CYBER-PHYSICAL SYSTEMS
2020, VOL. 6, NO. 1, 1–32
https://doi.org/10.1080/23335777.2019.1624620

Taylor & Francis
Taylor & Francis Group

ARTICLE

Integrating redundancy, diversity, and hardening to improve security of industrial internet of things

Aron Laszka[a], Waseem Abbas[b], Yevgeniy Vorobeychik[c] and Xenofon Koutsoukos[d]

[a]Department of Computer Science, University of Houston, Houston, TX, USA; [b]Department of Electrical Engineering, Information Technology University, Lahore, Pakistan; [c]Department of Computer Science and Engineering, Washington University in St. Louis, MO, USA; [d]Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN, USA

ABSTRACT

As the Industrial Internet of Things (IIoT) becomes more ubiquitous in critical application domains, such as smart water-distribution and transportation systems, providing security and resilience against cyber-attacks grows into an issue of utmost importance. Cyber-attacks against critical infrastructure pose significant threats to public health and safety. To alleviate the severity of these threats, various security techniques are available, including redundancy, diversity, and hardening. However, no single technique can address the whole spectrum of cyber-attacks that may be launched by a determined and resourceful attacker. In light of this, we consider a multi-pronged approach that integrates redundancy (deploying additional components and devices), diversity (using multiple implementation variants), and hardening (reinforcing individual components) techniques for designing secure and resilient IIoT systems. We introduce a framework for quantifying cyber-security risks and optimizing IIoT design. We show that finding optimal designs is an NP-hard problem, and then present an efficient meta-heuristic algorithm that finds near optimal designs in practice. To demonstrate the applicability of our framework, we present two case studies in water-distribution and transportation systems. Our numerical evaluation shows that integrating redundancy, diversity, and hardening can lead to reduced security risks at the same cost.

ARTICLE HISTORY
Received 30 January 2019
Accepted 23 May 2019

KEYWORDS
Cyber-physical systems (CPS); security; internet-of-things (IoT); resilience; graph theory; economics of security; cyber risks; water distribution; transportation networks

1. Introduction

Remarkable improvements in the operational efficiency, reliability, and overall functionality of present-day industrial systems and critical-infrastructure networks are the payoff of employing modern technology trends such as the Industrial Internet of Things (IIoT). Emerging industrial platforms such as the Industrial Internet (II) in the US and Industrie 4.0 in Europe are creating

CONTACT Aron Laszka laszka.aron@gmail.com University of Houston, Houston, TX, USA

An earlier version of this paper was presented at the 2018 IEEE International Conference on Industrial Internet and was published in its proceedings, http://aronlaszka.com/papers/laszka2018synergistic.pdf

novel systems that include devices, systems, networks, and controls used to operate and/or automate IIoT systems. Improved connectivity between system components and tight coupling between the cyber and physical domains are the characteristic features of these modern systems. On one hand, this enhanced connectivity and integration allow for data exchange and processing to fine-tune system processes. On the other hand, they open new threat channels, against which these systems need to be secured [1–4]. Critical infrastructure such as water management and transportation systems, in particular, have been growing more connected, and they can be targets for malicious attacks. Due to the tight coupling between the cyber and physical domains in such systems, new attack vectors are emerging. Attacks can include physical destruction, network spoofing, malware, data corruption, malicious insiders, and others. To exacerbate the situation, cyber-incidents can easily escalate and propagate to other components and systems due to the high-level of connectivity. The steady increase in the number of reported cyber-incidents evidences how difficult it is in practice to secure such systems against determined and sophisticated attackers.

Securing large-scale IIoT systems against cyber- and cyber-physical attacks is a complicated task since these systems often face a variety of threats, have large attack surfaces, comprise heterogeneous components, may contain a number of undiscovered vulnerabilities, and have constrained resources [1,2,5,6]. As a result, traditional security and resilience mechanisms – ranging from redundant deployments to diversifying and hardening systems components – may be useful but are not sufficient by themselves. In fact, there is clearly no 'silver bullet' technique that could protect such complex systems against the entire broad spectrum of possible attacks. Thus, to provide security solutions that are capable of supporting the continued expansion of such systems, we need a multi-pronged and holistic approach. In other words, instead of relying on a single technique, defenders must employ multi-pronged solutions, which combine multiple techniques for improving the security and resilience of IIoT. We can divide many of existing techniques into three canonical approaches:

- *Redundancy* for deploying additional redundant components in a system, so that even if some components are compromised or impaired, the system may retain normal (or at least adequate) functionality;
- *Diversity* for implementing components using a diverse set of component types (e.g. diverse hardware and software implementations) so that vulnerabilities which are present in only a single type have limited impact on the system; and

- *Hardening* for reinforcing individual components or component types (e.g. tamper-resistant hardware and firewalls), so that they are harder to compromise or impair.

A straightforward way to combine these approaches is to design and implement them independently of each other. However, for the improved security and resilience of IIoT systems, the real benefit of combining these approaches is exhibited when they are integrated and optimized simultaneously in the design. Various defense techniques – if deployed carefully – can complement each other in elevating the ability of a system to resist malicious attacks. Unfortunately, a sound framework and methodology for combining techniques from different approaches is lacking. In lieu of a unified framework or methodology, defenders must follow best practices and intuition when integrating techniques, which can result in the deployment of ineffective – or even vulnerable – combinations.

In this paper, we propose a framework for integrating redundancy, diversity, and hardening techniques for designing secure and resilient IIoT systems. The objective is to develop a systematic framework for prioritizing investments for reducing security risk. The contributions of the paper are as follows:

- Establishing a system model that can capture (1) a wide variety of components that are found in IIoT as well as the interactions between them, (2) a security investment model for redundancy, diversity, and hardening, and (3) a security risk model which quantifies the impact of attacks and defense mechanisms (Section 2).
- Formulating the resilient IIoT design problem as an optimization problem for prioritizing security investments and showing that the problem is NP-hard (Section 3).
- Developing an efficient meta-heuristic design algorithm based on simulated annealing for finding near-optimal designs in practice (Section 3).
- Evaluating the applicability of the approach using two case studies in canonical IIoT domains of water distribution and transportation systems (Sections 4 and 5).

We give an overview of related work in Section 6 and provide concluding remarks in Section 7.

2. Model

An IIoT system is comprised of a variety of components: sensors, controllers, actuators, and human-machine interfaces for interacting with users as shown in Figure 1. Our first step introduces a general system model for evaluating security risk. First, we present a high-level model of IIoT systems. Then, we

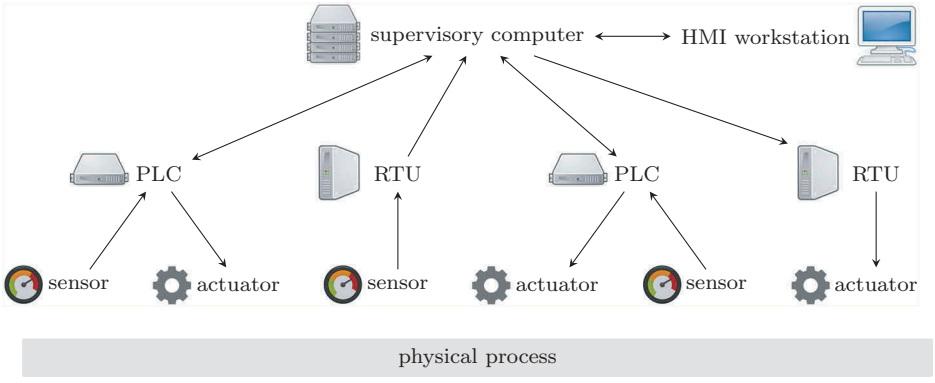

Figure 1. Example cyber-physical system. Arrows represent flows of sensor data and control signals.

introduce a model of security investments in redundancy, diversity, and hardening, and we quantify risks posed by cyber-attacks, considering both probability and impact. Based on this model, we formulate the problem of optimal system design. For a list of symbols used in this paper, see Table 1.

2.1. System model

We model the cyber part of the system as a directed graph $G = (C, E)$. The set of nodes C represents the components of the system, while the set of directed edges E represents connections between the components, which are used to send data and control signals. For each component $c \in C$, we let $O_c \subseteq C$ denote the set of origin components of the incoming edges of component c. Further, we let T_c denote the type of component c, which is one of the following:

Table 1. List of symbols.

Symbol	Description
Constants	
C	set of components
E	set of connections between components
O_c	set of components connecting to component $c \in C$
T_c	type of component $c \in C$
I	set of implementation types
I_c	set of implementation types available for component $c \in C$
R_i	cost of deploying an instance of implementation type $i \in I$
D_i	cost of deploying at least of instance of type $i \in I$
L_i	set of hardening levels available for type $i \in I$
S_l	probability that hardening level $l \in L_i$ is secure
H_l	cost of attaining hardening level $l \in L_i$
Deployment	
r_c	set of implementation types deployed for component $c \in C$
l_i	hardening level chosen for implementation type $i \in I$

- *sensor*: components that measure the state of physical processes (e.g. pressure sensors);
- *actuator*: components that directly affect physical processes (e.g. valves);
- *processing*: components that process and store data and control signals (e.g. PLCs);
- *interface*: components that interact with human users (e.g. HMI workstations).

The implementation of each component is chosen from a set of implementation types. We let I_c denote the set of types that may be used to implement component c, and we let I denote the set of all implementation types that may be used in the system (i.e. $I = \cup_{c \in C} I_c$).

2.2. Security investment model

2.2.1. Redundancy

We model redundancy as deploying multiple instances of the same component. For simplicity, we assume that for each component, at most one instance of each suitable implementation type is deployed.[1] We make this assumption because our goal is to address security risks posed by deliberate attacks, and if a security vulnerability exists in an implementation type, then attackers can typically compromise all instances of that type.

We let $r_c \subseteq I_c$ denote the set of implementation types that are deployed for component $c \in C$. To quantify the cost of redundancy, we let R_i denote the cost of deploying an instance of type $i \in I_c$. Then, the total cost of redundancy is

$$\text{cost of redundancy} = \sum_{c \in C} \sum_{i \in r_c} R_i. \tag{1}$$

Cost R_i captures all costs related to deploying an additional instance of implementation type i, which may include the cost of purchasing hardware (e.g. purchasing a sensor or a computational device), the cost of physical deployment (e.g. installing and configuring an instance), as well as maintenance and management costs associated with operating the deployed instance. In most IIoT systems, redundancy can lead to a straightforward improvement in robustness and security, and the cost of individual instances may be very low compared to the potential impact of a successful cyber-attack. However, for a large number of devices, the total cost can be substantial, especially the net present cost of operating the devices for the lifetime of the system. Therefore, an optimal design must achieve the best trade-off between reducing cyber-security risks and reducing costs (see Section 2.4 for a formal definition of Optimal Design Problem).

2.2.2. Diversity

We model diversity as deploying a diverse set of implementation types (e.g. employing different software or hardware implementations for components that perform the same task). Prior work has shown that software diversity alone may be effective at reducing cyber-risks [7,8]. In our framework, we model diversity as selecting different implementations r_c to be deployed for each component $c \in C$ (or at least attempting to use as many distinct sets as possible).

To quantify the cost of diversity, we let D_i denote the cost of using an implementation type $i \in I$ in any non-zero number of components (i.e. D_i is the cost incurred when the first instance of type i is deployed). Then, the cost of diversity is

$$\text{cost of diversity} = \sum_{i \in \bigcup_{c \in C} r_c} D_i. \tag{2}$$

Cost D_i captures all costs related to employing implementation type $i \in I$, which may include the cost of acquiring a software license, the cost of training personnel to use the implementation type, as well as management and maintenance costs associated with using the implementation type.

While increasing the diversity of implementation types can reduce risks by limiting the impact of a single vulnerability, it is actually 'double-edged sword' that may also elevate risks by broadening the attack surface of the system. In a system that employs more implementation types, the probability that all implementation types are secure is generally lower. Thus, increasing diversity can elevate the probability of the attacker finding an exploitable vulnerability, which our risk model inherently captures (see Section 2.3.1.1). In light of this, an optimal design must strike a balance between limiting the impact of vulnerabilities and reducing the probability of the attacker finding a vulnerability (see Section 2.4 for a formal definition of Optimal Design Problem).

2.2.3. Hardening

We model the hardening of an implementation type as decreasing the probability that a zero-day security vulnerability is discovered by an attacker. We assume that hardening is applied in steps (e.g. performing a code review), resulting in a discrete set of hardening levels.

We let L_i denote the set of hardening levels available for implementation type $i \in I$, and we let l_i denote the chosen level. To model the amount of security provided by hardening level $l \in L_i$, we let S_l denote the probability that the implementation type will be secure (i.e, no zero-day vulnerability is discovered) if level l is chosen. To quantify the cost of hardening, we let H_l denote the cost of attaining level $l \in L_i$. Then, the total cost of hardening is

$$\text{cost of hardening} = \sum_{i \in I} H_{l_i}. \tag{3}$$

Hardening techniques decrease the probability of an attacker finding and exploiting a vulnerability in an implementation type. These techniques range from following secure-coding standards [9] to adding firewall to the services that run on an implementation type [10]. Defenders can also find and patch vulnerabilities by hiring security experts for penetration testing [11] or by outsourcing vulnerability discovery through bug-bounty programs [12,13]. In our framework, a hardening level formally corresponds to employing a particular combination of techniques. For example, the hardening levels for a particular implementation type may be the following: $l_0 = \emptyset$, $l_1 =$ {implementing a secure-coding policy}, $l_2 = l_1 \cup$ {penetration testing by external expert}, $l_3 = l_2 \cup$ {organizing bug-bounty program}. Each one of these discrete choices has its own well-defined cost (e.g. cost of increased development time due to secure coding plus cost of security audit). Estimating the effectiveness of these techniques, which is a key problem in security management and economics [14], may be based on domain experts and historical data. Since hardening techniques can be quite expensive, an optimal design must find the best trade-off between reducing costs and reducing the probability of the attacker finding an unpatched vulnerability. The trade-off between cost and security has been studied extensively in the economics of cyber-security literature [14,15].

2.3. Security risk model

Next, we quantify the risks faced by a system with given redundancy, diversity, and hardening design. In principle, risk can be quantified as

$$Risk = \sum_{outcome} \Pr[outcome] \cdot Impact(outcome). \tag{4}$$

Accurately estimating cybersecurity risks is a central problem in cyber-risk management and, more generally, in the economics of security [14]. Since cybersecurity risks depend on both the impact and likelihood of incidents, we need to be able to accurately estimate both in order to accurately estimate risks. In IT systems, estimating impact is often difficult due to the intangible nature of the protected assets. In IIoT systems, on the other hand, impact can often be expressed in terms of tangible, physical consequences. However, these consequences can typically be estimated only using domain-specific approaches (e.g. vulnerability assessment for transportation networks [16]). In this paper, we present two case studies of our framework, in which we estimate the impact of cyber-attacks on water-distribution and transportation networks (Section 4). Prior work has introduced approaches for a variety of domains (e.g. transactive energy systems [17]); we refer the reader to [18] for

risk assessment methods for SCADA systems. Similar to impact, probability can also be challenging to estimate due to multiple factors, such as the lack of reliable public incident data [19]. In our framework, we describe a detailed model for estimating the probabilities of various outcomes (Section 2.3.1).

In our model, an outcome can be represented as a set of components that have been compromised by an attacker:

$$Risk(r, l) = \sum_{\hat{C} \subseteq C} \Pr[\hat{C} \text{ is compromised}] \cdot Impact(\hat{C}), \tag{5}$$

where $Impact(\hat{C})$ is the amount of loss inflicted on the system by an attacker who has compromised components $\hat{C}$. In the remainder of this subsection, we discuss how to measure $\Pr[\hat{C}\text{iscompromised}]$ and $Impact(\hat{C})$.

2.3.1. Probability

We quantify the probability that an attacker compromises a set of components $\hat{C} \subseteq C$ *implicitly* by describing a probabilistic process that models how an attacker can take control of the components of a system one-by-one. We consider two alternative attack models in our framework: non-stealthy attacks and stealthy attacks. The two attack models are summarized in Table 2.

2.3.1.1. Non-Stealthy Attacks. First, the attacker attempts to find exploitable vulnerabilities in all the implementation types that are deployed in the system. Based on our hardening model, we have that the attacker discovers a zero-day vulnerability in each implementation type $i \in I$ with probability $1 - S_{l_i}$ (independently of the other types). We let $\hat{I}$ denote the set of vulnerable implementation types. Then, the attacker exploits these vulnerabilities to compromise vulnerable instances. We assume that all instances that are implemented using a vulnerable implementation type are compromised. Notice that diversity (i.e. employing a broader set of implementation types, see Section 2.2.2) increases the expected number of vulnerable implementation types. However, diversity also limits the impact of each vulnerability by reducing the number of instances that may be affected by the vulnerability, thereby preventing catastrophic outcomes.

Table 2. Component compromise rules.

	Component Type			
Attack Type	sensor	actuator	processing	interface
non-stealthy attack	if majority of instances are compromised	if majority of instances are compromised or majority of input components are compromised		
stealthy attack	if all instances are compromised	if all instances are compromised or all input components are compromised		

Next, we determine the set of compromised components $\hat{C}$. We begin with sensor components and then address other component types. Each *sensor* component c is compromised if the majority of its instances r_c are vulnerable (i.e. if $|r_c \cap \hat{I}| \geq |r_c|/2$). The rationale behind this rule is the following: If the majority of the instances are compromised and report the same malicious sensor data, then other components cannot simply filter out the incorrect sensor input. If only a minority of the instances are compromised, then other components can easily determine the correct sensor input (e.g. median value [20]).

Finally, we determine which other components are compromised. We start with the set of compromised sensor components $\hat{C}$, and then extend the set $\hat{C}$ in iterations based on the following rule: An *actuator, processing*, or *interface* component c is compromised if the majority of its instances r_c are compromised or if the majority of its inputs are compromised (i.e. if $|O_c \cap \hat{C}| \geq |O_c|/2$). The rationale behind this rule is similar to the rule for sensors. If the attacker compromised only a minority of both the instances and the inputs, then the majority can still provide correct outputs; otherwise, the attacker can tamper with the outputs such that other components cannot recover the correct values. We repeat applying the above rule until the set of compromised components $\hat{C}$ cannot be extended any further.

2.3.1.2. Stealthy Attacks. For stealthy attacks, the process is the same except that 'majority' is replaced in both rules with 'all' (i.e. $|r_c \cap \hat{I}| = |r_c|$ and $|O_c \cap \hat{C}| = |O_c|$). The rationale behind the difference compared to non-stealthy attacks is the following: A tampering attack against a sensor component can easily be detected if there exists at least one component providing a correct sensor value since this value will differ from the tampered ones. Similarly, to tamper with all outputs of a non-sensor component, the attacker needs to compromise all instances or control all inputs.

2.3.2. Impact

We let $Impact(\hat{C})$ denote the financial and physical loss resulting from an attack that compromises and maliciously controls components in $\hat{C}$. The exact formulation of $Impact(\hat{C})$ depends on the system and the characteristics of its physical processes. In this paper, we consider two types of systems, water-distribution and transportation systems, which we will describe in detail in Section 4.

2.4. Optimal design problem

We first formulate the problem with fixed investments in redundancy, diversity, and hardening.

Definition 2.1 (Optimal Design Problem highlight(Fixed Redundancy, Diversity, and Hardening)highlight) Given redundancy, diversity, and hardening investments R, D, and H, an *optimal design* $(\boldsymbol{r}, \boldsymbol{l})$ is

$$\text{argmin}_{r,l}\ Risk(\boldsymbol{r}, \boldsymbol{l}) \tag{6}$$

subject to

$$\forall c \in C : r_c \subseteq l_c \tag{7}$$

$$\forall l \in l : l_i \in L_i \tag{8}$$

$$\sum_{c \in C} \sum_{i \in r_c} R_i \leq R \tag{9}$$

$$\sum_{i \in \cup_{c \in C} r_c} D_i \leq D \tag{10}$$

$$\sum_{i \in l} H_{l_i} \leq H. \tag{11}$$

Next, we introduce a more general formulation, in which we can determine the amounts to invest in redundancy, diversity, and hardening.

Definition 2.2 (Optimal Design Problem) Given security budget B, an *optimal design* $(\boldsymbol{r}, \boldsymbol{l})$ is

$$argmin_{\mathbf{r},\mathbf{l}}\ Risk(\boldsymbol{r}, \boldsymbol{l}) \tag{12}$$

subject to

$$\forall c \in C : r_c \subseteq l_c \tag{13}$$

$$\forall l \in l : l_i \in L_i \tag{14}$$

$$\sum_{c \in C} \sum_{i \in r_c} R_i + \sum_{i \in \cup_{c \in C} r_c} D_i + \sum_{i \in l} H_{l_i} \leq B. \tag{15}$$

3. Computational analysis and meta-heuristic algorithms

Since the number of feasible designs to choose from may be very large even for small systems, finding an optimal design using exhaustive search is computationally infeasible. In light of this, a key question for the practical

application of the proposed framework is whether there exist efficient algorithms for finding optimal or near-optimal designs. We first show that finding an optimal design is computationally challenging by showing that the problem is NP-hard. Then, we introduce an efficient meta-heuristic algorithm that can find a near-optimal solution in polynomial time.

3.1. Computational complexity

The objective of the design problem depends on the impact function, which could be any function, even one that is hard to compute. To show that the design problem is inherently hard (not only due to the potential complexity of computing the impact function), we consider computational complexity assuming a simplistic impact function, whose value is simply the number of compromised components. Formally, we consider $Impact(\hat{C}) = |\hat{C}|$.

To show that the optimal design problem is NP-hard, we first introduce a decision version of the problem.

Definition 3.1 (Optimal Design Problem (Decision Version)) Given security budget B and threshold risk $Risk^*$, determine if there exists a design $(\boldsymbol{r}, \boldsymbol{l})$ such that $Risk(\boldsymbol{r}, \boldsymbol{l}) \leq Risk^*$ and Equations (13), (14), and (15) hold.

We will show that the above problem is NP-hard using a reduction from a well-known NP-hard problem, the Set Cover Problem, which is defined as follows.

Definition 3.2 (Set Cover Problem) Given a set U, a set $\mathcal{F}$ of subsets of U, and a threshold k, find a subset $\mathcal{G} \subseteq \mathcal{F}$ consisting of at most k subsets such that $\mathcal{G}$ covers U (i.e. for every $u \in U$, there exists a $g \in \mathcal{G}$ such that $u \in g$).

Theorem 3.3. *The Optimal Design Problem is NP-hard.*

Proof. Given an instance $(U, \mathcal{F}, k)$ of the Set Cover Problem (SCP), we construct an instance of the Optimal Design Problem (ODP) with stealthy attacks as follows:

- let $C := U$, $E := \emptyset$, and $I := \mathcal{F}$,
- for every $c \in C$, let $T_c := sensor$,
- for every $c \in C$, let $I_c := \{i \in \mathcal{F} \,|\, c \in i\}$,
- for every $i \in I$, let $R_i := 0$,
- for every $i \in I$, let $D_i := 0$,
- for every $i \in I$, let $L_i := \{insecure, secure\}$,
- let $H_{insecure} := 0$ and $S_{insecure} := 0$,
- let $H_{secure} := 1$ and $S_{secure} := 1$,
- let $B := k$ and $Risk^* := 0$.

Clearly, the above reduction can be performed in a polynomial number of steps. It remains to show that the constructed instance of the ODP has a solution if and only if the SCP instance has a solution.

First, suppose that the SCP instance has a solution $\mathcal{G}$. Then, we show that there exists feasible design $(\boldsymbol{r}, \boldsymbol{l})$ that is a solution to the ODP instance. For every component $c \in C$, let $r_c = l_c$. For every implementation type $i \in I$, let $l_i = \mathit{secure}$ if $i \in \mathcal{G}$ (recall that in the construction of the ODP instance, we let the implementation types I correspond to the set of subsets $\mathcal{F}$, and the solution $\mathcal{G}$ is a subset of $\mathcal{F}$) and let $l_i = \mathit{insecure}$ if $i \mathcal{G}$. Clearly, this is a feasible design since its hardening cost is

$$\sum_{i \in I} H_{l_i} = \sum_{i \in \mathcal{G}} H_{\mathit{secure}} \sum_{i \in I \setminus \mathcal{G}} H_{\mathit{insecure}} \tag{16}$$

$$= \sum_{i \in \mathcal{G}} 1 \sum_{i \in I \setminus \mathcal{G}} 0 \tag{17}$$

$$= |\mathcal{G}| \leq k = B, \tag{18}$$

and all other costs are zero. Since $S_{\mathit{secure}} = 0$, implementation types from $\mathcal{G}$ are never vulnerable, and any component c that has at least one secure implementation type (i.e. $I_c \cap \mathcal{G} \neq 0$) is never compromised by a stealthy attack. If $\mathcal{G}$ is a set cover, then there exists at least one secure implementation type $i \in \mathcal{G}$ for each c such that $i \in I_c$, which implies that no component will be compromised. Therefore, $\hat{C} = \emptyset$ is the only possible outcome, which implies that $\mathit{Risk}(\boldsymbol{r}, \boldsymbol{l}) = 0$ as $\mathit{Impact}(\emptyset) = 0$ by definition.

Second, suppose that the ODP instance has a solution $(\boldsymbol{r}, \boldsymbol{l})$. Then, we can show that there exists a solution $\mathcal{G}$ to the SCP instance. Let $\mathcal{G} = \{i \in \mathcal{F} \,|\, l_i = \mathit{secure}\}$ (i.e. the set of implementation types that are secure). Clearly, $\mathcal{G}$ is a feasible solution due to the budget constraint. Next, using an argument that is similar to the one that we used in the previous case, we can show that if $\mathcal{G}$ was not a set cover, then $\mathit{Risk}(\boldsymbol{r}, \boldsymbol{l})$ would be greater than zero. The claim of the theorem then follows from this readily. □

3.2. Meta-heuristic design algorithm

We propose an efficient meta-heuristic algorithm for finding near-optimal designs in practice. Our algorithm is based on simulated annealing, which requires randomly generating feasible solutions that are 'neighbors' of (i.e. similar to) a given solution. Unfortunately, in our solution space (i.e. in the set of designs that satisfy the budget constraints), the feasible neighbors of a solution are not naturally defined. Hence, before we present our meta--heuristic algorithm, we first introduce an alternative representation of feasible designs, which we call design plans.

Definition 3.4 (Design Plan). A *design plan* is a pair $(\boldsymbol{ro}, \boldsymbol{lo})$, where

- $\boldsymbol{ro}$ is a list of component-implementation pairs $(c, i) \in C \times I$ such that $i \in I_c$ holds for every pair $(c, i) \in \boldsymbol{ro}$, and each possible pair (c, i) appears exactly once in $\boldsymbol{ro}$;
- $\boldsymbol{lo}$ is an ordered multiset of implementation types such that each implementation type $i \in I$ appears exactly $|L_i| - 1$ times in $\boldsymbol{lo}$.

ALGORITHM 1: *MapToDesign*(ro, lo)
Data: optimal design problem, list *ro*, ordered multiset *lo*
Result: design $(\boldsymbol{r}, \boldsymbol{l})$
$\forall c \in C : r_c \leftarrow \emptyset$
$\forall i \in I : l_i \leftarrow \text{argmin}_{l \in L_i} H_l$
for $(c, i) \in \boldsymbol{ro}$ **do**
  $\boldsymbol{r}' \leftarrow \boldsymbol{r}$
  $r'_c \leftarrow r_c \cup \{i\}$
  if $(\boldsymbol{r}', \boldsymbol{l})$ is *feasible* **then**
    $\boldsymbol{r} \leftarrow \boldsymbol{r}'$
  end
end
for $i \in \boldsymbol{lo}$ **do**
  $\boldsymbol{l}' \leftarrow \boldsymbol{l}$
  $l'_i \leftarrow \text{argmin}_{l \in L_i : H_l > H_{l_i}} \boldsymbol{H_l}$
  if $(\boldsymbol{r}, \boldsymbol{l}')$ is *feasible* **then**
    $\boldsymbol{l} \leftarrow \boldsymbol{l}'$
  end
end
output $(\boldsymbol{r}, \boldsymbol{l})$

Next, we show how to translate a design plan $(\boldsymbol{ro}, \boldsymbol{lo})$ into a feasible design. The translation is presented formally in Algorithm 1. Given redundancy, diversity, and hardening investments R, D, and H, we can obtain a feasible design $(\boldsymbol{r}, \boldsymbol{l})$ as follows: start from an empty design (i.e. no implementations deployed and lowest-cost hardening level chosen for every implementation type); iterate over *ro* in order and for each $(c, i) \in \boldsymbol{ro}$, add i to r_c if it does not lead to the violation of the budget constraints; finally, iterate over $\boldsymbol{lo}$ in order and for each $i \in \boldsymbol{lo}$, increase security level l_i if it does not lead to the violation of the budget constraint. Note this mapping is surjective.

ALGORITHM 2: Meta-Heuristic Design Algorithm
Data: optimal design problem, number of iterations $k_{\max}$, initial temperature T_0, cooling parameter β
Result: design (r, l)
choose $(\boldsymbol{ro}, \boldsymbol{lo})$ at random

$\rho \leftarrow Risk(MapToDesign(\boldsymbol{ro}, \boldsymbol{lo}))$
for $k = 1, \ldots, k_{\max}$ **do**
  $(ro', lo') \leftarrow Perturb(\boldsymbol{ro}, \boldsymbol{lo})$
  $\rho' \leftarrow Risk(MapToDesign(\boldsymbol{ro}', \boldsymbol{lo}'))$
  $T \leftarrow T_0 \cdot e^{-\beta k}$
  $pr \leftarrow e^{(\rho' - \rho)/T}$
  if $(\rho' < \rho) \ \vee \ (\text{rand}(0, 1) \leq pr)$ **then**
    $\boldsymbol{ro} \leftarrow \boldsymbol{ro}'$
    $\boldsymbol{lo} \leftarrow \boldsymbol{lo}'$
  end
end
output $MapToDesign(\boldsymbol{ro}, \boldsymbol{lo})$

Finally, we present our meta-heuristic design algorithm (see Algorithm 2), which can find a near-optimal design in polynomial time. The algorithm starts by choosing a random design plan $(\boldsymbol{ro}, \boldsymbol{lo})$. In practice, we can implement this simply as choosing a random permutation of the list of component-implementation pairs and a random permutation of the multiset of implementation types. The algorithm then performs a fixed number of iterations, in each iteration choosing a random neighbor $(\boldsymbol{ro}', \boldsymbol{lo}')$ of the current plan $(\boldsymbol{ro}, \boldsymbol{lo})$, and replacing the current plan with the neighbor with some probability. This probability depends on the risk of both the current and the neighboring plan, and decreases with the number of iterations, as we approach the final solution. A key step of the algorithm is $Perturb(\boldsymbol{ro}, \boldsymbol{lo})$, which chooses a random neighbor of $(\boldsymbol{ro}, \boldsymbol{lo})$. In practice, we implement this as taking two elements of *ro* at random and switching them with each other, by similarly switching the order of two random elements of $\boldsymbol{lo}$, and returning the re-ordered list and multiset as the neighbor $(\boldsymbol{ro}, \boldsymbol{lo})$.

4. Evaluation

To demonstrate the applicability of our framework, we present two case studies from two canonical IIoT domains: water distribution and transportation systems.

4.1. Cyber-physical contamination attacks against water-distribution networks

IIoT systems have a particularly significant and wide application in water distribution systems. Examples include monitoring water quality and detecting leaks. On the one hand, IIoT offers significant advantages, such as improved service and better maintenance at a low cost, but on the other hand, potential challenges include cost of the cyber infrastructure, reliability of communications, and of course, cyber-security.

As evidenced by the recent water crisis in Flint, MI [21], ensuring the quality of drinking water is of critical importance. Compromising systems that control

the treatment and distribution of drinking water may allow adversaries to suppress warnings about contaminations or to decrease the quality of water [22]. Cyber-attacks can also have a devastating environmental impact. For example, in 2000, a disgruntled ex-employee launched a series of attacks against the SCADA system controlling sewage equipment in Maroochy Shire, Australia [23,24]. As a result of these attacks, approximately 800,000 liters of raw sewage spilt out into local parks and rivers, killing marine life.

Here, we apply our framework to model cyber-physical contamination attacks against water-distribution systems. The system is modeled as a graph, in which links represent pipes, and nodes represent junctions of pipes, residential consumers, reservoirs, pumps, etc. IIoT components include:

- *Sensors*: water-quality sensors, which are located at certain nodes of the water-distribution network;
- *Processing*: components that collect, process, and forward water-quality data;
- *Interfaces*: components with human-machine interfaces, which can alert operators about contaminations.

We consider a malicious adversary who tries to cause harm by contaminating the water network with harmful chemicals. We assume that the adversary can introduce contaminants at certain nodes, such as unprotected reservoirs or tanks, which will then spread in the network, eventually reaching the residential consumers. We measure the impact of this physical attack as the amount of contaminants consumed by residential consumers before the detection of the attack.

To detect contaminations, each sensor continuously monitors the water flowing through the node at which it is deployed, and raises an alarm when the concentration of a contaminant reaches a threshold level. The alert generated by a *sensor* node is sent to a *processing* node, which forwards the alert to an *interface* node that can notify operators. Once operators are alerted, they respond immediately by warning residents not to consume water from the network.

We measure the impact of a physical attack as the amount of contaminants consumed by residential consumers before they are warned. This amount depends on the time between the physical attack and its detection, the contaminant concentration levels at the consumer nodes in this time interval, and the amount of water consumed in this interval. Note that this impact depends on the uncompromised components $C \setminus \hat{C}$ since the time of detection depends on the functionality of these components.

To increase the impact of the physical attack, the adversary launches a cyber-attack, which compromises and disables some of the components

$\hat{C}$. Since the adversary's goal is to suppress warnings, this attack can be modeled as a *stealthy attack* (Section 2.3.1.2). We assume that the adversary first compromises a set of components $\hat{C}$, and then decides where to introduce the contaminant, maximizing the impact $Impact(\hat{C})$. Our goal is to minimize the risks posed by such cyber-physical attacks by designing a resilient system based on a systematic allocation of investments to redundancy, diversity, and hardening. We present numerical results for this case study in Section 5.

4.2. Cyber-attacks against transportation networks

Transportation systems is another application domain that can benefit greatly from IIoT by driving down costs and minimizing system failures, while supplying vast amounts of data for operators, drivers, and facilities that result in significant operational improvements. Transportation systems include multiple components that are becoming susceptible to attacks through wireless interfaces or even remote attacks through the Internet [16]. Indeed, recent studies have shown that many traffic lights deployed in practice have easily exploitable vulnerabilities, which could allow an attacker to tamper with the configuration of these devices. Due to hardware-based failsafes, compromising a traffic signal does not allow an attacker to set the signal into an unsafe configuration that could immediately lead to traffic accidents [25]. However, compromising a signal does enable tampering with its schedule, which allows an attacker to cause disastrous traffic congestions.

Here, we apply the proposed framework to model cyber-attacks against traffic control. The physical part of the system may be modeled using a traffic model, such as Daganzo's well-known cell-transmission model [26]. The cyber-part of the system is compromised of the following components:

- *Interface*: components with human-machine interfaces, which operators use to control the schedules of traffic lights in the transportation network;
- *Processing*: components that process and forward control signals sent by operators;
- *Actuator*: traffic lights with software based controllers.

We consider a malicious adversary who tries to cause damage by compromising some components $\hat{C}$ of the traffic-control system and tampering with the schedules of traffic lights. We measure the impact $Impact(\hat{C})$ of this cyber-attack as the increase in traffic congestion, which is quantified as the total travel time of the vehicles in the network, compared to normal congestion without an attack. We assume that the adversary aims to cause maximum

damage without attempting to hide its attack. Hence, we model its attack as a *non-stealthy attack* (Section 2.3.1.1).

5. Numerical results

In this section, we present numerical results to evaluate the proposed approach. First, we focus on the evaluation of the approach in terms of reducing the security risks by integrating redundancy, diversity, and hardening. Then, we focus on the performance of the proposed design algorithm in terms of running time.

5.1. Case-study examples

5.1.1. Water distribution system

We use a real-world water-distribution network from Kentucky, which we obtained from the Water Distribution System Research Database[2] [27]. The topology of this network, which is called KY3 in the database, is shown by Figure 2. In addition to topology, the database also contains hourly water-demand values for each network node.

We assume that the adversary can introduce a contaminant at one of six given nodes in the network, which model three tanks and three reservoirs. Once the contaminant is introduced, we simulate its spread throughout the network using EPANET[3] . From the simulation, we obtain the contaminant concentration values at the various nodes as functions of time. For a given set of compromised components $\hat{C}$, we then use these values to compute the time of detection and the resulting impact $Impact(\hat{C})$ (i.e. amount of contaminant

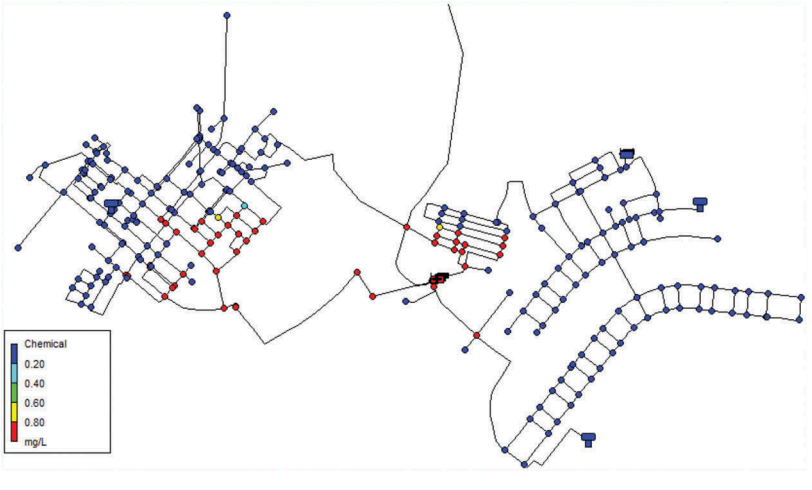

Figure 2. Topology of the water-distribution network. Colors show the spread of the contaminant from the first reservoir two hours after its introduction.

consumed by the time of detection). Finally, we use the following numerical parameter values:

- $I = \{i_1, i_2, i_3, i_4, i_5\}$;
- for every $c \in C$, $I_c = I$;
- $R_{i_1} = R_{i_2} = R_{i_3} = 0$[4] and $R_{i_4} = R_{i_5} = 1$;
- $D_{i_1} = 0$[5] and $D_i = 1$ for every $i \in \{i_2, i_3, i_4, i_5\}$;
- for every $i \in I$, $L_i = \{1, 2, 3, \ldots, 10\}$;
- for every $l \in L_i$, $S_l = 1 - 0.5^{0.5 \cdot l + 1}$ and $H_l = 4 \cdot l^2$.

5.1.2. Transportation network

We use the Grid model with Random Edges (GRE) to generate a random network topology [28], which closely resembles real-world transportation networks.[6] For a detailed description of this model, we refer the reader to [28,29]. We use Daganzo's cell transmission model to simulate traffic flowing through the generated network [26], computing the turn decisions of the vehicles based on a linear program that minimizes total travel time [30]. Following Daganzo's proposition, we model traffic lights as constraints on the inflow proportions [31], and we select the default (i.e. uncompromised) schedules of the traffic lights to minimize congestion. Finally, we allow the attacker to select any valid configuration for compromised lights.

We use the following parameter values for our illustrations:

- $I = \{i_1, i_2, i_3, i_4, i_5\}$;
- for every $c \in C$, $I_c = I$;
- $D_{i_1} = 0$ and $D_i = 20$ for every $i \in \{i_2, i_3, i_4, i_5\}$;
- for every $i \in I$, $R_i = 1$, $D_i = 20$, and $L_i = \{1, 2, 3, \ldots, 10\}$;
- for every $l \in L_i$, $S_l = 1 - 0.5^{0.5 \cdot l + 2}$ and $H_l = 10 \cdot l^2$.

5.2. Risk evaluation

Next, we study how security risks depend on investments into redundancy, diversity, and hardening, as well as their optimal combinations.

5.2.1. Water-distribution network

First, we study risks in the water-distribution network. Figure 3 shows the security risk in the water-distribution network for various budget values invested into the canonical approaches (i.e. redundancy, diversity, or hardening) and their optimal combination. Again, we note the logarithmic scaling on the vertical axis. We see that investing in a combination of redundancy, diversity, and hardening results in significantly lower risks than investing in only one of these approaches, thus demonstrating the efficacy and superior performance of a synergistic approach.

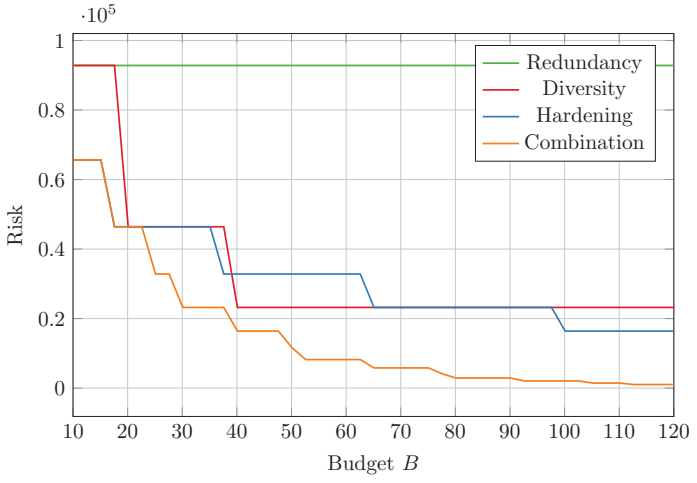

Figure 3. Security risk in the water-distribution network when investing only in redundancy, only in diversity, only in hardening, or in their combination.

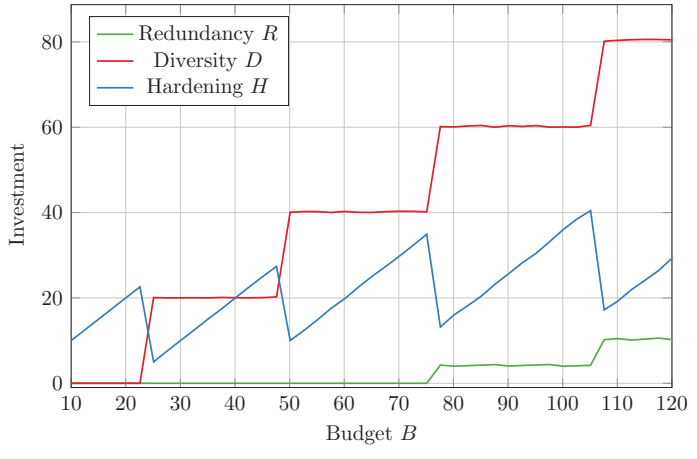

Figure 4. Optimal combination of redundancy, diversity, and hardening investments in the water-distribution network.

Figure 4 shows the optimal combination of redundancy, diversity, and hardening investments in the water-distribution network for various budget values. In this example, the optimal design is primarily a combination of diversity and hardening. However, with higher budget values, designers also need to invest in redundancy. Note that the design approach also determines the optimal deployment of components. Figure 5 shows the optimal deployment for budget $B = 90$. Colored disks represent component instances, different colors corresponding to different implementations.

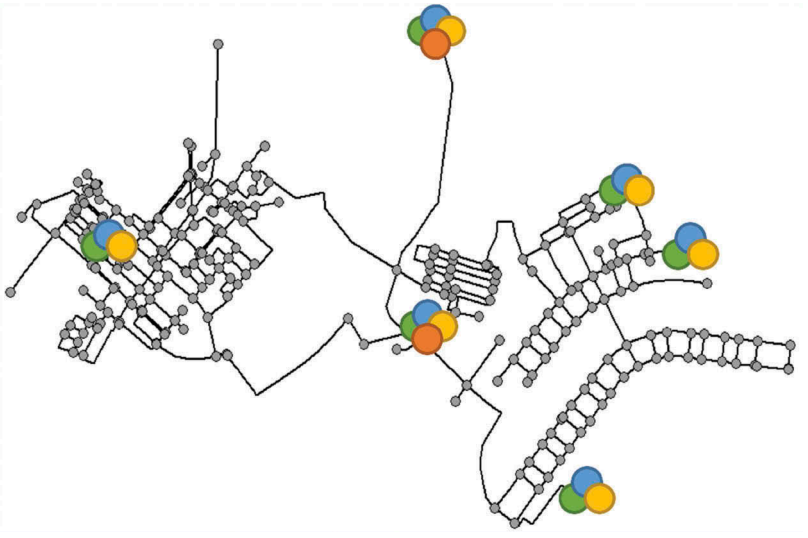

Figure 5. Optimal deployment with budget $B = 90$.

5.2.2. Transportation networks

Second, we consider security risks in the transportation network. In this case, we restrict our study to diversity and hardening since deploying multiple instances of a traffic light may be infeasible in practice. Hence, we assume that exactly one instance is deployed for each component.

Figure 6 shows the security risk in the transportation network with the canonical approaches and their combinations for various budget values. The figure shows that – similar to the case of water-distribution networks – the combined approach is clearly superior to canonical approaches.

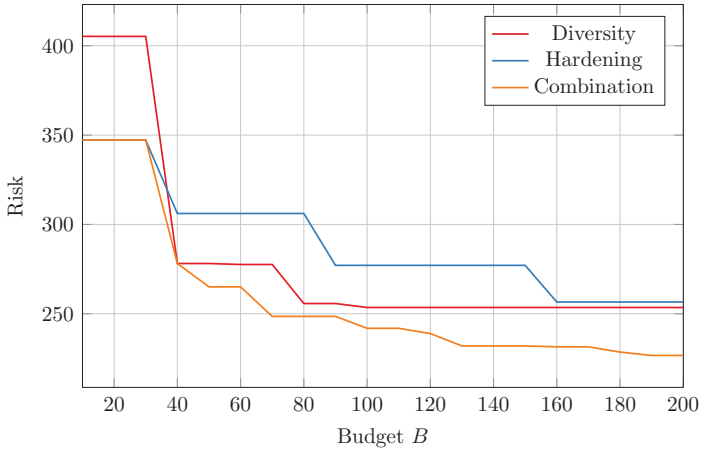

Figure 6. Security risk in the transportation network when investing only in diversity, only in hardening, or in their combination.

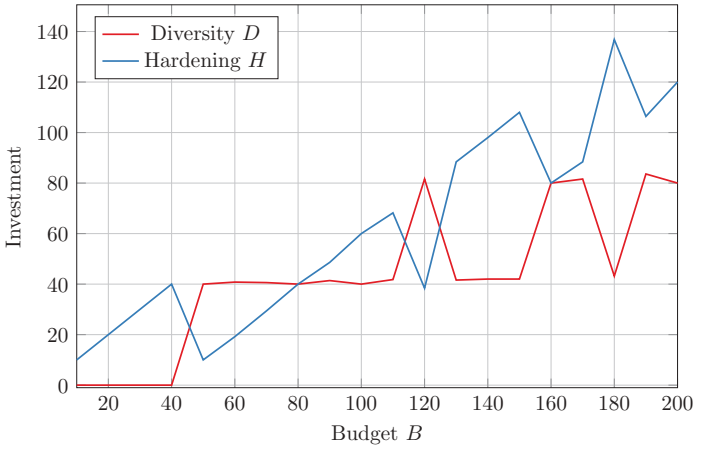

Figure 7. Optimal combination of diversity and hardening investments in the transportation network.

Figure 7 shows the optimal combination of diversity and redundancy in the transportation network for various budget values. Except for very low values, the optimal combination invests substantial amounts in both diversity and hardening.

5.3. Performance

To illustrate the performance of the proposed design algorithm, we use the water-distribution network with $R = 10$ and $D = H = 100$. We find that the meta-heuristic algorithm (Algorithm 2) is very efficient: a single iteration takes less than 6.4×10^{-4} seconds (more than 1,500 iterations per second) on an average laptop computer .[7] To determine the number of iterations that are necessary to find a good solution in practice, we focus on the solution quality (i.e. security risk) as a function of the number of iterations.

Figure 8 shows the security risk in each iteration of one particular execution of the meta-heuristic algorithm (Algorithm 2) with the current solution (solid red line) and with the best solution found so far (dashed blue line). Please note the logarithmic scaling on the vertical axis. We have executed the algorithm a number times, but since the results are qualitatively the same, we plot only one particular execution for illustration. The figure shows that risk decreases rapidly in the first few hundred iterations, but after around 400 iterations, the decrease becomes much slower. At around one thousand iterations, the risk reached its lowest value, so we omit the remaining iterations from the plot. In light of this, it is clear that the running time of the meta-heuristic algorithm is very low since it settles in a matter of seconds.

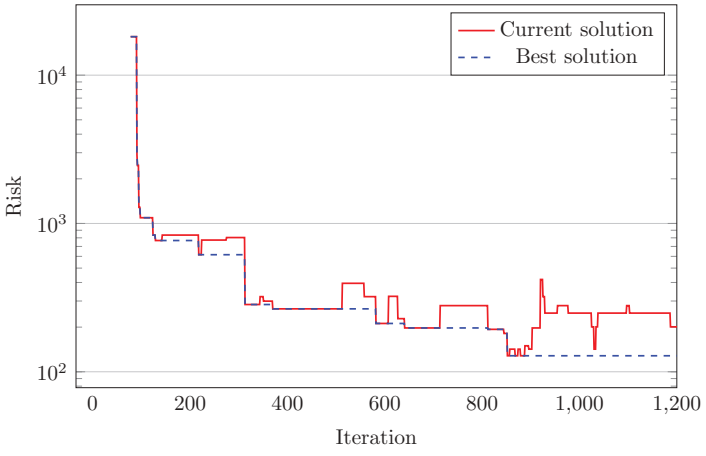

Figure 8. Security risk in each iteration of one execution of the the meta-heuristic algorithm (Algorithm 2).

6. Related work

Modern technology trends such as IIoT and cyber-physical systems (CPS) have significantly improved the overall functionality, reliability, observability, and operational efficiency of industrial control systems and critical infrastructure networks [32,33]. The integration and connectivity between various system components allow data exchange and information processing to fine tune system processes. However, this integration and connectivity also opens new threat channels in the form of cyber- and cyber-physical attacks, against which these systems need to be secured [1,34]. Conventional cybersecurity mechanisms are inadequate and thus need to be expanded to incorporate the complexity and physical aspects of such systems [1,5,34]. In fact, peculiar requirements in IIoT, such as hard time-constraints on the execution of operations, large physical base, wide interface for interaction with system components, and a direct impact on the physical world, require adjustment of security goals in IIoT as compared to typical IT systems [35,36]. Consequently, commercially available off-the-shelf IT cyber-security solutions are not sufficient for IIoT security. Greater connectivity results in higher risk due to broader attack surface and multiple data injection points within a system. From the perspective of controlling a network, false data injection attacks could be categorized into the following types [35,37]: corrupted data to controllers due to compromised sensors or compromised links between sensors and controllers, bogus data to the actuators, and denial-of-service attacks resulting in missed deadlines for executing system operations. A detailed overview of the security issues in industrial automation systems that are based on open communication systems is provided in [38]. Similarly, security issues associated with various documented standards in SCADA systems are

highlighted in [18,35,39]There are various other studies that mainly highlight the security threats and associated risk assessment in the domain of industrial IoT, for instance [36,40–44] . All of these studies discuss and point towards a holistic security framework to address the security issues in industrial IoT. In this paper, we provide a framework for synergistic security that combines various security mechanisms, including hardening [45–48], diversification of system components [17,49–52], and adding redundancy to effectively secure such systems. Next, we briefly discuss security issues in three critical infrastructure networks that are increasingly adapting and benefiting from IIoT paradigm: water distribution systems, traffic networks, and power systems.

6.1. Water-distribution security

The water-supply industrial sector can benefit significantly from applying the ideas and technology of industrial Internet [53]. An intelligent urban water-supply management system, which consists of IoT gateways connecting the water assets (for instance, water pumps, valves, and tanks) to the cloud service platform for advanced analytics, significantly improves the operational efficiency, safety, and service availability of the overall system [54,55]. There are ongoing efforts to develop efficient remote monitoring systems for pipeline monitoring (such as PIPENET deployed at Boston Water and Sewer Commission [56,57]), water quality monitoring [58–60], leak and burst detection [61,62], and other applications, for instance [63–65]. The adoption of new technologies (such as IoT, CPS) and networking devices enhances the monitoring capability, service reliability, and operational efficiency of water distribution systems, but also exposes them to malicious intrusions in the form of cyber- and cyber-physical attacks [22,66,67] . A number of attack scenarios against water distributions systems are specified and demonstrated through simulations in [22]. Recently, in [68], several attacks on simulated and a real water distribution testbed (WADI [69]) are demonstrated through cyber-physical botnets capable of performing adversarial control strategies under CPS constraints. The security breach in the SCADA system of Maroochy Water Services, Australia [24] is a famous incident, which also highlights the need for effective security mechanisms. To effectively address the security challenge in such complex, interconnected, and spatially expanded systems, we need to employ a combination of security mechanisms to protect them against cyber-physical attacks.

6.2. Traffic network security

Like other modern infrastructures, traffic networks are complex and are becoming increasingly connected with traffic lights, road sensors, and vehicles exchanging information with each other. This interconnectedness – though

useful at many levels – has also increased the attack surface for potential attackers that can significantly disrupt the traffic by taking control of a few network components, such as signal lights or sensors [16,25,70]. Recent studies outline the scope of the damage that can be caused by an adversary having an access to the traffic control infrastructure [71]. There are studies demonstrating attacks that can realize non-existent jams and virtual vehicles, tamper with signal schedules [72–75]. Considering the impact of successful attacks, it is imperative to systematically understand the existence of vulnerabilities, and design security frameworks to protect traffic infrastructure against such malicious attacks [76,77].

6.3. Power system security

Power utilities across the globe are increasingly adapting the IoT vision by upgrading to Smart Grids (SG), which are equipped with computation and communication capabilities to and from end users. This enables power utilities as well as consumers to utilize a rich space of SG services, including better management of energy resources. However, security issues predominantly hinder the large-scale deployment of IoT devices in SG. At a high level, security issues in SG can be classified into five groups as surveyed in [78]. These categories include device issues [79,80], networking issues [81–83], dispatching and management concerns [84], anomaly detection issues [85], and other relevant matters such as software architecture for the utility hosted on clouds [86–88] [81,83]. outlines the development and classification of communication systems in power grids along with the security concerns and possible threats in such systems [89]. highlights security threats unique to SG by comparing the security requirements of SG and the Internet. Further security risks in power systems that are equipped with bidirectional communication capabilities have been reported in recent surveys, for instance [90–93].

To cope with the identified and predicted security threats, various techniques have been proposed to improve the resilience and security of SG. For example, attacks that inject false data can undermine state estimation, which may lead to erroneous control and physical damage. Defenders can prevent these integrity attacks from succeeding through the strategic placement of phasor measurement units (PMU). In [94], an effective greedy algorithm is developed for optimal PMU placement, which ensures system observability and defends against data integrity attacks at the same time. In addition to proactive defense through deployment, defenders may also employ adaptive strategies to protect smart grids from ongoing attacks. In [95], an adaptive Markov strategy is proposed to defender against unknown attackers with dynamic and unpredictable behaviors. The effectiveness of this strategy against data integrity attacks that inject false voltage information is evaluated on standard test cases, demonstrating lower load shedding. Similarly, other techniques are employed to improve various

security aspects in SG such as deployment of redundant meters to verify the integrity of data collected from advanced metering infrastructure [96], hardware-in-the-loop reconfigurable system design with intelligent coordination schemes to deal with system vulnerabilities [97,98], hardening of system components [99], and various other strategies [34,100–103]. In one way or the other, these approaches are adding redundancy within the systems, diversifying system components, or hardening a subset of system components and devices. In this paper, we propose to strategically combine these individual approaches to improve resilience and security in SG against malicious disruptions.

7. Conclusion and future work

In this paper, we introduced a framework that considers three canonical approaches–redundancy, diversity, and hardening–for improving security and resilience of IIoT systems. Our goal is to provide theoretical foundations for designing systems that combine these canonical approaches. We showed that the problem of finding an optimal design is computationally hard, which means that practical designs may not be found using exhaustive searches. Therefore, we introduced an efficient meta-heuristic algorithm, whose running time is polynomial in the size of the problem instance. To illustrate the practical applicability of our results, we discussed two example application domains, water distribution and transportation systems. Our numerical evaluation shows that integrating redundancy, diversity, and hardening can lead to reduced security risk at the same cost.

7.1. Application to power systems

While the emergence of Smart Grid systems is envisioned to result in significant improvements in energy efficiency and sustainability, it also exposes power systems to threats posed by malicious cyber-attacks. The 2015 and 2016 cyber-attacks against the Ukrainian power grid, which both resulted in blackouts, have demonstrated that these threats are very real [104,105]. Therefore, it is crucial to ensure that Smart Grids are robust against such attacks. To this end, we can apply our framework to guide the design of robust power systems by combining redundancy, diversity, and hardening techniques optimally. Since the application and evaluation of our framework to power systems is outside the scope of this article, we provide a high-level overview of applying our framework to this domain in future work.

To capture the underlying physical and control systems, we can use existing models or simulation tools from the domain of power systems, such as GridLAB-D [106] or Transactive Energy Simulation Platform (TESP) for transactive energy systems.[8] The IIoT components of a power system can include:

- *Sensors*: may include a variety of sensing devices at different modeling granularity, such as phasor measurement units, power-line temperature sensors, or smart meters that are treated as indivisible components;
- *Processing*: components that collect, process, and forward sensor data and control signals, such as RTUs and PLCs;
- *Actuators*: can model lower-level actuator devices, e.g. individual electrical switches, or high-level components, e.g. distributed energy resources;
- *Interfaces*: components with human-machine interfaces that allow human operators to monitor and intervene.

A malicious adversary may compromise some of these components in order to tamper with the Smart Grid and to cause blackouts, resource waste, or physical damage. We can quantify the impact of cyber-attacks as the amount of unmet demand for electric power, the increase in operating costs, and the cost of damaged hardware. To estimate the impact of a particular attack, we can use a power-system model or simulator. However, this faces two key challenges. First, simulating power systems for a large number of possible attacks may be computationally expensive. Second, finding an optimal strategy for the attacker given a particular set of vulnerable components may be hard. In future work, we plan to address these challenges using novel techniques from the area of artificial intelligence: using deep learning to predict impact based on simulation results for a relatively low number of attacks, and using deep reinforcement learning to find optimal strategies for attackers [107–109].

Notes

1. Note that relaxing this assumption would be straightforward; however, such a generalization would provide little further insight into security.
2. http://www.uky.edu/WDST/database.html.
3. https://www.epa.gov/water-research/epanet.
4. We set these to zero to model existing deployment since we are interested in how to invest in improving security and resilience.
5. We set this to zero so that there always exists a feasible deployment.
6. We instantiated the model with $W = 5$, $L = 5$, $p = 0.507$, and $q = 0.2761$ based on [29].
7. MacBook Pro with 2.9 GHz Intel Core i5 processor.
8. https://github.com/pnnl/tesp.

Acknowledgments

This work was supported in part by the National Science Foundation under Grants CNS-1238959 and IIS-1905558, by the Air Force Research Laboratory under Grant FA 8750-14-2-0180, and by the National Institute of Standards and Technology under Grant 70NANB18H198.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was supported by the Air Force Research Laboratory [FA-8750-14-2-0180]; National Institute of Standards and Technology [70NANB18H198]; National Science Foundation [IIS-1905558]; National Science Foundation [CNS-1238959].

ORCID

Aron Laszka http://orcid.org/0000-0001-7400-2357
Xenofon Koutsoukos http://orcid.org/0000-0002-0923-6293

References

[1] Sadeghi AR, Wachsmann C, Waidner M Security and privacy challenges in industrial internet of things. In: Proceedings of the 52nd Annual Design Automation Conference; San Francisco, CA. ACM; 2015. p. 54.
[2] Humayed A, Lin J, Li F, et al. Cyber-physical systems security—a survey. IEEE Int Things J. 2017;4(6):1802–1831.
[3] Ling Z, Luo J, Xu Y, et al. Security vulnerabilities of internet of things: A case study of the smart plug system. IEEE Int Things J. 2017;4(6):1899–1909.
[4] Tomić I, McCann JA. A survey of potential security issues in existing wireless sensor network protocols. IEEE Int Things J. 2017;4(6):1910–1923.
[5] Cheminod M, Durante L, Valenzano A. Review of security issues in industrial networks. IEEE Trans Ind Inform. 2013;9(1):277–293.
[6] Siegel JE, Kumar S, Sarma SE. The future internet of things: secure, efficient, and model-based. IEEE Int Things J. 2018;5(4):2386–2398.
[7] Schneider FB, Birman KP. The monoculture risk put into context. IEEE Secur Privacy. 2009;7(1):14–17.
[8] Laszka A, Grossklags J Should cyber-insurance providers invest in software security? In: Proceedings of the 20th European Symposium on Research in Computer Security (ESORICS); Vienna, Austria; 2015sep. p. 483–502.
[9] Jones RL, Rastogi A. Secure coding: building security into the software development life cycle. Inf Syst Secur. 2004;13(5):29–39.
[10] Al-Shaer ES, Hamed HH. Modeling and management of firewall policies. IEEE Trans Network Serv Manage. 2004;1(1):2–10.
[11] Arkin B, Stender S, McGraw G. Software penetration testing. IEEE Secur Privacy. 2005;3(1):84–87.
[12] Zhao M, Grossklags J, Liu P An empirical study of web vulnerability discovery ecosystems. In: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (CCS); Denver, CO. ACM; 2015. p. 1105–1117.
[13] Laszka A, Zhao M, Grossklags J Banishing misaligned incentives for validating reports in bug-bounty platforms. In: Proceedings of the 21st European Symposium on Research in Computer Security (ESORICS); Heraklion, Greece; 2016sep. p. 161–178.

[14] Anderson R, Moore T. The economics of information security. Science. 2006;314 (5799):610–613.
[15] Gordon LA, Loeb MP. The economics of information security investment. ACM Trans Inf Syst Secur (TISSEC). 2002;5(4):438–457.
[16] Laszka A, Potteiger B, Vorobeychik Y, et al. Vulnerability of transportation networks to traffic-signal tampering. In: Proceedings of the 7th International Conference on Cyber-Physical Systems (ICCPS); Vienna, Austria. IEEE Press; 2016. p. 16.
[17] Zhang M, Wang L, Jajodia S, et al. Network diversity: a security metric for evaluating the resilience of networks against zero-day attacks. IEEE Trans Inf Forensics Secur. 2016;11(5):1071–1086.
[18] Cherdantseva Y, Burnap P, Blyth A, et al. A review of cyber security risk assessment methods for SCADA systems. Comput Sec. 2016;56:1–27.
[19] Hoo KJS. How much is enough? a risk management approach to computer security. Stanford (CL): Stanford University Stanford; 2000.
[20] LeBlanc HJ, Zhang H, Koutsoukos X, et al. Resilient asymptotic consensus in robust networks. IEEE J Sel Areas Commun. 2013;31(4):766–781.
[21] Kennedy M Lead-laced water in Flint: A step-by-step look at the makings of a crisis NPR; 2016. [cited 2019 Jan 29]. Available form: http://www.npr.org/sections/thetwo-way/2016/04/20/465545378/
[22] Taormina R, Galelli S, Tippenhauer NO, et al. Characterizing cyber-physical attacks on water distribution systems. J Water Resour Plann Manage. 2017;143.
[23] Abrams M, Weiss J Malicious control system cyber security attack case study – Maroochy Water Services, Australia; 2008. [cited 2019 Jan 29]. Available form: http://csrc.nist.gov/groups/SMA/fisma/ics/documents/Maroochy-Water-Services-Case-Study_report.pdf
[24] Slay J, Miller M. Lessons learned from the Maroochy water breach. In: Goetz E, Shenoi S, editors. Critical infrastructure protection. Boston (MA): Springer; 2008. p. 73–82.
[25] Ghena B, Beyer W, Hillaker A, et al. Green lights forever: analyzing the security of traffic infrastructure. In: Proceedings of the 8th USENIX Workshop on Offensive Technologies (WOOT); San Diego, CA; Vol. 14; 2014. p. 1–10.
[26] Daganzo CF. The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. Transp Res Part B Methodol. 1994;28 (4):269–287.
[27] Jolly MD, Lothes AD, Sebastian Bryson L, et al. Research database of water distribution system models. J Water Resour Plann Manage. 2014;140(4):410–416.
[28] Peng W, Dong G, Yang K, et al. A random road network model for mobility modeling in mobile delay-tolerant networks. In: Proceedings of the 8th International Conference on Mobile Ad-hoc and Sensor Networks (MSN); Chengdu, China. IEEE; 2012. p. 140–146.
[29] Peng W, Dong G, Yang K, et al. A random road network model and its effects on topological characteristics of mobile delay-tolerant networks. IEEE Trans Mobile Comput. 2014;13(12):2706–2718.
[30] Ziliaskopoulos AK. A linear programming model for the single destination system optimum dynamic traffic assignment problem. Transp Sci. 2000;34(1):37–49.
[31] Daganzo CF. The cell transmission model, part II: network traffic. Transp Res Part B Methodol. 1995;29(2):79–93.
[32] Moyne JR, Tilbury DM. The emergence of industrial control networks for manufacturing control, diagnostics, and safety data. Proc IEEE. 2007;95(1):29–47.

[33] Colombo AW, Karnouskos S, Kaynak O, et al. Industrial cyberphysical systems: A backbone of the fourth industrial revolution. IEEE Ind Electron Mag. 2017;11 (1):6–16.
[34] Koutsoukos X, Karsai G, Laszka A, et al. SURE: A modeling and simulation integration platform for evaluation of secure and resilient cyber–physical systems. Proc IEEE. 2018;106(1):93–112.
[35] Zhu B, Joseph A, Sastry S A taxonomy of cyber attacks on SCADA systems. In: 2011 International conference on internet of things and 4th international conference on cyber, physical and social computing; Dalian, China. IEEE; 2011. p. 380–388.
[36] Frustaci M, Pace P, Aloi G, et al. Evaluating critical security issues of the IoT world: present and future challenges. IEEE Int Things J. 2018;5(4):2483–2495.
[37] Radisavljevic-Gajic V, Park S, Chasaki D. Vulnerabilities of control systems in internet of things applications. IEEE Int Things J. 2018;5(2):1023–1032.
[38] Dzung D, Naedele M, Von Hoff TP, et al. Security for industrial communication systems. Proc IEEE. 2005;93(6):1152–1177.
[39] Gao J, Liu J, Rajan B, et al. SCADA communication and security issues. Secur Commun Networks. 2014;7(1):175–194.
[40] Da Xu L, He W, Li S. Internet of things in industries: A survey. IEEE Trans Ind Inform. 2014;10(4):2233–2243.
[41] Meltzer D. Securing the industrial internet of things. Inf Syst Secur Assoc J. 2015;24–30.
[42] Lee J, Bagheri B, Kao HA. A cyber-physical systems architecture for industry 4.0-based manufacturing systems. Manuf Lett. 2015;3:18–23.
[43] Jing Q, Vasilakos AV, Wan J, et al. Security of the internet of things: perspectives and challenges. Wireless Networks. 2014;20(8):2481–2501.
[44] Suo H, Wan J, Zou C, et al. Security in the internet of things: a review. In: Procedings of the International Conference on Computer Science and Electronics Engineering (ICCSEE); Hangzhou, China; 2012. p. 648–651.
[45] Durkota K, Lisỳ V, Bošanskỳ B, et al. Optimal network security hardening using attack graph games. In: Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI); Buenos Aires, Argentina; 2015.
[46] Anantharaman P, Locasto M, Ciocarlie GF, et al. Building hardened internet-of-things clients with language-theoretic security. In: 2017 IEEE Security and Privacy Workshops (SPW); San Jose, CA. IEEE; 2017. p. 120–126.
[47] Rahman MA, Al-Shaer E, Bobba RB Moving target defense for hardening the security of the power system state estimation. Proceedings of the First ACM Workshop on Moving Target Defense; Scottsdale, AZ. ACM, 2014. p. 59–68.
[48] Mourad A, Laverdière MA, Debbabi M. An aspect-oriented approach for the systematic security hardening of code. Comput Sec. 2008;27(3–4):101–114.
[49] Cox B, Evans D, Filipi A, et al. N-variant systems: A secretless framework for security through diversity. In: USENIX Security Symposium; Vancouver, BC. 2006. p. 105–120.
[50] Nguyen-Tuong A, Evans D, Knight JC, et al. Security through redundant data diversity. In: 2008 IEEE International Conference on Dependable Systems and Networks With FTCS and DCC (DSN); Anchorage, AK. IEEE, 2008. p. 187–196.
[51] Okhravi H, Hobson T, Bigelow D, et al. Finding focus in the blur of moving-target techniques. IEEE Secur Privacy. 2014;12(2):16–26.
[52] Touhiduzzaman M, Hahn A, Srivastava A A diversity-based substation cyber defense strategy utilizing coloring games. IEEE Transactions on Smart Grid. 2018.
[53] Kartakis S Next generation cyber-physical water distribution systems [dissertation]. Imperial College London; 2016.

[54] Intelligent urban water supply testbed; 2018. [cited 2018 Dec 02]. Available form: http://www.iiconsortium.org/intelligent-urban-water-supply.htm
[55] Liu Z, Kleiner Y. Computational intelligence for urban infrastructure condition assessment: water transmission and distribution systems. IEEE Sens J. 2014;14 (12):4122–4133.
[56] Stoianov I, Nachman L, Madden S, et al. PIPENET: A wireless sensor network for pipeline monitoring. In: Proceedings of the 6th International Symposium on Information Processing in Sensor Networks (IPSN); Cambridge, MA. IEEE, 2007. p. 264–273.
[57] Stoianov I, Nachman L, Whittle A, et al. Sensor networks for monitoring water supply and sewer systems: lessons from Boston. In: Proceedings of the 8th Annual Water Distribution Systems Analysis Symposium (WDSA); Cincinnati, OH; 2006. p. 1–17.
[58] Ali S, Qaisar SB, Saeed H, et al. Network challenges for cyber physical systems with tiny wireless devices: A case study on reliable pipeline condition monitoring. Sensors. 2015;15(4):7172–7205.
[59] McKenna SA, Wilson M, Klise KA. Detecting changes in water quality data. Am Water Works Assoc J. 2008;100(1):74.
[60] Storey MV, Van der Gaag B, Burns BP. Advances in on-line drinking water quality monitoring and early warning systems. Water Res. 2011;45(2):741–747.
[61] Perez R, Sanz G, Puig V, et al. Leak localization in water networks: a model-based methodology using pressure sensors applied to a real network in Barcelona [applications of control]. IEEE Control Syst. 2014;34(4):24–36.
[62] Perelman LS, Abbas W, Koutsoukos X, et al. Sensor placement for fault location identification in water networks: A minimum test cover approach. Automatica. 2016;72:166–176.
[63] Yoon S, Ye W, Heidemann J, et al. SWATS: wireless sensor networks for steamflood and waterflood pipeline monitoring. IEEE Network. 2011;25:1.
[64] Torbol M, Kim S, Chou P Remote structural health monitoring systems for next generation SCADA. Smart Structures and Systems. 2013;11.
[65] Suciu G, Bezdedeanu L, Vasilescu A, et al. Unified intelligent water management using cyberinfrastructures based on cloud computing and IoT. In: Proceedings of the 21st International Conference on Control Systems and Computer Science (CSCS); Bucharest, Romania. IEEE, 2017. p. 606–611.
[66] Perelman L, Amin S A network interdiction model for analyzing the vulnerability of water distribution systems. In: Proceedings of the 3rd International Conference on High Confidence Networked Systems; Berlin, Germany. ACM, 2014. p. 135–144.
[67] Amin S, Litrico X, Sastry S, et al. Cyber security of water SCADA systems—part I: analysis and experimentation of stealthy deception attacks. IEEE Trans Control Syst Technol. 2013;21(5):1963–1970.
[68] Antonioli D, Bernieri G, Tippenhauer NO Taking control: design and implementation of botnets for cyber-physical attacks with CPSBot. arXiv preprint arXiv:180200152. 2018.
[69] Ahmed CM, Palleti VR, Wadi MAP: A water distribution testbed for research in the design of secure cyber physical systems. In: Proceedings of the 3rd International Workshop on Cyber-Physical Systems for Smart Water Networks; Pittsburgh, PA. ACM, 2017. p. 25–28.
[70] Ghanavati M, Chakravarthy A, Menon PP. Analysis of automotive cyber-attacks on highways using partial differential equation models. IEEE Trans Control Network Syst. 2017;5:1775–1786.

[71] Reilly J, Martin S, Payer M, et al. Creating complex congestion patterns via multi-objective optimal freeway traffic control with application to cyber-security. Transp Res Part B Methodol. 2016;91:366–382.
[72] Jeske T Floating car data from smartphones: what google and waze know about you and how hackers can control traffic. Proceedings of BlackHat Europe; Amsterdam, Netherlands; 2013.p. 1–12.
[73] Grad S. Engineers who hacked into LA traffic signal computer, jamming streets, sentenced. Los Angeles Times; El Segundo, CA; 2009 Dec.
[74] Zetter K Hackers can mess with traffic lights to jam roads and reroute cars WIRED; 2014. [cited 2019 Jan 29]. Available form: https://www.wired.com/2014/04/traffic-lights-hacking/
[75] Tufnell N Students hack Waze, send in army of traffic bots WIRED UK; 2014. Available form: http://www.wired.co.uk/article/waze-hacked-fake-traffic-jam
[76] Ernst JM, Michaels AJ. Framework for evaluating the severity of cybervulnerability of a traffic cabinet. Transp Res Rec. 2017;2619:55–63.
[77] Feng Y, Huang S, Chen QA, et al. Vulnerability of traffic control system under cyber-attacks using falsified data. In: Annual Meeting of the Transportation Research Board; Washington, DC; 2018.
[78] Liu J, Xiao Y, Li S, et al. Cyber security and privacy issues in smart grids. IEEE Commun Surv Tutor. 2012;14(4):981–997.
[79] Anderson R, Fuloria S Who controls the off switch? In: 2010 First IEEE International Conference on Smart Grid Communications, Gaithersburg, MD. IEEE, 2010. p. 96–101.
[80] McDaniel P, McLaughlin S. Security and privacy challenges in the smart grid. IEEE Secur Privacy. 2009;7(3):75–77.
[81] Ericsson GN. Cyber security and power system communication—essential parts of a smart grid infrastructure. IEEE Trans Power Delivery. 2010;25(3):1501–1507.
[82] Gao J, Xiao Y, Liu J, et al. A survey of communication/networking in smart grids. Future Gener Comput Syst. 2012;28(2):391–404.
[83] Kabalci Y. A survey on smart metering and smart grid communication. Renew Sust Energ Rev. 2016;57:302–318.
[84] Wang W, Xu Y, Khanna M. A survey on the communication architectures in smart grid. Comput Netw. 2011;55(15):3604–3629.
[85] Zhang Y, Wang L, Sun W, et al. Distributed intrusion detection system in a multi-layer network architecture of smart grids. IEEE Trans Smart Grid. 2011;2(4):796–808.
[86] Simmhan Y, Kumbhare AG, Cao B, et al. An analysis of security and privacy issues in smart grid software architectures on clouds. In: 2011 IEEE 4th international conference on cloud computing, Washington, DC. IEEE, 2011. p. 582–589.
[87] Bera S, Misra S, Rodrigues JJ. Cloud computing applications for smart grid: A survey. IEEE Trans Parallel Distrib Syst. 2015;26(5):1477–1494.
[88] Botta A, De Donato W, Persico V, et al. Integration of cloud computing and internet of things: a survey. Future Gener Comput Syst. 2016;56:684–700.
[89] Wang W, Lu Z. Cyber security in the smart grid: survey and challenges. Comput Netw. 2013;57(5):1344–1371.
[90] Khurana H, Hadley M, Lu N, et al. Smart-grid security issues. IEEE Secur Privacy. 2010;8(1):81–85.
[91] Fang X, Misra S, Xue G, et al. Smart grid—the new and improved power grid: A survey. IEEE Commun Surv Tutor. 2012;14(4):944–980.
[92] Mo Y, Kim THJ, Brancik K, et al. Cyber–physical security of a smart grid infrastructure. Proc IEEE. 2012;100(1):195–209.

[93] Sridhar S, Hahn A, Govindarasu M. Cyber–physical system security for the electric power grid. Proc IEEE. 2012;100(1):210–224.
[94] Yang Q, An D, Min R, et al. On optimal PMU placement-based defense against data integrity attacks in smart grid. IEEE Trans Inf Forensics Secur. 2017;12(7):1735–1750.
[95] Hao J, Kang E, Sun J, et al. An adaptive markov strategy for defending smart grid false data injection from malicious attackers. IEEE Trans Smart Grid. 2018;9 (4):2398–2408.
[96] Varodayan DP, Gao GX Redundant metering for integrity with information-theoretic confidentiality. In: 2010 First IEEE International Conference on Smart Grid Communications; Gaithersburg, MD. IEEE, 2010. p. 345–349.
[97] Qi H, Wang X, Tolbert LM, et al. A resilient real-time system design for a secure and reconfigurable power grid. IEEE Trans Smart Grid. 2011;2(4):770–781.
[98] Thale SS, Wandhare RG, Agarwal V. A novel reconfigurable microgrid architecture with renewable energy sources and storage. IEEE Trans Sustain Energy. 2015;51 (2):1805–1816.
[99] Panteli M, Trakas DN, Mancarella P, et al. Power systems resilience assessment: hardening and smart operational enhancement strategies. Proc IEEE. 2017;105 (7):1202–1213.
[100] Panteli M, Mancarella P. The grid: stronger bigger smarter? Presenting a conceptual framework of power system resilience. IEEE Power Energy Mag. 2015;13(3):58–66.
[101] Bie Z, Lin Y, Li G, et al. Battling the extreme: A study on the power system resilience. Proc IEEE. 2017;105(7):1253–1266.
[102] Russell BD, Benner CL. Intelligent systems for improved reliability and failure diagnosis in distribution systems. IEEE Trans Smart Grid. 2010;1(1):48–56.
[103] Li F, Luo B, Liu P Secure information aggregation for smart grids using homomorphic encryption. In: 2010 First IEEE International Conference on Smart Grid Communications; Gaithersburg, MD. IEEE, 2010. p. 327–332.
[104] Zetter K Inside the cunning, unprecedented hack of Ukraine's power grid WIRED; 2016. [cited 2019 Jan 29]. Available form: https://www.wired.com/2016/03/inside-cunning-unprecedented-hack-ukraines-power-grid/
[105] Greenberg A 'Crash override': the malware that took down a power grid WIRED; 2017. Available form: https://www.wired.com/story/crash-override-malware/
[106] Chassin DP, Schneider K, Gerkensmeyer C. GridLAB-D: an open-source power systems modeling and simulation environment. In: 2008 IEEE/PES Transmission and Distribution Conference and Exposition; Chicago, IL. IEEE, 2008. p. 1–5.
[107] Yan J, He H, Zhong X, et al. Q-learning-based vulnerability analysis of smart grid against sequential topology attacks. IEEE Trans Inf Forensics Secur. 2017;12 (1):200–210.
[108] Ni Z, Paul S. A multistage game in smart grid security: A reinforcement learning solution. IEEE Trans Neural Netw Learn Syst. 2019.
[109] Lillicrap TP, Hunt JJ, Pritzel A, et al. Continuous control with deep reinforcement learning. arXiv preprint arXiv:150902971. 2015.