

[116] S. East, J. Butts, M. Papa, and S. Shenoi, "A taxonomy of attacks on the dnp3 protocol," in *International Conference on Critical Infrastructure Protection*. Springer, 2009, pp. 67–81.

[117] D. Jin, D. M. Nicol, and G. Yan, "An event buffer flooding attack in dnp3 controlled scada systems," in *Proceedings of the 2011 Winter Simulation Conference (WSC)*. IEEE, 2011, pp. 2614–2626.

[118] I. Darwish, O. Igbe, O. Celebi, T. Saadawi, and J. Soryal, "Smart grid dnp3 vulnerability analysis and experimentation," in *2015 IEEE 2nd International Conference on Cyber Security and Cloud Computing*. IEEE, 2015, pp. 141–147.

[119] I. Darwish, O. Igbe, and T. Saadawi, "Experimental and theoretical modeling of dnp3 attacks in smart grids," in *2015 36th IEEE Sarnoff Symposium*. IEEE, 2015, pp. 155–160.

[120] N. Rodofile, K. Radke, and E. Foo, "Real-time and interactive attacks on dnp3 critical infrastructure using scapy," 2015.

[121] J. A. Crain and S. Bratus, "Bolt-on security extensions for industrial control system protocols: A case study of dnp3 sav5," *IEEE Security & Privacy*, vol. 13, no. 3, pp. 74–79, 2015.

[122] B. P. Miller, L. Fredriksen, and B. So, "An empirical study of the reliability of unix utilities," *Communications of the ACM*, vol. 33, no. 12, pp. 32–44, 1990.

[123] R. Amoah, S. Camtepe, and E. Foo, "Formal modelling and analysis of dnp3 secure authentication," *Journal of Network and Computer Applications*, vol. 59, pp. 345–360, 2016.

[124] P. Huitsing, R. Chandia, M. Papa, and S. Shenoi, "Attack taxonomies for the modbus protocols," *International Journal of Critical Infrastructure Protection*, vol. 1, pp. 37–44, 2008.

[125] T. H. Morris, B. A. Jones, R. B. Vaughn, and Y. S. Dandass, "Deterministic intrusion detection rules for modbus protocols," in *2013 46th Hawaii International Conference on System Sciences*. IEEE, 2013, pp. 1773–1781.

[126] W. Gao and T. H. Morris, "On cyber attacks and signature based intrusion detection for modbus based industrial control systems," *Journal of Digital Forensics, Security and Law*, vol. 9, no. 1, p. 3, 2014.

[127] S. Bhatia, N. Kush, C. Djamaludin, A. Akande, and E. Foo, "Practical modbus flooding attack and detection," in *Proceedings of the Twelfth Australasian Information Security Conference (AISC 2014)[Conferences in Research and Practice in Information Technology, Volume 149]*. Australian Computer Society, Inc., 2014, pp. 57–65.

[128] R. Nardone, R. J. Rodríguez, and S. Marrone, "Formal security assessment of modbus protocol," in *2016 11th International Conference for Internet Technology and Secured Transactions (ICITST)*. IEEE, 2016, pp. 142–147.

[129] N. Tsalis, G. Stergiopoulos, E. Bitsikas, D. Gritzalis, and T. K. Apostolopoulos, "Side channel attacks over encrypted tcp/ip modbus reveal functionality leaks." in *ICETE (2)*, 2018, pp. 219–229.

[130] C. Parian, T. Guldimann, and S. Bhatia, "Fooling the master: Exploiting weaknesses in the modbus protocol," *Procedia Computer Science*, vol. 171, pp. 2453–2458, 2020.

[131] B. Chen, N. Pattanaik, A. Goulart, K. L. Butler-Purry, and D. Kundur, "Implementing attacks for modbus/tcp protocol in a real-time cyber physical system test bed," in *2015 IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR)*. IEEE, 2015, pp. 1–6.

[132] M. Baud and M. Felser, "Profinet io-device emulator based on the man-in-the-middle attack," in *2006 IEEE Conference on Emerging Technologies and Factory Automation*. IEEE, 2006, pp. 437–440.

[133] J. Akerberg and M. Bjorkman, "Exploring security in profinet io," in *2009 33rd Annual IEEE International Computer Software and Applications Conference*, vol. 1. IEEE, 2009, pp. 406–412.

[134] H. Hui and K. McLaughlin, "Investigating current plc security issues regarding siemens s7 communications and tia portal," in *5th International Symposium for ICS & SCADA Cyber Security Research 2018 5*, 2018, pp. 67–73.

[135] S. Pfrang and D. Meier, "Detecting and preventing replay attacks in industrial automation networks operated with profinet io," *Journal of Computer Virology and Hacking Techniques*, vol. 14, no. 4, pp. 253–268, 2018.

[136] S. Mehner and H. König, "No need to marry to change your name! attacking profinet io automation networks using dcp," in *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*. Springer, 2019, pp. 396–414.

[137] Y. Yang, K. McLaughlin, T. Littler, S. Sezer, B. Pranggono, and H. F. Wang, "Intrusion detection system for iec 60870-5-104 based scada networks," in *2013 IEEE Power Energy Society General Meeting*, 2013, pp. 1–5.

[138] P. Maynard, K. McLaughlin, and B. Haberler, "Towards understanding man-in-the-middle attacks on iec 60870-5-104 scada networks," in *2nd International Symposium for ICS & SCADA Cyber Security Research 2014 (ICS-CSR 2014) 2*, 2014, pp. 30–42.

[139] Y. Yang, H. Jiang, K. McLaughlin, L. Gao, Y. Yuan, W. Huang, and S. Sezer, "Cybersecurity test-bed for iec 61850 based smart substations," in *2015 IEEE Power & Energy Society General Meeting*. IEEE, 2015, pp. 1–5.

[140] M. Kabir-Querrec, S. Mocanu, J.-M. Thiriet, and E. Savary, "A test bed dedicated to the study of vulnerabilities in iec 61850 power utility automation networks," in *2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, 2016, pp. 1–4.

[141] M. G. da Silveira and P. H. Franco, "Iec 61850 network cybersecurity: Mitigating goose message vulnerabilities," 2019.

[142] S. Raza, A. Slabbert, T. Voigt, and K. Landernäs, "Security considerations for the wirelesshart protocol," in *2009 IEEE Conference on Emerging Technologies & Factory Automation*. IEEE, 2009, pp. 1–8.

[143] A. Samaddar, A. Easwaran, and R. Tan, "A schedule randomization policy to mitigate timing attacks in wirelesshart networks," *Real-Time Systems*, vol. 56, no. 4, pp. 452–489, 2020.

[144] J. Wright, "Killerbee: practical zigbee exploitation framework," in *11th ToorCon conference, San Diego*, vol. 67, 2009.

[145] T. Kennedy and R. Hunt, "A review of wpan security: attacks and prevention," in *Proceedings of the international conference on mobile technology, applications, and systems*, 2008, pp. 1–8.

[146] F. Farha and H. Chen, "Mitigating replay attacks with zigbee solutions," *Network Security*, vol. 2018, no. 1, pp. 13–19, 2018.

[147] G. Kambourakis, C. Kolias, D. Geneiatakis, G. Karopoulos, G. M. Makrakis, and I. Kounelis, "A state-of-the-art review on the security of mainstream iot wireless pan protocol stacks," *Symmetry*, vol. 12, no. 4, p. 579, 2020.

[148] R. Grandgenett, R. Gandhi, and W. Mahoney, "Exploitation of allen bradley's implementation of ethernet/ip for denial of service against industrial control systems," in *Proceedings of the 9th International Conference on Cyber Warfare and Security, ICCWS 2014, Academic Conferences and Publishing International Limited, Reading, UK*, 2014, pp. 58–65.

[149] P. Brook, "Ethernet/ip industrial protocol white paper," *IEEE EFTA*, 2001.

[150] D. I. Urbina, J. A. Giraldo, N. O. Tippenhauer, and A. A. Cárdenas, "Attacking fieldbus communications in ics: Applications to the swat testbed." in *SG-CRC*, 2016, pp. 75–89.

[151] F. Tacliad, T. D. Nguyen, and M. Gondree, "DoS exploitation of allen-bradley's legacy protocol through fuzz testing," in *Proceedings of the 3rd Annual Industrial Control System Security Workshop*. ACM, pp. 24–31. [Online]. Available: https://dl.acm.org/doi/10.1145/3174776.3174780

[152] X. Qi, P. Yong, Z. Dai, S. Yi, and T. Wang, "OPC-MFuzzer: A novel multi-layers vulnerability detection tool for OPC protocol based on fuzzing technology," vol. 3, no. 4, pp. 300–305. [Online]. Available: http://www.ijcce.org/index.php?m=content&c=index&a=show&catid=42&id=400

[153] T. Wang, Q. Xiong, H. Gao, Y. Peng, Z. Dai, and S. Yi, "Design and implementation of fuzzing technology for OPC protocol," in *2013 Ninth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*. IEEE, pp. 424–428. [Online]. Available: http://ieeexplore.ieee.org/document/6846668/

[154] M. Puys, M.-L. Potet, and P. Lafourcade, "Formal analysis of security properties on the opc-ua scada protocol," in *International Conference on Computer Safety, Reliability, and Security*. Springer, 2016, pp. 67–75.

[155] L. Roepert, M. Dahlmanns, I. B. Fink, J. Pennekamp, and M. Henze, "Assessing the security of opc ua deployments," *arXiv preprint arXiv:2003.12341*, 2020.

[156] S. McLaughlin, "On dynamic malware payloads aimed at programmable logic controllers," p. 6.

[157] S. McLaughlin and P. McDaniel, "Sabot: specification-based payload generation for programmable logic controllers," in *Proceedings of the 2012 ACM conference on Computer and communications security*, 2012, pp. 439–449.

[158] A. Keliris and M. Maniatakos, "Icsref: A framework for automated reverse engineering of industrial control systems binaries," *arXiv preprint arXiv:1812.03478*, 2018.

[159] S. Senthivel, S. Dhungana, H. Yoo, I. Ahmed, and V. Roussev, "Denial of engineering operations attacks in industrial control systems," in *Proceedings of the Eighth ACM Conference on Data and Application Security and Privacy*, 2018, pp. 319–329.

[160] S. Kalle, N. Ameen, H. Yoo, and I. Ahmed, "Clik on plcs! attacking control logic with decompilation and virtual plc," in *Binary Analysis Research (BAR) Workshop, Network and Distributed System Security Symposium (NDSS)*, 2019.

[161] H. Yoo and I. Ahmed, "Control logic injection attacks on industrial control systems," in *IFIP International Conference on ICT Systems Security and Privacy Protection*. Springer, 2019, pp. 33–48.

[162] N. Govil, A. Agrawal, and N. O. Tippenhauer, "On ladder logic bombs in industrial control systems," in *Computer Security*. Springer, 2017, pp. 110–126.

[163] A. Serhane, M. Raad, R. Raad, and W. Susilo, "PLC code-level vulnerabilities," in *2018 International Conference on Computer and Applications (ICCA)*. IEEE, pp. 348–352. [Online]. Available: https://ieeexplore.ieee.org/document/8460287/

[164] R. Spenneberg, M. Brüggemann, and H. Schwartke, "PLC-blaster: A worm living solely in the PLC," p. 16.

[165] L. A. Garcia, F. Brasser, M. H. Cintuglu, A.-R. Sadeghi, O. Mohammed, and S. A. Zonouz, "Hey, my malware knows physics! attacking PLCs with physical model aware rootkit," in *Proceedings 2017 Network and Distributed System Security Symposium*. Internet Society. [Online]. Available: https://www.ndss-symposium.org/ndss2017/ndss-2017-programme/hey-my-malware-knows-physics-attacking-plcs-physical-model-aware-rootkit/

[166] D. Beresford, "Exploiting siemens simatic s7 PLCs," p. 26.

[167] J. Klick, S. Lau, D. Marzin, J.-O. Malchow, and V. Roth, "Internet-facing plcs-a new back orifice," *Blackhat USA*, pp. 22–26, 2015.

[168] H. Wardak, S. Zhioua, and A. Almulhem, "PLC access control: a security analysis," in *2016 World Congress on Industrial Control Systems Security (WCICSS)*, pp. 1–6.

[169] A. Keliris, C. Konstantinou, and M. Maniatakos, "Ge multilin sr protective relays passcode vulnerability," *Black Hat USA*, 2017.

[170] A. Ayub, H. Yoo, and I. Ahmed, "Empirical study of PLC authentication protocols in industrial control systems," p. 15.

[171] P. Krishnamurthy, F. Khorrami, R. Karri, D. Paul-Pena, and H. Salehghaffari, "Process-aware covert channels using physical instrumentation in cyber-physical systems," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 11, pp. 2761–2771, 2018.

[172] D. Tychalas and M. Maniatakos, "Special session: Potentially leaky controller: Examining cache side-channel attacks in programmable logic controllers," in *2020 IEEE 38th International Conference on Computer Design (ICCD)*, pp. 33–36, ISSN: 2576-6996.

[173] CODESYS. Release codesys v35 sp16. [Online]. Available: https://www.codesys.com/news-events/news/article/release-codesys-v35-sp16-1.html

[174] D. Tychalas, A. Keliris, and M. Maniatakos, "Stealthy information leakage through peripheral exploitation in modern embedded systems," *IEEE Transactions on Device and Materials Reliability*, vol. 20, no. 2, pp. 308–318, 2020.

[175] A. Robles-Durazno, N. Moradpoor, J. McWhinnie, and G. Russell, "Waterleakage: A stealthy malware for data exfiltration on industrial control systems using visual channels," in *2019 IEEE 15th International Conference on Control and Automation (ICCA)*. IEEE, 2019, pp. 724–731.

GEORGIOS MICHAIL MAKRAKIS received the B.Sc. degree in Information and Communication Systems Engineering from the University of the Aegean, Greece in 2018, specialising in cybersecurity and software engineering. He holds commercial experience as Software Engineer. Currently he is pursuing a Ph.D. degree in Computer Science (CS) at University of Idaho. His research areas of interests involve the Intrusion Detection Systems for IoT, Industrial Control Systems and wireless networks.

CONSTANTINOS KOLIAS is an Assistant Professor with the Department of Computer Science, University of Idaho. Before that, he was a Research Assistant Professor with the Department of Computer Science (CS), George Mason University. He is active in the design of intelligent IDS with a special interest in privacy preserving distributed IDS. His main research interests include security and privacy for the IoT and critical infrastructures. Other areas of interest include mobile and wireless communications security, and privacy enhancing techniques for the Internet.

GEORGIOS KAMBOURAKIS is a full Professor at the dept. of Information and Communication Systems Engineering, University of the Aegean, Greece. He has served as the head of the dept. from Sept. 2019 to Oct. 2019, and was the director of Info-Sec-Lab from Sept. 2014 to Dec. 2018. Currently, Georgios is on unpaid leave from the University, while he is working for the European Commission at the European Joint Research Centre (JRC), Ispra, VA, Italy. His research interests are in the fields of mobile and wireless networks security and privacy, VoIP security, IoT security and privacy, DNS security, and security education, and he has more than 150 refereed publications in the aforementioned areas. More info at: http://www.icsd.aegean.gr/gkamb

CRAIG RIEGER Ph.D., P.E., is the Chief Control Systems Research Engineer and a Directorate Fellow at the Idaho National Laboratory (INL), pioneering interdisciplinary research in next generation resilient control systems. The grand challenge provided an integrated research strategy to address the cognitive, cyber-physical challenges of complex control systems into self-aware, trust-confirming, and threat-resilient architectures. In addition, he has organized and chaired thirteen co-sponsored symposia and one National Science Foundation workshop in this new research area and authored more than 45 peer-reviewed publications. Craig received B.S. and M.S. degrees in Chemical Engineering from Montana State University in 1983 and 1985, respectively, and a PhD in Engineering and Applied Science from Idaho State University in 2008. Craig's PhD coursework and dissertation focused on measurements and control, with specific application to intelligent, supervisory ventilation controls for critical infrastructure. Craig is a senior member of IEEE and has 20 years of software and hardware design experience for process control system upgrades and new installations. Craig has also been a supervisor and technical lead for control systems engineering groups having design, configuration management, and security responsibilities for several INL nuclear facilities and various control system architectures.

JACOB BENJAMIN is Director of Professional Services, at the industrial cyber security company Dragos, Inc. Prior to joining Dragos, Dr. Benjamin was a nuclear cybersecurity researcher at Idaho National Laboratory. He has substantial cybersecurity experience with operational technology at domestic and international critical infrastructures.

• • •