




Article

# Preliminaries of Orthogonal Layered Defence Using Functional and Assurance Controls in Industrial Control Systems

Mike Mackintosh <sup>1,†</sup>, Gregory Epiphaniou <sup>2,\*,†</sup> , Haider Al-Khateeb <sup>2</sup> , Keith Burnham <sup>2</sup>, Prashant Pillai <sup>2</sup> and Mohammad Hammoudeh <sup>3</sup> 

<sup>1</sup> Barhale Plc, Barhale House, Bescot Crescent, Wallsall WS1 4NN, UK; mike.mackintosh@barhale.co.uk

<sup>2</sup> Wolverhampton Cyber Research Institute (WCRI), School of Mathematics and Computer Science, University of Wolverhampton, Wolfruna Building, Wolverhampton WV1 1LF, UK; H.Al-Khateeb@wlv.ac.uk (H.A.-K.); K.Burnham@wlv.ac.uk (K.B.); p.pillai@wlv.ac.uk (P.P.)

<sup>3</sup> School of Computer, Mathematics and Digital Technology, Manchester Metropolitan University, Manchester M15 6BH, UK; m.hammoudeh@mmu.ac.uk

\* Correspondence: g.epiphaniou@wlv.ac.uk; Tel.: +44-190-023-1416

† These authors contributed equally to this work.

Received: 30 December 2018; Accepted: 27 January 2019; Published: 14 February 2019



**Abstract:** Industrial Control Systems (ICSs) are responsible for the automation of different processes and the overall control of systems that include highly sensitive potential targets such as nuclear facilities, energy-distribution, water-supply, and mass-transit systems. Given the increased complexity and rapid evolution of their threat landscape, and the fact that these systems form part of the Critical National infrastructure (CNI), makes them an emerging domain of conflict, terrorist attacks, and a playground for cyberexploitation. Existing layered-defence approaches are increasingly criticised for their inability to adequately protect against resourceful and persistent adversaries. It is therefore essential that emerging techniques, such as orthogonality, be combined with existing security strategies to leverage defence advantages against adaptive and often asymmetrical attack vectors. The concept of orthogonality is relatively new and unexplored in an ICS environment and consists of having assurance control as well as functional control at each layer. Our work seeks to partially articulate a framework where multiple functional and assurance controls are introduced at each layer of ICS architectural design to further enhance security while maintaining critical real-time transfer of command and control traffic.

**Keywords:** Industrial Control Systems; SCADA; Critical National Infrastructure

## 1. Introduction

A significant number of industries focus their activities and business in process automation and system control. Most of these activities are controlled by Industrial Control Systems (ICSs) with a key role and applications in nuclear facilities, manufacturing, and the Critical National infrastructure (CNI) [1]. As technology progressed, these systems did not seem to have developed system security or remedial action plans as required by legal and regulatory compliance. These systems have also become more reliant on conventional IT technology to facilitate communications leading to vulnerabilities and several attack vectors from the integration of IT operations with physical components [2]. The secure inclusion of these systems is still an open issue, with large-scale parallel computations over the Cloud being another factor influencing the security of these systems. There is now a growing concern about the safety and security of CNI components controlled by ICS due to the increased frequency of reactive malware attacks in such infrastructures transforming them into an emerging platform

for multistage cyberattacks, terrorism, and crime with direct impact on the physical domain [3]. Typical components that can be found in an ICS environment vary greatly from basic sensors and actuators to highly complex devices called Programmable Logic Controllers (PLCs). These devices use a variety of operational technology, including pneumatic, hydraulic, mechanical, and electrical. There are numerous manufacturers of these components all over the world, and there are varying standards that have been deployed. The underlying complexity and heterogeneity of the infrastructure almost exponentially increase its attack surface. Defence in depth is a strategy that often seeks to delay adversarial actions against an infrastructure by increasing the complexity and resilience of that infrastructure and allow time for detection response [4]. Many organizations employ defence in depth measures, particularly within information-technology infrastructures. However, they do not apply it to ICS operations. The legacy equipment used in these infrastructures has been traditionally considered as hack-proof due to its separation from the IT infrastructure/physical protection measures in place. Unfortunately, co-ordinated cyberattacks have proved how vulnerable critical infrastructure ICS is to protect its assets. Defence in depth often deploys specific controls to counter and neutralise security risks while employed as a holistic approach for cyber-resilience on all assets [5]. It takes under consideration the interconnection and dependencies between assets and available resources to provide adequate protection against security risks. The relationship between vulnerabilities and controls against operations, personnel, and technology that makes up ICS environments is a crucial factor to be considered in order to apply in-depth defensive measures.

This shifting paradigm shows the significant effect of the cyberspace as a key component in the safety and security of these close linked embedded systems. The inherently “closed” nature of these systems offered a certain degree of immunity against attacks over many years in the past. The security implications related to their core operation and communication with other legacy systems were not explored in-depth, and systemwide security was not regarded as a priority. Most of these systems have recently developed links and connections to the public infrastructure, with their threat landscape continually evolving [6]. Specific changes related to energy-sector deregulation and privatisation seems to have further contributed to concerns around secure ICS composition properties [7]. ICS environments are fundamentally different from conventional IT systems, further complicating the process of securing them. These differences between ICS and IT environments include the product-lifecycle period. For ICS devices, that is up to 15 years, whereas IT systems tend to be replaced every three to five years. Availability and contingency requirements are also higher for ICS due to the real-time nature of their operation. That makes it difficult for patching, updating, and replacement cycles to take place as frequently as in IT systems. Finally, protocol stacks in ICS, such as Modbus and DNP3, are usually bespoke and tailored in a given context of operation without the unnecessary overhead associated with IT protocol stacks [8].

The composition of the overall control system consists of different ICSs components, each of which is designed and deployed for the exact environment within which it operates. The expected outputs from the processes also play a crucial role in the design of these components. The geographical dispersion of these systems also adds to the complexity of the two main subsystems, namely, Supervisory Control and Data Acquisition (SCADA) systems and Distributed Control Systems (DCS), dictating deployment requirements [9,10]. Typical SCADA deployments span over large geographical locations, whereas DCS are usually factory-based. The sensitivity and criticality of these systems are often linked to the facility they control. Nuclear facilities, for example, are regarded as highly sensitive and potentially dangerous facilities with different risk appetites and thresholds. The systematic migration from analogue to digital controls and the integration of so-called “smart” technologies (e.g., smart grids [11], Intelligent Transport Systems [12], smart cities [13]), seems to increase further risk sensitivity in ICS environments [14]. An additional factor that contributes to increased security implications is the convergence of ICS Operational technology and Enterprise IT architectures. That signifies an over-reliance on IT for core security operations at both tactical and

operational levels. The discussed factors and ever-increasing costs to implementing defences in legacy ICS systems have rendered the protection of these systems highly problematic [15].

The remainder of this paper is structured as follows: Section 2 describes the ICS components and key differences between ICS and IT environments. Section 3 presents the key threats, attack vectors, and defences for ICSs, with emphasis on their impact and defence complexities. In Section 4 we analyse the exploits and defence strategies currently identified in ICS and the gaps with regards to the static controls against adaptive attack vectors against these environments. In Section 5, we present our framework in which functional and assurance controls are introduced in the different architecture layers of an ICS using orthogonality. Finally, Section 6 concludes our work.

## 2. ICS Components

A typical ICS environment includes a vast range of components from sensors and actuators to PLCs with different operational technologies applied in this ecosystem. The supply chain is also geographically dispersed with a variety of standards applied from production to distribution. The key categories of ICS components are: (1) Intelligent Electronic Devices (IEDs), (2) Remote Terminal Units (RTUs), (3) PLCs, (4) Master Terminal Units (MTUs), (5) Human–Machine Interfaces (HMI), (6) Data Historians, and (7) Input/Output (I/O) Servers [16]. One of the most basic forms of preprogrammed and often embedded software is firmware (ROM or EPROM). This software operates at the component level and enables cross-component communication as their primary function. Although the supplier usually releases firmware updates, the means to deliver and deploy these updates to the device are often susceptible to attacks. In several cases, the firmware-patching process is delayed or postponed to enable continuous processes to complete their cycles. These patches are usually bug- and security-related fixes, with significant implications to the secure and safe operation of these devices [15].

### 2.1. Software

Bespoke software usually runs on various components in ICSs, tailored to their operations, environment, and models. The dynamic nature of PLCs and their reliability and flexibility have increased their adaptation across several industries in the last forty years. As the integration of automated technology becomes the norm for industrial manufacturers, PLCs have several benefits over traditional hardware products [17]. These systems promise to facilitate more straightforward integration with existing networking platforms and further simplify their deployment. Software-based PLCs can reduce production costs and enhance security and safety with strict authorisation principles embedded in their core operation. Several guidelines under The International Electrotechnical Commission (IEC) were published to standardise programming languages in PLCs in IEC 61131 [18]. This set of guidelines has been widely revised and adopted with explicit references to object-oriented programming, standard data types, and acceptable programming processes, such as Ladder Diagrams (LD), Function Block Diagrams (FBD), and Sequential Function Charts (SFC). In LD, graphic symbols are used in a “ladder-type” formation that constitutes a core PLC programming language. Signal-flow lines are used in FBD, very similar to electric circuits, and vastly applied to DCSs. Sequential behaviour is defined as using SFC as a high-level language that originated from Petri-Net analysis. SFC is considered flexible regarding its integration with other languages. Textual languages are a separate category that includes Instruction List (IL) and Structured Text (ST). The architecture of the PLC software model is defined in Part 3 of the IEC 1131. There is often a distinction made between the notion of a configuration (high-level) and resources needed in the processing environment (low-level) [19,20].

### 2.2. ICS vs. IT Environments

The main difference between the two systems is that the IT environment essentially manages data, while the ICS manages the physical environment [21]. Over time, the advancement of communication technologies combined with a reduction in costs have attracted ICS to embrace the IT communication

technologies. This has introduced new risks within the ICS environment, and has potentially severe ramifications with regard to safety. The significant differences between the two environments are focused on performance-, availability-, risk-, and operation-related metrics [16]. The criticality of real-time communication is paramount in an ICS, whereas high throughput is usually a fundamental requirement of an IT system. Emergency responses are also more critical in an ICS environment due to potential safety and environmental impacts. Planned outages within an IT environment are generally acceptable within agreed parameters. However, ICSs require high availability and often need to continuously run. This makes it imperative that exhaustive testing is carried out, and to sometimes have fully redundant systems available. In the IT environment, the most significant risk is to data confidentiality, integrity, and availability, whereas ICS risks mostly revolve around system safety and availability, with an integrated risk-management approach fundamentally important to its operations [22]. This makes fault tolerance essential for ICSs, with punitive measures for noncompliance to government regulations that impact safety or the environment. System operation in an ICS environment is made more complicated due to the high number of proprietary operating systems. Upgrades are more complicated and need to be carefully implemented. Finally, system memory in ICS components is usually very restrictive compared with IT components.

### 3. Threats and Defence Technologies in Industrial Control Systems

Current practice to implement cyber defence is mainly scoped on vulnerability assessment rather than threat modelling, which includes risk identification and evaluation. This assumes a potential attack vector, an actor with malicious intent, and an opportunity to exploit an existing vulnerability [23]. There are three main approaches to perform risk assessment: qualitative, quantitative, or a hybrid scheme incorporating the first two types. While quantitative risk assessment estimates risk based on a numerical estimation of probabilities, a qualitative approach could mark risk as low, medium, or high. Therefore, qualitative risk analysis is more suitable for recognising the wider implication of hazards, and can be demonstrated and modelled using Petri Nets that can be utilised to analyse risk. Petri Nets are developed based on the calculated risk metrics together with the process function of the ICS or SCADA systems. This analysis method emphasises finding the preconditions under which various modes of failure would occur. Additionally, it helps to determine the associated consequences for such failure. Hence, outcomes are crucial for decision makers in an ICS environment when developing and implementing policies related to risk assessment and governance [24]. For a substantial evaluation of potential risk associated with the exploitation of vulnerabilities of an ICS, each system layer should be considered. However, the performance of the systematic procedures for conducting this assessment in a production environment can be limited by the criticality of the system to the strategic business continuity plan. As a workaround, testing is almost exclusively performed in testbeds or discrete laboratory environments.

ICS environments are complex by design, which presents technical challenges when replicating system's behaviour in an emulator. For example, scalability is a feature required to import the complete set characteristics of the tested physical system. In many cases, both new and legacy protocols and components exist and should be planned for. This introduces complexity constraints and costs that could limit the testing methodology to testbeds based on software-only simulations. Software-based simulations run with a margin of error as they cannot account for all various states of a cyber-physical system. It is also acknowledged that the accuracy of software-based testbeds requires validation and extensive verification of the accuracy of the results. A cost-effective approach that has gained popularity is the use of Hardware-In-The Loop Testbeds (HITL) as it reduces costs while maintaining efficiency [25].

#### 3.1. Hardware-in-the-Loop Testbeds

This type of testbeds can be utilised to examine physical hardware components (e.g., PLC). The I/O interface between the testbed and a component is a Serial Interface Board (SIB) that creates a

loop with the host computer. The SIB is responsible for converting digital to analogue signals (and vice versa) to facilitate communication between the two parties (the host and the component). Additionally, the host includes an emulator to simulate the process and system dynamics. This hybrid approach, combining both physical and virtual elements, provides better accuracy to how the several layers are represented. Figure 1 shows a schematic example of a typical HITL. The number of real-time simulators supporting HITL has increasingly been utilised to support the development of laboratory experiments. Furthermore, they can be used to perform penetration testing and therefore evaluates the efficiency of proposed countermeasures. It is inevitable to expect good testbeds to support a wide range of hardware components in addition to a variety of software packages. Hardware-in-the-loop testbeds have recently been utilised by the USF Smart Grid Power System Laboratory to evaluate several mitigation strategies against cyberattacks to the power grid and other particular energy-management schemes [26].

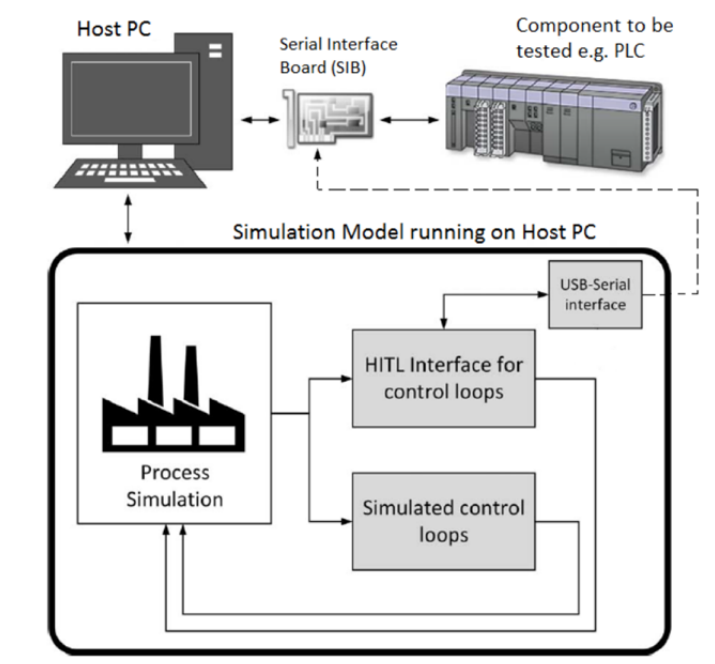


Figure 1. Typical Hardware-In-The Loop (HITL) testbed configuration [25].

### 3.2. Virtual SCADA (VSCADA) Testbeds

Facilitating testing in a virtual environment, this type of testbeds is a software-based solution. For a VSCADA testbed to be effective, the following design objectives should be met [27]:

1. Scalability: You should be able to have a scalable environment while simultaneously supporting interaction with multiple users.
2. Standardisation: The environment's virtual model must integrate various communication protocols, such as Modbus and DNP3.
3. Reconfigurability: The environment's ability to simulate several network topologies and physical devices must be maintained.
4. Effective virtualisation: The environment must provide an accurate representation of the physical system. This typically covers all main behavioural characteristics.

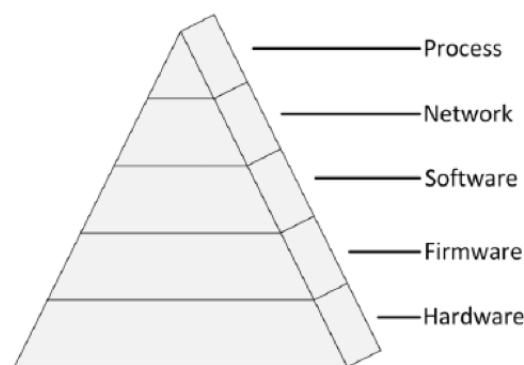
Authors in Reference [27] analysed VSCADA by separating between the back-end and the front-end of the infrastructure. The back-end is mainly responsible for generating data and processing system events to simulate the topology, while front-end infrastructure covers tasks such as the replication of control functions. A good example for such a testbed is the physical 1:87 scale minicity created by the SANS Institute. It has several represented ICS including mass-transit and



electrical-distribution systems. The application of this model ranges from regular training and testing to supporting military exercises [28]. Furthermore, some methods help to profile cyberattackers based on their knowledge, motivation, and technical resources [29]. Researchers could rely on the success rates of these profiles to accordingly model their testbeds. The National SCADA Testbed (NSTB) was developed to enhance the built-in security of the energy-supply systems by the U.S. Office of Electricity Delivery and Energy Reliability. The facility provides a suitable environment to perform research on threat detection and system-vulnerability assessment in SCADA systems within the energy sector.

### 3.3. ICS Architecture Layers

The architecture of the ICS is dissected into several layers to develop a specialised understanding of the overall system. This approach allows for detailed vulnerability assessment to be performed, which is inevitable to gain a better understanding of the problem and develop better cyberdefence techniques. The different layers of an ICS are demonstrated in Figure 2. A layered ICS model was developed in Reference [25] to look beyond vulnerability assessment to a holistic and more detailed approach. This helps to apply a defence-in-depth strategy that is especially useful since these layers contain interconnected functionality. Therefore, the consequences of exploiting an element within a layer affect others. For example, actual electronic components, such as microprocessors and microcontrollers, are represented within the hardware layer of a given ICS. This layer facilitates access (command and control) to all layers above. An accurate input/output process is therefore fundamental for the system to work. Threat modelling for actual devices recovers two main attack vectors: firstly, when the device is compromised by attackers while working in an ICS; secondly, when an actor in the supply chain (e.g., the manufacturer) deliberately compromises the device. Attacking an operational environment to target a device requires a lot of resources to perform the required level of reconnaissance and footprinting, in addition to building intelligence around the device and the system in which it operates [25].



**Figure 2.** Model of the Architecture Layers of an Industrial Control System (ICS) [25].

Attacks against the system's integrity can occur at any stage, as early as the product-design, chip-manufacturing, or supply-chain stage. Spoofing attacks can involve the modification of the actual chip by altering Boolean gates, bridging wires, or inserting additional buffers. When a backdoor is deployed to allow remote access and elevate privilege, the term used to describe this attack vector is Hardware Trojan. This allows access to further confidential data, such as encryption keys. Remote access with elevated privilege gives the ability to maintain access by installing more malicious software. Denial of service is another threat, for instance, a time bomb could be utilised to degrade performance or disable the device. Attackers tend to adopt transparent techniques to cause minimal modifications and avoid Intrusion Detection Systems (IDS) [25].

Side-channel attacks imposes another form of threat where small integrated circuits are attacked through wireless or power channels [30]. These attacks exploit electromagnetic emanations or even

acoustic vibrations to eavesdrop to information leaked from the device. Cryptographic algorithms can also be vulnerable to side-channel attacks by means of differential or simple power analysis. Some attackers manipulate selected values computed by the hardware component during normal operation by deliberately injecting errors. This is known as a Fault Injection attack and it aims to affect the performance of the device. Examples include tampering with environmental operating temperature, power supply, or even the device's clock [25].

The firmware layer is located between the hardware and software layers and acts as a bridge between the two. This layer supports the execution and functionality of compiled programmes. The low and powerful level of control at this layer makes it very attractive for attacks aimed at an ICS. An example of a prominent attack at this layer can be initiated by reverse-engineering the firmware; this attempt, if successful, recovers the underpinning code and provides insight into the core functionality of the device, including potential weaknesses and the opportunity to exploit it [31]. Further, the attackers would usually search for protocols and authentication techniques that have not been properly designed or implemented. Another attack is buffer overflow if a relevant vulnerability is identified as a result of bad programming practices. Reverse-engineering firmware exposes the system to injection attacks of malicious code at a low level. Below, we describe three known steps to perform reverse-engineering [25]:

1. Firmware image acquisition: the attacker acquires a copy of the firmware code. Acquisition can either be a copy obtained directly from the manufacturer or by utilising the Joint Test Action Group (JTAG) testing port to extract the code.
2. Binary analysis: Techniques such as "binwalk" and binary differentials can be used by the attacker to learn information about the used encoding, file systems, and checksum values.
3. Binary disassembly: Further analysis, such as extracting and studying ASCII strings from the firmware, is performed to understand firmware functionality.

PLCs within an ICS environment are known to be a popular target. Therefore, it is recommended to have a policy to keep the running firmware on these devices patched and updated. Further to the benefit of closing all known security flaws, such a policy could reduce the number of existing bugs affecting the usability side of the software [25]. That being said, there are challenges to be accounted for before planning or enforcing such policies. For example, some environments, due to reasons related to business continuity, are required to be running 99.999% of the time, which equates to a maximum downtime of under 6 min of per annum. This is problematic because firmware patches produced by vendors are regularly released and often require a device reboot. In practice, this is one of the factors to explain why patches are applied late, if applied at all, in a year's time. Before the Internet era, there was an argument around the efficiency of implementing a security-by-obscurity strategy, where many organisations took their chances running unpatched software since vulnerabilities were not published [32]. However, the way we are interconnected today and the new era of common vulnerabilities and exposure databases being regularly updated means that the probability of discovered vulnerabilities being exploited is a question of time.

The software layer contains the functionality responsible for translating human input and passing it through to the machine, and vice versa. Therefore, attackers on this level focus on HMI, and any available terminals controlling the ICS processes. The software can either be off-the-shelf (offered commercially to many clients) or bespoke and specifically designed for the ICS. This affects the type of attack vectors to assess; bespoke software is not widely tested while off-the-shelf and could open a wide range of attack surfaces, from kernel exploits, command injections, to known vulnerabilities reported by the professional community. HMIs and workstations on an ICS platform could also be exploited online via attacks on web browsers through Cross-Site Scripting (XSS), drive-by-download, malware, and other techniques. Additionally, the infamous zero-day exploits pose a prominent threat and can remain undetected to software developers for long periods of time [33]. PLCs are also targeted because they are widely used in ICS and SCADA systems. The presence of numerous security

implications at this layer was attributed to the complexity of the software running on PLCs, and the lack of highly skilled programmers working in this area and having sufficient training in cybersecurity. Attackers armed with the internal knowledge of the system can modify PLC operation. The Stuxnet worm is a recent and widely cited example of a cyberwarfare weapon exploiting a vulnerability at the software layer [34]. Compromising the software layer gives access to the legitimate control flow of the ICS to alter underpinning technologies. Similarities between software at this layer compared to toolkits in a typical ICT environment provides opportunities to utilise existing tools, either directly or after certain amendments to perform reconnaissance, footprinting, and eventually gain access to the ICS. Freely available and extremely powerful penetration-testing toolkits include Kali Linux and ParrotSec [35].

ICSs can be distributed across large distances, and such a design includes a central master system managing remote locations over several communication channels. The availability element is therefore critical, and mitigation against denial-of-service attacks becomes a priority. Attack vectors in this case depend on factors including the type of used pipeline-monitoring systems, signal (analogue and digital), and communication protocols. The protocols that the MTUs and RTUs share must be the same for the system to function, and it is possible to find existing implementations utilising governing standards developed prior to introducing the relevant International Standards Organisation (ISO) standards [36]. Two widely used protocols in SCADA systems are Modbus and DNP3, and they both have numerous known vulnerabilities. Therefore, many bespoke protocols operating at the lower two layers of the above model are not published to add a layer of security, strategically known as security by obscurity. However, since security was not considered when these protocols were first developed, they remain insecure by design. Since the risk of a cyberattack continues to increase, next-generation protocols are developed based on common information models, such as IEC 61850, which improves security.

The process layer comes next in the ICS architecture. It controls the processing logic within the ICS and governs the limits that have been programmed into the system. Cyberattacks could modify variables related to the control logic that changes process states. This could halt the system (denial of service). To avoid detection, attackers ensure that no component is operating beyond its acceptable limits [25]. The Stuxnet worm, which targeted a uranium-enriching plant in Iran, attacked the process layer. Stuxnet modified specific Dynamic-Link Library (DLL) files, and the attack vector is believed to have been a mobile storage device. The payload was able to manipulate the PLCs that controlled spin speed to eventually destroy several of the centrifuges. The vulnerability was within Siemens Step7 software that was running on Windows computers to program the PLCs. A rootkit was utilised to keep the infection hidden from system administrators for as long as possible [37].

#### 4. ICS Exploits and Defence Strategies

Extensive work has been done in Reference [25] to identify threats in each layer of the ICS model, with a detailed summary in Tables 1 and 2. Following this work, a taxonomy of these threats and exploits, where applicable, was discussed in Reference [38]. Their work provides a more granular and holistic approach to these threats and vulnerabilities, and underpins the key elements to an informed defence-in-depth strategy in the ICS concept of operations. In addition, empirical evidence suggests that multistage cyberattacks tend to be more successful [28]. This would necessitate further a comprehensive defence strategy across all layers of the ICS. The first fully operational emulated cyberattacks were conducted by the U.S. Department of Homeland Security under the codename "Aurora". The fully functional generator was connected to the power grid, successfully attacked, and physically destroyed using classified exploits [39]. The weaponisation of this space has led to the creation of an ecosystem, whereby organised crime rings can develop, reverse-engineer, and sell ICS crimeware and exploits to the highest bidder [40]. These attacks are devastating, as violations to the electronic space have an adverse effect to the physical space.



**Table 1.** Taxonomy of known threats.

Taxonomy of Known Threats			
Layer	Threat Category	Threats	Typical Examples of Known Exploits
Software	Memory attacks, control-flow attacks.	Modified software behaviour, buffer/stack overflows.	Data historian data compromised; buffer overflow attacks; software operation modified to perform unwanted actions.
	Web attacks.	Open ports on firewall cross-side scripting, SQL injection, database attacks.	Reconnaissance of network to compromise devices; rootkits installed; privileges elevated; data confidentiality, integrity, and availability compromised; data stolen; denial-of-service (DOS) attacks; Smurf attacks; spyware; malware viruses, installed trojans.
	Access Control	No privilege segregation, credentials stolen.	Unauthorised access, privilege escalation.
	Zero-day vulnerabilities.	Inherent flaw in software exploited unbeknown to the vendor, exploited by attackers' Kernel, design flaws; misconfigurations.	Zero-day attacks; viruses, worms, Trojans installed; buffer overflows; replay attacks; rootkits installed; privileges elevated; data compromised; spyware.
Network	Firewall misconfiguration.	Firewall rules incorrectly configured, open ports.	Backdoors inserted; logic or time bombs installed; data/information stolen; sniffers installed.
	Access control.	Unauthorised physical access, unauthorised logical access, Wi-Fi access points and communications.	Network traffic sniffed to steal credentials; man-in-the-middle attacks; malware installed from portable media; insider attacks; untrained employees subjected to phishing attacks.
	Protocol vulnerabilities.	Control signals modified, cryptographic attacks, communication hijacking and spoofing, communication stack attacks, Modbus and DNP3 vulnerabilities, covert channels exploited, replay attacks.	Syn/Ack attacks and flooding; fragmentation attacks; replay attacks; man-in-the-middle attacks; DOS attacks; bypassing controls; eavesdropping; traffic analysis; tunnelling; false command and control communications; exploits of TCP/IP stack; UDP port attacks; Smurf attacks; idle scans (e.g., Nmap); ARP spoofing; chain/loop attacks.
Process	Internet-facing threats.	Similar to web attacks above cryptographic attacks.	Man-in-the-middle attacks; Cinderella attacks; DDOS attacks; rootkits installed; masquerading; spying and sniffers.
	Process-aware malware.	Malware specifically designed to compromise/alter a specific process, sabotage/terrorism.	Stuxnet worm altered Programmable Logic Controller (PLC) operation.
	False-data injection.	Control logic-modified process variables and constants modified	Aurora vulnerability; modified state estimation in power grids.
	Automatic payload generation.	Malicious payload delivered to PLCs, Remote Terminal Units (RTUs), and Master Terminal Units (MTUs).	PLCs exploited, process hijacked.

**Table 2.** Taxonomy of known threats (Cont.).

<b>Taxonomy of Known Threats (Cont.)</b>			
<b>Layer</b>	<b>Threat Category</b>	<b>Threats</b>	<b>Typical Examples of Known Exploits</b>
<b>Hardware</b>	Hardware trojans.	Modification by: design architects, manufacturers, supply chain prior to operation, attacks during normal operation, stack-smashing, exceeding fixed memory allocation.	Backdoors for remote attacks; time-bombs; elevation of privileges; access to cryptographic keys; access to higher layers; degradation of performance; destruction of component or device.
	Fault-injection attacks.	Faults injected. Computational results modified, low-accuracy fault injection—modification of operating environment without operator aware. Ion beams causing bits to flip.	Device performance degraded; modification of operating temperatures; device clock modified; dataset points modified; false data passed to the controllers and data historian.
	Side-channel attacks.	Wireless snooping, electromagnetic emanations, acoustic vibrations, simple power analysis.	Information leakages to extract information such as cryptographic keys.
<b>Firmware</b>	Firmware reverse-engineering.	Firmware image acquisition, binary analysis, binary disassembly cryptographic attacks.	Directly from manufacturer; from the JTAG testing port to obtain firmware, using “binwalk” and binary differentials to reverse-engineer firmware/ASCII strings analysed and disassembled/communications decrypted.
	Firmware vulnerabilities.	Firmware updates and patches not applied, legacy systems with unsupported firmware.	Security vulnerabilities exploited and components compromised, higher layers compromised.
	Firmware modifications	Firmware modified to perform illegal processes.	Component malfunction/destruction.

Certain classes of controls have been introduced within the ICS environment to mitigate risks and ensure information security and safety. Senior management are responsible for setting up and monitoring the classes of controls as appropriate. These have been categorised as: Security Assessment and Authorisation, Planning and Auditing, Overall Risk Assessment (evaluation and analysis), System and Services Acquisition, and Programme Management. A detailed description and the subcomponents of each category can be found in Reference [41]. Physical security is also an important factor to ICS facilities for all local and remote stations with the aim to further reduce the risks against all assets in the environment. Assets include both tangible and intangible, including data specific to the ICS. Robust access controls and monitoring systems should be in place with different assigned authorisation levels and clearance as dictated in Reference [41]. De facto role-based access control (RBAC) is often used as a means to logically control-operator access in both the physical and electronic domain within the ICS operations. RBAC application has been proven a complex process that can often lead to disconnection between high-level policies and the implementation of physical controls. In addition, there are instances where additional discretionary access may need to be granted to certain individuals. It is therefore recommended that careful consideration be made on the two aspects of high-level access and low-level physical-system access [42].

#### *4.1. Traffic Encryption*

Encryption algorithms and protocols are essential to ICS to assure both confidentiality and integrity. These algorithms should also be compatible with existing and emerging authentication protocols, suitable in this environment for all different types of deployed sensors and actuators. Conventional cryptographic algorithms pose significant performance degradation in systems where real-time processes must be executed in a secure and safe manner. Recent protocols, such as IEC 62351-1 and AGA-12, have introduced delays due to asymmetric applied cryptography limiting their usage in specific scenarios within the ICS environment. Lightweight asymmetric encryption is an open issue in these environments, with several approaches introduced in the public domain [43]. The scope of the IEC 62351-1 is to provide a guided service for information security for power-system control operations. The main focus of the standard is not only to specify standardisation for communication protocols, but also to explicate various aspects of information security as applied to power-system operations. Technical specifications for the used communication protocols are explicitly defined in IEC 62351 to 62351-6, whereas end-to-end information security with a focus on enhancing overall management of the communication networks supporting power systems is described in IEC 62351-7. The standard contains provisions to ensure integrity, authenticity, and confidentiality for different protocols used in power systems [44].

#### *4.2. Intrusion Detection and Prevention*

Real-time transmissions in ICS can often be a penalty factor to the accuracy and speed of Intrusion Detection and Prevention systems (IDPS). Certain branches of mathematics in machine-learning approaches can support the process of building new models of behaviour within ICS, and better inform detection processes as a whole. The topology and synergies within the ICS network dictates the adaptation of these systems to better detect and stop attacks against ICS components [45]. IDPS components from sensors to management consoles can be connected via the actual ICS network or through a management network completely isolated from the production systems. The positioning of these sensors, the amount of data logged or collected, and modifications to the existing environment have to be considered prior to deployment. The whole process must be governed by a comprehensive risk assessment only for IDPS deployment within the ICS network [46].

#### *4.3. Disaster Recovery and Business-Continuity Planning*

Increased complexity and interdependencies in an ICS network mean that a problem in an area might have rapid and often unexpected impact on a different area within the same environment.

ICS environments are part of a broader CNI where all parts of the supply chain are interconnected, and many requirements must also be integrated. That environment creates additional technical risks with regard to business-continuity management, where impact is more likely to be greater and more frequent. ICS in particular is a clear manifestation in which recovery-point objectives (PRO) must be zero or near data loss to maintain both their efficacy and purpose. It is therefore fundamentally important to create more efficient continuity strategies and investment to identify areas in which entities are critical to mission achievement in ICS environments. The key standards introduced are the NFPA 1600 family, ISO 22301, ASIS SPC.1-2009, and BS 25999 Parts 1 and 2.

### 5. Orthogonal Defence-in-Depth Framework

Layered protection in ICSs does not only relate to adequate controls in IT infrastructure, as this is not enough to deter or prevent adversaries. Network-related impairment and strict quality requirements are additional factors to be considered when applying security controls in an ICS [47]. An informed defence-in-depth strategy must consider the output of detailed risk and vulnerability assessments, incident management, and support by senior management for all licensed operators in ICS facilities [48]. It is common to conduct iterative risk and vulnerability assessments as the first step of implementing a defence-in-depth framework. Figure 3 illustrates the possible attack surface when IT and ICS environments are integrated whereas Figure 4 illustrates the NIST approach to framing risk. Risk assessments and risk-treatment plans have been embedded in the IT environment for many years, but only recently been introduced into the cyber-physical ICS environment. Authors in Reference [49] detailed the impact and frequency product in the core of this operation, and the factor that underpins controls to be introduced in qualitative, quantitative, and hybrid approaches [50]. Various quantitative and qualitative hybrid models examples exist with typical four-by-four matrices, as shown in in Figure 5, which describe risk categories. The key elements to be considered is the method selected to carry out risk assessments, deployed risk models, and adaptation of risk appetite to properly enumerate the threat landscape in particular cases in which IT and ICS integration takes place. Risk-mitigation plans have recently found their way in the cyber-physical ICS environment [51–53]. The key components of an informed defence-in-depth framework, within which assurance and functional controls could be integrated, are presented as follows.

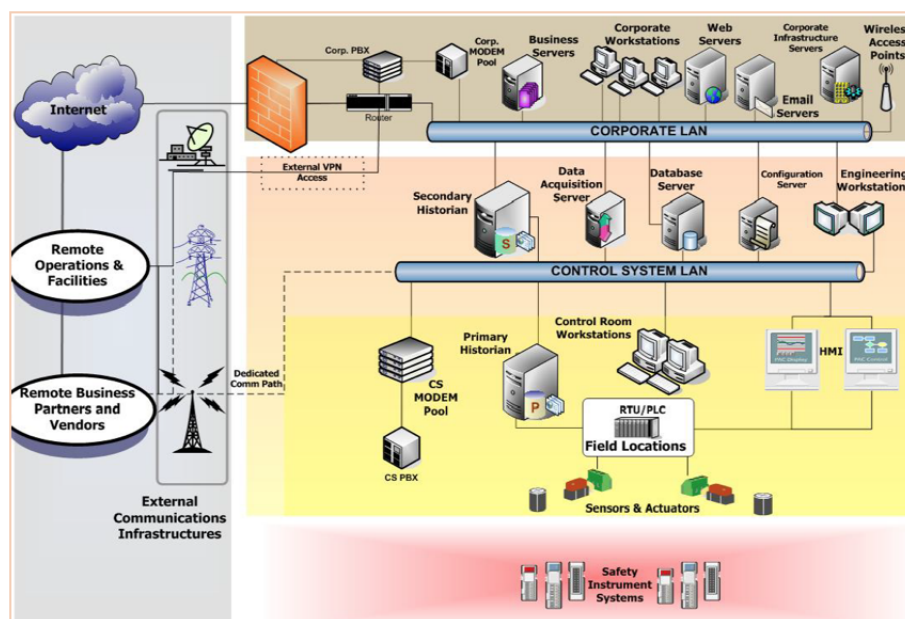


Figure 3. Example of integrated IT and ICS environment (U.S. Department of Homeland Security, 2009).

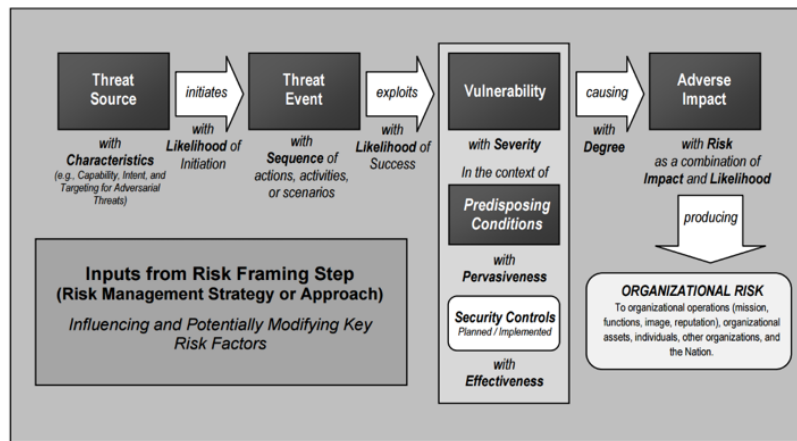


Figure 4. Generic risk model (NIST SP 800-82r2, 2015).

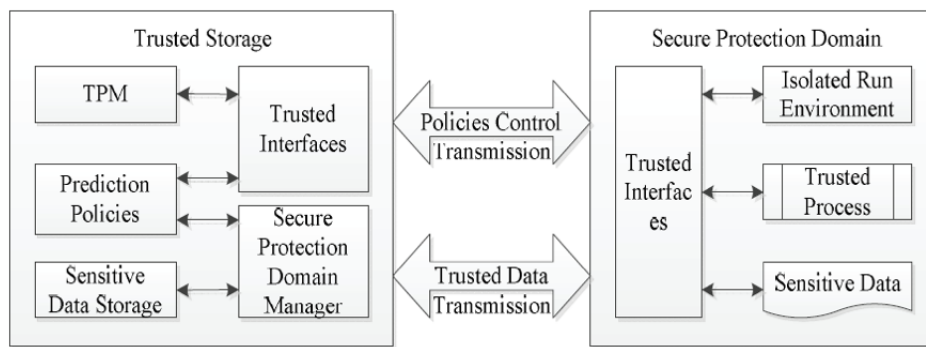
Likelihood		Impact				
Rating	Cost in £000's	Time Impact	Damage to Public Image	Safety Impact		
1	<25%	1	<5	<1 week	Concerning	Minimal
2	<50%	2	<15	<2 weeks	Serious	Minor
3	<75%	3	<50	<3 weeks	Very Serious	Significant
4	100%	4	<100	<6 weeks	Disasterous	Severe

Figure 5. Matrices to assign a quantitative value to likelihood and impact.

### 5.1. Integrated Trusted Protection

The concept of “trusted” protection evolved from several cyber–cybersecurity guidelines published over time by ENISA and the U.S. Department of Homeland Security (DHS). This concept has emerged as a security-hardening approach where reliability and availability are at its core, with recent applications in the energy sector [54]. Trust management is the basis of this concept subdivided into elements, such as the Trusted Computing Platform, Trusted Protection Mechanism, and Trusted Network Management. Within this trust-management environment, TCP control operates at the hardware and firmware levels of the ICS. That seeks to ensure strict authentication and verification techniques throughout the whole lifecycle of an ICS operation. Each process must be carefully monitored, and its credibility must be established via restrictive executions and protection against illegal read commands, impersonation, and fabrication attacks, while adequate intrusion detection and prevention is in place [55]. Specific isolation techniques have been introduced in the literature, very similar to VPN-encrypted tunnels that operate dynamically at the layer of the field devices. Authors in Reference [56] introduced the secure-protection domain (SPD) (Figure 6). Authentication and control policies should also be strictly applied so data flows are limited to trusted domains within the storage area. No untrusted processes should be executed involving sensitive or critical data. Finally, TCN control operates between the cyber–physical layer and the enterprise network, and prevents malicious activities from both internal and external attacks. Enterprise baseline security is used in these scenarios, from monitoring at the application layer up to trusted computing modules [57]. The credibility of nodes is judged on past and current performance upon which models of behaviour are developed to justify thresholds of acceptance. TNM control is aimed at detecting these types of attack where variation in attack strategy and path is manifested. These concepts together tie with the five layers of an ICS or SCADA system, and the development of an enhanced defence-in-depth strategy.





**Figure 6.** Trusted data-protection mechanism [56].

### 5.2. ICS and Incident Management

The ability to rapidly respond to security incidents is another key component of the defence-in-depth strategy. Critical parameters such as real-time availability, safety, and integrity must be hardcoded in the interaction of ICS components and services. A model was introduced in the literature from the decision-making theory to partially address these issues [58]. The model introduces the descriptive, predictive, and prescriptive parts to record incident details and project both development and lateral movement and communication mechanisms. Steps in the process include capturing traffic patterns and behaviours from all ICS components, including sensors and devices. The next step is to use standard Intrusion Detection Message Exchange Format (IDMEF), as defined in the RFC 4765, to formalise threat description and information exchange about severity and compromised subsystems in an ICS environment. Finally, data aggregation is used to build a profile of the attack and lateral movement within the ICS subsystems or systems that have been compromised. The predictive part is considered the correlation phase of the response-decision framework where the impact of the attacks is quantified. The system uses pattern-detection models and algorithms to extract information from raw data and build attack metadata during the correlation phase. Authors in Reference [59] introduced a statistical algorithm based on the Semi-Markov Chain that attempts to quantify the impact of an attack from the number of alerts generated from each component within the infrastructure. Information is then evaluated for those nodes that had already been compromised. As observations change, the state changes and may have a different duration (also called sojourn time) to the previous state. These changes of states allow for the prediction of a future state based on recorded changes in the state over time. The prescriptive portion of the model selects the best course of action with regard to remediating the impact of an attack. This process is not straightforward, as expert knowledge on ramifications for each action is required to achieve the objectives of the defined security policies. A three-step scenario for the prescriptive incident response was introduced in Reference [58] (See Figure 7). The purpose of the decision framework is to transform ICS from a “nonconformity” status during or after an attack to a “conformity status” by applying all corrective actions in a timely manner that effectively decreases Mean Time to Recovery (MTTR) and Mean Time Between Failures (MTBF) to the lowest possible values.

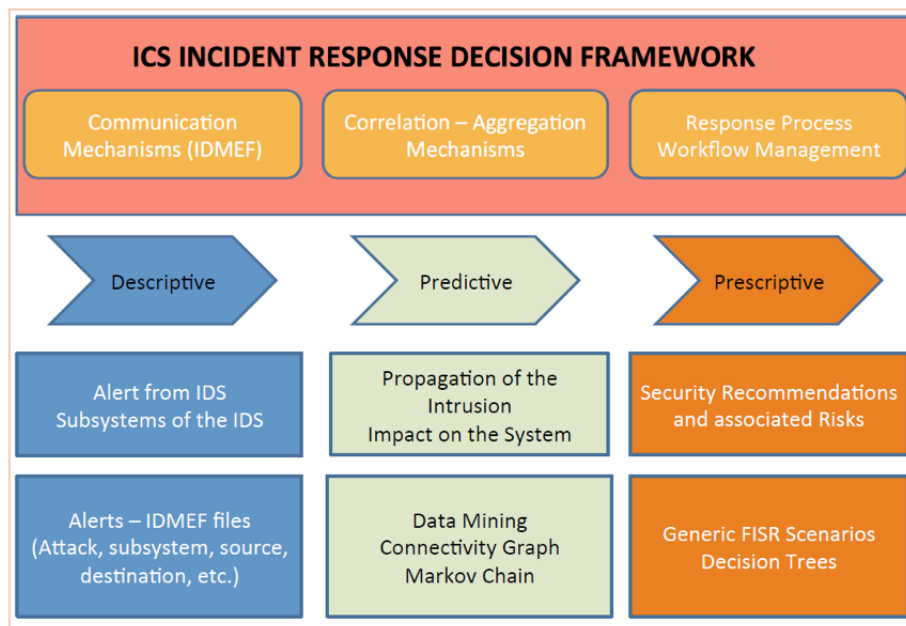


Figure 7. Incident-response decision framework [58].

### 5.3. Orthogonality of Security Controls

Orthogonal security controls have been introduced in the nuclear sector, whereby layered security is imposed in a two-dimensional approach at each layer in ICS environments. The idea is that, at every layer, at least two controls are deployed that are complementary to each other. The orthogonality principle has its roots in linear algebra, spherical trigonometry, Euclidean geometry, and other advanced mathematical approaches, and implies a perpendicular approach. The application of orthogonality in the nuclear facilities has been a clear case in which the controls could be tested in the most secure yet most sought-after target for cyberwarfare due to its significance. Authors in Reference [60] also addressed regulation issues imposed by the U.S. Government in Title 10 of the Codification of Federal Regulation (CFR). Subsection 73.54 requires compliance by the licensed operator of nuclear facilities to protect against security breaches and service and data violations. Approval is needed for all cybersecurity-related plans by the Nuclear Regulatory Commission (NRC), with high assurance always demonstrated to satisfy the terms of operators’ licence. Various templates have been developed to help with compliance using guidance documents such as NRC RG 5.71:2010 and NEI 08-09 Rev.6:2010. In addition, the process of defence in diversity follows the principles of multivendor-protection approach, whereby multiple suppliers are used to enhancing the overall security status of registered assets. The aim is to mitigate weaknesses present in specific products by strengths presented in different ones (NIST SP 800-53r4, 2013). Layered malware defence is an excellent example of such an approach where predictive analytics on top of its deployment can further improve malware defence.

The relationship between vulnerabilities and controls against operations, personnel, and technology that makes up ICS environments is a crucial factor to be considered in order to apply defence in depth measures. The application of layered security with functional assurance metrics in ICS architectures, as presented in our framework, identifies a target intent capability and opportunities against ICS operations, personnel, and technology. This allows for better-informed defence in alignment with standards and controls, and optimisation of policies and procedures, while increasing situational awareness and supply-chain security. The current framework can help policymakers and security personnel to identify evolving categories of threats in an ICS environment and reduce its long-term exposure. Overall security functions are clearly defined and distinguished from information technology based on the requirement that they can impact operation, resilience, and legal and regulatory compliance. This is because of elements such as patch management, change management,

and as a drastically different classification between IT and ICS environments. These differences dictate a drastically different approach to the used testing and auditing methods, the difficulty in forensic auditing processes in ICS environments, and interdependencies of business processes affected by security events. Our framework seeks to further explore the combination of people, technology, operations, and situational awareness from an adversarial point of view of emerging threats to an ICS. The existing framework improves overall resilience by clearly identifying and increasing the cost of intrusion while improving the probability of detection and capability to react while maintaining an explicit mapping against the assurance controls necessary. The framework can be used as a baseline to understand relative security risks to the support infrastructure both in IT and ICS, identify and prioritise process systems that can be affected by security threats, and analyse interconnections and dependencies to better articulate the impact of a security breach to the critical functions and related business processes. By aligning functional and assurance controls in traditional defence-in-depth strategies for ICS in this framework, a better understanding can be established on how compensating controls can be applied for protection in such a complex environment without raising risk to the overall system. Existing challenges with the proposed framework include its integration to existing risk-management practices related to corporate-strategy policies, tactical policies and procedures including guidance and constraints, and feedback with regard to monitoring results.

#### *5.4. Shortfalls of Existing Defence-in-Depth Strategies*

The systematic review of the previously presented strategies demonstrates their limitations to adequately protect against skilful and motivated adversaries. Most security operations are carried out by conventional IT systems operating within the ICS environment. The combination of existing strategies in a holistic and robust defence-in-depth strategy is therefore necessary given the increase in both frequency and sophistication of attacks in recent years. Examples of advanced persistent threats, such as Stuxnet and Baku-Tbilisi-Ceyhan, have manifested their ability to cause significant disruption and impact on infrastructure-supporting facilities. The legal and regulatory compliance for interconnected ICSs with online access is also of paramount importance that seems to be lacking the conformity status required by these systems [15]. Search engines for vulnerable systems, such as SHODAN, have also been used to map ICS equipment in railway signalling and traffic-control systems, posing additional threats to their overall safety and security. It is clear that, since electronic attacks on ICS have a significant impact in the physical domain, the transition on how to better understand how technology, people, and infrastructure are implemented to better protect against adaptive attack vectors becomes a necessity. Geographic distance between different ICS systems also seems to no longer be relevant for cyberdefence [61]. Attackers are now equally active from long distances, with an increased impact on their attacks and, to a certain degree, immune to prosecution from law enforcement due to legal restrictions across countries. The traditional defence-in-depth approach incorporates several elements of layered defence principles and often assumes counterattacks at its core. Counterattacking seems to be unrealistic in the public domain and ICS environments in particular given both the questionable benefit from such an action and strained resources allocated in cyberdefence.

#### *5.5. Enhanced Defence-in-Depth Strategies*

The U.S. NRC proposed a five-security-layer architecture with enhanced defence-in-depth attributes introduced at each layer. The criticality and sensitivity of each system dictate the security boundaries within Layers 3 and 4. These layers allow data flow only in one direction, toward less-secure layers, and provide additional protection to critical mission systems with a certain degree of isolation. All measures and controls must be implemented in a hierarchical fashion in order to be effective [26]. These layers consist of the following components:

1. Stateful firewall-inspection engines preventing access from untrusted zones. Commercially available and regularly updated databases preventing from malware infections using heuristic engines.
2. Use of unidirectional gateways and application proxies to restrict access toward critical and sensitive layers using adequate instructions. and controls specific to ICS and SCADA systems.
3. Use of IDPSs used transparently in the infrastructure.
4. Auditing and logging security events with appropriate log management and automation and orchestration of security events.

Recent developments in research explore the adaptation of Software Defined Networks (SDNs) into ICS and SCADA security. Many legacy protocols, such as ATM and Frame Relay, have been replaced by MPLS and SDN, and offer increased fault-tolerant transmissions and diversity in cyberdefence [62]. Traffic can now be more segregated, and data flows can be decomposed to better map the threat landscape in these environments. Traffic filtering and deep packet inspection (DPI) are also deployed to prevent unauthorised access to the ICS, although research on compatibility and interoperability between ICS and firewall operations is still in its infancy. Industrial ICS firewalls focus on the unique characteristics of ICS protocols and associated data flows that makes them, to certain degree, ineffective with protocols used in traditional IT environments. Bridging between IT and ICS communication is often done by dedicated appliances that segregate the two zones with access control lists independently applied to both. Any forwarding rules are carefully defined and thoroughly tested with several initiatives published in the public domain, such as IndusCAP-Gate [63]. The orthogonality principle can be applied to the original five-layer model of an ICS, including assurance control, available with all appropriate mappings introduced in Tables 3 and 4. Although these mappings are by no means exhaustive, they show how orthogonality could be applied to an ICS and used as the basis of future research directions and focus. The number of assurance controls is limited compared to the functional controls for each layer. That constitutes a significant gap even with applied orthogonality in the traditional concentric model presented by U.S. NRC.

**Table 3.** Applying orthogonality to enhance ICS defence-in-depth strategy.

<b>Enhanced Defence-in-Depth Strategy Using Orthogonality</b>			
<b>Layer</b>	<b>Typical ICS Components</b>	<b>Functional Controls</b>	<b>Assurance Controls</b>
<b>Network</b>	Network devices: modems, routers, firewalls; ICS communication protocols: Modbus, DNP3; standard network protocols: TCP/IP; ICS sublayers: device layer, monitoring and control layer, and management layer.	Vulnerability, risk assessment and treatment plan; encryption of ICS communication traffic., e.g., NTRU; encryption of TCP/IP traffic within VPN tunnels; use of shielded twisted pairs in sensitive areas; physical security preventing unauthorised access; role-based logical access control, staff-vetting procedures to minimise insider attacks; complete ban on removable-storage media; redundant critical network components for fail-over; defence in diversity; mitigation against DOS attacks; secure encryption and authentication of WiFi traffic: firewalls and unidirectional gateways for segregation; port blocking on network switches; establishment of DMZ between ICS and corporate network: VPNs terminate in DMZ; use of honeypots and honeynets.	Testing Environments: HIT1, VSCADA, NSTB, etc.; routine inspections and audit of access and error logs; use of biometrics to validate access; data-integrity checks; routine inspections of events and threat logs on firewalls; routine inspection of logs on honeypots and honeynets; intrusion detection and prevention; trusted network protection assurance; vulnerability and penetration testing; senior-management audits.
<b>Process</b>	MTUs; HMIs; data historian.	Vulnerability, risk assessment and treatment plan; physical security preventing unauthorised access; staff-vetting procedures to minimise insider attacks; ICS incident response and decision trees; disaster recovery and business-continuity planning; dedicated security team; data historian situated within DMZ; environmental-control systems.	Testing environments: HIT1, VSCADA, NSTB, etc.; inspection and audit of access logs; CCTV surveillance of sensitive areas; use of biometrics to validate access; routine testing of DR and business-continuity plans; routine backups of data-historian databases; senior-management audits.



**Table 4.** Applying orthogonality to enhance ICS defence-in-depth strategy (cont.).

<b>Enhanced Defence-in-Depth Strategy Using Orthogonality</b>			
<b>Layer</b>	<b>ICS Typical Components</b>	<b>Functional Controls</b>	<b>Assurance Controls</b>
<b>Hardware</b>	IEDs: sensors, relays, actuators, microprocessors, microcontrollers; RTUs; PLCs; MTUs.	Vulnerability, risk assessment, and treatment plan; in-depth auditing of designers, manufacturers, and supply chain of all hardware; defence in diversity; physical security preventing unauthorised access; screening to prevent electromagnetic emanations; staff-vetting procedures to minimise insider attacks; redundant critical components for fail-over.	Testing environments: HIT1, VSCADA, NSTB., etc.; testing IEDs for hardware trojans; inspection and audit of access logs; CCTV surveillance of sensitive areas; asset tracking; trusted computing platform assurance; senior-management audits.
<b>Firmware</b>	IEDs: sensors, relays, actuators, microprocessors, microcontrollers, etc.; servers and computer BIOS.	Vulnerability, risk Assessment and treatment plan; physical security preventing unauthorised access; staff-vetting procedures to minimise insider threat; confidentiality agreements with architects and manufacturers; routine firmware patching.	Inspection and audit of access logs; CCTV surveillance of sensitive areas; testing environments: HIT1, VSCADA, NSTB, etc.; trusted computing platform assurance; testing of all firmware patches.
<b>Software</b>	PLC programming languages; operating systems.	Vulnerability, risk assessment, and treatment plan; physical security preventing unauthorised access; role-based logical access control; staff-vetting procedures to minimise insider threat; complete ban on removable-storage media; routine software patching; defence in Diversity; reprogramming conducted by highly trained developers; antimalware software.	Testing environments: HIT1, VSCADA, NSTB, etc.; inspection and audit of access logs; testing of all software patches; testing for logic errors and bugs; CCTV surveillance of sensitive areas; intrusion detection and prevention; trusted data-protection mechanism assurance; routine data backup; senior-management audit.

## 6. Conclusions

There is a great deal of research that needs to be conducted, commencing with finding assurance controls at each layer to enhance the orthogonality strategy. In addition, real-time detection techniques need to be improved. Testbeds are another worthwhile research area, as are lightweight encryption technologies for ICS communication protocols. The confidential sharing of information between operators is also a subject that would be worthy of research, but it obviously has significant drawbacks should this information fall into the wrong hands. A standard of trust awarded to manufacturers and the supply chain might be an idea to limit the potential damage of fabrication attacks. From outside observations, the exact state of the ICS industry is difficult to determine, but it is probably only a matter of time before the next major attack is revealed. On the flip-side, the effort to secure systems seems to be gaining momentum. In this work, we presented a partial articulation of a model that deploys orthogonality in both functional and assurance controls in ICSs in an attempt to further normalise the secure integration of these rather legacy-embedded systems with core architectural components influencing and influenced by the cyber-physical domain.

**Author Contributions:** Conceptualization, M.M. and G.E.; methodology, M.M.; validation, G.E., M.M. and H.A.-K.; formal analysis, G.E. and M.M. and H.A.-K.; investigation, M.M. and G.E.; resources, M.H.; writing—original draft preparation, M.M. and G.E.; writing—review and editing, K.B. and P.P.; visualization, M.H.; supervision, G.E.; project administration, G.E.

**Funding:** This research received no external funding.

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

## References

1. Chikuni, E.; Dondo, M. Investigating the security of electrical power systems SCADA. In Proceedings of the AFRICON 2007, Windhoek, South Africa, 26–28 September 2007; pp. 1–7. [[CrossRef](#)]
2. Epiphaniou, G.; Karadimas, P.; Ismail, D.K.B.; Al-Khateeb, H.; Dehghantanha, A.; Choo, K.R. Nonreciprocity Compensation Combined with Turbo Codes for Secret Key Generation in Vehicular Ad Hoc Social IoT Networks. *IEEE Internet Things J.* **2018**, *5*, 2496–2505. [[CrossRef](#)]
3. Orojloo, H.; Azgomi, M.A. Evaluating the complexity and impacts of attacks on cyber-physical systems. In Proceedings of the 2015 CSI Symposium on Real-Time and Embedded Systems and Technologies (RTEST), Tehran, Iran, 7–8 October 2015; pp. 1–8. [[CrossRef](#)]
4. Heo, Y.; Kim, B.; Kang, D.; Na, J. A design of unidirectional security gateway for enforcement reliability and security of transmission data in industrial control systems. In Proceedings of the 2016 18th International Conference on Advanced Communication Technology (ICACT), Pyeongchang, Korea, 31 January–3 February 2016; pp. 310–313. [[CrossRef](#)]
5. Zhang, F.; Kodituwakku, H.A.D.E.; Hines, W.; Coble, J.B. Multi-Layer Data-Driven Cyber-Attack Detection System for Industrial Control Systems Based on Network, System and Process Data. *IEEE Trans. Ind. Inform.* **2019**. [[CrossRef](#)]
6. Venugopalan, V.; Patterson, C.D. Surveying the Hardware Trojan Threat Landscape for the Internet-of-Things. *J. Hardw. Syst. Secur.* **2018**, *2*, 131–141. [[CrossRef](#)]
7. Mosterman, P.J.; Zander, J. Cyber-physical Systems Challenges: A Needs Analysis for Collaborating Embedded Software Systems. *Softw. Syst. Model.* **2016**, *15*, 5–16. [[CrossRef](#)]
8. Fovino, I.N.; Carcano, A.; Murel, T.D.L.; Trombetta, A.; Masera, M. Modbus/DNP3 State-Based Intrusion Detection System. In Proceedings of the 2010 24th IEEE International Conference on Advanced Information Networking and Applications, Perth, Australia, 20–23 April 2010; pp. 729–736. [[CrossRef](#)]
9. Ranathunga, D.; Roughan, M.; Nguyen, H.; Kernick, P.; Falkner, N. Case Studies of SCADA Firewall Configurations and the Implications for Best Practices. *IEEE Trans. Netw. Serv. Manag.* **2016**, *13*, 871–884. [[CrossRef](#)]
10. Agre, J.; Clare, L.; Sastry, S. A Taxonomy of Distributed Real-time Control Systems. *Adv. Comput.* **1999**, *49*, 303–352. [[CrossRef](#)]

11. Anoh, K.; Ikpehai, A.; Bajovic, D.; Jogunola, O.; Adebisi, B.; Vukobratovic, D.; Hammoudeh, M. Virtual Microgrids: A Management Concept for Peer-to-peer Energy Trading. In Proceedings of the 2nd International Conference on Future Networks and Distributed Systems (ICFNDS '18), Amman, Jordan, 26–27 June 2018; ACM: New York, NY, USA, 2018; pp. 43:1–43:5. [[CrossRef](#)]
12. Aldabbas, O.; Abuarqoub, A.; Hammoudeh, M.; Raza, U.; Bounceur, A. Unmanned ground vehicle for data collection in wireless sensor networks: Mobility-aware sink selection. *Open Autom. Control Syst. J.* **2016**, *8*, 35–46. [[CrossRef](#)]
13. Benzerbadj, A.; Kechar, B.; Bounceur, A.; Hammoudeh, M. Surveillance of sensitive fenced areas using duty-cycled wireless sensor networks with asymmetrical links. *J. Netw. Comput. Appl.* **2018**, *112*, 41–52. [[CrossRef](#)]
14. Talari, S.; Shafie-khah, M.; Siano, P.; Loia, V.; Tommasetti, A.; Catalão, J.P.S. A Review of Smart Cities Based on the Internet of Things Concept. *Energies* **2017**, *10*, 421. [[CrossRef](#)]
15. Krotofil, M.; Gollmann, D. Industrial control systems security: What is happening? In Proceedings of the 2013 11th IEEE International Conference on Industrial Informatics (INDIN), Bochum, Germany, 29–31 July 2013; pp. 670–675. [[CrossRef](#)]
16. Stouffer, K.A.; Falco, J.A.; Scarfone, K.A. *Guide to Industrial Control Systems (ICS) Security: Supervisory Control and Data Acquisition (SCADA) Systems, Distributed Control Systems (DCS), and Other Control System Configurations such as Programmable Logic Controllers (PLC)*; SP 800-82; Technical Report; National Institute of Standards and Technology (NIST): Gaithersburg, MD, USA, 2011.
17. Ramanathan, R. The IEC 61131-3 programming languages features for industrial control systems. In Proceedings of the 2014 World Automation Congress (WAC), Waikoloa, HI, USA, 3–7 August 2014; pp. 598–603. [[CrossRef](#)]
18. John, K.H.; Tiegelkamp, M. *IEC 61131-3: Programming Industrial Automation Systems Concepts and Programming Languages, Requirements for Programming Systems, Decision-Making Aids*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 2010.
19. Buciakowski, M.; Witczak, P. A new Matlab coder for generating Structured Text Language from matrix expression for PLC and PAC controllers. *J. Phys. Conf. Ser.* **2017**, *783*, 012062. [[CrossRef](#)]
20. Palma, L.B.; Rosas, J.A.; Pecorelli, J.; Gil, P.S. Simulation of structured text language for PLC programming. In Proceedings of the 2015 3rd Experiment International Conference (exp.at'15), Ponta Delgada, Portugal, 2–4 June 2015; pp. 296–301. [[CrossRef](#)]
21. Ani, U.P.D.; He, H.M.; Tiwari, A. Review of cybersecurity issues in industrial critical infrastructure: manufacturing in perspective. *J. Cyber Secur. Technol.* **2017**, *1*, 32–74. [[CrossRef](#)]
22. Kure, H.I.; Islam, S.; Razzaque, M.A. An Integrated Cyber Security Risk Management Approach for a Cyber-Physical System. *Appl. Sci.* **2018**, *8*, 898. [[CrossRef](#)]
23. Kang, D.; Lee, J.; Kim, S.; Park, J. Analysis on cyber threats to SCADA systems. In Proceedings of the 2009 Transmission Distribution Conference Exposition: Asia and Pacific, Seoul, Korea, 26–30 October 2009; pp. 1–4. [[CrossRef](#)]
24. Henry, M.H.; Layer, R.M.; Snow, K.Z.; Zaret, D.R. Evaluating the risk of cyber attacks on SCADA systems via Petri net analysis with application to hazardous liquid loading operations. In Proceedings of the 2009 IEEE Conference on Technologies for Homeland Security, Waltham, MA, USA, 11–12 May 2009; pp. 607–614. [[CrossRef](#)]
25. Keliris, A.; Konstantinou, C.; Tsoutsos, N.G.; Baiad, R.; Maniatakos, M. Enabling multi-layer cyber-security assessment of Industrial Control Systems through Hardware-In-The-Loop testbeds. In Proceedings of the 2016 21st Asia and South Pacific Design Automation Conference (ASP-DAC), Macau, China, 25–28 January 2016; pp. 511–518. [[CrossRef](#)]
26. Aghamolki, H.G.; Miao, Z.; Fan, L. A hardware-in-the-loop SCADA testbed. In Proceedings of the 2015 North American Power Symposium (NAPS), Charlotte, NC, USA, 4–6 October 2015; pp. 1–6. [[CrossRef](#)]
27. Dayal, A.; Deng, Y.; Tbaileh, A.; Shukla, S. VSCADA: A reconfigurable virtual SCADA test-bed for simulating power utility control center operations. In Proceedings of the 2015 IEEE Power Energy Society General Meeting, Denver, CO, USA, 26–30 July 2015; pp. 1–5. [[CrossRef](#)]

28. Hink, R.C.B.; Goseva-Popstojanova, K. Characterization of Cyberattacks Aimed at Integrated Industrial Control and Enterprise Systems: A Case Study. In Proceedings of the 2016 IEEE 17th International Symposium on High Assurance Systems Engineering (HASE), Washington, DC, USA, 1–7 January 2016; pp. 149–156. [[CrossRef](#)]
29. Pricop, E.; Mihalache, S.F. Fuzzy approach on modelling cyber attacks patterns on data transfer in industrial control systems. In Proceedings of the 2015 7th International Conference on Electronics, Computers and Artificial Intelligence (ECAI), Bucharest, Romania, 25–27 June 2015; pp. SSS-23–SSS-28. [[CrossRef](#)]
30. Tsoutsos, N.G.; Maniatakos, M. Fabrication Attacks: Zero-Overhead Malicious Modifications Enabling Modern Microprocessor Privilege Escalation. *IEEE Trans. Emerg. Top. Comput.* **2014**, *2*, 81–93. [[CrossRef](#)]
31. Basnigh, Z.; Butts, J.; Lopez, J.; Dube, T. Firmware modification attacks on programmable logic controllers. *Int. J. Crit. Infrastruct. Prot.* **2013**, *6*, 76–84. [[CrossRef](#)]
32. Mahboob, A.; Zubairi, J. Intrusion avoidance for SCADA security in industrial plants. In Proceedings of the 2010 International Symposium on Collaborative Technologies and Systems, Chicago, IL, USA, 17–21 May 2010; pp. 447–452. [[CrossRef](#)]
33. Kim, I.; Kim, D.; Kim, B.; Choi, Y.; Yoon, S.; Oh, J.; Jang, J. A case study of unknown attack detection against Zero-day worm in the honeynet environment. In Proceedings of the 2009 11th International Conference on Advanced Communication Technology, Phoenix Park, Korea, 15–18 February 2009; Volume 3, pp. 1715–1720.
34. Langner, R. Stuxnet: Dissecting a Cyberwarfare Weapon. *IEEE Secur. Priv.* **2011**, *9*, 49–51. [[CrossRef](#)]
35. Beecroft, A.J.; Michael, J.B. Passive Fingerprinting of Network Reconnaissance Tools. *Computer* **2009**, *42*, 91–93. [[CrossRef](#)]
36. Daniela, T. Communication security in SCADA pipeline monitoring systems. In Proceedings of the 2011 RoEduNet International Conference 10th Edition: Networking in Education and Research, Iasi, Romania, 23–25 June 2011; pp. 1–5. [[CrossRef](#)]
37. Jie, P.; Li, L. Industrial Control System Security. In Proceedings of the 2011 Third International Conference on Intelligent Human-Machine Systems and Cybernetics, Hangzhou, China, 26–28 August 2011; Volume 2, pp. 156–158. [[CrossRef](#)]
38. Zhu, B.; Joseph, A.; Sastry, S. A Taxonomy of Cyber Attacks on SCADA Systems. In Proceedings of the 2011 International Conference on Internet of Things and 4th International Conference on Cyber, Physical and Social Computing, Dalian, China, 19–22 October 2011; pp. 380–388. [[CrossRef](#)]
39. Cárdenas, A.A.; Amin, S.; Sastry, S. Research Challenges for the Security of Control Systems. In Proceedings of the 3rd Conference on Hot Topics in Security (HOTSEC'08), Berkeley, CA, USA, 29 July 2008; USENIX Association: Berkeley, CA, USA, 2008; pp. 6:1–6:6.
40. Zineddine, M. The dilemma of securing industrial control systems: UAE context. In Proceedings of the 2016 International Conference on Information Technology for Organizations Development (IT4OD), Fez, Morocco, 30 March–1 April 2016; pp. 1–6. [[CrossRef](#)]
41. Jillepalli, A.A.; Sheldon, F.T.; de Leon, D.C.; Haney, M.A.; Abercrombie, R.K. Security management of cyber physical control systems using NIST SP 800-82r2. In Proceedings of the IEEE IWCMC, Valencia, Spain, 26–30 June 2017; pp. 1864–1870.
42. Cheminod, M.; Durante, L.; Seno, L.; Valenzano, A. On the description of access control policies in networked industrial systems. In Proceedings of the 2014 10th IEEE Workshop on Factory Communication Systems (WFCS 2014), Toulouse, France, 5–7 May 2014; pp. 1–10. [[CrossRef](#)]
43. Mohd, B.J.; Hayajneh, T. Lightweight Block Ciphers for IoT: Energy Optimization and Survivability Techniques. *IEEE Access* **2018**, *6*, 35966–35978. [[CrossRef](#)]
44. Schlegel, R.; Obermeier, S.; Schneider, J. Assessing the Security of IEC 62351. In Proceedings of the 3rd International Symposium for ICS & SCADA Cyber Security Research (ICS-CSR '15), Ingolstadt, Germany, 17–18 September 2015; BCS Learning & Development Ltd.: Swindon, UK, 2015; pp. 11–19. [[CrossRef](#)]
45. Jin, C.; Valizadeh, S.; van Dijk, M. Snapshotter: Lightweight intrusion detection and prevention system for industrial control systems. In Proceedings of the 2018 IEEE Industrial Cyber-Physical Systems (ICPS), Saint Petersburg, Russia, 15–18 May 2018; pp. 824–829. [[CrossRef](#)]
46. Nazir, S.; Patel, S.; Patel, D. Assessing and augmenting SCADA cyber security: A survey of techniques. *Comput. Secur.* **2017**, *70*, 436–454. [[CrossRef](#)]
47. Colbert, J.M.E.; Kott, A. *Cyber-Security of SCADA and Other Industrial Control Systems*; Springer: Berlin/Heidelberg, Germany, 2016.

48. anssi. Managing Cybersecurity for Industrial Control Systems. 2016. Available online: [https://www.ssi.gouv.fr/uploads/2014/01/Managing\\_Cybe\\_for\\_ICES\\_EN.pdf](https://www.ssi.gouv.fr/uploads/2014/01/Managing_Cybe_for_ICES_EN.pdf) (accessed on 29 August 2018).
49. StJohn-Green, M.; Piggan, R.; McDermid, J.A.; Oates, R. Combined security and safety risk assessment—What needs to be done for ICS and the IoT. In Proceedings of the 10th IET System Safety and Cyber-Security Conference 2015, Bristol, UK, 21–22 October 2015; pp. 1–7. [CrossRef]
50. Cherdantseva, Y.; Burnap, P.; Blyth, A.; Eden, P.; Jones, K.; Soulsby, H.; Stoddart, K. A review of cyber security risk assessment methods for SCADA systems. *Comput. Secur.* **2016**, *56*, 1–27. [CrossRef]
51. Frey, S.; Rashid, A.; Zanutto, A.; Busby, J.; Follis, K. On the Role of Latent Design Conditions in Cyber-physical Systems Security. In Proceedings of the 2nd International Workshop on Software Engineering for Smart Cyber-Physical Systems (SEsCPS '16), Austin, TX, USA, 16 May 2016; ACM: New York, NY, USA, 2016; pp. 43–46. [CrossRef]
52. Etigowni, S.; Tian, D.J.; Hernandez, G.; Zonouz, S.; Butler, K. CPAC: Securing Critical Infrastructure with Cyber-physical Access Control. In Proceedings of the 32nd Annual Conference on Computer Security Applications (ACSAC '16), Los Angeles, CA, USA, 5–9 December 2016; ACM: New York, NY, USA, 2016; pp. 139–152. [CrossRef]
53. Knowles, W.; Prince, D.; Hutchison, D.; Disso, J.F.P.; Jones, K. A Survey of Cyber Security Management in Industrial Control Systems. *Int. J. Crit. Infrastruct. Prot.* **2015**, *9*, 52–80. [CrossRef]
54. Otuoze, A.O.; Mustafa, M.W.; Larik, R.M. Smart grids security challenges: Classification by sources of threats. *J. Electr. Syst. Inf. Technol.* **2018**. [CrossRef]
55. Bartman, T.; Carson, K. Securing communications for SCADA and critical industrial systems. In Proceedings of the 2016 69th Annual Conference for Protective Relay Engineers (CPRE), College Station, TX, USA, 4–7 April 2016; pp. 1–10. [CrossRef]
56. Wang, J.; Lu, J.; Yang, S.; Li, D. Integrated trusted protection technologies for industrial control systems. In Proceedings of the 2015 IEEE Seventh International Conference on Intelligent Computing and Information Systems (ICICIS), Cairo, Egypt, 12–14 December 2015; pp. 418–423. [CrossRef]
57. Genge, B.; Nai Fovino, I.; Siaterlis, C.; Masera, M. Analyzing Cyber-Physical Attacks on Networked Industrial Control Systems. In *Critical Infrastructure Protection V*; Butts, J., Sheno, S., Eds.; Springer: Berlin/Heidelberg, Germany, 2011; pp. 167–183.
58. He, Y.; Maglaras, L.A.; Janicke, H.; Jones, K. An Industrial Control Systems incident response decision framework. In Proceedings of the 2015 IEEE Conference on Communications and Network Security (CNS), Florence, Italy, 28–30 September 2015; pp. 761–762. [CrossRef]
59. Hu, L.; Wang, Z.; Naem, W. Security analysis of stochastic networked control systems under false data injection attacks. In Proceedings of the 2016 UKACC 11th International Conference on Control (CONTROL), Belfast, UK, 31 August–2 September 2016; pp. 1–6. [CrossRef]
60. Zavarisky, P. High assurance cybersecurity plan templates for nuclear facilities: Two-dimensional layering of mutually orthogonal security controls for a high-assurance cybersecurity protection of critical computer-based systems in the post-Stuxnet era. In Proceedings of the International Conference on Information Society (i-Society 2014), London, UK, 10–12 November 2014; pp. 40–44. [CrossRef]
61. Genge, B.; Haller, P.; Kiss, I. Cyber-Security-Aware Network Design of Industrial Control Systems. *IEEE Syst. J.* **2017**, *11*, 1373–1384. [CrossRef]
62. Genge, B.; Haller, P. A hierarchical control plane for software-defined networks-based industrial control systems. In Proceedings of the 2016 IFIP Networking Conference (IFIP Networking) and Workshops, Vienna, Austria, 17–19 May 2016; pp. 73–81. [CrossRef]
63. Kim, B.; Kang, D.; Na, J.; Chung, T. Abnormal traffic filtering mechanism for protecting ICS networks. In Proceedings of the 2016 18th International Conference on Advanced Communication Technology (ICACT), PyeongChang, Korea, 31 January–3 February 2016; pp. 436–440. [CrossRef]

