Grigore Havarneanu
Roberto Setola
Hypatia Nassopoulos
Stephen Wolthusen (Eds.)

LNCS 10242

Critical Information Infrastructures Security

11th International Conference, CRITIS 2016
Paris, France, October 10–12, 2016
Revised Selected Papers

Springer

Lecture Notes in Computer Science **10242**

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

More information about this series at http://www.springer.com/series/7410

Grigore Havarneanu · Roberto Setola
Hypatia Nassopoulos · Stephen Wolthusen (Eds.)

Critical Information Infrastructures Security

11th International Conference, CRITIS 2016
Paris, France, October 10–12, 2016
Revised Selected Papers

Springer

Editors
Grigore Havarneanu
International Union of Railways
Paris
France

Roberto Setola
University Campus Bio-Medico
Rome
Italy

Hypatia Nassopoulos
Ecole des Ingénieurs de la Ville de Paris (EIVP)
Paris
France

Stephen Wolthusen
Royal Holloway, University of London
London
UK

and

Norwegian University of Science and Technology
Gjøvik
Norway

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Computer Science
ISBN 978-3-319-71367-0 ISBN 978-3-319-71368-7 (eBook)
https://doi.org/10.1007/978-3-319-71368-7

Library of Congress Control Number: 2017959628

LNCS Sublibrary: SL4 – Security and Cryptology

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

The 2016 International Conference on Critical Information Infrastructures Security (CRITIS 2016) was the 11th conference in the series, continuing a well-established tradition of successful annual conferences. Since its inception, CRITIS has been a global forum for researchers and practitioners to present and discuss the most recent innovations, trends, results, experiences, and concerns in selected perspectives of critical information infrastructure protection [C(I)IP] at large covering the range from small-scale cyber-physical systems security via information infrastructures and their interaction with national and international infrastructures, and ultimately also reaching policy-related aspects. In line with this tradition, CRITIS 2016 brought together experts from governments, regulators, scientists, and professionals from academia, industry, service providers, and other stakeholders in one conference to secure infrastructures and study ways to enhance their resilience to faults and deliberate attacks.

This volume contains the carefully reviewed proceedings of the 11th CRITIS conference, held in Paris, France, during October 10–12, 2016. The conference was organized by the International Union of Railways (Union Internationale des Chemins de Fer, UIC) — the worldwide professional association representing the railway sector and promoting rail transport. Following the call for papers, we received 58 high-quality submissions, which were thoroughly reviewed by the expert members of the international Program Committee (IPC). Out of the total submissions, 22 papers were accepted as full papers with eight further papers accepted as short papers offering work in progress; these short papers are also collected in this volume. Each paper was reviewed by at least three expert reviewers, and both full and short papers were retained for oral presentations during the conference. The technical papers were grouped into sessions that included topics on innovative responses for the protection of cyber-physical systems, procedures and organizational aspects in C(I)IP and advances in human factors, decision support, and cross-sector C(I)IP approaches. Furthermore, in continuation of an initiative first taken up at the 2014 CRITIS, the conference also included an award for young researchers in the area (the 3rd CIPRNet Young CRITIS Award), seeking to recognize and encourage the integration of talented younger researchers into the community. Five of the accepted papers were presented during a dedicated CYCA Session. This award was sponsored by the FP7 Network of Excellence CIPRNet. As in previous years, invited keynote speakers and special events complemented the three-day technical program. The five plenary talks were the following:

- Dr Artūras Petkus (NATO Energy Security Centre of Excellence, Lithuania) gave a CIPRNet Lecture entitled: “CEIP and Energy Security in Perspective of NATO Energy Security Center of Excellence.”
- Commander Cyril Stylianidis (Ministry of Interior, General Directorate for Civil Protection and Crisis Management, France), provided an overview of “The Crisis Interministerial Cell (CIC), the French Tool for Interministerial Level Crisis Management,” illustrated with recent examples from France.

- Mr. Kris Christmann (University of Huddersfield, Applied Criminology Centre, UK) offered "Findings from the PRE-EMPT Project: Establishing Best Practice for Reducing Serious Crime and Terrorism at Multi-Modal Passenger Terminals (MMPT)."
- Dr. Paul Theron (Thales Communications and Security, France) presented "A Way Towards a Fully Bridged European Certification of IACS Cyber Security," related to the work of DG JRC's ERNCIP Thematic Group on IACS cybersecurity certification.

In addition, the CRITIS 2016 participants had the opportunity to attend (with a limited number of places) an associated event organized at UIC the day after the main conference. The IMPROVER Workshop – "Meeting Public Expectations in Response to Crises" – addressed an important topic in C(I)IP, aiming to discuss how infrastructure operators meet these requirements today and how this can be improved.

It is our pleasure to express our gratitude to everybody that contributed to the success of CRITIS 2016. In particular, we would like to thank the general chair, Jean-Pierre Loubinoux (UIC Director-General), and the local UIC hosts, Jerzy Wisniewski (Fundamental Values Department Director) and Jacques Colliard (Head of UIC Security Division), for making CRITIS possible at the UIC headquarters in Paris – one of the most beautiful European capitals. Further, we would like to thank the members of the Program Committee, who did a tremendous job under strict time limitations during the review process. We also thank the members of the Steering Committee for the great effort and their continuous assistance in the organization of the conference. We are also grateful to the publicity chair and to the UIC Communications Department for their excellent dissemination support, and to the CIPRNet Network, which was an active supporting community. We are equally grateful to the keynote speakers who accepted our invitation and agreed to round off the conference program through presentations on hot topics of the moment. We would also like to thank the publisher, UIC-ETF, for their cooperation in publishing the selected papers from the pre-conference proceedings. Finally, we thank all the authors who submitted their work to CRITIS and who contributed to this volume for sharing their new ideas and results with the community. We hope that these ideas will generate further new ideas and innovations for securing our critical infrastructures for the benefit of the whole society.

September 2016

Grigore Havarneanu
Roberto Setola
Hypatia Nassopoulos
Stephen Wolthusen

Organization

Program Committee

Marc Antoni	International Union of Railways
Fabrizio Baiardi	Università de Pisa, Italy
Yohan Barbarin	CEA Gramat, France
Robin Bloomfield	CSR City University London, UK
Sandro Bologna	AIIC
Maria Cristina Brugnoli	CNIT
Arslan Brömme	GI Biometrics Special Interest Group (BIOSIG)
Emiliano Casalicchio	Blekinge Institute of Technology, Sweden
Michal Choras	ITTI Ltd.
Kris Christmann	Applied Criminology Centre, University of Huddersfield, UK
Myriam Dunn	ETH Center for Security Studies Zurich, Switzerland
Gregorio D'agostino	ENEA
Mohamed Eid	Commissariat à l'Energie Atomique et aux Energies Alternatives
Adrian Gheorghe	Old Dominion University, USA
Luigi Glielmo	Università del Sannio, Italy
Stefanos Gritzalis	University of the Aegean, Greece
Chris Hankin	Imperial College London, UK
Grigore M. Havarneanu	International Union of Railways
Bernhard M. Hämmerli	University of Applied Sciences Lucerne, Switzerland
Apiniti Jotisankasa	Kasetsart University, Thailand
Sokratis Katsikas	Center for Cyber and Information Security, NTNU
Marieke Klaver	TNO
Panayiotis Kotzanikolaou	University of Piraeus, Greece
Rafal Kozik	Institute of Telecommunications, UTP Bydgoszcz, Poland
Elias Kyriakides	University of Cyprus, Cyprus
Javier Lopez	University of Malaga, Spain
Eric Luiijf	TNO
José Martí	UBC
Richard McEvoy	Hewlett-Packard Enterprise
Maddalen Mendizabal	Tecnalia R&I
Igor Nai Fovino	Joint Research Centre
Aristotelis Naniopoulos	Aristotle University of Thessaloniki, Greece
Hypatia Nassopoulos	EIVP, France
Eiji Okamoto	University of Tsukuba, Japan
Gabriele Oliva	Campus Biomedico University of Rome, Italy

Evangelos Ouzounis	ENISA
Stefano Panzieri	Roma Tre University, Italy
Alexander Paz-Cruz	University of Nevada, Las Vegas, USA
Marios Polycarpou	University of Cyprus, Cyprus
Reinhard Posch	IAIK
Erich Rome	Fraunhofer IAIS, Germany
Vittorio Rosato	ENEA
Brendan Ryan	University of Nottingham, UK
Andre Samberg	SESOCUKR
Antonio Scala	Institute for Complex Systems/Italian National Research Council, Italy
Maria Paola Scaparra	Kent Business School, The University of Kent, UK
Eric Schellekens	ARCADIS
Roberto Setola	Università Campus Bio-Medico di Roma, Italy
George Stergiopoulos	Athens University of Economics and Business, Greece
Nils Kalstad Svendsen	Norwegian University of Science and Technology, Norway
Dominique Sérafin	CEA
Andre Teixeira	Delft University of Technology, The Netherlands
Marianthi Theocharidou	European Commission, Joint Research Centre
Alberto Tofani	ENEA
William Tolone	The University of North Carolina at Charlotte, USA
Simona Louise Voronca	Transelectrica
Marc Vuillet	EIVP
René Willems	TNO
Stephen D. Wolthusen	Royal Holloway, University of London and Norwegian University of Science and Technology, UK/Norway
Christos Xenakis	University of Piraeus, Greece
Enrico Zio	Politecnico di Milano, Italy
Inga Žutautaitė	Lithuanian Energy Institute, Lithuania

Additional Reviewers

Bernieri, Giuseppe
De Cillis, Francesca
Faramondi, Luca
Karyda, Maria
Kasse, Paraskevi
Kokolakis, Spyros
Leitold, Herbert
Maffei, Alessio
Nalmpantis, Dimitrios
Verrilli, Francesca

Contents

Stealth Low-Level Manipulation of Programmable Logic Controllers I/O by Pin Control Exploitation

Ali Abbasi[1(✉)], Majid Hashemi[2], Emmanuele Zambon[1,4], and Sandro Etalle[1,3]

[1] Services, Cyber Security and Safety Group, University of Twente, Enschede, The Netherlands
{a.abbasi,emmanuele.zambon,sandro.etalle}@utwente.nl
[2] Quarkslab, Paris, France
mhashemi@quarkslab.com
[3] Eindhoven University of Technology, Eindhoven, The Netherlands
s.etalle@tue.nl
[4] SecurityMatters BV, Eindhoven, The Netherlands
emmanuele.zambon@secmatters.com

Abstract. Input/Output is the mechanism through which Programmable Logic Controllers (PLCs) interact with and control the outside world. Particularly when employed in critical infrastructures, the I/O of PLCs has to be both reliable and secure. PLCs I/O like other embedded devices are controlled by a pin based approach. In this paper, we investigate the security implications of the PLC pin control system. In particular, we show how an attacker can tamper with the integrity and availability of PLCs I/O by exploiting certain pin control operations and the lack of hardware interrupts associated to them.

Keywords: PLC · Exploiting · SoC · ICS

1 Introduction

Programmable Logic Controllers (PLCs) are widely used today in various industries including mission critical networks that have to be both reliable and secure. One of the main purposes of the PLCs is to control the physical world equipment such as sensors, actuators or drivers within the context of an industrial process. PLCs communicate with this equipment by means of their I/O, which therefore need to be secure. Digging into their architecture, we know that the I/O interfaces of PLCs (e.g., GPIO, SCI, JTAG, etc.), are usually controlled by a so-called System on a Chip (SoC), an integrated circuit that combines multiple I/O interfaces. In turn, the pins in a SoC are managed by a pin controller, a subsystem

The work of the first, third and fourth authors has been partially supported by the European Commission through project FP7-SEC-607093-PREEMPTIVE funded by the 7th Framework Program.

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 1–12, 2017.
https://doi.org/10.1007/978-3-319-71368-7_1

of SoC, through which one can configure pin configurations such as the input or output mode of pins. One of the most peculiar aspects of a pin controller is that its behavior is determined by a set of registers: by altering these registers one can change the behavior of the chip in a dramatic way. This feature is exploitable by attackers, who can tamper with the integrity or the availability of legitimate I/O operations, factually changing how a PLC interacts with the outside world. Based on these observations, in this paper, we introduce a novel attack technique against PLCs, which we call pin control attack. As we will demonstrate in the paper, the salient features of this new class of attacks are: (a) it is intrinsically stealth. The alteration of the pin configuration does not generate any interrupt, preventing the Operating System (OS) to react to it. (b) it is entirely different in execution from traditional techniques such as manipulation of kernel structures or system call hooking, which are typically monitored by anti-rootkit protection systems. (c) it is viable. It is possible to employ it to mount actual attacks against process control systems.

To substantiate these points, we demonstrate the capabilities offered by Pin Control attack, together with technical details and the minimal requirements for carrying out the attack in Sect. 3.

To demonstrate the practical feasibility of our attack technique, in Sect. 4 we describe the practical implementation of an attack against a PLC environment by exploiting the runtime configuration of the I/O pins used by the PLC to control a physical process. The attack allows one to reliably take control of the physical process normally managed by the PLC, while remaining stealth to both the PLC runtime and operators monitoring the process through a Human Machine Interface, a goal much more challenging than simply disabling the process control capabilities of the PLC, which would anyway lead to potentially catastrophic consequences. The attack does not require modification of the PLC logic (as proposed in other publications [19,20]) or traditional kernel tampering or hooking techniques, which are normally monitored by Host-based Intrusion Detection systems. In Sect. 5.1 we discuss types of industrial network and utilities which can be affected by our attack. We also describe the consequences of the pin control attack on a specific utility. Additionally, In Sect. 5.2 we discuss potential mechanisms to detect/prevent Pin Control exploitation. However, because the pin configuration happens legitimately at runtime and the lack of proper hardware interrupt notifications from the SoC, it seems non-trivial to devise monitoring techniques that are both reliable and sufficiently lightweight to be employed in embedded systems. In Sect. 6 we discuss about similar works on attacking the I/Os. Finally we conclude our work in Sect. 7.

2 Background

For an attacker the ultimate objective when attacking an industrial control network is to manipulate the physical process without being detected [1] by advanced intrusion detection systems (IDS) or plant operators. Before the highly publicized Stuxnet malware, most of the attacks were trivial intrusions against

the IT equipment of industrial control network. However, the Stuxnet malware has intensified a race to the bottom where low-level attacks have a tactical advantage [2] and are therefore preferred [23]. PLCs play a significant role in the industry since they control and monitor industrial processes in critical infrastructures [13]. Successful low-level exploitation of a PLC can affect the physical world and, as a result, can have serious consequences [14]. There are few low-level techniques against PLCs which can help an attacker to reach his malicious objective. We can distinguish these techniques into the following major groups:

- *Configuration manipulation attacks:* these attacks allow an adversary to modify critical configuration parameters of a PLC (e.g. Logic) to alter the controlled process. Various research shown the feasibility of such attacks [8,19,20] against PLCs. To defeat this attack, the industry is using logic checksums and IDS [2,18,21].
- *Control-flow attacks:* in general, this category of attacks is achieved by exploiting a memory corruption vulnerability (e.g. buffer overflow), which allows the execution of arbitrary code by an adversary. Recent research has shown the possibility of control-flow attacks against PLCs [9,10,27]. Although several techniques have been proposed to detect or prevent control-flow attacks on general IT systems, effective countermeasures that are simultaneously applicable to the PLCs have yet to be developed.
- *Malicious code within the PLC:* possibility of remote code execution [9,26] in a PLC paves the way for an attacker to install malicious software. To counter this problem Reeves et al. [23] proposed Autoscopy Jr. Autoscopy Jr protects the PLC kernel from malicious software which modify or hooks the functions within the OS. Autoscopy Jr limits the attacker capability to target the PLCs kernel since almost all existing malicious attack needs a function hooking to operate stealthily. Our attack does not hook any function and therefore, will be invisible to Autoscopy Jr or similar techniques.

2.1 Pin Control Subsystem

In SoCs that are used in PLCs, pins are bases that are connected to the silicon chip. Each pin individually and within the group is controlled by a specific electrical logic with a particular physical address called a register. For example, "Output Enabled" logic means that the pin is an output pin and "Input Enabled" logic means that the pin is an input pin. In PLCs these logic registers are connected to "register maps" within the PLCs SoC and can be referenced by the OS. These "Register maps" are a mere translation of physical register addresses in the SoC to reference-able virtual addresses in the PLCs operating system. The concept of controlling these mapped registers with a software is called Pin Control. Pin control mainly consists of two subsystems namely Pin Multiplexing and Pin Configuration. Pin Multiplexing is a way to connect multiple peripherals within the SoC to each individual pin. Pin configuration is a process in which the OS or an application must prepare the I/O pins before using it.

2.2 How PLCs Control the Pins

The main component of a PLC firmware is a software called *runtime*. The runtime interprets or executes process control code known as *logic*. The logic is a compiled form of the PLC's programming language, such as function blocks or ladder logic. Ladder logic and function block diagrams are graphical programming languages that describe the control process. A plant operator programs the logic and can change it when required. The purpose of a PLC is to control field equipment (i.e., sensors and actuators). To do so, the PLC runtime interacts with its I/O. The first requirement for I/O interaction is to map the physical I/O addresses (including pin configuration registers) into virtual memory. The mapping process is usually carried by the OS or PLC runtime.

After mapping pin configuration registers, the PLC runtime executes the instructions within the logic in a loop (the so-called program scan). In a typical scenario, the PLC runtime prepares for executing the logic at every loop by scanning its inputs (e.g., the I/O channels defined as inputs in the logic) and storing the value of each input in the variable table. The variable table is a virtual table that contains all the variables needed by the logic: setpoints, counters, timers, inputs and outputs. During the execution, the instructions in the logic changes only values in the variable table: every change in the I/O interfaces is ignored until the next program scan. At the end of the program scan, the PLC runtime writes output variables to the mapped I/O virtual memory that eventually is written to the physical I/O memory by the OS Kernel. Figure 1 depicts the PLC runtime operation, the execution of the logic, and its interaction with the I/O. For example, the PLC runtime will put some pins of the PLC into input mode (for inputs and reading values) and some other pins into output pins (to control the equipments).

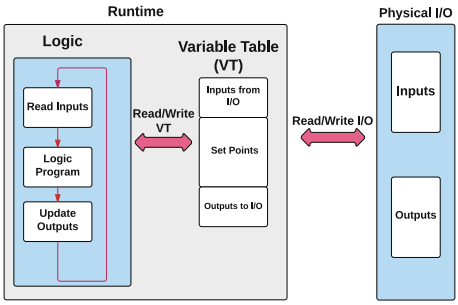


Fig. 1. Overview of PLC runtime operation, the PLC logic and its interaction with the I/O

3 Pin Control Attack

3.1 Security Concerns Regarding Pin Control

The Pin Control subsystem lacks of any hardware interrupt support. This raises a security concern about how the OS knows about the modification of pin configuration. Lack of interrupts when the configuration of a pin changes means neither the driver/kernel nor the runtime will notice about it when an attacker directly write to the pin registers. For example, in a PLC, if a pin (e.g. GPIO) that is set in Output mode gets reconfigured to Input mode by a malicious application, since there is no interrupt to alert the PLC OS about the change in pin configuration, the driver or kernel will assume that the pin is still in output mode and will attempt the write operation without reporting any error. However in reality the PLC SoC ignores the write operation (since the pin is in input mode) but will not give any feedback to the OS that the write operation was failed (because no interrupt is implemented). Therefore, the OS will assume that the write operation was successful while exactly opposite case happened. The OS then informs the PLC runtime that the operation was successful (which is wrong).

We have brought this security concern to the attention of the Linux Kernel Pin Control Subsystem group, which has confirmed our findings and its dangerous consequences for PLCs, but could not suggest a viable solution that did not require expensive artifacts such as ARM TrustZone or TPMs (Trusted Platform Module) in a PLC.

3.2 Pin Control Attack Details

A pin control attack basically consists of misusing the pin configuration functionalities of the PLCs SoC at runtime. An attacker can manipulate the values read or written to/from a peripheral by a legitimate process in the PLC without the PLC even noticing about those changes. As described in Sect. 3.1 if the I/O is in input mode and one tries to write to it, the I/O pin will ignore this request without raising an exception/error. Even if one change the I/O input/output status there will be no alert from the hardware for this change. Therefore, an attacker can manipulate the PLCs read and write operations to its I/Os by leveraging the configuration of pins as follows:

1. *For write operations*: if the PLC software is attempting to write a value to an I/O pin that is configured as output, the attacker reconfigures the I/O pin as input. The PLC runtime write to the input mode pin register in mapped virtual address. The PLC runtime write operation will not succeed, but the runtime will be unaware of the failure and return success since the runtime could carry out write operation in virtual memory. However, as mentioned earlier the PLCs SoC will ignore the write operation, since the configuration of the pin is in input mode.

2. *For read operations*: if the PLC software is attempting to read a value from an I/O pin that is configured as input, the attacker can reconfigure the I/O pin as output and write the value that he wishes to feed to the PLC software in the reconfigured pin.

This attack can have significant consequences due to two reasons: first, because it can be used to alter the way a PLC interacts (and possibly controls) the physical process. Second, because the attack is stealth in its nature since the PLC runtime will never notice about changes in the configuration due to lack of hardware interrupts for Pin Control system within PLCs SoC. To substantiate the feasibility of such attack we provide a practical implementation in Sect. 4.

3.3 Threat Model

In a critical physical process even randomly modifying the I/O can have a malicious consequence to the critical physical process. But when an attacker wants to remain stealth while maximizing the damage to the physical process then the pin control attack can be the most effective option. For pin control attack we consider three requirements which an attacker must satisfy. The requirements are the followings:

- *PLC runtime privilege:* we can envision an attacker with an equal privilege as the PLC runtime process which gives her the possibility to modify the pin configuration registers. Since the PLC runtime can modify such configuration registers, having equal privilege means that an attacker can also modify those registers. In recent years, multiple research has shown that the PLCs from multiple vendors such as Siemens [3,26], ABB [9], Schneider Electric [10,11], Rockwell Automation [12], and WAGO [6] are vulnerable to system-level code execution via the memory corruption vulnerabilities. Therefore, we can argue that getting equal privilege as PLC runtime is not a farreaching assumption.
- *Knowledge of the physical process:* we also assume that the attacker is aware of the physical process on the plant. Attacking critical infrastructure is usually carried out by state-sponsored attackers and usually such actors study their targets before launching their attack. For example, as reported [15] in Stuxnet [8] case, the attackers were very well aware of the physical process in their target plant. Therefore, we can assume that it is feasible for other state-sponsored attackers to study their target physical process and be aware of it.
- *Knowledge of mapping between I/O pins and the logic:* we assume that the attacker is aware of the mapping between the I/O pins and the logic. The PLC logic might use various inputs and outputs to control its process; thus, the attacker must know which input or output must be modified to affect the process as desired. The mapping between I/O pins and logic is already available in the PLC logic and therefore, an attacker can access to it within the PLC without any limitation. Additionally, the works presented by McLaughlin et al. [19,20] can be used to discover the mapping between the

different I/O variables and the physical world. Thus we can argue that it is reasonable to assume the attacker can be aware of the mapping between I/O pins and the logic.

4 A Pin Control Attack in Practice

4.1 Environment Setup

Target Device and Runtime. To mimic a PLC environment we choose a Raspberry Pi 1 model B as our hardware, because of the similarity in CPU architecture, available memory, and CPU power to a real PLC. For PLC runtime we use the Codesys platform. Codesys is a PLC runtime that can execute ladder logic or function block languages on proprietary hardware and provides support for industrial protocols such as Modbus and DNP3. The operating system of the Raspberry Pi is a Linux with real-time patch identical to the Wago PLC families. Currently, more than 260 PLC vendors use Codesys as the runtime software for their PLCs [6]. The combination of features offered by the Raspberry Pi and the Codesys runtime make such a system an alternative to low-end PLCs. The Raspberry Pi includes 32 general-purpose I/O pins, which represent the PLC's digital I/Os. These digital I/Os can also control analog devices by means of various electrical communication buses (e.g., SPI, I2C, and Serial) available for the Raspberry Pi.

The Logic and the Physical Process. We use pins 22 and 24 of the Raspberry Pi to control our physical process. In our logic, we declare pin 22 as the output pin and pin 24 as the input pin. In the physical layout, our output pin is connected to an LED and our input pin is connected to a button. According to our logic, the LED turns on and off every five seconds. If someone is pressing the button, the LED simply maintains its most recent state until the button is released, at which time the LED begins again to turn on and off every five seconds.

4.2 Attack Implementation

In this implementation we assume that the attacker has the same privileges as the PLC runtime. This is achievable for example by exploiting a memory corruption vulnerability that allows remote code execution, such as a buffer overflow [4,12,27]. A remote code execution vulnerability of such kind is known to be affecting the Codesys runtime [28]. The implementation consists of an application written in C that can be converted and used in an remote code execution exploit against a PLC runtime (in our example Codesys). The application uses /dev/mem, sysfs, or a legitimate driver call to access and configure the pins. In our target platform the Codesys runtime uses the /dev/mem for I/O access, therefore, our attack uses the same I/O interface. The application checks whether the processor I/O configuration addresses are mapped in the PLC runtime.

The list of all mapped addresses is system wide available in various locations for any user space application (e.g. via /proc/modules, or /proc/$pid/maps).

For manipulating write operations, the application needs to know a reference starting time. This is the relative time where the PLC runtime writes to the pin. While the application knows the logic and is aware that every five seconds there is a write operation to pin 22, it does not know at what second the last write operation happened. This can be easily found by monitoring the value of pin 22. Once the application intercepts the correct reference starting time, for every write operation in the logic it will carry out two tasks. First, right before the reference starting time (which is when the PLC runtime will start writing its desired original value to the I/O) the application reconfigures the pin to input mode. The Codesys runtime then attempts to write to the pin. However, the write operation will be ineffective, since the pin mode is set to input. Our application then switches the pin mode to output and writes the desired value to it. Manipulating read operations are almost identical to the write manipulation, except that the application changes the state of the pin from input to output and writes it constantly with the desired value. With this implementation, we can successfully manipulate the process. The LED turns on and off every ten seconds instead of five. Additionally, we can completely control input pin 24 and make its value 0 or 1 whenever we wanted, while the Codesys runtime was reading our desired value. Also, Codesys never notices about the failures of write operations (turning on or off the LED) whenever we modify pin configuration registers due to the fact that there is no interrupt to alert the Codesys about write operation failure. Beside that both the Codesys runtime and the PLC logic is untouched in our attack, therefore any mechanisms that checks the integrity of the PLC key components, applications and configurations would never notice about the attack.

This implementation is significantly lightweight and only causes a two percent CPU overhead. There is, however, a small chance that a race condition happens during read manipulation. However, in our tests, this race condition never happened.

5 Discussion

5.1 Implications of Attack on the ICS

The pin control attack can be used against PLCs in various utilities, but it can be considerably more powerful and stealthier when it is used against specific type of industrial processes where rapid control is not essential. The reason to choose slower processes is to reduce the chance of race conditions in the pin control attack. To understand which industrial processes mounts better for our attack we studied three different utilities namely electrical, gas, and water utilities. We created a reference taxonomy for each individual utility [22]. We identified their individual sub processes and the field equipment they most commonly used in the plant (including types of PLCs). We then confirmed our reference taxonomy by having interviews with different utilities and control engineers. According the

reference taxonomy we concluded that one of the suitable targets for pin control attack would be water utilities. Based on our interview sessions with water utility experts we identified the water distribution network as one of the most likely target for attackers who want to disrupt the water delivery service.

Disrupting the Water Distribution Pipes. The water distribution network consists of pipes, water tanks and pumps, and is instrumented with multiple sensors, such as pressure sensors and meters. Several components of the water distribution network are controlled by the water distribution automation system. One possible technique to disrupt water distribution pipes is based on creating a so-called water hammer effect. A water hammer effect is a pressure surge or wave caused when a fluid in motion is forced to stop or change direction suddenly. The water hammer effect can cause major problems, ranging from noise and vibration to pipe collapse. To mitigate the water hammer effect, water utilities equip the pipeline with shock absorbers and air traps. In addition, besides the physical countermeasures against the water hammer effect, the water distribution automation system has built-in safety logic to prevent actions that could cause a water hammer effect. However, both the physical countermeasures and the automation system are not designed to counter the effects of an intentional attack. In case safety controls are lost during a cyber-attack, valves can be controlled to cause a water hammer effect that can break the pipes in the distribution network.

Figure 2 depicts the attack graph that describes this attack scenario. In the first three paths, the attacker needs to exploit the PLC programming stations (similarly to Stuxnet) to alter the process and cause the damage. Regarding

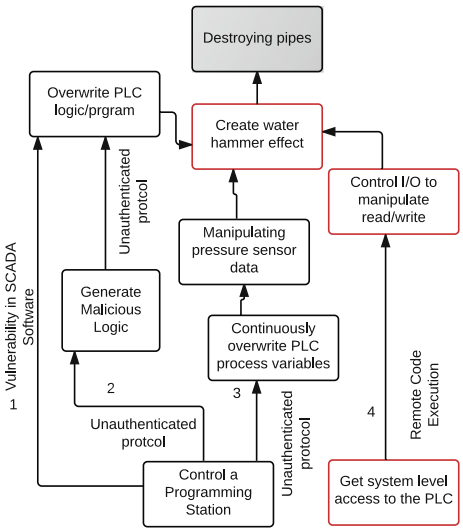


Fig. 2. Attack graph describing the water distribution pipe destruction scenario.

the stealthiness of the attack, all of those attacks rely on attacking the SCADA station infection. To remain hidden attacker needs to suppress various protection mechanism [17,29] in an SCADA station. Contrary, the attacker in the fourth path does not need to modify PLC logic, firmware or suppress any protection system. Also, even those few techniques exist to detect malicious behavior within the PLC [5,23] are not effective against our attack. Also due to little performance overhead imposed by the Pin Control attack it is difficult to tell whether the PLC is targeted by the attack or not. What actually the attacker does with the pin control attack will be based on what described by Larson [16] where the PLC will suddenly close the valves of the water distribution pipes with network of infected PLCs while the operator will still sees the fake profile of the valve speed or even can be tricked to see the state of the valves are open while in reality the valves are opening and closing. The reading of other sensors regarding the pipe pressure and heat level of the pipes (due to water hammer generated heat) can be altered with the same network of the infected PLC using pin control attack.

5.2 Detection of Pin Control Attack

One might believe that it is relatively simple to devise countermeasures that could detect and possibly block pin control attacks. In this section we enumerate three of these possible countermeasures and discuss their effectiveness and practical applicability to the PLCs.

1. *Monitoring the mapping of pin configuration registers*: an attacker needs to use the virtual addresses of the pin configuration registers to write to them. To do so, the attacker needs to either map the physical registers by herself or use already mapped addresses. This is not the case in PLCs since the PLC is already using the target I/O. Therefore, an attacker can use already mapped register addresses to carry the attack.
2. *Monitoring the change of pin configuration registers*: one may detect our attack by monitoring the frequency at which pin configuration registers are changed. This may be challenging for two reasons. First, since changes in configuration registers do not generate hardware interrupts, therefore, an attacker will be able to bypass monitoring mechanisms. Second, since pins get re-configured legitimately by a PLC, it may be difficult to tell with reliable accuracy whether a sequence of changes is legitimate or not.
3. *Using a trusted execution environment*: the reliable solution to prevent all pin control attacks would be running a micro kernel in a trusted zones (e.g. an ARM TrustZone) within the kernel to verify write operations on configuration pins. However, as confirmed by the Linux Kernel Pin Control Subsystem group, using TrustZone for I/O operations would cause a significant performance overhead.

6 Related Work

Few research focused on the possibility of system level attack against PLCs. A relevant stream of work has explored memory corruption vulnerabilities

against PLCs [27] which is the closest thing to attacking the PLCs at operating system level. Part of our work bears some similarities with System Management Mode (SMM) rootkits [7,24,25] for X86 architectures. These rootkits tap the system I/O, similarly to what we did in our Pin Control attack. However, the modification of system I/O in SMM causes interrupts which need to be suppressed by SMM rootkits, typically by attacking kernel interrupt handlers. In our case, this operation is not needed due to the lack of interrupts for pin configuration.

7 Conclusion

In this paper, we first looked into the pin control subsystem of embedded systems. We found that the lack of hardware interrupts for pin control subsystem brings an opportunity for the attacker. We showed that it is practical to target the Pin Control subsystem by implementing an attack against a PLC and manipulate a process in which it controlled. The result shows that attackers can stealthy manipulate the I/O of the PLCs without using traditional attack techniques such as function hooking or OS data structure modification. We now plan to investigate possible defensive techniques against Pin Control attack. We believe that defending PLCs against this new front will pose a notable hindrance to attackers, significantly reducing their success rate in the future.

References

1. Abbasi, A., Wetzels, J., Bokslag, W., Zambon, E., Etalle, S.: On emulation-based network intrusion detection systems. In: Stavrou, A., Bos, H., Portokalidis, G. (eds.) RAID 2014. LNCS, vol. 8688, pp. 384–404. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-11379-1_19
2. Basnight, Z., Butts, J., Lopez Jr., J., Dube, T.: Firmware modification attacks on programmable logic controllers. Int. J. Crit. Infrastruct. Prot. **6**(2), 76–84 (2013)
3. Beresford, D.: Exploiting siemens simatic S7 PLCs. In: Black Hat USA (2011)
4. Beresford, D., Abbasi, A.: Project IRUS: multifaceted approach to attacking and defending ICS. In: SCADA Security Scientific Symposium (S4) (2013)
5. Cui, A., Stolfo, S.J.: Defending embedded systems with software symbiotes. In: Sommer, R., Balzarotti, D., Maier, G. (eds.) RAID 2011. LNCS, vol. 6961, pp. 358–377. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-23644-0_19
6. DigitalBond: 3S CoDeSys, Project Basecamp (2012). http://www.digitalbond.com/tools/basecamp/3s-codesys/
7. Embleton, S., Sparks, S., Zou, C.C.: SMM rootkit: a new breed of os independent malware. Secur. Commun. Netw. **6**(12), 1590–1605 (2013)
8. Falliere, N., Murchu, L.O., Chien, E.: W32. stuxnet dossier. White paper, Symantec Corp., Security Response 5 (2011)
9. ICS-CERT: Abb ac500 plc webserver codesys vulnerability (2013). https://ics-cert.us-cert.gov/advisories/ICSA-12-320-01
10. ICS-CERT: Schneider electric modicon quantum vulnerabilities (update b) (2014). https://ics-cert.us-cert.gov/alerts/ICS-ALERT-12-020-03B

11. ICS-CERT: Schneider electric modicon m340 buffer overflow vulnerability (2015). https://ics-cert.us-cert.gov/advisories/ICSA-15-351-01
12. ICS-CERT: Rockwell automation micrologix 1100 plc overflow vulnerability (2016). https://ics-cert.us-cert.gov/advisories/ICSA-16-026-02
13. Igure, V.M., Laughter, S.A., Williams, R.D.: Security issues in SCADA networks. Comput. Secur. **25**(7), 498–506 (2006)
14. Koopman, P.: Embedded system security. Computer **37**(7), 95–97 (2004)
15. Langner, R.: To kill a centrifuge: A technical analysis of what stuxnets creators tried to achieve (2013). http://www.langner.com/en/wp-content/uploads/2013/11/To-kill-a-centrifuge.pdf
16. Larsen, J.: Physical damage 101: bread and butter attacks. In: Black Hat USA (2015)
17. Liang, Z., Yin, H., Song, D.: HookFinder: identifying and understanding malware hooking behaviors. In: Proceeding of the 15th Annual Network and Distributed System Security Symposium (NDSS 2008) (2008). http://bitblaze.cs.berkeley.edu/papers/hookfinder_ndss08.pdf
18. Maxino, T.C., Koopman, P.J.: The effectiveness of checksums for embedded control networks. IEEE Trans. Dependable Secure Comput. **6**(1), 59–72 (2009)
19. McLaughlin, S., McDaniel, P.: SABOT: specification-based payload generation for programmable logic controllers. In: Proceedings of the 2012 ACM Conference on Computer and Communications Security, CCS 2012, pp. 439–449. ACM, New York (2012)
20. McLaughlin, S.E.: On dynamic malware payloads aimed at programmable logic controllers. In: HotSec (2011)
21. Peck, D., Peterson, D.: Leveraging ethernet card vulnerabilities in field devices. In: SCADA Security Scientific Symposium, pp. 1–19 (2009)
22. PREEMPTIVE-Consortium: Reference taxonomy on industrial control systems networks for utilities (2014). http://preemptive.eu/wp-content/uploads/2015/07/preemptive_deliverable-d2.3.pdf
23. Reeves, J., Ramaswamy, A., Locasto, M., Bratus, S., Smith, S.: Intrusion detection for resource-constrained embedded control systems in the power grid. Int. J. Crit. Infrastruct. Prot. **5**(2), 74–83 (2012)
24. Schiffman, J., Kaplan, D.: The smm rootkit revisited: fun with USB. In: 9th International Conference on Availability, Reliability and Security (ARES), pp. 279–286 (2014)
25. Sparks, S., Embleton, S., Zou, C.C.: A chipset level network backdoor: bypassing host-based firewall & IDS. In: Proceedings of the 4th International Symposium on Information, Computer, and Communications Security, pp. 125–134. ACM (2009)
26. Spenneberg, R., Brüggemann, M., Schwartke, H.: PLC-blaster: a worm living solely in the PLC. In: Black Hat Asia (2016)
27. Wightman, R.: Project basecamp at s4. SCADA Security Scientific Symposium (2012). https://www.digitalbond.com/tools/basecamp/schneider-modicon-quantum/
28. Wrightman, K.R.: Vulnerability inheritance in PLCs. DEFCON 23 IoT Village (2015)
29. Yin, H., Song, D.: Hooking behavior analysis. In: Automatic Malware Analysis, pp. 43–58. Springer, New York (2013). https://doi.org/10.1007/978-1-4614-5523-3_5

Developing a Cyber Incident Communication Management Exercise for CI Stakeholders

Tomomi Aoyama(✉), Kenji Watanabe, Ichiro Koshijima, and Yoshihiro Hashimoto

Department of Architecture, Civil Engineering and Industrial Management Engineering, Nagoya Institute of Technology, Showaku, Gokiso, Nagoya, Aichi 4668555, Japan
{aoyama.tomomi,watanabe.kenji,koshijima.ichiro, hashimoto.yoshihiro}@nitech.ac.jp
http://shakai.web.nitech.ac.jp

Abstract. Existing cyber security training programs for Critical Infrastructures (CI) place much emphasis on technical aspects, often related to a specific sector/expertise, overlooking the importance of communication (i.e. the ability of a stakeholder to gather and provide relevant information). We hypothesise that the achievement of a secure and resilient society requires a shared protocol among CI stakeholders, that would facilitate communication and cooperation. In order to validate our hypothesis and explore effective communication structures while facing a cyber incident and during recovery, we developed a discussion-based exercise using an Industrial Control System (ICS) incident scenario, and implemented it in pilot workshops where a total of 91 experts participated. Results suggest there are three possible incident communication structures centered around the IT department, the production department, and management, respectively. In future, these structures can be used as the framework to build an ICS-Security Incident Response Team (ICS-SIRT), which would strengthen cooperation among CI stakeholders.

Keywords: CIP exercise · Cyber incident management · ICS security Communication management · Business continuity management

1 Introduction

1.1 Background

Cyber security training is a common measure to enhance the security capability of an organization. Numerous security *awareness training* programs are available for every type of expertise. Most of these training programs aim at educating basic knowledge about cyber protection and teaching how vulnerable a participant or an organization can be to cyber threats. Awareness training is important as a foundation of organizations' security capability, but it may not be enough. This is because they often focus only on the prevention of an incident,

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 13–24, 2017.
https://doi.org/10.1007/978-3-319-71368-7_2

and leave out cyber incident response. Therefore, the need for cyber security training *beyond awareness* is growing in the industry, and several key centres are providing training for cyber incident response. Training programs, such as the well-known *beyond awareness* adversarial Red team - Blue team Exercise, include classroom lectures and exercises to maximize the learning of technical skills with respect to cyber crisis management.

However, awareness and beyond awareness training programs focus on technical aspects and overlook the importance of soft skills in the management of a cyber incident. Indeed, in their annual cyber security awareness report, the SANS Institute claims that security personnel lacks soft skills [1], and that communication skills are among the most critical ones. In this context, communication skills are regarded as the ability to describe a critical situation effectively, to collect information relevant to an incident from other stakeholders, and to fit in an adaptive communication structure within the dynamics of a cyber attack. Based on on-site observation of the Red-Blue teams exercise, we found that many skillful engineers struggled to negotiate with others in an unorganized communication structure, or paid no attention to cooperation. This corroborates the hypothesis that lack of communication skills is a major issue in cyber incident management, and that the achievement of a secure and resilient society requires a shared protocol among CI stakeholders.

This paper especially focuses on cyber security training programs meant for those CI sectors that make use of ICS. However, given the generality of the proposed exercise, we believe the whole CI protection (CIP) community may benefit from it.

The next paragraph reviews the merits and demerits of the Red-Blue teams exercise. In the following section, a discussion-based exercise that aims at both enhancing communication skills and promoting cooperation among CI stakeholders is proposed, and its implementation is described in detail. The last section reviews the results of the implemented pilot exercises, and discuss their impact on the CIP community.

1.2 Case Study: Red Team - Blue Team Exercise

One of the leading cyber security incident response training in the field of ICS security is the ICS-CERT's 5 days training which includes a Red-team/Blue-team exercise. In this exercise, participants play the role of either the attacking (Red) or the defending (Blue) teams [2]. Similar adversarial exercises are provided by other key centres in the world, such as Queensland Institute of Technology in Australia [3,4], and European Network for Cyber Security (ENCS) in the Netherlands. The entire exercise is set up in a secure environment [5] for participants to experience how an organization can be compromised by a cyber attack. It should be noted that the exercise focuses on the impact of a cyber attack on a single organization, rather than on the whole CI stakeholder community. However, we believe that it represents one of the most recognized exercises in the field of ICS security. Therefore, we devote this subsection to the description of its characteristics.

Branlat et al. [6,7] studied the exercise operated by ICS-CERT, and pointed out that the realistic timeline of the exercise allows participants to simulate the complexity of incident handling. Encouraged by their work, we have been studying the dynamic adaptation of organizations' decision-making structures, by monitoring the training of ENCS [8,9]. Our on-site observation confirmed that the environment of the exercise provides valuable lessons regarding cyber incident management. Indeed, the reproducibility and the realistic timeline of the exercise allow participants to have an authentic experience. Moreover, it is a rare opportunity to establish technical skill-sets required in cyber defense, and to see how certain skills can impact the target system within the dynamics of a cyber attack. Arguably, one of the most noticeable strengths of the exercise is the heterogeneous background and expertise of the participants and facilitators. In fact, team-working among these professionals provides a new perspective to their mental model and enhances the impact of the training.

However, considering the technicality and the intensive nature of the exercise—even though it portrays the realistic speed of a cyber attack—, participants focus on their immediate task leaving little time for communication with each other, let alone for sharing ideas towards better incident management. As a result, the exercise does not explicitly provide a structured framework to learn about the importance of communication and cooperation among the different departments of an organization or across organization boundaries. Participants are not guided in understanding how an effective communication of their technical knowledge could influence the decision-making. Moreover, they are not taught to see the bigger picture, making it difficult for them to comprehend how dynamically the organization's communication structure should adapt to the timeline of a cyber attack.

2 Communication Management Exercise for ICS Security (CME-ICS)

Based on the realisation that soft skills are as important as hard skills in cyber incident management, we developed a discussion based ICS security exercise for improving communication management and creating a shared protocol in the community of CI stakeholders.

2.1 Peculiarity of Existing Japanese CIP Training

In Japan, there have been discussions about whether the Red-Blue teams exercise is necessary, and so far this format has not been adopted for domestic CI stakeholders. Conversely, there are several ICS security training programs that consist of class-room lectures and drills, which do not include active discussion among participants.

More importantly, participation to these training programs is restricted to certain expertise profiles or CI sectors (e.g. banking, chemical). However, large-scale cyber incident can cause an impact beyond boundaries of CI sectors in a

highly inter-connected society. In case of such an event, the cooperation of CI sectors and other stakeholders (e.g. government agencies) is essential [10]. Nevertheless, the current training system is isolated by sectors, and does not include stakeholders outside the organization. The results of such limited diversification of expertise are that the participants' perspective on cyber security issues is narrowed down, and that knowledge transfer across sectors is not facilitated.

2.2 Discussion-Based Exercise

Considering the sectorized nature of the existing Japanese CIP training programs, our aim is to develop an exercise that is open for any CI stakeholder, and that enables knowledge transfer among participants. This motivated the adoption of a *discussion-based* table-top exercise style, since it stimulates the discussion among participants with a large variety of backgrounds, allowing them to compare their views on an issue [11]. In addition, it is often used to develop new plans and procedures, focusing on strategic issues [12]. For all these reasons, it provides new perspectives to each participant's conceptual knowledge structure, and helps to build a shared mental model among them.

2.3 Theme of the Exercise: Communication Management

As previously mentioned, the lack of communication skills is a major issue in cyber incident management. Therefore, the exercise has the major objective of highlighting the importance of communication and cooperation among CI stakeholders. Specifically, the scenario represents a cyber incident within a simplified organization structure, where participants discuss and strategize countermeasures with a bird's-eye-view, that is without playing a specific role. This helps them understand the importance of effective communication among stakeholders, rather than focus excessively on technical aspects.

2.4 Scenario

The scenario of the exercise is based on our CI testbed (Fig. 1), that was originally built for the purpose of gaining public security awareness, as well as testing and developing security solutions for ICS [13]. The testbed consists of two plant systems, a controlling network for each plant, and a corporate network connecting the two control networks. Each plant is a closed hot water circulation system consisting of two tanks: water in the lower tank is heated by a heater, then circulated to the upper tank by a pump. With respect to the exercise, the testbed models a community heating/cooling facility of a fictitious company that provides services to two different areas [14]. Safety violations, such as spilling water or heating an empty tank, could not only damage the equipment and harm the personnel, but may cause the discontinuation of the plant.

The phases of the scenario follow the time line of incident handling proposed by Sheffi et al. [15]. They suggested that any significant disruption has a typical

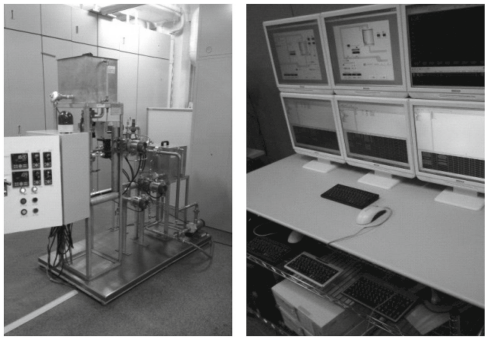


Fig. 1. The plant side (left) and the operator side (right) of the testbed. Testbed visitors can operate the system during the demonstration.

profile in terms of its effect on company performance. Moreover, the nature of the disruption and the dynamics of the company's response can be characterized by eight phases (Fig. 2). From the originally proposed, three phases were adopted in the exercise: first response to an disruptive event, preparation for recovery, and recovery. In the following paragraphs, the phases are described in detail, under the convention that *italicized* text represents the actual scenario descriptions provided to the participants.

Disruptive Event/First Response. The exercise starts when *an anomaly in network traffic is detected by the monitoring room* and *control room operators in Plant No. 2 notice unexpected value declaration in a level sensor.*

The goal of this phase is to determine that the incident is caused by a cyber attack, and that is not the result of either equipment or sensor failure. The participants discuss how to implement a cyber incident response for a transition

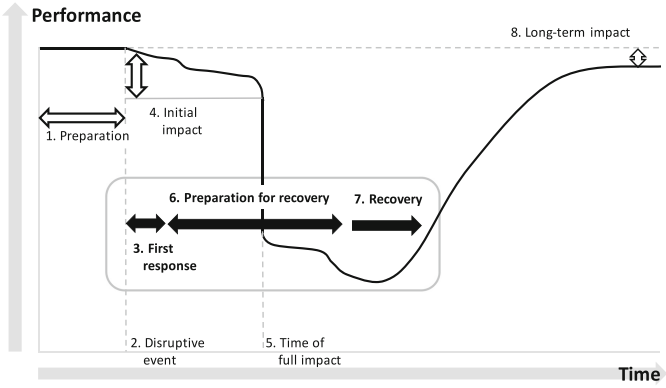


Fig. 2. Stages of disruption proposed by Sheffi [15], recreated by the authors.

to safe manual operation of the plant, and how IT and other departments can support the plant system to achieve safety.

Preparation for Recovery. The preconditions of this phase are that proofs of a cyber attack are confirmed (i.e. *no equipment/sensor malfunction detected*, *the configuration file of an OPC server in Plant No. 2 has been changed in an unauthorized manner*) and that *Plant No. 2 is operated manually*. The key decision-making in this phase is whether operation in Plant No. 2 should be shut down. Moreover, in case the plant is kept in manual operation, what measure should be taken to ensure safety. The participants discuss what kind of information is required to make a decision, if such information is available, and who has the authority to make a decision in this circumstance. They will conceive how to conduct business continuity management, in order to mitigate the further impact on business performance by the disruption. For example, what action should be taken at Plant No. 1 which is connected to Plant No. 2 through the corporate network, and what roles do the sales and public relations (PR) departments play.

Recovery. This phase assumes that the following conditions are met: *Plant No. 2 has been shut down* and *Plant No. 1 is operating without network connection (limited productivity)*. The task in this phase is to plan the efficient and safe plant reactivation based on the start up procedure manual. Additionally, participants review the past phases and discuss the measures to prevent a recurring failure.

As for the third and final phase of the exercise, the goal is to reexamine the balance of technical, management, and external cooperation capability to achieve high resiliency in the organization.

2.5 Exercise Steps

The exercise is composed of five steps: briefing, scene description, group work, discussion and debriefing. As mentioned in the previous section, the scenario is divided into three scenes (i.e. disruptive event/first response, preparation for recovery, and recovery). Therefore, scene description, group work and discussion are repeated as one cycle for each scene.

Briefing. At the beginning of the exercise, participants are divided into groups consisting of four to six members with different backgrounds. A facilitator introduces the group task and the general scenario. If needed, some ice breaker activities may be carried out to motivate all participants to become actively involved in the group work. Most importantly, the purpose of the exercise is shared with participants, so that they can all understand the significance of the activity.

Scene Description. As for the opening of each scene, the status of the plant and IT network system are revealed along with the (fictitious) organization's understanding of the situation. The scene is reenacted in a short video, which is used as visual aid.

Group Work. The group task is to create a work flow of actions that would solve the given situation. Each group is provided with a printed A0-sized worksheet, colored sticky-notes, and markers. On the worksheet, the columns of actors (e.g. IT dept., manufacturing dept., maintenance dept.) and the initial scenario injections are printed (Fig. 3). The list of actor names provided in the worksheet is not comprehensive, therefore participants are recommended to add/remove actor columns. At the beginning of each cycle, new worksheets including the scenario injections matching the current scene are distributed. The types of activity such as actor-system interaction (action) and actor-actor interaction (command) are color coded. In order to add an activity to the worksheet, a sticky-note of the matching color is used. In this way, the worksheet visualizes the flow of actors' actions and the organization's communication structure.

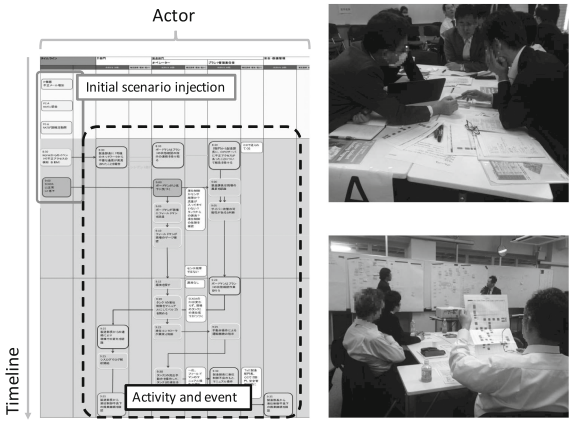


Fig. 3. The example of the worksheet (left) and pictures from the exercise (right) where participants engage in group work (top right) and present their group work at discussion time (bottom right). Participants' faces are blurred out for their privacy.

Discussion. The former process helps participants to create shared mental models within their group. On the other hand, discussion and debriefing are activities that create a shared mental model among all participants. Discussion is the final step of one cycle. Each group gives a short presentation of their work flow while displaying the worksheet to everyone. The members of other groups may raise some questions. In this way, participants compare their worksheets to discover similarities and differences among their subjective perspectives regarding the many degrees-of-freedom of the scenario (e.g. likelihood of an event, consequence of an action).

Debriefing. To conclude, the goals of the exercise are revisited, and participants share results and lessons learned. This activity helps the organizers to evaluate if the exercise method was appropriate, and more importantly, if the intended learning outcomes are achieved.

2.6 Administration Staff

The size and complexity of the exercise required a large number of personnel for assisting the exercise facilitation. For a smooth administration, the role were divided as follows: facilitator, adviser, and replier.

Facilitator. The facilitator guides participants through the exercise. He/she explains the exercise at briefing, and describes the scene at each cycle. During the group work, the facilitator pays attention to each group's progress, while keeping track of time. He/she also supervises the discussion and debriefing. In debriefing, he/she helps participants to summarize results and lessons learned.

Adviser. During group work, the adviser walks among tables and gives suggestions to each group based on his/her expertise. He/she also asks questions that trigger more actions and discussion. Therefore, the role requires knowledge and experience in the field. During discussion, the adviser provides positive feedback and comments for each group. We invited IT security specialists, ICS security researchers, and experts from ICS security agencies as advisers. These experts also helped during the process of scenario development.

Replier. The role of the replier is to reply the emails from each group as a (fictitious) "company employee" and to supplement the scenario. Participants cannot touch the system by themselves, given the nature of the table-top exercise. Therefore, some of the participants' emails are requests for additional information, while others are requests for taking action.

2.7 Pilot Exercises

Pilot exercises were conducted at the campus of Nagoya Institute of Technology as a part of two days ICS security workshop in August 2015 and March 2016.

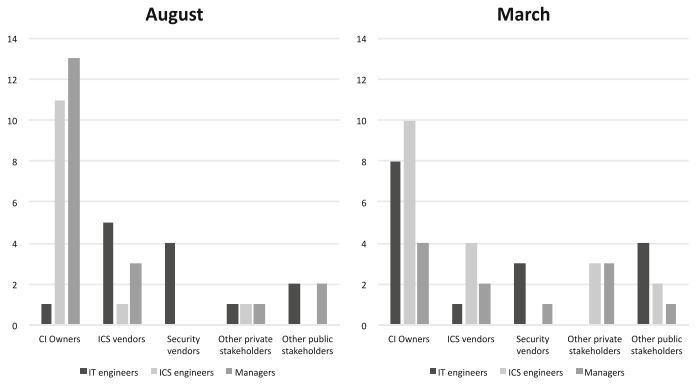


Fig. 4. Participant's profile distribution.

The number of participants was 45 and 46 respectively, and their expertise was heterogeneous. The distribution of participants' profiles at each workshop is shown in Fig. 4, where participants are classified by their organization types and occupational category. The sectors of CI owners included chemical, energy, gas, and telecommunication. In both exercises, participants were divided into six groups—totalling twelve groups—in order to facilitate the discussion. Since the exercise aims at stimulating the discussion and expand the participants' perspective, groups were carefully composed in order to maximise intra-group heterogeneity of expertise, background and position.

3 Results and Discussion

3.1 Variation of Incident Management Structure

The groups' worksheets were analyzed at discussion and debriefing time, by comparing the structure of their actions and commands. As a result, the following three types of incident management structures were found: IT department centered, production department centered, and management centered.

IT Department Centered. The IT department plays the leading role during the incident management. Specifically, it investigates the incident, and gives directions to the production department. Additionally, it provides updates about the situation to the management and to other departments who may be affected by the incident (e.g. sales, PR). In a real life situation, this structure may be applicable to an organization with a strong IT capability. Moreover, for the IT department to successfully lead the cyber incident response, it should have knowledge of the plant systems and a full understanding of the incident's impact on the business.

Production Department Centered. The production department leads the response, cooperates with the IT and other departments in charge of maintaining the production (e.g. the maintenance department), and gathers information related to the investigation and to the situation of the damage. This structure may be suitable for a large plant system where the production department has a strong leadership and authority. However, if the production department is unprepared to handle a cyber incident, the investigation may take longer than necessary, and potentially cause a bigger impact. Therefore, a thorough cyber security training of the production department personnel is necessary for this structure.

Management Centered. The management department leads the operation, by keeping an exclusive communication with the IT and the production departments, which don't directly exchange information with each other. One group even suggested to set up a crisis management headquarter, where all department

and management heads would cooperate. This structure is similar to the incident command system adopted for natural disasters [16], where plans and objectives are decided at the top of the hierarchy, while activities at the lower levels are a consequence of those decisions. In reality, this structure may be applicable to an organization with a highly centralized management system, or to a situation that requires the involvement of top management (e.g. large scale disaster, the critical service is not substitutable).

3.2 Results of the Survey

A survey was conducted after each pilot exercise. The results show that 94.7% (in August) and 90.9% (in March) of the participants were satisfied with the exercise, and that 83.0% (in August) and 90.6% (in March) would recommend the exercise to other CI stakeholders. In fact, some of the August workshop participants participated in the March workshop as well, and some extended the invitation to their colleagues.

3.3 Discussion

The proposed exercise aimed at training communication management skills of CI stakeholders and strengthen the cooperation capability of the CIP community, by engaging participants in discussion. We could observe that participants were stimulated by the exercise to express their point of view, acknowledge variety, and achieve a mutual understanding of an issue, regardless of their background. It can be said that the exercise encourages CI stakeholders to cultivate a shared mental model, which may positively influence performance [17]. Moreover, the exercise was general enough to stimulate the participants who did not belong strictly to the ICS security community (i.e. telecommunication sector), who in turn were satisfied by the acquisition of new knowledge. In conclusion, the unique experience of the exercise was appreciated by the CIP community.

As to the limitations of the current study, the evaluation of the pilot exercise was based on a subjective analysis. However, the overlap ('sharedness') of mental models can be explored using network analysis [18]. In future studies, the employment of objective evaluation techniques will be taken into consideration.

3.4 Future Work: "ICS-SIRT" Exercise

Based on consultation with the exercise participants, we realised that most organizations in the ICS field are not endowed with a tailored cyber incident procedure yet, and current incident management systems in the industry often miss coordinating capabilities. On the other hand, IT security organizations have adopted the Cyber Security Incident Response Team (CSIRT) as a common measure against cyber attacks. CSIRT is a team of IT security experts whose main duty is to mitigate, prevent, and respond to computer security incidents [19]. With the same philosophy, we believe that an ICS Security Incident Response

Team (ICS-SIRT), a team of ICS security experts, should be devised. In this context, the incident management structures proposed by the exercise participants reflect the possible communication structure of ICS-SIRT. Indeed, they are independent from the specific characteristics of CI sectors, which makes them suitable for any type of organization. Each organization would establish its own ICS-SIRT, which has a consistent communication structure across sectors and is connected with ICS-SIRTs of other organizations, strengthening the horizontal cooperation among CI stakeholders.

Future studies will explore the possibility of expanding the proposed exercise towards the development ICS-SIRT inside the organizations affiliated with Nagoya Institute of Technology.

Acknowledgements. This research is partially supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Scientific Research (A), No. 16H01837 (2016); however, all remaining errors are attributable to the authors.

References

1. SANS Institute: 2016 Security Awareness Report. SANS Institute (2016). http://securingthehuman.sans.org/resources/security-awareness-report
2. Department of Homeland Security: Training available through ICS-CERT. https://ics-cert.us-cert.gov/Training-Available-Through-ICS-CERT#workshop
3. Sitnikova, E., Foo, E., Vaughn, R.B.: The power of hands-on exercises in SCADA cyber security education. In: Dodge, R.C., Futcher, L. (eds.) WISE 2009. IAICT, vol. 406, pp. 83–94. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39377-8_9
4. Foo, E., Branagan, M., Morris, T.: A proposed Australian industrial control system security curriculum. In: 2013 46th Hawaii International Conference on System Sciences (HICSS), pp. 1754–1762. IEEE (2013)
5. European Network for Cyber Security: E.ON teams get trained on ICS and smart grid cyber security during the ENCS red team blue team course—ENCS. https://www.encs.eu/2015/11/10/
6. Branlat, M.: Challenges to adversarial interplay under high uncertainty: staged-world study of a cyber security event. Ph.D. thesis, The Ohio State University (2011)
7. Branlat, M., Morison, A., Finco, G., Gertman, D., Le Blanc, K., Woods, D.: A study of adversarial interplay in a cybersecurity event. In: Proceedings of the 10th International Conference on Naturalistic Decision Making (NDM 2011), 31 May–3 June 2011
8. Aoyama, T., Naruoka, H., Koshijima, I., Watanabe, K.: How management goes wrong? The human factor lessons learned from a cyber incident handling exercise. Procedia Manuf. **3**, 1082–1087 (2015). 6th International Conference on Applied Human Factors and Ergonomics (AHFE 2015) and the Affiliated Conferences, AHFE 2015. http://www.sciencedirect.com/science/article/pii/S2351978915001791
9. Aoyama, T., Naruoka, H., Koshijima, I., Machii, W., Seki, K.: Studying resilient cyber incident management from large-scale cyber security training. In: 2015 10th Asian Control Conference (ASCC), pp. 1–4. IEEE (2015)

10. Watanabe, K.: Developing public-private partnership based business continuity management for increased community resilience. J. Bus. Contin. Emerg. Plann. **3**(4), 335–344 (2009)
11. Borell, J., Eriksson, K.: Learning effectiveness of discussion-based crisis management exercises. Int. J. Disaster Risk Reduct. **5**, 28–37 (2013). http://www.sciencedirect.com/science/article/pii/S2212420913000332
12. US Department of Homeland Security and United States of America: Homeland security exercise and evaluation program (HSEEP) volume I: HSEEP overview and exercise program management (2007)
13. Aoyama, T., Koike, M., Koshijima, I., Hashimoto, Y.: A unified framework for safety and security assessment in critical infrastructures. In: Safety and Security Engineering V. Witpress Ltd., September 2013. http://dx.doi.org/10.2495/SAFE130071
14. Takagi, H., Morita, T., Matta, M., Moritani, H., Hamaguchi, T., Jing, S., Koshijima, I., Hashimoto, Y.: Strategic security protection for industrial control systems. In: 2015 54th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE), pp. 986–992. IEEE (2015)
15. Sheffi, Y., Rice Jr., J.B.: A supply chain view of the resilient enterprise. MIT Sloan Manag. Rev. **47**(1), 41 (2005)
16. Bigley, G.A., Roberts, K.H.: The incident command system: high-reliability organizing for complex and volatile task environments. Acad. Manag. J. **44**(6), 1281–1299 (2001)
17. Converse, S.: Shared mental models in expert team decision making. In: Castellan, N.J. (ed.) Individual and Group Decision Making: Current Issues, p. 221. Lawrence Erlbaum, Hillsdale (1993)
18. Mathieu, J.E., Heffner, T.S., Goodwin, G.F., Salas, E., Cannon-Bowers, J.A.: The influence of shared mental models on team process and performance. J. Appl. Psychol. **85**(2), 273 (2000)
19. Bronk, H., Thorbruegge, M., Hakkaja, M.: A step-by-step approach on how to set up a CSIRT (2006)

On Auxiliary Entity Allocation Problem in Multi-layered Interdependent Critical Infrastructures

Joydeep Banerjee(✉), Arunabha Sen, and Chenyang Zhou

School of Computing, Informatics and Decision System Engineering,
Arizona State University, Tempe, AZ 85287, USA
{joydeep.banerjee,asen,czhou24}@asu.edu

Abstract. Operation of critical infrastructures are highly interdependent on each other. Such dependencies causes failure in these infrastructures to cascade on an initial failure event. Owing to this vulnerability it is imperative to incorporate efficient strategies for their protection. Modifying dependencies by adding additional dependency implications using entities (termed as *auxiliary entities*) is shown to mitigate this issue to a certain extent. With this finding, in this article we introduce the Auxiliary Entity Allocation problem. The objective is to maximize protection in Power and Communication infrastructures using a budget in number of dependency modifications using the auxiliary entities. The problem is proved to be NP-complete in general case. We provide an optimal solution using Integer Linear program and a heuristic for a restricted case. The efficacy of heuristic with respect to the optimal is judged through experimentation using real world data sets with heuristic deviating 6.75% from optimal on average.

Keywords: Interdependent network · IIM model · Auxiliary entity
Dependency modification · $\mathcal{K}$ most vulnerable entities

1 Introduction

Critical infrastructures like power, communication, transportation networks etc. interact symbiotically to carry out their functionalities. As an example there exists strong mutual interactions between the power and communication network or infrastructure (in this article the term *infrastructure* and *network* are used interchangeably). Entities in the power network like generators, substations, transmission lines etc. relies on control signals carried over by communication network entities like routers, fiberoptic lines etc. Similarly all entities in the communication network relies on power supply from the power network to drive their functionalities. To capture this kind of dependencies the critical infrastructure can be modeled as a multilayered interdependent network. Failure of entities in either infrastructure impacts the operation of its own infrastructure as well as the other infrastructure. Owing to these dependencies the initial failure might result

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 25–37, 2017.
https://doi.org/10.1007/978-3-319-71368-7_3

in cascade of failures resulting in disastrous impact. This has been observed in power blackouts which occurred in New York (2003) [1] and India (2012) [11].

To study the nature of failure propagation in these interdependent networks it is imperative to model their dependencies as accurately as possible. Recent literature consists of a plethora of these models [3–8,10,12]. However each of these models have their own shortcoming in capturing the complex dependencies that might exist. For example consider a scenario with one power network entity a_1 and three communication entities b_1, b_2, b_3. The entity a_1 is operational provided that both entities b_1 *and* b_2 are operational *or* if entity b_3 is operational (note that the italicized words represent logical operations). None of the above models can accurately model this kind of a dependency. Sen et al. in [9] proposed a model that uses boolean logic to capture these interdependencies. This model is referred to as the Implicative Interdependency model (IIM)). To express the dependency of an entity on other entities it uses implications which are disjunction(s) and conjunction(s) of logical terms (denoting entities of the network). With respect to the example considered above the dependency implication for the entity a_1 can be represented as $a_1 \leftarrow b_1 b_2 + b_3$. The boolean implication depicting the dependency is termed as *Inter-Dependency Relation*. Our approach in designing solutions and analyzing the problem addressed in this paper is based on the IIM model.

We restrict our attention to an interdependent power and communication network in this paper. However the solutions can be extended to any two interdependent networks. As discussed earlier initial failure of a certain entity set in power and communication network may trigger cascading failure resulting in loss of a large number of entities. Authors in [2] proposed the *Entity Hardening* problem to increase the reliability of these interdependent systems. They assumed that an entity when hardened would be resistant to both initial and cascading failure. Given a set of entities that failed initially (that is at time $t = 0$) the problem was to find a minimal set of entities which when hardened would prevent failure of at least a predefined number of entities. On situations where entity hardening is not possible alternative strategies needs to be employed to increase the system reliability. Adding additional dependencies for entities in interdependent infrastructure can be beneficial in this regard. We elaborate this with the help of an example. Consider the dependency rule $a_1 \leftarrow b_1 b_2 + b_3$. With this dependency entity a_1 would fail if entities (b_1, b_3) or (b_2, b_3) fails. Now consider an entity b_4 is added as a disjunction to the IDR (with the new dependency being $a_1 \leftarrow b_1 b_2 + b_3 + b_4$). For entity a_1 to fail, either (b_1, b_3, b_4) or (b_2, b_3, b_4) should fail. This increases the reliability compared to the previous dependency for a_1. Hence adding additional dependency can be employed as a strategy when entity hardening is not possible. Any entity added to modify a dependency relation is termed as an *auxiliary entity*. However due to system, cost and feasibility constraints the number of such modifications are restricted. Hence when the number of IDR modifications are restricted one has to find which IDRs to modify and with what entities so that the impact of failure is minimized. We term this problem as the *Auxiliary Entity Allocation Problem*. It is to be noted that

inboth *Entity Hardening Problem* and Auxiliary Entity Allocation Problem the IDRs of the interdependent system are changed but these changes are carried out differently.

The rest of the paper is organized as follows. A brief explanation of the IIM model with formal problem definition is provided in Sect. 2. The computational complexity of the problem and proposed solutions are discussed in Sects. 3 and 4 respectively. We discuss the experimental results in Sect. 5 and conclude the paper in Sect. 6

2 Problem Formulation Using the Implicative Interdependency Model

We briefly describe the IIM model introduced in [9]. Two sets A and B represent entities in power and communication network. The dependencies between these set of entities are captured using a set of interdependency denoted as $\mathcal{F}(A, B)$. Each function in the set $\mathcal{F}(A, B)$ is termed as an *Inter-Dependency Relation* (IDR). We describe an interdependent network which composes of the entity sets A and B and the interdependency relations $\mathcal{F}(A, B)$ and denote it by $\mathcal{I}(A, B, \mathcal{F}(A, B))$. Through an example we explain this model further. Consider an interdependent network $\mathcal{I}(A, B, \mathcal{F}(A, B))$ with $A = \{a_1, a_2, a_3, a_4, a_5\}$ and $B = \{b_1, b_2, b_3\}$. The set of IDRs $(\mathcal{F}(A, B))$ for the interdependent network are provided in Table 1. Consider the IDR $a_3 \leftarrow b_2 + b_1 b_3$ in Table 1. It implies that the entity a_1 is operational if entity b_2 *or* entity b_1 *and* b_3 are operational. As evident, the IDRs are essentially disjunction(s) of entity (entities) in conjunctive form. We refer to each conjunctive term, e.g. $b_1 b_3$, as *minterm*. The example considers dependencies where an entity in network A(B) is dependent on entities in network B(A) i.e. inter-network dependency. However this model can capture intra-network dependencies as well.

The cascading procedure is described with respect to the interdependent network captured by IDRs in Table 1. The cascade proceeds in unit time steps (denoted by t). Consider two entities b_2 and b_3 are attacked and made non operational by an adversary at time step $t = 0$ (*initial failure*). Owing to the IDRs $a_2 \leftarrow b_1 b_2$, $a_3 \leftarrow b_2 + b_1 b_3$ and $a_4 \leftarrow b_3$ the entities a_2, a_3, a_4 becomes non operational at $t = 1$. Subsequently entities b_1 $(b_1 \leftarrow a_2)$ and a_1 $(a_1 \leftarrow b_1 + b_2)$ seize to operate at time step $t = 2$ and $t = 3$ respectively. The failure of entities after $t = 0$ is termed as *induced failure*. The cascade is represented in Table 2. It is to be noted that the maximum number of time steps in the cascade for any interdependent network is $|A| + |B| - 1$ (assuming initial time step as $t = 0$). Hence in Table 2 with number of entities being 8 the state (operational or non-operational) of all entities are shown till $t = 7$. Construction of these IDRs is a major challenge of this model. Possible strategies are (i) deep investigation of physical properties and flows in the interdependent network and (ii) consultation with domain experts. The methodology to construct these IDRs is ongoing and is expected to be addressed in future. The problem in this article assumes that the IDRs are already constructed for a given interdependent network.

Table 1. IDRs for the constructed example

Power Network	Comm. Network
$a_1 \leftarrow b_1 + b_2$	$b_1 \leftarrow a_2$
$a_2 \leftarrow b_1 b_2$	$b_2 \leftarrow a_2$
$a_3 \leftarrow b_2 + b_1 b_3$	$b_3 \leftarrow a_4$
$a_4 \leftarrow b_3$	$-$ $-$
a_5	$-$ $-$

Table 2. Cascade propagation when entities $\{b_2, b_3\}$ fail initially. 0 denotes the entity is operational and 1 non-operational

Entities	Time Steps (t)							
	0	1	2	3	4	5	6	7
a_1	0	0	0	1	1	1	1	1
a_2	0	1	1	1	1	1	1	1
a_3	0	1	1	1	1	1	1	1
a_4	0	1	1	1	1	1	1	1
a_5	0	0	0	0	0	0	0	0
b_1	0	0	1	1	1	1	1	1
b_2	1	1	1	1	1	1	1	1
b_3	1	1	1	1	1	1	1	1

Authors in [9] introduced the $\mathcal{K}$ most vulnerable entities problem. The problem used the IIM model to find a set of $\mathcal{K}$ (for a given integer $|\mathcal{K}|$) entities in an interdependent network $\mathcal{I}(A, B, \mathcal{F}(A, B))$ whose failure at time $t = 0$ (*initial failure*) would result in failure of the largest number of entities due to *induced failure*. For the example provided in this section consider the case where $|\mathcal{K}| = 2$. Failing entities b_2 and b_3 at $t = 0$ make entities $\{a_1, a_2, a_3, a_4, b_1, b_2, b_3\}$ not operational by $t = 3$. Hence $\mathcal{K} = \{b_2, b_3\}$ are one of the 2 most vulnerable entities in the interdependent network (it is possible to have multiple $\mathcal{K}$ most vulnerable entities in an interdependent network). The set of entities failed when $\mathcal{K}$ most vulnerable entities fail initially is denoted by $A' \cup B'$ with $A' \subseteq A$ and $B' \subseteq B$. Here $A' = \{a_1, a_2, a_3, a_4\}$ and $B' = \{b_1, b_2, b_3\}$.

For a given $\mathcal{K}$ most vulnerable entities of an interdependent network, the system reliability can be increased (i.e. entities can be protected from failure) by *Entity Hardening* [2]. On scenarios where entity hardening is not possible it is imperative to take alternative strategies. The number of entities failing due to *induced failure* can be reduced by modifying the IDRs. One way of modifying an IDR is adding an entity as a new minterm. For example, consider the interdependent network with IDRs given by Table 1 and b_2 and b_3 being the 2 (when $\mathcal{K} = 2$) most vulnerable entities (as discussed above). Let the IDR $b_1 \leftarrow a_2$ be modified as $b_1 \leftarrow a_2 + a_5$. Hence the new interdependent network is represented as $\mathcal{I}(A, B, \mathcal{F}'(A, B))$ with the same set of IDRs as that in Table 1 except for IDR $b_1 \leftarrow a_2 + a_5$ as the sole modification. The entity a_1 introduced is termed as an *auxiliary entity*. It follows that after the modification, failure of entities b_2 and b_3 at time $t = 0$ would trigger failure of entities a_2, a_3 and a_4 only. Thus before modification the failure set would have been $\{a_1, a_2, a_3, a_4, b_1, b_2, b_3\}$ and after the modification it would be $\{a_2, a_3, a_4, b_2, b_3\}$. Thus the modification would lead to a fewer number of failures.

We make the following assumptions while modifying an IDR —

- It is possible to add an auxiliary entity as conjunction to a minterm. However it is intuitive that this would have no impact in decreasing the number of entities failed due to *induced failure*. Hence we modify an IDR by adding only one auxiliary entity as a disjunction to a minterm.
- An auxiliary entity does not have the capacity to make an entity operational which fails due to *initial failure*. So to prune the search set for obtaining a solution we discard entities in $(A' \cup B')$ as possible auxiliary entities.
- If an IDR D is modified then it is done by adding only one entity not in $A' \cup B' \cup E_D$ where E_D is a set consisting of all entities (both on left and right side of the equation) in D. For any IDR $D \in \mathcal{F}(A, B)$ we denote $AUX = (A \cup B)/(A' \cup B' \cup E_D)$ as the set of auxiliary entities that can be used to modify D.

We quantify the number of modifications done as the number of IDRs to which minterms are added as a disjunction. It should also be noted than an attacker only have information about the initial interdependent network $\mathcal{I}(A, B, \mathcal{F}(A, B))$. Hence with a budget of $|\mathcal{K}|$ it attacks and kills the $\mathcal{K}$ most vulnerable entities to maximize the number of entities killed due to induced failure. Any modification in the IDR is assumed to be hidden from the attacker.

With these definitions the Auxiliary Entity Allocation Problem (AEAP) is defined as follows. Let $\mathcal{K}$ be the most vulnerable entities (already provided as input) of an interdependent network $\mathcal{I}(A, B, \mathcal{F}(A, B))$. With a budget S in number of modifications, the task is to find which are the S IDRs that are to be modified and which entity should be used to perform this modification such that number of entities failing due to *induced failure* is minimized. A more formal description given below.

The Auxiliary Entity Allocation Problem (AEAP)
Instance — An interdependent network $\mathcal{I}(A, B, \mathcal{F}(A, B))$, $\mathcal{K}$ most vulnerable entities for a given integer $|\mathcal{K}|$ and two positive integers S and P_f.
Decision Version — Does there exist S IDR auxiliary entity tuple (D, x_i) such that when each IDRs $D \in \mathcal{F}(A, B)$ is modified by adding auxiliary entity $x_i \in AUX$ as a disjunction it would protect at least P_f entities from *induced failure* with $\mathcal{K}$ vulnerable entities failing initially.

3 Computational Complexity Analysis

In this section we analyze the computational complexity of the AEAP problem. The computational complexity of the problem depends on nature of the IDRs. The problem is first solved by restricting the IDRs to have one minterm of size 1. For this special case a polynomial time algorithm exists for the problem. With IDRs in general form the problem is proved to be NP-complete.

3.1 Special Case: Problem Instance with One Minterm of Size One

The special case consist of IDRs which have a single minterm of size 1 and each entity appearing exactly once on the right hand side of the IDR. With entities a_i's and b_j's belonging to network $A(B)$ and $B(A)$ respectively, the IDRs can be represented as $a_i \leftarrow b_j$. The AEAP problem can be solved in polynomial time for this case. We first define *Auxiliary Entity Protection Set* and use it to provide a polynomial time heuristic in Algorithm 1. The proof of optimality is not included due to space constraint.

Definition 1. ***Auxiliary Entity Protection Set:*** *With a given set of $\mathcal{K}$ most vulnerable entities failing initially the Auxiliary Entity Protection Set is defined as the number of entities protected from* induced failure *when an auxiliary entity x_i is added as a disjunction to an IDR $D \in \mathcal{F}(A, B)$. It is denoted as $AP(D, x_i|\mathcal{K})$.*

Algorithm 1. Algorithm solving AEAP problem for IDRs with minterms of size 1

Data: An interdependent network $\mathcal{I}(A, B, \mathcal{F}(A, B))$ and set of $\mathcal{K}$ vulnerable entities
Result: A set D_{sol} consisting of IDR auxiliary entity doubles (D, x_i) (with $|D_{sol}| = S$ and P_f (denoting the entities protected from induced failure)

begin
  For each IDR $D \in \mathcal{F}(A, B)$ and each entity $x_i \in AUX$ (where $AUX = A \cup B/(A' \cup B' \cup E_D)$ as discussed in the previous section) compute the *Auxiliary Entity Protection Set* $AP(D, x_i|\mathcal{K})$;
  Initialize $D_{sol} = \emptyset$ and $P_f = \emptyset$;
  while $S \neq 0$ **do**
    Choose the Auxiliary Entity Protection Set with highest $AP(x_i, D|\mathcal{K})$. In case of tie break arbitrarily. Let D_{cur} be the corresponding IDR and x_{cur} the auxiliary entity;
    Update $D_{sol} = D_{sol} \cup (D_{cur}, x_{cur})$ and add auxiliary entity x_{cur} as a disjunction to the IDR D_{cur};
    Update $P_f = P_f \cup AP(D_{cur}, x_{cur}|\mathcal{K})$**for** *all IDR $D' \in \mathcal{F}(A, B)$ and $x_i \in D'_A$* **do**
      Update $AP(D', x_i|\mathcal{K}) = AP(D', x_i|\mathcal{K}) \backslash AP(D_{cur}, x_{cur}|\mathcal{K})$;
  return D_{sol} *and* P_f;

3.2 General Case: Problem Instance with an Arbitrary Number of Minterms of Arbitrary Size

The IDRs in general are composed of disjunctions of entities in conjunctive form i.e. arbitrary number of minterms of arbitrary size. This case can be represented as $a_i \leftarrow \sum_{k_1=1}^{p} \prod_{j=1}^{j_{k_1}} b_j$ where entities a_i and $b'_j s$ belong to network $A(B)$

and $B(A)$ respectively. The given example has p minterms each of size j_{k_1}. In Theorem 1 we prove that the decision version of the AEAP problem for general case is NP complete.

Theorem 1. *The decision version of the AEAP problem for Case IV is NP-complete.*

Proof. The hardness is proved by a reduction from Set Cover problem. An instance of a set cover problem consists of a universe $U = \{x_1, x_2, ..., x_n\}$ of elements and set of subsets $\mathcal{S} = \{S_1, S_2, ..., S_m\}$ where each element $S_i \in \mathcal{S}$ is a subset of U. Given an integer X the set cover problem finds whether there are $\leq X$ elements in $\mathcal{S}$ whose union is equal to U. From an instance of the set cover problem we create an instance of the AEAP problem. For each subset S_i we create an entity b_i and add it to set B. For each element x_j in U we add an entity a_j to a set A_1. We have a set A_2 of entities where $|A_2| = |B|$. Let $A_2 = \{a_{21}, a_{22}, ..., a_{2|B|}\}$ where there is an association between entity b_j and a_{2j}. Additionally we have a set of entities A_3 with $|A_3| = X$ which does not have any dependency relation of its own. The set A is comprised of $A_1 \cup A_2 \cup A_3$. The IDRs are created as follows. For an element x_i that appears in subsets S_x, S_y, S_z, an IDR $a_i \leftarrow b_x b_y b_z$ is created. For each entity $b_j \in B$ an IDR $b_j \leftarrow a_{2j}$ is added to $\mathcal{F}(A, B)$. The cardinality of $\mathcal{K}$ most vulnerable node is set to $|A_2|$ and it directly follows that $\mathcal{K} = A_2$ comprises the set of most vulnerable entities. The value of S (number of IDR modifications) is set to X and P_f is set to $S + |A_1|$.

Let there exist a solution to the set cover problem. Then there exist at least X subsets whose union covers the set U. For each subset S_k which is in the solution of the set cover problem we choose the corresponding entity b_k. Let B' be all such entities. We arbitrarily choose and add an entity from A_3 to each IDR $b_k \leftarrow a_{2k}$ with $b_k \in B'$ to form $S = X$ distinct IDR auxiliary entity doubles. As A_3 type entities does not have any dependency relation thus all the entities in B that correspond to the subsets in the solution will be protected from failure. Additionally protecting these B type entities would ensure all entities in A_1 does not fail as well (as there exists at least one B type entity in the IDR of A_1 type entities which is operational). Hence a total of $X + |A_1|$ are protected from failure.

Similarly let there exist a solution to the AEAP problem. It can be checked easily that no entities in $B \cup A_1 \cup A_2$ has the ability to protect additional entities using IDR modification. Hence set A_3 can only be used as auxiliary entities. An entity from A_3 for the created instance can be added to an IDR of A_1 type entity or B type entity. In the former strategy only one entity is protected from failure whereas two entities are operational when we add auxiliary entity to IDRs of B type entities. Hence all the auxiliary entities are added to the B type IDRs with a final protection of $X + |A_1|$ entities. For each IDR of the B type entity to which the auxiliary entity is added, the corresponding subset in $\mathcal{S}$ is chosen. The union of these subsets would result in U as the solution of the AEAP problem protects the failure of all A_1 type entities. Hence solving the set cover problem and proving the hardness stated in Theorem 1.

4 Solutions to the AEAP Problem

We consider the following restricted case where there exists at least S entities in the interdependent network which does not belong to any of the failing entities. This comprise the set of auxiliary entities that can be used. It is also imperative to use such set as auxiliary entities because they never fail from induced or initial failure when the $\mathcal{K}$ most vulnerable entities fail initially. The problem still remains to be NP compete for this case as in Theorem 1 the set of entities A_3 belong to such class of auxiliary entities. With these definition of the special case let $\mathcal{A}$ denote a set of such auxiliary entities which can be used for IDR modifications with $\mathcal{A} \subset A \cup B/(A' \cup B')$. Hence we loose the notion of IDR auxiliary entity doubles in the solution as any auxiliary entity from set $\mathcal{A}$ would produce the same protection effect. So in both the solutions we only consider the IDRs that needs to be modified and disregard which auxilairy entity is used for this modification. We first propose an Integer Linear Program (ILP) to obtain the optimal solution in this setting. We later provide a polynomial heuristic solution to the problem. The performance of heuristic with respect to the ILP is compared in the section to follow.

4.1 Optimal Solution to AEAP Problem

We first define the variables used in formulating the ILP. Two set of variables $G = \{g_1, g_2, ..., g_c\}$ and $H = \{h_1, h_2, ..., h_d\}$ (with $c = |A|$ and $d = |B|$) are used to maintain the solution of $\mathcal{K}$ most vulnerable entities. Any variable $g_i \in G$ ($h_j \in H$) is equal to 1 if $a_i \in A$ ($b_j \in B$) belongs to $\mathcal{K}$ and is 0 otherwise. For each entity a_i and b_j a set of variables x_{id} and y_{jd} are introduced with $0 \leq d \leq |A| + |B| - 1$. x_{id} (y_{id}) is set to 1 if the entity a_i (b_j) is non operational at time step d and is 0 otherwise. Let P denote the total number of IDRs in the interdependent network and assume each IDR has a unique label between numbers from 1 to P. A set of variables $M = \{m_1, m_2, ..., m_P\}$ are introduced. The value of m_i is set to 1 if an auxiliary node is added as a disjunction to the IDR labeled i and 0 otherwise. With these definitions we define the objective function and the set of constraints in the ILP.

$$min\Big(\sum_{i=1}^{|A|} x_{i(|A|+|B|-1)} + \sum_{j=1}^{|B|} y_{j(|A|+|B|-1)}\Big) \tag{1}$$

The objective function defined in 1 tries to minimize the number of entities having value 1 at the end of the cascade i.e. time step $|A| + |B| - 1$. Explicitly this objective minimizes the number of entities failed due to induced failure. The constraints that are imposed on these objective to capture the definition of AEAP are listed below —

Constraint Set 1: $x_{i0} \geq g_i$ and $y_{j0} \geq h_j$. This imposes the criteria that if entity a_i (b_j) belongs to the $\mathcal{K}$ most vulnerable entity set then the corresponding variable x_{i0} (y_{j0}) is set to 1 capturing the *initial failure*.

Constraint Set 2: $x_{id} \geq x_{i(d-1)}, \forall d, 1 \leq d \leq |A| + |B| - 1$, and $y_{id} \geq y_{i(d-1)}, \forall d, 1 \leq d \leq |A| + |B| - 1$. This ensures that the variable corresponding to an entity which fails at time step t would have value 1 for all $d \geq t$.

Constraint Set 3: We use the theory developed in [9] to generate constraints to represent the cascade through the set of IDRs. To describe this consider an IDR $a_i \leftarrow b_j b_p b_l + b_m b_n + b_q$ in the interdependent network. Assuming the IDR is labeled v it is reformulated as $a_i \leftarrow b_j b_p b_l + b_m b_n + b_q + m_v$ with $m_v \in M$. This is done for all IDRs. The constraint formulation is described in the following steps.

Step 1: All minterms of size greater than 1 are replaced with a single virtual entity. In this example we introduce two virtual entities C_1 and C_2 ($C_1, C_2 \notin A \cup B$) capturing the IDRs $C_1 \leftarrow b_j b_p b_l$ and $C_2 \leftarrow b_m b_n$. The IDR in the example can be then transformed as $a_i \leftarrow C_1 + C_2 + b_q + m_v$. For any such virtual entity C_k a set of variables c_{kd} are added with $c_{kd} = 1$ if C_k is alive at time step d and 0 otherwise. Hence all the IDRs are represented as disjunction(s) of single entities. Similarly all virtual entities have IDRs which are conjunction of single entities.

Step 2: For a given virtual entity C_k and all entities having a single midterm of arbitrary size, we add constraints to capture the cascade propagation. Let N denote the number of entities in the IDR of C_k. The constraints added is described through the example stated above. The variable c_1 with IDR $C_1 \leftarrow b_j b_p b_l$, constraints $c_{1d} \geq \frac{y_{j(d-1)} + y_{p(d-1)} + y_{l(d-1)}}{N}$ and $c_{1d} \leq y_{j(d-1)} + y_{p(d-1)} + y_{l(d-1)} \forall d, 1 \leq d \leq m + n - 1$ are added (with $N = 3$ in this case). This ensures that if any entity in the conjunction fails the corresponding virtual entity fails as well.

Step 3: In the transformed IDRs described in step 1 let n denote the number of entities in disjunction for any given IDR (without modification). In the given example with IDR $a_i \leftarrow C_1 + C_2 + b_q + m_v$, constraints of form $x_{id} \geq c_{1(d-1)} + c_{2(d-1)} + y_{q(d-1)} + m_v - (n-1)$ and $x_{id} \leq \frac{c_{1(d-1)} + c_{2(d-1)} + y_{q(d-1)} + m_v}{n} \forall d, 1 \leq d \leq m + n - 1$ are added. This ensures that the entity a_i will fail only if all the entities in disjunction become non operational.

Constraint Set 4: To ensure that only S auxiliary entities are added as disjunction to the IDRs constraint $\sum_{v=1}^{P} m_v = S$ is introduced.

4.2 Heuristic Solution to the AEAP Problem

In this section we provide a polynomial heuristic solution to the AEAP problem. We first redenote *Auxiliary Entity Protection Set* as $AP(D|\mathcal{K})$ as it is immaterial which entity is added as an auxiliary entity since no auxiliary entity can fail due to any kind of failure. Along with the definition of *Auxiliary Entity Protection Set*, we define *Auxiliary Cumulative Fractional Minterm Hit Value* (ACFMHV) for designing the the heuristic. We first define *Auxiliary Fractional Minterm*

Hit Value (AFMHV) in Definition 2 which is used in defining ACFMHV (in Definition 3).

Definition 2. *The Auxiliary Fractional Minterm Hit Value of an IDR* $D \in \mathcal{F}(A, B)$ *is denoted by* $AFMHV(D|\mathcal{K})$. *It is calculated as* $AFMHV(D|\mathcal{K}) = \sum_{i=1}^{m} \frac{1}{|s_i|}$. *Let* x_j *denote the entity in the right hand side of the IDR* D *and* m *denotes all the minterms in which the entity* x_j *appears over all IDRs. The parameter* s_i *denotes* i^{th} *such minterm with* $|s_i|$ *being its size. If an auxiliary entity is placed at* D *then the value computed above provides an estimate implicit impact on protection of other non operational entities.*

Definition 3. *The Auxiliary Cumulative Fractional Minterm Hit Value of an IDR* $D \in \mathcal{F}(A, B)$ *is denoted by* $ACFMHV(D)$. *It is computed as* $ACFMHV(D) = \sum_{\forall x_i \in AP(D|\mathcal{K})} AFMHV(D_{x_i}|\mathcal{K})$ *where* D_{x_i} *is the IDR for entity* $x_i \in AP(D|\mathcal{K})$. *The impact produced by the protected entities when IDR* D *is allocated with an auxiliary entity over set* $A \cup B$ *is implicitly provided by this definition.*

The heuristic is provided in Algorithm 2. At any given iteration the auxiliary entity is placed at the IDR which protects the most number of entities from failure. In case of a tie the entity having highest ACFMHV value is chosen. At any given iteration the algorithm greedily maximize the number of entities protected from *induced failure*. Algorithm 2 runs in polynomial time, more specifically the time complexity is $\mathcal{O}(Sn(n+m)^2)$ (where $n = |A| + |B|$ and $m =$ Number of minterms in $\mathcal{F}(A, B)$).

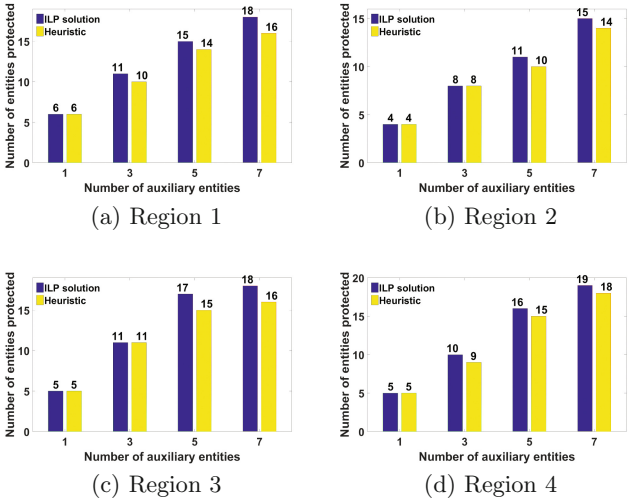


Fig. 1. Comparison of the number of entities protected in optimal solution (ILP) and heuristic in each of the 5 identified regions with $|\mathcal{K}| = 8$ and number of auxiliary entities (or modifications) varied as $1, 3, 5, 7$

Algorithm 2. Heuristic solution to the AEAP problem

Data: An interdependent network $\mathcal{I}(A, B, \mathcal{F}(A, B))$, set of $\mathcal{K}$ vulnerable entities and set $\mathcal{A}$ of auxiliary entities
Result: A set D_{sol} consisting of IDRs (with $|D_{sol}| = S$ to each of which an auxiliary entity is added as a disjunction and P_f (denoting the entities protected from induced failure)
begin
  Initialize $D_{sol} = \emptyset$ and $P_f = \emptyset$;
  while $\mathcal{A} \neq \emptyset$ **do**
    For each IDR $D \in \mathcal{F}(A, B)$ compute the *Auxiliary Node Protection Set* $AP(D|\mathcal{K})$;
    if *There exists multiple IDRs having same value of highest cardinality of the set* $AP(D|\mathcal{K})$ **then**
      For each IDR $D \in \mathcal{F}(A, B)$ compute the *Auxiliary Cumulative Fractional Minterm Hit Value* $ACFMHV(D)$;
      Let D_p be an IDR having highest $ACFMHV(D_p)$ among all D_i's in the set of IDRs having highest cardinality of the set $AP(D_i|\mathcal{K})$;
      If there is a tie choose arbitrarily;
      Update $D_{sol} = D_{sol} \cup D_p$ and add an auxiliary entity from $\mathcal{A}$ as a disjunction to the IDR D_p;
      Update $P_f = P_f \cup AP(D_p)$;
      Update $\mathcal{A}$ by removing the auxiliary entity added;
    else
      Let D_p be an IDR having highest cardinality of the set $D \in \mathcal{F}(A, B)$;
      Update $D_{sol} = D_{sol} \cup D_p$ and add an auxiliary entity from $\mathcal{A}$ as a disjunction to the IDR D_p;
      Update $P_f = P_f \cup AP(D_p|\mathcal{K})$;
      Update $\mathcal{A}$ by removing the auxilary entity added;
    Prune the interdependent network $\mathcal{I}(A, B, \mathcal{F}(A, B)$ by removing the IDRs for entities in $AP(D_p|\mathcal{K})$ and removing the same set of entities from $A \cup B$;
  return D_{sol} *and* P_f;

5 Experimental Results

The solution of the heuristic is compared with the ILP to judge its efficacy. We perform the experiments on real world data sets with the IDRs generated artificially based on some predefined rules. Platts (www.platss.com) and GeoTel (www.geo-tel.com) provided the power and communication network data respectively. The power network data consisted of two types of entity — 70 power plants and 470 transmission lines. There are three types of entity in the communication network data — 2,690 cell towers, 7,100 fiber-lit buildings and 42,723 fiber links. The data corresponds to the Maricopa county region of Arizona, USA. To perform the experimental analysis we picked four non overlapping regions in Maricopa county. They are labelled as Region 1, 2, 3, and 4 respectively.

It is to be noted that the union of these regions does not cover the entry county. For each region we filtered out the entities from our dataset and constructed the IDRs based on rules defined in [9].

The cardinality of $\mathcal{K}$ most vulnerable nodes was set to 8 and was calculated using the ILP described in [9]. The number nodes failed in each region due to initial failure of the most vulnerable nodes are $28, 23, 28, 28$ respectively. We vary the number of auxiliary entities placed (or modifications) from 1 to 7 in steps of 2. For each region and modification budget the number of entities protected from failure for the heuristic was compared with the ILP solution and is plotted in Fig. 1. The maximum possible percentage difference of the heuristic from optimal for any region and modification budget pair is observed to be a 11.76% in Region 3 with 5 auxiliary entities (Fig. 1c). On an average the heuristic performed very near to the optimal with a difference of 6.75%.

6 Conclusion

In this paper we introduce the auxiliary entity allocation problem in multilayer interdependent network using the IIM model. Entities in multilayer network can be protected from an initial failure event when auxiliary entities are used to modify the IDRs. With a budget on the number of modifications, the problem is proved to be NP-complete. We provide an optimal solution using ILP and polynomial heuristic for a restricted case of the problem. The optimal solution was compared with the heuristic on real world data sets and on an average deviates 6.75% from the optimal.

References

1. Andersson, G., Donalek, P., Farmer, R., Hatziargyriou, N., Kamwa, I., Kundur, P., Martins, N., Paserba, J., Pourbeik, P., Sanchez-Gasca, J., et al.: Causes of the 2003 major grid blackouts in north america and europe, and recommended means to improve system dynamic performance. IEEE Trans. Power Syst. **20**(4), 1922–1928 (2005)
2. Banerjee, J., Das, A., Zhou, C., Mazumder, A., Sen, A.: On the entity hardening problem in multi-layered interdependent networks. In: 2015 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 648–653. IEEE (2015)
3. Bernstein, A., Bienstock, D., Hay, D., Uzunoglu, M., Zussman, G.: Power grid vulnerability to geographically correlated failures-analysis and control implications, (2012). arXiv preprint arXiv:1206.1099
4. Buldyrev, S.V., Parshani, R., Paul, G., Stanley, H.E., Havlin, S.: Catastrophic cascade of failures in interdependent networks. Nature **464**(7291), 1025–1028 (2010)
5. Gao, J., Buldyrev, S.V., Stanley, H.E., Havlin, S.: Networks formed from interdependent networks. Nat. Phys. **8**(1), 40–48 (2011)
6. Nguyen, D.T., Shen, Y., Thai, M.T.: Detecting critical nodes in interdependent power networks for vulnerability assessment (2013)

7. Parandehgheibi, M., Modiano, E.: Robustness of interdependent networks: The case of communication networks and the power grid, (2013). arXiv preprint arXiv:1304.0356
8. Rosato, V., Issacharoff, L., Tiriticco, F., Meloni, S., Porcellinis, S., Setola, R.: Modelling interdependent infrastructures using interacting dynamical models. Int. J. Crit. Infrastruct. **4**(1), 63–79 (2008)
9. Sen, A., Mazumder, A., Banerjee, J., Das, A., Compton, R.: Identification of k most vulnerable nodes in multi-layered network using a new model of interdependency. In: 2014 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 831–836. IEEE (2014)
10. Shao, J., Buldyrev, S.V., Havlin, S., Stanley, H.E.: Cascade of failures in coupled network systems with multiple support-dependence relations. Phys. Rev. E **83**(3), 036116 (2011)
11. Tang, Y., Bu, G., Yi, J.: Analysis and lessons of the blackout in indian power grid on 30–31 july 2012. In: Zhongguo Dianji Gongcheng Xuebao(Proceedings of the Chinese Society of Electrical Engineering), vol. 32, pp. 167–174. Chinese Society for Electrical Engineering (2012)
12. Zhang, P., Peeta, S., Friesz, T.: Dynamic game theoretic model of multi-layer infrastructure networks. Netw. Spat. Econ. **5**(2), 147–178 (2005)

Cyber Targets Water Management

Pieter Burghouwt[1(✉)], Marinus Maris[1], Sjaak van Peski[1], Eric Luiijf[2], Imelda van de Voorde[2], and Marcel Spruit[1]

[1] The Hague University of Applied Sciences, The Hague, The Netherlands
{p.burghouwt,m.g.maris,j.vanpeski,m.e.m.spruit}@hhs.nl
[2] Netherlands Organisation for Applied Scientific Research TNO, The Hague, The Netherlands
{eric.luiijf,imelda.vandevoorde}@tno.nl

Abstract. Water management is a critical infrastructure activity in The Netherlands. Many organizations, ranging from local municipalities to national departments are involved in water management by controlling the water level to protect the land from flooding and to allow inland shipping. Another important water management task is the purification of waste water and sewage. To fulfill these tasks, such organizations depend on information and communication technologies, ranging from standard office IT facilities to Industrial Control Systems (ICS), for example to control excess water pumps and locks, as well as to monitor and control water purification plants. The worldwide increase of both volume and sophistication of cyber attacks made the Dutch government decide to sponsor a project to determine a cyber security posture of the water management organizations by benchmarking the cyber security state of their water management installations and processes. In this paper we present our benchmark approach to the security of ICS. Moreover, we discuss the major results of the benchmark as well as a cyber security simulator that was developed to raise awareness and develop further knowledge on the ICS-specific issues.

Keywords: Critical infrastructure protection · Water management
Cyber security · Industrial Control System · SCADA · Cyber resilience
Benchmark · Simulator

1 Introduction

For water management tasks in The Netherlands, numerous organizations are involved, ranging from local municipalities and regional organizations to national departments. Together they monitor and control water levels, ensure purification of waste water and sewage and regulate irrigation. The history of water management in the Netherlands dates back quite some time. Already in the year 1122, twenty communities collaboratively worked on building a dam to protect their cattle, land and properties against flooding. Much later, around the year 1900, the distribution of drinking water by means of water networks and steam pumps started. Nowadays, modern electric pumps have replaced the wellknown Dutch windmills and steam pumps.

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 38–49, 2017.
https://doi.org/10.1007/978-3-319-71368-7_4

1.1 Use of Industrial Control Systems

Critical water management services are remotely monitored and controlled by Industrial Control Systems (ICS), including Supervisory Control and Data Acquisition (SCADA) systems[1].

Most water management processes rely on a variety of ICS equipment such as Programmable Logic Controllers (PLC), pumps, sensors and valves from a multitude of vendors. The water management infrastructure is carefully designed and operated with a safety-first mindset. Typically, the control systems are designed in a redundant way, to ensure that operation continues in case of a breakdown of a part of the systems. In addition, the controls of the water management systems can be bypassed with manual operation, further increasing redundancy. Meanwhile, the ICS equipment is augmented with networking capabilities using the Internet Protocol [10], connecting the ICS domain to the office IT and public networks such as the Internet.

1.2 Cyber Threats and Risk

In the recent years, an increasing number of security-related incidents have occurred in ICS [10], also in critical infrastructure sectors [5,11]. But, not only the likelihood of cyber attacks on ICS in critical infrastructure increases; also the consequences become more serious, since critical services become increasingly dependent on ICS. The possibilities of manual intervention during or after a cyber attack are limited by: the dependency chains of key services, which in turn depend on the proper functioning of ICS, and the decreasing number of operators being familiar with the (old fashioned) manual control. The resulting increased cyber risk urges most organizations, which deploy ICS, to thoroughly examine their internal procedures and protection mechanisms in order to thwart cyber attacks by adversaries, such as vengeful employees, terrorists, criminals, and even rogue states. For these reasons, a project was granted by the Dutch government to The Hague University of Applied Sciences and TNO in order to conduct research on the cyber security posture of water management organizations.

Three objectives were identified to support future direction and decisions regarding cyber security in relation to water management:

1. A benchmark of the cyber security resilience of the ICS environment against all kinds of hazards, including cyber attacks.
2. A demonstration and simulation environment to gain both awareness and further knowledge about ICS-specific security issues, derived from the results of the benchmark.
3. A benchmark aimed at determining the cyber security maturity level of the organizations (not discussed in this paper).

[1] We will use the term ICS hereafter as a generic term for ICS and SCADA.

1.3 Structure of This Paper

Section 2 relates our work with existing publications. Section 3 presents the ICS cyber security benchmark and the main results. Section 4 elaborates on specific cyber security-related dilemmas. Section 5 presents the design and use of the physical simulation environment and tooling we developed. Section 6 concludes the research and proposes future work.

2 Related Work

Security Risk Assesment is well described in various standards and guidelines, such as the IEC/ISO27005 [4]. In general the assessment compares the actual situation with well-defined risk criteria. Our benchmark approach enables organizations with similar activities to compare themselves with each other in a relatively simple way by means of a questionnaire.

In 2007, Luiijf et al. developed an ICS security benchmark questionnaire for the Dutch drinking water sector. The main benchmark results were discussed in [7]. In our work we have assessed the cyber security of Dutch water management organizations. For our study we have updated and improved the questionnaire of 2007, reflecting technology changes, such as the move from ISDN to IP/VPN and 4G data links, and increased insights in the ICS set of organizational, technical and human-related threats [10].

Amin et al. propose a framework for benchmarking security risks in cyber physical systems [2]. The game-theoretic approach results in an assessment model. They propose Deterlab [12] as an environment for further assessing the related cyber risks. Our study benchmarks multiple organizations with similar activities by the use of a questionnaire. Our work also presents DESI, a self-designed and developed simulator for demonstration of and experiments with cyber attacks in ICS environments to raise awareness and increase knowledge. DESI can be seen as a testbed with demonstration facilities. In contrast to Deterlab, DESI is equipped with real hardware components, such as PLCs and HMI-panels in addition to the virtual implementation of standard computer components. The support of real ICS equipment and dedicated network components, in addition to virtualized components, results in a more realistic and better recognizable simulation environment that allows for a wide range of cyber attacks which exploit ICS-specific vulnerabilities.

Another security-related testbed that focusses on ICS is SCADAVT-A [1]. Unlike our simulator, SCADAVT-A is aimed at a Modbus-emulation and a custom TCP protocol for the connection with a simulator of dedicated I/O modules. This restricts the degree of reality as the possibilities of extending the simulator with real hardware are limited.

3 A Benchmark of the Resilience of the ICS Environment

To assess the security posture of ICS environments in water management systems, we have reused and further developed the benchmarking methodology,

based on elaborated questionnaires, as described by Luiijf et al. [7]. The original methodology has been used three times by the Dutch drinking water sector and twice by the Dutch energy sector. For this study, we updated and improved the methodology. The main reasons for assessing the cyber security posture of the ICS environments include the strong dependence of the water management systems on the ICS environments, the severe consequences in case the cyber security of these environments would fail, as well as the earlier results from the two critical sectors which showed a need for significant security improvements.

The questionnaire of the original methodology by Luiijf et al. contained 39 main questions which can be found in the annex to [7]. Several questions have been dropped or combined with another question based upon the experiences with the drinking water and energy sector benchmarks. In the meantime, the questionnaire has grown to 48 (closed and open) questions (in Dutch). Some questions are used to validate the reliability of earlier given answers. The additional questions help to clarify the security posture of the ICS environment, from the perspectives of governance, organization, system management, networking, new system acquisition, and (third party) maintenance.

The additional main questions regarding organizational aspects are:

1. Does the organization have a security officer with integral responsibilities? A CISO? A security officer of ICS? An internal audit department verifying the cyber security of ICS? An external audit service performing the same task?
2. Has the cyber security of your ICS been outsourced?
3. Does your organization perform a (cyber) security audit of a third party before in-sourcing services?

The additional main question regarding telecommunication aspects is:

1. Are you using IP version 6? If not, do you have a IPv6 migration plan?

The additional main questions regarding system management aspects are:

1. Do you screen employees? If yes, what is the frequency?
2. Do you screen third party employees or do you have contractual arrangements?
3. Which type of outside access to your ICS is allowed? By whom? What access rights are granted?
4. Has the system management of ICS been outsourced?
5. Does your acquisition process include cyber security requirements when contracting third party ICS services? (now an explicit question; formerly part of a more general question)
6. Do you make use of pen testing or white hat hackers to verify the security posture of your ICS environment?
7. Has the organization a recurring process to monitor security incidents in the ICS environment?
8. Are ICS security incidents reported as part of the management reporting process?

9. What type of physical security measures have been taken to protect the integrity and availability of your ICS?
10. Which ICS security topics need to be addressed sector-wide?

No changes have been made to the weights and scoring of the answers since 2007. In this way, the benchmark results can be compared with older results if the organizations involved are willing to share the results cross-sector in a trusted setting.

As described in [8], the received questionnaires are linked to a random organization number. The benchmark results have been reported anonymously under the Traffic Light Protocol (TLP) [3], whereas each individual benchmarked organization received its own relative performance to the benchmark average in the form of three radar diagrams - one for organizational issues, one for system management issues and one for networking, each accompanied by a concise explanation. A water management sector-specific baseline, derived from the ISO27001/2 standards has been used for metrics for the specific findings in the benchmark report.

The main observations regarding ICS cyber security are described below. 19 water management organizations participated in the benchmark. We will refer to them as the assessed organizations. We have omitted information that could hamper national and/or company security.

1. Some of the assessed organizations are more advanced in protecting their ICS environment than the benchmark average. Even for them, we identified measures to reduce the risk to the ICS environment considerably. Those comprise measures to increase management awareness with respect to the influence of ICS on their critical services, and with that funding for detailed risk analyses and improvements.
2. ICS security was in some cases approached from a holistic point of view. In the other assessed organizations technical measures, organizational measures and physical security fell under distinct responsibilities. Within the latter organizations coinciding measures, which strengthen each other, existed just by coincidence.
3. 35% of the organizations outsourced the complete installation and maintenance of their ICS environment to a third party. Because water management is a critical activity, screening of third party personnel is required. However, in practice only 20% of the organizations had such measures in place.
4. Not all of the assessed organizations had taken measures according to the Dutch cybercrime law which exempt them in court from revealing detailed cyber security measures in case of the prosecution of a hacker of the water management systems.
5. 40% of the assessed organizations did not state decisive cyber security requirements when acquiring new ICS or ICS-related services. Only one organization assessed the cyber security due diligence of their ICS hardware, software and service suppliers.
6. 50% of the assessed organizations discussed with their ICS suppliers a fast delivery of new equipment in case of an emergency, e.g. a fire.

7. Cyber security incidents in the ICS domain have been reported by a number of the assessed organizations over the last years. Other incidents could have happened unnoticed as some organizations did neither have intrusion detection and firewall monitoring measures in place, nor have incident reporting procedures.

Networking aspects:

1. Despite existing good practices [7–9], 15% of the assessed organizations did not separate their office IT network from their ICS network. 10% of the assessed organizations separated these networks only for new installations and at the larger locations. On the other hand, 30% of the organizations used a physical network separation.
2. Firewall logging and audits of firewalls were in some cases no part of daily operations, which means that intrusion attempts could go unnoticed for a long period of time.
3. Configuration management and change management of ICS and the ICS networks were often not seen as an operational measure.
4. At the time of the benchmark there were no plans at all for an IPv4 to IPv6 migration of systems and networks.
5. 75% of the assessed organizations allowed third party engineers and their ICS suppliers to remotely access their ICS domain. The majority of the assessed organizations used additional measures, such as strong authentication on this type of access.
6. However, some of the assessed organizations allowed third parties to plug in laptops and other components in the ICS network without any restriction. The only barrier was physical access control.

System management aspects:

1. Despite existing good practices [7–9], 30% of the assessed organizations use in some situations the default manufacturer passwords. Legacy aspects of ICS is one of the reasons mentioned.
2. The critical nature of water management requires individual passwords and disallows 'group passwords'. Passwords need to be changed within a given period. The reality is that 35% of the assessed organizations use a mixture of group and individual passwords in their ICS domain, and that even some of the assessed organizations use passwords 'lifelong'.
3. The far majority of the assessed organizations uses antivirus software, although not all assessed organizations regularly update their detection profiles. This results in a long vulnerability period average.
4. Security patching is far from being performed according to the base-line requirement, as is outlined in the next section.

To conclude, the methodology presented above has given a broad insight into the weak and strong areas of cyber security of the ICS environment of Dutch water management organizations. The benchmark produced very diverse results between the individual organizations, concerning the security level of ICS.

The main causes of this diversity are: the outsourcing of ICS maintenance, the divided or unclear security-related responsibilities of the ICS part of installations, and the lack of organization-wide awareness of specific ICS cyber risks. Some weaknesses can be resolved by individual organizations; on others, collaboration may be more efficient and effective. Despite the high diversity, certain technology-related security issues were seen in the majority of the participating organizations. We will elaborate on this in Sect. 4.

4 Observed ICS Security Dilemmas

The results of the ICS benchmark revealed several technology-related dilemmas, highly related with ICS and the distributed nature of the water management installations. Through further interviews with the technical staff and security officers, and field studies, including on-site visits, we elaborated these dilemmas. We found that the three most important dilemmas are:

1. *Patching vs. Continuity:* The difficulty of patching in the ICS environment is a notorious problem. The time interval between the discovery of a new vulnerability of an ICS-device and the actual deployment of a patch to remove the vulnerability or to mitigate its effect is often extremely long. Good practice states a maximum delay in the order of days for critical patches and a delay until the next regular maintenance activities for non-critical patches. In practice an average patch delay can be significantly longer. Interviews with the technical staff revealed that the main cause of this delay is not lack of awareness but the fear of process interruption. Patching an ICS device, such as a PLC, introduces the risk that the system will malfunction. Rollback, replacement, or repair is in such cases often difficult and time-consuming. As there is not always a realistic test facility available that can, thoroughly and in advance, test the effects of the patch, this dilemma is not solved easily.
2. *Isolated vs. Centralized Control:* Traditionally the water management installations were isolated entities with dedicated hardware and limited remote control facilities. This isolation created a de facto security layer. With the introduction of Internet technology and commercial off-the-shelf (COTS) solutions in ICS, this type of isolation has disappeared. The result may be an insecure network architecture with uncontrolled types of communication between the office network, the control room and the ICS-environment. A well-known and proven solution for this problem is the use of network compartments as well as traffic monitoring and restrictions per segment. However, due to the high variety of allowed traffic, detecting and filtering of undesired traffic can be difficult, especially in the case of a sophisticated cyber attack where malicious traffic is carefully crafted to mimic allowed traffic. Moreover, firewalls that filter too restrictively or Network Intrusion Detection Systems (NIDS) that detect too many false positives, obstruct the desired central control. In such cases, the operational staff can be tempted to configure or even completely bypass the security controls in order to obtain the desired functionality. Hence, the critical parts of the network should be logically isolated as much as possible, but this does not solve the dilemma.

3. *Automation vs. Disaster Recovery Capacity:* An important objective in automation is cost reduction because it allows a relatively small staff to operate a relatively complex and critical process. However, in the case of service disruption, people are needed to respond and maintain continuity. As long as the malfunctions are local, a mobile team of limited size can solve the problems, especially because in most situations there is enough time to respond before water levels become critical or failing purification impacts society. However, in case of a large scale sophisticated cyber attack that affects multiple key processes at the same time, process control continuity would require a large number of skilled personnel instantly available on the numerous sites of the water management organizations. Anticipating these types of cyber attacks limits the aforementioned cost reduction, as more people would continuously be on guard. Hence this dilemma asks for awareness and careful consideration of the cyber attack scenarios and new cyber threats.

5 Cyber Security Simulator for Water Management Control Systems

In order to demonstrate and examine the cyber-physical security issues, mentioned in Sects. 3 and 4, we developed DESI, a simulator for demonstration and experimentation purposes. In this simulation environment, attack scenarios with given vulnerabilities and controls can be set up, to demonstrate and evaluate attack consequences and the effectiveness of cyber security controls. The two main objectives of DESI are:

- Create awareness by demonstration of realistic cyber attack scenarios, related with the aforementioned security dilemmas. The intended audience is not limited to the technical staff of the organizations, but includes also its higher management, decision makers, and even students as the future designers and operators of these installations.
- Increase knowledge about ICS cyber attacks and defenses to contribute to optimally secure design and operation of the water management processes.

In addition to the objectives, DESI had to meet three important requirements:

1. Flexible and modular design to support the wide variety of equipment and configurations in use at the water management organizations and a wide variety of cyber attacks to ICS.
2. Realistic configuration, especially for the ICS part, and realistic attacks.
3. Provide a clear insight in a cyber attack and its consequences, by making the demonstration easily accessible for a large audience.

To support the first requirement, the simulator has a modular design as shown in Fig. 1. The virtualization of generic IT equipment, such as office computers, switches and routers, facilitates easy deployment of standard IT-components and network topologies. Generic end systems, such as desktop computers and

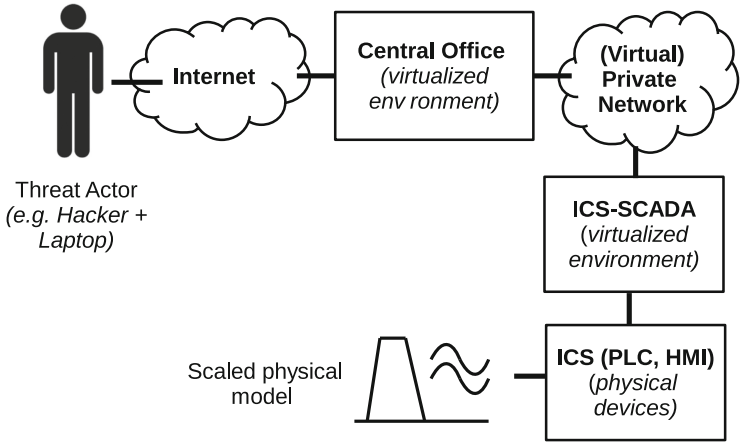


Fig. 1. Schematic overview of the DESI simulation environment.

servers are virtualized in KVM [6]. KVM is a completely open source virtualization platform that allows for an open and scalable implementation. The virtual environment of DESI is divided over two KVM hosts, each with a Linux workstation as hardware.

- *Central Office* hosts the end systems and generic network equipment located in the Central office, such as desktop computers, web, mail, and application servers.
- *ICS-SCADA* hosts the end systems and generic network equipment located in the vicinity of the physical process, such as on-site SCADA-systems and engineering workstations.

The generic network equipment is virtualized in Open vSwitch [14] that is also implemented on both KVM hosts. This allows for a flexible design of networks and interconnections.

To meet the second requirement, the simulator supports the inclusion of physical ICS devices, such as the Siemens Simatic S7-1200 PLC and the Siemens Simatic Basic human-machine interface (HMI) panel [16]. Each of both KVM hosts is connected to a physical L2-switch (Cisco Catalyst 2960) by a IEEE802.1q VLAN trunk. In both hosts, VLAN's and trunking are managed by Open vSwitch. In this way a physical device, such as a PLC, can be connected to the ICS-SCADA block that runs one or more virtualized end systems, such as a SCADA computer. In the same way, specialized network hardware, such as a dedicated firewall or a VPN-solution, can be connected with the switches of both KVM-hosts to create a realistic physical interconnection, outside the virtualized environment.

A scaled physical model of a typical water management process with typical equipment was developed to contribute further to the insightfulness of the demonstrations for the intended audiences, as stated in the third requirement. The physical model represents a Dutch polder with canals on different water levels, real water, pumps and level meters. Failure of the ICS system can result in flooding of the polder.

The modular design provides also a visible spatial separation between the on-site IT-equipment and the Central Office by a physical communication network that connects both locations, and the external threat actor. The external threat actor is typically a connected external computer (laptop) or optionally a virtual instance in one of the KVM hosts. In addition the physical L2-switches facilitate easy network traffic observations and the insertion of internal attack instances.

5.1 Deployment of Attack Scenarios

For demonstration purposes, university students of the department of Computer Science of our university, in collaboration with cyber security experts, designed and implemented realistic cyber attack scenarios. Each attack scenario is feasible for a medium skilled hacker with a laptop, running Kali Linux with common attack tools, such as Metasploit with Armitage [13]. This approach resulted in several highly realistic cyber security scenarios with a high likelihood of occurrance.

All cyber attacks are multi-staged, for example:

1. An infection, such as one caused by an employee who opens in the office an infected pdf file which was received as an attachment to an email.
2. The malware from the pdf file installs itself on the computer and creates a backdoor.
3. The hacker uses the backdoor and moves laterally through the network towards the ICS-part.
4. The hacker creates specially-crafted packets, e.g. Siemens S7-packets, to influence the processing of an unpatched PLC.
5. Pumps are disabled, without notification on HMI or SCADA. Sensor readings of water levels remain unchanged at the central consoles.
6. Flooding of a region, e.g. due to rain, while excess water is not pumped away.

In the demonstrations we make clear that once the PLC is compromised, on-site action by specialized personal is required to regain control. As an organization can operate thousands of PLC's, the recovery of a cyber attack may take long and may require an outrageous amount of expert man power.

One might say that in all stages, the hacker exploits obvious vulnerabilities. However, the countermeasures against such a cyber attack are manifold and not always trivial as pointed out in the aforementioned dilemmas.

5.2 DESI Results

DESI meets both objectives, as formulated at the start of Sect. 5:

- *Creation of cyber attack awareness by demonstration:* On several occasions the cyber-physical attack scenarios have been successfully demonstrated by DESI to technical staff and decision makers inside and outside water management organizations. This resulted in positive feedback and vivid discussions about the demonstrated cyber-physical attacks, the risks, and the appropriate controls in the ICS domain to mitigate the demonstrated risks. The feedback and discussions reflected the increase in awareness of the subject.
- *Knowledge development:* Students of our department have studied in DESI various controls to counter cyber-physical attacks, including: the deployment of restricted network compartments, custom firewal rules, and the effects of patching and various types of detection. One of the results was a custom signature for Snort [15], a NIDS (Network Intrusion Detection System) to detect unwanted ICS traffic in non-ICS subnets.

6 Conclusions and Future Work

The cyber security posture of the 19 organizations involved in water management is successfully measured by an enhanced ICS benchmark methodology which allows comparison with other organizations and facilitates cyber security dialogues. The benchmark identified various cyber-security related strengths and vulnerabilities in the assessed systems of the water management organizations. Some of the vulnerabilities are easy to solve by well-known controls. However, we also identified cyber security dilemmas in the ICS environment, related with patching, centralized control, and disaster recovery, with no trivial controls to solve these issues. Another observation was the high diversity between the security postures of the participating organizations, partly caused by outsourcing and divided or unclear security responsibilities.

To optimally control ICS-related dilemmas and cyber risk in general, awareness and knowledge is required. We designed and built DESI, a simulator for cyber-physical attack demonstrations, experiments, and solution verification. DESI is a scaled model of a water level management system, including ICS components and a virtualized central office environment. DESI is actively used for demonstrations to various stakeholders as well as students. DESI is also deployed for the development and test of custom controls.

6.1 Future Work

We foresee from our results two types of future work. First, while the assessment of cyber risk in ICS environments remains challenging by a dynamic threat landscape, increasing system complexity, and increasing dependence, continuous research is required to further improve, extend and adapt benchmark methodologies and tools. Second, the successful deployment of DESI to model water

management processes and support cyber-physical attack demonstrations and experiments, invites for further development of the simulation environment and also the deployment of DESI in other areas involving ICS-controlled processes.

Acknowledgment. The Dutch government funds research by universities which aim to generate knowledge which needs to flow to both the education of next generation students and to organizations. This funding scheme is called *'Regionale Aandacht en Actie voor Kenniscirculatie'*, abbreviated *RAAK* which translates into English as *on tar-get*.

References

1. Almalawi, A., Tari, Z., Khalil, I., Fahad, A.: SCADAVT-A framework for SCADA security testbed based on virtualization technology. In: 2013 IEEE 38th Conference on Local Computer Networks (LCN), pp. 639–646. IEEE (2013)
2. Amin, S., Schwartz, G.A., Hussain, A.: In quest of benchmarking security risks to cyber-physical systems. IEEE Netw. **27**(1), 19–24 (2013)
3. CIP: Traffic Light Protocol (TLP), April 2016. https://publicwiki-01.fraunhofer.de/CIPedia/index.php/Traffic_Light_Protocol_%28TLP%29 (2015)
4. ISO: ISO/IEC 27005:2011: Information technology - security techniques - information security risk management. Technical report, ISO (2011)
5. Karnouskos, S.: Stuxnet worm impact on industrial cyber-physical system security. In: IECON 2011–37th Annual Conference on IEEE Industrial Electronics Society, pp. 4490–4494. IEEE (2011)
6. Kivity, A., Kamay, Y., Laor, D., Lublin, U., Liguori, A.: KVM: The linux virtual machine monitor. In: Proceedings of the Linux symposium, vol. 1, pp. 225–230 (2007)
7. Luiijf, E., Ali, M., Zielstra, A.: Assessing and improving SCADA security in the dutch drinking water sector. Int. J. Crit. Infrastruct. Prot. **4**(3), 124–134 (2011)
8. Luiijf, H.: SCADA Security Good Practices for the Drinking Water Sector. TNO, Den Haag (2008)
9. Luiijf, H., te Paske, B.J.: Cyber security of industrial control systems. Technical report, TNO (2015)
10. Macaulay, T., Singer, B.L.: Cybersecurity for industrial control systems: SCADA, DCS, PLC, HMI, and SIS. CRC Press, Boca Raton (2011)
11. Mattioli, R., Moulinos, K.: Analysis of ICS-SCADA cyber security maturity levels in critical sectors. Technical report, ENISA (2015)
12. Mirkovic, J., Benzel, T.: Teaching cybersecurity with deterlab. IEEE Secur. Priv. **10**(1), 73–76 (2012)
13. O'Gorman, J., Kearns, D., Aharoni, M.: Metasploit: The Penetration Tester's Guide. No Starch Press, San Francisco (2011)
14. Pfaff, B., Pettit, J., Amidon, K., Casado, M., Koponen, T., Shenker, S.: Extending networking into the virtualization layer. In: Hotnets (2009)
15. Roesch, M., et al.: Snort: Lightweight intrusion detection for networks. In: Proceedindgs of the 13th USENIX Large Installation Systems Administration Conference, LISA 1999, vol. 99, pp. 229–238. USENIX Association (1999)
16. Siemens: System overview simatic s7–1200, April 2016. http://w3.siemens.com/mcms/programmable-logic-controller/en/basic-controller/s7-1200/system-overview/Pages/default.aspx

Integrated Safety and Security Risk Assessment Methods: A Survey of Key Characteristics and Applications

Sabarathinam Chockalingam[1(✉)], Dina Hadžiosmanović[2], Wolter Pieters[1], André Teixeira[1], and Pieter van Gelder[1]

[1] Faculty of Technology, Policy and Management, Delft University of Technology, Delft, The Netherlands
{S.Chockalingam,W.Pieters,Andre.Teixeira,P.H.A.J.M.vanGelder}@tudelft.nl
[2] Deloitte, Amsterdam, The Netherlands
DHadziosmanovic@deloitte.nl

Abstract. Over the last years, we have seen several security incidents that compromised system safety, of which some caused physical harm to people. Meanwhile, various risk assessment methods have been developed that integrate safety and security, and these could help to address the corresponding threats by implementing suitable risk treatment plans. However, an overarching overview of these methods, systematizing the characteristics of such methods, is missing. In this paper, we conduct a systematic literature review, and identify 7 integrated safety and security risk assessment methods. We analyze these methods based on 5 different criteria, and identify key characteristics and applications. A key outcome is the distinction between sequential and non-sequential integration of safety and security, related to the order in which safety and security risks are assessed. This study provides a basis for developing more effective integrated safety and security risk assessment methods in the future.

Keywords: Integrated safety and security risk assessment
Risk analysis · Risk evaluation · Risk identification
Safety risk assessment · Security risk assessment

1 Introduction

Information technologies and communication devices are increasingly being integrated into modern control systems [1]. These modern control systems are used to operate life-critical systems where the human lives are at stake in case of failure. At the same time, they are often vulnerable to cyber-attacks, which may cause physical impact. An incident in Lodz is a typical example where a cyber-attack resulted in the derailment of 4 trams, and the injury of 12 people [2]. It is therefore becoming increasingly important to address the combination of safety and security in modern control systems.

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 50–62, 2017.
https://doi.org/10.1007/978-3-319-71368-7_5

However, safety and security have been represented by separate communities in both academia and industry [3]. In our context, we think of the safety community as dealing with unintentional/non-malicious threats caused by natural disasters, technical failures, and human error. On the other hand, we think of the security community as dealing with intentional/malicious threats caused by intentional human behavior.

Risk management plays a major role in dealing with both unintentional/non-malicious, and intentional/malicious threats. In the recent years, we have seen a transformation among the researchers of safety and security community to work together especially in risk management. As an example, there are developments of integrated safety and security risk assessment methods [4–10]. Risk assessment is one of the most crucial parts of the risk management process as it is the basis for making risk treatment decisions [11]. The integrated safety and security risk assessment method helps to improve the completeness of risk assessment conducted by covering the interactions between malicious and non-malicious risks. However, a comprehensive review of integrated safety and security risk assessment methods which could help to identify their key characteristics and applications is lacking. Therefore, this research aims to fill this gap by addressing the research question: "What are the key characteristics of integrated safety and security risk assessment methods, and their applications?". The research objectives are:

- **RO 1.** To identify integrated safety and security risk assessment methods.
- **RO 2.** To identify key characteristics and applications of integrated safety and security risk assessment methods based on the analysis of identified methods.

The scope of this analysis covers important features of identified integrated safety and security risk assessment methods mainly, in terms of how these methods are created, and what the existing applications of these methods are. The analysis of identified methods is performed based on the following criteria: I. Citations in the Scientific Literature, II. Steps Involved, III. Stage(s) of Risk Assessment Process Addressed, IV. Integration Methodology, and V. Application(s) and Application Domain. The motivations for selecting these criteria are described in Sect. 5.

The remainder of this paper is structured as follows: Sect. 2 describes the related work, followed by the review methodology in Sect. 3. In Sect. 4, we present the identified integrated safety and security risk assessment methods, and describe the steps involved in these methods. In Sect. 5, we perform the analysis of identified methods based on the criteria that we defined above. Finally, we highlight key characteristics and applications of integrated safety and security risk assessment methods followed by a discussion of future work directions in Sect. 6.

2 Related Work

Cherdantseva et al. presented 24 cybersecurity risk assessment methods for Supervisory Control and Data Acquisition (SCADA) systems [12]. In addition, they

analyzed the presented methods based on the following criteria: I. Aim, II. Application domain, III. Stages of risk management addressed, IV. Key concepts of risk management covered, V. Impact measurement, VI. Sources of data for deriving probabilities, VII. Evaluation method, and VIII. Tool support. Based on the analysis, they suggested the following categorization schemes I. Level of detail and coverage, II. Formula-based vs. Model-based, III. Qualitative vs. Quantitative, and IV. Source of probabilistic data. However, Cherdantseva et al. did not present integrated safety and security risk assessment methods. We used and complemented some of the criteria provided by Cherdantseva et al. to perform the analysis of integrated safety and security risk assessment methods as described in Sect. 5.

Risk assessment methods like Failure Mode and Effects Analysis (FMEA) [13], Fault Tree Analysis (FTA) [14], Component Fault Tree (CFT) [15] have been used by safety community whereas the risk assessment methods like Attack Trees [16], Attack-Countermeasure Trees (ACT) [17], National Institute of Standards and Technology (NIST) 800-30 Risk Assessment [18] have been used by security community. Several authors used these methods as a starting point for the development of integrated safety and security risk assessment methods.

Kriaa et al. highlighted standard initiatives such as ISA-99 (Working Group 7), IEC TC65 (Ad Hoc Group 1), IEC 62859, DO-326/ED-202 that consider safety and security co-ordination for Industrial Control Systems (ICS) [1]. They described various generic approaches that considered safety and security at a macroscopic level of system design or risk evaluation, and also model-based approaches that rely on a formal or semi-formal representation of the functional/non-functional aspects of system. They classified the identified approaches based on the following criteria: I. Unification vs. Integration, II. Development vs. Operational, and III. Qualitative vs. Quantitative. However, Kriaa et al. did not primarily focus on integrated safety and security risk assessment methods that have been already applied in at least one real-case/example involving control system. Also, Kriaa et al. did not identify key characteristics and applications of integrated safety and security risk assessment methods. We included methods such as Failure Mode, Vulnerabilities, and Effect Analysis (FMVEA) [7], Extended Component Fault Tree (CFT) [9], and Extended Fault Tree (EFT) [10] from Kriaa et al. in our work as they satisfy our selection criteria. In addition, we included other methods that satisfy our selection criteria, such as Security-Aware Hazard Analysis and Risk Assessment (SAHARA) [4], Combined Harm Assessment of Safety and Security for Information Systems (CHASSIS) [5], Failure-Attack-CountTermeasure (FACT) Graph [6], and Unified security and safety risk assessment [8].

3 Review Methodology

This section describes the methodology for selecting the integrated safety and security risk assessment methods. The selection of these methods mainly consists of two stages:

- Searches were performed on IEEE Xplore Digital Library, ACM Digital Library, Scopus, DBLP, and Web of Science – All Databases. The search-strings were constructed from keywords "Attack", "Failure", "Hazard", "Integration", "Risk", "Safety", "Security", and "Threat". DBLP provided a good coverage of relevant journals and conferences.
- Methods were selected from the search results according to the following criteria:

– The method should address any or all of the following risk assessment stages: risk identification, risk analysis, and/or risk evaluation.
– The method should consider both unintentional and intentional threats.
– The method should have been already applied in at least one real-case/example involving control system.
– The literature should be in English language.

Once an integrated safety and security risk assessment method was selected, the scientific literature that cited it was also traced.

4 Integrated Safety and Security Risk Assessment Methods

This section presents the identified integrated safety and security risk assessment methods, and describes the steps involved in these methods. This section aims to address the RO 1. Based on the review methodology described in Sect. 3, we have identified 7 integrated safety and security risk assessment methods: I. SAHARA [4], II. CHASSIS [5], III. FACT Graph [6], IV. FMVEA [7], V. Unified Security and Safety Risk Assessment [8], VI. Extended CFT [9], and VII. EFT [10].

4.1 SAHARA Method

The steps involved in the SAHARA method [4] are as follows: I. The ISO 26262 – Hazard Analysis and Risk Assessment (HARA) approach is used in a conventional manner to classify the safety hazards according to the Automotive Safety Integrity Level (ASIL), and to identify the safety goal and safe state for each identified potential hazard; II. The attack vectors of the system are modelled. The STRIDE method is used to model the attack vectors of the system [4,19]; III. The security threats are quantified according to the Required Resources (R), Required Know-how (K), and Threat Criticality (T); IV. The security threats are classified according to the Security Level (SecL). SecL is determined based on the level of R, K, and T; V. Finally, the security threats that may violate the safety goals ($T > 2$) are considered for the further safety analysis.

4.2 CHASSIS Method

The steps involved in the CHASSIS method [5] are as follows: I. The elicitation of functional requirements which involve creating the use-case diagrams that incorporates the users, system functions and services; II. The elicitation of safety and security requirements which involve creating misuse case diagram based on the identified scenarios for safety and security involving faulty-systems and attackers respectively; III. Trade-off discussions are used to support the resolution of conflict between the safety, and security mitigations.

4.3 FACT Graph Method

The steps involved in the FACT Graph method [6] are as follows: I. The fault trees of the system analyzed are imported to start the construction of FACT graph; II. The safety countermeasures are attached to the failure nodes in the FACT graph; III. The attack trees of the system analyzed are imported to the FACT graph in construction. This is done by adding an attack-tree to the failure node in the FACT graph with the help of OR gate, if the particular failure may also be caused by an attack; IV. The security countermeasures are attached to the attack nodes in the FACT graph. This could be done based on the ACT technique [17].

4.4 FMVEA Method

The steps involved in the FMVEA method [7] are as follows: I. A functional analysis at the system level is performed to get the list of system components; II. A component that needs to be analyzed from the list of system components is selected; III. The failure/threat modes for the selected component are identified; IV. The failure/threat effect for each identified failure/threat mode is identified; V. The severity for the identified failure/threat effect is determined; VI. The potential failure causes/vulnerabilities/threat agents are identified; VII. The failure/attack probability is determined. Schmittner et al. described the attack probability as the sum of threat properties and system susceptibility ratings. The threat properties is the sum of motivation and capabilities ratings, whereas the system susceptibility is the sum of reachability and unusualness of the system ratings; VIII. Finally, the risk number is determined, which is the product of severity rating and failure/attack probability.

4.5 Unified Security and Safety Risk Assessment Method

The steps involved in the Unified Security and Safety Risk Assessment method [8] are as follows: I. The system boundary, system functions, system and data criticality, system and data sensitivity are identified; II. The threats, hazards, vulnerabilities, and hazard-initiating events are identified; III. The current and planned controls are identified; IV. The threat likelihood is determined; V. The hazard likelihood is determined; VI. The asset impact value is determined; VII. The combined safety-security risk level is determined; VIII. The control recommendations are provided; IX. The risk assessment reports are provided.

4.6 Extended CFT Method

The steps involved in the extended CFT method [9] are as follows: I. The CFT for the system analyzed is developed. This could be done based on [15]; II. The CFT is extended by adding an attack tree to the failure node with the help of OR gate, if the particular event may also be caused by an attack; III. The qualitative analysis is conducted by calculating Minimal Cut Sets (MCSs) per top level event. MCSs containing only one event would be single point of failure which should be avoided; IV. The quantitative analysis is conducted by assigning values to the basic events. Therefore, MCSs containing only safety events would have a probability P, MCSs containing only security events would have a rating R, MCSs containing both safety and security events would have a tuple of probability and rating (P, R).

4.7 EFT Method

The steps involved in the EFT method [10] are as follows: I. The fault tree for the system analyzed is developed by taking into account the random faults; II. The developed fault tree is extended by adding an attack tree to the basic or intermediate event in the fault tree, if the particular event in the fault tree may also be caused by malicious actions. The attack tree concept used in the development of EFT is based on [20]; III. The quantitative analysis is performed based on the formulae defined in [10] which help to calculate the top event probability.

5 Analysis of Integrated Safety and Security Risk Assessment Methods

This section performs the analysis of integrated safety and security risk assessment methods based on the criteria: I. Citations in the Scientific Literature, II. Steps Involved, III. Stage(s) of Risk Assessment Process Addressed, IV. Integration Methodology, and V. Application(s) and Application Domain. This allows us to identify key characteristics and applications of integrated safety and security risk assessment methods. This section aims to address the RO 2.

The integrated safety and security risk assessment methods described in the previous section are listed in Table 1. In Table 1, country is the country of the first author of the paper and citations is the number of citations of the paper according to Google Scholar Citation Index as on 31st August 2016.

From Table 1, we observe that the researchers started to recognize the importance of integrated safety and security risk assessment methods which resulted in the increase in number of papers produced especially during 2014, and 2015. The largest number of citations (63) is acquired by the EFT method published in 2009. The second most cited paper, among analyzed, with 17 citations, is the Extended CFT method published in 2013. However, it is understandable that the methods published during the last few years received lower number of citations ranging from 1 to 5.

Table 1. List of integrated safety and security risk assessment methods (Ordered by the number of citations)

Integrated safety and security risk assessment method	Year	Country	Citations
EFT [10]	2009	Italy	63
Extended CFT [9]	2013	Germany	17
FACT Graph [6]	2015	Singapore	5
CHASSIS [5]	2015	Austria	4
FMVEA [7]	2014	Austria	4
SAHARA [4]	2015	Austria	2
Unified Security and Safety Risk Assessment [8]	2014	Taiwan	1

Based on the steps involved in each method as described in Sect. 4, we conclude that there are two types of integrated safety and security risk assessment methods:

- Sequential Integrated Safety and Security Risk Assessment Method: In this type of method, the safety risk assessment, and security risk assessment are performed in a particular sequence. For instance, the Extended CFT method starts with the development of CFT for the system analyzed. Later, the attack tree is added to extend the developed CFT. This method starts with the safety risk assessment followed by the security risk assessment. Methods such as SAHARA, FACT Graph, Unified Security and Safety Risk Assessment, Extended CFT, and EFT come under the sequential type.
- Non-sequential Integrated Safety and Security Risk Assessment Method: In this type of method, the safety risk assessment, and security risk assessment are performed without any particular sequence. For instance, in the FMVEA method, the results of safety risk assessment and security risk assessment are tabulated in the same table without any particular sequence. Methods such as FMVEA and CHASSIS come under the non-sequential type.

Cherdantseva et al. used 'stage(s) of risk management process addressed' as a criteria to analyze the identified cybersecurity risk assessment methods for SCADA systems [12]. We adapted and used this criteria as 'stage(s) of risk assessment process addressed' because the major focus of our research is on risk assessment. This criteria will allow us to identify the predominant stage(s) of risk assessment process addressed by the integrated safety and security risk assessment methods.

A risk assessment process consists of typically three stages:

- Risk Identification: This is the process of finding, recognizing and describing the risks [21].
- Risk Analysis: This is the process of understanding the nature, sources, and causes of the risks that have been identified and to estimate the level of risk [21].

- Risk Evaluation: This is the process of comparing risk analysis results with risk criteria to make risk treatment decisions [21].

Table 2 highlights the integrated safety and security risk assessment method and the corresponding stage(s) of the risk assessment process addressed. This is done based on the definitions of risk identification, risk analysis, and risk evaluation. We also take into account the safety risk assessment method, and security risk assessment method that were combined in the integrated safety and security risk assessment method.

Table 2. Stage(s) of risk assessment process addressed

Integrated safety and security risk assessment method	Risk identification	Risk analysis	Risk evaluation
SAHARA	✓	✓	×
CHASSIS	✓	×	×
FACT Graph	✓	×	×
FMVEA	✓	✓	×
Unified security and safety risk assessment	✓	✓	✓
Extended CFT	✓	✓	×
EFT	✓	✓	×

In this Table 2, ✓(×) indicates that the particular method addressed (did not address) the corresponding risk assessment stage.

From Table 2, we understand that all methods addressed the risk identification, 5 out of 7 methods addressed the risk analysis, whereas only 1 out of 7 methods addressed the risk evaluation stage of the risk assessment process. This implies that the risk evaluation stage is not given much attention compared to the other stages of the risk assessment process in the integrated safety and security risk assessment methods. Cherdantseva et al. also highlighted that the majority of the cybersecurity risk assessment methods for SCADA systems concentrates on the risk identification and risk analysis stages of the risk assessment process [12]. The risk evaluation phase in the Unified Security and Safety Risk Assessment method starts by comparing the risk analysis result with the suggested four levels of risk to determine the appropriate level of risk. Once the level of risk is determined, the risk treatment decision is made accordingly.

We used the criteria 'Integration methodology' because this will allow us to understand which combination of safety, and security risk assessment methods are being used in the integrated safety and security risk assessment methods as summarized in Table 3.

Table 3. Integration methodology

Integrated safety and security risk assessment method	Safety risk assessment method	Security risk assessment method
SAHARA	ISO 26262: HARA	Variation of ISO 26262: HARA
CHASSIS	Safety misuse case (Involving faulty-systems)	Security misuse case (Involving attackers)
FACT Graph	Fault tree	Attack tree
FMVEA	FMEA	Variation of FMEA
Unified security and safety risk assessment	Variation of NIST 800-30 security risk estimation	NIST 800-30 security risk estimation
Extended CFT	CFT	Attack tree
EFT	Fault tree	Attack tree

From Table 3, we observe that there are four ways in which the integrated safety and security risk assessment methods have been developed:

- Integration through the combination of a conventional safety risk assessment method and a variation of the conventional safety risk assessment method for security risk assessment. The methods SAHARA and FMVEA come under this category.
- Integration through the combination of a conventional security risk assessment method and a variation of the conventional security risk assessment method for safety risk assessment. The Unified Security and Safety Risk Assessment method come under this category.
- Integration through the combination of a conventional safety risk assessment method and a conventional security risk assessment method. The methods FACT Graph, Extended CFT, and EFT come under this category.
- Others - There is no conventional safety risk assessment, and conventional security risk assessment method used in the integration. The CHASSIS method come under this category. The CHASSIS method used a variation of Unified Modeling Language (UML)-based models for both the safety and security risk assessment.

We used the criteria 'Application(s) and Application domain' because this will allow us to understand the type of application(s), and the corresponding application domain of integrated safety and security risk assessment methods. Table 4 highlights the integrated safety and security risk assessment method and the corresponding application(s) and application domain.

From Table 4, we observe that 4 methods were applied in the transportation domain, 2 methods were applied in the power and utilities domain, and 1 method was applied in the chemical domain. The major development, and application of integrated safety and security risk assessment methods, is in the transportation

Table 4. Application(s) and application domain

Integrated safety and security risk assessment method	Application(s)	Application domain
SAHARA	Battery Management System use-case [4]	Transportation
CHASSIS	Over The Air (OTA) system [5], Air traffic management remote tower example [22]	Transportation
FACT Graph	Over-pressurization of a vessel example [6]	Power and Utilities
FMVEA	OTA system [5], Telematics control unit [7], Engine test-stand [23], Communications-based train control system [24]	Transportation
Unified Security and Safety Risk Assessment	High pressure core flooder case-study [8]	Power and Utilities
Extended CFT	Adaptive cruise control system [9]	Transportation
EFT	Release of toxic substance into the environment example [10]	Chemical

domain. The Threat Horizon 2017 listed "death from disruption to digital services" as one of the threats especially in the transportation and medical domain [25]. In the transportation domain, there is a potential for cyber-attacks which compromises system safety and result in the injury/death of people which was illustrated by a tram incident in Lodz [2].

6 Conclusions and Future Work

In this paper, we have identified 7 integrated safety and security risk assessment methods. Although we cannot completely rule out the existence of other unobserved integrated safety and security risk assessment methods that fulfil our selection criteria, the review methodology that we adopted helped to ensure the acceptable level of completeness in the selection of these methods. Based on the analysis, we identified key characteristics and applications of integrated safety and security risk assessment methods.

- There are two types of integrated safety and security risk assessment methods based on the steps involved in each method. They are: a. Sequential, and b. Non-sequential.
- There are four ways in which the integrated safety and security risk assessment methods have been developed. They are: a. The conventional safety

risk assessment method as the base and a variation of the safety risk assessment method for security risk assessment, b. The conventional security risk assessment method as the base and a variation of the security risk assessment method for safety risk assessment, c. A combination of a conventional safety risk assessment method, and a conventional security risk assessment method, d. Others.
- Risk identification and risk analysis stages were given much attention compared to the risk evaluation stage of the risk assessment process in the integrated safety and security risk assessment methods.
- Transportation, power and utilities, and chemical were the three domains of application for integrated safety and security risk assessment methods.

The identified integrated safety and security risk assessment methods did not take into account real-time system information to perform dynamic risk assessment which needs to be addressed to make it more effective in the future. This study provided the list of combinations of safety, and security risk assessment methods used in the identified integrated safety and security risk assessment methods. In the future, this would act as a base to investigate the other combinations of safety, and security risk assessment methods that could be used in the development of more effective integrated safety and security risk assessment methods. Furthermore, this study provided the type of applications and application domains of the identified integrated safety and security risk assessment methods. In the future, this would act as a starting point to evaluate the applicability of these methods in the other domains besides transportation, power and utilities, and chemical.

Acknowledgements. This research received funding from the Netherlands Organisation for Scientific Research (NWO) in the framework of the Cyber Security research program. This research has also received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement ICT-318003 (TREsPASS). This publication reflects only the authors' views and the Union is not liable for any use that may be made of the information contained herein.

References

1. Kriaa, S., Pietre-Cambacedes, L., Bouissou, M., Halgand, Y.: A survey of approaches combining safety and security for industrial control systems. Reliab. Eng. Syst. Safety **139**, 156–178 (2015)
2. RISI Database: Schoolboy Hacks into Polish Tram System (2016). http://www.risidata.com/Database/Detail/schoolboy_hacks_into_polish_tram_system
3. Stoneburner, G.: Toward a unified security-safety model. Computer **39**(8), 96–97 (2006)
4. Macher, G., Höller, A., Sporer, H., Armengaud, E., Kreiner, C.: A combined safety-hazards and security-threat analysis method for automotive systems. In: Koornneef, F., van Gulijk, C. (eds.) SAFECOMP 2015. LNCS, vol. 9338, pp. 237–250. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24249-1_21

5. Schmittner, C., Ma, Z., Schoitsch, E., Gruber, T.: A case study of FMVEA and CHASSIS as safety and security co-analysis method for automotive cyber physical systems. In: Proceedings of the 1st ACM Workshop on Cyber Physical System Security (CPSS), pp. 69–80 (2015)
6. Sabaliauskaite, G., Mathur, A.P.: Aligning cyber-physical system safety and security. In: Cardin, M.A., Krob, D., Cheun, L.P., Tan, Y.H., Wood, K. (eds.) Complex Systems Design & Management Asia 2014, pp. 41–53. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-12544-2_4
7. Schmittner, C., Ma, Z., Smith, P.: FMVEA for safety and security analysis of intelligent and cooperative vehicles. In: Bondavalli, A., Ceccarelli, A., Ortmeier, F. (eds.) SAFECOMP 2014. LNCS, vol. 8696, pp. 282–288. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10557-4_31
8. Chen, Y., Chen, S., Hsiung, P., Chou, I.: Unified security and safety risk assessment - a case study on nuclear power plant. In: Proceedings of the International Conference on Trusted Systems and their Applications (TSA), pp. 22–28 (2014)
9. Steiner, M., Liggesmeyer, P.: Combination of safety and security analysis - finding security problems that threaten the safety of a system. In: Workshop on Dependable Embedded and Cyber-physical Systems (DECS), pp. 1–8 (2013)
10. Fovino, I.N., Masera, M., De Cian, A.: Integrating cyber attacks within fault trees. Reliab. Eng. Syst. Safety **94**(9), 1394–1402 (2009)
11. European Union Agency for Network and Information Security (ENISA). The Risk Management Process (2016). https://www.enisa.europa.eu/activities/risk-management/current-risk/risk-management-inventory/rm-process
12. 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. **56**, 1–27 (2016)
13. International Electrotechnical Commission (IEC).: IEC 60812: Analysis Techniques for System Reliability - Procedures for Failure Mode and Effects Analysis (2006)
14. Lee, W.S., Grosh, D.L., Tillman, F.A., Lie, C.H.: Fault tree analysis, methods, and applications - a review. IEEE Trans. Reliab. **R–34**(3), 194–203 (1985)
15. Kaiser, B., Liggesmeyer, P., Mackel, O.: A new component concept for fault trees. In: Proceedings of the 8th Australian Workshop on Safety Critical Systems and Software (SCS), vol. 33, pp. 37–46 (2003)
16. Schneier, B.: Attack trees. Dr. Dobb's J. **24**(12), 21–29 (1999)
17. Roy, A., Kim, D.S., Trivedi, K.S.: Scalable optimal countermeasure selection using implicit enumeration on attack countermeasure trees. In: Proceedings of the 42nd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), pp. 1–12 (2012)
18. National Institute of Standards and Technology (NIST): Risk Management Guide for Information Technology Systems (2002)
19. Scandariato, R., Wuyts, K., Joosen, W.: A descriptive study of Microsoft's threat modeling technique. Requirements Eng. **20**(2), 163–180 (2015)
20. Fovino, I.N., Masera, M.: Through the description of attacks: a multidimensional view. In: Górski, J. (ed.) SAFECOMP 2006. LNCS, vol. 4166, pp. 15–28. Springer, Heidelberg (2006). https://doi.org/10.1007/11875567_2
21. International Organisation for Standardization (ISO): ISO 31000: 2009 - Risk Management - Principles and Guidelines (2009)

22. Raspotnig, C., Karpati, P., Katta, V.: A combined process for elicitation and analysis of safety and security requirements. In: Bider, I., Halpin, T., Krogstie, J., Nurcan, S., Proper, E., Schmidt, R., Soffer, P., Wrycza, S. (eds.) BPMDS/EMMSAD -2012. LNBIP, vol. 113, pp. 347–361. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-31072-0_24
23. Schmittner, C., Gruber, T., Puschner, P., Schoitsch, E.: Security application of failure mode and effect analysis (FMEA). In: Bondavalli, A., Di Giandomenico, F. (eds.) SAFECOMP 2014. LNCS, vol. 8666, pp. 310–325. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10506-2_21
24. Chen, B., Schmittner, C., Ma, Z., Temple, W.G., Dong, X., Jones, D.L., Sanders, W.H.: Security analysis of urban railway systems: the need for a cyber-physical perspective. In: Koornneef, F., van Gulijk, C. (eds.) SAFECOMP 2015. LNCS, vol. 9338, pp. 277–290. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24249-1_24
25. Information Security Forum.: Threat Horizon 2017: Dangers Accelerate (2015). https://www.securityforum.org/uploads/2015/03/Threat-Horizon_2017_Executive-Summary.pdf

Railway Station Surveillance System Design: A Real Application of an Optimal Coverage Approach

Francesca De Cillis[1], Stefano De Muro[2], Franco Fiumara[3], Roberto Setola[1], Antonio Sforza[4], and Claudio Sterle[4(✉)]

[1] Complex Systems and Security Laboratory, University Campus Bio-Medico of Rome, Rome, Italy
{f.decillis,r.setola}@unicampus.it
[2] Security Department - Technical Area, Rete Ferroviaria Italiana, Rome, Italy
s.demuro@rfi.it
[3] Security Department - Directorate, Ferrovie dello Stato Italiane, Rome, Italy
[4] Department of Electrical Engineering and Information Technology, University Federico II of Naples, Naples, Italy
{antonio.sforza,claudio.sterle}@unina.it

Abstract. The design of an effective and efficient surveillance system is fundamental for the protection of the Critical Infrastructures. In a railway station, this requirement turns on as an urgent prerequisite: for its intrinsic nature, a station represents a complex environment to be monitored for both safety and security reasons. In this work, we show how the video surveillance system of a real terminal railway station can be effectively designed in terms of sensor placement problem using an *optimal coverage* approach. The obtained results confirm the effectiveness of the proposed method in supporting security experts in both the design and reconfiguration of a surveillance system, in order to increase the asset security level.

Keywords: Railway security · Video-surveillance · Sensor placement
Optimal coverage

1 Introduction

Risk prevention and preparedness in transportation systems represent key requirements for Homeland Security. Transportation systems represent a crucial element to preserve the growth and the development of a country. In this context, the railway infrastructure surely serves as the backbone of a country: capillary disseminated across the globe, it enables the transportation of people and goods for long distances, being the only form of transportation in many countries.

Due to the major role played in the society, the railway infrastructure is one of the most challenging target to protect in the security field [18]. As open and wide geographically deployed asset, the railway is indeed difficult to secure by nature. The high number of entry points and the heavy crowds, together with the inability to adopt airport-style security checks make the railway infrastructure definitively a soft target for potential

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 63–74, 2017.
https://doi.org/10.1007/978-3-319-71368-7_6

assailants (i.e., criminals, terrorists, vandals, copper thieves, etc.). In addition, potential acts carried out against a railway asset rapidly gain great attention from the mass media and the people, making the target rather appealing and the likelihood of copycat attacks more probable. Moreover, malicious acts executed against railway stations may cause many casualties, devastations, service outages with an obvious impact on the people's everyday life (e.g., mobility issues, fear, lack of confidence, etc.). In this context, as reported in [7, 8] railway stations are surely the most vulnerable and affected target by terrorist attacks and the protection of railway stations represents a priority in the security management field.

Assuming that a perfectly secure asset is far from reality, security systems may in general be very effective in preventing crimes and increasing the people's security perception. Focusing on railway stations, Closed-Circuit Television (CCTV), access control/intrusion detection and abnormal sound detection technologies represent the most commonly used security systems [3]. Especially in case of shortage of staff or stations deployed on large areas, we may understand the crucial role played by the CCTV systems for security and protection, with a prominent and twofold role. They indeed extend the monitoring activity to every point of the station and recording enables the on-line/off-line identification of the cause of a potential incident. At the same time, cameras help to bolster the passenger feelings of security as well as to alert potential criminals that they are being monitored.

In this context, the chance to have specific methodologies devoted to optimize the location of cameras, that is to minimize the number of cameras or to maximize the area coverage for a specific asset, may help in installation costs reduction and cameras' maintenance as well as in reducing the operator workload. In the literature, this problem is referred to as the Sensor Placement Problem (*SPP*), i.e. the problem of designing a sensor network, covering and monitoring an area of interest. Sterle et al. in [19] have tackled *SPP* by an optimal coverage approach. In this work, we experience and validate this approach on a real terminal railway station provided by RFI (Rete Ferroviaria Italiana). For the sake of confidentiality, the name of the station under investigation is not reported, but a representation of its layout is provided.

The remainder of the paper is arranged as follows: in Sect. 2 we provide a brief overview about contributions in the railway CCTV surveillance and sensor placement fields; Sect. 3 recalls the phases of the used approach showing its application to a real use case scenario. In Sect. 4, we illustrate the obtained results, discussing the related outcomes. Finally, conclusions are devoted to the future work perspectives.

2 Railway Station Surveillance and Sensor Placement Problem

Surveillance cameras are extensively used in railways, and especially in stations, to improve passenger safety and security. For these reasons, in the last few years a great effort has been made by the industrial and scientific communities to improve the CCTV system effectiveness, both in terms of technologies and design decisions. In the following, we provide a review of some contributions in this field in order to better frame our work in the security context.

According to [13], the use of the *first generation* CCTV systems for public security started around the 1967, when the visual information was entirely processed by human operators. These analogue CCTV systems mainly consisted of cameras located in multiple remote locations, connected to a set of monitors via switches, and the visual information was entirely processed by human operators [16, 21]. Despite some controversies due to privacy violation and potential redundant investments, early CCTV systems proved their usefulness in the fight against crime, from different perspectives. From a practical point of view, cameras provide real-time visual feedback about the asset to the security operators, allowing the promptly identification of a crime or of its causes. This duty does not represent the only CCTV systems peculiarity. Cameras serve also for psychological purposes. They indeed bolster the passengers' feelings of security and alert potential criminals that they are being monitored, helping to increase security perception and crime prevention.

For these reasons, starting from 2004 the number of cameras installed into railway stations sharply increases worldwide [13]. Improving the CCTV systems effectiveness, both in terms of technology and design decision, has become a priority for the industrial and scientific communities. According to this requirement, in the last few years CCTV systems experienced a technology evolution that led to the *second* and *third* CCTV generation. Current intelligent video-surveillance systems today use digital computing, advanced image processing and artificial intelligence algorithms for different purposes. In the second generation CCTV systems are connected to high performance computers to develop semi-automatic systems. Second-generation (or advanced) video-based surveillance are today able to identify passengers [17], to detect a specific activity [15], to detect, track and re-identify objects [22], to perform multi-camera tracking and cooperative video surveillance [23].

Combining computer vision technology with CCTVs, these systems provide operators with automatic real-time detection events tools, aiding the user to promptly recognise potential crime events. The development of distributed automatic surveillance systems, using multi-camera/multi-sensor and fusion of information retrieved across several devices, led to the *third generation* or *intelligent* video-surveillance systems [21]. Classic applications concern with objects detection, tracking and re-identification, passengers' identification and activity detection [24].

In the object detection and tracking problem, the objective is to detect a target object and to track it in consecutive video frames. Despite the fact it is usually presented in the literature as a single problem, it fundamentally entails specific and separated issues: the object detection and the object tracking. Specifically, the object identification concerns the estimation of the object location in a specific region of successive images. It may prove a challenging task, especially since objects can have rather complicated structures and may change in shape, size, location and orientation over subsequent frames. For these reasons, several algorithms have been proposed in the literature to solve the object detection problem. These methodologies commonly use two main conventional approaches to object detection, known as *temporal difference* and *background subtraction*.

In the first approach, the object is detected by subtracting two consecutive frames, while the second technique deals with the subtraction from each video frames of a *background* or *reference model*. The temporal difference technique is very adaptive and

usually presents good performance in dynamic environments; nevertheless, it has a poor performance on extracting all the relevant object pixels. On the other hand, background subtraction has a better performance extracting object information but it may be sensitive to dynamic changes in the environment. This lead to the development of *adaptive background subtraction* techniques that involve the implementation of a background model, constantly upgraded in order to avoid poor detection when changes in the environment may occur. Given its effectiveness, this technique is today widely used for the robust detection and re-identification of left object in public areas, such as stations, as illustrated in [21] and [25]. Concerning objects' tracking, techniques can be split into two main approaches, according to [21]: 2-D models with or without explicit shape models and 3-D models. In general, the model-based approach uses the geometrical knowledge of the objects to track, which in surveillance applications for railway stations are usually represented by people, luggage, backpacks, etc. As illustrated in the literature [1, 21], the knowledge of the object is usually retrieved by computing the object's appearance as a function of its position with respect to the camera. Once a priori knowledge is available, several algorithms (i.e. Kalman filter, Bayesian filter, particle filter, etc. [21]) can be used for tracking the object in different operating conditions (e.g., changing and/or poor illumination [11]), that are able to deal with occlusions and/or collisions [9].

As a further step to objects detection and tracking, motion and behavioural analysis represents one of the most intriguing challenge in the last generation CCTV systems. It deals with the identification of specific activities and behaviours of the tracked objects and corresponds to a classification problem of time-varying feature data that are provided by the previous stages. Given a specific object (e.g., a person), the behavioural analysis consists in matching a current video-sequence to pre-labelled sequences, which represent prototypical actions that the system collects during the training phase. Several approaches have been proposed in the literature for the behavioural analysis, such as hidden Markov models, Bayesian networks, neural networks, etc. [21]. Focusing on the behavioural analysis for applications in railway stations, Chow et al. in [6] present a neural network for crowd monitoring implemented at the Hong Kong underground stations. A CCTV system for detecting person crossing or walking on the platform is indeed presented in [14].

Despite the undeniable usefulness of a CCTV system, the shortage of staff connected with a large number of cameras is the major reason of the overlooking of incidents in stations as shown by Sun et al. in [20], reporting that railway operators usually have to monitor more than 15 cameras simultaneously. Although intelligent CCTV systems are able to support the role of the operator in the monitoring operation, it is clear that the supervision of several cameras represents a very laborious task. In addition, the continuous cost reduction of the cameras further contributes in the rising number of sensors typically installed into a railway asset.

In this context, it is fundamental to design an effective and efficient system in terms of number of CCTVs and their coverage performance. This problem has been widely studied in literature as the Sensor Placement Problem (*SPP*). For a complete review on the *SPP*, readers could refer to the surveys by Guvensan and Gokhan [10], Mavrinac and Chen [12] and Sterle et al. [19].

In its two basic variants, *SPP* can be summarized as follows: determining the minimum number and the positions of the sensors guaranteeing the total control of the area under investigations; determining the position of a predefined number of sensors, maximizing the portion of controlled area. These two variants correspond to two well-known coverage optimization problems: the set covering and the maximal covering optimization problems, known as *SCP* and *MCP*. For further details on the topic, readers could refer to the contributions reported in [2, 4, 5] for a review of the main exact and heuristic solving approaches for these problems.

3 Application of the Optimal Coverage Approach to a Real Case

In this paper, the optimal coverage approach presented in [19] has been applied to a real use case scenario related to a railway station whose layout can be recognized in several Italian cities. For the sake of brevity, we do not describe the procedure in detail. We limit ourselves to illustrate the phases composing the methodology and its application to the real use case under investigation. Specifically, the phases composing the methodology are related to:

- Area of interest and input data
- Coverage analysis and coverage matrix
- Modeling of the coverage problem and model solution.

3.1 Area of Interest and Input Data

The area of interest is a terminal railway station of RFI composed by two main sub-areas (Fig. 1). The first, referred to as *S1*, corresponds to the atrium and to the area before the turnstiles (78 × 26.5 m). Its access corridors and the side entrances have not been taken into account, since the control of these zones is performed by dedicated CCTVs. The second sub-area, referred to as *S2*, corresponds to the platforms and rails (276 × 26.5 m).

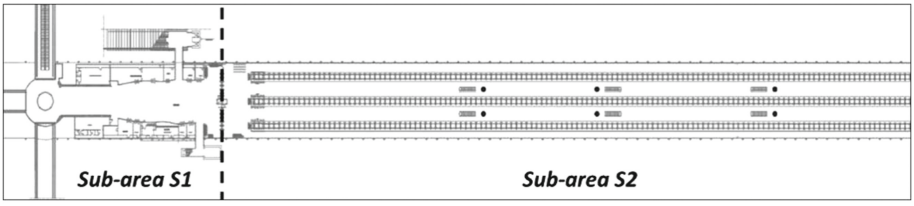


Fig. 1. RFI terminal station under investigation with the two sub-areas S1 and S2.

These two sub-areas can be treated separately because of the presence of a tarpaulin, creating a physical separation between them.

The geometrical and spatial information describing the area of interest has to be processed in order to generate the input for the successive phases of the approach.

To this aim, we discretize the area of interest by a grid of points and we define the potential CCTVs locations.

Discretization of the Area of Interest

The discretization of the area of interest is performed overlapping a grid of points with a prefixed step size on the area of interest. Obviously, the smaller is the step, the higher is the quality of the grid representation. The step sizes of *S1* and *S2* have been chosen equal to 1 and 2.5 m, respectively. The choice of a smaller step size for *S1* has a twofold motivation. The first concerns the fact that *S1* is a crowded area where several services are located (administrative and ticket offices, stores, etc.). The second is related to the geometry of *S1*, which is smaller and less regular than *S2*. The exemplifications of the discretization for the two sub-areas are represented in Figs. 2a and 3a respectively, where the grid is highlighted in red.

In this phase, the presence of obstacles that may potentially interdict the view of the CCTVs should be considered as well. Hence, the grid of *S1* has been then further refined taking into account the ticket machines and the checkpoint in the atrium. Moreover, the points of the grid corresponding to entrances and indentations along the boundaries have been deleted. This deletion has been performed since the coverage of these points can provide understandable distortions in the design of the surveillance system. The final *S1* grid is provided in Fig. 2b, where the points of the grid are represented by grey dots and the boundaries of the area and obstacles by straight black lines. Whereas the *S2* grid does not present physical obstacles. However, the presence of a train on a rail can significantly interdict the visual capability of potential CCTVs. For this reason, we introduced a virtual "linear" obstacle whose extension is equal to the length of the train. This allows us on the one side to simulate the effect of a train on the CCTV view and

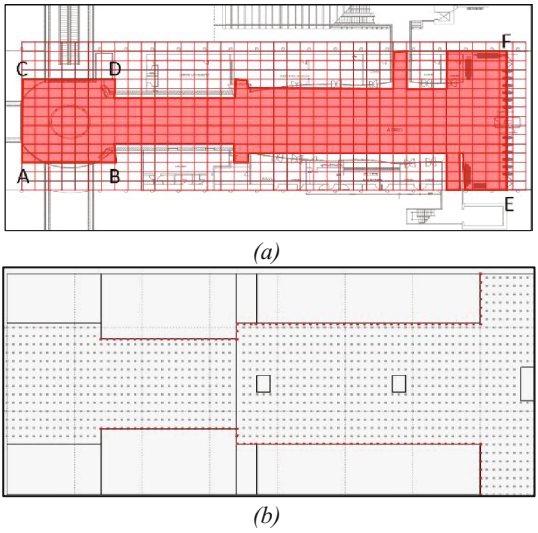


Fig. 2. Schematization of sub-area *S1*. *(a) S1* area under investigation. *(b) S1* grid schematization. (Color figure online)

on the other side to consider all the points between the two platforms as points to be covered. A representation of the final *S2* grid is provided in Fig. 3b. Also in this case, the points of the grid are represented by grey dots, while the boundaries of the area and virtual obstacles are represented by straight black lines.

Definition of the Potential Sensor Locations

The potential CCTV locations have to be defined taking into account the geometrical and spatial features, the constraints of the area under investigation and on the basis of the security requirements, that come from the experience or are imposed by regulations.

With reference to the sub-area *S1*, the potential CCTV locations correspond to all the points along the boundaries with a 1 m step size, except for those points located along the sides of the square ABCD and along the side EF of Fig. 2a.

With reference to the sub-area *S2*, potential locations correspond to all the points along the boundaries and along the center lines of the platforms, with a 2.5 m step size.

The other parameters defining the potential locations are equal for both sub-areas. In particular, in this application the installation height and the tilt angle are fixed and equal to 3 m and 45°, respectively. For each position of a CCTV, the orientation angle can vary between 0° and 315° (step of 45°) with 8 overlapping orientations.

Potential CCTV locations for the two sub-areas are represented by red dots in Figs. 2b and 3b.

3.2 Coverage Analysis and Coverage Matrix

The monitoring capability of a CCTV is expressed by its coverage area, i.e. the portion of the area of interest that it is able to cover. In order to determine it, we have to know the coverage angle and the coverage ray of each CCTV. Based on the technological features of the cameras provided by the security experts, we considered a coverage angle

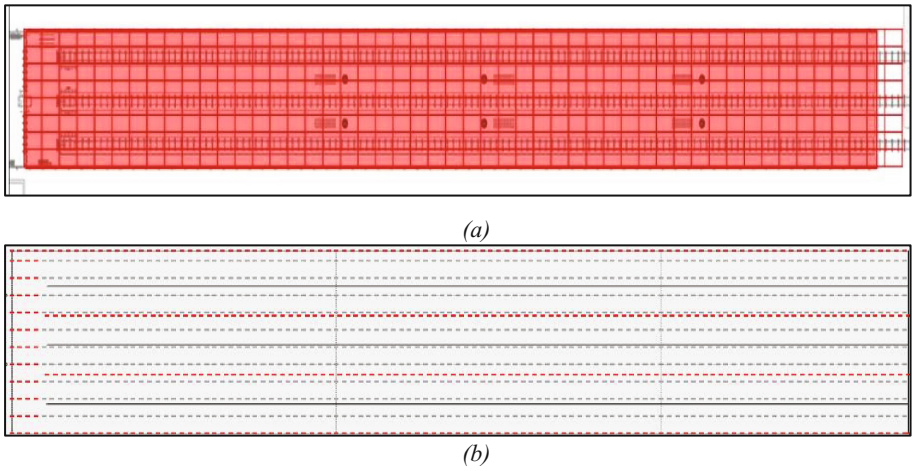


Fig. 3. Schematization of sub-area *S2*. *(a) S2* area under investigation*; (b) S2* grid schematization. (Color figure online)

equal to 90° and a coverage ray equal to 25 m for the sub-area *S1*. Concerning the sub-area *S2*, we considered CCTVs with coverage angle equal to 90° and coverage ray equal to 50 m. As explained above, the different values of the coverage ray are determined by the different security needs and the geometry of the two sub-areas. Using this information, the coverage (or visibility) analysis is performed. It consists in determining, for each CCTV placed at a given potential location with a possible orientation, the subset of grid points that it is able to cover. The result of this analysis is the construction of a 0/1 matrix, referred to as *coverage matrix*, with as many rows as the number of potential locations multiplied by possible orientations and as many columns as the number of grid points. With reference to the sub-area *S1*, the number of CCTV potential locations is equal to 151 and, for each of them, 8 possible orientations are considered, for a total of 1208 locations with orientation. The grid is composed by 1281 points and the size of the coverage matrix is [1208 × 1281]. Concerning the sub-area *S2*, the number of CCTV potential locations is equal to 584 and, also in this case, for each of them 8 possible orientations are considered for a total of 4672 locations with orientation. The number of grid points is equal to 2184 and the size of the coverage matrix is [4672 × 2184].

3.3 Modeling of the Coverage Problem and Solution of the Model

In this phase, the optimization model that best fits with the security design requirements has to be chosen. As discussed in [19], different *integer linear programming* (ILP) optimization models can be used to design a surveillance system. The choice evidently depends on the specific security needs that have to be achieved. For the sake of brevity, in this work we just focus on the results of the Set Covering (*SC*) and of the Maximal Covering (*MC*) models. Specifically, the first is devoted to determine the minimum number of CCTVs to locate in order to control the whole area under investigation, while the second aims for maximizing the area coverage with a prefixed number of CCTVs. These ILP models can be optimally solved using a commercial software that adopts exact general-purpose algorithms. As output, this software gives the number, the position and the orientation of the CCTVs to be installed. Moreover, for each point of the grid representing the asset, the software application returns the number of CCTVs covering it. If the size of the ILP model (i.e., the number of variables and constraints) is too large and it cannot solved exactly, several effective approximated approaches presented in the literature can be used in order to obtain a good sub-optimal solution.

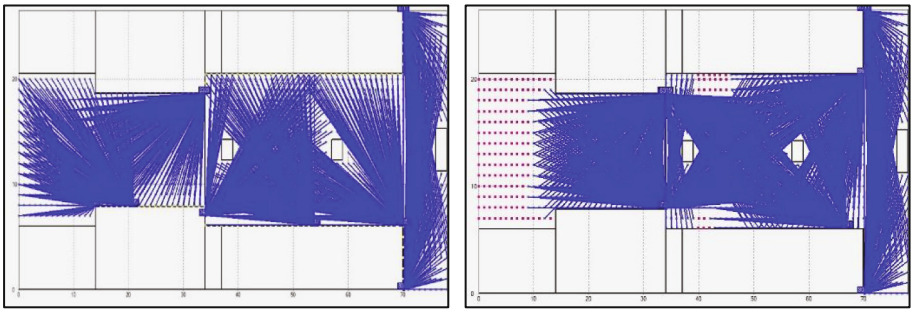
4 Experimental Results

In this section, we report the graphical representation of the results achieved with the *SC* and the *MC* models on the two sub-areas. Moreover, we compare the *SC* model solutions with the ones produced by the security experts, both in terms of number of used CCTVs and percentage of covered grid points.

The ILP covering models have been solved by the optimization software FICO™ Xpress-MP 7.9. For the simulation, we used an Intel® Core™ i7, 870, 2.93 GHz, 4 GB RAM, Windows Vista™ 64 bit.

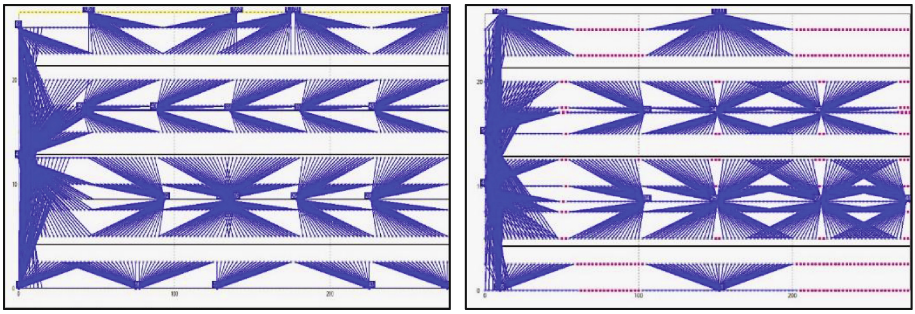
4.1 Set Covering Model

The results of the *SC* model for sub-area *S1* and *S2* are reported in Figs. 4a and 5a, while the related solution designed by the experts are represented in Figs. 4b and 5b, respectively, where the uncovered points are represented by purple dots.



(a) *(b)*

Fig. 4. Solutions for sub-area *S1*: *(a)* SCP model solution; *(b)* security expert solution. (Color figure online)



(a) *(b)*

Fig. 5. Solutions for sub-area *S2*: *(a)* SCP model solution; *(b)* security expert solution. (Color figure online)

Concerning the sub-area *S1*, from the analysis of the figures, we can note that the *SC* model returns a solution which uses 8 CCTVs and it entirely covers the area of interest. The solution designed by the experts also uses 8 CCTVs, but on the contrary it is able to cover just the 86.56% of the area. Hence, using the same number of CCTVs but with a different layout it is possible to cover all the area under investigation.

Concerning the sub-area *S2*, the *SC* model returns a solution with 23 cameras which entirely covers the area of interest. The solution of the experts uses 21 cameras with a coverage percentage equal to 83.59%. It is easy to see that the expert solution could be easily improved by locating two additional cameras at the end of the platforms. However,

also increasing the number of CCTVs, their disposition would not allow covering all the uncovered points.

4.2 Maximal Covering Model

The *MC* model using 8 and 23 CCTVs for *S1* and *S2* respectively clearly gives the same solution of the *SC* model. Hence, we experienced the *MC* model assuming that the number of available CCTVs is lower than the number provided by the *SC* model, in particular, 6 and 7 cameras for the sub-area *S1*, 21 and 22 cameras for the sub-area *S2*. The solutions provided by the *MC* model, represented in Figs. 6 and 7, show that some points remain uncovered, but, comparing them with the expert solutions reported in Figs. 4b and 5b, it is easy to observe that the number of uncovered points is significantly lower.

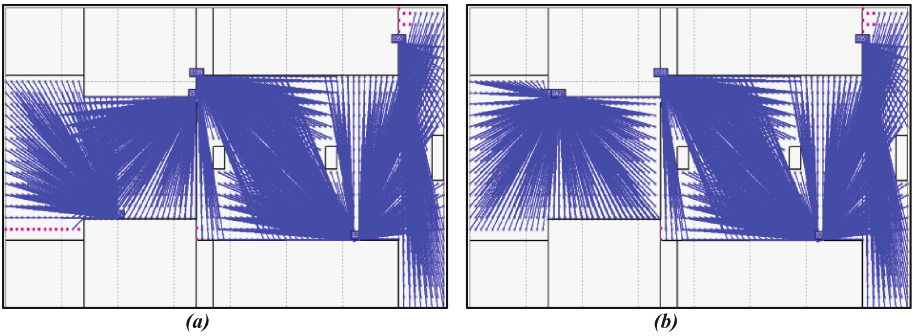


Fig. 6. *MC* model solutions for sub-area *S1*. *(a)* Solution with 6 CCTVs; *(b)* Solution with 7 CCTVs.

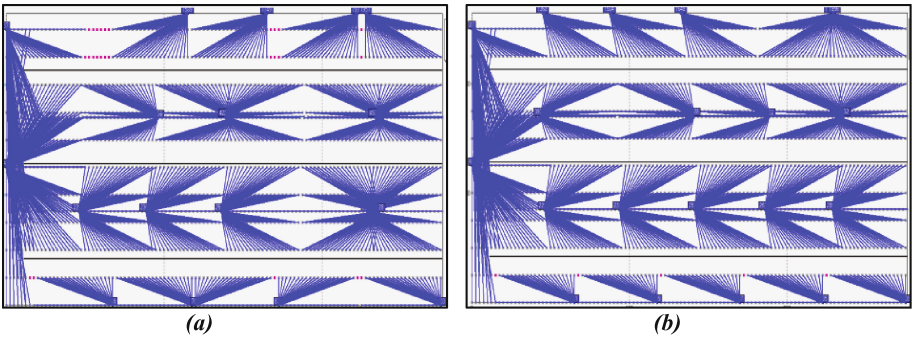


Fig. 7. *MC* model solutions for sub-area *S2*: *(a)* Solution with 21 CCTVs; *(b)* Solution with 22 CCTVs.

Specifically, concerning *S1* the percentage of uncovered points provided by the *MC* model with 6 and 7 cameras is 1.72% and 0.70%, respectively, against the 13% of the expert solution. In relation to *S2*, the percentage of uncovered points from the *MC* model with 22 and 23 cameras is 0.41% and 0.22%, respectively, while the one provided by

the expert solution is 11.26%. This confirms that, when the number of CCTVs to be used is limited, the *MC* model represents an important supporting tool for the expert in security decision making.

5 Conclusions

The obtained results show the effectiveness and the efficiency of the proposed approach in a real use case scenario. The coverage models are indeed able to optimize the number and/or the layout of the CCTVs in order to provide high coverage levels of the area under investigation. This demonstrates that, in the design of a surveillance system, the use of optimization methodologies may not only increase the asset security, but also reduce workload and security management costs.

The proposed approach presents a wide applicability since its usage is clearly not restricted to the railway stations (terminal or junction ones), but it can be adopted in different critical infrastructure assets. Moreover, it can be easily extended to other kind of sensors to be employed for control and monitoring activities of substantial regions of interest. Two main limitations of this approach have to be considered for future work perspectives. The first is mainly driven by the application of the used methodology. In particular, several important design issues and constraints have to be considered in real application domains. For instance, wiring constraints and volumetric issues should be properly taken into account, together with the possibility of integrating different kind of sensors. The second work perspective is motivated by the real problem instance solution. The volumetric issue provides indeed a significant increases in the size of the optimization problem under investigation. This pushes towards the refinement and the improvement of the used methodology in order to take into account larger size instances with an acceptable computational time.

References

1. Azim, A., Aycard, O.: Detection, classification and tracking of moving objects in a 3D environment. In: IEEE Intelligent Vehicles Symposium (IV), Alcal de Henares (2012)
2. Berman, O., Drezner, Z., Krass, D.: Generalized coverage: new developments in covering location models. Comput. Oper. Res. **37**, 1675–1687 (2010)
3. Bocchetti, C., Flammini, F., Pragliola, C., Pappalardo, A.: Dependable integrated surveillance systems for the physical security of metro railways. In: Third ACM/IEEE International Conference on IEEE Distributed Smart Cameras (ICDSC 2009), pp. 1–7 (2009)
4. Boccia, M., Sforza, A., Sterle, C.: Flow intercepting facility location: problems, models and heuristics. J. Math. Model. Algorithms **8**(1), 35–79 (2009)
5. Caprara, A., Toth, P., Fischetti, M.: Algorithms for set covering problem. Ann. Oper. Res. **98**, 353–371 (2000)
6. Chow, T., Cho, S.Y.: Industrial neural vision system for underground railway station platform surveillance. Adv. Eng. Inform. **16**(1), 73–83 (2002)
7. De Cillis, F., De Maggio, M.C., Pragliola, C., Setola, R.: Analysis of criminal and terrorist related episodes in railway infrastructure scenarios. J. Homel. Secur. Emerg. Manag. **10**(2), 447–476 (2013)

8. De Cillis, F., De Maggio, M.C., Setola, R.: Vulnerability assessment in RIS scenario through a synergic use of the CPTED methodology and the system dynamics approach. In: Setola, R., Sforza, A., Vittorini, V., Pragliola, C. (eds.) Railway Infrastructure Security, Topics in Safety, Risk, Reliability and Quality, pp. 65–89. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-04426-2_4
9. Ercan, A.O., El Gamal, A., Guibas, L.J.: Object tracking in the presence of occlusions using multiple cameras: a sensor network approach. ACM Trans. Sens. Netw. (TOSN) **9**(2), 16–52 (2013)
10. Guvensan, M.A., Gokhan, Y.: On coverage issues in directional sensor networks: a survey. Ad Hoc Netw. **9**(7), 1238–1255 (2011)
11. Joshi, U., Patel, K.: Object tracking and classification under illumination variations. Int. J. Eng. Dev. Res. (IJEDR) **4**(1), 667–670 (2016)
12. Mavrinac, A., Chen, X.: Modeling coverage in camera networks: a survey. Int. J. Comput. Vis. **101**(1), 205–226 (2013)
13. Moon, T.H., Heo, S.Y., Leem, Y.T., Nam, K.W.: An analysis on the appropriateness and effectiveness of CCTV location for crime prevention. World Acad. Sci. Eng. Tech. Int. J. Soc. Behav. Educ. Econ. Bus. Ind. Eng. **9**(3), 836–843 (2015)
14. Oh, S., Park, S., Lee, C.: A platform surveillance monitoring system using image processing for passenger safety in railway station. In: International Conference on Control, Automation and Systems, Seoul (2007)
15. Pathak, S., Aishwarya, P., Sunayana, K., Apurva, M.: Activity detection in video surveillance system. Int. J. Res. Advent Technol. 19–25 (2016). E-ISSN: 2321-9637
16. Ronetti, N., Dambra, C.: Railway station surveillance: the Italian case. In: Foresti, G.L., Mähönen, P., Regazzoni, C.S. (eds.) Multimedia Video-Based Surveillance Systems, pp. 13–20. Springer, Boston (2000). https://doi.org/10.1007/978-1-4615-4327-5_2
17. Shi, G., Chang, L., Shuyuan, Z.: Research on passenger recognition system based on video image processing in railway passenger station. In: International Conference on Information Sciences, Machinery, Materials and Energy (ICISMME), Chongqing (2015)
18. Setola, R., Sforza, A., Vittorini, V., Pragliola, C. (eds.): Railway Infrastructure Security. Topics in Safety, Risk, Reliability and Quality. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-04426-2
19. Sterle, C., Sforza, A., Esposito, A.A., Piccolo, C.: A unified solving approach for two and three dimensional coverage problems in sensor networks. Opt. Lett. **10**, 1–23 (2016)
20. Sun, J., Lo, B.P.L., Velastin, S.A.: Fusing visual and audio information in a distributed intelligent surveillance system for public transport systems. Acta Autom. Sin. **20**(3), 393–407 (2003)
21. Valera, M., Velastin, S.A.: Intelligent distributed surveillance systems: a review. IEEE Proc. Vis. Image Sig. Process. **152**(2), 192–204 (2005)
22. Xiao, Y., Farooq, A.R., Smith, M., Wright, D., Wright, G.: Robust left object detection and verification in video surveillance. In: IAPR International Conference on Machine Vision Applications, MVA 2013, Kyoto (2013)
23. Wang, B.: Coverage problems in sensor networks: a survey. ACM Comput. Surv. **43**(4), 32–43 (2011)
24. Wang, X.: Intelligent multi-camera video surveillance: a review. Pattern Recogn. Lett. **34**(1), 3–19 (2013)
25. Zhang, R., Ding, J.: Object tracking and detecting based on adaptive background subtraction. Procedia Eng. **29**, 1351–1355 (2012)

A Synthesis of Optimization Approaches for Tackling Critical Information Infrastructure Survivability

Annunziata Esposito Amideo(✉) and Maria Paola Scaparra

Kent Business School, University of Kent, Canterbury, UK
{ae306,m.p.scaparra}@kent.ac.uk

Abstract. Over the years, Critical Infrastructures (CI) have revealed themselves to be extremely disaster-prone, be the disasters nature-based or man-made. This paper focuses on a specific category of CI: Critical Information Infrastructures (CII), which are commonly deemed to include communication and information networks. The majority of all the other CI (e.g. electricity, fuel and water supply, transport systems, etc.) are crucially dependent on CII. Therefore, problems associated with CII that disrupt the services they are able to provide (whether to a single end-user or to another CI) are of increasing interest. This paper discusses some recent developments in optimization models regarding CII's ability to withstand disruptive events within three main spheres: network survivability assessment, network resource allocation strategy and survivable design.

Keywords: Critical Information Infrastructures (CII) · Survivability
Resource allocation strategy · Survivable network design

1 Introduction

Infrastructures considered critical (CI) are those physical and information-based facilities, networks and assets which if damaged would have serious impacts on the well-being of citizens, proper functioning of governments and industries or result in other adverse effects [20]. The nature of these infrastructures, along with the potential threats arising from disasters, whether nature-based or man-made, has prompted what is referred to as Critical Infrastructure Protection (CIP).

This paper focuses on a specific category of CI, namely the Critical Information Infrastructures (CII), and reviews recent developments in the optimization field aimed at addressing Critical Information Infrastructure Protection (CIIP) issues.

CII are described as those systems, belonging to the Information and Communication Technology, which are critical - not just for their own sakes, but for other CI that rely on them (e.g., transportation) [36]. Examples of CII are the public telephone network, Internet, terrestrial and satellite wireless networks and so on and so forth [25].

CIIP is defined as those plans and strategies developed by network operators, infrastructure owners and others, aimed at keeping the service level of CII above a predetermined threshold, despite the occurrence of disruptive events of various natures [35]. It is clear that CII are key elements in production and service systems. Even a local failure at the single CII level (e.g. shut down servers, interrupted cable connections, etc.)

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 75–87, 2017.
https://doi.org/10.1007/978-3-319-71368-7_7

may prompt far-reaching adverse effects on the CI relying on it. Bigger disruptions may have even more catastrophic cascading consequences. For example, the 2001 World Trade Center attacks crippled communications by destroying telephone and Internet lines, electric circuits and cellular towers ([10, 18]). This caused a cascade of disruptions at all levels, from fuel shortages, to transportation and financial services interruptions.

The disruptive events that can affect CII are primarily identified as physical attacks or cyber-attacks. This paper focuses on the former. Three main issues emerge. First, what are the most critical elements of the system that, if disrupted, would interrupt or significantly degrade the system's normal functioning? Second, how can such an interruption be prevented or mitigated by resource allocation plans, aimed either at hardening system elements or at recovering service? Third, is it possible and worthwhile to design and establish infrastructures that are intrinsically able to resist service failure when a disruptive event occurs?

The main optimization models developed to address these issues can be categorized as follows:

1. Survivability-oriented interdiction models, aimed at identifying interdiction scenarios of CII and quantifying the consequences deriving from potential losses of system critical components in terms of ability to provide service;
2. Resource allocation strategy models, aimed at optimizing the allocation of resources (i.e., budget) among the components of already existent systems in order to either protect them or to re-establish service level; and
3. Survivable design models, aimed at planning new CII which are able to meet survivability criteria when disruptive events occur.

In this paper, we provide a description of the seminal models in each category and suggest how these models can be taken as the starting point for further development in the CIIP field.

The remainder of this paper is organized as follows. Survivability-oriented interdiction, resource allocation strategy and survivable design models are described in Sects. 2, 3 and 4, respectively. In Sect. 5, further research suggestions in modeling CIIP problems are discussed. Section 6 offers concluding remarks.

2 Identifying Critical Network Components: Survivability-Oriented Interdiction Models

The identification of critical components in network-based systems can be traced back to a few decades ago in the context of transportation infrastructures for military purposes [38]. More recently, [4] introduced optimization models for identifying critical facilities in service and supply systems.

Interdiction models, as referred to in the literature, identify network components which are the most critical, i.e. the ones that, if disrupted, inflict the most serious damage to the system. The importance of these kinds of models is easily understandable: they not only shed light on a system's major vulnerabilities, but also help form the basis for developing protection and/or recovery plans.

Interdiction models are driven by specified criteria (also called impact metrics). When dealing with CII, such as communication and information networks, the two important criteria are network reliability and network survivability. In [31], network reliability is defined as the probability measure that a network functions according to a predefined specification; whereas, network survivability is defined as the ability of a network to maintain its communication capabilities in the face of equipment failure. Moreover, according to [31], it is possible to subdivide network survivability into two categories: physical survivability and logical survivability. A network is physically survivable if after the physical failure of some nodes or arcs, a path connecting all the nodes still exists. Logical survivability is about survivability at higher levels of the OSI model and assumes that the underlying physical network is survivable.

Our interest is in evaluating how disruptive events impact a network's physical survivability by identifying its critical components, which can be nodes and/or arcs. In the case of communication and information networks, nodes can be switches, multiplexers, cross-connects, routers; arcs represent connections among them [31, 32].

Murray [18] identifies four metrics to evaluate network physical survivability: maximal flow ([38]), shortest path ([7]), connectivity ([14, 31]), and system flow ([17, 19]). Here we provide an example of an optimization model designed to ascertain the survivability of system flow. This model is a variation of the model introduced in [19] and later extended and streamlined in [17]. It identifies the r most vital components of a network, i.e. those components which, if disrupted, maximize the amount of flow that can no longer be routed over the network. In the specific case of CII, the flow represents data and information. In the following, we will refer to this model as the Survivability Interdiction model (SIM).

Given a network $G(N, A)$, where N is the set of nodes and A is the set of arcs, let Ω be the set of origin nodes, indexed by o; H the set of elements (nodes/arcs) that can be disrupted, indexed by h; Δ the set of destination nodes, indexed by d, P the set of paths, indexed by p; N_{od} the set of paths enabling flow between an origin-destination pair o-d; Φ_p the set of components belonging to path p; f_{od} the flow routed between an o-d pair; and r the number of components to be disabled. The decision variables are: S_h equal to 1 if component h is disrupted, 0 otherwise; and X_{od} equal to 1 if flow cannot be routed between a pair o-d, 0 otherwise. The mathematical formulation is:

$$max\, z = \sum_{o \in \Omega} \sum_{d \in \Delta} f_{od} X_{od} \tag{1}$$

$$s.t.$$

$$\sum_{h \in \phi_p} S_h \geq X_{od} \quad \forall o \in \Omega, d \in \Delta, p \in N_{od} \tag{2}$$

$$\sum_{h \in H} S_h = r \tag{3}$$

$$S_h \in \{0, 1\} \quad \forall h \in H \tag{4}$$

$$X_{od} \in \{0, 1\} \quad \forall o \in \Omega, d \in \Delta \tag{5}$$

The objective function (1) maximizes the total flow disrupted (or interdicted). Constraints (2) state that the flow between an o-d pair can be considered lost $(X_{od} = 1)$, only if every path connecting nodes o and d is affected by the disruption (i.e., at least one of its arc is disrupted). Constraint (3) is a typical cardinality constraint which stipulates that exactly *r* arcs are to be disrupted. Finally, constraints (4) and (5) represent the binary restrictions on the interdiction and flow variables, respectively.

The original SIM in [19] only considers arc disruption. It was later modified to address node disruption in [17]. This work also presents a variant of SIM which identifies lower bounds to the flow loss caused by the disruption of *r* nodes, thus allowing the assessment of both best-case and worst-case scenario losses. This kind of analysis is useful to build the so-called reliability envelope, a diagram originally developed in [22] to depict possible outcomes for the failure of communication systems. SIM was applied to the Abilene network, an Internet-2 backbone with 11 routers and 14 linkages connecting US institutions. The analysis shows that the worst-case interdiction of one node (Washington, D.C.) can cause a data flow decrease of over 37%; a two-node interdiction scenario (Washington, D.C. and Indianapolis) a decrease of over 73%.

One arguable aspect of existing interdiction models such as SIM is that the number of components to be disrupted is fixed to a specific and known value *r*. This assumption is made to capture the possible extents of disruptive events: large values of *r* mimic large disruptions involving the simultaneous loss of several components, while small values are used to model minor disruptions [15]. In practice, it is difficult to anticipate the extent of a disruption and therefore select a suitable *r* value. In addition, the critical components identified for a small *r* value are not necessarily a subset of the critical components identified for larger values. Consequently, these models are usually run for several values of *r* so as to identify the most vital components across disruption scenarios of different magnitude [17].

Another aspect worth mentioning is that the use of cardinality constraints like (3) is useful for identifying worst-case scenario losses caused by natural disasters. However, in case of malicious attacks, models must capture the fact that different amount and type of resources (e.g., human, financial etc.) may be needed in a concerted attack to fully disable network components and cause maximum damage [28]. From an attacker's perspective, in fact, resources may vary significantly according to the target. This is particularly true within the context of physical survivability as opposed to logical survivability. For example, a physical attack on a relatively small number of major switching centers for long-distance telecommunications may require considerably more resources than launching a logic denial-of-service attack on the Internet. However, the former type of attack may cause much longer lasting damage [13].

This aspect can be captured by either replacing (3) with a budget constraint (see [1, 15] in the context of distribution systems) or by developing models that directly minimize the attacker expenditure to achieve a given level of disruption. Examples of the latter can be found in [14]. This work presents some mixed integer programming models which minimize the cost incurred by an attacker to disconnect the network according to different survivability metrics (e.g., degree of disconnectivity). These attacker models are then used to assess the robustness of two protection resource allocation strategies: a uniform allocation (the defense budget is distributed equally among the nodes) and a

degree-based allocation (the budget is distributed among the nodes proportionally to their degree of connectivity). As it will be discussed in the next section, this approach, where protection decisions are not tackled explicitly within a mathematical model but are only assessed and/or developed on the basis of the results of an interdiction model, often leads to a suboptimal allocation of protective resources.

Another aspect that interdiction models must capture is the fact that the outcome of an attack is highly uncertain. When dealing with malicious disruptions, this is a crucial issue as attackers, such as terrorists or hackers, aim at allocating their offensive resources so as to maximize their probability of success. Clearly, there is a correlation between the amount of offensive resources invested and the probability of success of an attack: the more the former, the higher the latter. Church and Scaparra [5] introduce an interdiction model for distribution systems where an interdiction is successful with a given probability and the objective is to maximize the expected disruption of an attack on r facilities. Losada et al. [15] further extend this model by assuming that the probability of success of an interdiction attempt is dependent on the magnitude/intensity of the disruption. Similar extensions could be developed for SIM to assess the survivability of physical networks to attacks with uncertain outcomes.

3 Enhancing Critical Network Survivability: Resource Allocation Strategy Models

Optimization approaches can be used to improve CII survivability by optimizing investments in protection measures and in service recovery plans.

CII protection measures may be divided into three different categories: technical (e.g. security administration), management (e.g. security awareness, technical training) and operational (e.g. physical security) (see [37]). Our interest lies in the last category. Examples of physical security measures include: alarms, motion detectors, biometric scanners, badge swipes, access codes, and human and electronic surveillance, e.g. Perimeter Intruder Detection Systems (PIDS) and Closed Circuit Television (CCTV) [20]. In a broader sense, protection strategies may include increasing redundancy and diversity [34]. Redundancy consists in creating one or more copies of the same network element/content and is key to tackle random uncorrelated failures. Diversity aims at avoiding components of a system to undergo the same kind of failure and is used to tackle correlated failures.

Service recovery is intimately connected with the concept of survivability since it involves bringing the infrastructure to the level of service it was able to provide before a disruption and, normally, as timely as possible. In this perspective, optimization approaches provide a useful tool to identify the optimal trade-off between the level of service to restore and the amount of resources to invest over a certain time horizon.

3.1 Optimization Models for Protecting CII Physical Components

Although interdiction models like SIM are instrumental for the identification of the most critical CII components, protection resource allocation approaches which solely rely on

this information to prioritize protection investments often result in suboptimal defensive strategies ([2, 6]). This is due to the fact that when a component (e.g., the most critical) is protected, the criticality of the other components may change. Protections and interdictions decisions must therefore be addressed in an integrated way. This is typically done by using bi-level optimization programs [8]. These programs are hierarchical optimization models which emulate the game between two players, referred to as leader and follower. In the CIIP context, the leader is the network operator or infrastructure owner, who decides which system components to protect; the follower represents a saboteur (hacker or terrorist) who tries to inflict maximum damage to the system by disabling some of its components. The defender decisions are modeled in the upper level program, whereas the inner-level program models the attacker decisions and, therefore, computes worst-case scenario losses in response to the protection strategy identified in the upper level.

Below we present a bi-level program for CIIP, which embeds SIM in the inner-level. We refer to it as the Survivability Protection Problem (SPP). In addition to the parameters and variables defined in Sect. 2, SPP uses the following notation: B is the total budget available for protection; c_h is the unit cost for protecting component h; Z_h is a decision variable equal to 1 if component h is protected, 0 otherwise.

SPP can be formulated as follows:

$$min\, H(z) \tag{6}$$

$$s.t. \quad \sum_{h\in H} c_h Z_h \leq B \tag{7}$$

$$Z_h \in \{0,1\} \quad \forall h \in H \tag{8}$$

$$H(z) = max \sum_{o\in\Omega} \sum_{d\in\Delta} f_{od} X_{od} \tag{9}$$

$$s.t. \quad (2)\text{ - }(5)$$

$$S_h \leq 1 - Z_h \tag{10}$$

The upper level model identifies which network components to protect given limited budgetary resources (7) so as to minimize a function, $H(z)$, which represents the highest flow loss (6) resulting from the interdiction of r components. The inner-level model is the SIM with the additional set of constraints (10) which guarantee that if a component is protected, it cannot be attacked.

Protection models like SPP can be extended in a number of ways. For example, protection investments over time could be considered, given that funds for enhancing CI security usually become available at different times. An example of bi-level protection models that considers dynamic investments can be found in [33] within the context of transportation infrastructure. Probabilistic extensions of SPP should also be considered, where the protection of an element does not completely prevent its interdiction, but may reduce its probability of failure. Other issues that should be captured are the

uncertainty in the number of simultaneous losses of components (see for example [11]), and the correlation among components failures [12].

Obviously, there are other approaches other than bi-level programming which can be used to optimize protection strategies. For example, Viduto et al. [37] combine a risk assessment procedure for the identification of system risks with a multi-objective optimization model for the selection of protection countermeasures. To mitigate cyber-threats, Sawik [27] uses mixed integer models in conjunction with a conditional value-at-risk approach to identify optimal protection countermeasure portfolios under different risk preferences of the decision maker (risk-adverse vs. risk neutral).

3.2 Optimization Models for CII Service Restoration

An interesting model for the optimization of recovery investments is the Networked Infrastructure Restoration Model (NIRM) introduced in [16]. NIRM is a multi-objective optimization model for the evaluation of tradeoffs between flow restoration and system costs over time.

NIRM uses the following additional notation: Γ^n is the set of inoperable nodes, Γ^l the set of inoperable arcs, Φ_p^n the set of disrupted nodes along path p, Φ_p^l the set of disrupted arcs along path p, T the set of planning periods; f_{od} is the flow routed between the pair o-d; c_{pt} the cost of traversing path p during planning period t; λ_i and λ_j the costs of restoring operation at node i and arc j, respectively; H_t^n and H_t^l the budget for node and arc restoration during planning period t; β_t the weight for importance of repair in time t; C_{odt} is a large quantity representing the cost of a disrupted pair o-d during planning period t. The decision variables are: Y_{pt}, equal to 1 if path p is available in time t, 0 otherwise; V_{it}^n (V_{jt}^l), equal to 1 if node i (arc j) is restored in time t, 0 otherwise; and W_{odt}, equal to 1 if connectivity does not exist between a pair o-d in time t, 0 otherwise. The formulation is the following:

$$max \sum_{o\in\Omega}\sum_{d\in\Delta}\sum_{p\in N_{od}}\sum_{t\in T}\beta_t f_{od} Y_{pt} \tag{11}$$

$$min \sum_{o\in\Omega}\sum_{d\in\Delta}\sum_{t\in T} C_{odt} W_{odt} + \sum_{o\in\Omega}\sum_{d\in\Delta}\sum_{p\in N_{od}}\sum_{t\in T} c_{pt} Y_{pt} \tag{12}$$

$$s.t.$$

$$\sum_{i\in\Gamma^n} \lambda_i V_{it}^n \le H_t^n \quad \forall t \in T \tag{13}$$

$$\sum_{j\in\Gamma^l} \lambda_j V_{jt}^l \le H_t^l \quad \forall t \in T \tag{14}$$

$$\sum_{t\in T} V_{it}^n \le 1 \quad \forall i \in \Gamma^n \tag{15}$$

$$\sum_{t\in T} V_{jt}^l \le 1 \quad \forall j \in \Gamma^l \tag{16}$$

$$Y_{pt} - \sum_{\hat{t}\le t} V_{i\hat{t}}^n \le 0 \quad \forall p \in P, i \in \Phi_p^n, t \in T \tag{17}$$

$$Y_{pt} - \sum_{\hat{\imath} \le t} V^l_{j\hat{\imath}} \le 0 \quad \forall p \in P, j \in \Phi^l_p, t \in T \tag{18}$$

$$\sum_{p \in N_{od}} Y_{pt} + W_{odt} = 1 \quad \forall o \in \Omega, d \in \Delta, t \in T \tag{19}$$

$$Y_{pt} \in \{0,1\} \quad \forall p \in P, t \in T \tag{20}$$

$$V^n_{it} \in \{0,1\} \quad \forall i \in \Gamma^n, t \in T \tag{21}$$

$$V^l_{jt} \in \{0,1\} \quad \forall j \in \Gamma^l, t \in T \tag{22}$$

$$W_{odt} = \{0,1\} \quad \forall o \in \Omega, d \in \Delta, t \in T \tag{23}$$

The objective function (11) maximizes system flow or connectivity while objective (12) minimizes system cost and it is made up of two components, disruption and path usage. Constraints (13) and (14) are budget constraints on node and arc recovery in each planning period t. Constraints (15) and (16) restrict node and arc repair to a single time period. Constraints (17) and (18) state that a path p is available in period t only if each of its disrupted component (node i or arc j respectively) is repaired in period t or in any of the preceding time periods. Constraints (19) track o-d pairs that are not connected in each time period and force the selection of at most one path between each o-d pair in each time period. Finally, constraints (20)–(23) represent the binary restrictions on the decision variables.

NIRM was applied to support recovery planning after a simulated High Altitude Electromagnetic Pulse attack on a sample telecommunications backbone network with 46 routers and 94 high-capacity backbones. Different restoration schedules over 6 repair periods were generated and analyzed so as to highlight the tradeoffs between flow restoration and system costs.

A limitation of NIRM is that the repair action is assumed to be instantaneous. Nurre et al. [21] consider the duration for component repair in an integrated restoration planning optimization model which identifies the network components to be installed/repaired after a disruption and schedules them to available work groups. The objective is to maximize the cumulative amount of flow that can be routed across the network over a finite planning horizon. An interesting addition to the restoration modeling literature is the model in [29] which considers the important issue of restoring multiple interdependent infrastructure systems (e.g., power, telecommunication, water). This work also presents tools to quantify the improvement in restoration effectiveness resulting from information sharing and coordination among infrastructures.

4 Planning Survivable Networks: Design Models

Given the crucial importance of CII to the vast majority of economic activities and services, telecommunication and information systems are designed in such a way that they are intrinsically survivable, i.e. they satisfy some more or less stringent connectivity criteria. The design of survivable network is a well-studied problem in the optimization

field. For an early survey, the interested reader can refer to [31]. A comprehensive review of survivable network design models would be outside the scope of this paper. To provide a complete treatment of survivability related optimization problems, we only briefly discuss the Survivable Network Design (SND) model found in [32], one of the earliest and most studied models.

Given an undirected graph $G(N, E)$, where N is the set of nodes and E is the set of undirected edges (i, j), each pair of communicating nodes is identified as a commodity k (being K the set of the commodities), whose origin and destination are labeled as $O(k)$ and $D(k)$ respectively. Let c_{ij} be the design cost of edge (i, j), and q the number of node disjoint paths required for all the commodities (so the system will be able to face $q - 1$ failures at most). The decision variables are: U_{ij} equal to 1 if edge (i, j) is included in the design, 0 otherwise; and X_{ij}^k equal to 1 if commodity k uses edge (i, j), 0 otherwise. The formulation is the following:

$$min\, z = \sum_{(i,j)\in E} c_{ij} U_{ij} \tag{24}$$

s.t.

$$\sum_{j\in N} X_{ij}^k - \sum_{j\in N} X_{ji}^k = \begin{cases} Q & if\ i \equiv O(k) \\ -Q & if\ i \equiv D(k) \\ o & \text{otherwise} \end{cases} \quad \forall k \in K \tag{25}$$

$$X_{ij}^k \leq U_{ij} \quad \forall k \in K, (i,j) \in E \tag{26}$$

$$X_{ji}^k \leq U_{ij} \quad \forall k \in K, (i,j) \in E \tag{27}$$

$$\sum_{i\in N} X_{ij}^k \leq 1 \quad \forall k \in K, j \in N \wedge j \neq D(k) \tag{28}$$

$$X_{ij}^k, X_{ji}^k = \{0, 1\} \quad \forall k \in K, i, j \in N \tag{29}$$

$$U_{ij} = \{0, 1\} \quad \forall i, j \in N \tag{30}$$

The objective function (24) minimizes the cost of the topological network design. Constraints (25) guarantee network flow conservation. Constraints (26) and (27) stipulate that flow can traverse an edge only if the edge is included in the design. The combined use of constraints (25), (26) and (27) enforce the edge-disjoint paths over the network. Constraints (28) guarantee that at most one unit of flow can traverse a node that is neither a commodity origin nor destination, thus ensuring the correct number of node-disjoint paths in the network. Finally, constraints (29) and (30) represent the binary restrictions on the variables.

Many other survivable network design models can be found in the literature which differ in terms of underlying network (wired vs. wireless), network topology (e.g., ring,

mesh, star, line, tree, etc.), connectivity requirements (e.g., edge and/or vertex-connectivity), path-length restrictions (e.g., hop limits [24]), cost minimization [23], and dedicated settings (e.g., path protection, link and path restoration [24]).

Note that recent survivability design models embed interdiction models to ascertain components criticality ([3, 30]). Such models are able to identify cost-effective CII configurations which are inherently survivable without the need to specify the number of disjoint paths required between each pair of communicating nodes, like in SND.

5 Future Research Suggestions

The research on CIIP issues aimed at hedging against potential physical attacks is still evolving. The demand for such work has been prompted by disasters of diverse nature, with 9/11 being a seminal one.

The survivability optimization models discussed in this paper are basic models that can be extended in a number of ways. For example, interdiction and protection models could be extended to tackle both physical and logical survivability issues by incorporating routing and link capacity assignment decisions. In addition, most of the optimization models developed so far are deterministic. However, failures and disruptions are random events, often difficult to predict. The probabilistic behaviour of complex CII under disruptions would be better modelled by using stochastic models, including uncertain parameters (e.g., uncertainty on arc/node availability, extent of a disruption, stochastic repair times, etc.). Alternatively, the uncertainty characterizing disruptions could be captured in scenario-based models which incorporate robustness measures for the identification of solutions which perform well across different disruption scenarios [26].

Future models could even combine the optimization of protection and restoration strategies in a unified framework so as to distribute resources efficiently across the different stages of the disaster management cycle (protection plans belong to the pre-disaster stage while recovery plans refer to the post-disaster stage). Other resource allocation models could consider identifying tradeoff investments in physical protection and cyber-security to mitigate the impact of both physical and logical attacks. Models which address design and restoration issues conjunctively, such as the one in [23], also deserve further investigation.

The models discussed in this paper have been solved by using a variety of optimization algorithms, including exact methods (e.g., decomposition) and heuristics (e.g. evolutionary algorithms). The development of more complex models, such as stochastic, bi-level and multi-objective models, would necessarily require additional research into the development of more sophisticated solution techniques, possibly integrating exact and heuristic methodologies.

Eventually, the ultimate challenge when developing optimization approaches for increasing CII survivability is to consider the interdependency among multiple CI and the potential cascading failures across different lifeline systems. As noted in [29], information sharing and coordination among infrastructures significantly improve the effectiveness of survivability strategies, as opposed to decentralized decision making.

However, existing models that address network interdependencies are either overly simplistic or too theoretical [9]. This area certainly warrants further research.

6 Conclusions

This paper reviewed the research activities conducted over recent years in the field of CIIP aimed at mitigating the effects of physical attacks against CII components. This paper has investigated three main research areas: survivability assessment models, resource allocation strategy models (aimed at either protection or recovery plans), and survivable design models.

Each model category has been designed to identify different crucial aspects: (a) under what circumstances is the infrastructure still able to provide its service; (b) how should resources be allocated in order to protect the infrastructure physical components or to restore its level of service; (c) how should a new infrastructure be designed in order to be naturally survivable.

The optimization models hereby discussed are valuable decision-making tools in tackling CII survivability issues but future work is undoubtedly needed. The reason lies in the intrinsic nature of CII: they are large-scale, heterogeneous, distributed systems whose complexity is continuously evolving in a risky environment. As such, modeling their dynamics and interdependency with other lifeline systems requires developing cutting-edge methodologies, which integrate methods from different disciplines (e.g., optimization, simulation, risk analysis, complex network theory and statistics) in a unified framework.

References

1. Aksen, D., Piyade, N., Aras, N.: The budget constrained r-interdiction median problem with capacity expansion. Cent. Eur. J. Oper. Res. **18**(3), 269–291 (2010)
2. Cappanera, P., Scaparra, M.P.: Optimal allocation of protective resources in shortest-path networks. Transp. Sci. **45**(1), 64–80 (2011)
3. Chen, R.L., Cohn, A., Pinar, A.: An implicit optimization approach for survivable network design. In: Network Science Workshop (NSW), pp. 180–187. IEEE (2011)
4. Church, R.L., Scaparra, M.P., Middleton, R.S.: Identifying critical infrastructure: the median and covering facility interdiction problems. Ann. Assoc. Am. Geogr. **94**(3), 491–502 (2004)
5. Church, R.L., Scaparra, M.P.: Analysis of facility systems' reliability when subject to attack or a natural disaster. In: Murray, A.T., Grubesic, T.H. (eds.) Critical Infrastructure. ADVSPATIAL, pp. 221–241. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-68056-7_11
6. Church, R.L., Scaparra, M.P.: Protecting critical assets: the r-interdiction median problem with fortification. Geogr. Anal. **39**(2), 129–146 (2007)
7. Corley, H., David, Y.S.: Most vital links and nodes in weighted networks. Oper. Res. Lett. **1**(4), 157–160 (1982)
8. Dempe, S.: Foundations of Bilevel Programming. Springer Science & Business Media, Heidelberg (2002)
9. Fang, Y.: Critical infrastructure protection by advanced modelling, simulation and optimization for cascading failure mitigation and resilience. Dissertation. Ecole Centrale Paris (2015)

10. Grubesic, T.H., O'Kelly, M.E., Murray, A.T.: A geographic perspective on commercial internet survivability. Telematics Inform. **20**(1), 51–69 (2003)
11. Liberatore, F., Scaparra, M.P., Daskin, M.S.: Analysis of facility protection strategies against an uncertain number of attacks: the stochastic r-interdiction median problem with fortification. Comput. Oper. Res. **38**(1), 357–366 (2011)
12. Liberatore, F., Scaparra, M.P., Daskin, M.S.: Hedging against disruptions with ripple effects in location analysis. Omega **40**(1), 21–30 (2012)
13. Lin, H.S., Patterson, D.A., Hennessy, J.L.: Information Technology for Counterterrorism: Immediate Actions and Future Possibilities. National Academies Press, Washington, D.C. (2003)
14. Lin, F.Y.-S., Yen, H.-H., Chen, P.-Y., Wen, Y.-F.: Evaluation of network survivability considering degree of disconnectivity. In: Corchado, E., Kurzyński, M., Woźniak, M. (eds.) HAIS 2011. LNCS, vol. 6678, pp. 51–58. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21219-2_8
15. Losada, C., Scaparra, M.P., Church, R.L., Daskin, M.S.: The stochastic interdiction median problem with disruption intensity levels. Ann. Oper. Res. **201**(1), 345–365 (2012)
16. Matisziw, T.C., Murray, A.T., Grubesic, T.H.: Strategic network restoration. Netw. Spat. Econ. **10**(3), 345–361 (2010)
17. Murray, A.T., Matisziw, T.C., Grubesic, T.H.: Critical network infrastructure analysis: interdiction and system flow. J. Geogr. Syst. **9**(2), 103–117 (2007)
18. Murray, A.T.: An overview of network vulnerability modeling approaches. GeoJournal **78**(2), 209–221 (2013)
19. Myung, Y., Kim, H.: A cutting plane algorithm for computing k-edge survivability of a network. Eur. J. Oper. Res. **156**(3), 579–589 (2004)
20. Nickolov, E.: Critical information infrastructure protection: analysis, evaluation and expectations. Inf. Secur. **17**, 105–119 (2006)
21. Nurre, S.G., Cavdaroglu, B., Mitchell, J.E., Sharkey, T.C., Wallace, W.A.: Restoring infrastructure systems: an integrated network design and scheduling (INDS) problem. Eur. J. Oper. Res. **223**(3), 794–806 (2012)
22. O'Kelly, M.E., Kim, H.: Survivability of commercial backbones with peering: a case study of Korean networks. In: Murray, A.T., Grubesic, T.H. (eds.) Critical Infrastructure. ADVSPATIAL, pp. 107–128. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-68056-7_6
23. Orlowski, S., Wessäly, R.: Comparing restoration concepts using optimal network configurations with integrated hardware and routing decisions. J. Netw. Syst. Manag. **13**(1), 99–118 (2005)
24. Orlowski, S., Wessäly, R.: The effect of hop limits on optimal cost in survivable network design. In: Raghavan, S., Anandalingam, G. (eds.) Telecommunications Planning: Innovations in Pricing, Network Design and Management. ORCS, vol. 33, pp. 151–166. Springer, Boston (2006). https://doi.org/10.1007/0-387-29234-9_8
25. Patterson, C.A., Personick, S.D.: Critical Information Infrastructure Protection and the Law: An Overview of Key Issues. National Academies Press, Washington, D.C. (2003)
26. Peng, P., Snyder, L.V., Lim, A., Liu, Z.: Reliable logistics networks design with facility disruptions. Transp. Res. B Methodol. **45**(8), 1190–1211 (2011)
27. Sawik, T.: Selection of optimal countermeasure portfolio in IT security planning. Decis. Support Syst. **55**(1), 156–164 (2013)
28. Scaparra, M.P., Church, R.L.: Location problems under disaster events. In: Laporte, G., Nickel, S., da Gama, F.S. (eds.) Location Science, pp. 623–642. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-13111-5_24

29. Sharkey, T.C., Cavdaroglu, B., Nguyen, H., Holman, J., Mitchell, J.E., Wallace, W.A.: Interdependent network restoration: on the value of information-sharing. Eur. J. Oper. Res. **244**(1), 309–321 (2015)
30. Smith, J.C., Lim, C., Sudargho, F.: Survivable network design under optimal and heuristic interdiction scenarios. J. Global Optim. **38**(2), 181–199 (2007)
31. Soni, S., Gupta, R., Pirkul, H.: Survivable network design: the state of the art. Inf. Syst. Front. **1**(3), 303–315 (1999)
32. Soni, S., Pirkul, H.: Design of survivable networks with connectivity requirements. Telecommun. Syst. **20**(1–2), 133–149 (2002)
33. Starita, S., Scaparra, M.P.: Optimizing dynamic investment decisions for railway systems protection. Eur. J. Oper. Res. **248**(2), 543–557 (2016)
34. Sterbenz, J.P., Hutchison, D., Çetinkaya, E.K., Jabbar, A., Rohrer, J.P., Schöller, M., Smith, P.: Resilience and survivability in communication networks: strategies, principles, and survey of disciplines. Comput. Netw. **54**(8), 1245–1265 (2010)
35. Brunner, E., Suter, M.: International CIIP Handbook 2008/2009: An Inventory of 25 National and 7 International Critical Infrastructure Protection Policies. Center for Security Studies, ETH Zurich (2008)
36. Theron, P.: Critical Information Infrastructure Protection and Resilience in the ICT Sector. IGI Global, Hershey (2013)
37. Viduto, V., Maple, C., Huang, W., López-Peréz, D.: A novel risk assessment and optimisation model for a multi-objective network security countermeasure selection problem. Decis. Support Syst. **53**(3), 599–610 (2012)
38. Wollmer, R.: Removing arcs from a network. Oper. Res. **12**(6), 934–940 (1964)

A Dataset to Support Research in the Design of Secure Water Treatment Systems

Jonathan Goh(✉), Sridhar Adepu, Khurum Nazir Junejo, and Aditya Mathur

iTrust, Center for Research in Cyber Security,
Singapore University of Technology and Design, Singapore, Singapore
jonathan_goh@sutd.edu.sg

Abstract. This paper presents a dataset to support research in the design of secure Cyber Physical Systems (CPS). The data collection process was implemented on a six-stage Secure Water Treatment (SWaT) testbed. SWaT represents a scaled down version of a real-world industrial water treatment plant producing 5 gallons per minute of water filtered via membrane based ultrafiltration and reverse osmosis units. This plant allowed data collection under two behavioral modes: normal and attacked. SWaT was run non-stop from its "empty" state to fully operational state for a total of 11-days. During this period, the first seven days the system operated normally i.e. without any attacks or faults. During the remaining days certain cyber and physical attacks were launched on SWaT while data collection continued. The dataset reported here contains the physical properties related to the plant and the water treatment process, as well as network traffic in the testbed. The data of both physical properties and network traffic contains attacks that were created and generated by our research team.

Keywords: Cyber Physical Systems · Datasets · Network traffic
Physical properties

1 Introduction

Cyber Physical Systems(CPSs) are built by integrating computational algorithms and physical components for various mission-critical tasks. Examples of such systems include public infrastructures such as smart power grids, water treatment and distribution networks, transportation, robotics and autonomous vehicles. These systems are typically large and geographically dispersed, hence they are being network connected for remote monitoring and control. However, such network connectivities open up the likelihood of cyber attacks. Such possibilities make it necessary to develop techniques to defend CPSs against attacks: cyber or physical. A "cyber attack" refers to an attack that is transmitted through a communications network to affect system behavior with an intention to cause some economic harm. A "physical attack" is on a physical component such as a motor or a pump to disrupt state of the system.

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 88–99, 2017.
https://doi.org/10.1007/978-3-319-71368-7_8

Research efforts in securing CPSs from such attacks have been ongoing. However, there is limited availability of operational data sets in this research community to advance the field of securing CPSs. While there are datasets for Intrusion Detection Systems (IDS), these datasets focus primarily on network traffic. Such datasets include, for example, the DARPA Intrusion Detection Evaluation Dataset [3] and the NSL-KDD99 [2] datasets. These data are a collection of RAW TCP dump collected over a period of time which includes various intrusions simulated in a military network environment. Such datasets are thus not suitable for CPS IDS. The only other publicly available datasets for CPS kown to the authors are provided by the Critical Infrastructure Protector Center at the Mississippi State University (MSU) [4]. Their datasets [4] comprise of data obtained from their Power, Gas and Water testbeds. The power dataset is based on a simulated smart grid whereas their water and gas datasets were obtained from a very small scale laboratory testbed. However, as acknowledged by the authors themselves, these datasets have been found to contain some unintended patterns that can be used to easily identify attacks versus non-attacks using machine learning algorithms. Although the gas dataset was updated in 2015 [4] to provide more randomness, it was obtained from a small scale testbed which may not reflect the true complexity of CPSs. Hence, there is no publicly available realistic dataset of a sufficient complexity from a modern CPS that contains both network traffic data and physical properties of the CPS.

The goal of this paper is to provide a realistic dataset that can be utilised to design and evaluate CPS defence mechanisms. In this paper, we present a dataset obtained from Secure Water Treatment testbed (SWaT).

The main objective of creating this dataset and making it available to the research community is to enable researchers to (1) design and evaluate novel defence mechanisms for CPSs, (2) test mathematical models, and (3) evaluate the performance of formal models of CPS. The key contributions of the paper are as follows:

1. A large scale labelled–normal & attack–dataset collected from a realistic testbed of sufficient complexity.
2. Network traffic and physical properties data.

The remainder of this paper is organised as follows. Section 2 describes the SWaT testbed in which the data collection process was implemented. Section 3 presents the attacks used in this data collection procedure. Section 4 describes the entire data collection process including the types of data collected. The paper concludes in Sect. 5.

2 Secure Water Treatment (SWaT)

As illustrated in Fig. 1, SwaT is a fully operational scaled down water treatment plant with a small footprint, producing 5 gallons/minute of doubly filtered water. This testbed replicates large modern plants for water treatment such as those

Fig. 1. Actual photograph of SWaT testbed

found in cities. Its main purpose is to enable experimentally validated research in the design of secure and safe CPS. SWaT has six main processes corresponding to the physical and control components of the water treatment facility. It has the following six-stage filtration process, as shown in Fig. 2.

2.1 Water Treatment Process

The process (P1) begins by taking in raw water and storing it in a tank. It is then passed through the pre-treatment process (P2). In this process, the quality of the water is assessed. Chemical dosing is performed if the water quality is not within acceptable limits. The water than reaches P3 where undesirable materials are removed using fine filtration membranes. After the residuals are filtered through the Ultra Filtration system, any remaining chorine is destroyed in the Dechlorination process (P4) using Ultraviolet lamps. Subsequently, the water from P4 is pumped into the Reverse Osmosis (RO) system (P5) to reduce inorganic impurities. In the last process, P6, water from the RO is stored and ready for distribution in a water distribution system. In the case of SWaT, the treated water can be transferred back to the raw tank for re-processing. However, for the purpose of data collection, the water from P6 is disposed to mimic water distribution.

2.2 Communications

SWaT consists of a layered communication network, Programmable Logic Controllers (PLCs), Human Machine Interfaces (HMIs), a Supervisory Control and Data Acquisition (SCADA) workstation, and a Historian. Data from the sensors is available to the SCADA system and recorded by the Historian for subsequent analysis.

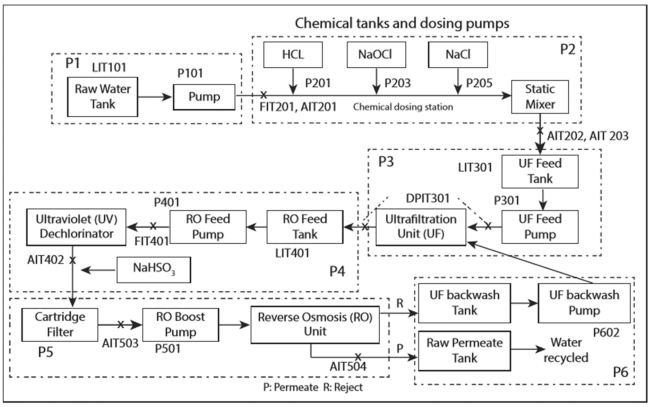


Fig. 2. SWaT testbed processes overview

As illustrated in Fig. 3, there are two networks in SWaT. Level 1 is a star network that allows the SCADA system to communicate with the six PLCs dedicated to each of the process. Level 0 is a ring network that transmits sensor and actuator data to the relevant PLC. The sensors, actuators and PLCs all communicate either via wired or wireless links (where manual switches allow the switch between wireless and wired modes).

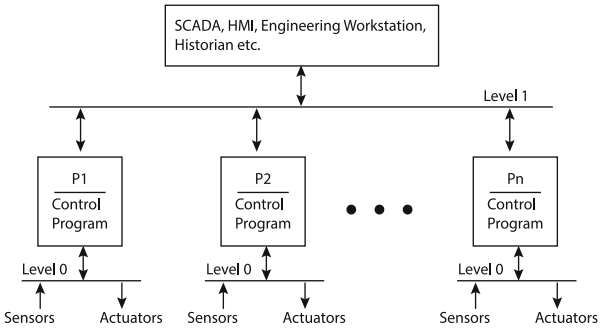


Fig. 3. SWaT testbed processes overview

In the data collection process, only network data through wired communications was collected.

3 Attack Scenarios

A systematic approach was used to attack the system. We used the attack model [1] that considers the intent space of an attacker for a given CPS in the attack

model. This attack model can be used to generate attack procedures and functions that target a specific CPS. In our case, the attack model to target the SWaT testbed was derived. We launched the attacks through the data communication link in Level 1 of the network (Fig. 3). In essence, we hijack the data packet and manipulate the sensor data before sending the packet to the PLCs. We assumed that an attacker succeeds in launching an attack. We assume that an attacker is successful in launching an attack, hence the number of possible attack scenarios is infinite.

The attack model [1] for CPS is abstracted as a sextuple $(M; G; D; P; S_0; S_e)$, where M is potentially an infinite set of procedures to launch attacks, G is a subset of a finite set of attacker intents, D is the domain model for the attacks derived from the CPS, P is a finite set of attack points, and S_0 and S_e are infinite sets of states of CPS, that denote, respectively, the possible start and end states of interest to the attacker. An attack point in CPS could be a physical element or an entry point through the communications network connecting sensors or actuators to the controllers (PLCs) and the SCADA system.

From the above discussion, it is clear that the space of potential attacks is large. The massive size of the attack space arises by changing the method M, potential attack points, P, as well as the start and end state of the CPS. SWaT consists of six stages where each stage contains different number of sensors and actuators. Based on attack points in each stage, the attacks are divided into four types.

1. Single Stage Single Point (SSSP): A Single Stage Single Point attack focuses on exactly one point in a CPS.
2. Single Stage Multi Point (SSMP): A Single Stage Multiple Point attack focuses on two or more attack points in a CPS but on only one stage. In this case set, P consists of more than one element in a CPS selected from any one stage.
3. Multi Stage Single Point (MSSP): A Multi Stage Single Point attack is similar to an SSMP attack except that now the SSMP attack is performed on multiple stages.
4. Multi Stage Multi Point (MSMP): A Multi Stage Multi Point attack is an SSMP attack performed two or more stages of the CPS.

For a detailed description of the attacks generated, we refer the reader to the dataset website[1]. The data collection process consisted of the following steps.
Step 1: Define each attack based on the number of attack points and places.
Step 2: Design each attack based on the attack point (i.e. the actuator or sensor to be affected affect), start state, type of attack, the value of the selected sensor data to be sent to the PLC, the intended impact.

A total of 36 attacks were launched during the data collection process. The breakdown of these attacks are listed in Table 1. The duration of the attack is varied based on the attack type. A few attacks, each lasting ten minutes, are performed consecutively with a gap of 10 min between successive attacks. Some

[1] http://itrust.sutd.edu.sg/research/datasets.

Table 1. Number of attacks per category

Attack category	Number of attacks
SSSP	26
SSMP	4
MSSP	2
MSMP	4

of the attacks are performed by letting the system stabilize before a subsequent attack. The duration of system stabilization varies across attacks. Some of the attacks have a stronger effect on the dynamics of system and causing more time for the system to stabilize. Simpler attacks, such as those that effect flow rates, require less time to stabilize. Also, some attacks do not take effect immediately.

4 Data Collection Process

The data collection process lasted for a total of 11 days. SWaT was functioning non-stop 24 hours/day, during the entire 11-day period. SWaT was run without any attacks during the first seven of the 11-days. Attacks were launched during the remaining four days. Various attack scenarios, discussed in Sect. 3, were implemented on the testbed. These attacks were of various intents and lasted between a few minutes to an hour. Depending on the attack scenario, the system was either allowed to reach its normal operating state before another attack was launched or the attacks were launched consecutively.

The following assumptions are made during the data collection process.

1. The system will stabilise and reach its operation state within the first seven days of normal operation.
2. Data is recorded once every second assuming that no significant attack on the SWaT testbed can be launched in less than one second.
3. The PLC firmware does not change.

All tanks in SWaT were emptied prior to starting data collection; i.e. the data collection process starts from an empty state of SWaT. This initialization was deemed necessary to ensure that all the tanks are filled with unfiltered water and not pre-treated.

4.1 Physical Properties

All the data was logged continuously once every second into a Historian server. Data recorded in the Historian was obtained from the sensors and actuators of the testbed. Sensors are devices that convert a physical parameter into an electronic output, i.e. an electronic value whereas actuators are devices that convert a signal into a physical output, i.e. turning the pump off or on.

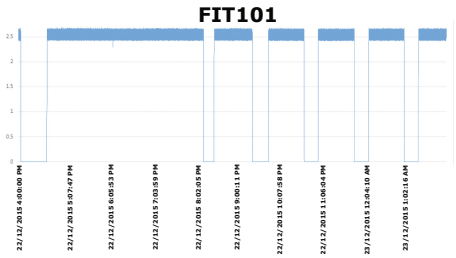
Table 2. Sensor and actuator description of the SWaT testbed.

No.	Name	Type	Description
1	FIT-101	Sensor	Flow meter; Measures inflow into raw water tank
2	LIT-101	Sensor	Level Transmitter; Raw water tank level
3	MV-101	Actuator	Motorized valve; Controls water flow to the raw water tank
4	P-101	Actuator	Pump; Pumps water from raw water tank to second stage
5	P-102 (backup)	Actuator	Pump; Pumps water from raw water tank to second stage
6	AIT-201	Sensor	Conductivity analyser; Measures NaCl level
7	AIT-202	Sensor	pH analyser; Measures HCl level
8	AIT-203	Sensor	ORP analyser; Measures NaOCl level
9	FIT-201	Sensor	Flow Transmitter; Control dosing pumps
10	MV-201	Actuator	Motorized valve; Controls water flow to the UF feed water tank
11	P-201	Actuator	Dosing pump; NaCl dosing pump
12	P-202 (backup)	Actuator	Dosing pump; NaCl dosing pump
13	P-203	Actuator	Dosing pump; HCl dosing pump
14	P-204 (backup)	Actuator	Dosing pump; HCl dosing pump
15	P-205	Actuator	Dosing pump; NaOCl dosing pump
16	P-206 (backup)	Actuator	Dosing pump; NaOCl dosing pump
17	DPIT-301	Sensor	Differential pressure indicating transmitter; Controls the backwash process
18	FIT-301	Sensor	Flow meter; Measures the flow of water in the UF stage
19	LIT-301	Sensor	Level Transmitter; UF feed water tank level
20	MV-301	Actuator	Motorized Valve; Controls UF-Backwash process
21	MV-302	Actuator	Motorized Valve; Controls water from UF process to De-Chlorination unit
22	MV-303	Actuator	Motorized Valve; Controls UF-Backwash drain
23	MV-304	Actuator	Motorized Valve; Controls UF drain
24	P-301 (backup)	Actuator	UF feed Pump; Pumps water from UF feed water tank to RO feed water tank via UF filtration
25	P-302	Actuator	UF feed Pump; Pumps water from UF feed water tank to RO feed water tank via UF filtration
26	AIT-401	Sensor	RO hardness meter of water
27	AIT-402	Sensor	ORP meter; Controls the $NaHSO_3$dosing(P203), NaOCl dosing (P205)
28	FIT-401	Sensor	Flow Transmitter; Controls the UV dechlorinator
29	LIT-401	Actuator	Level Transmitter; RO feed water tank level
30	P-401 (backup)	Actuator	Pump; Pumps water from RO feed tank to UV dechlorinator
31	P-402	Actuator	Pump; Pumps water from RO feed tank to UV dechlorinator
32	P-403	Actuator	Sodium bi-sulphate pump
33	P-404 (backup)	Actuator	Sodium bi-sulphate pump
34	UV-401	Actuator	Dechlorinator; Removes chlorine from water
35	AIT-501	Sensor	RO pH analyser; Measures HCl level
36	AIT-502	Sensor	RO feed ORP analyser; Measures NaOCl level
37	AIT-503	Sensor	RO feed conductivity analyser; Measures NaCl level
38	AIT-504	Sensor	RO permeate conductivity analyser; Measures NaCl level
39	FIT-501	Sensor	Flow meter; RO membrane inlet flow meter
40	FIT-502	Sensor	Flow meter; RO Permeate flow meter
41	FIT-503	Sensor	Flow meter; RO Reject flow meter
42	FIT-504	Sensor	Flow meter; RO re-circulation flow meter
43	P-501	Actuator	Pump; Pumps dechlorinated water to RO
44	P-502 (backup)	Actuator	Pump; Pumps dechlorinated water to RO
45	PIT-501	Sensor	Pressure meter; RO feed pressure
46	PIT-502	Sensor	Pressure meter; RO permeate pressure
47	PIT-503	Sensor	Pressure meter; RO reject pressure
48	FIT-601	Sensor	Flow meter; UF Backwash flow meter
49	P-601	Actuator	Pump; Pumps water from RO permeate tank to raw water tank (not used for data collection)
50	P-602	Actuator	Pump; Pumps water from UF back wash tank to UF filter to clean the membrane
51	P-603	Actuator	Not implemented in SWaT yet

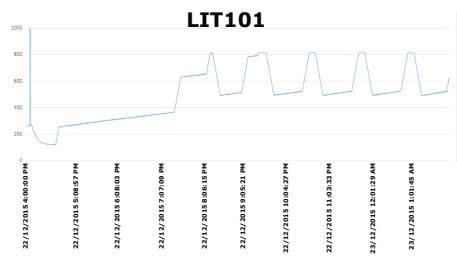
The dataset describes the physical properties of the testbed in operational mode. In total, 946,722 samples comprising of 51 attributes were collected over 11 days. Data capturing the physical properties can be used for profiling cyber-attacks. Table 2 describes the different sensors and actuators in SWaT that served as source of the data.

As the data collection process started from an empty state, it tool about 5 h for SWaT to stablise. Figure 4(a) indicates a steady flow of water into the tank in P1 (the level of tank is reported by sensor LIT101). Figure 4(b) shows that it took approximately 5 h for the tank to fill up and reach its operational state. For the tanks in stages P3 and P4 (level of tank reported by sensor LIT301 and LIT401 respectively), it took approximately 6 h for the tanks to be filled up. This is because the water from P1 is sent to P2 for chemical dosing before it reaches P3, hence an additional hour is needed to fill up the tank. The water from P3 is subsequently sent to P4 for reverse osmosis.

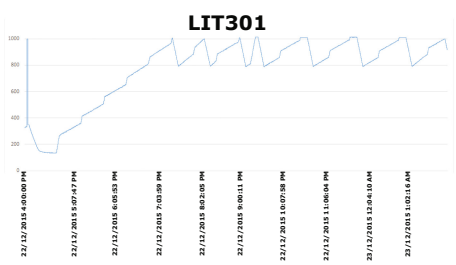
Figures 5(a) and (b) illustrate consequences of cyber attacks. Figure 5(a) illustrates a disturbance in the usual cycle of the reading from sensor LIT101 during 6:30 pm and 6:42 pm This was an SSSP attack with the intention of overflowing the tank by shutting pump P101 off and manipulating the value of LIT101 to be at 700 mm for 12 min. The effects are immediately observed over the next hour before the data stabilised nearly two hours later. Similarly Fig. 5(b) shows the consequence of an SSSP attack with the intention to underflow the tank and damage pump P101. In this attack sensor LIT-301 was attacked between



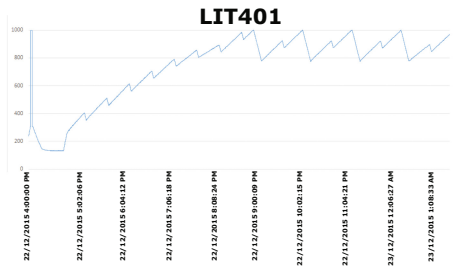
(a) Stablisation of FIT101



(b) Stablisation of LIT101

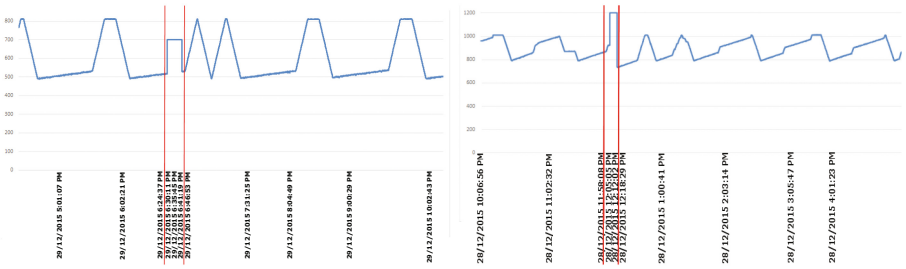


(c) Stablisation of LIT301



(d) Stablisation of LIT401

Fig. 4. First 10 h of data collection



(a) Single point attack on LIT101 (b) Single point attack on LIT301

Fig. 5. Attack data plots

12.08pm and 12.15pm to increase the sensor level to 1100 mm. This deceives the PLC to think that there is an over supply of water and turns the pump on to supply water to P4. In reality, the water level falls below the low mark while the pump is still active. Given sufficient time, this attack can cause the tank in P3 to underflow t, thus stagnating the output of the plant and damaging the pumps.

4.2 Network Traffic

Network traffic was collected using commercially available equipment from Check Point® Software Technologies Ltd[2]. This equipment was installed in the SWaT testbed. The use case of the equipment was specifically to collect all the network traffic for analysis. However, for the purpose of data collection, we retrieved network traffic data which is valuable for intrusion detection as in Table 3. Similarly, the data collection for network traffic began the moment the testbed was switched to operational mode. The attacks were performed at level 1 of the SWaT network as discussed in Sect. 2. The network data captures the communication between the SCADA system and the PLCs. Hence, the attacks were launched by hijacking the packets as they communicate between the SCADA system and the PLCs. During the process, the network packets are altered to reflect the spoofed values from the sensors.

4.3 Labelling Data

As the attacks performed in this paper were through a controlled process, labelling of the data turned out be straight forward. During the operation mode of the testbed, any actions to the testbed were required to be logged. Hence, all attacks performed for the purpose of data collection were logged with the information in Table 4.

[2] http://us.checkpointsystems.com/.

Table 3. Network traffic data

Category	Description
Date	Date of Log
Time	Time of Log
Origin	IP of server
Type	Type of log
Interface Name	Network interface type
Interface Direction	Direction of data
Source IP	IP Address of source
Destination IP	IP address of destination
Protocol	Network Protocol
Proxy Source IP	Proxy address of Source
Application Name	Name of application
Modbus Function Code	Function Code
Modbus Function Description	Description of Modbus Function
Modbus Transaction ID	Transaction ID
SCADA Tag	Sensor or Actuator ID
Modbus Value	Value transmitted
Service/Destination Port	Port number of Destination IP
Source Port	Port number of Source IP

Labelling of Physical Properties. Each data item corresponding to a sensor or an actuator data was collected individually into a CSV file. Each CSV file contains server name, sensor name, value at that point in time, time stamp, questionable, annotated and substituted. As the attributes are from the server, questionable, annotated and substituted are redundant and hence removed. All the remaining data was then combined into a single CSV file. Figure 6 illustrates

Table 4. Attack logs

Information	Description
Start time	Time when attack starts
End time	Time when attack ends
Attack points	Sensors or actuator which will be compromised
Start state	Current status of the point
Attack	Description of attack
Attack value	Substituted value of sensor (based on the attack)
Attacker's intent	The intended affect of the attack

Timestamp	FIT101	LIT101	MV101	P101	P102	AIT201	AIT202	AIT203	FIT201	MV201	P201
28/12/2015 10:00:00 AM	2.427057	522.8467	2	2	1	262.0161	8.396437	328.6337	2.445391	2	1
28/12/2015 10:00:01 AM	2.446274	522.886	2	2	1	262.0161	8.396437	328.6337	2.445391	2	1
28/12/2015 10:00:02 AM	2.489191	522.8467	2	2	1	262.0161	8.394514	328.6337	2.442316	2	1
28/12/2015 10:00:03 AM	2.53435	522.9645	2	2	1	262.0161	8.394514	328.6337	2.442316	2	1
28/12/2015 10:00:04 AM	2.56926	523.4748	2	2	1	262.0161	8.394514	328.6337	2.443085	2	1
28/12/2015 10:00:05 AM	2.609294	523.8673	2	2	1	262.0161	8.394514	328.6337	2.44411	2	1
28/12/2015 10:00:06 AM	2.637158	524.1028	2	2	1	262.0161	8.394514	328.6337	2.44411	2	1
28/12/2015 10:00:07 AM	2.652211	524.2206	2	2	1	262.0161	8.394514	328.6337	2.441803	2	1
28/12/2015 10:00:08 AM	2.655735	524.4954	2	2	1	262.0161	8.394514	328.6337	2.441803	2	1
28/12/2015 10:00:09 AM	2.64997	524.0636	2	2	1	262.0161	8.394514	328.6337	2.441803	2	1
28/12/2015 10:00:10 AM	2.630433	524.1028	2	2	1	262.0161	8.394514	328.6337	2.441803	2	1

Fig. 6. Example of physical properties data

a snap shot of the physical properties data. Using the attack logs, data was subsequently labelled manually based on the start and end-times of the attacks.

Labelling of Network Traffic. The network data was separated into multiple CSV files with a line limit of 500,000 packets for easier processing. However, as the data was captured at per second interval, there are instances of overlap where multiple rows reflect a different activity but carry the same time stamp. Similarly, based on the attack logs, the data was labelled based on the end and start time of the attacks. Figure 7 illustrates a snap shot of the presented network data saved as a CSV file.

Date	Time	Origin	Type	Interface	Interface Direction	Source	Destination	Protocol	Application Name	Proxy Source IP	Modbus_Function_Code
31-Dec-15	8:24:26	192.168.1.	log	eth1	outbound	192.168.1.	192.168.1.20	tcp	CIP_read_tag_service	192.168.1.10	76
31-Dec-15	8:24:26	192.168.1.	log	eth1	outbound	192.168.1.	192.168.1.10	tcp	CIP_read_tag_service	192.168.1.60	76
31-Dec-15	8:24:26	192.168.1.	log	eth1	outbound	192.168.1.	192.168.1.20	tcp	CIP_read_tag_service	192.168.1.10	76
31-Dec-15	8:24:26	192.168.1.	log	eth1	outbound	192.168.1.	192.168.1.10	tcp	CIP_read_tag_service	192.168.1.60	76
31-Dec-15	8:24:26	192.168.1.	log	eth1	outbound	192.168.1.	192.168.1.20	tcp	CIP_read_tag_service	192.168.1.60	76
31-Dec-15	8:24:26	192.168.1.	log	eth1	outbound	192.168.1.	192.168.1.20	tcp	CIP_read_tag_service	192.168.1.10	76
31-Dec-15	8:24:26	192.168.1.	log	eth1	outbound	192.168.1.	192.168.1.30	tcp	CIP_read_tag_service	192.168.1.20	76
31-Dec-15	8:24:26	192.168.1.	log	eth1	outbound	192.168.1.	192.168.1.40	tcp	CIP_read_tag_service	192.168.1.30	76
31-Dec-15	8:24:26	192.168.1.	log	eth1	outbound	192.168.1.	192.168.1.30	tcp	CIP_read_tag_service	192.168.1.20	76
31-Dec-15	8:24:26	192.168.1.	log	eth1	outbound	192.168.1.	192.168.1.20	tcp	CIP_read_tag_service	192.168.1.60	76

Fig. 7. Example of network data

5 Conclusion

The lack of reliable and publicly available CPS datasets is a fundamental concern for researchers investigating the design of secure CPSs. There are currently no such large scale public datasets available as there are no open CPS facilities. Real industrial CPS facilities would not be able to provide accurate datasets as faults or attacks can only be assumed at best.

The data collected from the SWaT testbed reflects a real-world environment that helps to ensure the quality of the dataset in terms of both normal and attack data. The attacks carried out by the authors illustrate how such attacks can take place in modern CPSs and provide us the ability to provide accurately label data for subsequent use. The information and data that is provided with this paper includes both network and physical properties stored in CSV file formats.

Our goal is to make the collection of CPSs datasets an on-going process to benefit researchers. The data collected will be continuously updated to include

datasets from new testbeds as well as new attacks derived from our research team.

Acknowledgments. This work was supported by research grant 9013102373 from the Ministry of Defense and NRF2014-NCR-NCR001-040 from the National Research Foundation, Singapore. The authors would like to thank Check Point® Software Technologies Ltd for the loan of their network equipment for data collection purposes.

References

1. Adepu, S., Mathur, A.: An investigation into the response of a water treatment system to cyber attacks. In: Proceedings of the 17th IEEE High Assurance Systems Engineering Symposium (2016)
2. Bay, S.D., Kibler, D., Pazzani, M.J., Smyth, P.: The uci kdd archive of large data sets for data mining research and experimentation. ACM SIGKDD Explor. Newslett. **2**(2), 81–85 (2000)
3. Lippmann, R.: The 1999 darpa off-line intrusion detection evaluation. Comput. Netw. **34**(4), 579–595 (2000). http://www.ll.mit.edu/IST/ideval/
4. Morris, T.: Industrial control system (ics) cyber attack datasets - tommy morris. https://sites.google.com/a/uah.edu/tommy-morris-uah/ics-data-set. Accessed 06 May 2016

Human Vulnerability Mapping Facing Critical Service Disruptions for Crisis Managers

Amélie Grangeat[1(✉)], Julie Sina[2], Vittorio Rosato[3], Aurélia Bony[2], and Marianthi Theocharidou[4]

[1] CEA/GRAMAT, 46500 Gramat, France
amelie.grangeat@orange.fr

[2] Institut des Sciences des Risques – Centre LGEI, Ecole des mines d'Alès, 30100 Alès, France
julie.sina@hotmail.fr, aurelia.bony-dandrieux@mines-ales.fr

[3] ENEA Casaccia Research Centre, Rome, Italy
vittorio.rosato@enea.it

[4] European Commission, Joint Research Centre (JRC), Via E. Fermi 2749, 21027 Ispra, VA, Italy
marianthi.theocharidou@ec.europa.eu

Abstract. Societies rely on the exchange of essential goods and services produced by Critical Infrastructures (CI). The failure of one CI following an event could cause "cascading effects" on other CI that amplifies the crisis. Existing tools incorporate modelling and simulation techniques to analyze these effects. The CIPRNet tools go a step further by assessing the consequences on population in a static manner: people are located at their census home; their sensibility to a resource lack varies during the day. This paper improves human impacts assessment by mapping people mobility thanks to the DEMOCRITE project methodology. It focuses on location of people with regards to their activities and the time period (night/day, holidays), and discuss their sensibility to the lack of key infrastructure services. Human vulnerability maps of Paris area during periods of a working day time show the importance to take into account people mobility when assessing crisis impacts.

Keywords: Cascading effects · Critical Infrastructure · Vulnerability mapping Risk analysis · GIS method

1 Introduction

Societies rely on the function of Critical Infrastructures (CI). These are considered critical because of the importance of the services they provide to the country and its population: one finds energy, water, telecommunications, transports, and other sectors. This means also that

---

Marianthi Theocharidou is an employee of the European Union and the copyright on her part of the work belongs to the European Union.

The original version of this chapter was revised: A footnote was added to acknowledge that the copyright on Marianthi Theocharidou's part of the work belongs to the European Union. An erratum to this chapter can be found at https://doi.org/10.1007/978-3-319-71368-7_31

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 100–110, 2017.
https://doi.org/10.1007/978-3-319-71368-7_9

a CI failure may have a severe impact on both a country and its people. Moreover, CI are highly interdependent: water network needs electricity, telecommunication networks may need water for cooling their datacentres, electricity networks rely on SCADA (Supervisory Control And Data Acquisition) networks which use telecommunication systems, and the list of dependencies is long. This network of dependencies improves the efficiency of CI functioning in a normal time. However, the failure of one component poses the risk to spread through this graph and to disrupt other CI by cascading effect.

Major disasters like the hurricane Sandy in 2012 have proved the necessity to take into account cascading effects when managing a crisis. Indeed, a cascading effect means an extension of impacts in terms of location and duration, with a different aspect than the initial consequences of the disaster. For instance, the flooding of only some of the New-York subway stations blocked the city in a larger scope and slowed down the business recovery after the hurricane [5].

Emergency management institutions are perhaps well prepared to a strong event, but it is known that the cascading effect is a weak point of the preparation [12]. It necessitates the understanding of each CI functioning, but also the knowledge of the global system behavior facing a crisis. For helping crisis managers to have a better awareness on cascading effects some tools propose to model CI dependencies and components, based on real data at least at a city scale [7]. However, crisis managers require on top of these simulations a timely, accurate and realistic assessment of the consequences of an event on the population.

The Critical Infrastructure Preparedness and Resilience Research Network or CIPRNet is a European FP7 project that establishes a Network of Excellence in CI Protection (CIP) [2]. Its long term goal is to build a long-lasting virtual centre of shared and integrated knowledge and expertise for the CIP community, with capabilities of modelling, simulation and analysis. This virtual centre aims at forming the foundation for a European Infrastructures Simulation & Analysis Centre (EISAC) by 2020.

The CIPRNet Decision Support System (DSS), comprises of five parts: Operational DSS (gathering of real time external inputs), DSS with Event Simulator (modeling of natural events), DSS with synthetic Harm Simulator (estimating infrastructures damages), an impact assessment tool (modeling cascading effect) and a What-if analysis tool. This last part aims at "designing, comparing, and validating mitigation and healing strategies through (what-if) analysis of potential consequences they produce." [3]. When considering preparedness and planning from the crisis manager's point of view, the more accurate the impact assessment, the better the quality of the responders' coordination. The following section explains the methods used for assessing consequences of cascading effects in CIPRNet DSS. Then, this article presents a way to enrich this impact assessment with respect to the impact on the population, by taking in account the mobility of population.

2 Consequence Assessment

A lot of work has been performed on the economic impact analysis of the disruption of services during or following crises, e.g. on the energy [10] or water sectors [1, 8]. One way to assess consequences of a crisis is based on Service Availability Wealth (SAW) Indexes, which is a method implemented in the CIPRNet DSS. After having predicted the occurrence

of damages along one (or more) CI present in a given area, the CIPRNet DSS estimates the reduction (or the loss) of functionality of the CI, due to predicted damages and dependency-induced cascading effects. These impacts are then transformed into perceived societal consequences expressed in terms of "reduction of wealth" of the different societal domains: citizens, availability of primary services, economic sectors and the environment. SAW indexes indicate the relevance of a specific service supplied by a CI on a given societal domain. The consequences estimate enables to "weight" the different disaster scenarios and to compare their severity in terms of consequences. This indicator of "reduction or loss of well-being" is composed of four criteria [15]:

- C_1: Reduction of well-being to the most vulnerable population (categories concern old [C_1, 1], young [C_1, 2], disabled people [C_1, 3] and others [C_1, 4]);
- C_2: Reduction of primary services that affect the wealth and the well-being of the population;
- C_3: Economic losses due to services outages;
- C_4: Direct and indirect environmental damages (if any) caused by the outages (release of pollutants in the environment etc.).

The previous criteria are affected directly by the event, but also by the lack of primary technological and energy services on different territories, for different time frames (Di Pietro et al. 2015): Electricity (R_1), telecommunications (R_2), water (drinking water, waste water management) (R_3), gas and other energetic products (R_4) and mobility[1] (roads, railways) (R_5).

The consequences of the crisis scenario on each criterion are calculated on the basis of [6]:

- the quality of the considered services which contribute to wealth (electricity, telecommunication, gas, water and mobility), i.e. their level of availability during the event (this is a function of time),
- the relevance of each service for the achievement of the maximum level of the wealth quantity for a given element of criteria, and
- the metrics for wealth measure (for example the number of people affected in a population segment during the time period t).

The CIPRNet project gathers statistical data on the consumption of primary technological and energy services like average monthly households expenditure on electricity or gas, to compute the relevance indexes of the service "telecommunication".

By this way, a table of the SAW indexes of service relevance with respect to the criterion "Citizens" for each couple ($C_{1,i,}$ R_j) with $i \in [1, 4]$ and $j \in [1, 5]$ is given.

At the end, a typical day (working vs. non-working day) with time schedule and statistical activities is proposed. For instance, electricity use during a day is split into nine different functions: lighting, refrigerator/freezer, air conditioning, TV, oven, microwave, washing machine & dryer, and a global section for other appliances. Evaluating the importance of various activities requiring services within a daily time schedule, CIPRNet project obtains a normalized indicator of relevance of Services r_k table (SAW

[1] Mobility refers here to the availability of transport.

Indexes) for each service and each category of citizen every 30 min. Similar tables are created for the other different societal domains and for each element in which they are described.

In the long term, SAW indexes will be properly estimated (through the analysis of economic, social and environmental data); their time-of-the-day dependence will be also properly assessed. The CIPRNet DSS will thus deliver to Crisis Managers, CI Control-Room operators a number of forecast data (expected damages, possible impacts on services, societal consequences) and, in some cases, optimization strategies to recover crisis situation. A typical case is the optimization of the recovery strategy of the electrical distribution service (in terms of optimal sequence of restoration of the different faulted elements) in order to minimize societal consequences (each possible restoration sequence is weighted with an optimizing function related to the consequences that the specific sequence will produce). The resulting information provided by the CIPRNet DSS is meant to improve the systemic resilience, particularly in large and complex urban scenarios. The systemic resilience considers both the societal system and the system of technical networks. The term systemic is here in opposition with a local analyze of one infrastructure in an independent manner of its environment.

In terms of people mobility, the population is linked to specific residential segments. Movement is affecting the relevance of the service 'electricity', 'telecommunications' or 'water', but in a static way. For example, Di Pietro et al. [6] assume that the relevance of specific services drops for the 'Citizens 18–64' profile during working hours as they are considered absent from home. On the contrary, the increase of service relevance in a business area due to this movement is not clearly depicted. Therefore, an improvement to the model would be to take into account and visualize the mobility of people. In the following section, a methodology to achieve this is described and its contribution to improve the assessment of CI's disruptions is discussed.

3 Modelling People Mobility

Having statistical information on people location is a significant help for crisis management – and can also be used to enrich crisis scenario simulation and analysis. Osaragi underlines the necessity to accurately estimate the population exposure when assessing crisis consequences. This precision means to understand the spatiotemporal variation of the population distribution and not to rely only on census static data [14]. The Ile-de-France French civil safety institution handles a research project named DEMOCRITE [11] to map dynamically (among other tasks) human vulnerability in Paris. We define "human vulnerability" of one territory as the spatiotemporal distribution of people: the more concentrated is the population, the more important is the human vulnerability. They are a "vulnerability" in the sense that people are the main stake to protect during a crisis, facing a threat. The methodology is shortly presented below [16] and on Fig. 1.

In order to assess this spatiotemporal distribution of people, the population is split in seven categories depending on its activities: people staying at home, people in public transports (roads and subways), people in buildings open to public (as hospitals, schools, malls, etc.), people working in companies and industries and people in touristic locations

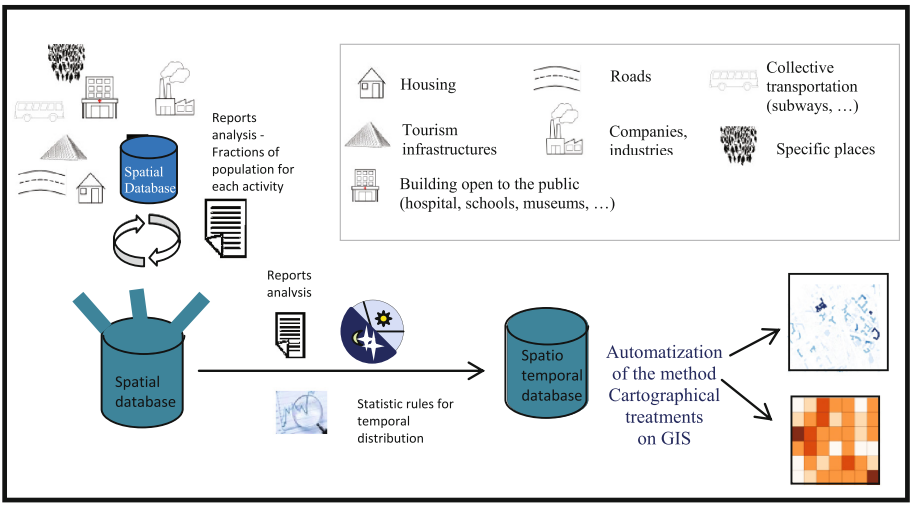


Fig. 1. Flowchart of the retained methodology [16]

(as museums, monuments, etc.). Those places had been identified because they are likely to concentrate a significant number of people.

Nevertheless, the concentration may vary over time. In order to take into account this variation, a year has been divided into four periods:

- Weekdays;
- Saturdays;
- Sundays;
- Summer holidays (1 July–31 August), which are assimilated to Sundays.

Then, each period has been divided into 4 time slots representative of a typical day for most of population:

- Morning rush hour: 7:30 am–9:30 am;
- Daytime: 9:30 am–4:30 pm;
- Evening rush hour: 4:30 pm–7:30 pm;
- Night: 7:30 pm–7:30 am.

Numerous databases, mostly open source ones, and various reports were explored. In total, more than 70 databases were used. Only the more complete and accurate were retained. Once the spatial database is built, temporal distribution had to be calculated. This is performed according to specific rules derived from statistic treatments of available reports concerning the living habits in Paris (opening hours and days of museums, underground frequentation during a working/non-working day and so on). It enables us to simulate how many people may be in buildings as a function of the buildings categories and the time slot.

For example, on the basis of geographical census data and of various statistics on population (age, unemployment, etc.), it is possible to deduce more information. Then the percentage of people staying at home is calculated and may include percentage of

unemployed people; of young babies and of retired people. The same approach is used to estimate people present in shops: on the basis of the shopping surfaces of buildings, one can deduce the maximum capacity of shoppers, and on the basis of statistics on hourly shopping habits, one can calculate the potential numbers of people in these places.

In the same way, education buildings are assumed to be full during class hours but empty during the night, such as the companies' buildings and so on. The visitor numbers of museums and tourist sites are investigated and are associated with their opening hours. Furthermore the number of subway users is also analysed in order to obtain temporal distribution of people in the subway stations.

All these results have been merged to propose the more complete database possible, containing spatial and temporal distribution of population. This database is not exhaustive: data about employees per industries/companies, road traffic part of public transports are not available on the overall studied territory. The database presents also some imprecisions and it needs therefore to be completed and modified in the future. This database is nevertheless a very useful tool to assess the spatiotemporal distribution of population in Paris. Finally, the method is automated and proposes maps of vulnerability by counting people present in each mesh composing the territory for the different period of times identified.

4 Results and Interests of Mapping Human Vulnerability

The following maps (illustrative examples) show the evolution of human vulnerability during different periods for a working day. The information concerning people's locations and number is gathered and aggregated in a grid mesh (the scale is not given for security reasons). The represented value in each small mesh is the number of persons

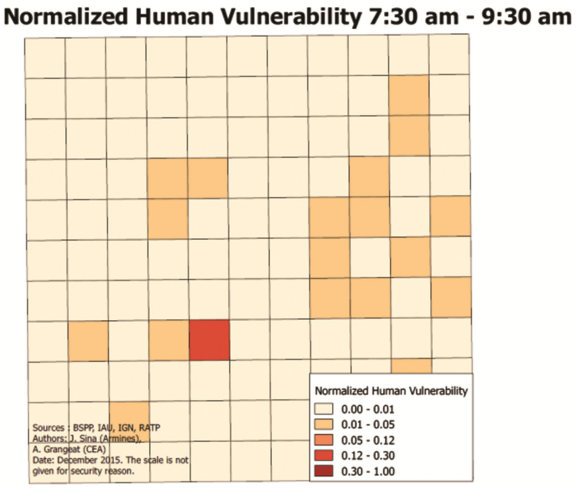


Fig. 2. Human vulnerability maps on the morning of a working day

present in this small mesh normalized by the highest value obtained over all the periods studied and over the overall mesh.

The example which is shown below focuses on a district in the Paris region which is characterized mainly by tertiary industries. This area is crowded during working days and has less population during the night. The surrounding areas are more residential. The maps of human vulnerability simulate correctly this behavior in Figs. 2, 3 and 4. During working hours, the vulnerability of several parts of the mesh are high due to the presence of buildings, commercial malls, museums and other touristic zones, subway stations which concentrate workers, shoppers and tourists. During the night, the hotels

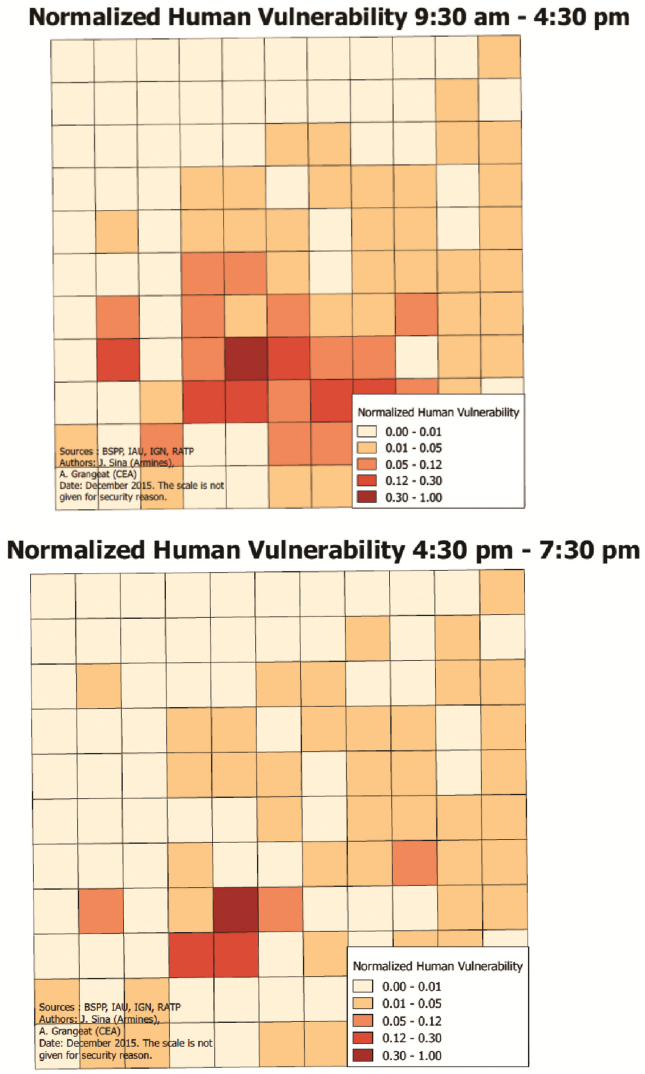


Fig. 3. Human vulnerability maps during a working day

found in the area maintain a high human vulnerability. The high number of shopping centers or public access buildings maintains the relative high number of people in several inner meshes between 4:30 pm to 7:30 pm. The night map computes the residential area and the hotels filling. This last map reveals the difference between a static representation of human vulnerability (taking only the census of people into account) and the dynamic human vulnerability of the day, taking into account offices, shops and public access buildings.

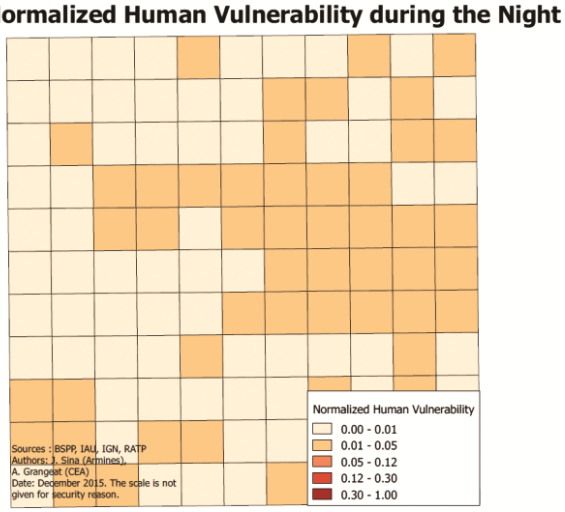


Fig. 4. Human vulnerability maps on the evening of a working day

The differences between the maps obtained for various periods over the day reveal that taking into account the location of people – in a statistic way while including temporal variation – has a huge importance when one wants to assess the consequences of a crisis.

This spatial analysis methodology may always be further improved but this work is already an important basis. Another step could improve the final information quality for decision-makers during a crisis. People and business are differently sensitive to the failure of technical services as a function of their activities. The following paragraph proposes some key aspects that could be integrated to a further work on support for crisis management.

5 Improving Human Vulnerability Assessment

The distribution of people on the territory is now identified. These results must be coupled with the impact on them caused by the inoperability of the main services delivered by CI. The people vulnerability varies with the kind of resource disruption (electricity, internet, phone services, etc.) the hour and the duration of this interruption and

their activity. Two human vulnerability states may be distinguished: the discomfort state (initial impact) and the crisis state.

Under normal conditions, people's location suggests a probable everyday activity: people in shopping centers are shopping, people in substations are travelling, people in office are working, and so on. The activities' sensitivity to a resource shortage varies. It may be modelled as a deterioration of the well-being indicator following the CIPRNet consequence analysis approach. For instance, people shopping may not be sensitive to a water disruption. Contrary, people working in office towers will be more disturbed by the lack of water for sanitary reasons. Moreover, well-being of clients or workers determines for one part the local economy functioning. The maps produced in the previous section -when combined with the direct impact of a disruption to a CI service- allow for an estimate of the vulnerability of the population due to its location, activity and the disruption of services. One can also extend the analysis to other consequences, such as economic, environmental, etc. These do not refer only to the direct effect on the well-being of population, but to social vulnerability [4, 9] as well. Quantifying these various impacts may help to compare resource disruption scenarios over a brief period.

However, when such CI disruptions expand over time, they may cause cascading effects and escalate to a crisis. For more long-term effects, one needs to consider whether the normal mobility maps produced are applicable in this case. People may not be returning to their offices, they may not visit shopping malls and tourists may not stay in a hotel in the affected area. This requires different maps to be created or specific ones to be selected as representative, e.g. the use of the maps corresponding to the night time period. Moreover, the CI service relevance for specific areas may be modified, as during crisis the needs of the population may shift or be altered. In this case, one may need to prioritize on the services, which are considered essential for the crisis manager [13]. An example would be to base the consequence analysis on the minimum levels of CI service needed, so as minimum essential daily needs can be met for the population. An example is the work of the Sphere Project [17] which defines minimum water intake levels to

Table 1. Vulnerability states definition

Vulnerability states	Definition
Discomfort state	It concerns a short period of resource disruption or a much localized disruption. People can adapt themselves to the resource lack, either by using an alternative resource for a limited period, either because the resource is available closed by, either because they can live/work/travel without the missing resource for a limited period of time and can deal with it
Crisis state	It is the degradation of the discomfort state, because of the too long duration of one resource lack on one important area. This long service disruption causes either sanitarian problems (for instance no more heating caused by a gas disruption in winter, toilettes out of order following a too long water disruption, and so on) either a strong obstacle for working (electricity or telecommunication interruptions) or travelling. This state means that people will evacuate their living place or will not go to work because of this long interruption of resource

be attained for the coverage of basic survival needs (water intake by food and drinking), basic hygiene practices and basic cooking needs. Table 1 proposes a first definition of the limit between the discomfort state and the crisis state.

6 Conclusion

Assessing human vulnerability as a function of their location and activity helps crisis managers in assessing more accurately the first impacts of a resource disruption. This paper proposes a method to simulate people mobility and activities according to their location during a typical day. The next step is to couple this new database with the relevance of each service for the well-being or the health of people. Future work aims at mapping the minimum level of service needed when a disruption occurs. This research would help to prioritize actions for the emergency provision of resources or for the restoration of CI.

Acknowledgments. This work was derived from the FP7 projects CIPRNet and PREDICT, which has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreements no. 312450 and no. 607697, respectively. The contents of this article do not necessarily reflect the official opinion of the European Union. Responsibility for the information and views expressed herein lies entirely with the authors. The authors want to thank the DEMOCRITE (ANR-13-SECU-0007) project that enables the research on this topic.

References

1. Brozović, N., Sunding, D.L., Zilberman, D.: Estimating business and residential water supply interruption losses from catastrophic events. Water Resour. Res. **43**, W08423 (2007)
2. CIPRNet: CIPRNet project website (2015). http://ciprnet.eu/summary.html. Accessed 11 Jan 2016
3. CIPRNet: CIPRNet Flyer: Critical Infrastructure Research and Resilience Network, 4 pp., 26 February 2015
4. Cutter, S.L., Boruff, B.J., Shirley, W.L.: Social vulnerability to environmental hazards. Soc. Sci. Q. **84**(2), 242–261 (2003). https://doi.org/10.1111/1540-6237.8402002. ©2003 by the Southwestern Social Science
5. Gibbs, L.I., Holloway, C.F.: Hurricane Sandy After Action Report and Recommendations to Mayor Michael R. Bloomberg, New-York City (2013)
6. Di Pietro, A., La Porta, L., Lavalle, L., Pollino, M., Rosato V., Tofani, A.: CIPRNet deliverable D7.4 Implementation of the DSS with consequence analysis. Technical report, ENEA (2015)
7. Amélie, G., Aurélia, B., Emmanuel, L., Mohamed, E., Gilles, D.: The challenge of critical infrastructure dependency modelling and simulation for emergency management and decision making by the civil security authorities. In: Rome, E., Theocharidou, M., Wolthusen, S. (eds.) CRITIS 2015. LNCS, vol. 9578, pp. 255–258. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-33331-1_23

8. Heflin, C., Jensen, J., Miller, K.: Understanding the economic impacts of disruptions in water service. Eval. Program Plan. **46**, 80–86 (2014). © 2014 Elsevier Ltd. Published by Elsevier Ltd. Editor-in-chief: Morell J.A.
9. Holand, I.S., Lujala, P., Röd, J.K.: Social vulnerability assessment for Norway: a quantitative approach. Nor. Geogr. Tidsskr. **65**(1), 1–17 (2011)
10. LaCommare, K.H., Eto, J.H.: Cost of power interruptions to electricity consumers in the United States (US). Energy Int. J. **31**(12), 1845–1855 (2006). Editor-in-Chief: N. Lior
11. Lapebie, E.: Concepts, Systèmes et Outils pour la Sécurité Globale (CSOSG) 2013: Projet DEMOCRITE. Emmanuel LAPEBIE, CEA. Agence Nationale de la Recherche. http://www.agencenationalerecherche.fr/?Projet=ANR13SECU0007. Accessed Nov 2015
12. Luiijf, E.: Avoid enlarging a disaster. Presentation from TNO for Engineers Without Borders, 28 November 2015. 22 slides
13. Nieuwenhuijs, A., Luiijf, E., Klaver, M.: Modeling dependencies in critical infrastructures. In: Papa, M., Shenoi, S. (eds.) ICCIP 2008. IFIPAICT, vol. 290, pp. 205–213. Springer, Boston (2008). https://doi.org/10.1007/978-0-387-88523-0_15
14. Osaragi, T.: Estimation of transient occupants on weekdays and weekends for risk exposure analysis. In: Tapia, Antunes, Bañuls, Moore and Porto de Albuquerque (eds.) Proceedings of the ISCRAM 2016 Conference – Rio de Janeiro, Brazil, May 2016, 17 p. (2016)
15. Rosato, V., Tofani, A., Pollino, M.: CIPRNet deliverable D7.1: design of the DSS with consequence analysis. Technical report, ENEA (2014)
16. Sina, J., Bony, A.: Livrable n°1: extension des approches vulnérabilité humaine et fonctionnelle. DEMOCRITE report number 150531-DEMOCRITE-ARMINES-Livrable-01-01, 101 p. (2015)
17. Sphere: Sphere project website (2015). http://www.sphereproject.org/. Accessed Jan 2016

A Methodology for Monitoring and Control Network Design

István Kiss(✉) and Béla Genge

"Petru Maior" University of Tîrgu Mureş, Tîrgu Mureş, Romania
istvan.kiss@stud.upm.ro, bela.genge@ing.upm.ro

Abstract. The accelerated advancement of Industrial Control Systems (ICS) transformed the traditional and completely isolated systems view into a networked and inter-connected "system of systems" perspective. This has brought significant economical and operational benefits, but it also provided new opportunities for malicious actors targeting critical ICS. In this work we adopt a Cyber Attack Impact Assessment (CAIA) technique to develop a systematic methodology for evaluating the risk levels of ICS assets. The outcome of the risk assessment is integrated into an optimal control network design methodology. Experiments comprising the Tennessee Eastman chemical plant, the IEEE 14-bus electricity grid and the IEEE 300-bus New England electricity grid show the applicability and effectiveness of the developed methodology.

Keywords: Industrial Control Systems · Impact assessment
Cyber attack · Optimal network design · Risk assessment

1 Introduction

The pervasive adoption of commodity, off-the-shelf Information and Communication Technologies (ICT) reformed the architecture of ICS. This has brought numerous benefits including the reduction of costs, greater flexibility, efficiency and interoperability between components. However, the massive penetration of ICT hardware and software into the heart of modern ICS also created new opportunities for malicious actors and facilitated the adoption of traditional cyber attack vectors for the implementation of a new breed of *cyber-physical* attacks. These represent a more sophisticated class of attacks where the characteristics of the cyber and the physical dimensions are exploited by the adversary in order to cause significant damages to the underlying physical process. Stuxnet [4], Flame [5] and Dragonfly [18] represent only a fraction from the number of threats showcasing the impact of exploiting ICT vulnerabilities in ICS.

As a response to these events, a significant body of research focused on the identification of critical ICS assets and on improving the security of these kind infrastructures [13,15–17]. However, previous studies do not address the dynamic behavior of ICS, the complexity of ICS, the existing inter-dependencies

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 111–122, 2017.
https://doi.org/10.1007/978-3-319-71368-7_10

between ICT and the physical process. Furthermore, the output of risk assessment methodologies is not integrated into a network design framework. Therefore, in this work we extend our solutions given in [12] by developing a framework for assessing the impact of cyber attacks on ICS and for the optimal design of ICS networks. The framework adopts a Cyber Attack Impact Assessment (CAIA) methodology in order to evaluate the risk levels associated to each ICS asset. The output of this approach is then used in an Integer Linear Programming (ILP) problem for optimally designing control networks. The aim of the optimization is to minimize the distance between concentrator nodes and end devices, as well as to maintain link capacity and security level constraints. As a result, the ICS designed according to the proposed framework ensures the protection of critical devices as well as their low installation and operational costs. The proposed framework is evaluated by means of extensive experiments including the Tennessee Eastman chemical plant (TEP) [6], the IEEE 14-bus electrical grid, the IEEE 300-bus electrical grid and various attack scenarios.

The remaining of this paper is structured as follows. Section 2 provides a brief overview of the related research. The description of the risk assessment technique is given in Sect. 3, which is followed by the network design technique in Sect. 4, experimental assessment in Sect. 5, and conclusions in Sect. 6.

2 Related Work

First, we investigate related cyber attack impact assessment techniques, such as the work of Kundur *et al.* [13]. Here, the authors proposed a graph-based model to evaluate the influence of control loops on a physical process. Differently, in [16], Sgouras *et al.* evaluated the impact of cyber attacks on a simulated smart metering infrastructure. The experiments implemented disruptive Denial of Service attacks against smart meters and utility servers, which caused severe communication interruptions. In a different work, Sridhar and Govindarasu [17] showed that cyber attacks may significantly impact power system stability by causing severe decline of system frequency. Bilis *et al.* in [2] proposed a complex network theory-based assessment methodology to show the individual importance of electric buses in power systems. Next, we briefly mention the related network design techniques. Carro-Calvo *et al.* [3] developed a genetic algorithm-based optimal industrial network partitioning, which maximized intra-network communications and minimized inter-network data transfers. Zhang *et al.* [19,20] elaborated a network design problem, which reduces network delays. In [9] Genge and Siaterlis revealed that the impact of local actuation strategies to other controllers should also be considered in network design procedures. Another work of Genge *et al.* in [8] proposed a Linear Programming-based network design optimization problem that accounts for the presence of primary and secondary networks, as well as for the capacity of links and devices, the security and real-time traffic requirements.

In contrast with the aforementioned techniques, the primary contribution of this work is that it delivers a complete framework for assessing the impact of

cyber attacks on ICS, for establishing concrete risk levels and for designing ICS. This represents a significant contribution over the state of the art since it closes the gap between risk assessment and security-aware network design.

3 Asset Risk Assessment in ICS

The architecture of modern ICS is structured according to two different layers: (i) the physical layer, which encompasses a variety of sensor, actuator, and hardware devices that physically perform the actions on the system; (ii) and the cyber layer, which encompasses all the ICT hardware and software needed to monitor the physical process and to implement complex control loops. From an operational point of view, hardware controllers, i.e., Programmable Logical Controllers (PLC), receive data from sensors, elaborate local actuation strategies, and send commands to the actuators. These hardware controllers also provide the data received from sensors to Supervisory Control and Data Acquisition (SCADA) servers and eventually execute the commands that they receive. Hereinafter, the devices that acquire and transmit data from multiple sources are referred to as Concentrator Nodes (CN).

3.1 Overview of the CAIA Approach

The cyber attack impact assessment (CAIA) technique proposed in our previous work [10] adopts a procedure inspired from the field of System Dynamics [7]. According to [7], a change in the systems' behavior is the result of interventions induced by control variables. In CAIA this effectively translates to the reduction of the state-space and more specifically of the number of variables and attacks that need to be evaluated. At the core of CAIA is a technique that records the *behavior* of complex physical processes in the presence of accidental or deliberate interventions, e.g., faults, events, and cyber attacks. Essentially, the cyber attack impact assessment procedure calculates the cross co-variance of observed variables before and after the execution of a specific intervention. Accordingly, an instance of CAIA results in an impact matrix denoted by C which columns correspond to observed variables and the rows to control variables. Therefore, the C impact matrix enables a detailed impact assessment in various scenarios by providing answers to research questions such as measuring how the intervention on the i-th control variable affects the j-th observed variable. The next section employs the impact matrix as the input to the *risk assessment procedure*. More details about the CAIA procedure are given in [10].

3.2 Risk Assessment Based on the Impact Measures

The CAIA procedure delivers relative impact values for one specific type of attack. Therefore, the risk assessment expands the CAIA approach and combines the output of multiple executions of CAIA for different attacks. First we define $A = \{1, 2, ..., \iota, ...\alpha\}$ as the set of attack types. We use C^{ι}_{ij} to denote the impact matrix

values such that C_{ij} is the calculated impact for the ι-th attack type ($\iota \in A$). Next, as a pre-processing step, a PCA (Principal Component Analysis) based weighting technique [14] is used to combine the results of impact assessments of multiple attack types and to construct a *severity* matrix Ω used later in the calculation of risk values. The values of this matrix denoted by $\omega_{i\iota}$, represent the severity of the intervention of type ι on the i-th variable, i.e., end device. The final outcome of risk assessment is given in Eq. (1) and is a vector of risk values for each attacked variable.

$$\Re_i = \sum_{\iota \in A} \omega_{i\iota} \cdot p_\iota, \forall i \in I, \tag{1}$$

where $p_\iota | \sum_{\iota \in A} p_\iota = 1$, is a vector containing the predefined probabilities for each type of cyber attack. Following, we provide the mathematical description for determining the severity matrix Ω. First, we compute the co-variance matrix $\Sigma^\iota, \forall \iota \in A$, as indicated in Eq. (2). Then, according to Eq. (3) we compute the factor loadings Z^ι by using the eigenvectors of Σ denoted by U. As stated in [14], the square of factor loadings represents "the proportion of the total unit variance of the indicator which is explained by the factor". Accordingly, Eq. (4) defines the variance ϑ_i^ι for each factor, where $v_i^\iota, \forall i \in I$ are the eigenvalues of Σ^ι. Next, using the above formulations, the severity matrix is defined by Eq. (5). For further convenient usage the severity matrix is normalized in the $[0, 1]$ interval. Finally, by *a priori* knowing the vector of probabilities p_ι, the application of Eq. (1) to determine the risk values $\Re_i$ for each attacked device becomes straightforward.

$$\Sigma^\iota = \frac{1}{n} C^\iota \cdot C^{\iota T}, \forall \iota \in A. \tag{2}$$

$$Z^\iota = U^{\iota T} \cdot C^{\iota T}, \forall \iota \in A. \tag{3}$$

$$\vartheta_i^\iota = \frac{v_i^\iota}{\sum_{l \in I} v_l^\iota}, \forall i \in I, \forall \iota \in A, \tag{4}$$

$$\omega_{i\iota} = \sum_{i \in I} Z_{il}^\iota \cdot \vartheta_l^\iota, \forall i, l \in I, \forall \iota \in A. \tag{5}$$

4 Optimal Control Network Design

In this section we employ the risk values to define a finite set of security levels. These serve as security level requirements for vulnerable variables, hereinafter referred to as end devices (ED). Then, a single linkage hierarchical clustering technique is applied to the risk values to develop a predefined number of ED groups. Each group corresponds to a security level requirement. As a constraint, an ED can connect to one of the concentrator nodes (CN) in order to maintain communication with supervisory and control stations. Therefore, the optimization problem discussed in this paper seeks the optimal connection of ED to CN

by minimizing the overall distance between ED and CN, but taking into account the maximum link capacity of CN and the security level requirement of ED.

In the description of the optimization problem we use the following notations: we define $\mathcal{C} = \{1, 2, ..., \imath, ..., \mathfrak{c}\}$ to denote the set of concentrator nodes, $\mathcal{D} = \{1, 2, ..., \jmath, ..., \mathfrak{d}\}$ the set of end devices and $\mathcal{S} = \{1, 2, ..., \kappa, ..., \mathfrak{s}\}$ as the set of available/predefined security levels. Furthermore, the optimization problem needs to account for other network parameters, such as link capacity, traffic demands and security levels. Let $s^{\mathcal{C}}_{\imath\kappa}$ be a binary parameter to denote if the $\imath$-th CN supports the κ-th security level. Then, let $s^{\mathcal{D}}_{\jmath\kappa}$ be a binary parameter to denote if the risk assessment procedure has assigned the κ-th security level requirement to the $\jmath$-th ED. Next, we define a set of parameters to identify the geographical positioning of CN and of ED. In this respect the optimization problem uses two-dimensional coordinates involving $(x^{\mathcal{C}}_{\imath}, y^{\mathcal{C}}_{\imath})$ for CN and $(x^{\mathcal{D}}_{\jmath}, y^{\mathcal{D}}_{\jmath})$ for ED. Furthermore, since each CN has a limited processing capability, we define the link capacity parameter as $\zeta^{\mathcal{C}}_{\imath}$ and $\xi^{\mathcal{D}}_{\jmath}$ as the traffic demand of the connected ED. Depending on the values of $\xi^{\mathcal{D}}_{\jmath}$, the parameter $\zeta^{\mathcal{C}}_{\imath}$ indirectly defines the number of ED that can be connected to the $\imath$-th CN. Lastly, we define the variables for the connection of each ED to a CN. More precisely, we define the binary variable $\nu_{\imath\jmath}$ with value 1 if ED $\jmath$ connects to CN $\imath$. Essentially, the network design problem will identify the values for this variable such as to minimize the objective function. In practice, the objective of network design is to efficiently connect the ED to the CN, while taking into account the security requirements and the traffic demands. The overall installation cost and later on the operational costs depend on the distance between nodes and the required security levels. Furthermore, shorter distances can also reduce the communication delays. Therefore, the objective of the ILP problem is to minimize the Euclidean distances in such a way to connect each ED to the closest CN:

$$min\left(\sum_{\imath\in\mathcal{C}}\sum_{\jmath\in\mathcal{D}}\left[(x^{\mathcal{C}}_{\imath} - x^{\mathcal{D}}_{\jmath})^2 + (y^{\mathcal{C}}_{\imath} - y^{\mathcal{D}}_{\jmath})^2\right]\cdot\nu_{\imath\jmath}\right), \tag{6}$$

which is subject to the following constraints:

$$\sum_{\imath\in\mathcal{C}}\nu_{\imath\jmath} = 1, \forall\jmath\in\mathcal{D}, \tag{7}$$

$$s^{\mathcal{D}}_{\jmath\kappa}\cdot\nu_{\imath\jmath} \leq s^{\mathcal{C}}_{\imath\kappa}, \forall\imath\in\mathcal{C}, \jmath\in\mathcal{D}, \kappa\in\mathcal{S}, \tag{8}$$

$$\sum_{\jmath\in\mathcal{D}}\xi^{\mathcal{D}}_{\jmath}\cdot\nu_{\imath\jmath} \leq \zeta^{\mathcal{C}}_{\imath}, \forall\imath\in\mathcal{C}, \tag{9}$$

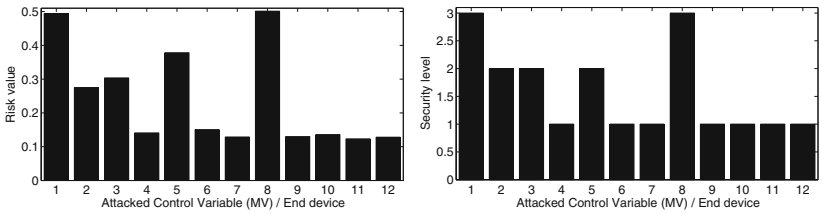
where Eq. (7) enforces that each ED is connected to a single CN. Constraint (8) enforces the connection of end devices to the concentrator nodes that support the required security level. Finally, constraint (9) ensures that the CN processing capacity is not exceeded.

5 Experimental Results

In this section we first apply the risk assessment technique to identify the risk levels of cyber assets. The risk levels are then applied in the evaluation of the network optimization problem, which is implemented and tested in AIMMS [1] by using the CPLEX solver. In the first instance we adopt the Tennessee Eastman chemical process (TEP) model [6]. Then, we validate the developed methodology by using the IEEE 14-bus electricity grid model enriched with control loops specific to real-world power systems, e.g., Power System Stabilizer (PSS), Automatic Voltage Regulators (AVR), Turbine Governors (TG), secondary voltage regulators including Cluster Controllers (CC), and Central Area Controllers (CAC). To demonstrate the scalability of the proposed techniques we perform the risk assessment and the network design on the IEEE 300-bus test system. The attack scenarios employed in the following experiments include bias injection and replay.

5.1 Results on the TEP

This case study considers four attack types and an attack probability defined by $p_\iota = [0.4, 0.3, 0.1, 0.2]$. In detail, the first attack type involves a 15% bias injection on control variables, with a duration interval of 0.1 h. In contrast, the second attack scenario injects a 60% bias value to each variable. The last two attack types are replay attacks, one with a duration of 4 h ($\epsilon = 4$ h) and the second on with a duration of $\epsilon = 3$ h. By knowing the probability values and the severity matrix resulted from the CAIA technique, we determine the final risk values for each investigated control variable or end device. For the case of the TEP, the risk values for devices are given in Fig. 1(a). Additionally, Fig. 1(b) presents the results of grouping the pure risk values using hierarchical clustering in 3 groups of security levels. Here, the usage of single linkage hierarchical clustering is needed to effectively categorize the end devices in a predefined number of security level groups. As the figure shows, the devices denoted by 1 and 8 have been assigned a higher security level, meaning that these devices require the most secure communication channels. In contrast, devices 4, 6, 7, 9, 10, 11 and 12



(a) (b)

Fig. 1. Risk assessment outcome: (a) Pure risk values; and (b) Risk values enforced in 3 security level groups.

have been associated with low security level requirements, meaning that short-term cyber attacks performed on these devices won't critically affect the normal operation of TEP. Finally, in the network design phase the security requirement parameter of $s_{j\kappa}^{\mathcal{D}}$ is initialized based on device groups, as depicted in Fig. 1(b).

In the following, we use the proposed network design approach in accordance with the above resulted risk values of the end devices. For network design experiments we assume five CN in five different locations of the plant. The end devices corresponding to control variables and the end devices corresponding to observed variables will connect to these five CN. However, in practice, CN are placed by considering the physical areas delimited by the experts or by the need of CN to cover the security and performance requirements of ED. Later on in this section multiple experiments with various CN are performed to identify the optimal number of CN in the case of the TEP. Let us first analyze the connection layout in Fig. 2(a). Security levels are illustrated with simple, double and triple symbol outlines in the case of CN, and simple outlined circle, double outlined circles and septagons in the case of ED, respectively. Furthermore, each node is placed as specified by the position parameters. As it is shown in Fig. 2(a) the overall connection distance is minimized in the presence of security and link capacity constraints. Subsequently, by refining the security level parameters of CN, and rerunning the network design framework, a different connection graph is

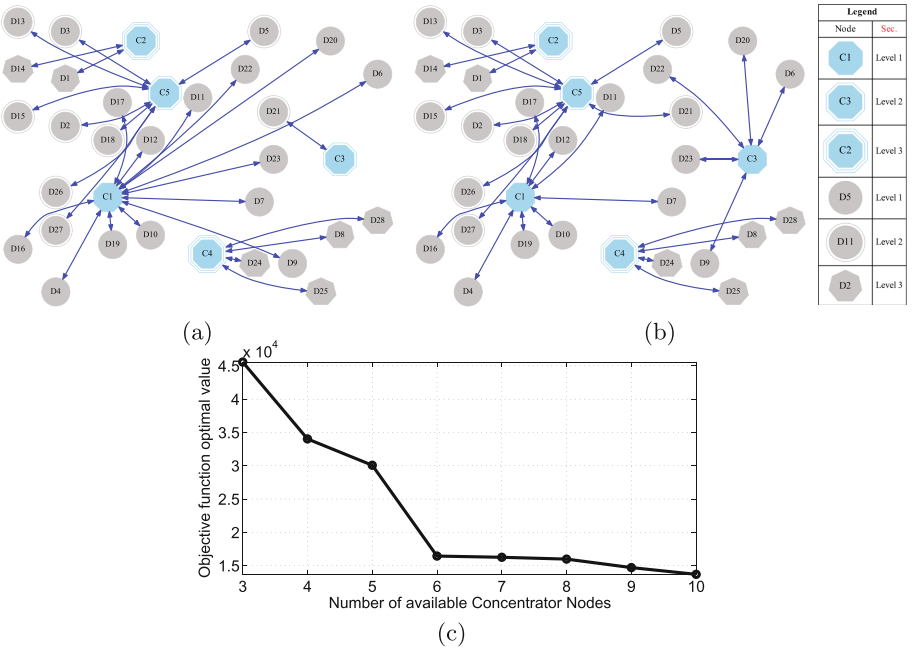


Fig. 2. Resulted network layout: (a) with initial parameters; (b) with C3's security level changed to 1 and (c) objective function's optimal values for different number of CN.

obtained in Fig. 2(b). The changing of C3's security level from 2 (mid-level) to a low level, radically changes the optimal connection layout as well. Therefore, the placement of CN heavily influences the network topology. This fact is illustrated through a series of experiments performed for different number of CN, while the rest of the parameters remained unchanged. Accordingly, the results shown in Fig. 2(c) express the final value of the objective function in contrast with the number of CN. Finally, we notice in Fig. 2(c) the steep decrease of the objective function's value, which indicates that 6 should be the optimal number of CN used for the network design (after 6 the cost function decreases slowly). In contrast with these results, practical situations may include additional restrictions in placing CN in some special areas, e.g., hazardous areas.

5.2 Results on the IEEE 14-Bus Electricity Grid

Making a step further, we show the application of the developed risk assessment technique on the 14-bus electricity grid model. Even though the grid comprises a slightly different architecture from that of the TEP's, the application of CAIA and of risk assessment remains mostly the same. First, we identify the assessed devices corresponding to each substation of the power grid. In this study it is considered that each substation is represented by a cyber device, which is part of the grid's SCADA network. Overall, four attack scenarios have been defined for the risk assessment. These are aimed to represent the main cyber security threats for the cyber realm. The first scenario implements cyber attacks that ultimately cause faults at substation levels. Assuming that proper load measurements and load control are key requirements in the stable operation of power grids, the second attack scenario induces load compensation disturbances. Considering the architecture of control loops, i.e., voltage controller modules localized at the substations including generator components, the third attack scenario launches integrity attacks against the IEC61850 protocol, which ensures the communication between AVR and other high level controllers, i.e., CC and CAC. Finally, the fourth attack scenario compromises remotely controlled line breaker devices to cause severe disruptions in the grid's structural stability. Figure 3 illustrates the changes in the output of risk assessment based on different attack scenarios and parameters. These results underline that different cyber attack vectors may yield different impact values and a different behavior of the physical process. Therefore, it is imperative that risk assessment to be conducted on multiple attack scenarios embracing a wide palette of parameters. Lastly, the final risk values for each end device are presented in Fig. 4(a). It is shown that substations 1, 2, 3 and 8 exhibit high risk values in the case of the four attack types. As a result, the hierarchical clustering approach (Fig. 4(c)) allocates to each available security level the appropriate device group. Since three security groups are expected, Fig. 4(b) illustrates the final security level of groups as an outcome of the risk assessment.

Lastly, we evaluate the developed methodology in the context of the IEEE 14-bus power grid. We assume a total of 14 ED, which need to be optimally connected to 5 CN. The results of the optimal security-aware configuration are

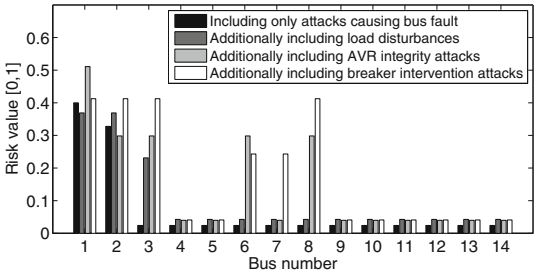


Fig. 3. Changes in risk assessment results by different attack scenarios.

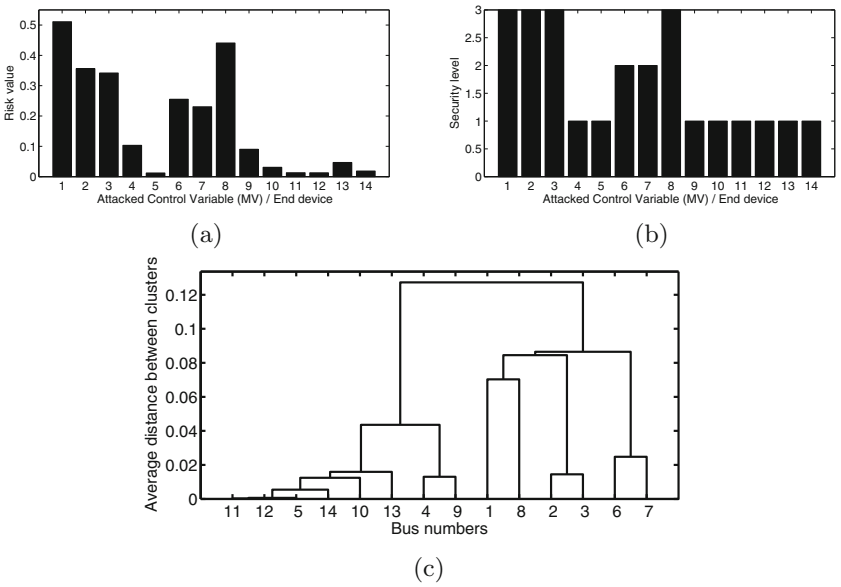
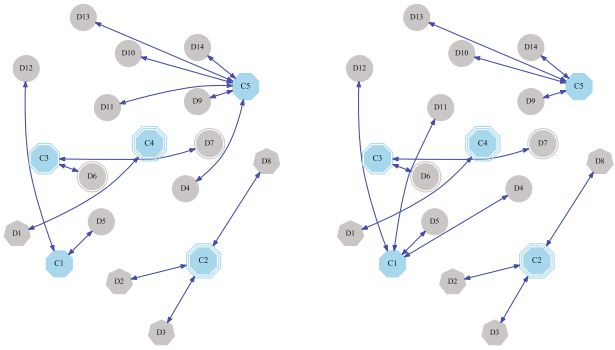


Fig. 4. Risk assessment outcome on the IEEE 14-bus grid model: (a) Pure risk values; (b) Risk values enforced in 3 groups with different security levels; and (c) Risk assessment dendrogram.

presented in Fig. 5(a). By changing the capacity of CN 5 in Fig. 5(b) we depict the changes in the output of the optimization problem.

5.3 Results on the IEEE 300-Bus Electricity Grid Model

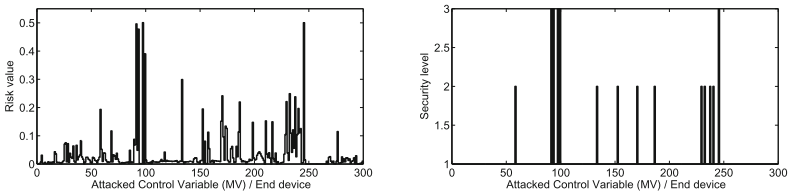
Finally, we show the scalability of the elaborated risk assessment procedure. The results in this section demonstrate its successful and representative application in the case of the large-scale IEEE 300-bus electricity grid model. To represent a wide variety of cyber attacks, the following experiments use two attack vectors, i.e., load disturbance attack and substation compromise. In the case of large-scale infrastructures we measure the isolation phenomenon, that is, attacks



(a) (b)

Fig. 5. Optimal network layout for the IEEE 14-bus electricity grid model: (a) unconstrained, and (b) constrained by the link capacity of C5.

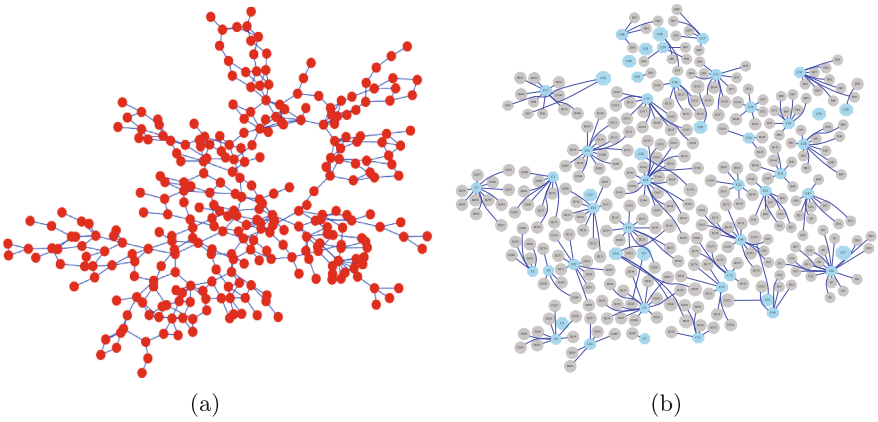
impacting a certain region of the grid. The calculated risk values are presented in Fig. 6, where we assume the same attack probability. However, depending on the expert's judgments the risk assessment can be changed to include different probabilities for each attack type. Figure 6(a) illustrates the peaks in risk values in the proximity of substations 100 and 250. This means that these devices have a greater impact on the overall operational stability of the grid. Accordingly, Fig. 6(b) groups the substations in three security level groups.



(a) (b)

Fig. 6. Risk assessment outcome on the IEEE 300-bus grid model: (a) Pure risk values; and (b) Risk values enforced in 3 security groups.

In accordance with the size of the model, the network design problem assumes the presence of 300 ED. In the first step the security levels of the CN are assigned proportionally according to the risk assessment results and the designated security requirements of ED (see Sect. 5.3). For illustration purposes we assume a scenario comprising of 50 CN. The connection diagram of Fig. 7(b) includes 36 low-level security CN, 10 mid-level security CN and 4 high-level security CN. Figure 7(a) represents the electrical topology, while Fig. 7(b) shows the final connection of ED to CN. The network layout accounts for the geographical distances



(a) (b)

Fig. 7. Optimal network layout for the IEEE 300-bus power grid model: (a) electrical topology [11], and (b) communication infrastructure with 50 CN.

between the nodes, but also embraces their security requirements, as delivered by the risk assessment methodology.

6 Conclusions

We developed a methodology for the optimal design of industrial networks. The approach embraces a risk assessment technique and an optimization problem to minimize connection distances, while enforcing security and capacity requirements. The experimental results revealed the importance of security-aware network design. As future work we intend to build a specialized software to assist engineers in designing optimal ICS networks.

Acknowledgment. This work was supported by a Marie Curie FP7 Integration Grant within the 7th European Union Framework Programme (Grant no. PCIG14-GA-2013-631128).

References

1. AIMMS: Advanced Interactive Multidimensional Modeling System (2015). http://www.aimms.com/aimms/. Accessed May 2016
2. Bilis, E., Kroger, W., Nan, C.: Performance of electric power systems under physical malicious attacks. IEEE Syst. J. **7**(4), 854–865 (2013)
3. Carro-Calvo, L., Salcedo-Sanz, S., Portilla-Figueras, J.A., Ortiz-Garca, E.: A genetic algorithm with switch-device encoding for optimal partition of switched industrial Ethernet networks. J. Netw. Comput. Appl. **33**(4), 375–382 (2010)
4. Chen, T., Abu-Nimeh, S.: Lessons from Stuxnet. Computer **44**(4), 91–93 (2011)
5. CrySiS Lab: sKyWIper (a.k.a. flame a.k.a. flamer): a complex malware for targeted attacks, May 2012

6. Downs, J.J., Vogel, E.F.: A plant-wide industrial process control problem. Comput. Chem. Eng. **17**(3), 245–255 (1993)
7. Ford, D.N.: A behavioral approach to feedback loop dominance analysis. Syst. Dyn. Rev. **15**(1), 3–36 (1999)
8. Genge, B., Haller, P., Kiss, I.: Cyber-security-aware network design of industrial control systems. IEEE Syst. J. **11**(3), 1373–1384 (2015)
9. Genge, B., Siaterlis, C.: Physical process resilience-aware network design for SCADA systems. Comput. Electr. Eng. **40**(1), 142–157 (2014)
10. Genge, B., Kiss, I., Haller, P.: A system dynamics approach for assessing the impact of cyber attacks on critical infrastructures. IJCIP **10**, 3–17 (2015)
11. Hines, P., Blumsack, S., Cotilla Sanchez, E., Barrows, C.: The topological and electrical structure of power grids. In: 2010 43rd Hawaii International Conference on System Sciences (HICSS), pp. 1–10, January 2010
12. Kiss, I., Genge, B., Haller, P.: Behavior-based critical cyber asset identification in Process Control Systems under Cyber Attacks. In: 16th Carpathian Control Conference (ICCC), pp. 196–201, May 2015
13. Kundur, D., Feng, X., Liu, S., Zourntos, T., Butler-Purry, K.: Towards a framework for cyber attack impact analysis of the electric smart grid. In: First SmartGridComm, pp. 244–249, October 2010
14. Nardo, M., Saisana, M., Saltelli, A., Tarantola, S., Hoffman, A., Giovannini, E.: Handbook on Constructing Composite Indicators. OECD Publishing, Paris (2005)
15. Sandberg, H., Amin, S., Johansson, K.: Cyberphysical security in networked control systems: an introduction to the issue. IEEE Control Syst. **35**(1), 20–23 (2015)
16. Sgouras, K., Birda, A., Labridis, D.: Cyber attack impact on critical smart grid infrastructures. In: 2014 IEEE PES Innovative Smart Grid Technologies Conference (ISGT), pp. 1–5, February 2014
17. Sridhar, S., Govindarasu, M.: Model-based attack detection and mitigation for automatic generation control. IEEE Trans. Smart Grid **5**(2), 580–591 (2014)
18. Symantec: Dragonfly: cyberespionage attacks against energy suppliers. Technical report (2014)
19. Zhang, L., Lampe, M., Wang, Z.: A hybrid genetic algorithm to optimize device allocation in industrial ethernet networks with real-time constraints. J. Zhejiang Univ. Sci. C **12**(12), 965–975 (2011)
20. Zhang, L., Lampe, M., Wang, Z.: Multi-objective topology design of industrial ethernet networks. Frequenz **66**(5–6), 159–165 (2012)

Effective Defence Against Zero-Day Exploits Using Bayesian Networks

Tingting Li(✉) and Chris Hankin

Institute for Security Science and Technology,
Imperial College London, London, UK
{tingting.li,c.hankin}@imperial.ac.uk

Abstract. Industrial Control Systems (ICS) play a crucial role in controlling industrial processes. Unlike conventional IT systems or networks, cyber attacks against ICS can cause destructive physical damage. Zero-day exploits (i.e. unknown exploits) have demonstrated their essential contributions to causing such damage by Stuxnet. In this work, we investigate the possibility of improving the tolerance of a system against zero-day attacks by defending against known weaknesses of the system. We first propose a metric to measure the system tolerance against zero-day attacks, which is the minimum effort required by zero-day exploits to compromise a system. We then apply this metric to evaluate different defensive plans to decide the most effective one in maximising the system tolerance against zero-day attacks. A case study about ICS security management is demonstrated in this paper.

1 Introduction

Cyber security of industrial control systems has increasingly become a severe and urgent problem, owing to the wide use of insecure-by-design legacy systems in ICS, and the potential physical damage of breached ICS to infrastructures, environment and even human health [19]. The rapid integration of ICS with modern ICT technology has further intensified the problem. Increasing attention has been drawn to this issue from various sectors such as industry, government and academia. Whilst most ICS vulnerabilities inherited from IT systems are well studied and can be defended by conventional security controls, very little effort can be made to combat zero-day exploits, because they are often unknown to the vendor and hence there is no patch available to fix them. One of the most famous cyber attacks against ICS is Stuxnet disclosed in 2010 [4], which was distributed by an infected USB flash drive, propagated across the corporate network, and eventually compromised the PLCs to disrupt the operation of industrial plants. Four zero-day vulnerabilities played crucial roles in gaining access to targets and propagating the malware. Until September 2010, there were about 100,000 hosts over 155 countries infected by Stuxnet [4]. The threat from zero-day exploits is still on the rise. In 2014, 245 incidents were reported to ICS-CERT and 38% of these incidents were identified as having an "unknown access vector", and ICS-CERT specifically mentioned the exploitation of zero-day vulnerabilities as one

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 123–136, 2017.
https://doi.org/10.1007/978-3-319-71368-7_11

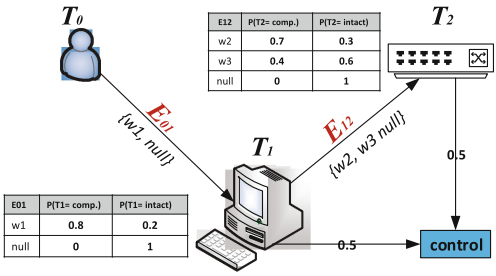
of the methods used by attackers [11]. In July 2015, a zero-day vulnerability in Adobe Flash Player has been acknowledged by ICS-CERT, which is able to gain access to critical infrastructure networks via spear-phishing emails [10]. Later in August, ICS-CERT continuously released six advisories and six alerts about zero-day vulnerabilities on Siemens SIMATIC S7-1200 CPUs, Schneider Electric DTM, Rockwell Automation PLCs, etc.

It is extremely difficult to detect and defend against zero-day exploits. Sophisticated hackers are able to discover zero-day exploits before the vendors become aware of them. Since it is difficult to directly stop zero-day attacks, we consider the problem from a novel perspective, by seeking a way to make ICS sufficiently robust against zero-day attacks. We are able to reduce the risk of potential zero-day exploits to an acceptable level by strategically defending against the known attack vectors.

A typical APT attack targeting ICS has to exploit a chain of vulnerabilities at different hosts to eventually breach the control devices (e.g. PLCs). The involved exploits use either known or zero-day vulnerabilities to propagate across the network. Whilst we can hardly defend against the exploitation of zero-day vulnerabilities, we can alternatively deploy effective defences against the known vulnerabilities such that the risk of the whole attack chain being exploited can be overall reduced. A key attribute "exploitability" of weaknesses is borrowed from CWE [2] to reflect the sophistication of a zero-day weakness and the required attacking effort. Weaknesses with higher exploitability are likely to cause higher risk for the system. With regard to an acceptable level of risk, we define the tolerance against a zero-day weakness by the minimal required exploitability of the weakness to cause the system risk exceed the acceptable level. By using Bayesian Networks, we can prove that defending against known weaknesses is able to increase the tolerance, and find out the defence that maximizes the tolerance.

We express an acceptable level of risk by conceptual safety-related requirements of ICS [14] such as the integrity of monitoring data, the availability of control and reliable communication. Cyber attacks targeting ICS might violate such requirements to a certain degree. By modelling these requirements as nodes of a Bayesian Network, we can define the acceptable risk by the severity of a requirement being violated. Next we use a simple example to further illustrate the motivation of this work.

Figure 1 shows a Stuxnet-like attack path launched by an adversary T_0 to gain access to a workstation T_1 (by exploiting w_1), which is then used as a foothold to further compromise the PLC T_2 (by exploiting w_2 or w_3). Exploitability of each known weakness is given in the tables. Both T_1 and T_2 equally contribute to the satisfaction of the requirement about available control. The requirement about control is indicated by a box in the bottom-right of Fig. 1. We set the acceptable risk as the probability of the control requirement being violated must be lower than 27%. We also assume there is a zero-day weakness at T_1 or T_2. By using our approach based on Bayesian networks, we obtained results in the right table of Fig. 1. Without any control deployed, either a zero-day exploit at T_1 with 34%



	tole. of T_1	tole. of T_2
no control	34%	10%
c_1 against w_1	42%	70%
c_2 against w_2	36%	17%
c_3 against w_3	36%	16%

Fig. 1. Left: A simple Stuxnet-like attack scenario. **Right**: Tolerance improvement

exploitability or a zero-day exploit at T_2 with 10% exploitability is sufficient to bring the risk beyond the acceptable level. It can be found that by deploying different controls, the tolerance against zero-day exploits is generally increased. Some increments are rather small as in this demo example the effectiveness of controls is set to 10% only. Different controls bring about different improvement on the tolerances depending on their effectiveness, the exploitabilities of their combating weaknesses and their influence coverage over targets. Our approach is able to take all these factors into account and find out the most effective defence against zero-day attacks.

2 Modelling and Problem Representation

In this section we formally use Bayesian Networks (BN) to model ICS-targeted attacks with zero-day exploits involved and evaluate the risk. A discrete random variable is captured by a chance node in BN with a finite set of mutually exclusive states and a conditional probability distribution over the states. We further defined three types of chance nodes for different purposes: (i) *target nodes* indicate valuable assets in ICS with a set of known and zero-day weaknesses, (ii) *attack nodes* captures available attack methods between a pair of targets, and (iii) *requirement nodes* are designed to model particular objectives for evaluation. A *Bayesian Risk Network* is established based on the three types of nodes, where complete attack paths are modelled by target and attack nodes, and the damage of successful attacks are evaluated against requirement nodes.

Definition 1. *Let $\mathcal{T}$ be a set of* ***target nodes*** *$\mathcal{T} = \{T_1, \ldots, T_n\}$. Parent nodes of a target T_x is denoted by $T'_x \in pa(T_x)$. The domain states of a target node is $\Omega(T_x) = \{c, i\}$, representing the target being compromised or intact respectively. Let $\mathcal{R} = \{R_1, \ldots, R_m\}$ be a set of* ***requirement nodes*** *and the domain states of a requirement node is $\Omega(R_x) = \{c, v\}$, indicating respectively the requirement being complied or violated. Parents of a requirement node could be target nodes or other requirement nodes $pa(R_x) \subseteq \mathcal{T} \cup \mathcal{R}$.*

Target nodes represent valuable assets where zero-day weaknesses might be exploited in addition to the known weaknesses. Requirement nodes capture

safety-related objectives, which are used to evaluate the impact of cyber attacks on the system safety. Detailed modelling and reasoning about these requirements can be found in [14].

Definition 2. *Let $W = \{w_1, \ldots, w_m\}$ be a set of weaknesses, $\omega : \mathcal{T} \times \mathcal{T} \to 2^W$ gives possible weaknesses that can be exploited from one target to another. Let $\mathcal{E} = \{E_{T_1'T_1}, \ldots, E_{T_n'T_n}\}$ be a set of **attack nodes** connecting a target and its parents. The domain states $\Omega(E_{T_iT_j}) = \omega(T_i, T_j) \cup \{null\}$, including all weaknesses on T_j that can be exploited from T_i, or none of them is exploited (i.e. null). The **exploitability** e_{w_x} of a weakness w_x is the likelihood of w_x being successfully exploited.*

Definition 3. *Let w_z be a **zero-day exploit** with uncertain exploitability $e_{w_z} \in [0, 1]$, $w_z \in \Omega(E_{T_x'T_x})$ indicates there is a zero-day exploit at the target T_x.*

Unlike Bayesian Attack Graphs (BAG) [18] that are constructed based on states and attributes, we build a Bayesian network at the level of assets and model multiple weaknesses between a pair of assets by a single attack node, rather than multiple attack edges. Each attack node hence becomes a decision-making point for attackers to choose a (known or zero-day) weakness to proceed. Such Bayesian networks enable us to model zero-day exploits without knowing details about them (e.g. pre-requisites or post-conditions), but concentrate on analysing the risk caused by zero-day exploits.

Definition 4. *Let $\mathcal{C} = \{c_1, \ldots, c_k\}$ be a set of **defence controls** and $d(c_x) \in 2^W$ be weaknesses that can be defended by c_x. If $w_i \in d(c_j)$, then by deploying c_j, the exploitability of w_i is scaled by $\varepsilon \in [0, 1]$, where ε is the effectiveness of c_j.*

A defence control is able to reduce the exploitability of its combating weaknesses. If ε is set to 50%, then applying c_j reduces the exploitability of $w_i \in d(c_j)$ by 50%.

Definition 5. *Let $\mathcal{B} = \langle \mathcal{N}, \mathcal{P}_T, \mathcal{P}_E, \mathcal{P}_R, P_{T_0} \rangle$ be a **Bayesian Risk Network**, where*

- *$\mathcal{N} = \mathcal{T} \cup \mathcal{E} \cup \mathcal{R}$, including target nodes, attack nodes and requirement nodes.*
- *$\mathcal{P}_T = \{P_{T_1}, \ldots, P_{T_n}\}$ includes conditional probabilities of all non-root target nodes given their parents such that P_{T_x} denotes $P(T_x | \bigcup_{T_x' \in pa(T_x)} E_{T_x'T_x})$, where $P(T_x | \bigcup_{T_x' \in pa(T_x)} E_{T_x'T_x}) = 1 - \prod_{T_x' \in pa(T_x)} (1 - P(T_x | E_{T_x'T_x}))$ by noisy-OR operator [17]. $P(T_x | E_{T_x'T_x})$ is the probability of T_x given the weakness used at $E_{T_x'T_x}$.*
- *$\mathcal{P}_E = \{P_{E_{T_1'T_1}}, \ldots, P_{E_{T_n'T_n}}\}$ includes conditional probability distribution for all attack nodes such that $P_{E_{T_x'T_x}}$ denotes $P(E_{T_x'T_x} | T_x')$.*
- *$\mathcal{P}_R = \{P_{R_1}, \ldots, P_{R_n}\}$ includes decomposition of all requirement nodes such that P_{R_x} denotes $P(R_x | pa(R_x))$, where $P(R_x | pa(R_x)) = \Sigma_{R_x' \in pa(R_x)} P(R_x | R_x')$, and $P(R_x | R_x')$ is the assigned proportion of R_x' in R_x.*
- *P_{T_0} is the prior probability distribution of the root node T_0.*

$P(T_x)$ is the unconditional probability of $T_x \in \mathcal{T}$, which can be obtained by:

$$P(T_x) = \begin{cases} \sum_{E_{T'_x T_x}} P_{T_x} \sum_{T'_x} P_{E_{T'_x T_x}} P(T'_x) & \text{if } w_z \notin \Omega(E_{T'_x T_x}) \\ \sum_{E_{T'_x T_x}} P_{T_x} \sum_{T'_x} P_{E_{T'_x T_x}} P(T'_x) + P(T_x | E_{T'_x T_x} = w_z) \sum_{T'_x} P_{E_{T'_x T_x}} P(T'_x) & \text{otherwise} \end{cases}$$

$P(T_x)$ is obtained by its parent node $P(T'_x)$ recursively until it hits the root T_0 whose probability distribution is known. $\sum_{E_{T'_x T_x}}$ denotes $E_{T'_x T_x}$ is marginalized. P_{T_x}, P_{R_x} and $P_{E_{T'_x T_x}}$ are given by $\mathcal{P}_T$, $\mathcal{P}_R$ and $\mathcal{P}_E$ respectively. $P(T_x | E_{T'_x T_x} = w_z)$ equals to the uncertain exploitability of the zero-day exploit w_z at T_x.

$P(R_x)$ denotes the unconditional probability of $R_x \in \mathcal{R}$ given its parents R'_x and $P(R_x) = \sum_{R'_x} P_{R_x} \prod_{R'_x \in pa(R_x)} P(R'_x)$, where $pa(R_x)$ are marginally independent.

$\mathcal{P}_T$ is given by conditional probability tables (CPT) for each target node. Each entry of the CPT is the probability of a target being compromised (resp. intact) when a weakness is chosen, which equals to the exploitability e_{w_x} (resp. $1 - e_{w_x}$) of the chosen weakness. Such a CPT is shown in the upper part of Fig. 2(a). When w_1 is used, the chance of T_1 being compromised $P(T_1 = c)$ is 0.8, equivalent to the exploitability of w_1. When a target node has multiple parent nodes, noisy-OR operator [17] is applied to calculate the joint probability of parents, as in BAG [15,18]. $P_{E_{T'_x T_x}}$ decides the chance of each weakness being used. Here we assume attackers choose uniformly from available weaknesses. As given in the lower CPT in Fig. 2(a), when the parent target is intact ($T_2 = i$), no attack can be continued towards the next target (i.e. *null* is the only choice). When the parent target is compromised, the probability is equally distributed over the available weaknesses $\Omega(E_{T_2 T_4}) = \{w_4, w_5, null\}$. If a zero-day exploit exists at T_x, the extra contribution of w_z is added to $P(T_x)$. We make the same assumption as in [8,18] that such *Bayesian Risk Networks* are directed acyclic graphs.

Definition 6. *A Bayesian Risk Network $\mathcal{B}$ is constructed for a given system. The **tolerance against zero-day attacks** of the system is represented by (κ, Z), where*

- *κ is a defined acceptable level of risk, expressed by $\kappa := P(N_a = s) \leqslant L$, where the probability of a fixed node $N_a \in \mathcal{T} \cup \mathcal{R}$ being at a particular state $s \in \Omega(N_a)$ is used to define the risk and L is the upper bound of $P(N_a = s)$.*
- *$Z := \langle z_1, \ldots, z_n \rangle$ is a tolerance tuple with each element corresponding to the tolerance against a zero-day exploit at each target node. Thus the tolerance $z_i \in Z$ against a zero-day at an arbitrary target $T_i \in \mathcal{T}$ is obtained by:*

$$z_i = \underset{P(T_i | E_{T'_i T_i} = w_z)}{\operatorname{argmax}} \kappa := P(N_a = s) \leqslant L$$

$P(T_i|E_{T_i'T_i} = w_z)$ equals to the exploitability of the zero-day exploit w_z at T_i and z_i is the maximum exploitability of w_z subject to κ. $P(N_a)$ is the unconditional probability of a target or requirement node, which can be obtained by Definition 5.

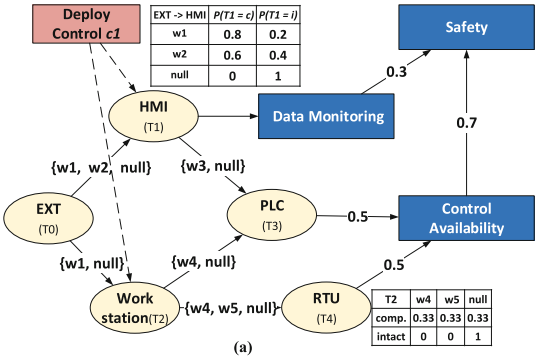
We select a particular node N_a to define the risk κ, which could be a valuable target node or a critical requirement. Thus κ is defined by the likelihood of N_a being compromised or violated, e.g. the likelihood of a requirement being violated must be less than 30%. The presence of a zero-day exploit at any target is likely to increase the likelihood as its exploitability increases. Thus, we define the tolerance by the minimum required exploitability of a zero-day exploit at each target to violate κ, or alternatively the maximum exploitability of a zero-day exploit the system can tolerate subject to κ.

3 Case Study and Results

In this section, we present a hypothetical example to demonstrate our approach of finding effective defence against zero-day exploits. We start with the configuration of the example and then discuss the results by applying different defence controls.

3.1 Case Study Settings

A simple network is constructed in Fig. 2(a) consisting of common types of assets in ICS – a *HMI*, a *workstation*, a *PLC* and a *RTU*. The four assets are modelled as four target nodes $\mathcal{T} = \{T_1, T_2, T_3, T_4\}$ of a Bayesian network. A special node EXT(denoted by T_0) represents the external environment of the network. We also select five common weaknesses $\{w_1, w_2, w_3, w_4, w_5\}$ from the *ICS Top 10 Threats and Countermeasures* [1] and *Common Cybersecurity Vulnerabilities in ICS* [3]. These weaknesses are enumerated in Fig. 2(b), which are attached to relevant attack nodes between a pair of targets. Exploiting different weaknesses yields different consequences depending on the exploitability of the chosen weakness. For instance, in order to compromise T_1, an attacker can choose to exploit w_1 or w_2, or keep hiding *null*. Currently we assume that attackers choose relevant weaknesses uniformly at each attack node. The chance of exploiting a node successfully is given in the relevant CPTs. An example CPT of T_1 is shown in Fig. 2(a). When w_1 is chosen, the attacker has a priori 80% chance to compromise T_1. The exploitabilities of weaknesses are essential to construct such CPTs. In this case study, we consistently convert different levels of the CWE attribute "*Likelihood of Exploit*" [2] and the metric "*Exploitability*" from [1] into certain values. Weaknesses that are identified as *"Very High"* by CWE or *"Easy to Exploit"* in [1] are set to 0.8; Weaknesses with *"High"* level of exploitability are set to 0.7 and *"Moderate"* weaknesses have exploitabilities of 0.6. Thus, we derive the rightmost column of the table in Fig. 2(b). The table in Fig. 2(c) lists a set of common defence controls [1] that are used in this case. We set a uniform effectiveness ε of all controls to 50%. Therefore the exploitability of w_1 becomes 0.4 after deploying c_1.



Weakness	Description	Location	Exploitability
w1	Internet Malware	T1, T2	0.8
w2	Removable Drive Malware	T1	0.6
w3	DoS Attacks	T3	0.7
w4	Buffer overflow	T3, T4	0.8
w5	Man-in-the-middle	T4	0.6

(b)

Control	Control Descriptions	Combat
c1	anti-virus software	w1
c2	removable media check/disable	w2
c3	IDS and Firewalls	w3
c4	data validation & software security analysis	w4
c5	fine-grained access controls & integrity check	w5

(c)

Fig. 2. (a) Network; (b) Selective common weaknesses; (c) Selective common controls.

To model the cyber-physical effects of potential exploits, we consider three key requirements in the example. The target node T_1 has direct and dominated influence on the requirement about *data monitoring*. As two core field controllers, the *PLC* and *RTU* equally contribute to satisfying the requirement about *control availability*. The overall *safety* jointly relies on data monitoring (30%) and control availability (70%). We make this particular configuration to reflect the common requirement of ICS that system availability generally outweighs the other aspects [13]. In the Fig. 2(a), we use dashed lines to indicate the impact of deploying c_1 against w_1 at the target T_1 and T_2.

We construct the corresponding *Bayesian Risk Network* in Fig. 3, where the unconditional probability distribution over possible states of each node is computed. The node T_0 denotes the untrusted external environment where attackers can launch any attacks, and thus the probability of its compromised state is 100%. Figure 3 simulates the example ICS *without* any control deployed or any zero-day exploits, and the chance of the safety being violated is about 30.94%. In the following parts of the paper, the *risk* of the system is referred to the probability of the safety requirement being violated $P(R_{\text{safety}} = v)$.

In the next sections, we add zero-day exploits to each target and deploy different controls, in order to evaluate the impact of controls on the system tolerance against those zero-day exploits. We first present the results with an individual control in Sect. 3.2 and further discuss the results with multiple controls deployed in Sect. 3.3.

3.2 Results – Deploying a Single Control

We run four trials of the experiment in each of which a zero-day exploit w_z is added to each target. In each trial, different defence controls are individually deployed and the updated risks over scaled exploitabilities of the zero-day exploit (e.g. 20%, 40%, 60% and 80%) are computed. In the four charts of Fig. 4, the upper curve with markers illustrates the trend of the risk with *none* control by varying exploitabilities of w_z. This curve is used as the baseline to evaluate the

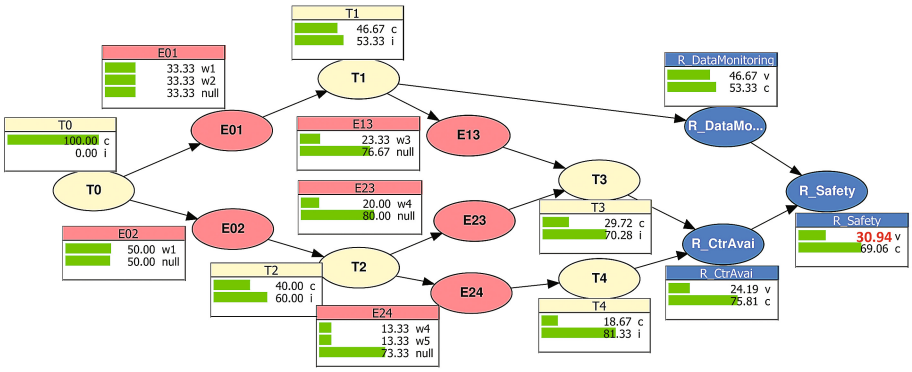


Fig. 3. Posterior risk distribution with no control deployed. (by *Hugin Lite* [9])

mitigated risk by deploying each control, which are indicated by the coloured bars respectively.

We discover that the existence of zero-day exploits w_z increases the risk of the system. As shown in Fig. 3, the *apriori* risk is about 30.94% without any zero-day exploit, which is raised to 34.23% with a w_z of 80% exploitability at T_1, and to 34.6% at T_2. It is worth noting that the risk caused by zero-day exploits starts to *exceed* the risk without zero-day exploits only when the zero-day exploit reaches a certain exploitability, which is seemingly counter-intuitive. At T_1, the risk exceeds the apriori risk (30.94%) when the w_z reaches a higher exploitability (49%). It is because the known weaknesses w_1 and w_2 at T_1 already have rather high exploitabilities, the presence of low-level zero-day exploits would actually reduce the overall chance of T_1 being compromised as we assume the attacking methods are chosen uniformly. Therefore it is possible that the risk *with* zero-day exploits is lower than the risk *without* zero-day exploits when the zero-day exploits are at very low exploitabilities. However, since zero-day exploits can be hardly detected, their exploitabilities tend to be very high in reality. From Fig. 4, the zero-day exploit at T_2 is the most threatening one as it brings the greatest increment to the risk, while that at T_4 is the least threatening one. This is simply because T_2 influences more subsequent nodes than T_4.

It can be found that the control c_1 is the most effective one to reduce the risk such that the risk drops to 24.59% from 34.23% with a w_z of 80% exploitability at T_1, In the bottom-left chart of Fig. 4, we notice similar risk mitigation of c_3 and c_5 to combat the w_z at T_3. These two controls mainly target for w_3 at T_3 and w_5 at T_4 respectively. The similar mitigation is probably due to the symmetric positions of T_3 and T_4 in the network and their equal contribution to satisfy the control availability requirement.

The tolerance of the system against zero-day exploits has been improved by deploying controls. In the top-right chart of Fig. 4, at least a zero-day exploit with exploitability 31% is needed at T_2 to produce the risk 30%. With the help of c_2, a zero-day exploit with much higher exploitability 74% at T_2 is required to reach the same level of risk.

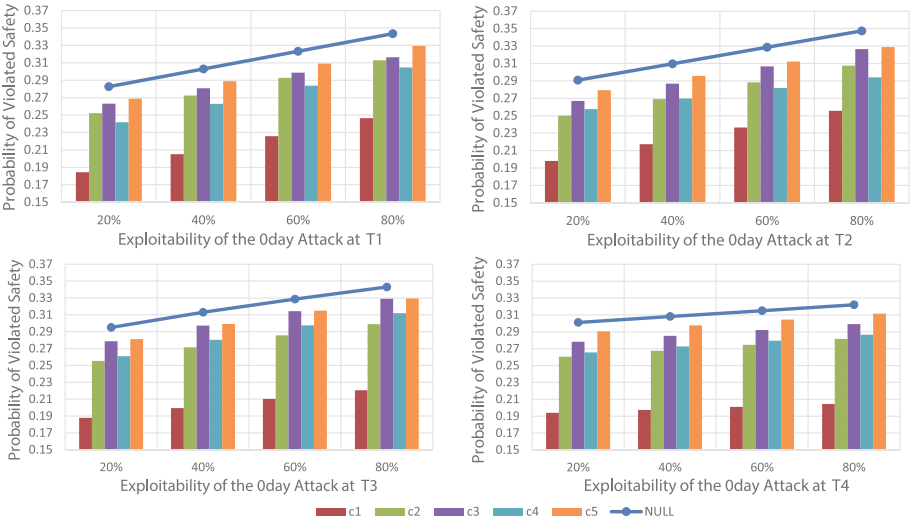


Fig. 4. Risk distribution with different controls on each target with an zero-day exploit (Color figure online)

3.3 Results – Deploying Combined Controls

A defence plan consists of multiple controls. There are five controls in the example with 2^5 combinations of them. We use bit vectors to represent including or excluding a control in a plan. For the controls $\{c_1, c_2, c_3, c_4, c_5\}$, a defence plan "10011" indicates to apply c_1, c_4 and c_5. We define $|d|$ as the number of controls included in a plan. Each row of Table 1 shows the result of deploying a defence plan, with the maximal risk $max(r)$ when the zero-day exploit at each target reaches its maximal exploitability 100%, and the mean risk reduction $\bar{\Delta}_x$ by deploying the plan. The mean risk reduction over the four targets is given by $\bar{\Delta}$. The acceptable risk κ is set to $P(R_{\text{safety}} = v) \leqslant 20\%$. The rightmost column shows the resulting tolerance against zero-day exploits. The symbol min indicates the risk already exceeds the acceptable level regardless of the existence of a zero-day exploit, while max denotes the system is fully tolerant to a zero-day exploit at the target, i.e. the acceptable level κ can never be violated even if the zero-day exploit reaches its maximal exploitability.

The first row 00000 with no control deployed is still used as the baseline. In terms of the risk reduction, 10000 is the most effective one when $|d| = 1$, while the plans 11000, 11010 and 11110 are the most effective choice if we can implement 2, 3 or 4 controls respectively. We discover that implementing more controls does not always produce stronger defence. For instance, deploying c_2, c_3 and c_5 (i.e. the plan 01101) has risk reduction 0.273, which is lower than the reduction 0.414 by deploying c_1 only. Each control combats different weaknesses that are distributed over different nodes. Defending against more widespread weaknesses would generally produce more risk reduction across the network. Besides,

Table 1. Results of Selective Defence Plans

Plan	T_1		T_2		T_3		T_4		$\bar{\Delta}$	Tolerance
	$max(r)$	$\bar{\Delta}_1$	$max(r)$	$\bar{\Delta}_2$	$max(r)$	$\bar{\Delta}_3$	$max(r)$	$\bar{\Delta}_4$		
00000	0.362	0	0.365	0	0.357	0	0.328	0	0	min, min, min, min
10000	0.267	0.097	0.274	0.092	0.231	0.113	0.208	0.112	**0.414**	0.36, 0.23, 0.44, 0.57
11000	0.236	0.128	0.234	0.132	0.183	0.156	0.166	0.153	0.569	**0.66, 0.65, max, max**
10001	0.260	0.104	0.258	0.102	0.224	0.120	0.202	0.117	0.443	0.43, 0.31, 0.56, 0.87
10100	0.239	0.118	0.258	0.109	0.219	0.126	0.189	0.130	0.483	0.57, 0.41, 0.66, max
10010	0.247	0.118	0.225	0.123	0.214	0.131	0.189	0.130	0.502	0.56, 0.59, 0.75, max
00101	0.319	0.037	0.324	0.038	0.329	0.030	0.295	0.033	0.138	min, min, min, min
11100	0.212	0.145	0.223	0.145	0.175	0.165	0.153	0.166	0.620	0.87, 0.77, max, max
01101	0.293	0.064	0.289	0.073	0.287	0.068	0.259	0.069	**0.273**	min, min, min, min
11010	0.215	0.149	0.184	0.165	0.166	0.174	0.147	0.172	0.660	0.86, max, max, max
01111	0.253	0.105	0.227	0.118	0.253	0.103	0.222	0.106	0.430	0.42, 0.50, 0.32, 0.37
11011	0.208	0.156	0.167	0.175	0.159	0.180	0.142	0.178	0.689	0.92, max, max, max
11110	0.192	0.166	0.173	0.177	0.158	0.182	0.134	0.185	0.710	max, max, max, max
11111	0.185	0.173	0.156	0.187	0.151	0.189	0.129	0.190	0.740	max, max, max, max

weaknesses near the attack origin (i.e. the node T_0 in this case) tend to have greater impact on the risk of all subsequent nodes, and hence applying defences against *earlier* attacks are relatively more effective. The control c_1 combats a common weakness w_1 at both T_1 and T_2, and w_1 provides the initial access to the system for the adversary to induce further attacks.

Looking at the tolerance against zero-day attacks, implementing no control 00000 is obviously one of the worst cases. The control c_1 yields a tolerance $\langle 0.36, 0.23, 0.44, 0.57\rangle$, indicating certain sophistication of each zero-day exploits is required to individually violate κ. Deploying c_1 and c_5 further enhances the tolerance to $\langle 0.43, 0.31, 0.56, 0.87\rangle$. 11000 makes the system be fully tolerant of a zero-day at T_4 or T_5 (least threatening ones). At least 11110 is needed for the system to be tolerant of a zero-day at any target.

Two radar charts are shown in Figs. 5 and 6 to provide an intuitive way to visualise the tolerance at the four different targets. The 100% coverage corresponds to the symbol *max* in tolerance tuples. Figure 5 shows that deploying more controls does not always guarantee a larger tolerance coverage. The defence plans with c_1 involved tend to be most effective ones in terms of risk reduction and tolerance coverage. 10100 is able to fully protect the system from the zero-day exploit at T_4 because c_1 defends both T_1 and T_2 (where all attacks have to pass through), and c_3 further defends T_3, which greatly limits the damage the zero-day at T_4 can cause to their subsequent sharing requirement node. The tolerance of four effective plans (in terms of $\bar{\Delta}$) is drawn in Fig. 6. The coverage against four targets are expanded at various rates. The zero-day exploit at T_4 seems to be the easiest one to be defended, while T_1 and T_2 are the most difficult ones. Three out of the four plans in Fig. 6 make the system immune from the zero-day exploit at T_4, but only 11010 can protect the system from the zero-day exploits at T_1 and T_2.

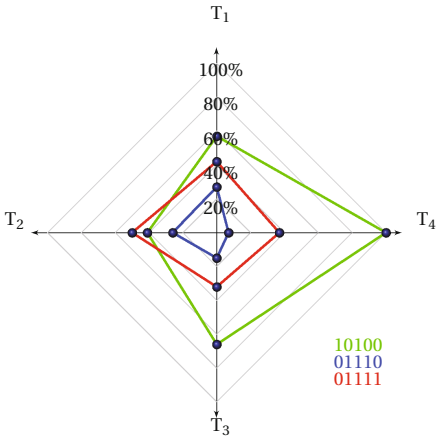


Fig. 5. Tolerance coverage on each target

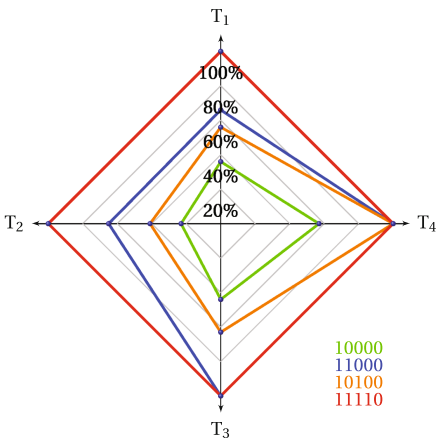


Fig. 6. Comparing plans of high $\bar{\Delta}$

4 Related Work

Bayesian Networks (BN) have been widely applied in complex systems modelling, risk assessment and diagnostics [22]. BN offer a graphical model to combine and capture complex causal relations between various factors. BN are also able to reason about possible effects and update beliefs in the light of emerging events (e.g. deploying controls), and finally produce decisions that are visible and auditable. *Bayesian Attack Graphs* (BAG) was introduced by Liu and Man [15] by combining attack paths and Bayesian inference methods for probabilistic analysis. Poolsappasit et al. [18] introduced static risk analysis and dynamic risk analysis based on BAG in order to find the most cost-efficient security plans. Muñoz-González et al. [16] further improved the work by providing an efficient probabilistic inference mechanism based on *Belief Propagation* and *Junction Tree* to compute unconditional probabilities. BN were used in [12] to study interdependencies between safety and security for CPS, where the main focus is on analysing the impact of different factors on safety and security. By contrast, we explicitly modelled all possible attack paths by exploiting a chain of known or unknown weaknesses, and evaluated the damage of such cyber attacks against key safety-related requirements.

Combating zero-day attacks has attracted an increasing attention. Wang et al. [20] present a novel security metric *k-zero day safety* to count the minimum number of zero-day vulnerabilities required for compromising network assets. The following work in [21] evaluated the robustness of networks against zero-day attacks in terms of network diversity. Particularly network diversity was formally defined as a security metric by the number of distinct resources, the least and average attacking effort. With regard to the most effective defence

for ICS, our work focused on the impact of deployed controls on mitigating the risk from zero-day attacks. Fielder et al. [7] compared three key decision making techniques (i.e. game theory, combinatorial optimisation and a hybrid of the two) to find effective defence for security managers. The work [5] provided a co-evolutionary agent-based simulation to find optimal defences for ICS, and then [6] considered the cost-effectiveness of defences in various zones of ICS.

5 Conclusion and Future Work

In this paper we studied the possibility of improving the tolerance of ICS against zero-day attacks by means of defending against known weaknesses. We first formally defined the tolerance as a metric by the minimum required exploitability of a zero-day exploit to bring the system into a critical state. Such a metric captures the required zero-day attacking effort, and hence higher tolerance indicates more effort should be invested by an adversary to discover a more sophisticated zero-day flaw. Tolerance against the zero-day exploits at different assets is diverse, depending on the topological position and known weaknesses of an asset. We further built a simulation based on Bayesian Networks to analyse the zero-day threat propagation across ICS. Attackers are able to choose a known or a zero-day (if there is one) weakness at each step, to propagate the risk from one target to the next. Depending on the exploitability of the chosen weakness and its previous exploited targets, the probability of success can be computed. A complete attack path needs to successfully exploit a chain of such weaknesses to reach the final valuable targets of ICS. Deploying security controls combating known weaknesses at each step could actually reduce the chance of the whole attack path being breached. In this case, higher exploitability of zero-day weaknesses is required to reach the same risk level, which means the tolerance of the system against zero-day exploits has been improved. Our approach is able to find the most effective combination of available defence controls to maximize the tolerance and the zero-day attacking effort. A case study about security management of ICS was also demonstrated in this paper.

There are several promising lines of research following this work: (i) we currently considered only the individual zero-day weakness at different targets, and we will explore the consequence of combining multiple zero-day exploits. (ii) intelligent adversarial models are needed to decide the likelihood of different attack paths. (iii) we can also efficiently capture a defensive control combating multiple weaknesses, in which case the exploitabilities of all these weaknesses would be reduced by applying the control. (iv) as addressed in [14], security controls might have negative impact on the other criteria of ICS. We will look for possible extensions to model those criteria into the simulation. The cost of deploying controls is also an essential factor to decide the most effective defence, which will be considered in our future work.

Acknowledgement. This work is funded by the EPSRC project RITICS: Trustworthy Industrial Control Systems (EP/L021013/1).

References

1. BSI: Industrial control system security top 10 threats and countermeasures 2014, March 2014. www.allianz-fuer-cybersicherheit.de/ACS/DE/_downloads/techniker/hardware/BSI-CS_005E.pdf
2. Christey, S., Glenn, R., et al.: Common weakness enumeration (2013)
3. U.S. Department of Homeland Security: Common cybersecurity vulnerabilities in industrial control systems (2011). www.ics-cert.us-cert.gov/sites/default/files/documents/DHS_Common_Cybersecurity_Vulnerabilities_ICS_20110523.pdf
4. Falliere, N., Murchu, L.O., Chien, E.: W32: Stuxnet dossier. White paper, Symantec Corp., Security Response 5 (2011)
5. Fielder, A., Li, T., Hankin, C.: Defense-in-depth vs. critical component defense for industrial control systems. In: Proceedings of the 4th International Symposium for ICS & SCADA Cyber Security Research. British Computer Society (2016)
6. Fielder, A., Li, T., Hankin, C.: Modelling cost-effectiveness of defenses in industrial control systems. In: Skavhaug, A., Guiochet, J., Bitsch, F. (eds.) SAFECOMP 2016. LNCS, vol. 9922, pp. 187–200. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-45477-1_15
7. Fielder, A., Panaousis, E., Malacaria, P., Hankin, C., Smeraldi, F.: Decision support approaches for cyber security investment. Decis. Support Syst. **86**, 13–23 (2016)
8. Frigault, M., Wang, L.: Measuring network security using Bayesian network-based attack graphs. In: 2008 32nd Annual IEEE International Computer Software and Applications Conference, pp. 698–703, July 2008
9. Hugin Expert A/S. Hugin lite 8.3 (2016). http://www.hugin.com
10. ICS-CERT: Incident response activity July 2015–August 2015 (2015). https://ics-cert.us-cert.gov/monitors/ICS-MM201508
11. ICS-CERT: Incident response activity September 2014–February 2015 (2015). www.ics-cert.us-cert.gov/monitors/ICS-MM201502
12. Kornecki, A.J., Subramanian, N., Zalewski, J.: Studying interrelationships of safety and security for software assurance in cyber-physical systems: approach based on Bayesian belief networks. In: 2013 Federated Conference on Computer Science and Information Systems (FedCSIS), pp. 1393–1399. IEEE (2013)
13. Langer, R.: Robust Control System Networks-How to Achieve Reliable Control After Stuxnet. Momentum Press, New York (2012)
14. Li, T., Hankin, C.: A model-based approach to interdependency between safety and security in ICS. In: Proceedings of the 3rd International Symposium for ICS & SCADA Cyber Security Research, pp. 31–41. British Computer Society (2015)
15. Liu, Y., Man, H.: Network vulnerability assessment using Bayesian networks. In: Defense and Security, pp. 61–71. International Society for Optics and Photonics (2005)
16. Muñoz-González, L., Sgandurra, D., Barrère, M., Lupu, E.: Exact inference techniques for the dynamic analysis of attack graphs. arXiv preprint arXiv:1510.02427 (2015)
17. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann, Burlington (2014)
18. Poolsappasit, N., Dewri, R., Ray, I.: Dynamic security risk management using Bayesian attack graphs. IEEE Trans. Dependable Secure Comput. **9**(1), 61–74 (2012)

19. Stouffer, K., Falco, J., Scarfone, K.: Guide to industrial control systems (ICS) security. NIST special publication (2011). http://csrc.nist.gov/publications/nistpubs/800-82/SP800-82-final.pdf
20. Wang, L., Jajodia, S., Singhal, A., Cheng, P., Noel, S.: k-zero day safety: a network security metric for measuring the risk of unknown vulnerabilities. IEEE Trans. Dependable Secure Comput. **11**(1), 30–44 (2014)
21. Wang, L., Zhang, M., Jajodia, S., Singhal, A., Albanese, M.: Modeling network diversity for evaluating the robustness of networks against zero-day attacks. In: Kutyłowski, M., Vaidya, J. (eds.) ESORICS 2014. LNCS, vol. 8713, pp. 494–511. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-11212-1_28
22. Weber, P., Medina-Oliva, G., Simon, C., Iung, B.: Overview on Bayesian networks applications for dependability, risk analysis and maintenance areas. Eng. Appl. Artif. Intell. **25**(4), 671–682 (2012)

Power Auctioning in Resource Constrained Micro-grids: Cases of Cheating

Anesu M. C. Marufu[1(✉)], Anne V. D. M. Kayem[1], and Stephen D. Wolthusen[2,3]

[1] Department of Computer Science, University of Cape Town, Rondebosch, Cape Town 7701, South Africa
{amarufu,akayem}@cs.uct.ac.za
[2] Department of Information Security and Communication Technology, Norwegian University of Science and Technology, Gjøvik, Norway
[3] School of Mathematics and Information Security, Royal Holloway, University of London, Egham, UK
stephen.wolthusen@ntnu.no

Abstract. In this paper, we consider the Continuous Double Auction (CDA) scheme as a comprehensive power resource allocation approach on micro-grids. Users of CDA schemes are typically self-interested and so work to maximize self-profit. Meanwhile, security in CDAs has received limited attention, with little to no theoretical or experimental evidence demonstrating how an adversary cheats to gain excess energy or derive economic benefits. We identify two forms of cheating realised by changing the trading agent (*TA*) strategy of some of the agents in a homogeneous CDA scheme. In one case an adversary gains control and degrades other trading agents' strategies to gain more surplus. While in the other, K colluding trading agents employ an automated coordinated approach to changing their *TA* strategies to maximize surplus power gains. We propose an *exception handling* mechanism that makes use of allocative efficiency and message overheads to detect and mitigate cheating forms.

Keywords: Micro-grid · Power auctioning
Continuous Double Auctioning · Cheating attacks · Agent strategy

1 Introduction

Continuous Double Auction (CDA) is a market mechanism that can be used to support power trading in resource constrained (RC) smart micro-grids [1–4]. We understand RC smart micro-grids to be small, integrated energy systems intelligently managing interconnected loads and distributed energy resources. Such smart micro-grids are capable of autonomous operation in case of failure of national grids, or may operate independently in remote locations. In order to minimise costs community-oriented smart micro-grids usually operate on resource limited information technology infrastructure.

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 137–149, 2017.
https://doi.org/10.1007/978-3-319-71368-7_12

Among a plenitude of resource allocation mechanisms proposed, decentralised CDAs are better than standard auctioning schemes, for smart micro-grids that are based on distributed energy generation sources [2]. The decentralised CDA have three advantages. First they enable coordination of distributed components by performing local decision-making based on incomplete and imperfect information to fairly balance elastic demand and partially elastic supply. Second, decentralised CDAs incur minimal computational cost and so operate well on information technology resource limited micro-grids. Third they ensure robust, reliable and fair energy allocation. In the CDA scheme sellers aim to sell power at a higher price than the currently going value while buyers aim to purchase power the lowest possible rate. Employing the CDA algorithm to support auctioning enables trading to occur in a fully decentralized fashion without incurring a high message-passing overhead [2,3]. Auctioning algorithms, such as the CDA scheme, with a low message-passing overhead are therefore desirable for use in smart micro-grids that are supported by lossy networks[1]. However, in studying the operation of CDA schemes, it is necessary to consider the problem of cheating attacks that are aimed at power theft. Cheating attacks can basically take on one or both of two forms, in the first case the adversary seeks to trick buyers into paying more than the lowest possible price for power; while the second case involves the adversary tricking the sellers into accepting the worst bid instead of the best. In the following description we provide an illustrative example of how cheating adversarial cases can be provoked on the CDA scheme.

Example: If a homogeneous population of trading agents is used, a decentralised CDA [2] appears to be robust to traditional forms of cheating that include: multiple bidding, bid shading, false bids and so on. On the other hand, if we consider the possibility of adversaries changing their agent trading strategy in violation of the auction rules, cheating becomes possible in the decentralized CDA scheme.

Case 1: Suppose an adversary uses an automated tool such a malicious software that attaches itself to other participants *TA*s granting them full control. In this case, he/she could provoke these agents to 'downgrade' to an inferior trading strategy. A number of trading agent strategies have been developed for the CDA over the years [5–8]. Importantly, some strategies are superior, with the ability to gain more surplus from trade than their inferior counterparts. Experimental evidence indicates that the Adaptive Aggressive strategy (AA) is one such superior strategy, while the Zero Intelligence (ZI) is the most inferior [5,7,8]. Once in control of victim agents the adversary's 'payload' will trigger a strategy change from AA to ZI. If no further coordination occurs between the infected agents and the adversary, it would be difficult to use communication overheads to infer occurrence of such cheating.

[1] Networks made up of many embedded devices with limited power, memory, and processing resources.

Case 2: We assume a number of traders collude to change bidding strategy in a way that benefits them. According to Vach and Mašál [9], the *Gjerstad-Dickhaut* Extended strategy (GDX) challenges the supremacy attributed to AA from previous research. Results in [9] indicate that GDX wins in an overwhelming number of rounds with significant surplus differences showing the clear effect of changing population shares from one strategy to another. A number of adversary nodes form a coalition using an automated tool to leverage on this phenomenon. The aim of such a tool is to maintain the adversarial agents population to truthful traders population share, such that they gain more surplus in the market. As indicated in Vach and Mašál [9] the minority can dominate in average profit while the allocative efficiency may decrease.

In this paper, we explore the two cases of cheating attacks provoked when an adversary trader(s) change their agent strategies to gain additional surplus power in violation of the auction rules. Further we suggest plausible countermeasures to mitigate the attacks. The remainder of this paper is structured as follows: Sect. 2 discusses the state of the art. Section 3 describes the decentralised CDA model, while Sect. 4 details two cheating attacks towards the CDA mechanism. We propose some mitigation solutions to the cheating attacks in Sect. 5. Section 6 concludes this article and identifies on-going and future work.

2 Related Work

Most widely studied cheating forms are in single-sided auctions and to a small extent centralised CDAs. The decentralised CDA in Marufu *et al.* [2] is a fairly comprehensive auction mechanism for energy resource allocation discouraging some common known forms of cheating. These known forms of cheating include:

Multiple Bidding: A bidder can place multiple bids on the same item using different aliases [10]. Some of these bids can be higher than the bidders reservation price of the product. The multiple bids drive prices to such an extent that other participants prefer to withdraw. Towards the end of the auction the cheater also withdraws all his "fake" bids except the one, which is just above the second highest bid and acquires the product in the lowest possible price. Such cheating is possible in single-sided auctions and in mechanisms that allow bid withdrawal. In considered CDA [2], a single trader agent can only have a single alias, without the possibility of bid withdrawal. The decentralised CDA ensures that a single offer at a time is serviced and once a matching ask is available the market clears instantly.

Bid Shading: A bidder may adopt some unfair ways to examine the sealed-bids before the auction clears and revise his bid to win the auction at a minimum price far below his valuation. This practice is called bid shading [11]. Bid shading is possible in a Periodic Double Auction (PDA) in which offers are collected and cleared at the end of a trading day. A cheater may adopt some unfair ways to examine the bids and revise their bid to win the auction far below his/her reservation price. The considered decentralised CDA is an open-bid auction that

clears continuously as soon an offer is submitted, which eliminates the possibility of such cheating from occurring.

Rings: A group of bidders form a coalition called the ring. These ring members collude not to compete with each other and do not raise the price of the object [13]. This traditional understanding of coalitions is ideal in single-sided auction mechanisms.

Shill Bidding: A corrupt seller appoints fake bidders (shills) who place bids to increase the price of the item without any intention of buying it [12,13]. Shill bidding may only occur on the premise that traders withdraw their bid before the market clears. In the CDA considered, once a "shill" submits an offer it is cleared instantly and there is no chance to forfeit the purchase.

False Bids: A bidder cheats in a second-price sealed-bid auction by looking at the bids before the auction clears and submitting an extra bid just below the price of the highest bid. Such extra bids are often called false bids [11]. Cheating of this form is evident in closed bid auctions e.g. second-price sealed bid auction as opposed to CDA mechanisms which are mostly open bid formulations.

Misrepresented/Non Existent Items: A seller(s) might make false claims about the item they have for sale, or attempt to sell an item they do not have. An assumption made in Marufu *et al.* [2] is that every participants owns up to their bargain for an energy they possess; sold or bought through a community agreement.

Automation of trade within the decentralised CDA provides an easy, efficient and almost seamless way for energy-allocation in constrained micro-grids. Such automation in auctioning facilitates an interesting set of automated forms of cheating which to the best of our knowledge have not been documented prior to this article. We explore plausible automated cheating attacks with the goal of understanding how such attacks can be manifested in a decentralised CDA supporting energy distribution.

Cheating is auction mechanism specific, thus cheating forms and the countermeasures employed to deter the cheating rely on the auction mechanism. To the best of our knowledge research in [14,15], is the only closest work that specifically addresses CDA security. Wang and Leung in [14], describe an anonymous and secure CDA protocol for electronic marketplaces, which is strategically equivalent to the traditional CDA protocol. Trevathan *et al.* in [15], demonstrated that, Wang and Leung's scheme [14] allows the identity of a bidder to be revealed immediately after his/her first bid. Furthermore, it allows profiles to be created about a bidder's trading behaviour, as bids are linkable. Hence in their scheme they propose incorporating a group signature scheme to anonymise traders and secure the trading process. Wang and Leung's scheme is given in the context of Internet retail markets while Trevathan *et al.*'s scheme was designed specifically for share market applications. Although the schemes described in these works could protect the customer's privacy including affording anonymity, robustness and non-repudiation, but they are not suitable for deterring automated cheating

in decentralised CDAs such as the one described by Marufu *et al.* [2] for the following reasons:

Absence of a centralised auctioneer the CDA mechanisms discussed in the aforementioned works [14,15] rely on a centralised architecture where bids are relayed through a central component - Auctioneer; which determines the winner according to the auction rules. The problem arises when an Auctioneer influences the auction proceedings in a manner inconsistent with the auction rules. For example, the Auctioneer might choose to block bids, insert fake bids, steal payments, profile bidders, prematurely open sealed bids, artificially inflate/deflate prices or award the item to someone other than the legitimate winner. Protecting a bidder's identity and bidding information is crucial since each bidder/seller's private information can be inferred at the central Auctioneer. Further more, the Auctioneer presents a single point of failure, is open to biases and can be easily be manipulated to obtain favourable trades or reveal traders' reserved information. In a decentralised CDA [2], the some of the Auctioneer-duties are carried out by a mobile token distributed among the participants following a MUTEX protocol.

Different auction clearing mechanisms work in [14,15] considers a double auction mechanism where the market clears periodically. Such a market mechanism allows bidders and sellers to submit bids for a period of time where the auctioneer then clears the matching bids. The Marufu *et al.* CDA mechanism lasts a fixed period of time, known as the trading period (at the end of which the market closes and no more offers are accepted). The traders will continuously submit offers at any time during a trading period while the market continuously clears matching *bids* and *asks* (i.e., whenever a new transaction is possible between an acceptable bid and ask). The trading parties mutually and exclusively submit an order into a mobile *order-book* to negotiate a deal (see Sect. 3). These works are valuable to our work as they provide a form of benchmark for our security considerations towards a cheating attack.

3 Decentralised Continuous Double Auction Model

The community is made up of a number of clustered households within a particular area with close-neighbouring households sharing a single smart meter. Members use mobile phones to connect to the network of smart meters allowing community members to participate in a CDA market. Similar to [2], we consider that trading agents are hosted on mobile device that is securely connected to the micro-grids' communication network. Each agent makes use of the public information made available to them, such as: the prices of previous transactions; the behaviour of the other agents in previous rounds; and market information provided by the competition. This homogeneous population of agents employ the AA strategy developed by [7]; presented in [8]; and adopted in [2,4]. Marufu *et al.* [2] opted for the AA strategy because it was shown to outperform other benchmark agent strategies (in terms of market efficiency both

for heterogeneous and homogeneous populations) [7] and even proved superior against human traders in human vs. agent experiments [5]. We assume a hierarchically clustered network with two distinct sets of entities: a large number of mobile phones M_{mp} and relatively fewer, fixed smart meters M_{sm}, hence $M_{mp} >> M_{sm}$ (see Fig. 1).

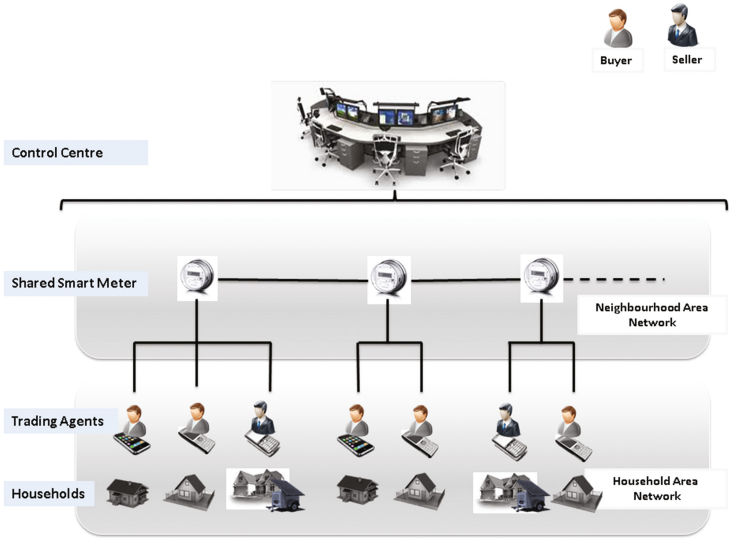


Fig. 1. Decentralised CDA architecture

We conform to a market institution[2] specified in [2,3]. Sellers and buyers can submit their *asks* and *bids* at any time during a CDA trading day. The current lowest *ask* in the market is called the *outstanding ask* (o_{ask}). The current highest bid in the market is called the *outstanding bid* (o_{bid}). The CDA scheme includes the *New York Stork Exchange (NYSE) spread-improvement* and the *no-order queuing* rules. NYSE spread-improvement requires that a submitted bid or ask 'improves' on the o_{bid} or o_{ask}. The no-order queuing rule specifies that offers are single unit, therefore are not queued in the system but are erased when a better offer is submitted as bid-spread (the difference between o_{bid} and o_{ask}) decreases. We also consider discriminatory pricing and information dissemination done by the moving order-book. To each *TA* (seller or buyer agent), a unit of energy has a reservation price (limit price) secret to them. If a seller (buyer) submits an offer lower (higher) than the reservation price, they will lose profit. An offer (bid/ask) is formed through an agent trading strategy and submitted into the market. Thus, the CDA mechanism continuously allows offers from traders and matches the new ones with the already submitted offers. A deal is made if the

[2] How the exchange takes place; formalising rules of engagement of traders and the final allocation of commodities.

price of a *bid* on the market is greater or equal to price of the *ask*. We consider only a single *TA* can exclusively submit an offer through a MUTEX protocol (refer to [2,3]). *TA*s can send a new offer when their earlier offer has been cleared (if it was matched and the deal was accomplished). If the deal for energy is not accomplished the unmatched bids can be queued in an order-book, ordered by decreasing bid prices such that the higher and more desirable offers -from sellers perspective, are at the top. Information about the market state namely the o_{bid}, o_{ask} and current transaction price p is made public to all participants through the order-book. Activities in Marufu *et al.* [2] decentralised CDA scheme can be summarised into:

- **Registration:** TAs must first register (with a registration manager) to participate in the auction. This is a once-off procedure that allows TAs to participate in any number of auctions rounds.
- **Initialisation:** A token (mobile object) is initialised at the beginning of a trade day.
- **Participation Request:** *TA*s that want to submit an offer in the market will request for the token. A MUTEX protocol is used to serialize auction market access.
- **Bid Formation:** A *TA* with the token will compute their offer (bid/ask) using the AA trading strategy and submit it into the mobile order-book (which is part of the token).
- **Transacting:** If matching offers occur, a trade occurs, otherwise the offer is put in the order-book as an outstanding ask/bid. The trade information and outstanding offers are made public and visible to other agents.
- **Termination:** The order book is mutual exclusively distributed until the end of a trade day.

4 Cheating CDA Attacks

In this section, we present cases that indicate the susceptibility of a decentralised CDA protocol to adversary changing agent strategies in a homogeneous agent population.

4.1 Case 1: Victim Strategy Downgrade

We consider a single adversary uses an automated tool (e.g. malware) to gain control of other participants' trading agents. The adversary then 'downgrades' the victim *TA*s to an inferior trading strategy. Further assumptions made are:

- the number of all *TA*s n is known to all participants;
- n *TA*s are in the market and the adversary has control of $n-1$ *TA*s (including his agent);
- if the victims' strategy is downgraded to ZI strategy, the *TA*s will seamless continue participating in the auction as normal;

Literature on the matter supports that there are a number of trading agent strategies that have been developed for the CDA over the years. Importantly, some strategies are superior, with the ability to gain more surplus from trade than their inferior counterparts. Experimental evidence indicates that the Adaptive Aggressive strategy is one such superior strategy, while the Zero Intelligence is the most inferior [5,7,8]. Ma and Leung [16] demonstrates that AA agents are adaptive to different combinations of competitors; and to different supply and demand relationships. ZI agents behaved worse since they do not analyse their environment and the other agents whom they are competing with. The AA obtains huge profit margins in comparison to ZI strategy [16].

Proof sketch: Marufu *et al.* CDA scheme uses a token-based mutual exclusion protocol to ensure TA randomly and fairly trade in the auction. Similarly the schemes in [5,7,8,16] operate under the same assumption. The same agent strategy (GDX and AA) specification is used in all these works indicating how the abstract results can support the attacks described. The additional assumptions we make will not affect the initial correctness of the Marufu *et al.* scheme.

Attack: We assume an adversary distributes a malicious code to attach itself onto other participants *TA*s. Once in control of victim agents the adversary's 'payload' will trigger a strategy change from AA to ZI. This attack can take two forms: static or dynamic downgrade. In static downgrade, an agent will instantly change its strategy on infection (once adversary payload is delivered). This change is somehow permanent. If no further coordination occurs between the infected agents and the adversary, it would be difficult to use communication overheads to infer occurrence of such cheating. However, such attack can easily be detected by analysis of market efficiency as ZI agent population yields fairly lower market efficiency than a homogeneous AA population. Thus, an advanced adversary would employ a dynamic downgrade to victims *TA* strategies allowing victim agents to revert back to the AA strategy based on a clock-based trigger. The attacker uses a fixed but arbitrary duration to downgrade or revert back changes. The payload uses clocks that ensures at the beginning of a trade round i ($i \in R$, where $R = 1000$ trading rounds in a trading day) all victims will downgrade to ZI strategy. After a lapse of a certain period of time or number of rounds or trading days another clock-based trigger will revert the victim strategy back to the AA strategy.

4.2 Case 2: Collusion Attack

We assume a number of traders collude to change bidding strategy. In this subsection we discuss additional assumptions made in carrying out the cheating attack. We further consider that:

- the symmetry of participants, that means the number of buyers and sellers pursuing one strategy is the same;
- all *TA*s get the same amount of units to trade;
- a single trading day has 1000 rounds and there are infinitely many trading days;

– if a *TA* changes its strategy it can seamlessly continue participating in the auction;
– All *TA*s receive a signal to mark beginning of the trading day.

Vach and Maršál in [9] demonstrated through experimental evidence that, GDX[3] challenges the overall supremacy attributed to AA from previous research. Table 1 shows results of 100 rounds of each experiment. First row shows how many buyers and sellers following one strategy compete against traders following the other strategy[4]. The abstract results indicate that AA dominates clearly one AA vs. many GDX and unbalanced 2 AA vs. 4 GDX experiments. For the rest of mixed experiments, GDX is a more favourable one and for unbalanced 2 GDX vs. 4 AA and one GDX vs. many AA experiments, GDX wins in an overwhelming number of rounds. The surplus difference shows the clear effect of changing population shares from one strategy to another one. Thus, by forming a coalition with other traders using an automated tool potential cheating leveraging on this phenomenon may occur. The aim of such a tool is to ensure the adversarial agents population share is such that they gain more surplus in the market. As indicated the minority dominates in average profit.

Table 1. AA vs. GDX agent strategies (Source: [9, p. 47])

GDX vs. AA	**6:0**	**5:1**	**4:2**	**3:3**	**2:4**	**1:5**	**0:6***
GDX won rounds	1000	176	366	696	913	976	0
AA won rounds	0	824	634	304	87	24	1000
GDX efficiency	55.31%	94.7%	97.92%	100.48%	104.18%	114.29%	-
(s.s.d.)	(32.6%)	(9.6%)	(3.85%)	(3.72%)	(5.19%)	(8.83%)	-
AA efficiency	-	112.95%	101.15%	94.26%	87.93%	77.38%	-
(s.s.d.)	-	(17.26%)	(8.87%)	(7.78%)	(8.56%)	(10.76%)	-
Total efficiency	55.31%	97.74%	99%	97.37%	93.35%	83.53%	-
(s.s.d.)	(32.6%)	(7.2%)	(1.57%)	(2.56%)	(4.54%)	(7.93%)	-
Surplus difference	-	27.39	4.84	9.33	24.36	55.37	-
Winner	**GDX**	**AA**	**AA**	**GDX**	**GDX**	**GDX**	**AA**

Proof Sketch: The CDA schemes described in the baseline study by [6,9] are closely similar to the CDA scheme we follow by Marufu *et al.* [2]. For instance, market clearing is continuous, TAs are chosen randomly to submit an offer, and the same agent strategy (GDX and AA) specification is used in all schemas. Thus, the results were produced under assumptions similar to the ones we make in this paper. Furthermore, the assumptions we add to the Marufu *et al.* such as number of traders, limit prices of each unit and the amount of traded units do not change

[3] Strategy developed by Tesauro and Bredin [6] as a modification of the Gjerstad-Dickhaut (GD) strategy that uses dynamic programming to price orders.
[4] This means 2 buyers and 2 sellers following the first strategy compete against 4 buyers and 4 sellers following the second strategy.

or alter the correctness of the initial Marufu *et al.* CDA algorithm. Correctness of the Marufu *et al.* CDA algorithm was based on guaranteeing that the four mutual exclusion properties are not violated. The additional assumptions we make do not affect the mutual exclusion properties.

Attack: The adversary *TA*s to truthful *TA*s ratios are maintained by coordinating the number of colluding agents allowed in a trading day. Colluders, K, are a subset of *TA*s (buyer or sellers) who agree to cheat by changing their strategy to GDX. Each colluding *TA* installs or allows installation of a tool (piece of software or script) that enables them to coordinate and strategically change agent strategy from AA to GDX. Thus, $K \subset TA$. The colluding *TA*s use a separate channel to communicate among themselves. On receipt of the beginning of trading day signal the strategy changing automated tool will allow a number of adversary agents to shift their strategy to the GDX. The adversary agent population ratio to the truthful agent population can be either 2 to 4 or 1 to m to ensure a high surplus on adversary population. To ensure such coordination the strategy changing automated tool can use the k-MUTEX protocol to select the maximum number of colluders that can change their strategy. The k-MUTEX algorithm will allow at most k colluders at a time to change their strategy (enter critical section). The protocol is token based and k tokens are used. Thus a colluding *TA* can only change its strategy to GDX when it is in possession of the token. Chaudhuri and Edward [17] proposed one such protocol which performed on a worst case message complexity of $O(\sqrt{n})$. The single trade-day-signal that is used to trigger the selection of colluding *TA*s can be altered to a number of signals (therefore trading days) to allow the selected colluders more number of rounds to benefit before the change. Importantly, the developer of the collusion tool can benefit from selling the opportunity to cheat to other participants. Tolerance to node and link failure ensures robustness of the k-MUTEX protocol. We assert that as long as the colluders are coordinated in such a manner this attack will not deviate, with high probability that colluders will continuously take turns and cheat.

5 Sketch Countermeasures

Cheating attacks give rise to exceptions, situations which fall outside the normal operating conditions expected of the TAs. One way to deal with exceptions is employing *exception handling* by distinct domain-independent agents. This approach has been described as the citizen approach [17], by analogy with the way that exceptions are handled in human society. Citizens adopt relatively simple and optimistic rules of behaviour, and rely on a range of social institutions (law enforcement, the legal system, disaster relief agencies, and so on) to handle most of the exceptions that arise. Two measures can be used by the exception handlers to address cheating attacks discussed in Sect. 4. One is allocative efficiency a measure of how well the market runs. This measure is given as a ratio of the profit made during the auction to the profit that could be made if the agents traded in the most efficient way (if each offered at its private value, and the

traders were matched to maximise the profits obtained). This provides a measure of the effectiveness of the market in economic terms. As indicated in the experimental results motivating our cheating attacks, colluding *TA*s may gain higher surplus while a decrease in the allocative efficiency will be observed. Intuitively, an exception handling mechanism can make use of such information to detect and mitigate cheating. The second exception measure takes care of the number of messages passed among *TA*s in the auction. This gives a computational measure of efficiency, that is, how many resources the auction consumes in a run. A slight and sudden increase in the number of messages will give off a red flag for possible cheating. The two measures can be used to compliment each other to detect and deal with automated forms of cheating we describe in this article.

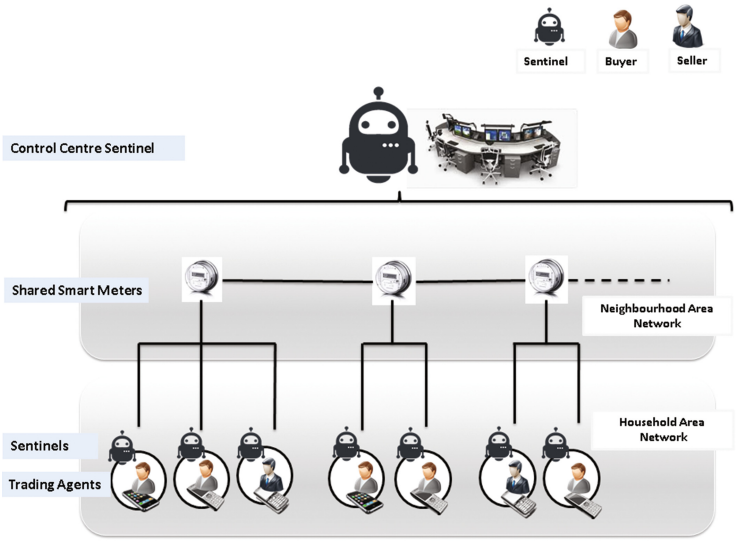


Fig. 2. Exception handling architecture

To provide a citizen approach to exception handling we define an exception handling infrastructure, which similar to Parson [18], associates a sentinel/observer to each *TA*. The resulting system is as shown in Fig. 2. To provide exception handling services, we assume each sentinel has no access to the internal state of the agent it is associated with. It is essential that sentinels need not have such access as this could open up the system to a myriad array of adversaries through sentinel compromise. All sentinels work with a control centre sentinel and communicate through the main micro-grid network. We consider that each sentinel is also hosted in the mobile device and knows the maximum number of messages u that are supposed to be transmitted to and from a *TA*. The sentinel is able to observe some irregular messages v transmitted to and

from the agent. If a sentinel observes the anomaly $(m+n)$ it will raise a red-flag on the incident and send it to the control centre for further enquiry. In order to reduce the message overhead, a sentinel will only send a feedback message to the control centre if an incident is red-flagged. If a red-flagged incident, of say, TA_i happens to coincide with other *TA*s red-flag incidents, agent manipulation/collusion can be affirmed. The control centre which analyses trends in trade history and allocative efficiency of the involved agents will identify possible collusions or downgrades.

For instance, in a *Strategy-Downgrade Cheating*, positive cheating detection involves: identification of an individual TA with a constantly higher surplus, while the other *TA*s have distinctly lower surplus; and a significant number of simultaneous red-flags reported by the sentinels, low allocative efficiency of the mechanism. *Collusion Attack* identification includes observing a subset of K *TA*s constantly obtaining a higher surplus in as many rounds; a significant number of simultaneous red-flag reported by the sentinels. If perpetrators are identified there can be suspended from the auction for a number of rounds/trading days.

6 Conclusions

In this article we explored the two cases of cheating attacks provoked when an adversary trader(s) change their agent strategy to gain additional surplus in violation of the auction rules. These are Strategy Downgrade cheating where an adversary uses an automated tool to downgrade other traders strategy; and Collusion attack with a number of traders change their bidding strategy. We argue the plausibility of these attacks based on experimental evidence found in literature. The cheating attacks proposed give rise to exceptions, which are situations which fall outside the normal operating conditions expected of the *TA*s. We adopt a similar approach to one in [18] of employing *exception handling* by distinct domain-independent agents. The exception handling mechanism makes use of allocative efficiency and message overheads to detect and mitigate cheating forms described herein. We argue that exception handling could deter the two forms of cheating while yielding low message overheads. As future work, we plan to validate and evaluate the exception handling protocol through some theoretical and experimental evidence. Due to the significance of CDAs in resource allocation, we believe that exploring and mitigating cheating will be interesting to micro-grid research community.

References

1. Haque, A., Alhashmi, S.M., Parthiban, R.: A survey of economic models in grid computing. Future Gener. Comput. Syst. **27**(8), 1056–1069 (2011)
2. Marufu, A.M.C., Kayem, A.V.D.M., Wolthusen, S.D.: A distributed continuous double auction framework for resource constrained microgrids. In: Rome, E., Theocharidou, M., Wolthusen, S. (eds.) CRITIS 2015. LNCS, vol. 9578, pp. 183–196. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-33331-1_15

3. Marufu, A.M.C., Kayem, A.V.D.M., Wothulsen, S.D.: Fault-tolerant distributed continuous double auctioning on computationally constrained microgrids. In: Proceedings of the 2nd International Conference on Information Systems Security and Privacy, ICISSP 2016, pp. 448–456. SCITEPRESS (2016)
4. Stańczak, J., Radziszewska, W., Nahorski, Z.: Dynamic pricing and balancing mechanism for a microgrid electricity market. In: Filev, D., Jabłkowski, J., Kacprzyk, J., Krawczak, M., Popchev, I., Rutkowski, L., Sgurev, V., Sotirova, E., Szynkarczyk, P., Zadrozny, S. (eds.) Intelligent Systems'2014. AISC, vol. 323, pp. 793–806. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-11310-4_69
5. De Luca, M., Cliff, D.: Human-agent auction interactions: adaptive-aggressive agents dominate. In: IJCAI Proceedings-International Joint Conference on Artificial Intelligence, vol. 22, p. 178. Citeseer (2011)
6. Tesauro, G., Bredin, J.L.: Strategic sequential bidding in auctions using dynamic programming. In: Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems: Part 2, pp. 591–598. ACM (2002)
7. Vytelingum, P.: The structure and behaviour of the Continuous Double Auction. Ph.D. thesis, University of Southampton (2006)
8. Vytelingum, P., Cliff, D., Jennings, N.R.: Strategic bidding in continuous double auctions. Artif. Intell. **172**(14), 1700–1729 (2008)
9. Vach, D., Maršńl, A.M.A.: Comparison of double auction bidding strategies for automated trading agents (2015)
10. Yokoo, M., Sakurai, Y., Matsubara, S.: The effect of false-name bids in combinatorial auctions: new fraud in Internet auctions. Games Econ. Behav. **46**(1), 174–188 (2004)
11. Porter, R., Shoham, Y.: On cheating in sealed-bid auctions. Decis. Support Syst. **39**(1), 41–54 (2005)
12. Chakraborty, I., Kosmopoulou, G.: Auctions with shill bidding. Econ. Theor. **24**(2), 271–287 (2004)
13. Trevathan, J., Read, W.: Detecting shill bidding in online English auctions. In: Handbook of Research on Social and Organizational Liabilities in Information Security, p. 446. Information Science Reference, Hershey (2008)
14. Wang, C., Leung, H.: Anonymity and security in continuous double auctions for internet retails market. In: Proceedings of the 37th Annual Hawaii International Conference on System Sciences, 10 pp. IEEE (2004)
15. Trevathan, J., Ghodosi, H., Read, W.: An anonymous and secure continuous double auction scheme. In: Proceedings of the 39th Annual Hawaii International Conference on System Sciences, HICSS 2006, vol. 6, p. 125b. IEEE (2006)
16. Ma, H., Leung, H.-F.: An adaptive attitude bidding strategy for agents in continuous double auctions. Electron. Commer. Res. Appl. **6**(4), 383–398 (2008)
17. Chaudhuri, P., Edward, T.: An algorithm for k-mutual exclusion in decentralized systems. Comput. Commun. **31**(14), 3223–3235 (2008)
18. Parsons, S., Klein, M.: Towards robust multi-agent systems: handling communication exceptions in double auctions. In: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems, vol. 3, pp. 1482–1483. IEEE Computer Society (2004)

Using Incentives to Foster Security Information Sharing and Cooperation: A General Theory and Application to Critical Infrastructure Protection

Alain Mermoud[1,2(✉)], Marcus Matthias Keupp[2,3], Solange Ghernaouti[1], and Dimitri Percia David[1,2]

[1] Swiss Cybersecurity Advisory and Research Group (SCARG), University of Lausanne, 1015 Lausanne, Switzerland
alain.mermoud@unil.ch
[2] Department of Defence Management, Military Academy at ETH Zurich, 8903 Birmensdorf, Switzerland
[3] Institute of Technology Management, University of St. Gallen, Dufourstrasse 40a, 9000 St. Gallen, Switzerland

Abstract. Various measures have been proposed to mitigate the underinvestment problem in cybersecurity. Investment models have theoretically demonstrated the potential application of security information sharing (SIS) to Critical Infrastructure Protection (CIP). However, the free rider problem remains a major pitfall, preventing the full potential benefits of SIS from being realised. This paper closes an important research gap by providing a theoretical framework linking incentives and voluntary SIS. This framework was applied to CIP through a case study of the Swiss Reporting and Analysis Centre for Information Security. The SIS model was used to analyse the incentive mechanisms that most effectively support SIS for CIP. Our work contribute to an understanding of the free rider problem that plagues the provision of the public good that is cybersecurity, and offer clues to its mitigation.

1 Introduction

Investment in cybersecurity[1] remains suboptimal because of the presence of externalities that cannot be completely internalised by the investor [23]. As a result, a Nash-stable yet inefficient equilibrium emerges in which each cybersecurity investor attempts to free-ride on the investments of others, producing a suboptimal global level of cybersecurity in the economy [22]. While such free-rider problems are pervasive in the private sector, presenting significant risks to the national economy, the situation is exacerbated when national security is also

[1] In our study we use the term "cybersecurity" as a synonym for "information security," referring to the protection of information that is transmitted over the Internet or any other computer network.

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 150–162, 2017.
https://doi.org/10.1007/978-3-319-71368-7_13

taken into account [22]. Universities, critical infrastructure (CI) providers, government, and the armed forces all rely heavily on information technology systems to fulfill their mandates. This makes them vulnerable to cyberattacks, making the consequences of cybersecurity breaches especially harmful for society as a whole [2]. Various measures have been proposed to mitigate this underinvestment problem, among which the sharing of cybersecurity-relevant information appears the most viable and relevant, as a way of simultaneously mitigating the externalities and increasing individual utility, inter-investor information, and social welfare [17]. Further, such sharing can help to reduce the information asymmetry between an attacker and the defender in the case of zero-day vulnerability attacks, in which the defender has little, if any, time to react [28]. By "sharing of cybersecurity-relevant information", we refer to a process by which cybersecurity investors provide each other with information about threats, vulnerabilities, and successfully defended cyberattacks. For the sake of brevity, we will refer to this as "security information sharing" (SIS)[2]. The remainder of this paper will proceed as follows. In the first and second section of the paper, we develop a theoretical framework and present our propositions. Section 3 reports a case study. In Sect. 4, we discuss the limitations and possible extensions of the model. Our concluding remarks and proposals for future work are given in Sect. 5.

2 Theoretical Framework and Propositions

The importance of SIS for Critical Infrastructure Protection (CIP) is widely acknowledged by academics, policy-makers, and industrial actors, as it can reduce risks, deter attacks, and enhance the resilience of the CI [10]. Cybercriminals and hackers have a long history of sharing experiences, tools, and vulnerabilities, and this has contributed to the success of major cyber-attacks. The timely introduction of SIS is therefore vital for CIP, because sharing by attackers is eroding the effectiveness of traditional defense tools [8]. The Gordon-Loeb model has theoretically demonstrated the potential application of SIS to CIP [20]. However, empirical studies have shown that a significant free rider problem exists, preventing the full potential of SIS from being realised [12]. Although SIS offers a promising way of reducing the global investment needed to establish cybersecurity, extant empirical research shows that both the frequency of SIS (i.e., the number of security information sharing transactions between investors in a given time interval) and its intensity (i.e., the depth of information shared in each transaction, represented by the number of comments related to each incident shared) remain at rather low levels in the absence of any further intervention. In the absence of appropriate extra incentives, SIS is likely to be conducted at a suboptimal level [5].

[2] "security information sharing" SIS can be defined as the mutual exchange of cybersecurity-relevant information on vulnerabilities, phishing, malware, and data breaches, as well as threat intelligence, best practices, early warnings, and expert advices and insights.

2.1 Regulation Alone Cannot Solve the Free Rider Problem

Incentives can be provided either positively, by increasing the economic and social benefits gained when investors share security information, or negatively, by punishing investors that fail to share. Attempts to introduce negative incentive through regulation have been rather unsuccessful [14]. Despite the introduction of several bills in the USA[3] and in the EU[4] encouraging security information sharing, actual SIS remains at low levels [21]. However, while these have attempted to impose legal requirements to share information on both the private and public sectors, to date such regulations do not seem to be producing the desired effect of increasing SIS [27]. When forced to share SIS, firms may even choose to share irrelevant or incomplete information, especially with competitors [31]. This regulatory failure does not seem to be country-specific, since attempts elsewhere at negative incentivisation by means of regulation, laws, and the imposition of punishments have also been unsuccessful [35].

2.2 Linking Incentives to Voluntary SIS

In this paper we propose an alternative approach to regulation. Security information, once obtained, may be either shared at a small marginal cost or kept private and hoarded by the producer. A theoretical understanding is therefore needed of the mechanisms that would cause investors to voluntarily share SIS [26]. We propose that both the frequency and intensity of SIS will increase if investors are provided with appropriate positive incentives to share information (as opposed to being forced or encouraged to share through regulation). We do not presuppose any particular institutional or organisational design; incentives could be provided by government, through contractual arrangements between participants, or by public-private partnerships (PPPs). While previous research has identified this as a promising approach [24], very little is known about the particular incentives (if any) that actually increase voluntary SIS, or about the mechanisms by which they work. While past contributions have repeatedly stressed the need to develop and test theories linking incentives with SIS outcomes [17], to the best of our knowledge no such theory has yet been produced or tested. As a result, the existing literature provides little serious discussion of the causal linkages by which incentives may be expected to increase the intensity and frequency of SIS. While the collective benefits of SIS have been contrasted with the low levels of sharing actually observed, very little work has been done to identify the types of incentive that may successfully correct this. To close this gap, we propose a theoretical framework that links incentives with voluntary SIS. Our theory identifies the incentives that are expected to increase the frequency and intensity of voluntary SIS, and clarifies the causal mechanisms by which they function.

[3] In particular the 2002 Sarbanes-Oxley Act (SOX) and the 2015 Cybersecurity Information Sharing Act (CISA).

[4] In December 2015, the European Parliament and Council agreed on the first EU-wide legislation on cybersecurity, adopting the EU Network and Information Security (NIS) Directive.

2.3 A Holistic and Multidisciplinary Approach

Arguments have been presented in the psychology and sociology literature on the role of positive incentives in persuading economic actors that they can improve their economic situation by behaving in a particular way [3]. For example, research in behavioral economics has demonstrated that incentives can channel human behavior towards particular options using rewards and sanctions [6]. More generally, this literature has identified human behavior as the weakest link in the cybersecurity chain [18]. In these models, positive incentives offer the agent a Pareto-superior state vis-a-vis the current state, at the cost of behavioral compliance. When applied to SIS, these models suggest that investors will only share information if they expect that the particular incentives provided will allow them to reduce their individual investment in cybersecurity [4]. As our overarching theoretical mechanism, we therefore propose that incentives "work" by changing investors expectations in a first step. These changed expectations then trigger individual actions that result in SIS. We therefore view both the frequency and intensity of SIS primarily as functions of the change in agents' expectations. It is this change itself, rather than the particular incentive which induces it, that is the phenomenon of interest. In practice, a variety of incentives could be used to change expectations, and these are likely to be context-specific to particular countries, political and economic systems, and cultures. The current study differs from previous research in this domain by being grounded in empirical observations from an Information Sharing and Analysis Centre (ISAC)[5]. Our findings constitute an evidence base and an important contribution to the new fast growing field of the "economics of cybersecurity", and are generalisable to other jurisdictions. Most importantly, our results will support the design of the next generation of ISACs, namely Information Sharing and Analysis Organisations (ISAOs)[6], in which incentives and voluntary SIS will play a key role [32]. Fusion centers[7] and the emerging Threat Intelligence Platform (TPI) technology[8] might also benefit from our findings, by providing the right incentives to their members to share more real-time threat data.

2.4 A Model Linking Incentives, Behavior, and SIS

Previous research suggests that four expectations are particularly relevant to the human interactions involved in sharing: reciprocity, value, institutional expectations, and reputation expectations [12]. We designed a two-stage SIS model. In the first step, incentives are provided to change the expectations of the agents.

[5] An ISAC is a generally a nonprofit organisation that provides a platform for SIS between the government and CIs.

[6] Unlike ISACs, ISAOs are not directly tied to CIs and offer a flexible and voluntary approach for SIS.

[7] A fusion center is an information sharing center designed to promote information sharing between different agencies.

[8] The TPI technology helps organizations to analyze and aggregate real-time threat data in order to support defensive actions.

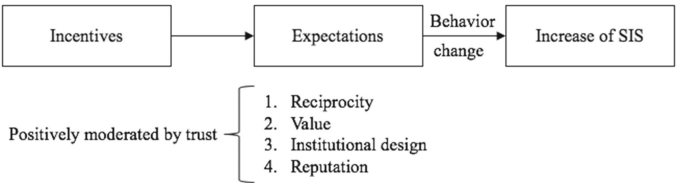


Fig. 1. Design of a two-stage SIS model

In the second step, these changed expectations trigger actions that result in an increase in SIS (Fig.1).

In summary, our model describes how incentives change expectations, modifying the behavior of actors to improve voluntary SIS. We define frequency by the number of shared transactions between participants, and intensity by the depth of SIS in one single transaction. We further propose that the relationship between each of these effects and SIS is positively influenced by the degree of trust between individual agents.

2.5 Reciprocity Expectation

The extent to which human agents will engage in voluntary SIS depends on their expectations of reciprocity or recompense for the information they share [36]. For example, peer-to-peer (P2P) networks often confront the problem of free riders (or so called "leechers"), because most participants would prefer to avoid seeding while enjoying the benefits of the network. As a result, most P2P networks have been forced to remove free riders, or to force them to contribute by making seeding mandatory. Open source studies have shown that, in the absence of an expectation of reciprocity, the benefits that a P2P network can provide are unlikely to be realised [34]. Reciprocity may be self-reinforcing, because participants will share more when provided with an incentive that ensures reciprocity. Hence,

Proposition 1. *The increase in the frequency of SIS will depend on the extent to which investors expect an act of sharing to be reciprocated.*

Proposition 2. *The increase in the intensity of SIS will depend on the extent to which investors expect an act of sharing to be reciprocated.*

2.6 Value Expectation

Previous studies have identified the value of the information obtained as a result of sharing as an important precursor to SIS [13]. SIS and cooperation between industry peers can improve the relevance, quality, and value of information, because the actors are often facing similar cyber-threats. Collective intelligence and crowdsourcing studies have shown that organizations working together have

greater threat awareness. On the other hand, SIS can also place extra burdens on the participants if they lack the resources to understand and analyse the security information that is shared with them. Therefore, each investor is expected to conduct a cost-benefit analysis before deciding whether or not to engage in SIS. The benefits are likely to increase as the value of the information increases; ideally, an investor should conclude that the benefits of the security information received outweigh the costs of the security information shared. Cost saving is generally the most direct and visible benefit of SIS to the participants. This makes the cost argument both subordinate to and linked with the value argument. Hence,

Proposition 3. *The frequency of SIS will increase to the extent that investors expect an increase in the value of the information they hold.*

Proposition 4. *The intensity of SIS will increase to the extent that investors expect an increase in the value of the information they hold.*

2.7 Institutional Expectation

There is a consensus in the literature that the management and institutional design of an ISAC is key to building trust and facilitating effective SIS [33]. A poorly designed ISAC may deter agents from joining, thus reducing the probability of an increase in SIS. Previous researches have identified three elements that are key to an optimal structure: leadership, processing and labelling of shared information, secure storage and access to shared data [15]. A clear taxonomy is needed to create a common vocabulary and culture for participants. Next, minimal standards have to be implemented by the ISAC to organise the formation of the information that is shared. The central point in ISAC management is to create a core of quality participants, in order to encourage further quality participants to join, while avoiding the introduction of free riders [11]. Non participants should perceive themselves to be missing access to important information. If the membership is too large it is difficult to create relationships of trust, whereas if the membership is too small, the amount of shared data will be insufficiently attractive. A sound institutional design should also result in a stable membership. Participants might be reluctant to engage in SIS activities if inappropriate actors are allowed to become members of the ISAC. It is essential that the platform applies up-to-date security standards, in order to provide a safe forum. Indeed, the high value of an ISAC database makes it a high value target (HVT) for hackers. Hence,

Proposition 5. *The frequency of SIS will increase to the extent that investors trust the institutional management.*

Proposition 6. *The intensity of SIS will increase to the extent that investors trust the institutional management.*

2.8 Reputation Expectation

Agents will evaluate the potential reputational benefits of their SIS activities, as well as potential reputational risks. Participants are often reluctant to engage in SIS activities, fearing that they might be damaging to the reputation of the organisation. Reputation is related to customer trust, the protection of customer data, and the quality of service offered [19]. Common fears include information leaks and the use by competitors of critical information to damage the reputation of the client. Studies have shown that disclosing information on a cyberattack may reduce consumer trust, impacting the market value of the company [7]. As a result, agents have a strong interest in protecting their reputation. However, some participants may see SIS as a way of cultivating their reputation as good corporate citizens. Association with government agencies can also enhance the reputation of participants. The fear of being publicly accused of being a free rider might also provide an incentive to participate in SIS. Reputation is strongly moderated by trust, and this can mitigate the reputational problem. If agents know and trust each other they will not exploit any revealed weaknesses. Hence,

Proposition 7. *The frequency of SIS will increase to the extent that investors expect an improvement in reputation.*

Proposition 8. *The intensity of SIS will increase to the extent that investors expect an improvement in reputation.*

2.9 The Moderating Role of Trust

The psychology literature suggests that knowledge-based trust might be the most significant in the context of SIS [25]. Our assumption is that trust is a necessary condition for SIS, but not a sufficient one. As a result, the four main effects noted above will each be positively moderated by trust, strengthening them when trust between agents is present. In many jurisdictions, government and private industry have worked together to create ISACs, as neutral and anonymous facilitators of social networks, thereby supporting the emergence of trusted relationships between cybersecurity investors, the private sector, and the government [16]. The existence of networks of collaboration and trust in other fields of activity can be leveraged for SIS. For instance, preexisting relationships between the private and the public sector can be used to build trust [9]. Hence,

Proposition 9. *The relationship between the expectation of reciprocity and SIS will positively reflect the degree of trust between the sharing agents.*

Proposition 10. *The relationship between value expectations and SIS will positively reflect the degree of trust between the sharing agents.*

Proposition 11. *The relationship between institutional expectations and SIS will positively reflect the degree of trust between the sharing agents.*

Proposition 12. *The relationship between reputational expectations and SIS will positively reflect the degree of trust between the sharing agents.*

3 Application of the Proposed Model to Critical Infrastructure Protection

Today, CIP is more an economic policy than a technology policy. The capacity of a modern society to preserve the conditions of its existence is intimately linked to the proper operation of its CIs. Cybersecurity concerns are the main challenge faced by the operators of this infrastructure, not least because of the high degree of interconnection [2]. This raises the threat of a "cyber subprime scenario", i.e. a cascading series of failures from an attack on a single point in the infrastructure[9]. As a consequence, most OECD countries have adopted national CIP programs to increase preparedness and improve the response to critical cyber incidents. To illustrate our theoretical framework, we present a case study showing how SIS can improve cybersecurity in the financial sector, a particularly sensitive area of CIP. The national financial infrastructure of Switzerland is highly important for national security, given the presence of at least five systemically strategic (too big to fail) banks. For a potential attacker, the Swiss financial system is an attractive target that can be attacked at very little cost.

3.1 The Swiss Reporting and Analysis Centre for Information Security

The Swiss Reporting and Analysis Centre for Information Security (MELANI) is a forum in which participants from the information security technology sector and CI providers share security information. The Centre is organised as a PPP between the federal government and private industry. It operates an ISAC (MELANI-Net), which brings together over 150 CI providers from all sectors in Switzerland [9]. To conduct this case study, we were granted access to the MELANI-Net quantitative log-file, as well as the qualitative results of a survey conducted in 2016. Each of the four main effects and the moderating role of trust are illustrated with real examples.

3.2 Reciprocity Expectation

Only half of the MELANI-Net participants are active on the platform. A first analysis of the log-file therefore suggested the existence of a free rider problem. The main reason for this seems to be that some participants have no information to share, or they believe that the information they hold is insufficiently relevant to justify sharing. This phenomenon may be unrelated to the provision of incentives or the free rider problem. However, it is possible that participants are using those arguments merely as an excuse to justify free riding. The most promising reciprocity incentive appears to be the sharing of best practice, i.e. the response to a cyber incident. The fear of free riders seems to be an important barrier to engaging in SIS.

[9] The interconnected 2008 global financial crisis bears several resemblances to what could happen in a major cyber "risk nexus" scenario.

3.3 Value Expectation

Participants appreciate the aggregated information received from MELANI, which is perceived as the main added-value of SIS. MELANI recently developed an information radar that provides CIs with an aggregated overview of the cyber threat landscape in Switzerland. This is the product that is most appreciated by the financial sector. It gives CIs providers a strong incentive to engage in a wider range of SIS activities, because in the future this could provide the basis for an SIS-Early Warning System (EWS), controlling and mitigating the cascade effect [1]. Participants reduce their costs through participation in SIS by benefiting from free MELANI consulting, IT support, and access to exclusive and timely Cyber Threat Intelligence (CTI) from the government.

3.4 Institutional Expectation

Most participants believe MELANI to be well managed. The financial sector regards the MELANI staff as reliable and credible. Switzerland offers a conducive environment for SIS, with a low corruption rate and strong trust between the government, the citizens, and the industry. More than half of the participants have been members of the platform for more than five years. In the financial sector, a clear taxonomy has been established and most participants believe that the platform has the right number of participants. However, impediments remain to the development of effective SIS, including legal issues that deter CI providers from engaging in SIS activities. These include antitrust laws, patent protection, national security laws, and data privacy laws, such as the Swiss banking secrecy laws. These make sharing of client-related data problematic, especially in cross-border or multi-jurisdictional contexts.

3.5 Reputation Expectation

After the public disclosure of the Heartbleed security bug in 2014, affected banks have experienced a decrease in their market share value. This event has confirmed that a security information leak might seriously damage the reputation of participants. Therefore, participants need to trust the ISAC on their reputation and anonymity preservation. As a result, the shared data has to be properly sanitised, in order to make sure that competitors will never use the shared information to damage other participants reputation.

3.6 The Moderating Role of Trust

The Swiss tradition of banking secrecy and non-cooperation in the financial sector is an established social norm that could act as an impediment to voluntary SIS. Surprisingly, the financial sector has the highest level of engagement in SIS activities, and was the sector most willing to join MELANI at its foundation a decade ago. This has allowed the sector to build trust over time, based on already existing relationships and the regular face-to-face meetings at workshops

or roundtables that take place between the MELANI staff and their contacts in the banks. The financial sector participants are therefore on average satisfied with the service received and appreciate the importance of MELANI to their own activities, the financial sector overall, and Switzerland's national security.

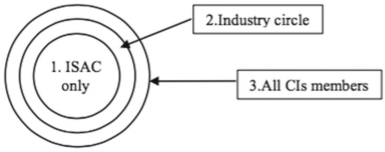


Fig. 2. MELANI trust-circles

The participants' willingness to engage in SIS activities reflects the levels of trust in three circles: first MELANI staff, second the industry circle, and third the total participants (Fig. 2). For each SIS transaction, participants can choose with which circle they wish to engage. These established relationships allow the four effects discussed above to be moderated by trust. As a result, the trust that has been established over a long period in this sector positively moderates our four expectations: reciprocity, value, institutional design, and reputation.

4 Discussion

Unlike in other OECD countries, CIs in Switzerland are not usually in competition, because most are state-owned, especially those in the energy sector. Privately-owned CI providers might have a different incentives and barriers than state-owned providers. This awaits further research and investigation. Moreover, CI operators sometimes fail to engage in SIS activities simply because they have no information to share. In contrast, other CIs may share greater amounts of security information because they have more security incidents to report. This phenomenon is unrelated to the provision of incentives or the free rider problem. Another possible bias that should be taken into account is the many SIS activities that take place outside MELANI-Net, for instance bilaterally, in peer-to-peer groups, or through industry-based ISACs. This is typically the case in the financial industry, with its successfully established FS-ISAC [30]. Additionally, Security solution vendors have recently created the Cyber Threat Alliance, in order to engage in mutual SIS. A further example is the newly created Industrial Control System - ISAC for threat intelligence sharing among nations[10]. As a result, we were unable to observe those SIS activities that are taking place outside of MELANI-Net.

[10] The goal of this platform is to bring together CI stakeholders to improve SIS at the international level.

5 Concluding Comments and Next Steps

We have provided a first blueprint for an innovative incentive-based SIS model, closing an important gap in the literature. This model can work as a complement to or extension of the Gordon-Loeb model. Further economic and social incentives could be used to extend the expectations indicators in our model. The design and analysis of such alternative indicators is a task for future research. For instance, the criticality of a CI operator could be linked to the frequency and intensity of SIS. Indeed, systemic risks (too big to fail) and the large externalities associated with high criticality might themselves provide an incentive to engage in extended SIS activities [29]. This paper constitutes conceptual work-in-progress. The developments in this paper, which focus on theory generation, will be complemented by empirical propositions testing and policy recommendations at national and international level. We hope that this study will inspire other researcher to extend and contribute to our model.

References

1. Alcaraz, C., Balastegui, A., Lopez, J.: Early warning system for cascading effect control in energy control systems. In: Xenakis, C., Wolthusen, S. (eds.) CRITIS 2010. LNCS, vol. 6712, pp. 55–66. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21694-7_5
2. Anderson, R., Fuloria, S.: Security economics and critical national infrastructure. In: Moore, T., Pym, D., Ioannidis, C. (eds.) Economics of Information Security and Privacy, pp. 55–66. Springer, Boston (2010). https://doi.org/10.1007/978-1-4419-6967-5_4
3. Anderson, R., Moore, T.: Information security: where computer science, economics and psychology meet. Philos. Trans. A Math. Phys. Eng. Sci. **367**(1898), 2717–2727 (2009)
4. Anderson, R., Moore, T., Nagaraja, S., Ozment, A.: Incentives and information security. In: Algorithmic Game Theory, pp. 633–649. Cambridge University Press, New York (2007)
5. Aviram, A., Tor, A.: Overcoming impediments to information sharing. Alabama Law Rev. **55**, 231 (2003–2004)
6. Bauer, J.M., Van Eeten, M.J.: Cybersecurity: stakeholder incentives, externalities, and policy options. Telecommun. Policy **33**(10), 706–719 (2009)
7. Campbell, K., Gordon, L.A., Loeb, M.P., Zhou, L.: The economic cost of publicly announced information security breaches: empirical evidence from the stock market. J. Comput. Secur. **11**(3), 431–448 (2003)
8. De Bruijne, M., Van Eeten, M.: Systems that should have failed: critical infrastructure protection in an institutionally fragmented environment. J. Contingencies Crisis Manag. **15**(1), 18–29 (2007)
9. Dunn Cavelty, M.: Cybersecurity in Switzerland. SpringerBriefs in Cybersecurity. Springer, Cham (2014)
10. Dunn-Cavelty, M., Suter, M.: Public-private partnerships are no silver bullet: an expanded governance model for critical infrastructure protection. Int. J. Crit. Infrastruct. Prot. **2**(4), 179–187 (2009)
11. ENISA: Good Practice Guide on Information Sharing. Report/study (2009)

12. ENISA: Incentives and Barriers to Information Sharing. Report/study (2010)
13. ENISA: Economic Efficiency of Security Breach Notification. Report/study (2011)
14. ENISA: Cyber Security Information Sharing: An Overview of Regulatory and Non-regulatory Approaches. Report/study (2015)
15. ENISA: Information sharing and common taxonomies between CSIRTs and Law Enforcement. Report/study (2016)
16. Vazquez, D.F., et al.: Conceptual framework for cyber defense information sharing within trust relationships, June 2012
17. Gal-Or, E., Ghose, A.: The economic incentives for sharing security information. Inf. Syst. Res. **16**(2), 186–208 (2005)
18. Ghernaouti, S.: Cyber Power: Crime, Conflict and Security in Cyberspace. EPFL Press, Burlington (2013)
19. Gordon, L., Loeb, M., Sohail, T.: Market value of voluntary disclosures concerning information security. Manag. Inf. Syst. Q. **34**(3), 567–594 (2010)
20. Gordon, L.A., Loeb, M.P., Lucyshyn, W.: Sharing information on computer systems security: an economic analysis. J. Account. Public Policy **22**(6), 461–485 (2003)
21. Gordon, L.A., Loeb, M.P., Lucyshyn, W., Sohail, T.: The impact of the Sarbanes-Oxley Act on the corporate disclosures of information security activities. J. Account. Public Policy **25**(5), 503–530 (2006)
22. Gordon, L.A., Loeb, M.P., Lucyshyn, W., Zhou, L.: Externalities and the magnitude of cyber security underinvestment by private sector firms: a modification of the Gordon-Loeb model. J. Inf. Secur. **06**(01), 24–30 (2015)
23. Gordon, L.A., Loeb, M.P., Lucyshyn, W., Zhou, L.: Increasing cybersecurity investments in private sector firms. J. Cybersecur. **1**(1), 3–17 (2015)
24. Grudzien, W., Hämmerli, B.: Voluntary information sharing. Technical report, Networking Information Security, Chapter 3 Voluntary Information Sharing (2014)
25. Haemmerli, B., Raaum, M., Franceschetti, G.: Trust networks among human beings. In: Multimedia Computing, Communication and Intelligence, May 2013
26. Harrison, K., White, G.: Information sharing requirements and framework needed for community cyber incident detection and response, November 2012
27. Hausken, K.: Information sharing among firms and cyber attacks. J. Account. Public Policy **26**(6), 639–688 (2007)
28. Laube, S., Böhme, R.: The economics of mandatory security breach reporting to authorities. In: Proceedings of the 14th Workshop on the Economics of Information Security (WEIS), Delft, Netherlands (2015)
29. Leu, P.O., Peter, D.: Case study: information flow resilience of a retail company with regard to the electricity scenarios of the Sicherheitsverbundsübung Schweiz (Swiss Security Network Exercise) SVU 2014. In: Rome, E., Theocharidou, M., Wolthusen, S. (eds.) CRITIS 2015. LNCS, vol. 9578, pp. 159–170. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-33331-1_13
30. Liu, C., Zafar, H., Au, Y.: Rethinking FS-ISAC: an IT security information sharing network model for the financial services sector. Commun. Assoc. Inf. Syst. **34**(1), 15–36 (2014)
31. Moran, T., Moore, T.: The phish-market protocol: securely sharing attack data between competitors. In: Sion, R. (ed.) FC 2010. LNCS, vol. 6052, pp. 222–237. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-14577-3_18
32. PricewaterhouseCoopers: information sharing and analysis organizations: putting theory into practice. Technical report (2016)
33. Suter, M.: The Governance of Cybersecurity: An Analysis of Public-Private Partnerships in a New Field of Security Policy. ETH, Zürich (2012)

34. von Hippel, E., von Krogh, G.: Open source software and the "Private-Collective" innovation model. Organ. Sci. **14**(2), 208–223 (2003)
35. Weiss, E.: Legislation to Facilitate Cybersecurity Information Sharing: Economic Analysis
36. Xiong, L., Liu, L.: PeerTrust: supporting reputation-based trust for peer-to-peer electronic communities. IEEE Trans. Knowl. Data Eng. **16**(7), 843–857 (2004)

Dynamic Risk Analyses and Dependency-Aware Root Cause Model for Critical Infrastructures

Steve Muller[1,2,3(✉)], Carlo Harpes[1], Yves Le Traon[2], Sylvain Gombault[3], Jean-Marie Bonnin[3], and Paul Hoffmann[4]

[1] itrust consulting s.à r.l., Niederanven, Luxembourg
{steve.muller,harpes}@itrust.lu
[2] University of Luxembourg, Luxembourg, Luxembourg
yves.letraon@uni.lu
[3] Telecom Bretagne, Rennes, France
{sylvain.gombault,jm.bonnin}@telecom-bretagne.eu
[4] Luxmetering G.I.E., Contern, Luxembourg
paul.hoffmann@luxmetering.lu

Abstract. Critical Infrastructures are known for their complexity and the strong interdependencies between the various components. As a result, cascading effects can have devastating consequences, while foreseeing the overall impact of a particular incident is not straight-forward at all and goes beyond performing a simple risk analysis. This work presents a graph-based approach for conducting dynamic risk analyses, which are programmatically generated from a threat model and an inventory of assets. In contrast to traditional risk analyses, they can be kept automatically up-to-date and show the risk currently faced by a system in real-time. The concepts are applied to and validated in the context of the smart grid infrastructure currently being deployed in Luxembourg.

1 Introduction

Cascading effects constitute a major issue in Critical Infrastructures (CI), for they are often unforeseen and have severe consequences [1]. With the launch of the 'Smart Grid Luxembourg', a project which started in 2012 and aims at deploying a country-wide smart grid infrastructure, the Grand-Duchy of Luxembourg is facing new risks in the energy distribution domain.

Although risk assessments can help to get a good overview of the threats faced by a CI operator, and comprehensive frameworks [2,3] exist for taking appropriate decisions, these methodologies deal with risk scenarios one-by-one and do not analyse the interactions between them. Indeed, in a risk analysis, threats may share common causes or consequences, which will be accounted for multiple times if each scenario is considered separately – see Fig. 1. Several authors have handled this issue; Aubigny et al. [4] provide a risk ontology for highly interdependent infrastructures and use Quality of Service (QoS) as weighting instrument. Foglietta et al. [5] present the EU-funded project *CockpitCI* that

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 163–175, 2017.
https://doi.org/10.1007/978-3-319-71368-7_14

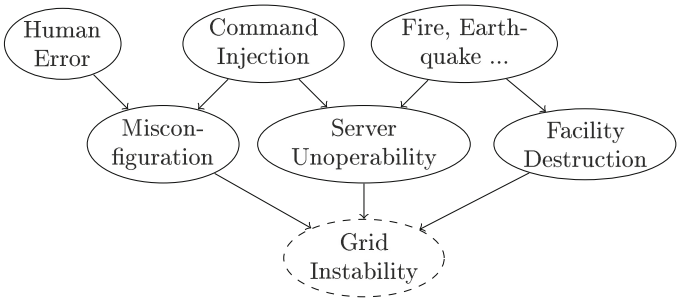


Fig. 1. Illustration of several causal effect chains that share common consequences. In the example of a smart grid operator, the availability of the electrical grid is of first priority, and the latter is endangered by various accidental, intentional and environmental causes. However, if each of the 6 depicted risk scenarios is analysed separately, the risk assessment would account 6 times for the grid instability, although only 3 causes (human error, command injection and fire) would be ultimately responsible for it.

identifies consequences of cyber-threats in real-time. Suh et al. [6] deliberate various factors that influence the relative importance of risk scenarios. Tong et al. [7] classify assets into three layers, viz. business, information and system. On the lowest layer, risk is computed traditionally as risk = impact × likelihood. Dependencies appear in the model as weighted impact added to the risk of dependent higher-level assets. Breier et al. [8] adopt a hierarchical approach and model dependencies as added-risk for parent assets; cyclically dependent components are not supported. Stergiopoulos et al. [9] use dependency graphs to describe cascading effects, but the risk of individual nodes is not expressed with respect to the graph structure.

In addition, such a risk analysis is only valid for the time it was created. However, whenever a security control is put in place or the network topology changes, the situation may change enormously, especially if the infrastructure is characterised by many interdependencies. Smart grids are especially affected by this issue, as smart meters are constantly re-organised with new living spaces being built. Since critical infrastructures should know the risk they are currently facing at all time, the risk analysis should be dynamically generated, taking into account all changes to the system. Several authors [10–12] propose algorithms for computing the likelihoods of interdependent risk scenarios, and thus, adopt a similar approach as in this paper, but all of them assume a fixed system and do not discuss the time needed to build the model, which is exponentially large in general and thus unsuitable for real-time updates.

The objective of this work is to provide a *systematic* approach for identifying the most critical parts in an cyber-physical system with the help of a risk analysis. Indeed, instead of relying on human intuition to properly cover all interdependencies in the risk assessment, the proposed model automatically deduces the risk scenarios from the dependencies provided as input by the risk assessor.

This way of proceeding has several advantages: for one, it accounts for each risk scenario to an appropriate extent (proportionally to its likelihood of occurring). Second, it eases updating or reusing the risk analysis generated by the proposed model, since all aspects are explicitly included in the model. Especially in the context of huge systems, this saves a lot of time. This work also paves the way for real-time risk monitoring (see Sect. 6) and deep analysis, such as determining the most critical sequence of cascading effects.

Section 2 defines the notions used throughout this paper, while Sect. 3 presents the risk model itself. Section 4 describes how to efficiently encode threats shared by multiple assets. The concepts developed in the paper are applied to the Luxembourgish smart grid in Sect. 5 and conclusions are drawn in Sect. 6.

2 Terminology

The (quantitative) risk modelling approach adopted by this paper relies on notions originating from probability theory. A *risk event* is a stochastic event which may occur with a certain probability. Two properties are associated with it:

- The *likelihood* (or *expected frequency*) of an event is defined to be the number of times it is estimated to happen per time unit. This notion is analogous to the probability of a probabilistic event, but different on principle: whereas traditional probabilities are bounded by 'absolute certainty (100%)', the likelihood (frequency) can be arbitrarily large.
- The *impact* describes the consequences of a risk event that are estimated to arise whenever the latter occurs. For simplicity, this paper focuses on financial damage (in €), but the latter can be substituted by any other impact measure instead (such as reputation, number of affected people ...), as long as this is done consistently throughout the whole risk assessment.

Risk is then defined to be the expected impact with respect to the likelihood. Note the analogy to the 'expected value' in probability theory:

$$\underbrace{\text{risk}}_{\text{unit: €/time}} = \underbrace{\text{likelihood}}_{\text{unit: 1/time}} \times \underbrace{\text{impact}}_{\text{unit: €}}.$$

In a risk analysis composed of many scenarios, the total risk is the sum of all partial risks (engendered by each event):

$$\text{risk} = \sum_{i:\text{event}} \text{risk}_i = \sum_{i:\text{event}} \text{likelihood}_i \times \text{impact}_i.$$

The collection of events in a risk analysis can be embedded into a directed graph $G = (V, E)$ as follows. Whenever two events α and β bear a causal relation, in the sense that α is likely *to cause* β to happen, an edge is drawn from vertex α to vertex β, which is written as "$\alpha \to \beta$". An example of a causal graph is depicted in Fig. 1. When acyclic, such graphs are called *Bayesian networks* [13], but the 'cycle-free' assumption is not necessary in the context of this paper.

In order to encode the extent to which two events are dependent, let $p : E \mapsto [0, 1]$ be the map which associates to each edge $(\alpha \to \beta)$ the *probability* that α causes β directly. Note that this is *not* the same as the probability that β occurs given α occurs; indeed, suppose the graph consists of a single chain $\alpha \xrightarrow{0.1} \gamma \xrightarrow{0.5} \beta$, then $\Pr[\beta|\alpha] = 0.05$, but $p(\alpha \to \beta) = 0$ since α cannot cause β directly.

The graph $G = (V, E, p)$ consisting of the vertex set V of events, the (directed) edge set E of causal relations and the probability map p, is called the *dependency graph* associated with the risk analysis. Vertices without parents are called *root causes* and are supposed to occur independently of one another. Denote the set of root causes by $V_R \subset V$. Those will be the events that ultimately trigger any risk scenario encoded in the graph.

3 Risk Assessments Using the Dependency-Aware Root Cause (DARC) Model

The model described in Sect. 2 is called Dependency-Aware Root Cause (DARC) Model and was introduced by Muller et al. in [14]. It is designed for conducting a risk analysis in complex systems featuring many interdependencies (from a physical, logical or management point of view), such as cyber-physical systems.

Whereas scenarios in traditional risk analyses usually cover direct consequences *and* all cascading effects, events in the DARC model are supposed to be as specific as possible. In fact, in order to eliminate any redundancy or repetition in a risk analysis, all involved intermediate events should be made explicit (by adding them as vertices to the dependency graph), so that the precise nature of the dependencies can be properly encoded (using directed edges); see Fig. 2 for an example.

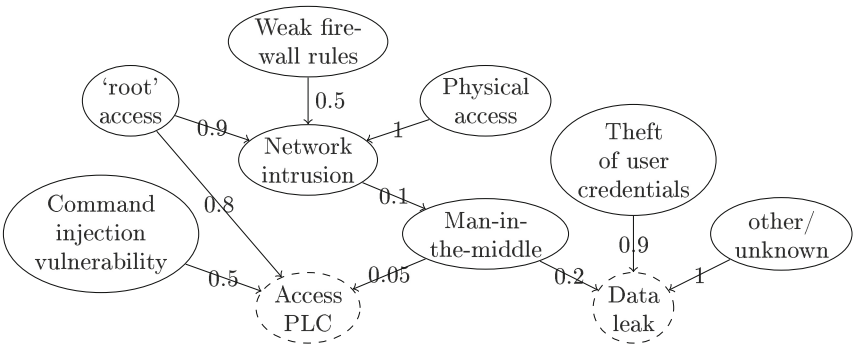


Fig. 2. Example of two security events ('full control over a Programmable Logic Controller (PLC)' and 'leak of sensitive data') that share common causes. Edge labels represent the probability map $p : E \to [0, 1]$.

The motivation behind the dependency graph is to identify all possible root causes that can bootstrap a chain of effects in a risk scenario. Note that in order to yield sensible results, the model requires the set of causes to be exhaustive for each node, but determining *all* possible (significant) causes can be hard and time-consuming. A separate node accounts for any uncertainty resulting from unknown, rare or minor causes; notice the node 'other/unknown' in Fig. 2. Since its likelihood is usually unknown, an estimated upper bound shall be used instead (which should be lower than the likelihood of all other parent nodes).

Once a dependency graph has been established, the risk assessor estimates the probability p of each edge and likelihoods $\mathcal{L}$ (expected frequencies) of the *root events only*, since the remaining ones can be deduced from the dependency graph and its probability map p. Indeed, say $\mathcal{L}(\cdot)$ denotes the likelihood[1] of an event and $V_R \subset V$ represents the set of (independent) root causes, then

$$\forall \alpha \in V \quad \mathcal{L}(\alpha) = \sum_{r \in V_R} \underbrace{\mathcal{L}(r)}_{\substack{\text{estimated} \\ \text{manually}}} \cdot \underbrace{\Pr[r \text{ eventually causes } \alpha]}_{\substack{\text{programmatically} \\ \text{deduced from model}}}.$$

Define $\mathcal{P}(r, \alpha) := \Pr[r \text{ eventually causes } \alpha]$ for root causes r and any event α. Efficient randomized algorithms exist [14] which can compute these values $\mathcal{P}$. Now, if $\mathcal{I}(\cdot)$ denotes the (estimated) impact of an event, then the global risk is

$$\text{risk} = \sum_{\alpha \in V} \mathcal{I}(\alpha) \cdot \mathcal{L}(\alpha) = \sum_{\alpha \in V} \mathcal{I}(\alpha) \cdot \sum_{r \in V_R} \mathcal{L}(r) \cdot \mathcal{P}(r, \alpha), \tag{1}$$

where $V_R \subset V$ denotes the set of root causes. The benefit of this reformulation is that it is no longer necessary to know the likelihood of *all* events, but only the one of *root causes*. When the involved maps are interpreted as vectors and matrices, the previous line is equivalent to

$$\text{risk} = \underbrace{\mathcal{I}^\top}_{1 \times V \text{ matrix}} \cdot \underbrace{\mathcal{P}^\top}_{V \times V_R \text{ matrix}} \cdot \underbrace{\mathcal{L}|_{V_R}}_{V_R \times 1 \text{ matrix}}, \tag{2}$$

where $\mathcal{L}|_{V_R}$ denotes the restriction of $\mathcal{L}$ to V_R. Note that $\mathcal{P}$ is entirely deduced from the dependency graph, whereas $\mathcal{I}$ and $\mathcal{L}$ have to be estimated by a risk assessor.

Writing Eq. (2) in matrix form permits to embed it into a spreadsheet file where it can be further edited, formatted or analysed (e.g. using diagrams).

4 Risk Taxonomy for Critical Infrastructures

A major drawback of dependency-aware risk analyses is the increase in size of the risk description, resulting from the additional values to be estimated.

[1] From a probability theoretic point of view, $\mathcal{L}$ is an expected value (a frequency, in fact) and not a probability. Formally, $\mathcal{L}(\alpha) = \sum_r \mathbb{E}\big[\mathcal{L}(r) \cdot \mathbb{1}[r \text{ causes } \alpha]\big]$, where $\mathcal{L}(r)$ is a non-probabilistic constant, for the probability space only includes edges, not nodes.

However, many assets share the same threats, which gives rise to information redundancy in the graph. To avoid this, and thus to reduce the size of the model, the risk taxonomy presented in this section aims at providing a framework where threats can be defined generically for a set of assets. It extends the DARC model introduced in [14] and briefly presented in Sect. 2.

In fact, most (yet not all) security events are related to a (physical or digital) asset. For this reason, it makes sense to group assets facing similar threats, so as to express the dependencies between *asset classes* as causal relations between *threats* acting upon them. For instance, the natural dependence of sensitive data on its database is contained in the statement that threats to any software also put data managed by it at risk. At the same time, one should be able to specify risks for a specific asset, but not for the whole class. For instance, when (specific) login credentials are stolen, a (specific) application can be accessed, whereas this is not the case when other kinds of information are stolen.

The following asset classes have been identified; see Fig. 3 for an overview.

- A *Network* is a closed environment of interconnected devices. Communication with the outside is possible, but subject to rules (often imposed by a firewall). Compromising a network amounts to compromising the flows provided by devices inside that network.
- A *Device* is any physical hardware. Devices are subject to mechanical damage and physical access.
- An *Application* is the functional counterpart of a device; it covers programs, operating systems and firmwares. Unlike a device, an application is threatened by software vulnerabilities and remote attacks. The idea is to separate the soft- and hardware layer, since both are exposed to different risks.
- A *Flow* transports data from one application to another over a network. Flows can be manipulated/blocked and tapped.

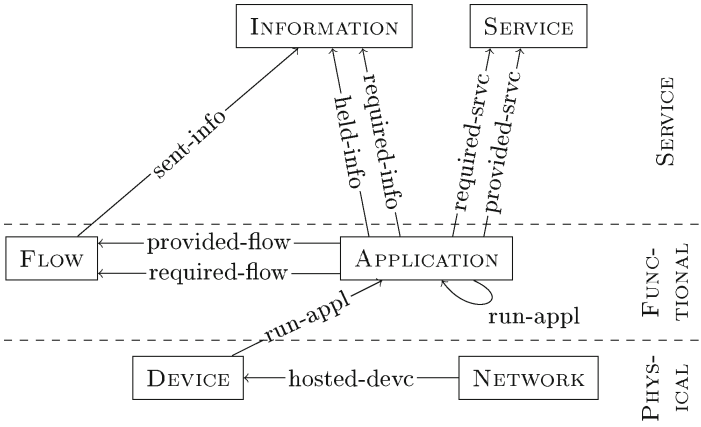


Fig. 3. Class diagram representing the taxonomy of assets involved in a risk analysis, grouped by layer. Read '$A \xrightarrow{\xi} B$' as 'B is a ξ of A'.

– *Information* comprises all kind of data – including contents of a database, keys, passwords and certificates. Information can be destroyed, be tampered with or leak.
– A *Service* consists in a *goal* that one or more applications are pursuing. They represent the purpose of having a particular application running in the system, and are often linked to business processes. Unavailability constitutes the main threat that services are exposed to. For instance, a firewall ultimately prevents network intrusions; if the associated *service* fails, this may have an impact on the integrity of all flows within that network.

The idea of classifying assets is not new; Tong et al. [7] have suggested systematising assets on different 'levels' (system, informational, business), but their motivation is primarily to do a risk assessment on each layer and interconnect the results, whereas the DARC model goes one step further by linking risk scenarios directly to asset classes. In the critical infrastructure context, Aubigny et al. [4] adopt a similar approach for interdependent infrastructures (using layers 'implementation', 'service' and 'composite'). The IRRIIS Information Model [15] separates the topological, structural and functional behaviour of dependencies and analyses their interactions across these three layers. In contrast to all of these works, this paper uses asset classes only to define similar threats for similar assets, but considers each asset individually when computing the risk.

4.1 Dependency Definition Language

The encoding of the causal relations is achieved using a simple mark-up language that is based on the GraphViz[2] DOT syntax. They are expressed as

"$node_A$" -> "$node_B$" [p = x];

which reads as

If $node_A$ occurs, it causes $node_B$ to occur with probability x;

where $node_A$ and $node_B$ are the IDs of the respective nodes in the dependency graph, and $x \in [0, 1]$. The syntax is extended in such a way that the node IDs can contain placeholders which match whole asset classes. Placeholders are enclosed with angular brackets < and > and are of the following form.

< *assetclass* >
< *assetclass* . *selector* . *selector* ... >
< *assetclass* # *filter* . *selector* # *filter* ... >

where

– *assetclass* is one of `net`, `dev`, `app`, `flow`, `inf`, `svc`;

[2] GraphViz is an open-source graph visualization software. For more information on the DOT language, see http://graphviz.org/content/dot-language.

- *selector* is one of `hosted-devc`, `run-appl`, `provided-flow`, `required-flow`, `sent-info`, `held-info`, `required-info`, `required-svc`, `provided-srvc`, depending on the context, meant to navigate through the class model depicted in Fig. 3;
- *filter* is a keyword restricting the choice of selected nodes (e.g. `#fw` for only selecting firewall devices, `#key` for only selecting keys).

For instance, the following line encodes the fact that if an attacker has full control over some (any) application, there is a 10% chance that he gets access to any associated secret keys.

```
"control of <app>" -> "leak of <app.held-info#key>"
[p = 0.1];
```

4.2 Generating the Dependency Graph

In order to be able to deduce the final dependency graph from the definitions, an inventory describing the assets themselves needs to be created. Note that organisations usually have such an inventory, especially if they have a security management systems. The inventory itself can be encoded as a directed graph in the DOT syntax as well.

```
# Defining assets as ID (for reference) and label (for display)
"info:custdat" [label = "Customer data"];
"appl:db" [label = "Database"];

# Defining relations between assets (by their ID's)
"appl:db" -> "info:custdat" [label = "held-info"];
```

Node IDs should be prefixed by the asset class (e.g. `info:` for information assets) so that the class can be inferred from the ID. Edge labels express the kind of relation which the second (right) node maintains to the first (left); in the example above, customer data (`info:custdat`) is information held (`held-info`) by the database (`appl:db`).

Once the dependencies have been defined and the inventory has been established, both inputs can be programmatically combined to yield the dependency graph – in the use-case described below, this is achieved using a *Python* script.

5 The 'Smart Grid Luxembourg' Use-Case

The 'Smart Grid Luxembourg' (SGL) project aims at innovating the electricity and gas transmission networks by deploying a nation-wide 'smart' grid infrastructure. By law, starting from July 2016, every new electricity or gas meter deployed in Luxembourg will be a smart meter. At the end of the project, by 2020, the system will count 380.000 smart meters. In the following, the concepts presented in the previous sections are applied to SGL.

5.1 Compiling a Dependency-Aware Inventory

In a first phase, the inventory of all relevant assets has been compiled, covering hard- and software, physical wiring, network flows, database tables and their contents, certificates, other kinds of information and the services provided by the various applications. Figures 4, 5 and 6 provide anonymised variants of the complete, confidential graphs.

The Luxembourgish smart grid manages its own Public Key Infrastructure (PKI), so as to guarantee complete independence of any external providers. The certificates in a PKI bear a natural dependency hierarchy with them, in the sense that compromising any certificate authority (CA) allows one to reproduce any dependent certificates and thus, ultimately, to undermine an encrypted communication channel.

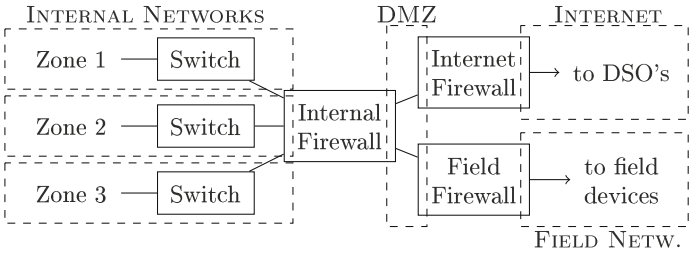


Fig. 4. Anonymised network diagram of the central system architecture showing devices and their affinity to the respective networks. *DSO = Distribution System Operator*; *DMZ = DeMilitarised Zone*; field devices include data concentrators and meters.

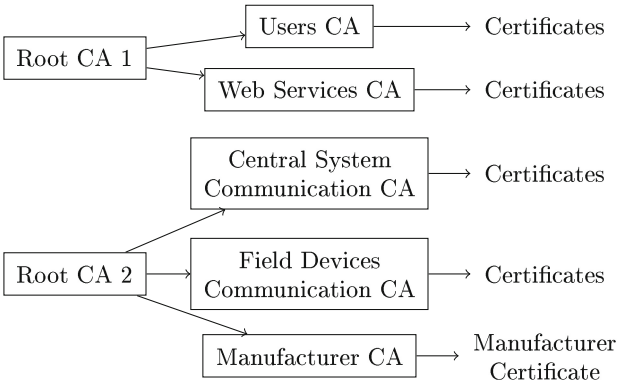


Fig. 5. Anonymised hierarchy of the PKI. *CA = Certificate Authority.*

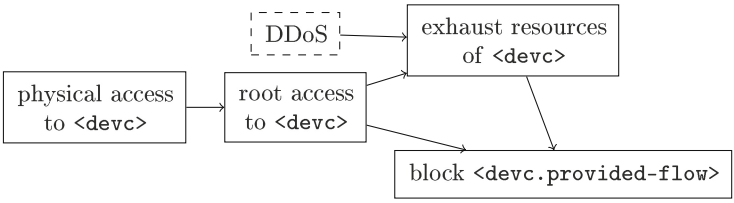


Fig. 6. Excerpt of a dependency graph containing placeholders. Note the singleton node 'DDoS' which is not associated to any particular asset.

5.2 Threat Model

The second phase consisted in identifying all possible threats faced by the system and encoding them properly in a dependency graph (with placeholders).

The elaborated threat model is based, on the one hand, on other research work by Grochocki et al. [16] and ENISA [17], who determine the threats faced by a smart grid infrastructure. On the other hand, the Smart Grid Luxembourg specific dependencies could be extracted from former risk analyses and from documentation material that was kindly provided by Luxmetering.

It turns out that a large portion of the threats can be expressed as a tuple consisting of an asset (class) and an endangered security property (such as confidentiality, integrity or availability). For instance, the generic risk scenarios faced by applications comprehend malfunctioning, unauthorized access (e.g. by faking login credentials), lose of control (e.g. due to code injection) and denial of service. The risk scenarios that are not directly associated to an asset (such as distributed denial-of-service or fire), are added as singleton nodes to the graph.

5.3 Generation of the Dependency Graph

Once the threat model was set up, the final dependency graph could be programmatically derived from the inventory. For this purpose, a small script written in *Python* reads in the dependency definitions and applies them to the inventory (by replacing all placeholders by the respective IDs of the assets in the inventory).

The algorithm developed in [14] then allows one to identify all root causes, that is to say, to find those events which are ultimately responsible for all risk scenarios in the threat model. Moreover, it determines the probabilities that *each* of these root causes eventually leads to *each* of the other events by cascading effect – thus computing the probability matrix $\mathcal{P} : V_R \times V \to [0, 1]$ introduced in Sect. 3.

5.4 Results

The inventory consists of 12 different devices (each with multiplicity), 9 networks, 37 applications, 43 flows, 14 data sets, 26 certificates, 9 sets of credentials and 18

services. The generic dependency graph encoding the threat model (with placeholders) has 53 nodes and 104 edges. The time needed for conducting the risk analysis for Luxmetering is composed as follows:

Gather asset inventory from documentation material and past (static) risk analyses	18 h (2 md)
Define (generic) dependency graph	30 h (4 md)
Estimate $\mathcal{P}$ and $\mathcal{L}$	9 h (1 md)
Fine-tuning of the model	7 h (1 md)
Total	64 h (8 md)

Since the generic dependency graph contains (almost) no SGL-specific information, it can be easily recycled for other, similar use-cases.

Computing the full final dependency graph (consisting of 502 nodes and 1516 edges) took 3.79 s on a 2.0 GHz dual-core processor (which includes the parsing time of the inventory files). The probability matrix $\mathcal{P}$ has been computed using a *C#* implementation of the algorithm proposed in [14]; it is composed of 502 rows (as many as nodes) and 25 columns (root causes), comprising thus a total of 12550 probability values. Its computation took 39.14 s.

The following 25 root causes were read off the model:

- phishing, social engineering,
- bad input validation, XSS, CSRF, broken authentication, buffer overflow,
- DDoS, jamming, smart meter intrusion, physical access to facility,
- data center incidents (fire ...), device construction faults, mechanical attrition
- and 11 SGL-specific attacks.

The most critical risks read off from the risk matrix were the following (most critical first):

1. manipulation of billing data,
2. disclosure of customer data, requiring reporting to authorities and informing customers,
3. power outages,
4. forensics,
5. loss of smart meter configuration data, which involves reconfiguring all 380.000 devices, and
6. loss of billing data.

In total, a yearly risk of an order of magnitude of 300 k€ was estimated.

6 Conclusion and Future Work

The Darc model allows one to perform a risk analysis that accounts for the interdependencies in complex infrastructures such as cyber-physical systems.

This paper extends the Darc risk model developed in [14] by a risk taxonomy and describes an automated approach to generate a dependency graph from an

asset inventory. That way, any changes in the infrastructure can be automatically replicated to the risk analysis, which renders it dynamic.

This is a first step towards dynamic risk management. Future work is devoted to the question how other sources (apart from the inventory) of real-time information can be used to automatically update the parameters of the risk analysis. Practical examples include intrusion detection systems, firewalls or patch management tools, which allow to infer the likelihoods of certain root causes from historical data (such as log files) collected by these sources.

In this spirit, such real-time information can also be used to directly compute the risk currently faced by an organisation. In a dashboard-like interface, technical alerts can be translated to the equivalent losses that an intrusion or fault can cause, expressing technical issues in a language understood by decision-makers. Such an interface is planned to be implemented in TRICK SERVICE[3], an existing web application developed by *itrust consulting* designed to conduct risk assessments according to ISO 27005.

Acknowledgements. This work was supported by the Fonds National de la Recherche, Luxembourg (project reference 10239425) and was carried out in the framework of the H2020 project 'ATENA' (reference 700581), partially funded by the EU.

References

1. Rinaldi, S.M.: Modeling and simulating critical infrastructures and their interdependencies. In: Proceedings of the 37th Annual Hawaii International Conference on System Sciences, p. 8. IEEE (2004)
2. International Organization for Standardization: ISO/IEC 27019 (2013)
3. Bundesamt für Sicherheit in der Informationstechnik (BSI): IT-Grundschutz (2005)
4. Aubigny, M., Harpes, C., Castrucci, M.: Risk ontology and service quality descriptor shared among interdependent critical infrastructures. In: Xenakis, C., Wolthusen, S. (eds.) CRITIS 2010. LNCS, vol. 6712, pp. 157–160. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21694-7_14
5. Foglietta, C., Panzieri, S., Macone, D., Liberati, F., Simeoni, A.: Detection and impact of cyber attacks in a critical infrastructures scenario: the CockpitCI approach. Int. J. Syst. Syst. Eng. **4**(3–4), 211–221 (2013)
6. Suh, B., Han, I.: The IS risk analysis based on a business model. Inf. Manag. **41**(2), 149–158 (2003)
7. Tong, X., Ban, X.: A hierarchical information system risk evaluation method based on asset dependence chain. Int. J. Secur. Appl. **8**(6), 81–88 (2014)
8. Breier, J.: Asset valuation method for dependent entities. J. Internet Serv. Inf. Secur. (JISIS) **4**(3), 72–81 (2014)
9. Stergiopoulos, G., Kotzanikolaou, P., Theocharidou, M., Lykou, G., Gritzalis, D.: Time-based critical infrastructure dependency analysis for large-scale and cross-sectoral failures. Int. J. Crit. Infrastruct. Prot. **12**, 46–60 (2016)
10. Baiardi, F., Sgandurra, D.: Assessing ICT risk through a Monte Carlo method. Environ. Syst. Decis. **33**(4), 486–499 (2013)

[3] https://www.itrust.lu/products/.

11. Wang, L., Islam, T., Long, T., Singhal, A., Jajodia, S.: An attack graph-based probabilistic security metric. In: Atluri, V. (ed.) DBSec 2008. LNCS, vol. 5094, pp. 283–296. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-70567-3_22
12. Homer, J., Ou, X., Schmidt, D.: A sound and practical approach to quantifying security risk in enterprise networks. Kansas State University Techn. Report (2009)
13. Pearl, J.: Causality: Models, Reasoning, and Inference. Cambridge University Press, New York (2000)
14. Muller, S., Harpes, C., Le Traon, Y., Gombault, S., Bonnin, J.-M.: Efficiently computing the likelihoods of cyclically interdependent risk scenarios. Comput. Secur. **64**, 59–68 (2017)
15. Klein, R.: Information modelling and simulation in large dependent critical infrastructures – an overview on the european integrated project IRRIIS. In: Setola, R., Geretshuber, S. (eds.) CRITIS 2008. LNCS, vol. 5508, pp. 131–143. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-03552-4_12
16. Grochocki, D., Huh, J.H., Berthier, R., Bobba, R., Sanders, W.H., Cárdenas, A.A., Jetcheva, J.G.: AMI threats, intrusion detection requirements and deployment recommendations. In: 2012 IEEE Third International Conference on Smart Grid Communications (SmartGridComm), pp. 395–400. IEEE (2012)
17. ENISA: Communication network interdependencies in smart grids (2016)

Selecting Privacy Solutions to Prioritise Control in Smart Metering Systems

Juan E. Rubio(✉), Cristina Alcaraz, and Javier Lopez

Department of Computer Science, University of Malaga,
Campus de Teatinos s/n, 29071 Malaga, Spain
{rubio,alcaraz,jlm}@lcc.uma.es

Abstract. The introduction of the Smart Grid brings with it several benefits to society, because its bi-directional communication allows both users and utilities to have better control over energy usage. However, it also has some privacy issues with respect to the privacy of the customers when analysing their consumption data. In this paper we review the main privacy-preserving techniques that have been proposed and compare their efficiency, to accurately select the most appropriate ones for undertaking control operations. Both privacy and performance are essential for the rapid adoption of Smart Grid technologies.

Keywords: Smart Grid · Data privacy · Control · Metering

1 Introduction

In comparison with the traditional electric grid, the Smart Grid (SG) enables a more accurate monitoring and prevision of energy consumption for utilities so they can adjust generation and delivery in near real-time. Users also receive detailed consumption reports that can help them to save money by adapting their power usage to price fluctuations. The Advanced Metering Infrastructure (AMI) is a smart metering system that makes this possible by processing a huge data collection generated at a high frequency [1]. This information can then be analysed to draw surprisingly accurate conclusions about customers.

In order to preserve privacy, consumption data should not be measured. However, this is not feasible: the energy supplier needs to know the sum of the current electricity consumption of all its customers (or a group of them concentrated in a certain region) primarily to perform monitoring operations and Demand Response. Secondly, the supplier also needs to collect attributable information to know the total consumption of a single customer over a given time period (e.g., a month), in order to calculate the bill. As a result, privacy-preserving techniques must be implemented to prevent the Energy Service Provider (ESP in the following) from checking the current energy consumption of a single customer.

There does not seem to be a clear difference in research between protocols that address privacy when metering for billing and those which concern metering

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 176–188, 2017.
https://doi.org/10.1007/978-3-319-71368-7_15

for monitoring the grid and handling Demand Response. Even though some of the surveyed protocols enable both operations, this paper divides them into those which principally concentrate on providing privacy when carrying out billing operations and those which focus on the monitoring tasks. However, it is equally useful to assess not only how user privacy is protected but also the impact of these mechanisms on the performance of the collection and supervision systems (e.g., not saturating net communications or running hard time-consuming protocols).

The techniques discussed here make use of the traditional Privacy Enhancing Technologies (PETs) [2,3]:

- **Trusted Computation (TC)**: the Smart Meter (SM) itself or a third party is entrusted to aggregate consumption data before it is sent to the energy supplier.
- **Verifiable Computation (VC)**: the smart meter or a third party calculates the aggregated data and sends proof to the provider to ensure its correctness.
- **Cryptographic Computation (CC)**: by using secret sharing or homomorphic cryptographic schemes, so the provider can only decrypt the aggregate of consumption data.
- **Anonymization (Anon)**: removal of the smart meter identification or substitution with pseudonyms.
- **Perturbation (Pert)**: random noise is deliberately added to the measurements data while keeping it valid for control purposes.

In this paper, we present a description and analysis of some of the most representative solutions that address privacy in the context of smart metering, emphasising their effects in control and supervising tasks. In addition, other privacy technologies based on the use of batteries (denoted as *Batt* in the tables) to mask the energy usage will also be considered. The aim is to guide both customers and grid operators in the search for techniques that fit their needs while balancing both privacy and control.

To accomplish the analysis, all techniques are presented as follows: In Sect. 2 the privacy and performance properties analysed in the solutions are introduced. In Sect. 3 the techniques are categorised according to the PETs they integrate and their suitability for billing and monitoring operations, and then they are analysed. Conclusions and future work are discussed in Sect. 4.

2 Privacy and Automation Properties

Currently, numerous approaches related to data privacy in SG [4,5], can be found but not all of them consider aspects related to the efficient management of the real demand. To do this, it is necessary to consider a set of essential properties concerning not only privacy but also data monitoring itself, so as to find a desirable trade-off between security and automation. For example, privacy schemes must ensure the user's privacy and security in the supervision tasks without producing disruptions or delays in the data collection processes. Given this and the need to preserve the consumption data and its availability

for the control, the following lines describe the set of essential properties needed to select the most suitable privacy techniques for SG environments.

To organise the intrinsic features of the privacy in relation to the control, it is first necessary to consider the main control requirements defined in [6]. Any control system in charge of supervising specific areas must take into account: the performance in real time (privacy solutions must not interfere with the monitoring tasks), sustainability in terms of maintainability and configurability, and dependability and survivability in relation to reliability and security. Based on these criteria, the goal is to define the different properties related to the privacy that can affect the monitoring tasks, and therefore the requirements of automation.

Real-time Performance: addresses the operational delays caused by the processing of information, application of techniques and the transference of the data to control utilities. When handling this control requirement with respect to the features of the privacy solutions, these fundamental properties should be considered:

- **Speed**: as some protocols discussed in this paper have not been implemented, speed cannot be measured in quantitative terms. An estimation can be made by counting and considering all the communication and cryptographic steps required to run it.
- **Storage**: subject to the excess of operations and the massive storage, which can require extra resources to maintain meter values.
- **Communication overhead**: the excess of communication and the data transference rate (e.g., for synchronisation) may hamper the data recollection and the supervision of the area.
- **Synchronisation**: it focuses on the time when data streams are being sent from the producer (i.e., the smart meter) to the consumer (i.e., the energy service provider). Whereas certain protocols may require all data producers to send it at the same time, in others the data producers send it independently of each other. It must be noted that the use of synchronisation increases the complexity of the protocol.

Sustainability: defined as "that development that is able to meet the needs of the present without compromising the ability of future generations to meet their own needs" in [7]. Namely, the privacy techniques must not provoke compatibility problems, conflicts or errors; and for this, it is necessary to consider aspects related to the configuration, maintainability and updating of the techniques.

- **Configurability**: related to the easy way to carry out not only the commissioning and setup phase of the PETs but also their configurability throughout its life-cycle.
- **Maintainability**: this property comprises updating and upgrading measures, where the updating or upgrading process must not imply a reduction in the control tasks.

Dependability: can be defined as "the ability of the system to properly offer its services on time, avoiding frequent and several faults" in Al-Kuwaiti *et al.* [8], and includes reliability and security as main properties. However, we only address the reliability since the security is already part of the privacy solutions. In this category, we consider:

- **Fault-tolerance**: some of the protocols discussed here may be robust enough as to bear unlimited software and hardware failures or just a certain number of them (or no failures at all).
- **Aggregate error**: a protocol can be considered as exact or noisy depending on the presence of errors in the aggregated data that are a result of metering failures or a consequence of applying perturbation to preserve privacy.

Survivability: capability of a system to fulfil its mission and thus address malicious, deliberate or accidental faults in a timely manner. It also includes security against external attacks, which is briefly analysed in Sect. 3.1. However, for the purposes of the work presented here, where we prioritise control, **resilience** is specifically studied, which allows the system to continue its services when part of its security is compromised.

On the other hand, it is essential to take into account the mode of configuration of the nodes and the data management. Depending on the scenario, the communication model can vary, as it defines how smart meters (producers) are connected to utilities (e.g., for control). There are different communication models, from distributed systems to hierarchical or decentralised systems composed of aggregators or trusted third parties. In addition, and related to the communication model, it is also important to take into account the type of commissioning and setup needed to specify the group management, and data spatial distribution to determine how the data subsets are aggregated spatially (over a set of data producers) or temporally (over a set of one data producer's data items).This feature is also known as the aggregate function.

3 Selecting Techniques: Analysis and Discussion

In this section, the privacy-reserving metering solutions are assessed. The current literature has been reviewed to provide the most discussed techniques of each of the PETs presented in the introduction, resulting in ten protocols. Firstly, an introduction to their main architecture and privacy features is given, and then a discussion is proposed to compare the efficiency of each one according to the control requirements indicated in Sect. 2. Note that since most of the techniques lack a real implementation, the comparison is done based on an estimation of the properties in each solution.

3.1 Analysis of Privacy Techniques

As introduced, this subsection gives a brief overview of the main solutions proposed in the literature, focusing on the aggregation and communication model

that they put into practice, along with the underlying security. Techniques are classified into those which are mainly suitable for monitoring and those that address privacy when performing billing operations. Finally, all these characteristics are presented in Table 1, where the solutions appear, ordered by their implemented PET.

Among the **privacy techniques for billing operations**, the following solutions have been considered: Bohli *et al.* [9] propose a model where a Trusted Third Party (TTP) is introduced to aggregate (i.e., sum all smart meters' readings) before sending them to the ESP. Specifically, SMs transmit their data through an encryption channel to the TTP, which sums up individual consumption for each smart meter at the end of the billing period and also informs the ESP about the current status of that part of the grid. It can be considered efficient as it uses symmetrical encryption (usually AES) and robust since it can detect the presence of fake groups (i.e., sets of SMs controlled by the ESP that emit default values in order to isolate the real customer's consumption). However, it introduces some communication overheads due to the permanent data submitted by SMs to the TTP to monitor the electricity consumption of a certain area.

Molina-Markham *et al.* [10], to the contrary, describe a Zero-Knowledge (ZK) protocol that allows a prover (the smart meter in this case) to demonstrate the knowledge of a secret (the power readings needed to compute the bill) to the verifier (the ESP) without revealing the electricity usage or the ability to under-report it. In addition to this, neighbourhood gateways are placed between the SMs and the ESP to relay aggregated power readings corresponding to an area without disclosing any particular origin, enabling Demand Response operations by this means. Zero-Knowledge protocols are computationally expensive, although the communication between the SM and the ESP takes place only once per billing cycle.

Jawurek *et al.* [11] also specify another Zero-Knowledge protocol for billing based on Pedersen commitments [12]. It introduces a plug-in Privacy Component (PC) between the SM and the ESP that intercepts consumption data and sends the provider signed commitments and the final calculation together with the random parameters used to create the Pedersen commitments from individual measurements. Taking advantage of the homomorphic property of this schema, the ESP can effectively check the bill validity computing the calculation on the received commitments, which result in a new commitment of the bill amount and random numbers presented. It is worth commenting that the PC is invisible to the SM and it calculates the final price. Also, it does not have to be trustworthy, since the VC protocol itself ensures a correct bill calculation, and therefore it can be implemented easily with no special hardware-protected components.

Lemay *et al.* [13] propose isolating the bill calculation in the smart meter by using a Trusted Platform Module (TPM). More specifically, its architecture is composed by independent virtual machines intended to perform diverse applications like billing or Demand Response. A hypervisor controls the access to the hardware (hence the power measurements) and integrity and confidentiality

are guaranteed through remote attestation, which proves to the provider that the hardware and software being used are deemed as trustworthy. To achieve this, the device includes hardware-protected storage, cryptographic modules and other tamper detection components. In terms of control requirements, this solution reduces the amount of information transmitted between SM and ESP, as all data processing (i.e., the bill calculation) occurs at the point of the origin of the data. The TPM also allows the service provider and the customer to run their own applications relying on the strong isolation that virtualisation technology provides.

Likewise, Kalogridis *et al.* [14] define the concept of 'load signature moderation' to shape the electricity consumption so it does not expose any sensitive data. They propose the introduction of a decentralised energy production system within the household, so power can be dynamically drawn from re-chargeable batteries and other energy storage and generation devices. Thus, actual energy usage curves can be changed, hidden, smoothed, obfuscated or emulated. This solution protects against attackers that have a physical control over the SM and does not depend on specific grid architectures or trust relationships, while being compatible with other additional mechanisms and enabling grid monitoring. However, it requires extra computation when the battery is almost charged or empty, in order to keep masking the consumption and hence preserving privacy.

As for the **privacy techniques for monitoring operations**, we hightlight the Efthymiou *et al.*'s work [15]. They establish a division between two kinds of data generated by the SM. On the one hand, high-frequency measurements (e.g., collected every 15 min) transmitted to the ESP to perform monitoring operations over a set of SMs, which have to be pseudoanonymised due to the information they provide about a user's private life. On the other hand, low-frequency metering data (e.g., collected monthly) that is attributable for billing purposes. An identification is assigned to each type: HFID (High-Frequency ID) and LFID (Low-Frequency ID), respectively. Whilst high-frequency data is sent to an escrow with the HFID and remains unknown to the ESP, low-frequency data is disclosed publicly and is linked to LFID. The escrow can be queried by the ESP to verify the connection between a HFID/LFID pair. Its principal disadvantages are the complex setup process and the strong data privacy policy the escrow has to comply with.

Petrlic *et al.* [16] propose an anonymisation technique that uses a trusted third party. It issues pseudonym certificates to the SMs, which are used to encrypt and sign power readings. This data is relayed by the TTP once it verifies the signature and removes any identifiable information, subsequently forwarding it to the ESP. Therefore, no aggregation is performed, and a TPM is assumed to be present in the household. This is for calculating the bill at the end of the month, while still being able to detect manipulations of the meter through remote attestation. However, the solution has some overheads because of the permanent data delivery between the SM and the TTP.

Rottondi *et al.* [17] propose introducing Privacy-Preserving Nodes (PPNs) between the SMs and the ESP that, according to a central configurator, aggregate data based on space (for a set of SMs spread in an area) and time (for a single SM) depending on the need and access rights. Privacy is preserved with the use of a secret sharing scheme: a secret (i.e., the energy usage information) is divided into shares that are distributed among the nodes, so that the ESP cannot reconstruct the measurements until it collects, at least, a defined number of them. Exploiting the homomorphic properties of this scheme, the data can be aggregated in the PPNs and then delivered to the ESP without revealing individual measurements. This architecture is resilient against faulty or compromised PPNs as long as the number of healthy ones is above a certain threshold.

Li *et al.* [18], to the contrary, suggest an architecture where the smart meters are placed in a tree topology. Each smart meter, beginning with the leaves, encrypts its own individual electricity measurements and passes them to its parent, which aggregates them with the rest of the children using the Paillier homomorphic cryptosystem [19]. A collector is placed as the root node to ultimately aggregate the data for the ESP. Thus, no inner-node can access any individual measurements and the ESP can only obtain the sum of them. The complexity derives from the creation of the tree prior to running the protocol. Its height should be small enough to reduce the hops and its nodes should not have too many children to avoid excessive computation and communication load.

Lastly, Lin *et al.* [20] propose a semi-trusted storage system which securely stores all the data from meters in an area. The Load Monitoring Center (LMC) can only access a sum of meter readings from several SMs in a single time unit. On the other hand, the ESP can only take the sum of readings from a single SM over a time period. As a result, both load monitoring and billing operations are supported. It is important to remark that random noise is introduced in the sum of encrypted readings from a set of SMs. Thus, LMC obtains an approximate aggregation that can be considered accurate with a given probability. A TPM is used to compute remote attestation and generate the pseudorandom numbers needed for the measurements encryption. One drawback that this approach has is the continuous communication that occurs between the ESP or LMC and the SMs in order to regenerate these numbers to decrypt the readings.

In the remainder of this paper, all these solutions are closely studied and compared with each other to decide on how they fit the expected control requirements for such a critical infrastructure as the Smart Grid.

3.2 Discussion: Privacy vs. Control

In Sect. 3.1, the ten solutions considered in this paper have been introduced, and their main features have been described. Regarding their architecture and implemented PET, some aspects can be pointed out.

On the one hand, with respect to the PETs applied, it is noteworthy that some of the protocols often combine more than one privacy-enhancing technology. The clearest example is trusted computation, through establishing a trust in a third party (e.g., Efthymiou *et al.* [15] with the escrow) or embedding the SM

Table 1. Main features of the surveyed privacy techniques

Implemented PET	**Trusted Computation**		**Verifiable Computation**		**Cryptographic Computation**		**Anonymization**		**Perturbation**	**Batteries**
Privacy Technique	[9] Bohli *et al.*	[13] Lemay *et al.*	[10] Molina-Markham *et al.*	[11] Jawurek *et al.*	[17] Rottondi *et al.*	[18] Li *et al.*	[15] Efthymiou *et al.*	[16] Petrlic	[20] Lin *et al.*	[14] Kalogridis *et al.*
Communication model	SM → Aggregator (TTP) → ESP	SM (TPM) → ESP	SM → Aggregator → ESP	SM → PC (Privacy Component) → ESP	SM → PPN (Privacy-preserving node) → ESP	Tree of smart meters with a collector in the root	SM → Escrow → ESP	SM → Anonymiser (TTP) → ESP	SM → Storage System → ESP, LMC	SM (LSM+Power Router) → ESP
Setup and commissioning	Symmetric keys exchange required	Asymmetric keys generation for attestation	Asymmetric keys for commitments signing	Privacy component installation in the household required	Secret sharing scheme parameters initialization	Aggregator tree construction+ distribution of encryption keys	Setup needed to establish respective attributable and anonymous data identities	Initialization of certificates with the TTP required	Key generation at TPM	No setup phase. However, additional resources (Battery, LSM, power router) required
Aggregate function	Arbitrary subsets of meters, individual consumption for each meter	Monthly consumption data aggregation	Arbitrary subsets of meters, individual consumption for each meter	Monthly consumption data aggregation per SM	Customizable space and time aggregation complying with privacy and consumer policies	Arbitrary subsets of meters	Arbitrary subsets of meters, individual SM data items along time	Arbitrary subsets of meters submitting real-time consumption data to the TTP	Subsets of meters in an area, individual consumption data per SM	No aggregation performed, no specific architecture or trust relationship required
Security	Symmetrical encryption to transmit data from SM to ESP	Remote attestation, Hardware-protected storage, encrypted and signed measurements	Homomorphic encryption of data	Homomorphic encryption of data, commitments signing for authentication	Secret sharing scheme	Homorphic encryption to aggregate data, asymmetric encryption for signing	Pseudo-anonymisation of data, assumed encryption for setup phase	Asymmetric encryption of data and signing using pseudonym certificates	Encryption of data in the storage system using pseudorandom numbers	Load signature moderation (no encryption performed)

in a TPM to securely perform the cryptographic operations (like Petrlic *et al.* do to calculate the bill with an anonymisation mechanism). Regardless of the trust assigned to these parties or devices, most of the techniques opt to introduce an element between the SM and the ESP in order to intercept the communication and optionally perform an aggregation (e.g., a privacy component in Jawurek *et al.* [11]). Apart from this approach, there are other solutions that prefer to process all data at source, as specified in Lemay *et al.* [13] through a TPM or by using a battery to mask the real power usage, like the solution of Kalogridis *et al.* [14]. The technique of Li *et al.* [18] still performs an aggregation of multiple SM readings without involving any third party, securely routing the data through a spanning-tree with a homomorphic scheme. With respect to the aggregation of power measurements, it is performed over a time period for a single meter only in solutions that pursue billing operations, as do Jawurek *et al.* [11]. On the other hand, some solutions only aggregate data spatially to comply with monitoring operations, as is the case of Li *et al.* [18]. There are also approaches, such as Rottondi *et al.* [17], that are able to aggregate measurements in space (over a set of SMs) and time (for each SM over a billing period) following the rules of a central configurator.

Aside from analysing how these solutions contribute to privacy when measuring power readings, a study of how they behave in terms of control and automation procedures must be done, as stated in Sect. 2 and reflected in Table 2. All considered features of the privacy techniques (denoted as FPT, defined in Sect. 2) are evaluated. Beginning with their **speed**, a technique can be considered as fast when its underlying cryptographic scheme is not complex and it does not imply taking several computational steps (e.g., aggregating, encrypting, and signing). In this sense, Bohli *et al.* [9], Lin *et al.* [20] and Kalogridis *et al.* [14] are fast due to the use symmetrical encryption, modular additions and a load signature moderator, respectively. Other solutions like Molina-Markham *et al.* [10] are far less efficient because of the Zero-Knowledge protocol that requires high computational capabilities. Other techniques are somewhat competent but require various operations. These are marked with a $\backsim$ in Table 2.

When **storage** is considered, the techniques discussed are positive as long as the smart meter does not hold consumption data or if all this information, used for aggregation, is stored in a third party (e.g., the PC in Jawurek *et al.* [11]). For this reason, the solution of Lemay *et al.* [13], for example, cannot be considered as such, as it saves all measurements on the TPM. As for **communication overhead** it is related to the frequency of data delivery and the number of messages transmitted between the SM, an eventual third party and the ESP. Here, a solution has been treated as efficient if there is only one message and it occurs once per billing period, between the SM and the ESP (so only some techniques for billing meet this requirement, for example Lemay *et al.* [13]). An intermediate level of overhead can be conceived when the SM contacts the ESP once, but it is always transmitting data to a third party to accomplish monitoring operations or that frequency depends on customisable aggregation rules (denoted as $\backsim$). A protocol is inefficient when there is a frequent communication between the SM and the ESP, like Lin *et al.* [20].

Concerning **synchronisation** between all parties involved to relay data, only Rottondi *et al.* [17] require the privacy-preserving nodes to gather measurements from a set of smart meters simultaneously, which can be considered negative from the perspective of performance.

With regards to **sustainability** features, three of the surveyed techniques offer the possibility to extend their functionality which makes their solutions more configurable to fit both customer and utility needs. In the case of Lemay *et al.* [13] their local TPM virtualisation system is able to implement new virtual machines with different purposes. In Rottondi *et al.* [17], configurability is achieved through the central configurator, which can adapt to new aggregation and privacy policies. Both of these solutions are maintainable for the same reason, as is the approach of Petrlic *et al.* [16], which contemplates remote TPM updates to integrate new mechanisms and fix possible errors.

Most of the techniques described **tolerate unlimited failures** when taking measurements with the help of a third party or because of the underlying protocol features. One exception is Lin *et al.* [20], where the LMC needs the meters to reply with their blind factors used for decrypting data. Also, Kalogridis *et al.* [14] has not been considered as fault-tolerant because a failure in the power routing system leaves all the real measurements exposed. A special case is Rottondi *et al.* [17], whose secret sharing scheme makes it possible for the ESP to reconstruct the data as long as it has a minimum number of them spread across all the privacy-preserving nodes. As a result, it is fault-tolerant depending on the number of working nodes. As to the presence of error in the aggregated data, Lin *et al.* [20] perturbation protocol is the only one which introduces noise in the measurements (in particular when aggregating data spatially).

Lastly, it is worth commenting on the **resilience** of these techniques. Some of them include in their papers a brief description of response to certain attacks. For example, Bohli *et al.* [9] explains the detection of fake groups of SMs; Lemay *et al.* [13] suggest remote attestation and tamper-proof components to protect against malicious software and physical attacks; Li *et al.* [18] mention the resistance to dictionary attacks against ciphertexts; Efthymiou *et al.* [15] propose sanctioning nodes when a power theft is detected, by temporarily lifting their anonymity; and Petrlic *et al.* [16] is resistant against false data injections and also includes software integrity attestation, which is also presented in Lin *et al.* [20] due to the use of TPMs. The trust given in this device turns out to be the main problem of this approach: what is executed within the TPM is the responsibility of the ESP, which can always run code to transfer sensitive information. Therefore, some auditing processes have to be performed by third parties in order to check the software and its possible vulnerabilities.

In light of the comparison in Table 2, it is noticeable that TC is the PET that tends to show a better behaviour when performing control in smart metering. In particular, we can highlight Lemay *et al.* [13], which provides more benefits due to the efficient handling of measurements in the TPM. Nonetheless, Petrlic *et al.* [16] also demonstrates suitable capacities for power consumption and control management, and also involves trust in a third party and with the use of a TPM, as stated at the start of this comparison. Alternatively,

Table 2. Control requirements for surveyed privacy protocols

Implemented PET		TC		VC		CC		Anon		Pert	Batt
CR[a]	FTP[b]	[9]	[13]	[10]	[11]	[17]	[18]	[15]	[16]	[20]	[14]
Performance	**Speed**	✓	∽	×	∽	∽	∽	∽	∽	✓	✓
	Storage	✓	×	×	✓	✓	×	✓	✓	✓	✓
	Comm.	∽	✓	∽	✓	∽	∽	∽	∽	×	∽
	Sync.	✓	✓	✓	✓	×	✓	✓	✓	✓	✓
Sustainability	**config.**		✓			✓			✓		
	maint.		✓			✓			✓		
Dependability	**Fault-tol.**	✓	✓	✓	✓	∽	✓	✓	✓	×	×
	Agg. error	✓	✓	✓	✓	✓	✓	✓	✓	×	✓
Survivability	**Resil.**	✓	✓				✓	✓	✓	✓	

[a] Control Requirements
[b] Features of the Privacy Techniques

Bohli *et al.* [9] and Efthymiou *et al.* [15] are similar solutions to the ones commented earlier, which present a lower complexity due to the use of symmetric encryption and pseudoanonymisation, respectively. To sum up, a good approach for designing privacy solutions is the combination of PET solutions of the kind: TC and Anonymisation.

4 Conclusions and Future Work

Despite it being accepted that accurate readings provided by smart meters improve Demand Response control and help customers fulfil their needs, it also raises several privacy issues. New techniques must be implemented to prevent other parties involved in the Smart Grid infrastructure from accessing personal consumption data that leads to the extraction of life patterns. We have conducted a concise analysis to classify some of the most relevant solutions considering different criteria: their implemented PET, their suitability for billing or monitoring purposes and other factors, like the aggregation and architecture type, that affect how privacy is preserved. Moreover, since automation efficiency also has to be considered, we have compared the main control requirements expected for these protocols. Future work will involve defining a more precise taxonomy of the privacy and control features of each of these protocols to systematically find the best solution depending on the needs of customers and grid operators. Also, it would be interesting to implement real prototypes of these solutions to perform a quantitative comparison.

Acknowledgements. The second author receives funding from the *Ramón y Cajal* research programme financed by the Ministerio de Economía y Competitividad. In addition, this work also has been partially supported by the same ministry through the research project PERSIST (TIN2013-41739-R), by the Andalusian government through the project FISSICO (P11-TIC-07223) and by the European Commission through the H2020 project NECS (H2020-MSCA-ITN-2015- 675320).

References

1. Mohassel, R.R., Fung, A., Mohammadi, F., Raahemifar, K.: A survey on advanced metering infrastructure. Int. J. Electr. Power Energy Syst. **63**, 473–484 (2014)
2. Jawurek, M., Kerschbaum, F., Danezis, G.: Sok: Privacy technologies for smart grids-a survey of options. Microsoft Res., Cambridge, UK (2012)
3. Mahmud, R., Vallakati, R., Mukherjee, A., Ranganathan, P., Nejadpak, A.: A survey on smart grid metering infrastructures: threats and solutions. In: 2015 IEEE International Conference on Electro/Information Technology (EIT), pp. 386–391. IEEE (2015)
4. Souri, H., Dhraief, A., Tlili, S., Drira, K., Belghith, A.: Smart metering privacy-preserving techniques in a nutshell. Proc. Comput. Sci. **32**, 1087–1094 (2014)
5. Finster, S., Baumgart, I.: Privacy-aware smart metering: a survey. IEEE Commun. Surv. Tutorials **16**(3), 1732–1745 (2014)
6. Alcaraz, C., Lopez, J.: Analysis of requirements for critical control systems. Int. J. Crit. Infrastruct. Prot. (IJCIP) **5**, 137–145 (2012)
7. Brundtland Commission et al.: Our common future, Towards sustainable development. World Commission on Environment and Development (WCED). Geneva: United Nation (1987) Chap. 2
8. Al-Kuwaiti, M., Kyriakopoulos, N., Hussein, S.: A comparative analysis of network dependability, fault-tolerance, reliability, security, and survivability. IEEE Commun. Surv. Tutorials **11**(2), 106–124 (2009)
9. Bohli, J.M., Sorge, C., Ugus, O.: A privacy model for smart metering. In: 2010 IEEE International Conference on Communications Workshops (ICC), pp. 1–5. IEEE (2010)
10. Molina-Markham, A., Shenoy, P., Fu, K., Cecchet, E., Irwin, D.: Private memoirs of a smart meter. In: Proceedings of the 2nd ACM Workshop On Embedded Sensing Systems For Energy-efficiency in Building, pp. 61–66. ACM (2010)
11. Jawurek, M., Johns, M., Kerschbaum, F.: Plug-in privacy for smart metering billing. In: Fischer-Hübner, S., Hopper, N. (eds.) PETS 2011. LNCS, vol. 6794, pp. 192–210. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22263-4_11
12. Pedersen, T.P.: Non-interactive and information-theoretic secure verifiable secret sharing. In: Feigenbaum, J. (ed.) CRYPTO 1991. LNCS, vol. 576, pp. 129–140. Springer, Heidelberg (1992). https://doi.org/10.1007/3-540-46766-1_9
13. LeMay, M., Gross, G., Gunter, C.A., Garg, S.: Unified architecture for large-scale attested metering. In: 2007 40th Annual Hawaii International Conference on System Sciences, HICSS 2007, pp. 115–115. IEEE (2007)
14. Kalogridis, G., Efthymiou, C., Denic, S.Z., Lewis, T.A., Cepeda, R.: Privacy for smart meters: Towards undetectable appliance load signatures. In: 2010 First IEEE International Conference on Smart Grid Communications (SmartGridComm), pp. 232–237. IEEE (2010)
15. Efthymiou, C., Kalogridis, G.: Smart grid privacy via anonymization of smart metering data. In: 2010 First IEEE International Conference on Smart Grid Communications (SmartGridComm), pp. 238–243. IEEE (2010)
16. Petrlic, R.: A privacy-preserving concept for smart grids. Sicherheit in vernetzten Systemen **18**, B1–B14 (2010)
17. Rottondi, C., Verticale, G., Capone, A.: Privacy-preserving smart metering with multiple data consumers. Comput. Netw. **57**(7), 1699–1713 (2013)

18. 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 (SmartGridComm), pp. 327–332 (2010)
19. Paillier, P.: Public-key cryptosystems based on composite degree residuosity classes. In: Stern, J. (ed.) EUROCRYPT 1999. LNCS, vol. 1592, pp. 223–238. Springer, Heidelberg (1999). https://doi.org/10.1007/3-540-48910-X_16
20. Lin, H.-Y., Tzeng, W.-G., Shen, S.-T., Lin, B.-S.P.: A practical smart metering system supporting privacy preserving billing and load monitoring. In: Bao, F., Samarati, P., Zhou, J. (eds.) ACNS 2012. LNCS, vol. 7341, pp. 544–560. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-31284-7_32

A Six-Step Model for Safety and Security Analysis of Cyber-Physical Systems

Giedre Sabaliauskaite(✉), Sridhar Adepu, and Aditya Mathur

Singapore University of Technology and Design, Singapore 487372, Singapore
{giedre,sridhar_adepu,aditya_mathur}@sutd.edu.sg

Abstract. A Six-Step Model (SSM) is proposed for modeling and analysis of Cyber-Physical System (CPS) safety and security. SSM incorporates six dimensions (hierarchies) of a CPS, namely, functions, structure, failures, safety countermeasures, cyber-attacks, and security countermeasures. The inter-dependencies between these dimensions are defined using a set of relationship matrices. SSM enables comprehensive analysis of CPS safety and security, as it uses system functions and structure as a knowledge-base for understanding what effect the failures, cyber-attacks, and selected safety and security countermeasures might have on the system. A water treatment system is used as an example to illustrate how the proposed model could serve as a useful tool in the safety and security modeling and analysis of critical infrastructures.

Keywords: Cyber-Physical Systems · Failures · Safety
Cyber-attacks · Security · GTST-MLD · 3-Step Model

1 Introduction

Complex system architecture and modeling has been an important subject of research since the early 1960s [12]. Most complex systems are formed as hierarchies, such as functional, structural, behavioural, etc. [8,12]. Early research in this area focused on complex physical systems [8]. However, advances in digital electronics allowed integrating the digital (cyber) systems with the physical world. Such systems are known as Cyber-Physical Systems (CPS).

Modarres and Cheon [8] proposed GTST-MLD - a function-centered approach as a framework for modeling complex physical systems. This framework can be used for system reliability and risk analysis. It comprises of Goal Tree-Success Tree (GTST) and Master Logic Diagram (MLD). GTST is a functional hierarchy of a system organized into different levels. The role of GT is to describe system functions starting with the goal (functional objective) and then defining functions and sub-functions, needed for achieving this goal. ST is aimed at describing the structure (configuration) of the system, used to achieve functions identified in GT. Finally, MLD is used to model the interrelationships between functions (GT) and structure (ST).

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 189–200, 2017.
https://doi.org/10.1007/978-3-319-71368-7_16

Brissaud et al. [2] extended GTST-MLD by integrating faults and failures into it. A new framework was named the 3-Step Model. It allowed modeling the relationships between faults and failures, and the system functions and structure. The analysis of these relationships could be used to assess the effect of any fault or failure on any material element and/or function of the system.

Although the GTST-MLD and the 3-Step Model are powerful tools for physical system safety analysis, they are not sufficient for the vulnerability analysis of a CPS that, in addition to faults and failures, are exposed to cyber-security vulnerabilities and related threats that may compromise system safety [5,6]. In this paper, we extend the 3-Step Model [2] and propose a Six-Step Model (SSM) for integrated CPS safety and security modeling and analysis.

The extended model incorporates cyber-attacks, and safety and security countermeasures, into the 3-Step Model [2]. SSM enables modeling of interrelationships between faults and failures, attacks, safety and security countermeasures, and system functions and structure. System functions and structure are used as a knowledge-base in SSM for the analysis of its safety and security, and the evaluation of the effect of failures and attacks on the system.

In this paper, the Secure Water Treatment (SWaT) [13] system is used as an example to illustrate each step of SSM and to demonstrate the applicability of the proposed model for critical infrastructure safety and security modeling and analysis.

Organization: The remainder of this paper is structured as follows. Preliminaries and background are described in Sect. 2. Section 3 explains SSM. Section 4 contains a summary and conclusions from this work.

2 Preliminaries and Background

2.1 CPS Safety and Security

Safety and security are two key properties of CPSs. They share similar goals, i.e., protecting systems from failures and from attacks [5,9]. Safety is aimed at protecting the systems from accidental failures to avoid hazards, while security focuses on protecting systems from intentional attacks. Safety and security are particularly important in critical infrastructures where hazards include explosions, fires, floods, chemical/biochemical spills and releases, potential crashes of vehicles, etc.

Safety and security are interdependent, often complementing or conflicting, each other [5,6,11]. There are at least four types of inter-dependencies [10]: (1) conditional dependencies: security is a condition for safety and vice versa; (2) reinforcement: safety and security countermeasures can strengthen each other; (3) antagonism: they can weaken each other; and (4) independence: no interaction between safety and security.

In [6], Kriaa et al. presented a survey of existing approaches for design and risk assessment that consider both safety and security for industrial control system. Several approaches have been proposed so far. They are either generic,

which consider both safety and security at a very high level, or model-based, which rely on a formal or semi-formal representation of functional/non-functional aspects of system [6]. However, there is still a need of approaches that aim at assessing the impact of safety on security and vice versa [5], and their impact on the system (its functions and structure).

2.2 GTST-MLD and the 3-Step Model

The Goal Tree Success Tree, GTST, has been initially introduced in 1980s as a functional decomposition framework for modeling complex physical system, and has been used for nuclear power plant risk assessment [4,7]. The main idea behind GTST is that complex systems can be best describe by hierarchical frameworks [8]. GTST is a functional hierarchy of a system organized in levels.

The goal tree (GT) starts with the goal, i.e., functional objective at the top, which is then decomposed into functions and sub-functions, a realization of which assures that the goal can be achieved [7,8] (see Fig. 1(a)). Different types of functions can be distinguished, such as main and supporting functions, to facilitate the analysis of complex systems. Main functions are the functions directly derived from the goal function. Supporting functions contribute to fulfilling main functions. They may provide information, control, or appropriate environment.

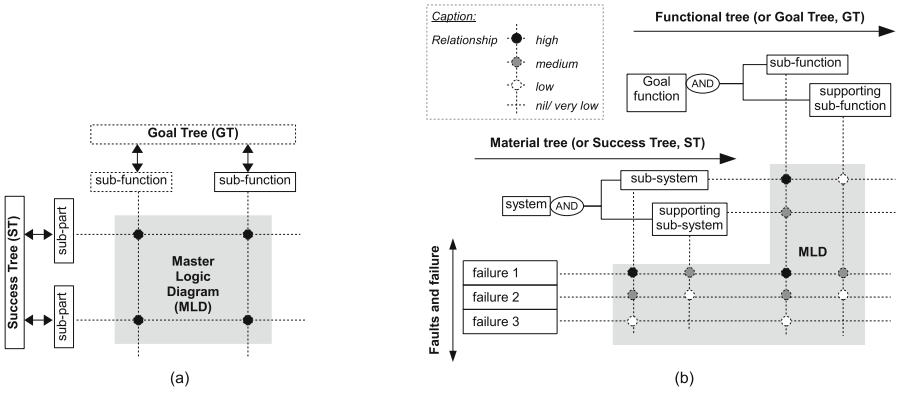


Fig. 1. (a) The GTST-MLD framework [8]; (b) the 3-Step Model [2,3].

The success tree (ST) describes the structure of the system as a collection of sub-systems and objects used to achieve the functions identified in GT. Similar to GT, the primary and supporting elements (objects) are identified in ST. The supporting elements are needed to control the primary elements to provide adequate operating environment.

The hierarchical relationship reveals the connections between different nodes of a hierarchy, or between nodes of two different hierarchies. The relationships

in GTST can be specified by AND and OR gates. Master logic diagrams, MLDs, can be used for modeling these relationships. As shown in Fig. 1(a), MLD is a relationship matrix between GT and ST elements in the form of a lattice.

The combined GTST-MLD framework provides a functional and structural description and modeling method that could be used as a knowledge base for further system analysis [4,8].

Brissaud et al. [2] added faults and failures as the third main part of GTST-MLD framework, and introduced a 3-Step Model as shown in Fig. 1(b). In the first step of a 3-Step Model, functional tree, or GT, breaks a system goal into main and supporting functions. In the second step, material tree, or ST, breaks the system up into subsystems, units, and supporting materials. The third step comprises the list of faults and failures of the material elements [2,3].

MLD is used in the 3-Step Model to exhibit relationships between faults or failures, material elements, and functions. The dot color of the relationship between two elements shows the degree of relationship: black (high), gray (medium), and white (low) (see Fig. 1(b)).

2.3 The SWaT System

The Secure Water Treatment (SWaT) testbed [13] is used in this paper to demonstrate the applicability of SSM in critical infrastructures. SWaT is an operational scaled down water treatment plant. It is designed and built for research in the design of secure cyber physical systems. In a small footprint producing 5-gallons/minute of doubly filtered water, this testbed mimics large modern plants for water treatment such as those found in cities. It consists of the following six main sub-processes, labeled P1 through P6, to purify raw water (see Fig. 2).

P1 Supply and storage. P1 supplies water to the water treatment system.
P2 Pre-treatment. In P2, the water from tank in P1 is pumped via a chemical dosing station to the ultra-filtration feed tank in process P3. In P2 the water quality properties are evaluated and the water pre-treated before it enters P3.
P3 Ultra Filtration (UF), the ultra-filtration process is used to remove water solids by using fine filtration membranes.
P4 De-Chlorination, in this process any free chlorine in water is converted to harmless chlorides through the ultraviolet chlorine destruction unit and by dosing a solution of sodium bisulphite. This step is necessary to avoid damage the the reverse osmosis unit in sub-process P5.
P5 Reverse Osmosis (RO), this process is designed to reduce inorganic impurities by pumping the water with high pressure through semipermeable membranes.
P6 RO permeate transfer, backwash and cleaning. The filtered water from the RO process is stored in raw permeate tank and then transferred to the raw water tank in process P1 for reuse. Sub-process P6 controls the cleaning of membranes in the UF unit in P3 by turning on or off the UF backwash pump. The backwash cycle is initiated automatically once every 30 min and takes less than a minute to complete.

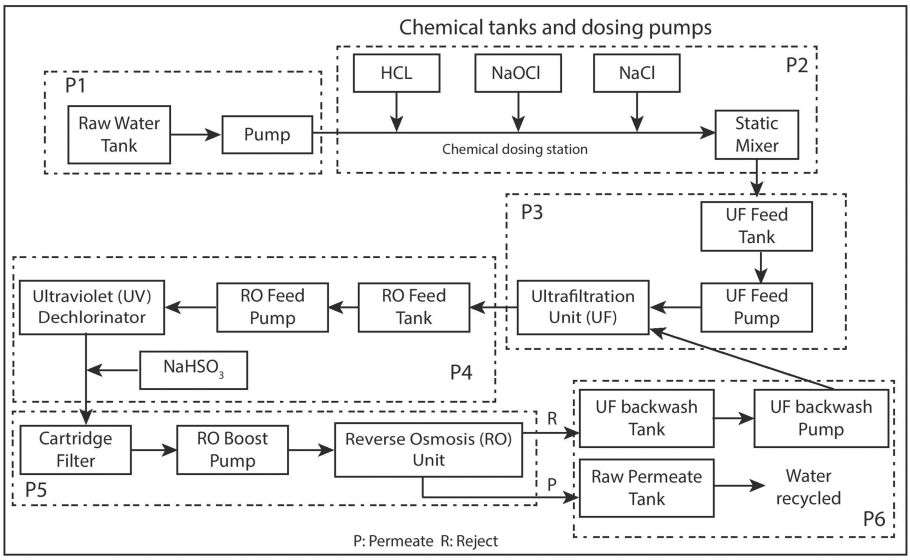


Fig. 2. Water treatment processes P1-P6 in SWaT testbed.

The network architecture of SWaT is organized into three layers labeled L0,L1, and L2. Each of the six processes P1-P6 contains sensors and actuators, and is controlled by a Programmable Logic Controller (PLC). The physical process is manipulated by distributed actuators and measured by sensors. Remote Input/Output (RIO) units associated with each PLC convert the analog signals from sensors into digital form that are sent to PLCs via the L0 network. PLCs communicate with each other through the L1 network, and with centralized Supervisory Control and Data Acquisition (SCADA) system and Human-Machine Interface (HMI), through the L2 network. Communications among sensors, actuators, and PLCs can be using either wired or wireless links; manual switches allow switching between the wired and wireless modes.

3 Complex System Safety and Security Modeling: SSM

Steps in creating a system safety and security model are described next.

Step 1: The first step of SSM includes description of system functions by constructing a goal tree and identifying relationships between the main and supporting functions. This step is analogous to GTST-MLD and step one in the 3-Step Model described in Sect. 2.2. First, GT is formed starting with the goal function at the top that is decomposed into functions, sub-functions, and basic functions. The functions are grouped into main and supporting functions. Second, MLD is constructed to show the relationships between the main and supporting functions (similar to the 3-Step Model in Sect. 2.2). Dotted lines are drawn

from each main and supporting function. The intersection of lines indicated the degree of the relationship: no circle - no relationship; white circle - low relationship; grey circle - medium relationship, and, finally, a black circle indicates that these elements are highly related. As a result, the relationship matrix MF-SF (main functions : supporting functions) is generated.

The functions (F), and the relationships between main and supporting functions of the SWaT system, are shown in Fig. 3. The goal function is the "safe and secure water purification process". Processes P1-P6 are the main functions, while power, control and network are the supporting functions. As processes P1-P6 need power (electricity) for the control and network to function, there is a high level relationship between them. This relationship is indicated by black dots in the MF-SF matrix in Fig. 3.

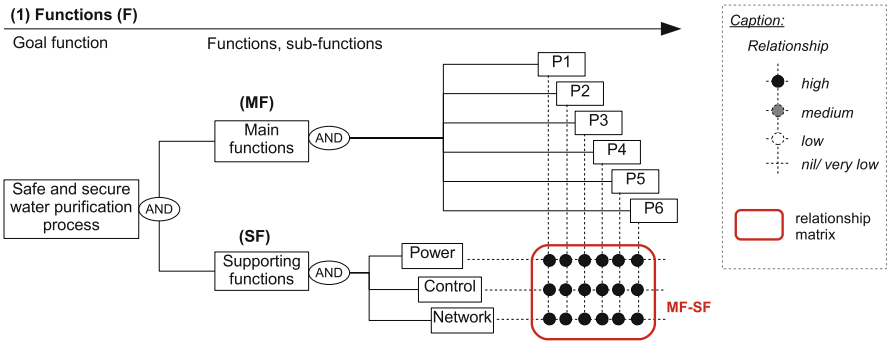


Fig. 3. SWaT system's functions.

Step 2: In the second step, the structure of the system is defined by the use of the success tree (analogous to GTST-MLD and step two in the 3-Step Model as in Sect. 2.2), and the relationships between system structure and functions are identified. At first, system is decomposed into sub-systems and units, which are grouped into main system and supporting systems. Then, MLD is used to model the relationships. The same symbols are used to indicate the degree of the relationship as in Step 1. As the result, relationship matrix S-F (structure:functions) is constructed, as shown in Fig. 4.

Recall from Sect. 2.3 that SWaT consists of six main sub-systems that correspond to processes P1-P6. They are further decomposed into units: sub-system P1 consists of Raw water tank and Pump, while sub-system P5 includes Cartridge filter, RO Booster pump, a multi-stage RO unit, and so on (Fig. 2).

There are three supporting sub-systems in SWaT: control, power supply, and network. Control sub-system can be further decomposed into supervisory and process control sub-systems. Supervisor control comprises of SCADA and HMI, as described in Sect. 2.3. Process control sub-system includes PLCs, RIOs, sensors and actuators. Power supply subsystem consists of three phase power

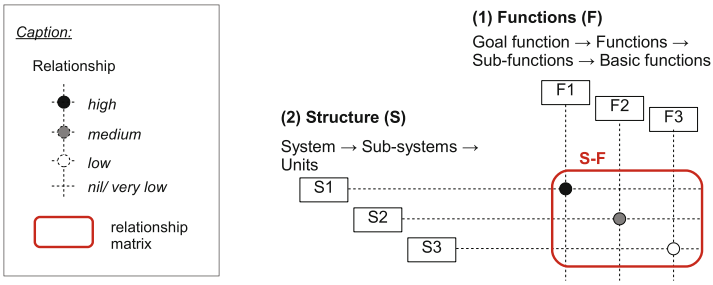


Fig. 4. SSM Step 2: system's structure, and relationships between functions and structure.

supply to distribution board from outside plant, from which power is supplied to each PLC panel, and from there it is supplied to all the devices, sensors and communication switches. Finally, network sub-system is further decomposed into L0-L2 network sub-systems and their components. After definition of all sub-systems and components, their relationships with the system functions are identified by the use of the relationship matrix S-F.

Step 3: In the third step, system failures are identified and added to the model, as shown in shown in Fig. 5. Furthermore, the relationships between failures and system structure and functions are established as in step three of the 3-Step Model (see Sect. 2.2, Fig. 1 (b)). The resulting SSM at the end of step 3 is shown in Fig. 5. It includes two additional relationship matrices: B-S (relationships between failures and structure), and B-F (relationships between failures and functions). The degree of relationships is specified in the same way as in steps 1 and 2.

Failures in SWaT are the main system element failures, such as of pipe, pump, tank, or filter, and the supporting component failures such as of PLC, sensor, actuator, loss of power supply, and network. They are added to SSM, and the relationships between them and the system structure and functions are established. The degree of the relationships between failures and system structure (matrix B-S) indicate which system units could be affected by a failure, while the relationships between failures and functions (matrix B-F) indicate how severely the failures could affect system functions.

Step 4: In this step, safety countermeasures are added to the model. Once the safety countermeasures are identified and added to the model, the relationships between them and the failures, as well as system structure and functions, are modeled, and three additional matrices, shown in Fig. 5, are constructed: X-B (safety countermeasures : failures); X-S (safety countermeasures : structure); and X-F (safety countermeasures : functions).

Additional symbols are used in matrix X-B to specify the failure coverage by safety countermeasures: white rhombus indicates that the countermeasure

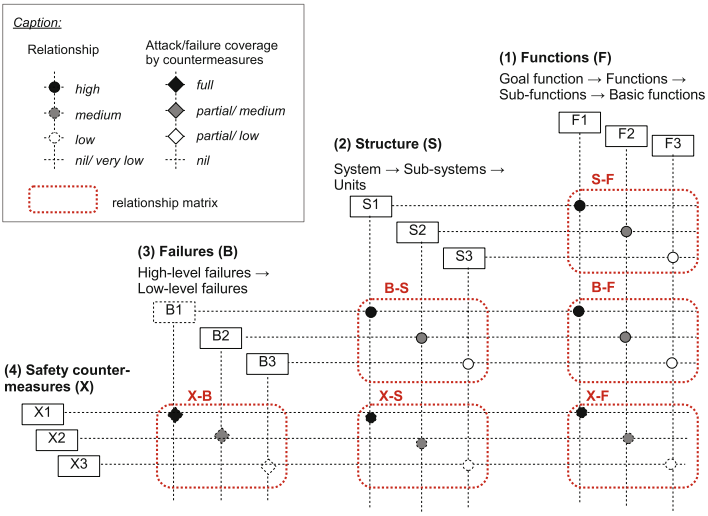


Fig. 5. SSM Steps 3 and 4: failures, safety countermeasures, and their relationships.

provides low protection from the failure; gray rhombus - medium protection; black rhombus - full protection from the failure.

Matrix X-S (safety countermeasures : structure) shows how safety countermeasures affect system structure. If safety countermeasures require the use of additional equipment, e.g., duplicated PLCs or sensors, they are added to the system structure, initially defined in step 2 of the model. Then, their relationships to other elements of the model are established. Matrix X-F (safety countermeasures: functions) shows how system functions are affected by safety countermeasures.

The safety measures implemented in SWaT can be divided as: safety measures to human beings who are working in the SWaT, and safety measures to the system components, hardware devices and complete system itself. In SWaT, there are safety precautions for fire, flooding water, water leakage, electrical shock, injury and chemical exposure.

Hardware related safety countermeasures:

- PLC and pump duplication (one under normal operation and one on standby);
- Overheating switches for UV to prevent heating the UV system;
- IP cameras to monitor the plant;
- Overflow pipe associated with each tank, to prevent water overflow from tanks;
- Pressure switches to monitor UF, RO;
- Overload relay for pumps, UV: this is a circuit breaker that acts as a mechanical protection to the devices;
- Power distribution board has a residual current circuit breaker, which helps to prevent entire system from short circuits.

Software related safety countermeasures (these are encoded in the PLC logic to provide safety to specific components):

- Each pump in SWaT is protected by outlet flow, when the flow is not detected by controller, it sends "OFF" signal immediately;
- To prevent pipe bursts, states of respective motorized valves is verified by ensuring that a valve is open or not before moving a pump into the "ON" state;
- UV dechlorinator is protected with flow meters, as with pumps;
- UF and RO are protected by pressure transmitters such as PIT and DPIT.

In addition, SCADA and HMI alarms are implemented (visual alarms, that are activated under certain conditions, e.g., when water flow reaches a high or a low level). As mentioned earlier, additional equipment, required by safety countermeasures, such as additional PLCs, sensors, and pumps, are added to the "structure" part of the model, and their relationships with the remaining elements of the model are established in this step.

Step 5: In this step, attacks are added to the model, and their relationships with the remaining elements identified, as shown in Fig. 6. Attacks can be either physical or cyber. Currently, only cyber-attacks are considered in the model. In the future, physical attacks could be considered as well. Attack trees can be used in this step for defining possible attacks on the system. Four matrices are constructed in this step: A-X (attacks : safety countermeasures), A-B (attacks : failures), A-S (attacks : structure), and A-F (attacks : functions).

Safety countermeasures, defined in step 4, can be useful for protecting the system from failures and from some of the attacks. Thus, matrix A-X is used to define the coverage of attacks by safety countermeasures. Matrix A-B shows interrelated attacks and failures, while matrices A-S and A-F identify the parts of the system and the functions, which might be affected by each attack.

After completion of step 5, it is necessary to go back to steps 3 and 4 to verify if there are no changes in failures and safety countermeasures due to attacks identified in step 5.

There could be numerous cyber-attacks on the SWaT system, such as attacks on the communication channels between sensors to PLC, PLC to PLC, PLC to actuators, PLC to SCADA, and PLC to HMI; firmware attacks such as attacks on PLC or sensor firmware; attacks on SCADA such as in Stuxnet [14].

After adding potential cyber-attacks on the SWaT system into SSM, we analyze which of these attacks are already covered by safety countermeasures, identified in the previous step. For example, if an attacker's goal is to destroy one of the pumps by over-heating it, hardware overload relay implemented for each pump is helpful in protecting it, thus security counter-measures are not necessary in this case. Over-heating switches, overload relay switches associated with UV, aid in protecting UV from cyber-attacks in most of the cases. The drain system with each tank, and on the floor of the SWaT area, prevents the realization of an attacker's goal of flooding the SWaT area by overflowing tanks and potentially leading to short circuits. Thus, there is no need to implement

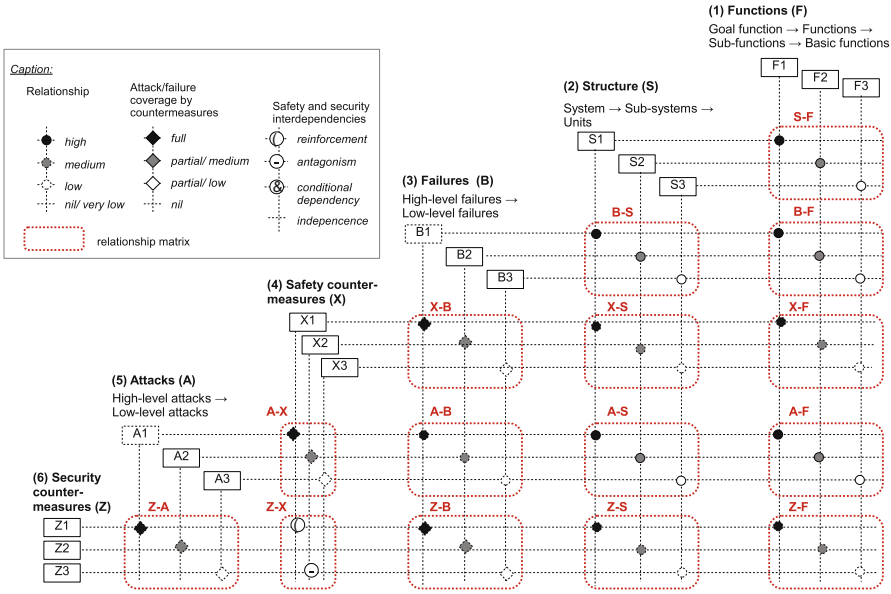


Fig. 6. The Six-Step Model.

additional security countermeasures for the above attacks as they are already covered by safety countermeasures.

Step 6: In this last step of SSM, security countermeasures are added to the model, and the relationships between them and the other elements of the model are identified. The resulting SSM is shown in Fig. 6. The following five relationship matrices should be constructed: Z-A (security countermeasures : attacks), Z-X (security countermeasures : safety countermeasures), Z-B (security countermeasures : failures), Z-S (security countermeasures : structure), and Z-F (security countermeasures : functions).

Matrix Z-X shows the coverage of attacks by security countermeasures, while matrix A-X - the coverage of the same attacks by the safety countermeasures. If an attack is fully covered by existing safety countermeasures (as shown in matrix A-X), then there is no need for a security countermeasure for this attack. Matrix Z-B shows the coverage of failures by security countermeasures. This matrix could be used together with matrix X-B to analyze how well the failures are covered by both safety and security countermeasures. If security countermeasures require the use of additional equipment (e.g., additional sensors), they are added to the structure diagrams, and their relationships with remaining elements of the model are identified.

Matrix Z-X is crucial in the analysis of system safety and security alignment and consistency. This matrix captures the inter-dependencies between safety and security countermeasures: reinforcement, antagonism, conditional dependency, and independence (as defined in Sect. 2.1). Additional symbols are used in SSM

for describing these relationships (a circle with a "+", "−", or "×" symbol), as shown in Fig. 6.

As in step 5, it is necessary to return to steps 3 and 4 after completion of step 6, and to revise the failures and safety countermeasures.

An access control mechanism has been implemented to protect SWaT from unauthorized intrusion. Different intrusion detection and response to attacks mechanisms could be used to protect SWaT from cyber-attacks [1]. For example, if the communication channel between sensors and PLC is attacked and sensor readings are corrupted, they could be replaced by estimated values computed using historical data or a model of the system dynamics. If a PLC is compromised, an additional, standby PLC, which is primarily used in the system as a safety countermeasure, could be used to control the process.

When adding security countermeasures to the SWaT system, it is important to consider their inter-dependencies with safety countermeasures as implemented in step 4. In most cases security complements (reinforces) safety. Safety countermeasures can prevent accidents and abnormal behavior of components, while security countermeasures are helpful, when an attacker is manipulating the signals in communication channel or firmware, etc. For example, if to realize the goal of increasing the water level in a tank while staying unnoticed by the safety alarm system, an attacker could send manipulated sensor measurements to the corresponding PLC. However, if security countermeasures detect such an attack, they will activate a security alarm and will provide PLC with estimated water level to continue safe plant operation.

In some cases safety and security may contradict each other. For example, to monitor plant safely and provide permanent access to the managers to SCADA and HMI, remote access could be allowed, which could raise the possibility of security breaches.

4 Summary and Conclusion

In this paper, SSM for safety and security modeling of cyber-physical systems is proposed. The model incorporates six hierarchies of a CPS: functions, structure, failures, safety countermeasures, cyber-attacks, and security countermeasures. Furthermore, it facilitates the analysis of the inter-relationships and inter-dependencies between these hierarchies by the use of sixteen relationship matrices: MF-SF, S-F, B-S, B-F, X-B, X-S, X-F, A-X, A-B, A-S, A-F, Z-A, Z-X, Z-B, Z-S, Z-F.

These six dimensions of the system and sixteen above-mentioned matrices in SSM, enable a comprehensive analysis of the safety and security of CPS. The analysis makes use of system functions and structure as a knowledge-base for understanding what effect the failures, cyber-attacks, and selected safety and security countermeasure set might have on the systems.

SSM is particularly useful for cyber-physical system analysis, as it enables the analysis of interactions between cyber and physical parts of the system with focus on their safety and security. Through the application of SSM for modeling

of SWaT it is shown how the proposed modeling process could serve as a useful tool in protecting critical infrastructures.

References

1. Adepu, S., Mathur, A.: Distributed detection of single-stage multipoint cyber attacks in a water treatment plant. In: The 11th ACM Asia Conference on Computer and Communications Security, May 2016, in Press
2. Brissaud, F., Barros, A., Bérenguer, C., Charpentier, D.: Reliability study of an intelligent transmitter. In: 15th ISSAT International Conference on Reliability and Quality in Design, pp. 224–233. International Society of Science and Applied Technologies (2009)
3. Brissaud, F., Barros, A., Bérenguer, C., Charpentier, D.: Reliability analysis for new technology-based transmitters. Reliab. Eng. Syst. Saf. **96**(2), 299–313 (2011)
4. Kim, I., Modarres, M.: Application of goal tree-success tree model as the knowledge-base of operator advisory systems. Nucl. Eng. Des. **104**(1), 67–81 (1987)
5. Kornecki, A.J., Subramanian, N., Zalewski, J.: Studying interrelationships of safety and security for software assurance in cyber-physical systems: approach based on Bayesian belief networks. In: 2013 Federated Conference on Computer Science and Information Systems (FedCSIS), pp. 1393–1399. IEEE (2013)
6. Kriaa, S., Pietre-Cambacedes, L., Bouissou, M., Halgand, Y.: A survey of approaches combining safety and security for industrial control systems. Reliab. Eng. Syst. Saf. **139**, 156–178 (2015)
7. Modarres, M., Roush, M., Hunt, R.: Application of goal trees for nuclear power plant hardware protection. In: Proceedings of the Eight International Conference on Structural Mechanics in Reactor Technology, Brussels, Belgium (1985)
8. Modarres, M., Cheon, S.W.: Function-centered modeling of engineering systems using the goal tree-success tree technique and functional primitives. Reliab. Eng. Syst. Saf. **64**(2), 181–200 (1999)
9. Novak, T., Treytl, A.: Functional safety and system security in automation systems-a life cycle model. In: IEEE International Conference on Emerging Technologies and Factory Automation, ETFA 2008, pp. 311–318. IEEE (2008)
10. Pièrre-Cambacédès, L., Bouissou, M.: Modeling safety and security interdependencies with BDMP (boolean logic driven markov processes). In: 2010 IEEE International Conference on Systems Man and Cybernetics (SMC), pp. 2852–2861. IEEE (2010)
11. Pièrre-Cambacédès, L., Bouissou, M.: Cross-fertilization between safety and security engineering. Reliab. Eng. Syst. Saf. **110**, 110–126 (2013)
12. Simon, H.A.: The architecture of complexity. In: Proceedings of the American Philosophical Society, pp. 467–482 (1962)
13. SWaT: Secure Water Treatment Testbed (2015). http://itrust.sutd.edu.sg/research/testbeds/
14. Weinberger, S.: Computer security: is this the start of cyberwarfare? Nature **174**, 142–145 (2011)

Availability Study of the Italian Electricity SCADA System in the Cloud

Stefano Sebastio[1(✉)], Antonio Scala[1,2], and Gregorio D'Agostino[1,3]

[1] LIMS London Institute of Mathematical Sciences, London, UK
stefano.sebastio@alumni.imtlucca.it
[2] ISC-CNR, Sapienza Università di Roma, Rome, Italy
antonio.scala@phys.uniroma1.it
[3] ENEA, CR "Casaccia", Rome, Italy
gregorio.dagostino@enea.it

Abstract. Recently, the allure of the cloud is also affecting the SCADA systems. Cloud-based commercial solutions to manage small private SCADA systems are spreading, while the utilities are still evaluating the cloud adoption. For electric utility companies, reasons for moving in the cloud can be traced to the need of storing, accessing and analyzing a huge amount of records collected from their new smart meters, and managing grid-connected small-scale decentralized energy generation from renewable sources. Moreover, the cloud allows an easy and affordable deploy of monitor dashboards, populated with accurate consumption data, accessible at anytime from anywhere, directly to the customers' smartphones. On the other hand, cloud poses fears on security, privacy and system downtime. In this work, we focus on the latter aspect analyzing the availability of the SCADA system managing the Italian power transmission network for different possible adoptions of the cloud.

Keywords: Network availability · Power grid · Power transmission SCADA system

1 Introduction

Nowadays, all the public utilities (electricity, natural gas, water and sewage) rely on the telecommunication infrastructure for managing their own network. Industrial Control System (ICS) is a general term referring to this control architecture used in industrial sectors, utilities and critical infrastructures. Within the ICS domain the SCADA (Supervisory Control And Data Acquisition) system is in charge for performing infrastructure tuning (over short and long time periods), ensuring service availability and reliability, enhancing the Quality of Services (QoS), metering and monitoring the network for displaying or recording functions. According to the NIST (National Institute of Standards and Technology) [22]: *"the control centers in SCADA systems collect and log information gathered by the field sites, display information to the HMI (Human Machine*

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 201–212, 2017.
https://doi.org/10.1007/978-3-319-71368-7_17

Interface), and may generate actions based upon detected events. The control centers are also responsible for centralized alarming, logging, historical data archiving, trend analysis and reporting".

Examining the power grid, it is possible to identify several ongoing changes mainly due to the growth of smart grids and green policies which are pushing the adoption of small-scale decentralized generation of power from renewable sources (e.g., solar, wind, geothermal, fuel cells and biomass) [9]. The SCADA system managing the power grid is therefore subjected to sweeping changes. In particular, the geographical distribution of the power sources and the smart meters, are posing both opportunities and essential challenges to the electricity operators. Distributed power generation makes harder continuing to adopt only a proprietary telecommunication network among the regional zones, pushing for the adoption of the public telecommunication network with standardized Internet protocols such as TCP/IP in place of the old Modbus/TCP, DNP3 and IEC-104 protocols. On the other hand, smart meters record a huge amount of usage data that needs to be stored, analyzed and accessed frequently [2]. Moreover, utilities' investments in smart grid technologies have blurred the distinction among the once clearly siloed Operational Technologies (OT) and Information Technology (IT) infrastructures. The increasing distribution of communications and sensors networks, pledges to make autonomous the power rerouting in case of faults (as proved during the Metcalf sniper attack in April 2013 [10]).

The cloud technology can thus offer a virtually limitless scalable solutions in terms of storage and computation in addition to cost saving and new attractive applications. A significant cost saving in the cloud could come outsourcing part of the ICT infrastructure where costs for the on-site data centers in terms of hardware purchases, their management and maintenance are erased (moving from a *CAPital EXpenditure - CapEx* to an *OPerating EXpenditure - OpEx* accounting model). Finally, the cloud can furnish to the power utilities the tools for deploying appealing dashboards, populated with accurate consumption data, to the customers' smartphone [8]. This could be more appealing in an ongoing scenario where the utility's customers act as *prosumers* (i.e., acting as both producers and consumers) e.g., installing grid-connected solar panels on a private rooftop.

The downside is that the cloud poses risks and fears in particular in terms of: *security, privacy and system downtime*. SCADA systems are appealing to an attacker for both the data they produce and the devices under their control.

Different techniques to mitigate security and privacy risks, suited for the SCADA domain, have been proposed in literature. Notably, [3] relying on two recent database techniques (i.e., *searchable encryption* and *private information retrieval*) proposes a cloud environment able to mitigate data confidentiality and operational privacy risks, named *Virtual SCADA in the Cloud* (VS-Cloud) therein. At this regard, it is worth considering that despite only few attacks against SCADA networks are reported for confidentiality reasons, many works have shown the threats for the architectures currently adopted [11,14,21,23], in particular for the lack of strong-authentication procedures in the old but widely adopted Modbus/TCP, DNP3 and IEC-104 protocols.

Although the power system is also physically vulnerable to extreme weather conditions (such as floods, windstorms, hurricanes, prolonged droughts that trigger wildfires), focusing the attention on the Critical Information Infrastructure (CII), in a cloud-enabled SCADA environment the *system downtime* can be *due to a failure in the cloud infrastructure or to issues affecting the telecommunication network*. If the failure of a cloud server can be easily mitigated adopting redundant and backup systems (that could be placed in a different geographical location), telecommunication issues are a bigger problem. Finally, it is worth considering that power grid and telecommunication network are, probably, the most critical infrastructures and the two are interdependent. The power grid provides the energy required by the telecommunication devices, such as routers, switches, signal transmitters and cell towers, to operate, while as discussed beforehand the nationwide power grid is managed through a SCADA system that relies on the public telecommunication network. A minor problem can be exacerbated due to this interdependency causing cascading problems to other CIs [4,5,7,13,15,17,18]. This cascading risk is not only possible but real, as witnessed by the failure occurred in Rome in January 2003 [4]. All this motivate the cautious and slow adoption rate of the cloud for the utilities.

In this work we analyze the dependency from the telecommunication network, for different cloud adoptions, of the SCADA system of the Italian power grid, following the recent work [18]. In particular our focus is on the system availability considering both network (adopting the accurate model in [24]) and nodes availability.

Synopsis. The paper is structured as follows. In the remaining of this section the SCADA system managing the electricity grid is initially introduced (Subsect. 1.1), then possible cloud deployments in this context are briefly presented (Subsect. 1.2). Section 2 presents the methodology adopted evaluating the SCADA system availability for different cloud configurations. Numerical results of our availability study are shown and commented in Sect. 3. Finally, Sect. 4 concludes the work with some final remarks.

1.1 The Hierarchical SCADA System

A nationwide SCADA system is usually arranged as a hierarchical network to act for both near real-time and relaxed timing or non critical operations. From bottom to top, the SCADA hierarchy for a power grid operator is composed by: *Remote Units (RUs)*, *Regional Control Centers (RCCs)*, a *National Center (NC)* and an interface with the ENTSO-E (European Network of Transmission System Operators for Electricity). RUs govern small geographical zones (about the size of cities) with real-time interventions and manage possible events of interruptions. Such RUs are supervised and grouped in larger geographical zones by the RCCs. At a national level, the hierarchy is headed by a NC connected to all the RCCs for gathering data and dispatching commands, and interfaced with the ENTSO-E for cooperation purposes across the European electricity transmission operators.

Lack of data, for both telecommunication and electricity networks, has forced us to restrict our study to the sole Italian region.

The SCADA system works properly when all the connections in its hierarchy (hereinafter referred as *connections of interest*) are well-functioning. While adopting the public telecommunication network, the utility operators build a *virtual proprietary network* by signing specific SLAs (Service Level Agreements) with the telecommunication operators demanding for high availability and low latency. Usually, for economic reasons, SLAs consider the minimal number of links required for having the system working.

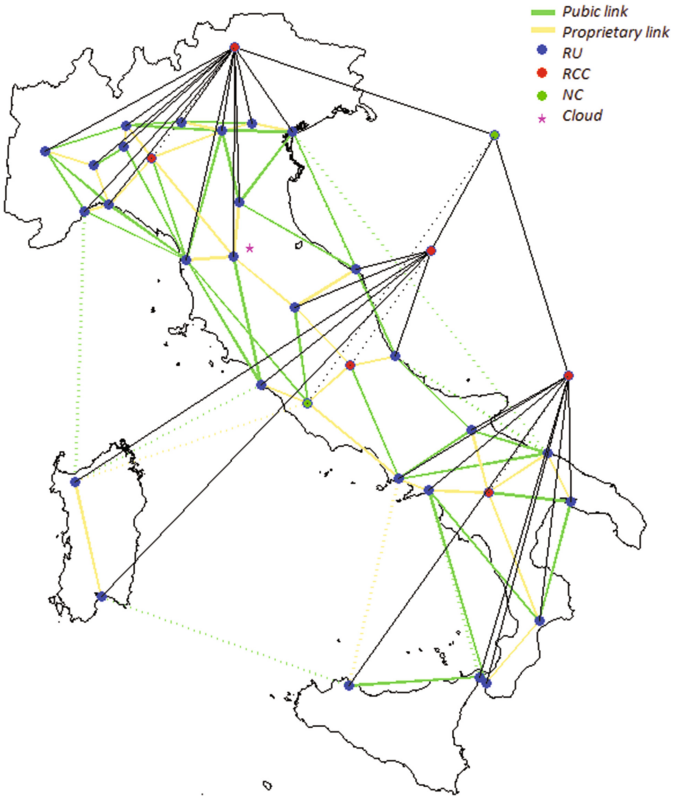


Fig. 1. (A realistic) Italian SCADA network managing the electricity grid (from [18]) (Color figure online)

In [18] (and here reported in Fig. 1) the hierarchical SCADA network managing the Italian electricity grid is shown. In Fig. 1 dashed links represent submarine connections in the optical telecommunication network and links in yellow the *virtual proprietary* network. The hierarchical structure is represented by the black connections while RUs, RCCs and NC are depicted respectively in blue,

red and green. For the analysis considered in this work we followed the assumption in [18] locating the cloud data center in the city of Florence (and shown in figure with a purple star).

1.2 Cloud Deployments for a Nationwide SCADA System

Cloud computing is one of the most recent ICT trends with a large adoption in different domains thanks to its ability of providing a transparent scalability to storage and computing resources even if accessed remotely. A cloud taxonomy can consider the achieved software abstraction or the physical deploy of the machines. Three main kind of abstractions are usually associated with the cloud: *Infrastructure as a Service (IaaS)* in which only the hardware resources are abstracted, *Platform as a Service (PaaS)* in which basic services are provided to the cloud users (such as data-bases or programming tools), and *Software as a Service (SaaS)* in which a license for software running in the cloud is offered to the users. Considering the deployment of the hardware resources, it is possible to discern three main solutions: *private cloud* hosted on-premises, *public cloud* hosted off-premises or by a third party, and *hybrid cloud* interconnecting some resources hosted locally and others remotely. The last approach is emerging as the prominent cloud solution considered by companies closely involved on data managing.

Focusing our attention to the SCADA system, it is possible to identify two deployments in the cloud [6,16,25]: (i) *partially cloud hosted*: in which the cloud works (for storing, remote accessing and analytics purposes) on control and historical data obtained from the SCADA system that works as in the current proprietary configuration; (ii) *fully cloud hosted*: in which the SCADA application directly runs in the cloud that provides command and control to the remotely connected control systems. The first approach seems to be the most reasonable to be applied to a critical infrastructure, since it could provide some of the capability required by the next-generation of the power grid, once sifted through some approaches for mitigating security and privacy risks, while preserving moderate economic benefits.

For a nationwide SCADA system, considering the hierarchical structure of the network, [1,18] propose three possible adoptions of the cloud and here reported for sake of completeness and comparison with the results therein (see Sect. 3). In the `all cloud` configuration the cloud centralizes all the connections governing the SCADA system. The `RCCs to cloud` envisages the cloud working at the same hierarchical level of the RCCs, thus RUs communicate with the cloud, and the latter is interfaced with the RCCs. In the `NC to cloud` the cloud acts at a higher level brokering the communications between the RCCs and the NC. Obviously, each solution has its own set of connections that need to be working for having a proper functioning of the system (called *connections of interest* therein). E.g., in the public network configuration without the cloud: all the RUs need to be connected with the responsible RCC and all the RCCs to the NC. In Fig. 2 the different cloud deployments considered in this work are depicted,

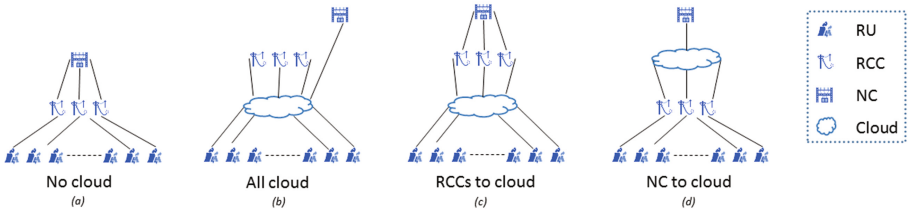


Fig. 2. Cloud deployments for a nationwide SCADA system, from left to right: (a) *no cloud*, (b) `all cloud`, (c) `RCCs to cloud` and (d) `NC to cloud`.

where the *no cloud* case represents both the proprietary and the public network configurations.

Our main reference for this work is [18]. There the Italian SCADA system of the electricity grid is evaluated for a possible adoption of the cloud. The main difference is the type of analysis. While in [18] network reliability and reachability analysis are performed, in the present work we study, for each connection of interest, the availability (adopting the realistic optical link availability model proposed in [24]).

2 Availability Computation in a Hierarchical SCADA Network

In reliability theory, *availability* ($A \in [0, 1]$) is defined as the fraction of time the system is operational [20]. The first steps consist in the identification of all the *functional blocks* and in modeling their availability with parameters obtained from experiments, information from technology vendors/adopters or in the worst case with proper assumptions. In case the system is constituted by a network, the functional blocks can be both nodes and links.

A frequently adopted approximation considers a constant failure rate. In such an approximation, for repairable systems, two main metrics are contemplated: *Mean Time To Failure (MTTF)* and *Mean Time To Repair (MTTR)*. The *Mean Time Between Failures (MTBF)* correlates the two previously defined metrics through: $MTBF = MTTF + MTTR$. From the above introduced metrics the availability can be defined as: $A = \frac{MTTF}{MTTF+MTTR} = \frac{MTTF}{MTBF}$. The *unavailability (U)* is the complement of A, i.e., $U = 1 - A$.

In a network model, each path (p) is composed by multiple blocks, thus it is worth discerning the scenario with a single from the one with multiple block failures [26]. In the first case, since the block availability (A_b) is close to 1, the path availability A_p^s is computed as:

$$A_p^s = min_{b \in p} A_b \tag{1}$$

In the more general case, with multiple simultaneous failures, computing A_p^m, the availability of all its blocks is considered:

$$A_p^m = \prod_{\forall b \in p} A_b \tag{2}$$

Finding the path with the highest availability could easily become computationally demanding even for moderately large networks. The simple trick of applying $-log(A_p^m)$ can be of help, transforming the original problem in a shortest path problem.

However, in network models, more than one *path* is often available to keep a given connection of interest (as defined in Subsect. 1.2) working. It is possible discerning two scenarios: (i) *fully link-disjoint paths* and (ii) *partially link-disjoint paths*. In the latter case at least two, of the k possible paths, share at least one link. In presence of fully-disjoint paths the system availability is computed straightforwardly as:

$$A_{FD} = 1 - \prod_{i=1}^{k} (1 - A_{p_i}) \tag{3}$$

Instead, in the case where the paths overlap on one or more links, the availability computation is more articulated. Intuitively, under the same number of paths and links, in this case the availability is lower because the failures are no more independents. An example, a proof by induction and a way to compute it can be found in [26]. Here, the availability computation for this case is reported:

$$A_{PD} = 1 - \coprod_{i=1}^{k} (1 - A_{p_i}) \tag{4a}$$

$$\coprod \{X, Y\} := \begin{cases} X \cdot Y & iff \quad X \neq Y \\ X & iff \quad X = Y \end{cases} \tag{4b}$$

2.1 Availability Model for an Optical Network

In our work we adopt the availability model for an optical link proposed in [24]:

$$U = U_{tx} + \left[ceil\left(\frac{L}{L_{span}} \right) - 1 \right] \cdot U_{span} + U_{rx} \tag{5}$$

An optical link is indeed composed by a series of components: transmitter, amplifier(s) (one or more according to the link length), and receiver. The distance between two amplifiers differs if the cable is terrestrial or placed undersea. The unavailability parameters presented in [24] are summarized in Table 1 after a precomputation phase.

Table 1. Unavailability parameters for optical links (from [24])

$U_{tx} = 22.98 \cdot 10^{-6}$	$U_{rx} = 22.09 \cdot 10^{-6}$
Submarine	$L_{span} = 57$ Km
	$U_{span} = 9.48 \cdot 10^{-6}$
Terrestrial	$L_{span} = 100$ Km
	$U_{span} = 1.68 \cdot 10^{-5}$

2.2 Availability Computation

Our approach to compute the availability for the SCADA network is presented in Algorithm 1. The adjacency matrix is a square matrix in which elements indicate if the pairs of vertexes (identified by column and row) are directly connected or not. In our work, having undirected connections the matrix is symmetric with zeros for the elements on the main diagonal. The next steps of the algorithm are required to adopt the model for the optical link availability ($availMatrix$) and then transform the availability problem in a shortest paths problem ($availToSP$). For each connection of interest (c), fully link-disjoint paths are found ($findshortestpath$) removing the links already considered by other paths for the same connection. Once all the disjoint paths for a given connection have been found, the $availMatrix$ is restored to its initial state.

Algorithm 1. (Approximate) Availability computation

$adjMatrix \leftarrow$ Build adjacency matrix for the network N
$availMatrix \leftarrow$ Apply model in Eq. 5 to $adjMatrix$
$availToSP \leftarrow -log(availabilityMatrix)$
for all $config \in [proprietary, public, AllCloud, RCCsToCloud, NCtoCloud]$ **do**
  for all connections of interest c in $config$ **do**
    while *path* $p_i \leftarrow findshortestpath(availToSP, c) \neq$ inf **do**
      $A_{p_i} \leftarrow$ as in Eq. 2
      $availToSP \leftarrow availToSP - links \in p_i$
    end while
    restore $availToSP$
    $A_c \leftarrow$ as in Eq. 3
  end for
  $A_{config} \approx$ as in Eq. 4a
end for

It is worth noting that, since the problem is NP-hard [26], to alleviate the computational demand, we adopted an approximated approach, in particular, forcing all the paths involved in a given connection of interest, to be fully disjoints. Moreover, for a given configuration we are overestimating the availability since we approximate A^{PD} considering the possible overlapping only on the nodes availability but not on the links. For larger networks, more sophisticated techniques for computing the network reliability (such as [12,19,27]) are advocated.

3 Availability Assessment in the SCADA System Managing the Italian Electricity Grid

If, for the parameters of the link availability, in [24] we found realistic values that authors have compared with operators' requirements and vendors' specifications, the node availability is another issue. SCADA operators are not willing to disclose such sensitive information and even assumption are not easy to state. In [18] some values for the nodes reliability have been assumed. Despite reliability and availability are different concepts (the first deals with the outage time while the latter with the disconnection probability), for ease of comparison and lacking of any reasonable data on the nodes availability in the electricity SCADA network, we decided to adopt the same values for the node availability (reported in Table 2). Most probably the real availability parameters are significantly higher, especially concerning the SCADA nodes.

In Fig. 3 the histograms with the availability for all the connections of interest in all the considered SCADA configurations are shown. It is possible to observe that `proprietary` and `public` configurations behave almost the same, with great part of the connections having a low availability (remembering the extreme assumptions on the node availability in Table 1). Instead, the `all cloud` and the `NC to cloud` show a similar number of connections with low availability but the `NC to cloud` is able to outperform the `all cloud` having more connection

Table 2. Nodes availability

Remote Unit (RU)	Regional Control Center (RCC)	National Center (NC)	Cloud
0.6	0.7	0.8	0.999

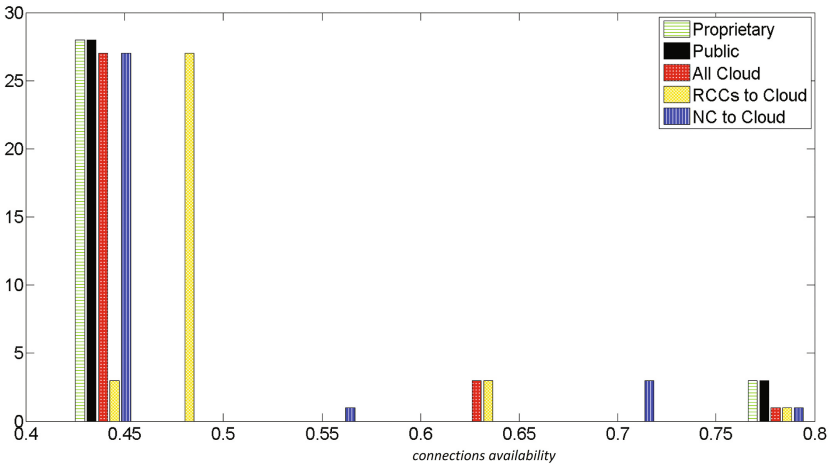


Fig. 3. Histograms with the availability for all the connections of interest in the different SCADA configurations

around 0.7. Finally, the `RCCs to cloud` has great part of its connections (more than 25) around 0.5 instead of 0.45 as for all other configurations.

In Table 3 are reported the availability results, for all the considered configurations, grouped by layer (i.e., at RUs, RCCs and NC). From the results it is possible observing that the `public` configurations mimic the performance of the `proprietary` configuration. At this regard, it is worth highlighting that *the approach currently adopted by the utility operators of building a virtual* `proprietary` *configuration (i.e., selecting a subset of the links of the nationwide telecommunication network for signing SLAs) is a sound solution* by observing the availability in each layer of the SCADA network. Nonetheless, the `NC to cloud` approach is able to slightly improve the performance on the upper layers of the hierarchy. On the other hand, the `all cloud` and `RCCs to cloud` are able to achieve, respectively, four and five nines of availability.

Table 3. Availability Analysis - per layer results

Connection	`Proprietary`	`Public`	`All cloud`	`RCCs to cloud`	`NC to cloud`
$RUs \leftrightarrow RCC_i$	0.699995	0.699988	-	-	0.699988
	0.697129	0.699426	-	-	0.699426
	0.699816	0.699816	-	-	0.699816
$RCCs \leftrightarrow NC$	0.778379	0.778400	-	0.778400	-
$RUs \leftrightarrow Cloud$	-	-	0.998999	0.998999	-
$RCCs \leftrightarrow Cloud$	-	-	0.972027	0.972027	0.972027
$NC \leftrightarrow Cloud$	-	-	0.799200	-	0.799200
Total	0.993955	0.994001	0.999988	0.999993	0.998479

It is worth noting that, systems providing two nines availability (such as `proprietary` and `public`) have about 3.65 days of downtime per year, in case of 99.8% (`NC to cloud`) the downtime per year is reduced to 17.52 h, whereas for a system offering four or five nines availability (see `All cloud` and `RCCs to cloud`) the downtime per year decreases respectively to 52.56 and 5.26 min. Obviously, these observations come from the mere observation of the considered performance parameters and overlook other important aspects which are part of a complex system such as the CII of the power grid.

Comparing our availability results with the robustness and reliability analysis presented in [18], it is possible concluding that all the configurations can be considered sound having robustness (in case of a single link failure) and availability close to one. Among the compared configurations the best candidates for further investigation and analysis, with opportunity for improving the network performance, are the `RCCs to cloud` and the `NC to cloud`.

4 Conclusion

The coming years will show significant changes for the utility operators of the electrical grid thanks to the diffusion of the smart meters and the integration

of grid-connected small-scale decentralized power generation from renewable sources.

In this work, we studied the availability for the SCADA system managing the electricity grid, and evaluated the impact of the cloud adoptions at different layers of the hierarchical structure of a nationwide SCADA network. The study has been restricted to the Italian case since it has been recently inspected, for the network reliability and reachability, in another work [18]. That work has been our data source for what concerns the structure of the Italian SCADA network.

Results of our analysis show that, once the transition to the public telecommunication network has been completed, the cloud could be worth to be further evaluated by the utilities and potentially implemented. Furthermore, other more complex configurations such as the hybrid cloud should be considered.

Concluding, we would like to remark that the lack of real data concerning the nodes availability does not yield to conclusive results. The purpose of this work was to scrape the surface of the problem, turning attentions and fostering discussions towards an ongoing change to one of the most critical infrastructure.

Acknowledgments. Research partially supported by the EU through the HOME/2013/CIPS/AG/4000005013 project CI2C. AS thanks CNR-PNR National Project Crisis-Lab and EU FET project DOLFINS nr 640772 for support. The contents of the paper do not necessarily reflect the position or the policy of funding parties.

References

1. Cloud Computing and Critical Information Infrastructures (CI2C) - network reliability analysis. http://ci2c.eu/reliability.html. Accessed 18 May 2016
2. White paper: Cloud computing for energy and utilities. Technical report, IBM Corporate Headquarters, October 2013
3. Alcaraz, C., Agudo, I., Nunez, D., Lopez, J.: Managing incidents in smart grids à la cloud. In: CloudCom (2011)
4. Bobbio, A., Bonanni, G., Ciancamerla, E., Clemente, R., Iacomini, A., Minichino, M., Scarlatti, A., Terruggia, R., Zendri, E.: Unavailability of critical SCADA communication links interconnecting a power grid and a Telco network. Reliab. Eng. Sys. Saf. **95**(12), 1345–1357 (2010)
5. Buldyrev, S.V., Parshani, R., Paul, G., Stanley, H.E., Havlin, S.: Catastrophic cascade of failures in interdependent networks. Nature **464**, 1025–1028 (2010)
6. Combs, L.: Cloud Computing for SCADA. Technical report, InduSoft (2013)
7. D'Agostino, G., Scala, A. (eds.): Networks of Networks: The Last Frontier of Complexity. Understanding Complex Systems. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-03518-5
8. Electric Light and Power - Bryon Turcotte: Electric utilities consider the cloud. http://www.elp.com/articles/powergrid_international/print/volume-17/issue-5/features/electric-utilities-consider-the-cloud.html. Accessed 18 May 2016
9. Geberslassie, M., Bitzer, B.: Future SCADA systems for decentralized distribution systems. In: UPEC (2010)
10. IEEE Spectrum - Katherine Tweed: Attack on california substation fuels grid security debate. http://spectrum.ieee.org/energywise/energy/the-smarter-grid/attack-on-california-substation-fuels-grid-security-debate. Accessed 28 Aug 2016

11. Kim, T.h.: Securing communication of SCADA components in smart grid environment. Int. J. Syst. Appl. Eng. Dev. **50**(2), 135–142 (2011)
12. Manzi, E., Labbe, M., Latouche, G., Maffioli, F.: Fishman's sampling plan for computing network reliability. IEEE Trans. Reliab. **50**(1), 41–46 (2001)
13. Matsui, Y., Kojima, H., Tsuchiya, T.: Modeling the interaction of power line and SCADA networks. In: HASE (2014)
14. Murray, A.T., Grubesic, T. (eds.): Critical Infrastructure - Reliability and Vulnerability. Advances in Spatial Science. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-68056-7
15. Piggin, R.: Are industrial control systems ready for the cloud? Int. J. Crit. Infrastruct. Prot. **9**, 38–40 (2015)
16. Piggin, R.: Securing SCADA in the cloud: managing the risks to avoid the perfect storm. In: IET ISA Instrumentation Symposium (2014)
17. Scala, A., Sebastio, S., De Sanctis Lucentini, P.G., D'Agostino, G.: A mean field model of coupled cascades in flow networks. In: CRITIS (2015)
18. Sebastio, S., D'Agostino, G., Scala, A.: Adopting the cloud to manage the electricity grid. In: IEEE International Energy Conference (ENERGYCON) (2016)
19. Sebastio, S., Trivedi, K.S., Wang, D., Yin, X.: Fast computation of bounds for two-terminal network reliability. Eur. J. Oper. Res. **238**(3), 810–823 (2014)
20. Shooman, M.L.: Reliability of Computer Systems and Networks: Fault Tolerance. Analysis and Design. Wiley, New York (2002)
21. Spellman, F.R., Bieber, R.M.: Energy Infrastructure Protection and Homeland Security. Government Institutes, Rockville (2010)
22. Stouffer, K.A., Falco, J.A., Scarfone, K.A.: SP 800–82. 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). Technical report, NIST (2011)
23. Ten, C.W., Liu, C.C., Manimaran, G.: Vulnerability assessment of cybersecurity for SCADA systems. IEEE Trans. Power Syst. **23**(4), 1836–1846 (2008)
24. Tornatore, M., Maier, G., Pattavina, A.: Availability design of optical transport networks. IEEE J. Sel. Areas Commun. **23**(8), 1520–1532 (2006)
25. Wilhoit, K.: SCADA in the Cloud: A security Conundrum? Technical report, Trend Micro (2013)
26. Yang, S., Trajanovski, S., Kuipers, F.A.: Availability-based path selection and network vulnerability assessment. Networks **66**(4), 306–319 (2015)
27. Yeh, W.C., Lin, Y.C., Chung, Y., Chih, M.: A particle swarm optimization approach based on monte carlo simulation for solving the complex network reliability problem. IEEE Trans. Reliab. **59**(1), 212–221 (2010)

Railway System Failure Scenario Analysis

William G. Temple(✉), Yuan Li, Bao Anh N. Tran, Yan Liu, and Binbin Chen

Advanced Digital Sciences Center, Illinois at Singapore, 1 Fusionopolis Way, Singapore 138632, Singapore
{william.t,yuan.li,baoanh.t,yan.liu,binbin.chen}@adsc.com.sg

Abstract. Cyber security has emerged as an important issue for urban railway systems (URS) due to the increasing usage of information and communication technologies (ICT). As a safety-critical public infrastructure with complex, interconnected, and often legacy systems, URS pose challenges for stakeholders seeking to understand cyber threats and their impact, and prioritize investments and hardening efforts. However, other critical infrastructure industries such as the energy sector offer best practices, risk assessment methodologies, and tools that may be both useful and transferable to the railway domain. In this work we consider one successful security initiative from the energy sector in North America, the development of common failure scenarios and impact analysis (NESCOR failure scenarios), and assess their applicability and utility in URS. We use a publicly-available software tool that supports failure scenario analysis to assess example failures on railway supervisory control systems and identify directions for further improving railway failure scenario analysis.

Keywords: Railway · Security assessment · Risk assessment
System modelling

1 Introduction

Urban transportation systems are increasingly reliant on information and communication technology (ICT) for more efficient operation with lower cost. However, such systems come with an elevated risk of malware and targeted cyber attacks by malicious agents. Multiple cyber incidents affecting the rail industry have been reported publicly [4]. For example, in 2003 the "So Big" virus caused a morning shutdown of CSX's signalling and dispatch systems in 23 states in the U.S. In another case from 2008, a Polish teenager used a wireless remote controller to change track points, derailing multiple trains and injuring 12 people. More recently, in 2016, a UK-based cybersecurity firm disclosed the discovery of four cyber attacks against UK rail infrastructure within a period of twelve months [13]. As the threat landscape changes, coping with the increasing risks of cyber attacks has become a major concern for transit agencies. For those organizations, systematic risk assessment is essential for exploring system vulnerabilities and supporting the design and deployment of more secure systems.

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 213–225, 2017.
https://doi.org/10.1007/978-3-319-71368-7_18

One approach that has proven useful in other critical infrastructure domains is the practice of failure scenario analysis. A cyber security failure scenario is a realistic event in which the failure to maintain confidentiality, integrity, and/or availability of cyber assets creates a negative operational impact. The most prominent example of this practice is the failure scenario and impact analysis compiled by the U.S. NESCOR (National Electric Sector Cybersecurity Organization Resource) Technical Working Group 1 (TWG 1) in the electric power industry. The NESCOR failure scenarios [21] describe specific types of undesirable cyber incidents and their impacts, as well as the vulnerabilities and potential mitigations associated with the failures. Their work, progressively updated and released from 2012 through 2015, pushes toward more comprehensive and rigorous security assessment in the industry.

However, as far as we know there is no similar systematic failure scenario analysis effort in transit control systems, and the extent to which scenarios from the power domain may be translated to railway is unclear. In this work, we seek to bridge this gap by (1) assessing the structural differences between power and railway infrastructure, (2) identifying the applicability of NESCOR failure scenarios to railway systems, and (3) proposing sample failure scenarios that are then analyzed for a railway supervisory control system. To carry out the railway case study we leverage a publicly-available software tool called CyberSAGE, which has been recently used for electric power grid failure scenario analysis [20].

The paper is organized as follows. In Sect. 2 we introduce the concept of failure scenarios and analyze the translatability of power grid scenarios to the railway domain. In Sect. 3 we provide examples of railway failure scenarios for the supervisory control system. In Sect. 4 we provide a case study by using the CyberSAGE software tool to analyze scenarios on a specific system model. We discuss related work in Sect. 5 and conclude in Sect. 6.

2 Failure Scenario Analysis: From Power Grid to Railway

The cybersecurity of critical infrastructure systems has received significant attention over the last decade. One infrastructure in particular that has emerged at the forefront of this awareness and system hardening effort is the electric power grid. High-profile research projects and academic/industry partnerships have brought new tools and best practices into the power industry and increased system resilience [8,9]. This proved to be much needed: according to ICS-CERT in the U.S., over 40% of cyberattacks targeting industrial control systems in 2014 were focused on the energy sector [19]. Transportation, however, was also significant at 5% and in other parts of the world (e.g., Europe, Asia) public transport plays an even more critical role in citizens' lives than in the U.S.

In the U.S. electric power sector there was an important government/industry initiative to establish a set of electric sector failure scenarios and impact analyses. That reference document allowed utility companies to share a common understanding of typical cyber threats facing the industry, and provided a resource for educating staff, communicating requirements to vendors, and conducting risk

assessment. It is our belief that other industries—particularly the rail transportation industry—can benefit from adopting a similar failure scenario framework that is shared throughout the industry. In this section we describe the key features of NESCOR failure scenarios and provide an example. We then discuss our process and the challenges for translating this concept into the railway domain.

2.1 NESCOR Failure Scenarios for the Energy Sector

Detailed documentation from the NESCOR team may be found online [16]. In this section we summarize salient features and objectives of the effort to provide readers with context for our own work on railway system failure scenarios.

According to NESCOR Technical Working Group 1 [21]: *A cyber security failure scenario is a realistic event in which the failure to maintain confidentiality, integrity, and/or availability of sector cyber assets creates a negative impact on the generation, transmission, and/or delivery of power.* These failure scenarios have been developed for various power grid subsystems, including generation (GEN), advanced metering infrastructure (AMI), and distributed energy resources (DER) among others.

Irrespective of the power grid subsystem, the failure scenarios follow a common format: (1) a text description of the failure; (2) a list of relevant vulnerabilities enabling the failure; (3) a list of impacts; and (4) a list of potential mitigations. Below we provide an excerpt from scenario DER14 [21], which is relevant to renewable generation or energy storage systems, as a reference:

Example Scenario: DER Systems Shut Down by Spoofed SCADA Control Commands

- **Description:** A threat agent spoofs DER SCADA control commands to perform emergency shutdowns of a large number of DER systems simultaneously.
- **Sample Vulnerability:** *Users lack visibility of threat activity*, specifically messages sent to DER systems but not originated by the SCADA system
- **Sample Impact:** Power system instability, including outages and power quality problems
- **Sample Mitigation:** *Authenticate data source* for the DER SCADA protocols

Certainly the impacts in a NESCOR failure scenario are specific to the power grid, but other aspects are more general. While the original NESCOR document contained ad hoc vulnerabilities and mitigations for each scenario, the newer versions have systemized these to establish 82 common vulnerabilities and 22 common mitigations across all subsystems. In the scenario description above these are indicated by italicized text, while non-italicized text contextualizes the vulnerability or mitigation more specifically to the scenario at hand. Table 1 provides additional examples of common vulnerabilities and mitigations.

Table 1. Examples of NESCOR common vulnerabilities and mitigations.

Common vulnerabilities	Common mitigations
System takes action before confirming changes with user	Authenticate users
Users lack visibility that unauthorized changes were made	Check message integrity
Network is connected to untrusted networks	Limit remote modification
Default password is not changed	Test after maintenance

2.2 Toward Railway Transportation Failure Scenarios

Our process for adapting cyber failure scenario analysis for railway applications is to (1) identify critical sub-systems that form the basis for scenario categories; (2) assess which features and information can be leveraged from the electric power grid effort; and (3) develop the failure scenario details. We address steps 1 and 2 in this section, and provide example scenarios in the next section. This preliminary work has been undertaken as part of an ongoing research project [7]. However, to realize a similar level of impact as the power grid NESCOR scenarios, a broader consortium or working group effort is necessary.

To address the first step, it is necessary to differentiate between main line (long distance, typically above ground) and urban (shorter distance, often underground) railway systems. Each present certain unique cyber security challenges. For example, urban railway systems (i.e., metro systems) typically rely more heavily on automation than main line systems due to their more controlled environment. Main line systems place more emphasis on interoperability, leading to standardization efforts like the European Train Control System. As a starting point we focus on urban railway systems in this paper. Through survey and discussions with railway stakeholders, our team assessed the system architecture of automated passenger railway (metro) infrastructure and identified operationally-critical subsystems which may be appropriate for failure scenario analysis. These include the *traction power system*, *signalling system*, *tunnel ventilation system*, *communication systems*, and *trainborne systems*. Particularly in newer rail systems, the various operational subsystems will be integrated and managed through a common supervisory control and data acquisition (SCADA) system (see Sect. 3). This represents a key point of differentiation with respect to the power grid, which often has siloed communication and control systems (e.g., the advanced metering network, the substation control network). Following on this point, the second phase in developing railway failure scenarios is leveraging information from the NESCOR power grid effort.

Translating Scenarios Across Domains. Due to the fundamental differences between power and railway systems (see Fig. 1) it is not straightforward to directly translate failure scenarios between domains. We examined each of the 123 NESCOR failure scenarios (excluding the 'generic' category, which is applicable by definition) to identify those which could translate to the

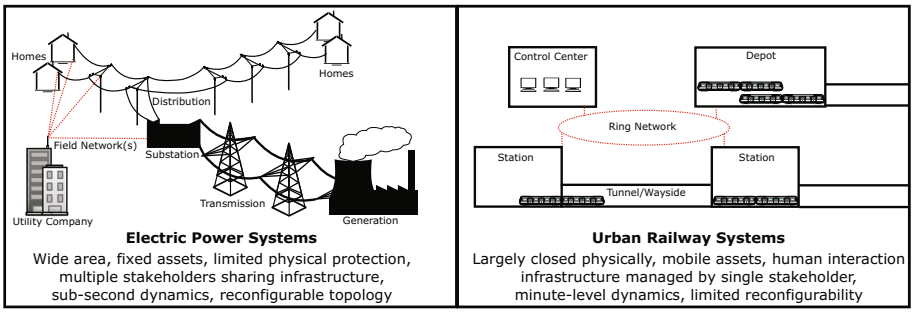


Fig. 1. Contrasting power grid and urban railway systems.

railway domain. Of the original power grid scenarios, 64 (52%) were found to be translatable, either due to the nature of the cyber threat/failure, or the types of systems/devices that were affected. For example, one could imagine a scenario similar to AMI.8–*False Meter Alarms Overwhelm AMI and Mask Real Alarms* unfolding with false alarms in railway supervisory control systems. Similarly, DER.9–*Loss of DER Control Occurs due to Invalid or Missing Messages* could be reinterpreted as *Loss of train control occurs due to invalid or missing messages.*

To identify common failure modes, we categorized those 64 scenarios according to the following list:

- **Message** [34%] spoofing, false data injection, or improper commands
- **Malware** [19%] introduction of compromised or malicious software
- **Configuration** [17%] incorrect or compromised device/system settings
- **Access control** [14%] inadequate physical or logical access control
- **Denial of service** [9%] degradation or disruption of a system service
- **Process** [4%] absent or inadequate business processes

While the relative frequency of those failure classes are most meaningful within their application setting (i.e., power grid), the classification process provides insights into the types of failures that may be of concern in railway systems. The railway supervisory control system in particular, identified as a critical subsystem earlier, may be particularly susceptible to message, malware, or denial of service failure scenarios. In the next section we use the NESCOR common vulnerabilities/mitigations to develop two failure scenarios focusing on these issues.

3 Sample Railway System Failure Scenarios

In the previous section, messages, malware, and denial of service were identified as high-risk failure modes for railway systems. In this section we examine a sample railway control system loosely based on the reference architecture in NIST Special Publication 800-82 [24], which is shown in Fig. 2, and describe two railway failure scenarios that could impact that system. In Sect. 4 we introduce

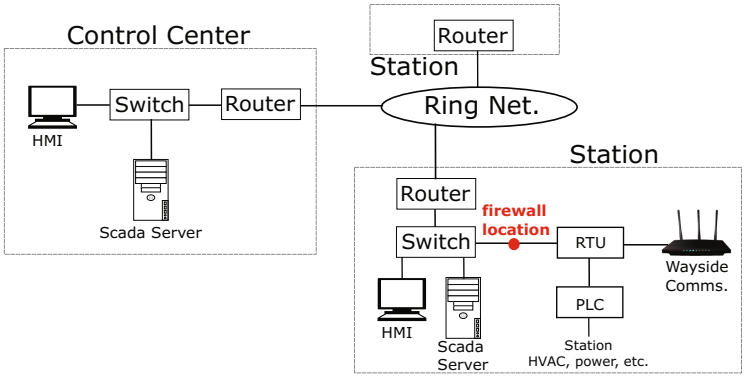


Fig. 2. Sample railway control system architecture emphasizing station devices.

and apply a tool to analyze these scenarios. As we discuss in the case study (Sect. 4.2), these scenarios illustrate trade-offs and conflicting objectives for system operators seeking to secure their infrastructure.

3.1 Compromised HMI Sends Malicious Commands to Devices

A human-machine interface is infected with malware, either through a USB flash drive, or through the network. This malware can send unauthorized commands to devices in the station or at the trackside to disrupt railway operations.

Vulnerabilities

- *system permits installation of malware* in the HMI
- *system permits potentially harmful command sequences* enabling the compromised device to affect operations

Impact

- Settings are changed, degrading system performance
- Devices are remotely shut down, affecting train service

Mitigation

- *authenticate users* for all software changes
- *test after maintenance* for malware
- *create audit logs* of all changes to software
- *protect audit logs* from deletion
- *validate inputs* with the user for all control actions
- *authenticate messages* communicated in the SCADA network.

3.2 SCADA Firewall Fails and Critical Traffic Cannot Reach Devices

SCADA firewalls are installed to only allow authorized computers to send commands to control devices. In some cases, these firewalls are susceptible to Denial-of-Service attacks (DoS) that are exploitable by low-skill attackers [12]. Once under DoS, no packet can go through the firewall, which disrupts critical real-time communication with devices.

Vulnerabilities

- *commands or other messages may be inserted on the network by unauthorized individuals*
- *unnecessary access is permitted to system functions*

Impact

- Operators from the control center lose sight of the status of devices
- Operators from the control center are unable to send commands to devices

Mitigation

- *restrict network access* to firewall administrative functions
- *require intrusion detection and prevention* as part of the SCADA network.

4 Analyzing Scenarios for a Railway System

While there are ancillary benefits to developing a common set of railway system failure scenarios, the main objective is enhancing cyber risk assessment. To this end, tool support is important to help operators quickly and easily assess their risk exposure and make decisions about how best to harden their systems. In this section we adapt the CyberSAGE tool [2], which is freely available for academic users, to conduct a case study using the railway supervisory control failure scenarios from Sect. 3.

4.1 Failure Scenario Analysis Tool

The CyberSAGE tool was originally developed for analyzing security properties in cyber-physical systems [25]. It was subsequently extended to support analysis of NESCOR failure scenarios over a user-defined system model in the power and energy sector [20]. We use that version of the tool as a starting point for our work analyzing railway system failure scenarios.

At a high level, the process for analyzing a failure scenario is as follows:

1. Represent the scenario description as a mal-activity diagram [23]
2. Draw the system architecture and specify properties
3. Specify adversary profiles describing skills, resources, access, and intention

4. Modify graph generation rules, if necessary
5. Generate the assessment results

Note that steps 1–3 need not be followed in any particular order. Since identifying assets and interfaces is typically the first step in cyber risk assessment [22], we created the system architecture model first. As an output, CyberSAGE calculates the probability of a failure scenario occurring based on user-defined properties. An analyst may then multiply this probability with an impact value (e.g., financial loss) for the failure scenario to compute a risk score.

Certain modifications were necessary to adapt CyberSAGE for railway systems. The software has a GUI for drawing a system using devices and edges. There are a number of default devices, and each device has a list of properties. For the most part, the existing devices were sufficient to model railway SCADA systems, but some new devices (e.g., programmable logic controller) had to be created during step 2 of the process. This involves naming the device and specifying an image to use as the icon. Each device has a set of properties associated with it. Rather than creating new properties, we re-used the property set from other devices. During the assessment, the user can deactivate or otherwise modify a specific device's properties, so having a large default set was not restrictive.

Finally, during step 4 it was necessary to create new rules for combining the various inputs. These are specified in a drools format [3]. To model scenario 3.2 we created a *reboot system template* to capture the vulnerabilities and mitigations associated with the firewall.

4.2 Case Study: Deploying SCADA Firewalls

We observe from the failure scenarios in Sect. 3 that adding SCADA firewall devices to station control networks can help to mitigate certain failures (e.g., malicious commands), while potentially introducing new failure scenarios. In this case study we assume the role of a railway operator seeking to holistically evaluate the benefits of adding SCADA firewalls to the system shown in Fig. 2. This example is intended to illustrate the role that systematic failure scenario analysis can play in the rail industry.

Model Inputs. We model failure scenario 3.1 for the system with and without the firewall between the station switch and RTU. Figure 3 shows the first input: a mal-activity diagram depicting how the scenario unfolds when the firewall is present. The "Firewall" swimlane and the activity steps within it are removed when there is no SCADA firewall in place. Figure 4 shows the mal-activity diagram for the second failure scenario.

The next step in the assessment process is modelling the system and devices. Figure 5 shows the system architecture after it has been modelled in the tool. By clicking on each device the user can specify which cyber security controls (i.e., mitigations in failure scenario terminology) are present, and the degree to which they are effective (taking values within 0–1). The default value is 0.5.

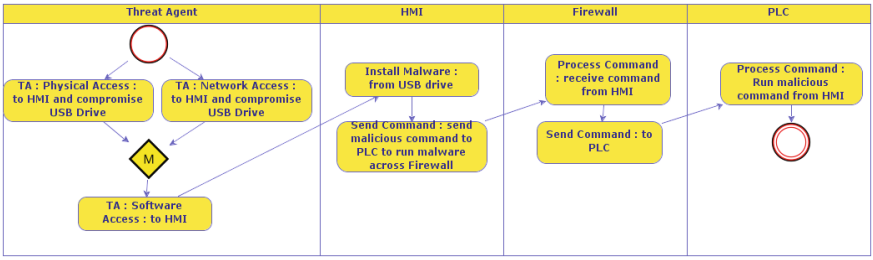


Fig. 3. Mal-activity input for scenario 3.1 with SCADA firewall.

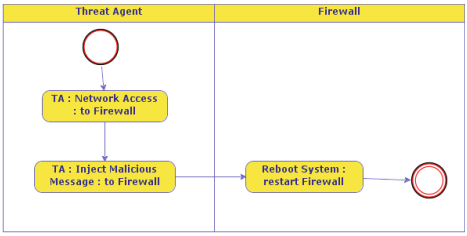


Fig. 4. Mal-activity input for scenario 3.2.

An essential step in any cybersecurity risk assessment is understanding the potential threat actors. For critical infrastructure systems, *insiders* and *nation states* or advanced persistent threat actors are likely to be the top threats. A third threat actor that may be overlooked is 3rd party contractors or technicians who access certain systems (e.g., HVAC) for maintenance. Table 2 summarizes the adversary properties.

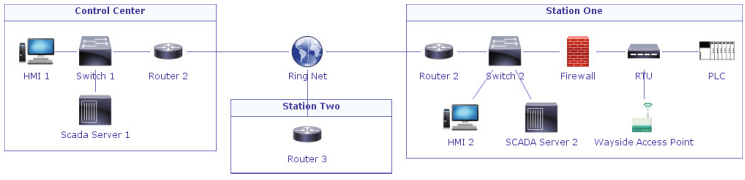
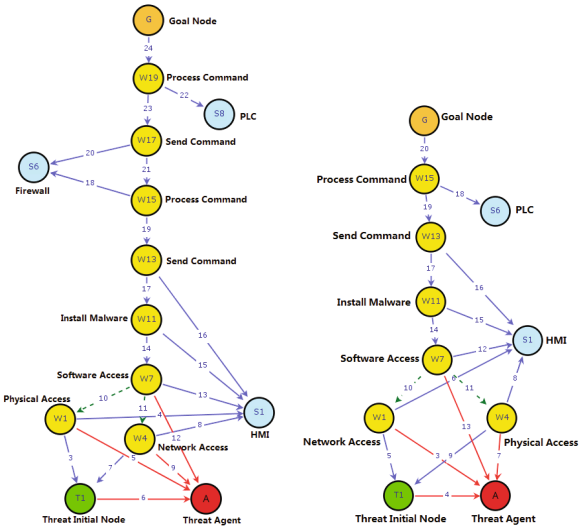


Fig. 5. Railway SCADA topology created in the tool.

Assessment Outputs. We run the evaluation for scenario 3.1 under two system configurations: with and without the SCADA firewall between the station switch and RTU. Figure 6 shows the resulting graph for both assessments. Although it is not easy to interpret the meaning of nodes and edges without the tool's GUI, the graph uses information about vulnerabilities and mitigations for the modelled devices to produce a system-level probability that represents the specified attacker's chance of causing the failure.

Table 2. Adversary profiles

Attacker	IT skill	Domain knowledge	Physical access	Logical access
Insider	Medium	High	True	True
Contractor	Low	Low	True	False
Nation State	High	High	False	True



(a) With firewall (b) Without firewall

Fig. 6. Annotated graphs generated for failure scenario 3.1

Table 3 lists the assessment results, i.e., the probability of the specified adversary reaching the failure state, for each scenario. For scenario 3.1, *compromised HMI sends malicious commands to devices* adding a SCADA firewall has a significant impact on the system security level, reducing the success probability from 16–35% depending on the threat model. As expected, the Nation State adversary poses the largest threat, followed by insiders.

However, depending on the device, SCADA firewalls can introduce potentially serious vulnerabilities into the system [12]. In some cases, a denial of service attack on a firewall may be worse, operationally speaking, than an incorrect or malicious command to a device. The results from scenario 3.2 indicate that the threat actors that are of utmost concern for a rail operator (Nation State, Insider) have a significantly higher chance of causing an operational issue by exploiting a vulnerable firewall than they do of compromising an HMI and sending malicious commands (scenario 3.1). This is due to the small number of attack steps for scenario 3.2—there are few opportunities for mitigation. In this case, the operator may be better off hardening the HMI than introducing a SCADA firewall device. Ultimately, the onus is on system operators to carefully

Table 3. Attacker success probability for failure scenarios 3.1 and 3.2

Attacker	Scenario 3.1			Scenario 3.2
	Before firewall	After firewall	Improvement	After firewall
Insider	0.0119993	0.00953225	21%	0.387352
Contractor	0.00499653	0.00322911	35%	0.0177942
Nation State	0.015291	0.0128755	16%	0.573079

evaluate their security strategy and ensure that systems are implemented in a manner that achieves their goals. By systematically conceiving, documenting, and analyzing failure scenarios, operators can have greater confidence that their system will perform.

4.3 Discussion

The example failure scenarios and the case study presented above touch on only a small portion of the railway infrastructure. The large scope and interconnected nature of railways, spanning communications, power, control devices, and mechanical systems necessitates thorough and repeated risk assessment as the cyber threat landscape changes. Our team, as part of an ongoing project, has developed failure scenarios for certain railway subsystems, but we recognize the need for broader involvement from the rail industry and other research organizations to make this type of analysis more systematic and impactful in the industry.

While there is much to learn from the electric power grid failure scenario effort, there are several challenges that emerged. First and foremost, the original NESCOR scenarios are slanted toward physical impact, which is domain and often subsystem-specific. A consequence of this is that similar cyber failures may appear for different subsystems, but those failures may have very different impact. It is our belief that more work should be done to assess the impact of cyber failures in railway, particularly as they vary with time, commuter traffic, and geographic location. This is an area of future work for our project [7].

5 Related Work

In recent years there have been several research projects focusing on security (physical or cyber) and safety for the rail industry [1,5,6]. Unfortunately the results and outcomes of many of those projects are not publicly available. One of those projects released a white paper [10] recommending, among other things, interoperability of risk assessment methodologies and creating a knowledge repository. Similarly, the American Public Transportation Association (APTA) recently published a series of recommended practice documents [11] for securing control and communication in transit environments. Those documents examine common features of transit systems, classify security zones, and suggest cyber

risk assessment practices such as attack tree modeling. Our work shares the intent of the above efforts to elevate cyber security awareness within the industry, and suggests one possible approach for sharing knowledge and risk assessment practices that has been implemented in another industry.

From a research standpoint, there is a rich literature focusing on risk assessment frameworks, tools, and processes [22]. Techniques such as failure mode and effects analysis [18], hierarchical holographic modeling [17], HAZOP studies [26], and the CORAS method [15] are intended to support decision makers in the identification of potential risks from system failure or malicious activity. Ultimately, practitioners will decide which method, or combination of methods, best meets their needs. Perhaps the most similar work to our own analyzes the European Rail Traffic Management System (ERTMS) [14]. The authors refer to two confidential technical reports analyzing attack scenarios on this system. Their scenarios included some elements found in NESCOR-style failure scenarios, such as vulnerabilities exploited, potential mitigations, and impact. Our work advocates a broader adoption of this practice, and extends tool support for failure scenario analysis to the railway industry.

6 Conclusion

In this work, we demonstrate how the practice of failure scenario analysis can be readily adapted from the power grid sector and applied to railway infrastructure. Using a case study focusing on railway SCADA systems, we model two failure scenarios with a software tool and provide metrics that can help guide railway operators as they harden their systems. We then discuss opportunities for further enhancing and adapting failure scenario analysis to suit the railway domain.

Acknowledgments. This work was supported in part by the National Research Foundation (NRF), Prime Minister's Office, Singapore, under its National Cybersecurity R&D Programme (Award No. NRF2014NCR-NCR001-31) and administered by the National Cybersecurity R&D Directorate. It was also supported in part by the research grant for the Human-Centered Cyber-physical Systems Programme at the Advanced Digital Sciences Center from Singapore's Agency for Science, Technology and Research (A*STAR).

References

1. ARGUS. www.secret-project.eu/IMG/pdf/20150128-02-uic-argus.pdf
2. CyberSAGE portal. https://www.illinois.adsc.com.sg/cybersage/index.html
3. Drools business rule management system. www.drools.org/
4. Repository of industrial security incidents. www.risidata.com/Database
5. Secured urban transportation project. www.secur-ed.eu/
6. Security of railways against electromagnetic attacks. www.secret-project.eu/
7. SecUTS: A cyber-phyiscal approach to securing urban transportation systems. www.secuts.net
8. Smart grid protection against cyber attacks. https://project-sparks.eu/

9. Trustworthy cyber infrastructure for the power grid. https://tcipg.org/
10. SECRET project white paper, November 2015. www.secret-project.eu/IMG/pdf/white_paper_security_of_railway-against_em_attacks.pdf
11. APTA security for transit systems standards program, July 2016. http://www.apta.com/resources/standards/security/Pages/default.aspx
12. Moxa EDR-G903 vulnerabilities, May 2016. https://ics-cert.us-cert.gov/advisories/ICSA-16-042-01
13. UK rail cyber attacks, July 2016. http://www.telegraph.co.uk/technology/2016/07/12/uk-rail-network-hit-by-multiple-cyber-attacks-last-year/
14. Bloomfield, R., Bloomfield, R., Gashi, I., Stroud, R.: How secure is ERTMS? In: Proceedings of SAFECOMP (2012)
15. den Braber, F., Hogganvik, I., Lund, M., Stølen, K., Vraalsen, F.: Model-based security analysis in seven stepsa guided tour to the CORAS method. BT Technol. J. **25**(1), 101–117 (2007)
16. Electric Power Research Institute: Smart Grid Resource Center - NESCOR. http://smartgrid.epri.com/NESCOR.aspx
17. Haimes, Y.Y., Kaplan, S., Lambert, J.H.: Risk filtering, ranking, and management framework using hierarchical holographic modeling. Risk Anal. **22**(2), 383–397 (2002)
18. IEC 60812: Analysis techniques for system reliability - procedure for failure mode and effects analysis (FMEA) (2006)
19. Industrial Control Systems Cyber Emergency Response Team: ICS-CERT year in review. https://ics-cert.us-cert.gov/Year-Review-2014
20. Jauhar, S., Chen, B., Temple, W.G., Dong, X., Kalbarczyk, Z., Sanders, W.H., Nicol, D.M.: Model-based cybersecurity assessment with NESCOR smart grid failure scenarios. In: Proceedings of IEEE PRDC (2015)
21. National Electric Sector Cybersecurity Organization Resource (NESCOR) Technical Working Group (TWG) 1. Electric Sector Failure Scenarios and Impact Analyses, Version 3.0 (2015)
22. Refsdal, A., Solhaug, B., Stølen, K.: Cyber-Risk Management, pp. 33–47. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23570-7
23. Sindre, G.: Mal-activity diagrams for capturing attacks on business processes. In: Sawyer, P., Paech, B., Heymans, P. (eds.) REFSQ 2007. LNCS, vol. 4542, pp. 355–366. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-73031-6_27
24. Stouffer, K., Falco, J., Scarfone, K.: Guide to industrial control systems (ICS) security. NIST special publication 800–82 (2011)
25. Vu, A.H., Tippenhauer, N.O., Chen, B., Nicol, D.M., Kalbarczyk, Z.: CyberSAGE: a tool for automatic security assessment of cyber-physical systems. In: Norman, G., Sanders, W. (eds.) QEST 2014. LNCS, vol. 8657, pp. 384–387. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10696-0_29
26. Winther, R., Johnsen, O.-A., Gran, B.A.: Security assessments of safety critical systems using HAZOPs. In: Voges, U. (ed.) SAFECOMP 2001. LNCS, vol. 2187, pp. 14–24. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-45416-0_2

Tamper Resistant Secure Digital Silo for Log Storage in Critical Infrastructures

Khan Ferdous Wahid[1(✉)], Helmut Kaufmann[1], and Kevin Jones[2]

[1] Airbus Group Innovations, Munich, Germany
khan-ferdous.wahid@airbus.com
[2] Airbus Group Innovations, Newport, UK

Abstract. Tamper resistant secure data storage is necessary to store event logs in critical environments, as it enables trustworthy evidence collection during an incident, and subsequently allows investigators to analyze attack behaviour, impact of the incident, source of attacks, risk factors, and if necessary it should also offer the procurement of admissible proof at trial. Recent advancements in hardware based security allows us to build such storage mechanisms, and cope with the advanced threats. In this paper, we describe the existing problems in secure storage of logs, and generate requirements to address those problems. Finally, we present our solution with commercial off-the-shelf (COTS) hardware based security technologies, which assures that the system is practical and suitable for integration with current systems. In order to show the feasibility of design, we also implement the solution using open source platforms.

Keywords: Secure logging · Secure storage · Anti-tamper storage
Secure log architecture

1 Introduction

In recent years security of Critical Infrastructures (CIs) has become one of the primary concerns for Governments and Industries worldwide, because of the evolution of cyber threats, leaving the underlying systems increasingly vulnerable to cyber attack. Like other developed and industrialized countries (e.g. USA, Canada, Australia etc.), the European Commission also has been investing in huge research efforts to protect the CIs within the member states. As part of these efforts, several European research groups [1–6] have researched different kinds of pan-European collaborative networks or platforms to share information for better protection of their CIs. But such large interconnected and interdependent systems can only work efficiently, securely and reliably when all associated cyber elements share their information accurately and operate properly. Hence, unexpected behaviors need to be detected and reasons behind the disruption (e.g. attacks or failures etc.) need to be checked for the safeguarding of CIs. In order to investigate any disruptive situation, a tamper resistant Secure Digital Silo (SDS) is required to store all logs of the relevant systems, so that the

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 226–238, 2017.
https://doi.org/10.1007/978-3-319-71368-7_19

stored contents can always present forensic values at all conditions. Therefore, memory protection during cryptographic operations along with secure storage of logs is necessary to eliminate the risk of tampering with data at rest. Although there are a large number of research works available for secure data storage in Cloud, they are not yet ready for the CIs due to various security challenges [7]. Moreover, prevalent secure logging mechanisms, described in detail in Sect. 2, are mostly concerned about forward secrecy rather than protecting memory during sensitive operations. So, in this paper, we present a practical tamper resistant secure log storage mechanism (a.k.a., SDS) using COTS hardware based security technologies, that can utilize necessary computing power of the host machine to avoid any compromise in performance.

However, critical infrastructures generally consist of legacy, proprietary or resource constrained devices, so it is not possible to modify or extend their functionalities in most cases. Hence, our solution of log storage in the SDS starts from the log reception by the log server, and ends at the encrypted log storing in tamper-resistant silo.

If it is possible to modify or extend the log generator, we recommend to extend it with our solution to build a tamper resistant communication channel. Such setup is described in Sect. 4 for clarification.

1.1 Contribution

The contribution of this paper is a practical tamper-resistant SDS for log storage. We mainly focus on practical system rather than theoretical solution, that enables Industries or Governments to store their valuable system logs without security and integrity concerns. Our design mainly covers the standard reception of log messages from different components of critical infrastructure, and also covers secure encryption and signature of the received logs inside hardware protected container to forward them to the tamper-resistant silo.

Although the log server and tamper-resistant silo communicate over a mutually generated secure channel in our solution, they store logs locally with different hardware protected secrets. Hence, this solution bases on the separation of secret knowledge, which protects the logs at rest even if one system is compromised by highly-advanced attacks. Moreover, it is not possible to tamper any stored logs even by the attacker with root privilege in the host machine, because the hardware shielded container is totally protected from privileged software or OS. To achieve such practical solution we use recent COTS hardware based security technologies: (i) Intel Software Guard Extensions (SGX), (ii) Trusted Platform Module (TPM) along with open source security packages, (iii) Enterprise Cryptographic Filesystem (eCryptfs), and (iv) Secure Block Device.

1.2 Organization

We describe our motivation behind the solution in Sect. 2, and depict the technologies used in the solution in Sect. 3. Section 4 sketches the design in detail

and provides the architecture. Moreover, Sect. 5 explains the complete implementation, as well as the analysis of the solution. Finally we draw conclusions in Sect. 6.

2 Motivation

Software based secure logging mechanisms is a well-studied topic [8–13]. However, the prime concern in those solutions is the disclosure of the keys to attackers or the untrusted logger, because secret keys can be captured from memory during cryptographic operations or the logger can be compromised if sufficient protection is unavailable. Originally, such issue of key disclosure is very common in every security solution where the user or operating system (OS) has access to the secret key, and memory protection is not available, because an adversary can access the system without generating any event (when the access is valid according to the policy or bug of the system) to tamper the logging mechanism. Hence, hiding his presence for polluting future logs is an easy task. Even decrypting the encrypted key to perform cryptographic operations allows the key to remain in memory for limited time, which facilitates the attack surface to destroy the security protection. If we can isolate the memory for secret key storage and all related sensitive calculations by hardware mechanism, then every key related security challenge addressed in those solutions can be achieved by default. In similar way, it is not possible to jeopardize the logger when it operates within hardware protected containers.

Hardware supported secure logging solutions are also abundant [14–19]. However, Chong et al. [14] uses resource constrained Java iButton as a tamper resistant hardware token. But the random session keys or timestamp wrapper keys generate inside iButton are shared with client or server, which allows the key to remain in memory for vital operations. Hence, it does not bring better protection than software based secure logging mechanisms. Also the authors conclude that the system is not practical for frequent logging. Wouters et al. [15] and Pulls et al. [16] offer custom hardware design for secure logging, so it is not possible to build a practical solution using COTS hardware.

Besides, Accorsi [17] presents a secure log architecture where he uses an evolving cryptographic key created and stored in TPM [20] to generate the symmetric key for each log encryption. The operation of evolving key itself and the generation of symmetric key from that evolving key for each log entry, in total, produce quite a load on resource constrained TPM, and highly degrades the performance of the solution to make it incompatible with large applications. Sinha et al. [18] uses TPM functionalities to guarantee forward integrity of logs and also addresses unexpected power failures. Nevertheless, the initial key is shared with verifier, so the security can be at stake if the verifier is compromised. Andersson et al. [19] uses the TPM signing mechanism to integrity protect log entries, but such continuous asymmetric operation using resource constrained security hardware is not practical to cope with large number of inputs within short time frame.

Many of the above software solutions [9–12] and Java iButton based hardware solution [14] do not address truncation attack and delayed detection attack [8]. Truncation attack is caused by attacker deleting tail log entries that represents the break-in information, and delayed detection attack is caused by attacker modifying log entries transmitted to verifier where the trusted remote server has no means to get updated information immediately from the logging machine to vouch for those messages.

However, Dijk et al. [21] uses the monotonic counter of TPM version 1.2 to time-stamp data, but most of the manufacturers throttle counter increments because the TPM specification recommends 7 years time-frame for counter increments in every 5 s without causing a hardware failure [22]. So a solution based on such constrained component is not good for Industrial Control System (ICS), because equipments in ICS generally operates far more than 7 years.

Furthermore, Flicker [23] provides trusted execution environment on an untrusted OS using late launch technology available in AMD Secure Virtual Machine (SVM) and Intel Trusted eXecution Technology (TXT) extensions, but it suspends the OS and all functionalities (e.g., other processor cores, keyboard, mouse and network inputs) during trusted code execution. For this reason, it is not possible to use such solution in log server where hundreds or thousands of log messages can be arrived continuously within that suspended time range. Also malicious kernel modules can launch Iago attack during memory allocation by protected programs [24], which can be perfectly thwarted by hardware aided validation as supported by EACCEPT instruction in Intel SGX [25].

After reviewing the previous works about secure logging or storage, we can clearly formulate that we need COTS hardware protected and processor supported highly efficient secure containers which provide tamper-proof secure memory for sensitive cryptographic calculations.

3 Background

This section provide an overview of the security technologies used in the solution.

3.1 Intel Software Guard Extensions (SGX)

SGX allows an application to execute sensitive calculations or generate secrets in hardware-protected container (a.k.a., enclave) supported by 17 new instructions, new processor structure and new mode of execution [25]. The code, data and memory resided in an enclave are totally separated from the OS and other applications. The most intriguing fact about SGX over late launch technology is that it does not suspend the OS or any functionalities. Moreover, it supports multi-core operations inside the enclave.

Enclave software can use persistent sealing keys to encrypt and integrity-protect data [26], and the sealing keys themselves can be sealed to the Enclave Identity or Sealing Identity. When the sealing key is sealed to an Enclave Identity, it is only available to that enclave. No future version of the enclave can access

this key. On the contrary, when the sealing key is sealed to a Sealing Identity, it is available to all future versions of enclave signed by the Sealing Authority. Hence, it allows offline data migration. Intel SGX also provides quoting enclave, which can vouch for an enclave resided in the same system. The details of SGX and attestation mechanisms are however out of scope for this document but are documented in [25–27].

3.2 Trusted Platform Module (TPM)

TPM, standardized by Trusted Computing Group (TCG), is a tamper-proof crypto-processor available in hundreds of millions of devices all over the world. It contains encryption, decryption and signature engine. It also contains random number, RSA keys and SHA hash generator including access protected non-volatile storage. In addition, the TPM offers clock and monotonic counters.

There are several platform configuration registers (PCRs) available in TPM along with a set of instructions acts as the Core Root of Trust for Measurement (CRTM), which is difficult to manipulate. They support integrity measurement using chain of trust to facilitate remote attestation. During power-up, the CPU, assisted by the TPM, passes the control to the CRTM, and then the CRTM measures BIOS code and stores the hash in PCR. Finally the CRTM passes the control to BIOS. Likewise, BIOS measures firmware and boot loader, and records the measurement (or hash) in PCR. Then the control is passed to the boot loader and so on. All these measurements are stored in PCRs. The storage in PCR is actually an extend operation, which combines new digest value with the current value in PCR. Linux Integrity Measurement Architecture [28], developed by IBM and integrated in mainstream Linux, uses this feature to record all changes of file system in PCRs to help validating the integrity of the system remotely.

3.3 Enterprise Cryptographic Filesystem (eCryptfs)

eCryptfs is a stacked cryptographic filesystem integrated in mainstream Linux to encrypt data at rest. It provides per-file encryption and decryption as the data are written to or read from the lower filesystem [29]. Nonetheless, dm-crypt, part of device mapper infrastructure of Linux kernel, is a disk encryption subsystem which provides block level encryption to create an entire encrypted and pre-allocated block device with manual provision of secret passphrase during system-boot. As our secure digital silo needs to store logs continuously for several years, it is not possible to pre-allocate fixed storage for data in rest. Unlike dm-crypt, eCryptfs processes keys and records cryptographic metadata on per-file basis. Thus, the migration of encrypted data and change of the secret keys are quite feasible to achieve, and pre-allocated fixed storage capacity for data in rest is not necessary. Moreover, incremental backup utilities do not need to take any supplementary measures for functional backup of the encrypted files [29].

3.4 Secure Block Device (SBD)

SBD wraps the storage system like a block device without providing any file system [30], but stores data in a file with block read and write interface. The most flexible part of SBD is its simple security requirements of a cryptographic key and a root hash storage. If one can guarantee the perfect security of these two values, then cryptographic confidentiality and integrity of data at rest can easily be achieved with tamper-proof property. It uses advance encryption standard offset codebook mode (AES OCB) for authenticated encryption of the data supported by message authentication code (MAC), and Merkle Hash Tree for data integrity. Each block of data is secured individually with random access capability.

The SBD library is very small, so it offers minimal trusted computing base when integrated with Intel SGX solution. Moreover, SGX can easily secure its cryptographic key and root hash inside secure enclave. Hence, the combination of SGX and SBD greatly simplifies the design of tamper-resistant silo.

4 Design

The main four requirements of reliable storage for logs are **integrity**, **unforgeability**, **availability** and **authenticity**, but to offer a secure storage facility, the design also should consider the following factors:

- **Remote verification**: one should be able to remotely check the integrity and the authenticity of the system.
- **Fail-safe (closed)**: all logs need to be encrypted, so that the logs never disclose infrastructure information in the event of compromise and should not allow an attacker to locate any specific log.
- **Accountability**: discontinuity of logging mechanism needs to be detected
- **Safety**: timely backup to another secured device
- **Correctness**: all events must be logged with the state of logging mechanism
- **Seamlessness**: no disruption of the system over power cycles

If it is possible to support all above features, the SDS can securely and efficiently store or retrieve logs when necessary. In our solution both log server and tamper-resistant silo are supported by Intel SGX and TPM platform. Hence, both systems can perform cryptographic operations using secure enclaves, and at the same time can confirm system integrity and authenticity using TPM attestation techniques. Our usage of TPM and SGX together prevents any discontinuity in the security protection of the system. The integrity measurement architecture (IMA) supported by TPM allows the system to measure and store its overall operational steps from CRTM upto and including the loaded OS, but it cannot guarantee that the actual measured code is running or not after the measurement has taken place [31]. So our inclusion of SGX secured enclave helps to keep the measurement sound for *remote verification*, because the measurement value stored in PCR validates the correct initiation of host system including the SGX enclave, and hardware isolation of SGX guarantees the secure execution

of our code. Moreover, the trusted system initiation permits us to access the eCryptfs protected storage where we keep the data at rest. Also due to such protected operation time-of-check time-of-use (TOCTOU) class of attacks [31] are not practical in our solution.

In typical setup, the log receiver module of log server is not a part of the secure enclave, but it relays the received log immediately to the secure enclave, and also captures its *own state*. Our solution inside the secure enclave stores all logs locally using SBD where the key and root hash are only known to the secure enclave (isolated by hardware). It counts the logs using sequence number which is incremented by 1 for each received log. It also forwards the log to the tamper-resistant silo, which eventually does the same operation for secure storage. Therefore, two copies of the logs are stored using our system for redundancy and *safety*. The storage capacity in the log server can be fixed to store only logs for several days. When there is no incident within that time frame, the old logs can be overwritten with new logs. The log storage capacity inside the tamper-resistant silo should be unlimited (or based on company policy), because it is the main storage for future reference. There is no good reason to include log receiver in secure enclave when the log generator (or client) is running without memory protection, because it is rather easy and safe to attack legacy or simple ICS devices (or log generators) than our log receiver which securely logs its own state.

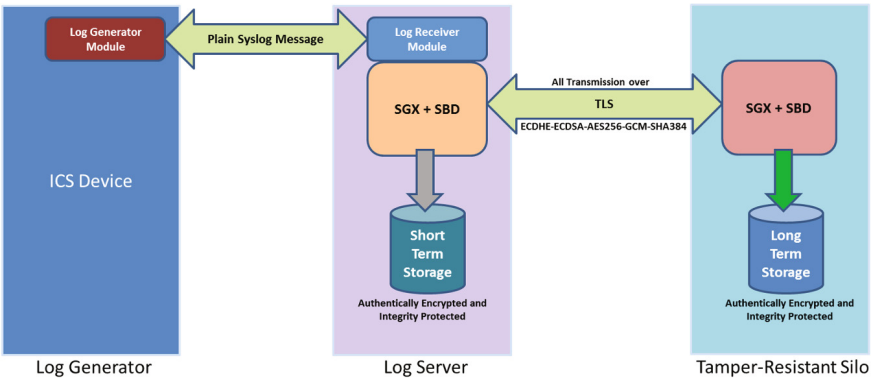


Fig. 1. Log generator only supports plain syslog

However, our log server always enforces maximum security level for communication that is also supported by the client. For example, if the ICS device only supports plain *syslog* message, then the log server receives the logs using standard syslog server as depicted in Fig. 1, but when the ICS device integrates Intel SGX functionality, our log server enforces custom transport layer security (TLS) protection for communication where all cryptographic calculations and operations are only performed inside the secure enclaves as portrayed in Fig. 2.

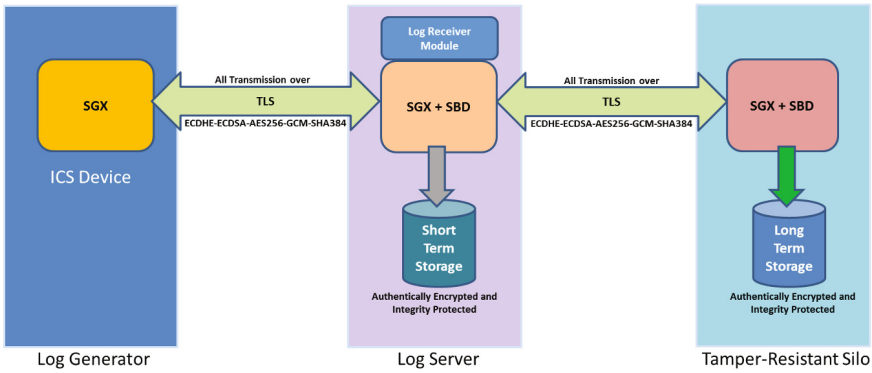


Fig. 2. Log generator supports Intel SGX

Although the communication security with the log generator is based on the maximum security level supported by that client, the communication channel between the log server and tamper-resistant silo is protected using high level security sufficient for many years to come. In our case the default ciphersuite is *ECDHE-ECDSA-AES256-GCM-SHA384* (Figs. 1 and 2), because it has no known issue regarding security vulnerabilities and it is supported by all open source SSL/TLS libraries, but organizations can select their preferred algorithm for such communication.

When the log server or tamper-resistant silo boots in trusted state for the first time, it initiates the secure enclave and generates a local AES key to use with SBD for *authenticated encryption* and *integrity protection* of local log storage. This local AES key is then sealed to the sealing identity of the platform, so the enclave authority can get the key using newer enclaves in controlled environment if the real enclave is somehow destroyed or malfunctioned, and finally can recover all logs encrypted using that key. The SBD allows us to securely store or retrieve data outside the secure enclave. Otherwise, data retrieval from outside the enclave can facilitate an attack surface. After sealing the AES key, the newly initiated secure enclave requests the enclave authority for certificates. The enclave authority verifies the enclave using a quoting enclave, and upon successful verification it delivers certificates. The log server and tamper-resistant silo then initiates TLS protocol to mutually generate a secret key for communication. This key is refreshed as well as the local AES key is unsealed in every boot, so storing these events in the log storage allows to detect system restarts or power cycles or *discontinuity of the logging mechanism*. However, restarting the tamper-resistant silo does not pose serious risk, because log server locally stores all logs in our solution. So all missing logs can be fetched from the log server when tamper-resistant silo is in operation again, but restarting the log server fails to capture some logs due to unavailability. Only a backup log server can help to get logs during that time, so we leave the deployment choice to the users. In addition, both log server and tamper-resistant silo send heartbeat messages

to check the *availability* of the other end in every 5 min to consider an alarm for the administrator. This value can be modified based on the deployment scenario.

Our solution also generates hash of the received log by concatenating the log with the hash of the previous log. So the final hash always contains measurement of all received logs. The final hash is stored using SBD in a different location than log storage, but our solution periodically stores the recent hash as a log inside the log storage with the corresponding sequence number. Hence, it is not possible for the attacker to cut the tail of log file to hide his presence. We store the last hash periodically just to keep the information with captured logs for future references. The SBD already provides authenticity tag for each log, so it is not necessary to extend the size of stored data by adding final hash with each log. Furthermore, storing the final hash in separate location allows us to fetch the sequence number and validates the storage integrity during *power cycles*, and also helps us to count in the correct order and avoid *unforgeability* problems. For this reason, in case of log server reboot, we might fail to capture some logs but the continuation of the log storage operation will not be hampered at all.

To offer better security and avoid network related problems, we recommend to place the log server and tamper-resistant silo in different networks. Less critical systems can incorporate both functionalities in a single machine. Also periodical backup of the tamper-resistant silo is required to avoid hazardous situations.

5 Implementation and Evaluation

Rsyslog [32] was chosen as the log server because it is well-accepted and widely deployed in industries, plus provides high performance in log processing. It is not mandatory in our secure digital silo to place the rsyslog server inside hardware isolated enclave, because security is only as robust as its weakest link, and almost all ICS devices are proprietary, so extending their functionalities with hardware assisted execution environment are not possible. Thus, TLS session keys can be hijacked from the memory of ICS client. Anyway, our system supports plaintext rsyslog, rsyslog over TLS and custom TLS protected communication channel using secure enclave. The log sever always selects maximum security level (among these three) for communication based on the available capability in log generator (a.k.a., ICS client). For top-grade security, the ICS client should be customized to perform cryptographic TLS operations inside hardware isolated container, and the critical part of rsyslog server needs to be placed inside SGX enclave.

Due to unavailability of Intel SGX hardware, we develop the experimental demonstration using open source SGX platform named OpenSGX [24]. It provides memory protected container with no support for TLS communication. Hence, we extend the OpenSGX with PolarSSL [33] TLS functionalities and SBD for building our tamper-resistant silo. Both log server and tamper-resistant silo are implemented in the single machine. So the logs are collected on-the-fly from rsyslog module and then stored using our custom SBD functionalities of OpenSGX enclave. To control and validate memory access from non-enclave region, OpenSGX provides *stub* and *trampoline* functionalities [24], where *stub*

is used to set the input parameters and to get the result of the enclave requests, and *trampoline* is used to pass the control to OS or host program by exiting enclave mode to process enclave requests. We use *stub* and *trampoline* features to store and retrieve data outside the enclave.

The SDS can be queried for specific logs using three criteria- ID, time and sequence number. When the SDS stores logs, it checks the time, counts the sequence number and captures the ID of the client, and adds all these information to the original received log. Therefore, authorized client or forensic tool can query using those criteria to fetch logs in clear text. We use a LONG_MAX value for the sequence number inside secure enclave, because the sequence number should not be expired very soon. The LONG_MAX value of 922337203685477580 7L in 64-bit CPUs, can provide the service for long time before being set to 0 again.

System security is guaranteed by the use of TPM measured and memory protected container called enclave. The secure digital silo uses eCryptfs with TPM to open the door to user session as well as host storage when the system is in trusted state. Otherwise, the user session remains in encrypted format. Such security can protect the log contents when an intruder takes the hard drive to attach it in another system to gain access. As the system changed, the access is not possible. However, sophisticated attackers do not need to physically remove the component, they could access with root privilege while the system is running. Hence, eCryptfs like security mechanism cannot prevent them from tampering the logs. When the system is in trusted state, all files in the eCryptfs file system are accessible for standard operations. Also such attackers can monitor the memory operations to get the secret key used in cryptographic operations. This is where the secure enclave comes to the rescue, malicious root users cannot decrypt or modify the contents encrypted by enclave using SBD, even though they have complete access to the eCryptfs protected storage, because the key never leaves the protected memory of enclave in clear. So our double layer of security thwart attackers from normal to the root privileged level, and guarantee complete tamper-resistant secure storage.

Furthermore, evaluating the performance of our solution cannot reflect the real operation and performance, because we use OpenSGX platform instead of Intel SGX. There is no multi-threading, asynchronous I/O or multi-core support as well as no automatic encryption and integrity protection while storing outside the enclave memory. On the contrary, Intel SGX supports all these features.

In our solution, availability is attained by mutual query between the log server and secure silo. However, addressing Denial of Service (DoS) attack is completely out of scope of this paper, because it is not possible to rule out physical damage to the communication cable or storage medium etc. Hence, non-availability caused by DoS is also out of scope of our research work. Also due to such attack possibilities, we cannot call our solution *tamper-proof secure digital silo*, instead we call it *tamper-resistant secure digital silo.*

6 Conclusions and Future Work

Our solution shows that a tamper-resistant secure digital storage is perfectly achievable using COTS hardware security technologies. The only drawback is the presence of legacy ICS devices in Industrial sectors that are proprietary and unmodifiable. The set-up of tamper-resistant communication between log generator and log server is only achievable when the log generator can be replaced by COTS hardware. It will not only prevent information leakage during transmission, but also guarantees complete protection from the source (i.e., log generation point) to the destination (i.e., tamper-resistant silo).

Our future work targets optimization of our solution and integration of multi-threading in OpenSGX, so that the performance can reach to industry grade. At the same time we plan to implement our solution using real Intel SGX hardware to benchmark both solutions side-by-side.

Acknowledgements. This work is funded by the European FP7 security project ECOSSIAN (607577).

References

1. EU FP7-Security: PERSEUS - Protection of European seas and borders through the intelligent use of surveillance, Project reference: 261748 (2011–2015)
2. EU Horizon 2020: EU CIRCLE - A pan-European framework for strengthening Critical Infrastructure resilience to climate change, Project reference: 653824 (2015–2018)
3. EU EPCIP: NEISAS - National and European Information Sharing and Alerting System, Project reference: JLS/2008/CIPS/016 (2008–2011)
4. EU FP5-EESD: EFFS - An european flood forecasting system, Project reference: EVG1-CT-1999-00011 (2000–2003)
5. EU FP7-Security: ECOSSIAN - European COntrol System Security Incident Analysis Network, Project reference: 607577 (2014–2017)
6. EU FP7-SEC-2010-1: BRIDGE - Bridging resources and agencies in large-scale emergency management, Project reference: 261817 (2011–2015)
7. Younis, Y.A., Merabti, M., Kifayat, K.: Secure cloud computing for critical. infrastructure: a survey. In: 14th Annual PostGraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting (2013)
8. Ma, D., Tsudik, G.: A new approach to secure logging. Cryptology ePrint Archive: Report 2008/185 (2008)
9. Bellare, M., Yee, B.: Forward integrity for secure audit logs. Technical report, Computer Science and Engineering Department, University of San Diego (1997)
10. Schneier, B., Kelsey, J.: Cryptographic support for secure logs on untrusted machines. In: Proceedings of the 7th USENIX Security Symposium (1998)
11. Holt, J.E.: Logcrypt: forward security and public verification for secure audit logs. In: Proceedings of the 2006 Australasian Workshops on Grid Computing and e-Research, Australia (2006)
12. Waters, B., Balfanz, D., Durfee, G., Smeters, D.K.: Building an encrypted and searchable audit log. In: ACM Annual Symposium on Network and Distributed System Security (NDSS04) (2004)

13. Crossby, S.A., Wallach, D.S.: Efficient data structures for tamper-evident logging. In: Proceedings of the 18th Conference on USENIX Security Symposium (2009)
14. Chong, C., Peng, Z., Hartel, P.: Secure audit logging with tamper resistant hardware. Technical report TR-CTIT-02-29, Centre for Telematics and Information Technology, University Twente, The Netherlands (2002)
15. Wouters, K.: Hash-chain based protocols for time-stamping and secure logging: formats, analysis and design. Dissertation report. Arenberg Doctoral School of Science, Engineering and Technology, Katholieke Universiteit Leuven, Belgium (2012)
16. Pulls, T., Wouters, K., Vliegen, J., Grahn, C.: Distributed Privacy-Preserving Log Trails. Faculty of Economic Sciences, Communication and IT, Computer Science, Karlstad University Studies, Sweden (2012)
17. Accorsi, R.: BBox: a distributed secure log architecture. In: Camenisch, J., Lambrinoudakis, C. (eds.) EuroPKI 2010. LNCS, vol. 6711, pp. 109–124. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22633-5_8
18. Sinha, A., Jia, L., England, P., Lorch, J.R.: Continuous tamper-proof logging using TPM 2.0. In: Holz, T., Ioannidis, S. (eds.) Trust 2014. LNCS, vol. 8564, pp. 19–36. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-08593-7_2
19. Andersson, M., Nilsson, A.: Improving integrity assurances of log entries from the perspective of intermittently disconnected devices. Masters thesis no. MECS-2014-10, Faculty of Computing, Blekinge Institute of Technology, Sweden (2014)
20. Trusted Computing Group: Trusted Computing Group, TPM Library Specification. http://www.trustedcomputinggroup.org/tpm-library-specification/
21. Dijk, M.V., Sarmenta, L.F.G., Rhodes, J., Devadas, S.: Securing shared untrusted storage by using TPM 1.2 without requiring a trusted OS. Technical report, MIT Computer Science and Artificial Intelligence Laboratory (CSAIL) (2007)
22. Trusted Computing Group: ISO/IEC 11889–2:2009(E), Information technology Trusted Platform Module Part 2: Design principles, 2009 (2009). http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=50971
23. McCune, J.M., Parno, B., Perrig, A., Reiter, M.K., Isozaki, H.: Flicker: an Execution Infrastructure for TCB Minimization. In: Proceedings of the 3rd ACM SIGOPS/EuroSys European Conference on Computer Systems (2008)
24. Jain, P., Desai, S., Kim, S., Shih, M.W., Lee, J., Choi, C., Shin, Y., Kim, T., Kang, B.B., Han, D.: OpenSGX: an open platform for SGX research. In: Proceedings of the Network and Distributed System Security Symposium, NDSS (2016)
25. McKeen, F., Alexandrovich, I., Berenzon, A., Rozas, C., Shafi, H., Shanbhogue, V., Savagaonkar, U.: Innovative instructions and software model for isolated execution. In: Proceedings of the 2nd International Workshop on Hardware and Architectural Support for Security and Privacy (2013)
26. Anati, I., Gueron, S., Johnson, S.P., Scarlata, V.R.: Innovative technology for CPU based attestation and sealing. In: Proceedings of the 2nd International Workshop on Hardware and Architectural Support for Security and Privacy (2013)
27. Hoekstra, M., Lal, R., Rozas, C., Phegade, V., Cuvillo, J.D.: Using innovative instructions to create trustworthy software solutions. In: Proceedings of the 2nd International Workshop on Hardware and Architectural Support for Security and Privacy (2013)
28. Linux Integrity Measurement Architecture (IMA): Linux Integrity Subsystem. http://linux-ima.sourceforge.net/
29. Halcrow, M.A.: eCryptfs: An Enterprise-class Encrypted Filesystem for Linux. https://www.kernel.org/doc/ols/2005/ols2005v1-pages-209-226.pdf
30. Hein, D., Winter, J., Fitzek, A.: Secure block device - secure, flexible, and efficient data storage for ARM TrustZone Systems. In: TrustCom (2015)

31. Bratus, S., D'Cunha, N., Sparks, E., Smith, S.W.: TOCTOU, traps, and trusted computing. In: Lipp, P., Sadeghi, A.-R., Koch, K.-M. (eds.) Trust 2008. LNCS, vol. 4968, pp. 14–32. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-68979-9_2
32. rsyslog: The rocket-fast system for log processing. http://www.rsyslog.com/
33. PolarSSL: PolarSSL SSL library. https://tls.mbed.org/download/start/polarssl-1.3.9-gpl.tgz

Access Control and Availability Vulnerabilities in the ISO/IEC 61850 Substation Automation Protocol

James G. Wright[1] and Stephen D. Wolthusen[1,2(✉)]

[1] School of Mathematics and Information Security, Royal Holloway, University of London, Egham TW20 0EX, UK
james.wright.2015@live.rhul.ac.uk, stephen.wolthusen@rhul.ac.uk
[2] Norwegian Information Security Laboratory, Norwegian University of Science and Technology, Trondheim, Norway

Abstract. The ISO/IEC 61850 protocol for substation automation is a key component for the safe and efficient operation of smart grids, whilst offering a substantial range of functions. While extension standards, particularly ISO/IEC 62351 provide further security controls, the baseline protocol offers the assurances of access control and availability. In this paper a systematic study of selected aspects of the basic ISO/IEC 61850 protocol demonstrates that protocol-level vulnerabilities exist. The main finding is the development of a credential interception attack allowing an adversary, without credentials, to hijack a session during an initial association; the feasibility of this attack is proven using a formal language representation. A second attack based on a workflow amplification attack which relies on the assumptions in the protocol's substation event model, which is independent of layered security controls and only relies on the protocol's communication patterns is shown.

Keywords: Smart grid · ISO/IEC 61850 · Access control
Amplification attack · Substation automation protocol

1 Introduction

Smart grid technologies allow for more flexible generation and demand coordination whilst reducing costs with their greater bidirectional communication and control requirements [24]. However, this technological advancement degrades the "air gap" security principle that has been used in the power systems engineering community for the past few decades. The addition of networked intelligent electronic devices to the existing distribution infrastructure makes security through the obscurity of supervisory control and data acquisition (SCADA) protocols untenable, particularly since networks are increasingly interacting with internet protocols which is substantially increasing the attack surface.

Attacks against cyber-physical systems, such as electrical distribution networks, in recent history have shown that the threat is no longer a theoretical one.

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 239–251, 2017.
https://doi.org/10.1007/978-3-319-71368-7_20

Whether it is a direct attack against the industrial control systems itself [7], or an attempt to remove the ability to control as in the case of Shamoon [19], or to manipulate control as seen with BlackEnergy3 [21], comprehensive strategies are needed to protect critical infrastructure systems. Whilst both the academic and industrial research communities are now focusing on solving the unique security challenges faced in the deployment of smart grid technologies, there is very little focus dedicated to checking if the security promises made by the various smart grid protocols hold true. Having a secured protocol could prevent some of the theorised attacks against smart grids.

The following analysis focuses on the limited security objectives that are explicitly stated in IEC61850. These are access control and accessibility. It does not look at the objectives defined in IEC 62351, the protocol which is designed to extended the security specified of the information network controlling a smart grid's substation automation. The key contribution of this paper is to show that these explicit objectives are not upheld. A credential intercept attack against the protocol's two party association model is proved, through the use of context-free grammar, which undermines access control. It is also shown that the generic substation event model can be used against the smart grid's information network. A workflow amplification attack is shown, by example, to create the conditions to deny the flow of packets across communications infrastructure.

The remainder of this paper proceeds as follows: Sect. 2 describes the related work in the field. Section 3 then describes the aforementioned attacks against IEC61850, before giving conclusions and a sketch of future work in Sect. 4.

2 Related Work

Research into the cyber-physical security of power grids has been under way for over a decade, starting with North American Electric Reliability Corporation published its Critical Infrastructure Protection Cyber Security Standards [1]. However, there has been limited research into the specific threats facing individual protocols. There are plenty of taxonomies of attacks against general smart grid technologies [8,15,25,27], but only since 2010 have there been taxonomies focusing on specific attacks against IEC61850 [5,17,18]. Most of the theorised attacks against smart grids are either derivatives of computer network exploits, or an infiltration into the smart grid's information network via compromising the affiliated corporate network. Most taxonomies put forward solutions for there proposed attacks based upon their computer network counter parts, without considering if it will conflict with the quality of service requirements of the protocols. For example to validate the authenticity of the packets passing through the computer network, IEC62351 recommends using asymmetrical encryption schemes. However, as this comes into conflict with the latency requirements for packets declared in IEC61850, IEC62351 states "*for applications using GOOSE and IEC 61850-9-2 and requiring 4ms response times, multicast configurations and low CPU overhead, encryption is not recommended*" [11].

There has been some research directly focusing on attacks using IEC61850's generic object oriented substation events (GOOSE) multicast messaging service.

Hoyos *et al.* proposed a GOOSE spoofing attack where the adversary injects malicious copies of legitimate packets, but with the values in the data sets switched [6]. The aim of their attack is to get an intelligent electronic device to perform an undesirable action, like tripping a circuit breaker, by providing it with incorrect information. Another GOOSE attack authored by Kush *et al.* They developed a denial of service attack using the GOOSE status number variable [9]. In this attack the adversary injects a GOOSE message with a higher status number than all the legitimate GOOSE messages on the network. This forces the intelligent electronic device to process this malicious message before any legitimate ones.

Substantial efforts have been made to analyse and secure the older DNP3 protocol. Although it was designed to be a SCADA protocol that could be applied across the general spectrum of critical infrastructure, proposals have been made to use it in the substation automation space. East *et al.* published a taxonomy of attacks against DNP3, which specifically distinguishing how traditional network attacks can be applied against different abstraction layers of the communications network [4]. They also proposed the use of the security promises of awareness and control for SCADA systems. Mander *et al.* developed a system to extend the traditional IP security applied to DNP3, by creating a set of rules that are based upon DNP3's data objects to make sure that only legitimate packets flow across the network [14].

Finite state machines have been used to validate the general promises of communications protocols for decades [2]; however, they have only recently been applied to security promises. Poll and Rutiter used automata, along with black box fuzzing techniques, to show that session languages are usually poorly defined leading to vulnerabilities [16]. Wood and Harang proposed a framework for using formal language theory to secure protocols, as it is better at defining the data transiting between points of a network [26].

The use of context-free grammars has been applied to various aspect of the security theatre. Sassaman *et al.* used context-free grammars and pushdown automata to create a framework for a language based intrusion detection systems [20]. Liu *et al.* used probabilistic context-free grammar to prove that an adversary could impersonate authentication server in a Point-to-Point Protocol over Ethernet protocol [10].

3 Attack Taxonomy

Below the attacks against IEC61850 that serve to invalidate its stated security objectives are described. It is assumed throughout that the attacks are instigated on a reliable communications channel that are implemented on a network substrate, such as IP.

3.1 An Attack on Access Control: Credential Intercept Attack

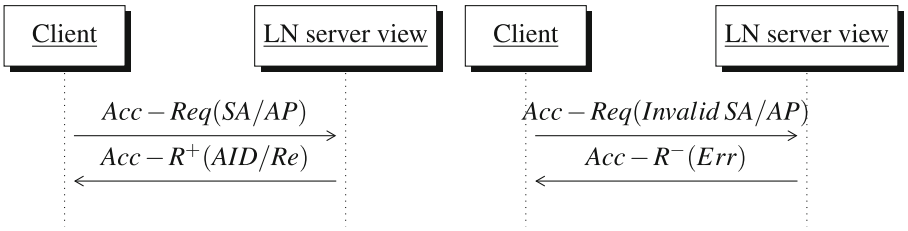
The first security promise that was analysed was access control. This is proposed in IEC61850-5 [13], as a solution to denial of service attacks against the

communication infrastructure of the grid. During the investigation it was found that an attack against two party association model, described in IEC61850-7-2 section 8.3, undermines this promise [12].

An adversary, who has no login credentials on the network, is able to hijack the login credentials of legitimate user whilst they are logging into their logical node server view. This attack can be perpetrated against a client that is logging into the logical node for the first time, or who hasn't already been given a predetermined authentication parameter. This scenario is predicated on the adversary doing some passive surveillance of the communications network, as the two party association model is only instigated when a new entity is connected to the it. Once this precondition is fulfilled the attacker is able to proceed with the attack.

The Two Party Association Model. The two party association model describes how a client program can connect and transfer packets with a logical node server view. The standard procedure for the model is that the client sends an access request, shortened to $Acc - Req(SA/AP)$, message to a virtual view on the logical node server, LN. Included within the request are the client's login credentials, which includes an authentication parameter, AP, and the server access point reference, SA.

Once the server has received this request, it then decides how to proceed. If the client's login credentials are correct then the server will reply to the client with an affirmative message, $Acc - R^{+}(AID/Re)$, that will include an authentication ID, AID, and the result of the attempt, Re. However, if the client's login credentials are invalid then the server will reply in the negative, $Acc - R^{-}(Err)$, with an error message, Err (Fig. 1).



(a) The logical nodes affirmative response (b) The logical nodes negative response

Fig. 1. Session diagram depicting the two party association model.

The Adversary Model. The adversary in the this scenario is based on upon the one described by the Dolev-Yao model [3]. This adversary is constrained by the following requirements:-

– The adversary can see all packets passing between the client and the LN server

- The adversary cannot send any message that they have not already seen.
- The adversary has no buffer on messages they have seen. They have to send the message directly after seeing it.
- The adversary can forward and intercept packets.

Whilst there are some similarities, the reason that this model doesn't duplicate the Dolev-Yao model is the protocol being attacked is a SCADA protocol rather than a cryptographic one.

The Attack Premise. The attack happens by combining the two potential responses of the server into one session. It begins when the client sends a legitimate login request to their own virtual view of the LN server. The adversary sees the client's packet go through their intercept and then sends a invalid login attempt to their own server view. When the client's view responds in the affirmative with the authentication ID, the adversary intercepts this packet. When the adversary's view replies in the negative, the adversary forwards the packet with the error message to the client. After this the client cannot use their login credentials (Fig. 2).

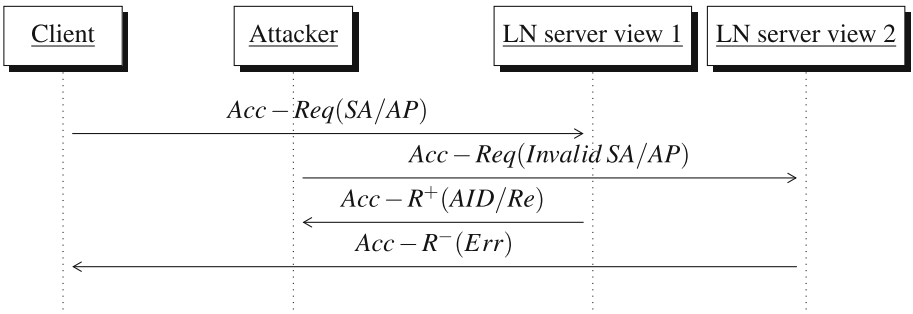


Fig. 2. Session diagram of the proposed attack.

The Automata [22]**.** Figures 3 and 4 depict the automaton modelling the process of one client logging in, and the union of two one client automata to form one that can model two users logging in simultaneously. The two user automaton allows the depiction of the attack described in the previous section. In the two person automaton S represents the standby state, C represents the check state, and A represents the awaiting state.

The Context Free Grammar. The rules that describe a legitimate message that passes through the two person automaton are as follows:-

- A login attempt for one access view must be completed before a second login can be attempted.

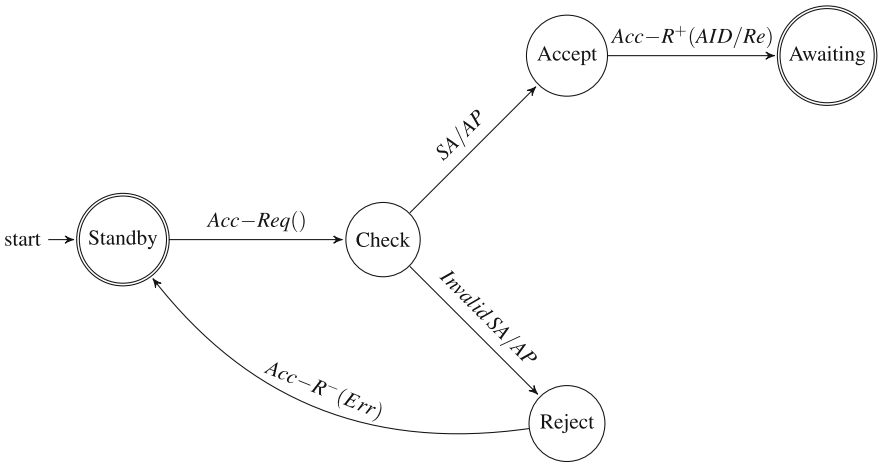


Fig. 3. The automaton depicting one client logging into a logical node server.

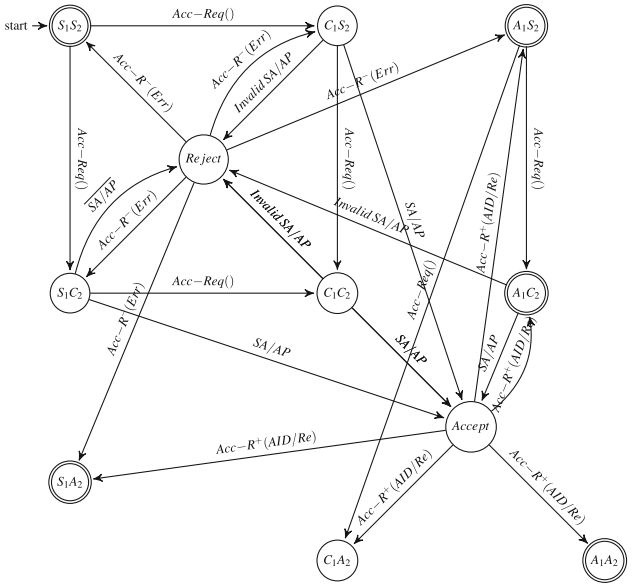


Fig. 4. The automaton depicting two clients logging into a logical node server simultaneously.

– There can only be two successful attempts per run of the automaton.
– An infinite number of failed attempts can be made before the first successful message and between the first and final successful message.

The following rules describe the form of the message that can pass through the two person automaton that leads to an undesired result:-

- The adversary can only duplicate a message that has passed their intercept.
- The adversary's duplicate message can only be sent immediately after seeing it. They have no buffer.
- The adversary's '$Acc - Req()$' must come before the client's 'AP/SA' is processed by their login view.
- The adversary can only send invalid SA/AP credentials. This is to make sure it ends up in the state they desire (S_1A_2 or A_1S_2)
- The legitimate user cannot login after the attacker has intercepted their credentials.

The objective of the adversary is to make sure that the automaton is driven through the C_1C_2 state. This collision state represents the adversary's intercept where they hijack the authentication ID and forward their error message.

The context free grammar that represent the above rules are as follows:-

$$
\begin{array}{c}
\hline
S \rightarrow TATV|TWTW|TV|TW \\
R \rightarrow Acc - Req() \\
V \rightarrow Invalid\ SA/AP\ Acc - R^{-}(Err) \\
W \rightarrow SA/AP\ Acc - R^{+}(AID/Re) \\
T \rightarrow RU \\
U \rightarrow VT|\epsilon \\
A \rightarrow W|RVW|RWV \\
\hline
\end{array}
$$

Mapping to IEC61850-7-2. The above grammar maps to the two party association model in the following way:

- Rule S: Presents the four different message types. From left to right.
  1. Is the comprised attack form. If the attack is not attempted this leads to $n = 0...$ failed attempts, followed by a successful login and then another $n = 0...$ failed logins. However if rule inserts either 'RVW' or 'RWV' instead of 'W', then the undesired form of the message begins. This leads to two '$Acc{-}Req()$' messages in a row. They can both be seen as undesired as the attacker controls all messages passing through its intercept.
  2. A word with two successful logins with $n = 0...$ failed messages before the first and between the subsequent successful logins.
  3. $n = 0...$ failed logins.
  4. $n = 0...$ failed logins followed by one successful message.
- Rule R: Maps to the request message parameter.
- Rule V: Maps to the incorrect form of 8.3.2.2.2.1, the server access point reference, and 8.3.2.2.2.2, the authentication parameter. Followed by 8.3.2.2.5, response showing the failed attempt error, which "*shall indicate that the service request failed*".
- Rule W: Maps to the correct form of 8.3.2.2.2.1, the server access point reference, "*which shall identify the server, with which the application association shall be established*", and 8.3.2.2.2.2, the authentication parameter, "*for this application association to be opened*". Followed by 8.3.2.2.3, response showing

the successful login returning the authentication ID, which "*may be used to differentiate the application associations*", and request message, which indicates "*if the establishment of the application association was successful or not*".

- Rule T: Is the rule that facilitates the $n = 0...$ repeats of the failed login, or it provides an '$Acc - Req()$' packet before terminating the loop.
- Rule U: Provides the terminals to facilitate rule 'T'.
- Rule A: Is the production rule for the attack. From left to right.
  1. Facilitates the normal success message stuck between to infinite failed attempts.
  2. Two '$Acc - Req()$' packets followed by a failed login attempt and then a successful login.
  3. Like 2, but the error and success messages are reversed.

  2 and 3 are the undesired message forms

The above shows that the security promise of access control does not hold for the two party association model.

3.2 An Attack on Availability: Generic Workflow Event Amplification Attack

The second security promise that was analysed was that of availability of service. This promise is proposed in IEC61850-5 section 13 as the general message performance requirements. During the investigation it was found that an attack using the generic substation event class model, as described in IEC61850-7-2 section 15, could be used to create a denial of service of attack to undermine this promise.

The aim of the adversary in this scenario is to degrade the performance of packet transfer between points on the smart grid topology to below the acceptable standard. The adversary achieves this by sending messages that connects additional subscribers or topological branches to a LN's generic substation event subscriber list. This leads to the routers and LNs on the network having to process, and potentially discard, extra messages. Whilst the analysis allows for the calculation of the number of extra bits processed by the grid, it doesn't cover the additional latency. This is due to the amount of time for a computation to take place on a LN being beyond the scope of IEC61850.

This analysis assumes that generic substation event model has been implemented on PIM multicast framework that has been applied to a network substrate that supports it.

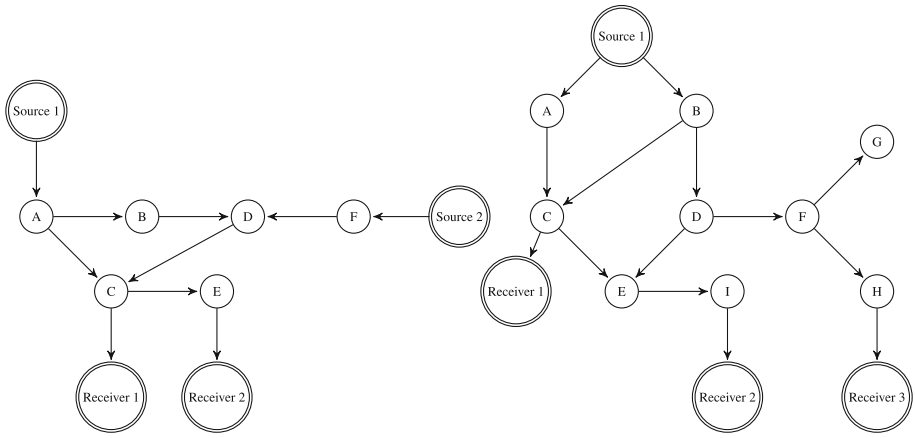
The Generic Substation Event Class Model. The generic substation event class model describes the way a LN can broadcast data regarding its current, or changing, status to the devices that subscribe to its announcements. It is based on a producer/subscriber multicast model, and is implemented as an unidirectional process. There are two types of message in this model. Firstly, the

GOOSE message, that is used to broadcast the LN's data, and the second is the generic substation state events (GSSE) message, which broadcasts any changes in state. When a LN is connected to the network it sends a GOOSE message that announces to devices its current status.

A LN on a generic substation event network will only check to see if the message it has received is a duplicate of a previous message, or if parts of it are missing. The method used to check for this is, again, beyond the scope IEC61850. In this analysis it is assumed that the logical node does not have access to the complete address space of the network and the packets received aren't cryptographically signed.

PIM Multicast [23]**.** In PIM sparse mode when a receiver issues a join request to be added to the network, a reverse path forwarding (RPF) check is triggered. A PIM-join message is sent toward rendezvous point (RP), in Fig. 5a that is *D*. The join message is multicast hop by hop upstream to the all the PIM routers until it reaches the RP. The RP router receives the PIM-join message and adds it to the outgoing interface list. The same process is done for when a router wishes to leave the network, but instead sends a PIM-prune message. When a source is added, it multicasts a PIM-register message and sends them by means of unicast to the RP router.

In PIM dense mode the outgoing interface list is created by the source sending out a PIM-flood message periodically. This registers all devices on the network to the list. If a receiver no longer wishes to be on the list, it sends a PIM-prune message upstream to the source, which then removes it from the list. If a new receiver wishes to join before the next PIM-flood, they can send a PIM-graft message to the source to be added.



(a) A sparse topology. (b) A dense topology.

Fig. 5. Examples of PIM multicast system topology. [23]

The Adversary Model. The adversary model is the same as the one described in Sect. 3.1, however this adversary has a buffer so they are not required to send messages they have discerned straight away.

The Attack Premise. The attack is predicated on the adversary performing some passive surveillance on the communications network. Through the adversary's observation they decide which branch or LN they which to attach to the LN's subscriber list, and which type of PIM network it is. They also discern the logical node's specific application ID, which is required for subscribers to receive the GOOSE messages. The adversary sends either a PIM-flood or PIM-graft, for dense PIM networks), or PIM-join or PIM-register for sparse PIM-networks. The next time the publisher LN sends out a generic substation event message to the network the LNs that have been maliciously subscribed to the network will receive messages they weren't expecting. As they have no access to the address space they cannot tell whether they were meant to receive the message.

The Workflow Amplification Factor. The workflow amplification factor describes the ratio of messages produced to messages the adversary sent to initiate the attack in the number of bits.

$$\text{Amplification factor} = \frac{\text{Message produced as a consequence of the attack}}{\text{Messages sent by the adversary}} \tag{1}$$

The amplification factor for a GOOSE message is,

$$\text{Amplification factor}_{\text{GOOSE}} = \frac{A + \text{length of data set} + \text{data set}}{B + C}, \tag{2}$$

where is $A = 187$, $B = 65$, and $C = 4 or 32$ depending on whether the adversary chooses to edit the PIM message type, or create a new PIM message.

For a GSSE,

$$\text{Amplification factor}_{\text{GSSE}} = \frac{D + (2 * \text{length of data set})}{B + C}, \tag{3}$$

where $D = 170$

Examples. Below are example calculations given for the workflow amplification factor for an adversary instigating the attack from various points in the networks depicted in Figs. 5a and b. All of the below examples takes the average number of status logical node variables, which is three, as the length of the data set. As the status variables are usually a boolean variable type, it is assumed for these calculations that they are boolean. For the purpose of these examples the adversary will create a whole new PIM message for their attack.

	AF_{GOOSE}	AF_{GSSE}
Case 1	3.96	3.63
Case 2	23.75	22.14
Case 3	11.87	11.07

Case 1 is set in the depicted dense PIM network. In this case the adversary has chosen to connect router I to the network, so to send malicious messages to *Receiver* 3. Case 2 is when the adversary connects a new source to the network.

Case 3 is the attack scenario applied to the sparse PIM network example. In this instance the adversary connects source 2 to the network to send malicious messages to both *Recievер* 1 and *Receiver* 2. In the case of adding another receiver to the network, the amplification factor would be the same as case 1.

4 Conclusion

The above analysis has shown that the explicit security promises of IEC61850 are not upheld throughout the protocol's technical specification. A credential intercept attack has been developed and proved using context-free grammar against the two party association model. This attack undermines the promise of access control, and would allow the adversary to potentially completely control a logical node if the they intercepted someone with administrative privileges. This scenario would allow them to cause physical damage to the smart grid, for example they could trip circuit breakers and cause undue stress on the distribution network. The second attack developed undermined the security promise of accessibility. It was shown by example that a workflow amplification type denial of service attack could be instigated against an intelligent electronic device by an adversary generating a malicious message that would connect the target node to a GOOSE subscriber list that it did not want to receive messages from. The denial of service comes from the intelligent electronic device having to process more messages than it was expecting. The scale of the amplification factor of the attack is proportional to the number of nodes and routers that have to process the extra malicious messages.

Although the attacks mentioned above are limited to IEC61850, there is a reasonable likelihood that other smart grid protocols, such as DNP3, will also be found deficient when upholding their security promises. The above methodologies can be used to perform the same analysis on these protocols to develop, and attempt to mitigate, such attacks.

Progressing onwards from the above analysis the intention is to see if there are any other protocol models that contain flaws that would undermine the security promises we have access to. The next vector of attack that has been considered is to see if we can get a client and/or the logical node server to be uncertain what state it is in due to an interruption in the communication channel. It is hoped that it will be possible to formally verify theses future attacks with a context-free grammar approach.

Once this line of inquiry has been exhausted, the focus of the investigation will proceed to see if the attacks that have been discovered can still be executed when IEC62351 has been used to secure IEC61850.

Acknowledgement. This work is supported by an EPSRC Academic Centres of Excellence in Cyber Security Research PhD grant.

References

1. NERC implementation plan for cyber security standards CIP-002-1 through CIP-009-1. Technical report, NERC, 2006
2. Brand, D., Zafiropulo, P.: On communicating finite-state machines. J. ACM **30**(2), 323–342 (1983)
3. Dolev, D., Yao, A.: On the security of public key protocols. IEEE Trans. Inf. Theor. **29**(2), 198–208 (1983)
4. East, S., Butts, J., Papa, M., Shenoi, S.: A taxonomy of attacks on the DNP3 protocol. In: Palmer, C., Shenoi, S. (eds.) ICCIP 2009. IAICT, vol. 311, pp. 67–81. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04798-5_5
5. Elgargouri, A., Virrankoski, R., Elmusrati, M.: IEC 61850 based smart grid security. In: 2015 IEEE International Conference on Industrial Technology (ICIT), pp. 2461–2465, March 2015
6. Hoyos, J., Dehus, M., Brown, T.X.: Exploiting the GOOSE protocol: a practical attack on cyber-infrastructure. In: 2012 IEEE Globecom Workshops, pp. 1508–1513, December 2012
7. Karnouskos, S.: Stuxnet worm impact on industrial cyber-physical system security. In: 37th Annual Conference on IEEE Industrial Electronics Society, IECON 2011, pp. 4490–4494, November 2011
8. Konstantinou, C., Maniatakos, M., Saqib, F., Hu, S., Plusquellic, J., Jin, Y.: Cyber-physical systems: a security perspective. In: 2015 20th IEEE European Test Symposium (ETS), pp. 1–8, May 2015
9. Kush, N., Ahmed, E., Branagan, M., Foo, E.: Poisoned GOOSE: exploiting the GOOSE protocol. In: Proceedings of the Twelfth Australasian Information Security Conference, AISC 2014, Darlinghurst, Australia, vol. 149, pp. 17–22. Australian Computer Society Inc. (2014)
10. Liu, F., Xie, T., Feng, Y., Feng, D.: On the security of PPPoE network. Secur. Commun. Netw. **5**(10), 1159–1168 (2012)
11. TC 57 Power Systems Management and Associated Information Exchange: Power systems management and associated information exchange, data and communication security. IEC standard 62351. Technical report, International Electrotechnical Commission (2007)
12. TC 57 Power Systems Management and Associated Information Exchange: Communication networks and systems for power utility automation - Part 7–2: basic information and communication structure - abstract communication service interface. IEC standard 61850-7-2. Technical report, International Electrotechnical Commission (2010)
13. TC 57 Power Systems Management and Associated Information Exchange: Communication networks and systems for power utility automation - Part 5: communication requirements for functions and device models. IEC standard 61850–5. Technical report, International Electrotechnical Commission (2013)
14. Mander, T., Nabhani, F., Wang, L., Cheung, R.: Data object based security for DNP3 over TCP/IP for increased utility commercial aspects security. In: 2007 IEEE Power Engineering Society General Meeting, pp. 1–8, June 2007
15. Mo, Y., Kim, T.H.J., Brancik, K., Dickinson, D., Lee, H., Perrig, A., Sinopoli, B.: Physical security of a smart grid infrastructure. Proc. IEEE **100**(1), 195–209 (2012)
16. Poll, E., Ruiter, J.D., Schubert, A.: Protocol state machines and session languages: specification, implementation, and security flaws. In: 2015 IEEE Security and Privacy Workshops (SPW), pp. 125–133, May 2015

17. Premaratne, U., Samarabandu, J., Sidhu, T., Beresh, R., Tan, J.C.: Security analysis and auditing of IEC61850-based automated substations. IEEE Trans. Power Deliv. **25**(4), 2346–2355 (2010)
18. Rashid, M.T.A., Yussof, S., Yusoff, Y., Ismail, R.: A review of security attacks on IEC61850 substation automation system network. In: 2014 International Conference on Information Technology and Multimedia (ICIMU), pp. 5–10, November 2014
19. Kaspersky Lab's Global Research and Analysis Team: Shamoon the wiper copycats at work. https://securelist.com/blog/incidents/57854/shamoon-the-wiper-copycats-at-work/
20. Sassaman, L., Patterson, M.L., Bratus, S., Locasto, M.E.: Security applications of formal language theory. IEEE Syst. J. **7**(3), 489–500 (2013)
21. Shamir, U.: Analyzing a new variant of BlackEnergy 3 likely insider-based execution. Technical report, SentinelOne (2016)
22. Sipser, M.: Introduction to the Theory of Computation, 1st edn. International Thomson Publishing, Boston (1996)
23. Cisco Systems: IP multicast technology overview. https://www.cisco.com/c/en/us/td/docs/ios/solutions_docs/ip_multicast/White_papers/mcst_ovr.html
24. Wang, W., Lu, Z.: Survey cyber security in the smart grid: survey and challenges. Comput. Netw. **57**(5), 1344–1371 (2013)
25. Wei, D., Lu, Y., Jafari, M., Skare, P.M., Rohde, K.: Protecting smart grid automation systems against cyberattacks. IEEE Trans. Smart Grid **2**(4), 782–795 (2011)
26. Wood, D.K.N., Harang, D.R.E.: Grammatical inference and language frameworks for LANGSEC. In: 2015 IEEE Security and Privacy Workshops (SPW), pp. 88–98, May 2015
27. Yang, Y., Littler, T., Sezer, S., McLaughlin, K., Wang, H.F.: Impact of cybersecurity issues on smart grid. In: 2011 2nd IEEE PES International Conference and Exhibition on Innovative Smart Grid Technologies (ISGT Europe), pp. 1–7, December 2011

A Case Study Assessing the Effects of Cyber Attacks on a River Zonal Dispatcher

Ronald Joseph Wright[1(✉)], Ken Keefe[2], Brett Feddersen[2], and William H. Sanders[1]

[1] Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA
{wright53,whs}@illinois.edu

[2] Information Trust Institute, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA
{kjkeefe,bfeddrsn}@illinois.edu

Abstract. A river zonal dispatcher is a system that sends collected environmental data to a national dispatcher and sends warnings in case of danger (such as flooding of river basins). If the system fails to function normally, warnings may cease, putting lives and property in serious peril. We have examined the security of a river zonal dispatcher using the ADVISE modeling formalism in the Möbius modeling tool. This work both illustrates the usefulness of ADVISE in choosing among alternative approaches to system security and provides a quantitative evaluation of the dispatcher itself. In doing so, it shows whether intrusion detection systems (IDSes) make a difference in the behavior of an adversary, and which path of attack is most attractive to particular types of adversaries.

Keywords: Control systems security · Quantitative security metrics
State-based security model · Discrete event simulation

1 Introduction

Critical infrastructures must be resilient to real-world threats. According to the Department of Homeland Security, dam systems are dependent and interdependent on a multitude of sectors, such as the water sector (for delivering potable water to customers) and the emergency services sector (for delivering water in case of emergencies such as firefighting). Dam systems contribute to numerous projects, such as hydroelectric power generation, navigation control, levees, and waste impoundments [10,11]. There is a need to be vigilant against any threats to the integrity of such water control systems so that they function with little or no interruption.

Incidents have occurred that have threatened the integrity of water control systems. In 2001, a former employee of a sewage treatment plant in Queensland, Australia, maliciously released over 264,000 gallons of raw sewage, which flooded

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 252–264, 2017.
https://doi.org/10.1007/978-3-319-71368-7_21

nearby rivers and parks [12]. In 2006, a foreign attacker installed malicious software remotely on systems at a water filtering plant in Harrisburg, Pennsylvania, negatively affecting the operations of the plant [12]. Because those systems are similar to dam systems, the same forms of attack can also target dam systems.

Supervisory Control and Data Acquisition (SCADA) systems are a major part of many industrial control systems. What makes them critical is the fact that they are centralized and communicate with large-scale system components that typically operate on entire sites [9]. Components in most SCADA systems communicate with each other using the Modbus protocol, which offers no protections against denial-of-service (DoS) attacks and malicious data modification. Moreover, today's SCADA systems are based on open standards, so attackers can easily learn how they work.

To study the effects of specific attack behaviors on a water control system, we developed a case study of a river zonal SCADA dispatcher. It demonstrates that different system configurations can play a role in protecting a system, and that attack targets vary depending on the type of attacker. The main contribution of this work is an analysis of different attack scenarios and attacker types through stochastic modeling and quantitative metrics.

In Sect. 2, prior work related to the security evaluation of river zonal systems is discussed. Section 3 details the configurations that we considered. Section 4 introduces the modeling formalism, as well as the models used to analyze different attack scenarios and attacker types. Section 5 describes the experimental setup to carry out the analysis; Sect. 6 discusses the experimental results and analysis. Finally, Sect. 7 concludes the paper with a final discussion, including possible future improvements to this work.

2 Related Work

The authors of [1] study common attacks on SCADA control systems, such as command injection, data injection, and denial of service attacks. They used actual systems in a physical lab, such as a water storage tank control system, to study the effects of these attacks on the systems' operation, and designed an anomaly detection system that analyzes whether the data is normal or abnormal. Our approach is different in that it generalizes the anomalies by modeling attacks and intrusion detections and investigates how long it takes for an attack to unfold.

Common types of attacks on water control systems are provided in [6]. The authors grouped the attacks into four different classes: reconnaissance, response and measurement injection, command injection, and denial of service. *Reconnaissance attacks* gather control system information to help attackers locate hosts to attack, and *injection attacks* involve corruption of responses to make the system function abnormally. Reconnaissance and injection attacks were used in our study.

In [13], attacks on SCADA systems were described and classified by possible attack targets. Classifications include access control, memory protection,

and protocol failure due to bugs. One interesting type of attack described in [13] is an Open Platform Communications (OPC) attack, in which an attacker compromises an HMI and then executes arbitrary code or attacks other servers. We used that attack to model how devices can be controlled by outsiders without the use of an HMI.

The authors of [7] described a river zonal dispatcher system, a SCADA traffic monitoring system for detecting potential intrusions, and a software agent model for the intrusion detection system for forensic analysis. We also considered the role of intrusion detection systems in the dispatcher, but we simply used the concept to analyze quantitative timings so that we could determine how they play a role in inhibiting attacks.

3 Background

The system we examined involves a river zonal dispatcher described in [7]. A simplification of its architecture is shown in Fig. 1. The architecture consists mainly of four networks: a management LAN, an operations LAN, a supervisory LAN, and a bus network composed of on-site devices such as remote terminal units (RTUs), programmable logic controllers (PLCs), and intelligent electronic devices (IEDs). The management LAN consists of servers responsible for water resource management, and the operations LAN consists of servers collectively responsible for forecasting future outcomes given historical data. Each of the management LAN, operations LAN, and supervisory LANs are protected by an intrusion detection system (IDS) to detect possible intrusions. A supervisory LAN and a bus network together make up the meat of the system, and for this particular case study, there are two groups of these two types of networks. Each supervisory LAN consists of a human-machine interface (HMI),

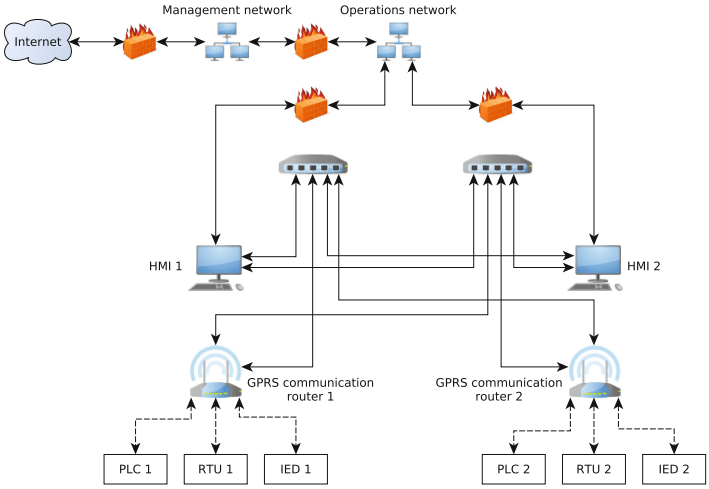


Fig. 1. River zonal SCADA dispatcher architecture without isolation.

and each bus network consists of on-site devices. Each supervisory LAN is connected to its corresponding bus network through a General Packet Radio Service (GPRS) communication router, which is used for wireless communication between HMIs and devices that use Modbus, a widely used protocol for industrial communication systems [5]. In the Modbus protocol, an HMI sends special unencrypted packets containing a function code with parameters specific to the function of a device, and then the device typically replies back by echoing the original function code and returning the output data. Typically, communication is done through an Open Platform Communications (OPC) server that converts outgoing HMI commands to the proper Modbus commands, and converts incoming replies back to a format that the HMI can recognize. Sensor networks are often used in SCADA systems to collect and use information such as hydrometric, pluviometric, and meteorological data, so the main purpose of the HMI is to use information from the sensors to decide how to control a particular device.

The first supervisory LAN controls one subriver basin, and the second supervisory LAN controls the other subriver basin; however, the first subriver basin is assumed to cascade into the second one. Therefore, the basin corresponding to the first supervisory LAN is referred to as the *upper* subriver basin, and the basin corresponding to the second supervisory LAN is referred to as the *lower* subriver basin.

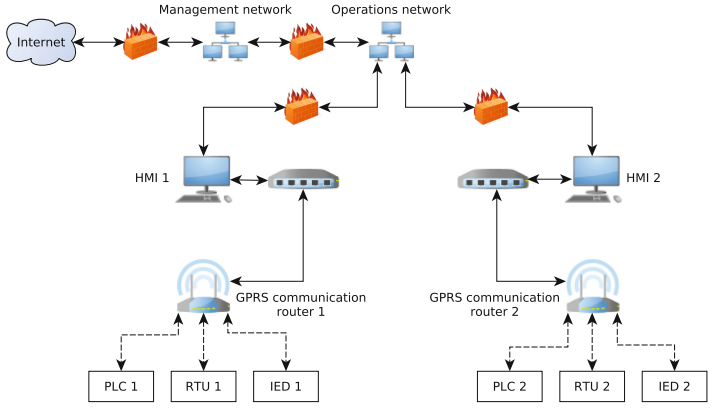


Fig. 2. River zonal SCADA dispatcher architecture with isolation.

Some assumptions were made: (1) the most critical part of the architecture is the SCADA system and is therefore the most desirable place for an adversary to target, which means that any given attack path from the Internet to the SCADA system is just for the purpose of gaining access to the SCADA system, and (2) the SCADA server and HMI reside on a single machine. [7] assumes that the SCADA systems are accessible from each other, but one potential way of minimizing the impact of attacks is by isolating the different SCADA systems in the dispatcher system, as illustrated in Fig. 2. In this case, the operator of an HMI

cannot use one HMI to access another HMI, and a one-to-one correspondence exists between HMIs and GPRS communication routers, so the operator can access devices through one router only.

4 Methodology

4.1 Attacks

The attack behavior in the system was modeled in the ADVISE formalism [2,3], which is a stochastic model that assumes the perspective of an adversary who selects an optimal attack based on its level of attractiveness compared to all other attacks available to him or her. The ADVISE formalism consists of an attack execution graph that contains a collection of attack steps. Attack step preconditions dictate whether it is possible at a certain time for an attack step to be executed, and outcomes make up the effects of the attack step. Access, skill, knowledge, and goal state variables are connected to attack steps in the attack execution graph, signifying that the connected state variables are used in the preconditions (arcs from state variable to attack step) or effects (arcs from attack step to state variable) of the attack step. In an attack execution graph, yellow rectangles represent attack steps, and red squares, green circles, blue triangles, and orange ovals represent access, knowledge, skills, and goals, respectively.

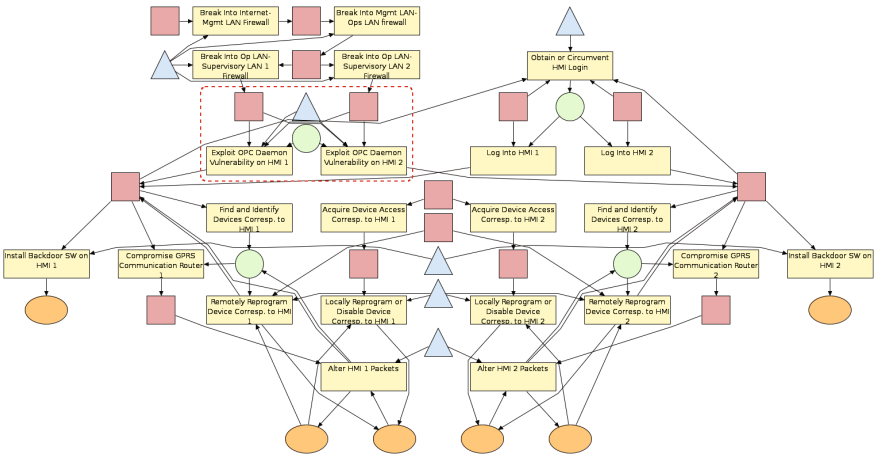
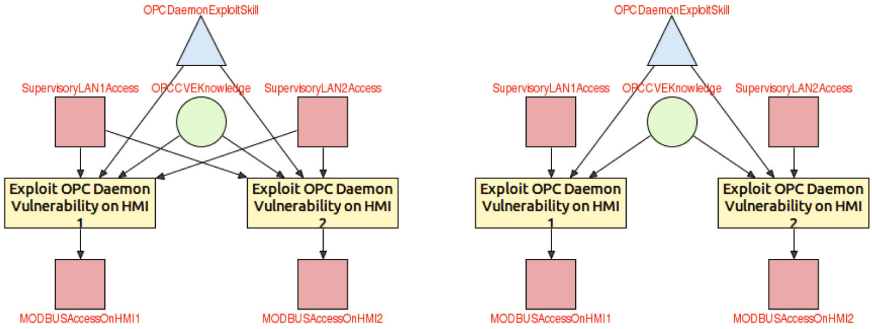


Fig. 3. ADVISE model of the river zonal SCADA dispatcher. (Color figure online)

The ADVISE model for the river zonal dispatcher is shown in Fig. 3. The section of the model in the dashed box exhibits different behavior depending on whether the supervisory LANs are isolated. The corresponding two cases are shown in Fig. 4. Specifically, if the supervisory LANs are isolated, then



(a) HMI attacks without isolation. (b) HMI attacks with isolation.

Fig. 4. Detail of dashed section in Fig. 3. (Color figure online)

exploitation of the OPC daemon vulnerability is not possible on supervisory LAN 1 through supervisory LAN 2 and vice versa. This means that the edge from *SupervisoryLAN1Access* to the HMI 2 exploitation step and the edge from *SupervisoryLAN2Access* to the HMI 1 exploitation step, seen in Fig. 4a, have no effect on enabling those attack steps, so they effectively disappear, resulting in Fig. 4b. However, if the network is not isolated (see Fig. 4a), then exploitation of the OPC daemon vulnerability is possible on supervisory LAN 1 through supervisory LAN 2 and vice versa.

In order to attack the SCADA system, an outside attacker must first gain access to the management LAN, operations LAN, and any of the supervisory LANs containing the HMI. Those attacks correspond to attacking the firewalls that lie between the networks; those attacks are represented as four attack steps at the top left of Fig. 3. Access to the supervisory LAN is one requirement for gaining control of the SCADA system. The ability to send commands to the on-site devices is another. The attacker may already have login access to an HMI (if he is an operator), or he can simply steal the password (if he is some other type of insider). If the attacker is an outsider, he can exploit the OPC server and issue commands to devices from there. In the case of non-isolated SCADA systems, gaining access to any of the HMIs grants the attacker access to all devices in the dispatcher, and in the case of isolated SCADA systems, gaining access to a single HMI grants the attacker access to devices through just one of the routers. Once the adversary has control of the HMI, he or she can attack the system in four possible ways: (1) install backdoor software on the HMI; (2) compromise the GPRS communication router that allows the HMI to interact with the devices, and then maliciously alter packets going through the router; (3) remotely reprogram the devices via the HMI so that they behave maliciously; or (4) directly reprogram the devices at the corresponding bricks-and-mortar facility so that they behave maliciously. In all, barriers to the attack include the need to defeat firewalls that connect pairs of networks together, to log into HMIs, or to compromise OPC servers, and to slip past IDSes that are placed on every LAN to detect possible intrusions.

Packets going through the router can be maliciously altered through injection attacks, as described in [6]. A more sophisticated attack, on the other hand, involves reprogramming of the devices at the facilities. If a device is equipped with function code 126 access [8], which allows a device to be remotely reprogrammed, then an attacker can simply use the HMI to find and identify the devices and remotely alter their behavior. If the attacker is an insider and has access to the bricks-and-mortar facility that houses a device, all he or she needs to do is enter the facility and directly reprogram the device. This type of attack satisfies the goal of compromising the device at that particular subriver facility.

One important assumption regarding these attack models is that no defense model is present, so when an attack goal is met, it remains met for the remainder of time. However, the model considers the effects of IDSes, which affect attack steps involving the management LAN-operations LAN firewall, operations LAN-supervisory LAN firewalls, OPC daemon exploitation, HMI backdoor software installation, GPRS communication router compromise, router injection, and reprogramming of devices. In these cases, all accesses upon which the attack depends are restored to the initial state when those attacks are detected.

4.2 Response

A response model complements the ADVISE model and restores the state of the system after an attack is carried out. The response behavior in the system was modeled with the Stochastic Activity Network (SAN) formalism [4], which is an extension of Petri nets.

Whenever backdoor software is installed on any of the HMIs, it must first be detected, and then the actual repair process of uninstalling the backdoor takes place. The repair process restores the initial access that the adversary had; i.e., insider attackers still have access to the HMIs after the repair process completes, but outside attackers lose that access. Also, the repair process forces the attacker to re-achieve the goal of installing the backdoor software when he or she has the chance. Whenever the system is compromised via a device or router, someone must recognize that the system is operating abnormally, and then the actual repair process of restoring the functionality of the router or device takes place. Just like the repair process of uninstalling backdoors, the device and router functionality repair processes restore the initial access that the adversary had, and they force the attacker to re-achieve the goals of compromising the system via the device or router.

5 Experiment

To understand the different behaviors of each adversary and the various conditions that can increase the difficulty of an attack, simulations of the models were executed using different types of adversaries with and without IDSes present in the system. The goal of these experiments was to understand the types of scenarios that can negatively impact the security of the system and to determine

the best practices for protecting it. Five types of adversaries were considered: (1) a foreign government, (2) a lone hacker, (3) a hostile organization, (4) an insider engineer, and (5) an insider operator. A foreign government is primarily concerned with installing backdoors on the HMIs and cares little about costs. A hacker is interested in most of the possible goals and is highly skilled, but must consider a balance of concern regarding cost, payoff, and detection. A hostile organization is also highly skilled, but is interested only in compromising the supervisory LANs and is mostly seeking the best payoff. The insider engineer is interested in all goals, but is poorly skilled in attacks, while the insider operator has access to many parts of the system already, is highly skilled, and is primarily concerned with reprogramming the devices.

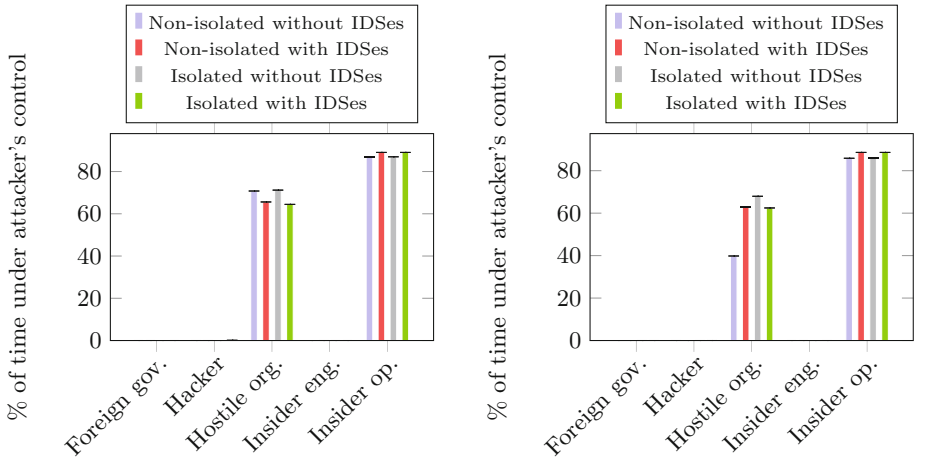
There were a total of 20 cases for each simulation, as there are five different types of adversaries, supervisory LANs may be isolated or non-isolated, and IDSes may be present or not present. For each case, the percentage of the simulated time that an attacker has control of an HMI, router, or device was studied, as well as the percentage of time the adversary takes to attack the system before reaching his goal. The systems were simulated for up to 8,760 h (i.e., one year). The simulations were run for a minimum of 1,000 iterations and continued to run until either a 90% confidence interval for all measurements or a maximum of 10,000 iterations was achieved.

6 Results and Analysis

In the results that follow, a zero cost and detection probability of 0.95 were specified for the do-nothing step so that the attacker does not give up too easily when trying to accomplishing attack goals [2,3].

6.1 Control of Device

Figure 5 shows the percentages of time that the subsystems were compromised via devices with respect to different attackers. As expected, the foreign government did not see gaining control of the system through devices as the most attractive goal. For the most part, the hacker also did not see any value in gaining control of the system through devices for the most part. However, we did observe one interesting result, which happened so rarely in the simulation that it can barely be seen in the figure: the attacker was able to penetrate the devices only when IDSes were enabled and indeed chose to do so, even though this attack was not in his best interests cost-wise, detection-wise, and payoff-wise. Specifically, at one point in the simulation, after finding the devices to compromise, rather than gain control of the system by launching router attacks or simply do nothing, the hacker saw that reprogramming the devices remotely was the most attractive course of action to take. The hostile organization proved to be successful in reprogramming all the SCADA devices in the system. It reprogrammed the devices corresponding to HMI 2 far less often than the devices corresponding to HMI 1 in a non-isolated system without IDSes, but reprogrammed the devices



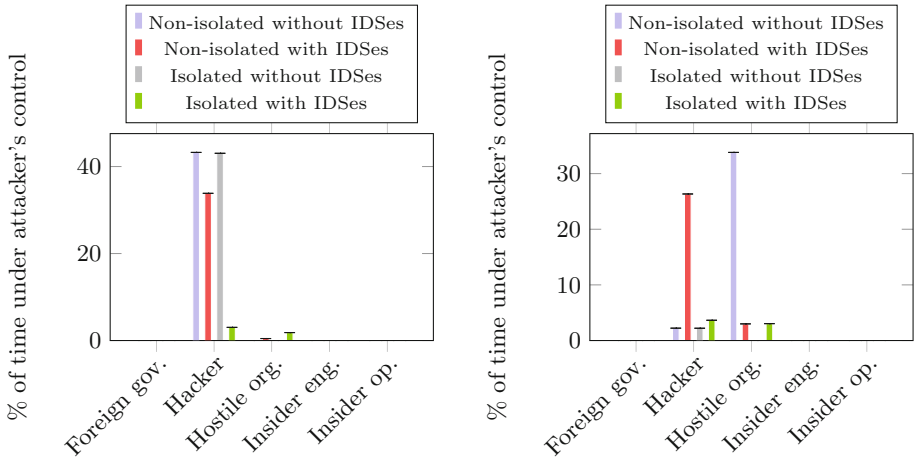
(a) Supervisory LAN 1. (b) Supervisory LAN 2.

Fig. 5. Average percentages of time in which the attacker has control of an on-site device on a particular supervisory network.

with nearly equal interest in other scenarios. The insider engineer saw a large payoff in controlling the system through the devices, but because of low skill proficiency and cost considerations, he did not see it as the most attractive goal. The insider operator, on the other hand, has high attack proficiency for nearly everything (other than firewall attacks) and also had direct access to the facilities housing the devices, so he was able to gain control of the system without running into the obstacles faced by all other types of attackers.

IDSes proved to be not very effective in stopping device reprogramming attacks. Moreover, IDSes only helped the attacker stay more focused on goals with large payoffs. This is especially apparent when the hostile organization attacked the non-isolated networks in subriver system 2; the percentage of time in control jumped by 23% when IDSes were added. The lack of IDS effectiveness is also particularly obvious in the case of the insider operator, for which the percentage of time in control jumped by 3% when IDSes were added. This phenomenon results from the moderately high tolerance of detection by the hostile organization and an even higher tolerance of detection by the insider operator (i.e., their detection weights were 0.2 and 0.1, respectively). They were also highly skilled in attacking HMI components (with attack proficiencies of 0.7 or higher), which meant that it took very little time for them to bypass IDS protections in the supervisory LAN to gain further access.

The higher payoffs of attack goals for the upper subriver system did not make much of a difference under any attacker, since, for the most part, the attackers were constantly attacking the system until all possible goals were achieved.



(a) Supervisory LAN 1. (b) Supervisory LAN 2.

Fig. 6. Average percentages of time in which the attacker has control of a GPRS communication router on a particular supervisory network

6.2 Control of Router

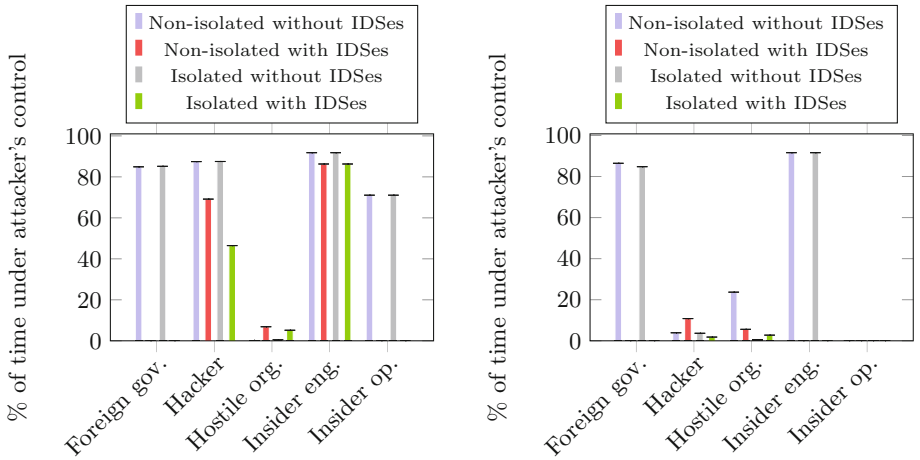
Figure 6 shows the percentages of time that the subsystems were compromised via routers with respect to different attackers. Just as before, the foreign government did not see gaining control of the system through router attacks as the most attractive goal. The insider engineer and insider operator did not see much value in leveraging control of the system through router attacks, either. The hacker saw more value in performing attacks through the router corresponding to HMI 1, but did not see the same value in performing attacks through the other router (corresponding to HMI 2), although he found such attacks on the HMI 2 router to be more attractive (than the do-nothing step) in a non-isolated system with IDSes present. The hostile organization did not see much value in gaining control of the system through router attacks, as it is more interested in device reprogramming attacks, although there were several exceptions. First, the presence of IDSes made the attacker consider other types of attacks, such as device reprogramming attacks. Second, in the case of non-isolated, unprotected networks, the hostile organization saw as much value in router attacks as in device reprogramming attacks.

Overall, IDSes helped minimize router attacks in subriver system 1, but they did not perform as well in minimizing router attacks in subriver system 2. Specifically, for the hacker attacking subriver system 1, the presence of IDSes reduced the percentage of time in control by 10% in the non-isolated case and by 40% in the isolated case, which meant that isolation provided an extra layer of security in this scenario. On the other hand, when the same hacker attacked subriver system 2, the presence of IDSes increased the percentage of time in control by 24% in the non-isolated case and by less than 2% in the isolated case. Despite

the increase in percentages, isolation helped provide an extra layer of security with a smaller increase in control of the system through router attacks.

6.3 Control of System via Backdoor Infection

Figure 7 shows the percentages of time that the subsystems were compromised via installation of HMI backdoors with respect to different attackers. Interestingly, every single type of attacker succeeded in installing backdoor software on the HMI in at least two different network configurations. However, this finding is unsurprising, because the goal of installing backdoor software is completely independent of the goals of controlling the system through devices that were directly altered by reprogramming or indirectly altered by router attacks, which meant the attackers had more leverage in installing backdoors on the HMIs.



(a) Supervisory LAN 1. (b) Supervisory LAN 2.

Fig. 7. Average percentages of time that an HMI on a particular network has backdoor software installed.

The foreign government was interested in installing backdoors on both HMIs only when IDSes were not present. The hacker was more inclined to install backdoor software on the HMI in subriver system 1 over the HMI in subriver system 2. The hostile organization was also more inclined to install backdoor software on the HMI in subriver system 1, but only in cases where IDSes were present. Moreover, the hostile organization found that installing backdoor software on the HMI in subriver system 2 was one of the most attractive goals for an unprotected, non-isolated network configuration. Despite low proficiency in attack skills, the insider engineer was most successful in installing backdoor software on the HMIs, as he found it to be the least risky and most attractive goal. However, he was

not interested in installing backdoor software on the HMI corresponding to subriver system 2 when IDSes were present. The insider operator was interested in installing backdoor software only on the HMI in subriver system 1 and only when IDSes were not present.

Overall, IDSes helped minimize backdoor software installations. They helped reduce system compromise via backdoor installations by anywhere from 2% to 40%; in some cases, they helped stop backdoor installations by the foreign government and insiders. However, there were a few exceptions. For example, a 7% increase was seen when IDSes were added in the case of backdoor software installation on the HMI in subriver system 2 by the hacker, and except when there were non-isolated HMIs, the hostile organization compromised the system through backdoor software installation for a slightly longer period of time when IDSes were present.

7 Conclusion

It was shown that network isolation and an IDS presence play a major role in the security of a system. We used ADVISE to study attacks on a river zonal dispatcher by investigating the effects of isolation and the presence of IDSes. This work is important because such systems send data to a national dispatcher that informs people of any danger, and the smallest difference in protection could have life or death consequences. If the system fails to function normally, warnings may cease, putting lives and property in serious peril. In many cases, IDSes help reduce the amount of time that a system remains in a compromised state, and isolation makes it more time-consuming for the attacker to explore attack paths. Mitigation of attacks through IDSes and isolation helps ensure that the national dispatcher can help save lives following catastrophes such as basin flooding.

The results indicated that attackers with certain abilities and focused goals are the most dangerous ones. For example, in the case of the insider operator, he not only had unrestricted access to the system, but also had a specific goal of attacking the devices. As a result, he was very successful in compromising them. Further, the results suggest that security practitioners must account for as many types of adversaries as possible, since different types have different mindsets and different target goals. The best approach for security practitioners is to be proactive and keep their systems up to date.

Acknowledgments. The work described here was performed, in part, with funding from the Department of Homeland Security under contract HSHQDC-13-C-B0014, "Practical Metrics for Enterprise Security Engineering." The authors would also like to thank Jenny Applequist for her editorial efforts.

References

1. Gao, W., Morris, T., Reaves, B., Richey, D.: On SCADA control system command and response injection and intrusion detection. In: Proceedings of the 2010 eCrime Researchers Summit (eCrime), pp. 1–9, October 2010
2. LeMay, E., Ford, M., Keefe, K., Sanders, W., Muehrcke, C.: Model-based security metrics using ADversary VIew Security Evaluation (ADVISE). In: Proceedings of the 2011 Eighth International Conference on Quantitative Evaluation of Systems (QEST), pp. 191–200, September 2011
3. LeMay, E.: Adversary-driven state-based system security evaluation. Ph.D. thesis, University of Illinois at Urbana-Champaign, Urbana, IL (2011). http://www.perform.illinois.edu/Papers/USAN_papers/11LEM02.pdf
4. Meyer, J.F., Movaghar, A., Sanders, W.H.: Stochastic activity networks: structure, behavior, and application. In: Proceedings of the International Conference on Timed Petri Nets, Torino, Italy, pp. 106–115, July 1985
5. Modbus: Modbus application protocol specification v1.1b3, April 2012. http://www.modbus.org/docs/Modbus_Application_Protocol_V1_1b3.pdf
6. Morris, T.H., Gao, W.: Industrial control system cyber attacks. In: Proceedings of the 1st International Symposium on ICS & SCADA Cyber Security Research 2013, ICS-CSR 2013, pp. 22–29. BCS, UK (2013)
7. Stoian, I., Ignat, S., Capatina, D., Ghiran, O.: Security and intrusion detection on critical SCADA systems for water management. In: Proceedings of the 2014 IEEE International Conference on Automation, Quality and Testing, Robotics, pp. 1–6, May 2014
8. Tenable Network Security Inc.: Modicon Modbus/TCP programming function code access (2016). https://www.tenable.com/plugins/index.php?view=single&id=23819
9. U.S. Department of Homeland Security: Dams sector-specific plan: an annex to the national infrastructure protection plan (2010). http://www.dhs.gov/xlibrary/assets/nipp-ssp-dams-2010.pdf
10. U.S. Department of Homeland Security: Dams Sector (2015). http://www.dhs.gov/dams-sector
11. U.S. Department of Homeland Security: National infrastructure protection plan: dams sector, August 2015. https://www.dhs.gov/xlibrary/assets/nipp_snapshot_dams.pdf
12. U.S. Environmental Protection Agency: Cyber security 101 for water utilities, July 2012. https://nepis.epa.gov/Exe/ZyPURL.cgi?Dockey=P100KL4T.TXT
13. 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, pp. 380–388 (2011)

Reliable Key Distribution in Smart Micro-Grids

Heinrich Strauss[1(✉)], Anne V. D. M. Kayem[1], and Stephen D. Wolthusen[2]

[1] Department of Computer Science, University of Cape Town,
University Avenue North, Rondebosch, South Africa
{hstrauss,akayem}@cs.uct.ac.za
[2] NISLab, Faculty of Computer Science and Media Technology,
NTNU – Norwegian University of Science and Technology, Gjøvik, Norway
stephen.wolthusen@ntnu.no

Abstract. Authentication in smart micro-grids facilitates circumventing power theft attacks. For economic reasons, we consider a smart micro-grid model in which several users (households) share smart meters. The caveat is that users belonging to neighbouring households can provoke misattribution attacks, resulting in unfair billing. Unfair billing can lead to user distrust in the reliability and dependability of the micro-grid and thus an unwillingness to participate in the scheme. The is undesirable since grid stability is impacted negatively by user withdrawals. In this paper, we make two contributions. First, we propose an attack model for power theft by misattribution. Second, we propose a key management scheme to circumvent such an attack. We show that the proposed scheme is secure against impersonation attacks and performance efficient.

Keywords: Key distribution · Smart micro-grid · Attack model

1 Introduction

Smart grids facilitate power consumption monitoring by matching generation to demand [1]. Deploying smart grids in rural and remote regions is a cost-intensive procedure and so it makes sense to use a distributed power generation solution [2].

However, SMG architectures that rely on distributed energy sources cannot rely on generator inertia to compensate for measurement and state estimation errors. As such, SMG reliability is tightly intertwined with stability. Consequently, billing errors provoked by mis-reporting to facilitate energy theft, can result in user withdrawals from grid participation. Since participation is critical to grid stability, withdrawals can result in break down of the grid, which is undesirable.

It therefore becomes important to devise a method of protecting grid communications, namely consumption reports and billing data, by ensuring non-repudiability. In this paper we make two contributions. First, we propose an attack model aimed at provoking energy theft. Second, we propose a key management scheme to detect and circumvent the attack.

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 265–270, 2017.
https://doi.org/10.1007/978-3-319-71368-7_22

The rest of the paper is structured as follows: In Sect. 2, we present related work on SMG key-management schemes in general, along with our system model, scheme and an analysis sketch. We summarise the main points in Sect. 6.

2 Related Work

The microgrid architecture is shown in Figs. 1 and 2. Each property is controlled by a non-empty set of higher-powered managers, with numerous sensor nodes distributed inside the property. The Eschenauer-Gligor (EG) algorithm [6] for key-distribution pioneered word on authentication in distributed sensor networks. This provides a method of probabilistically establishing a secure key sharing protocol, using symmetric encryption together with randomly predistributed shared-keys. Various improvements on the scheme have been developed [4,5,10], improving on resilience against attackers with a partial key-ring, optimizing key-selection on constrained devices, improving key-deployment targeting and resilience against message injection. However, these schemes do not provide a method of establishing non-repudiation which is necessary in a distributed system such as ours. Liu et al. [9] propose alleviating this problem with the Multi-Level (ML) μTESLA scheme which is an authenticated key distribution framework for sensor networks with roughly calibrated time. We make use of vector clocks [7] to provide this without a central time source.

3 Attack Model

Our system model can be represented by a graph of vertices representing nodes and edges representing transitive secure communication channels (A and B are securely connected if and only if a messaging path exists in which the message is never unencrypted). In the AN graph, the managers are the edge nodes (since the AN cannot directly communicate with sensors). From the sensors' perspective, the entire PKC-secured network is reduced to a unique node, the AN.

A connected subgraph containing a manager (but excluding the AN) is called a *location*, and defines the area in which shared-keys are used. If the adversary is able to insert a node inside a location, they may generate false reports within that location, resulting in a *loss of integrity*. If the adversary is able to replace a vertex with one which it controls, an *impersonation* attack results.

An adversary may inject spurious usage reports to a targeted manager using the same SSM, attributing their own usage to the target (*impersonation*). Since the management network has no visibility beyond the SSM, the aggregation network would not be aware of any problems. This is not viable if the attacker is on a different SSM, though, since the distribution network would show a discrepancy between consumed and billed usage, and so the attack requires physical proximity to the target. This can further affect the local *availability* of energy to the target, if the AN imposes sanctions. Since the attacker would not have learned any message contents, *confidentiality* remains intact.

4 Key Management Scheme

In this section we present our key management (KM) scheme that is aimed at supporting the authentication mechanism to ensure non-repudiation. Our KM extends the EG scheme [6], and works by assigning all participating sensors and nodes in the SMG a key-ring drawn from a disjoint partition on a global key ring into equal-sized location-based pools. The security of the key-ring lies in the intractability of predicting pseudo-random functions [3]. We present our SMG architecture briefly and then proceed to describe the key generation, distribution and update procedures.

4.1 Smart Micro-Grid Architecture

Our conceptual MG architecture is structured as a group of three sub-networks: the *power*, *control*, and *communication networks*. Power is delivered to MG users from distributed generation sources over the *power network*. Users with private generators can supplement the grid. The *control network* includes the technology and algorithms used for the metering, monitoring, and management of the system. The structure is show in Fig. 1.

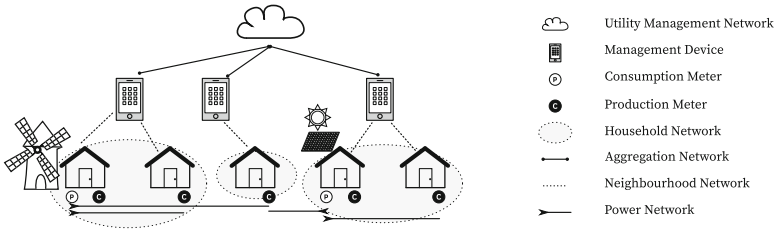


Fig. 1. *Overview of networks:* Managers are shared across sensors and properties

Metering devices consist of low cost, untrustworthy sensors placed into electrical home appliances to measure the consumption. Usage reports are transmitted to a local mobile device used as a grouping station ("manager"). The manager aggregates household consumption data and communicates this data to a shared smart-meter (SSM). Each manager can therefore linked to a household owner, who is liable for the usage cost. The SSM collects power consumption values from the households in the cluster over which it has control and finally transmits this to the MG control centre where billing and demand management algorithms are applied to regulate the grid operation. The physical layout is shown in Fig. 1. The *communication network* is a three-tiered hierarchical architecture structured as follows: The *household network* consists of all electrical appliances that communicate with the managers via a WSN using communication technologies such as Bluetooth or ZigBee. The *neighbourhood network* consists of clusters of houses linked to an SSM. SSMs are connected via a mesh network and are capable

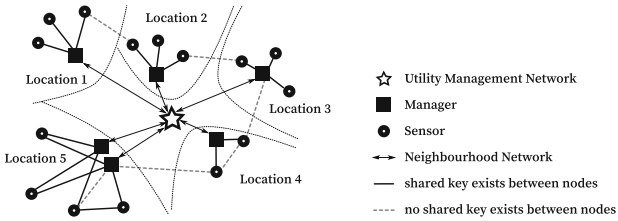


Fig. 2. *Network Topology:* unconnected nodes do not have keys in common

of secure, non-repudiable communication through PKC. Finally an *aggregation network* at the MG level, through which all the SSMs transmit data to the MG control centre. The communication technologies include WiMAX and LTE. It is natural to view a household as the perimeter of a particular location.

4.2 Key Management

A global key-pool, G, is established by selecting pseudorandom keys of l-bits from a bitstream, along with a random public key-tag of length t [8]. A λ-sized subset of keys are stored in a set, L_n, termed the *location-ring* associated with the location/household, n. There is a disparate and equally-sized subset L_i for each location. From L_n, each node draws a set of k keys and, along with attributes such as the validity status of a key, are stored in a list termed the *key-ring*. The sizes of G, L_n, and k should be chosen such that sharing keys is sufficiently likely, as described in Du et al. [5].

We term a key-ring "useful" if it has sufficient keys shared with existing nodes in a particular location to establish a secure connection. We track the number of key-tag issuances at the Key-Distribution Center (KDC). A special case of distribution occurs when all nodes in a location share the same key-ring, which may occur when $k \to \lambda$. In this case, a single-node compromise becomes devastating, however, it implies that a key-ring can always be distributed to a node given a location. New network nodes are provisioned offline, ensuring that keys specific to the location of use are available to that node.

Before any sensor enters the network, it obtains a "useful" key-ring from a KDC, located in a secure facility in the AN, where external communication is prevented. A manager for that location must also first be provisioned, since sensors cannot communicate directly with the AN. If insufficient keys match the recorded key-tags for a location at the KDC, a new random key-ring will be selected until it is "useful". The KDC ensures during issuance that the node is a manager or that one already exists for the intended location. A sensor will be able to establish a secure channel with that manager. Subsequent sensors either share keys with the manager or sensor (by the choice of the key-ring), so that messages can always be relayed to a manager. If the new node is a manager, either there are shared keys with some sensors able to relay messages to the partner manager or it shares keys with the partner manager. Thus, each

node can obtain a key-ring by specifying a location with an existing manager. Remediation of a degenerate manager is handled by the KDC, since sensors' key-tags in the location are known to the KDC.

Node authentication is guaranteed by the existence of useful key-rings and through the use of an authenticated broadcast protocol, such as ML-μTESLA [9]. Since the sensor nodes' hardware is not trustworthy, the use of vector clocks [7] can provide time-intervals suitable for ML-μTESLA. This adds latency to the creation of a secure channel, since a High-Level (HL) interval needs to rollover before a node is assured of the manager's identity. As long as the HL intervals do not exceed the length for which sensors can store usage reports, those reports can be stored locally and delivered to a manager once a secure channel is established. Since the HL intervals can be tuned per-location to coincide with periods of low reporting requirements, there need not be a loss of collected usage reports. A sensor entering the location waits for the ML-μTESLA broadcasts announcing a shared key from the set of managers, and the subsequent rollover, after which it can be assured of the manager's identity. It sends the manager its node ID and a random partial key-tag list, to which the manager responds with a challenge containing the encrypted under any matching key-tag's key. A valid response by the sensor proves possession of a valid key. If the request is from an attacker controlled sensor, would not have an expected sensor node ID for the solicitor. Key revocation is handled locally on the node, by marking the key compromised on the key-ring. Upon secure channel establishment, the manager for a location can relay messages to and from the AN. If a sensor requests a new key-ring, it can be passed to the AN and a new ring can be sent encrypted under a symmetric key that the manager does not possess (if such a key exists). In the case that no such key exists, there are relatively few (or no) keys in L_n which the manager does not possess, implying that there will be considerable overlap with the current key-ring. In networks with such a choice of k and λ, the entire location would need to be re-keyed, preferably, with a smaller value of k.

5 Complexity Analysis

The selection and distribution of the node's key-ring is inefficient. The KDC draws k random keys from a location pool ($\mathcal{O}(k)$), then compares each key-tag to the list of key-tags issued to the n nodes in the location ($\mathcal{O}(k \times n)$) before the number of shared keys is available. This results in an algorithmic complexity of $\mathcal{O}(n \times k^2) \approx \mathcal{O}(n^3)$. But, since key-ring selection is only done infrequently (upon introduction to the network and upon key-ring or location-pool exhaustion), the impact on general operation of the network is minimised.

If a sensor wishes to re-key remotely, the additional overhead of transferring the new key-ring includes asymmetric (E, D) and symmetric encryption costs of an ephemeral key to the manager under PKC and further symmetric encryption under the key known to the manager and sensor. For an k-key ring using a message size of μ and (key+tag)-length of ℓ, the overhead in number of messages would be $n = \frac{k \times (\ell)}{\mu} + C$ additional messages (key-ring size divided by network

message size), since the symmetric encryption does not pad the key-ring and the small number of additional messages required to transfer the symmetric key-tag to the manager under PKC (C) is negligible. This results in a complexity at the AN and manager of $\mathcal{O}(1)$ for asymmetric operations (E, D) and $\mathcal{O}(n)$ symmetric operations for delivery of the key-ring from KDC to manager. The cost between manager and sensor is an additional $\mathcal{O}(n)$ symmetric operations at each device.

6 Conclusions

We have shown that an attack where an adversary wishes to impersonate a target in an SMG can be prevented by our Key Management Scheme. By ensuring that managers expect node additions and that sensors validate the identity of the manager, the risk associated with these attacks is mitigated. By securely authenticating the initial channel establishment between sensor and manager a level of non-repudiation comparable to native PKC is achievable, even for the computationally-constrained sensors.

References

1. Albadi, M.H., El-Saadany, E.: Demand response in electricity markets: an overview. In: IEEE Power Engineering Society General Meeting, vol. 2007, pp. 1–5 (2007)
2. Ambassa, P.L., Kayem, A.V.D.M., Wolthusen, S.D., Meinel, C.: Secure and reliable power consumption monitoring in untrustworthy micro-grids. In: Doss, R., Piramuthu, S., Zhou, W. (eds.) FNSS 2015. CCIS, vol. 523, pp. 166–180. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19210-9_12
3. Canetti, R., Chen, Y., Reyzin, L.: On the correlation intractability of obfuscated pseudorandom functions. In: Kushilevitz, E., Malkin, T. (eds.) TCC 2016. LNCS, vol. 9562, pp. 389–415. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49096-9_17
4. Chan, H., Perrig, A., Song, D.: Random key predistribution schemes for sensor networks. In: Proceedings of the 2003 Symposium on Security and Privacy, pp. 197–213. IEEE (2003)
5. Du, W., Deng, J., Han, Y.S., Varshney, P.K., Katz, J., Khalili, A.: A pairwise key predistribution scheme for wireless sensor networks. ACM Trans. Inf. Syst. Secur. **8**(2), 228–258 (2005)
6. Eschenauer, L., Gligor, V.D.: A key-management scheme for distributed sensor networks. In: Proceedings of the 9th ACM Conference on Computer and Communications Security, pp. 41–47. ACM (2002)
7. Fidge, C.J.: Timestamps in message-passing systems that preserve the partial ordering. Department of Computer Science, Australian National University (1987)
8. Kayem, A., Strauss, H., Wolthusen, S.D., Meinel, C.: Key Management for secure demand data communication in constrained micro-grids. In: Proceedings of the 30th International Conference on Advanced Information Networking and Applications Workshops. IEEE (2016)
9. Liu, D., Ning, P.: Multilevel μTESLA: broadcast authentication for distributed sensor networks. ACM Trans. Embed. Comput. Syst. (TECS) **3**(4), 800–836 (2004)
10. Liu, D., Ning, P., Li, R.: Establishing pairwise keys in distributed sensor networks. ACM Trans. Inf. Syst. Secur. (TISSEC) **8**(1), 41–77 (2005)

Security Validation for Data Diode with Reverse Channel

Jeong-Han Yun(✉), Yeop Chang(✉), Kyoung-Ho Kim(✉), and Woonyon Kim(✉)

National Security Research Institute, Jeonmin-dong, Yuseong-gu, Daejeon, Korea
{dolgam,ranivris,lovekgh,wnkim}@nsr.re.kr

Abstract. Hardware-based data diode is a powerful security method that removes the reverse channel for network intrusion. However, simple removal leads to data unreliability and user inconvenience. A reverse channel is forbidden if it affects physical unidirectionality without an exact security analysis. If a reverse channel is used restrictively and its security is validated, the data diode can be a secure solution. Thus, we propose security criteria based on an application environment for a data diode that was implemented with a reverse channel and validate the data diode's security by unit/integration/system testing based on our security criteria.

Keywords: Unidirectional network · Data diode · Security
Control system · Unidirectional gateway · One-way data transfer

1 Introduction

Traditionally, SCADA systems are physically disconnected from outside networks to block external threat. However, the recent trend is integrating SCADA systems into IT networks. The operation data of SCADA systems can be used for financial purposes such as production optimization; thus security solutions are needed for data transmission from SCADA systems to IT networks. These solutions can have different security levels depending on application characteristics and its environment.

A unidirectional data transfer system is a network device that enables outgoing data flow, but it restricts incoming data flow for removing incoming data lines. Its function seems similar to that of a firewall; however, a significant difference exists between two devices. Specifically, the firewall enables bidirectional communication if it satisfies the access control list (ACL), whereas a unidirectional data transfer system enables only a unidirectional data transfer although it satisfies ACL. The unidirectional data transfer system is thus referred to as a data diode or unidirectional security gateway owing to this characteristic. We use the term 'data diode' to refer to any type of physical unidirectional network devices in this paper.

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 271–282, 2017.
https://doi.org/10.1007/978-3-319-71368-7_23

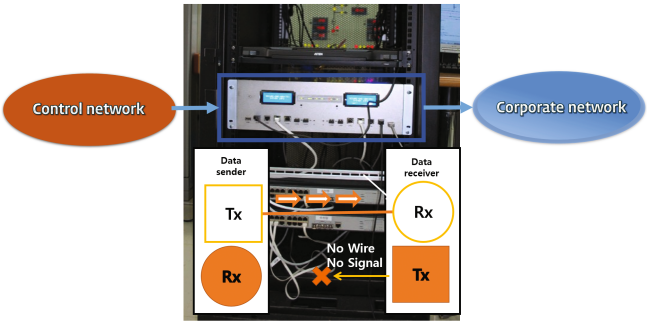


Fig. 1. Field-test of our data diode in a power grid system

When a firewall has a security vulnerability – such as an ACL misconfiguration, vulnerabilities of permitted services, or authentication bypassing – an adversary can penetrate the network through the firewall. However, in the case of a data diode, the attacker cannot enter the protected area through the network line because it has no route of entrance. Therefore, many SCADA security guidelines (NIST special publication 800-82, NRC regulatory guide 5-71, IEEE standard criteria for digital computers in safety systems of nuclear power generating stations (7-4.3.2), NEI 08-09 cyber security plan for nuclear power reactors, and ANSSI cybersecurity for industrial control systems[1]) recommend utilities to use data diodes for protection of SCADA systems.

In a SCADA system having critical infrastructures, some utilities use their unique or customized protocol for data transmission. To support such protocols, we developed data diode hardware and proxy applications to send information from a SCADA network to a corporate network in a power grid system.

When we installed and tested our data diode in real power grid systems, as shown in Fig. 1, operators wanted to manage proxy applications and confirm data transmission to a destination server in the corporate network. However, because our data diode removed the physical path, operators could not check the current status of the proxy application and the destination server. Thus data reliability cannot be logically achieved despite our data diode experimentally supporting a 100% success rate of data transmission between unidirectional data paths.

The hardware-based data diode is a powerful security method that removes the reverse physical path. In case of Fig. 1, the data diode protects the control network against all attacks from the corporate network. However, simple removal cannot ensure data reliability and the removal forbids checking the status of data receivers. However, if a reverse channel can be securely used, the data diode with a reverse channel can solve these issues. To the best of our knowledge, no systematic approach exists for security validation and verification of data diode, which has a restricted reverse channel. We thus implemented a data diode with a

[1] http://www.ssi.gouv.fr/uploads/2014/01/industrial_security_WG_detailed_measures.pdf.

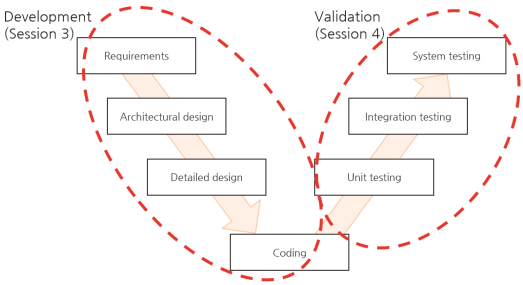


Fig. 2. TOS Implementation and validation following V-model [1]

reverse channel, and we validated its security by unit/integration/system testing based on our security criteria shown in Fig. 2. The main contribution of our paper is the proposal of security criteria based on an application environment for a data diode with a reverse channel.

The remainder of this paper is organized as follows. Section 2 describes general data diode implementations and techniques for reliable unidirectional data transfers. Section 3 describes the design and implementation results of our traffic one-way system (TOS) prototype product. Systematic TOS security test results are presented in Sect. 4. Next, we describe a TOS field experiment conducted for a water treatment control system and suggest suitable applications of TOS in Sect. 5. Finally, Sect. 6 presents our conclusions.

2 Background: Data Transfer Reliability of Data Diodes

Automatic repeat request (ARQ) is a renowned strategy for guaranteeing the reliability of communication. However, ARQ cannot be selected for data diode because senders never know whether receivers successfully obtain the data. It is difficult for data diode to confirm whether receive (RX) nodes soundly receive every network packet.

Under normal conditions, the bit error rate (BER) is under 10^{-12}. Although this value represents a low probability, it is nonetheless difficult to ignore during long time. Furthermore, the possibility of packet loss caused by system performance cannot be eliminated. In the remainder of this section, we examine the two main techniques that are used to increase data transfer reliability and that can be used as products.

Forward Error Correction (FEC). FEC is considered an alternative to ARQ. Additional duplication codes are added to the original data for error recovery. This is useful when the retransmission cost is high or inadequate. In our study, we determined that packet loss frequently occurs in a unidirectional data path, whereas bit errors do not. Packet loss can be considered a burst error. Interleaving [2] is a good solution for packet loss recovery. However, FEC decoding

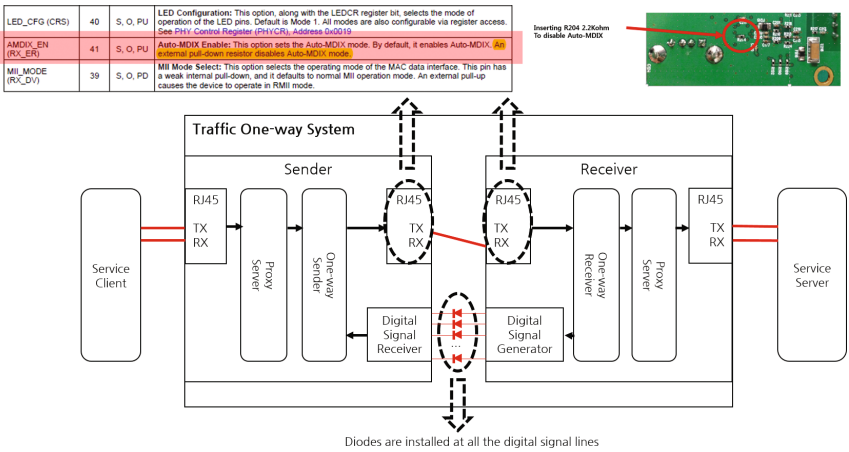


Fig. 3. TOS architecture

for error recovery requires considerable computing power. To develop data diode with FEC over 1 Gbps, dedicated encoding/decoding chips are required.

Restricted reverse signals. Some vendors developed restricted reverse signaling to check whether the RX node successfully received data. To avoid breaching the unidirectionality of data diodes, these vendors devised restricted reverse signaling. Hitachi and some other companies suggested a structure that has a special electrical signal line in addition to a unidirectional data path [3]. NNSP, one of the data diode vendors, produced an altered data diode with a special structure to check the receiver status without an explicit physical reverse channel using an electrical switch between transmit+ (TX+) and transmit- (TX−) lines [4] in the RX node opens when a predefined error occurs. Then, the transmit (TX) node detects disconnection of network line and waits until the connection returns to normal. After the connection is restored, the TX node again transmits from the moment of the disconnection.

3 Traffic One-Way System: Physical Unidirectional Data Transfer System with Reverse Channel

To execute real testing for security validation of reverse channels in data diodes, we implemented a TOS(Traffic One-way System). TOS is a physical unidirectional data transfer system with reverse channels for an acknowledgement mechanism. In this section, we introduce the internal structure of TOS and outline the experiment conducted to assess its performance.

3.1 System Overview

TOS consists of two separate embedded systems for applying the physical unidirectional data transfer technology. Its structure is shown in Fig. 3. TOS basically has the same architecture as a typical data diode using RJ45 Ethernet interface. We disconnected the sender's RX cables and receiver's transmit (TX) cables for physical unidirectional data transfer.

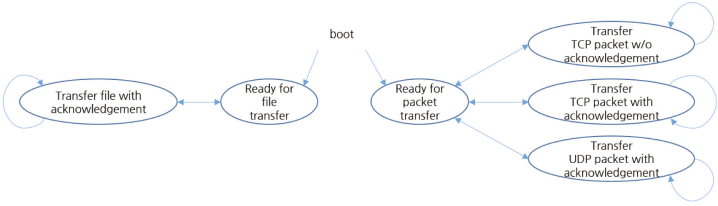
However, the disconnection was not sufficient for achieving real physical unidirectional flow on account of the Auto MDI-X[2] feature of the Ethernet chip. Auto MDI-X automatically detects the required cable connection type and eliminates the requirement for crossover cables to interconnect switches or for connecting PCs peer-to-peer; however, an attacker may change the direction of the data channel by exchanging the TX cable for the RX cable using this feature. Thus, we disabled the Auto MDI-X feature on each physical circuit to fundamentally remove the threat.

For the reverse channel, we connected digital signal generators between the sender and receiver. To protect bidirectional communication through a reverse channel, we installed 'real' diodes on each digital signal line. It is impossible to change the direction of communication through the reverse channel because of the diodes.

3.2 Prototype Implementation

We developed a TOS prototype targeting a 100 Mbps communication environment. Figure 4a shows the appearance of the system.

(a) Appearance



(b) State diagram of the TOS sender

Fig. 4. TOS

Implementation. The sender and receiver were developed as an embedded board with the same specifications. The board provides 256 MB of flash memory and a MicroSD card slot for storage expansion based on the ARM Coretex-A8 720 MHz and RAM 256 MB. In addition, the board provides two RJ45 100 Mbps Ethernet interfaces and an RS232 interface. On each board one RJ45 100 Mbps Ethernet interface is used for the unidirectional data channel between the

[2] https://en.wikipedia.org/wiki/Medium-dependent_interface.

sender and receiver; the other interface is used for the communication channel for an external communication partner. The RS232 interface is used for the management interface of each board.

The operating system is embedded Linux 3.2.0. TOS provides TCP/UDP traffic forwarding service [5] and file transfer service. The total LOC of the application is approximately 8,500 lines in C++. Figure 4b shows the execution states of the code following the specification and code review.

Table 1. Functions of digital signal lines for a reverse channel of TOS

Pin no.	Description (System code)	Pin no.	Description (Error code)
1	Rx node power-on/off	5	Storage is available
2	Rx node network status	6	Storage is almost full
3	Feedback	7	Same file exists already
4	N packet acknowledgment	8	Reserved

Acknowledgement mechanism using digital signal lines. One digital signal line was sufficient for only an simple acknowledgement mechanism. First, the TOS user sets N. The receiver sends acknowledgement signal to the sender when every N packets are normally received. If the sender does not receive the acknowledgement signal, the sender retransmits the N packets. However, we connected eight digital signal lines between the sender and receiver for convenient management of TOS, as shown in Table 1.

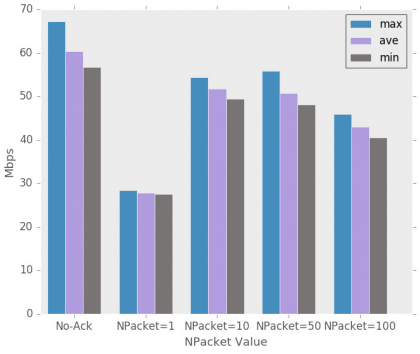


Fig. 5. Performance test results of TOS

Performance. We performed experiments to measure the data transfer performance of TOS in which the acknowledgement was applied. The experiments were

repeated three times for each of the transmissions 10 MB and 100 MB under the same conditions and the values were set to N. Figure 5 shows the experimental results.

According to the results, the transmission rate is approximately 40–50 Mbps when N is 10–50. Data reliability is always achieved when the acknowledgement mechanism is applied. When the acknowledgement mechanism is not applied (No-Ack at Fig. 5), transmission performance slightly increases(up to 60 Mbps) but data loss occurs inevitably. The data loss may be overcome by system tuning, such as in the receiving buffer performance; nevertheless, it can not be a fundamental solution to the data loss.

4 Security Testing of TOS

We then confirmed that TOS does not allow reverse data transmission, although TOS was attacked at the external network connected to the TOS receiver. Our validation process followed the V-model, as shown in Fig. 2, with a unit test, integration test, and system test. To ensure due diligence in the objectivity of the verification tests, all processes were carried out by external specialists[3] to test the equipment of the control system and infrastructure.

4.1 Attack Assumption

Our assumptions about attackers are outlined below.

- Assumption 1: The attacker cannot physically access TOS.
- Assumption 2: The attacker can access to the TOS receiver over the network.
- Assumption 3: The attacker can take control of the TOS receiver.

TOS was installed in the internal network of the control system and infrastructures to block intrusion form the external network. If the attacker can physically access TOS, the attacker can alter all aspects of TOS. Thus, we assumed that the attacker cannot have direct physical access, and we only focused on attacks against TOS from an external network. Aside from the physical access, we considered all of possibilities of attacks against TOS as Assumptions 2 and 3.

4.2 Validation Requirements

TOS is a physical one-way system for applying the acknowledgement mechanism using a reverse channel. Each channel ensures uni-directionality. Specifically, a disabled Auto MDI-X forbids reverse data transmission through the data channel, and diodes-installed on the reverse channel cannot send forward signals from the sender to the receiver. The attacker cannot send data from the receiver to the sender through the TOS data channel.

[3] TestMidas co., Ltd. (http://www.testmidas.com).

To validate the safety of TOS, we have to check the possibility of attack through the reverse channel, which is the only path that could enable an attacker to send data or command in the reverse direction. Thus, requirements for validating the security statement in this paper are organized as follows.

- Requirement 1 (data): When the sender receives information from the TOS reverse channel, the sender does not store the information in the files.
- Requirement 2 (command): Although any information is sent through the TOS reverse channel, the sender performs only the functions defined in Sect. 3.2.

Because we assume that the attacker can assume control of the receiver as Assumption 3, it is necessary to ensure the security of TOS in a situation in which an attacker can arbitrarily modify the receiver. Thus, all verifications performed for TOS consisted of an original sender and an arbitrary receiver that is transformed by the attack.

4.3 Unit Testing

Unit testing is a software testing method by which various parts–individual units (or functions) of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures–are tested to determine whether they are fit for use[4]. For unit testing, we performed dynamic tests for all functions in the source code of the sender to validate that each function of the sender satisfies Requirements 1 and 2.

We used a unit testing tool, $CodeScroll^{TM}$ $Controller\ Tester$[5]. Based on statements and branch coverage, we generated 1,060 test cases for 141 sender functions. No test results violated Requirement 1 or 2. The generated test cases covered 98.90% of statements and 97.19% of branches. The test cases did not cover all statements and branches because dead codes remained to be generated during development.

4.4 Integration Testing

Integration testing (also called integration and testing) is the phase in software testing in which individual software modules are combined and tested as a group. It employs for input the modules that were unit-tested, groups them in larger aggregates, applies tests defined in an integration testing plan to those aggregates, and delivers as its output the integrated system that is prepared for system testing[6]. To proceed with integration testing based on Requirements 1 and 2, we generated and executed 314 test cases using $CodeScroll^{TM}$ $Controller\ Tester$. No test results violated Requirements 1 and 2. The generated test cases covered 96.70% of all function calls. The test cases did not cover all function calls because some unnecessary functions comprise impossible exception cases.

[4] https://en.wikipedia.org/wiki/Unit_testing.
[5] http://www.suresofttech.com/products/controller-tester/.
[6] https://en.wikipedia.org/wiki/Integration_testing.

4.5 System Test

System testing of software or hardware is conducted on a complete, integrated system to evaluate the system's compliance with specified requirements[7]. To perform system testing using TOS, we built a test environment, as shown in Fig. 6.

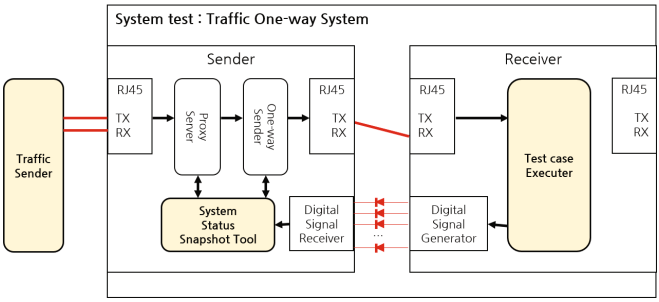


Fig. 6. System testing environment

Following Assumption 3, the receiver can be attacked. For the system testing, the test case executor simulated all actions of an attacked receiver. A system status snapshot tool recorded the states of the processes and the file system of the TOS sender. Using the records, we checked the difference between states before and after testing on the sender. To track the flow of information carried in the digital signal of the sender, we checked the source code processing in the line when a signal occurred at all locations (14 locations). To this end, we modified source code to record the flow. The current I-node tree, transformed I-note list, and the current disk capacity involved to record the snapshot to check for changes in the physical disk information. We recreated the situation in which the sender traffic continues to flow in order to proceed with the test.

Test case generation. Based on our assumptions about attacks, the reverse channel comprised of eight digital signal lines is the only way for an attack. In the source code, the receiver sends only 0 or 1 through each digital signal line. However, if the receiver is attacked as in Assumption 3, the receiver sends any value; i.e., infinite cases. For efficient tests, we used eight inputs for each input case: 0, 1, negative number, negative large number (for checking overflow), positive number, positive large number (for checking overflow), Unicode, and ASCII code. To cover all possible combination cases of eight digital signal lines, we required 8^8 (16,777,216) test cases. It is practically impossible to test the all test cases. Thus, we selected 95 test cases to use the pairwise technique [6]. Pairwise (a.k.a. all-pairs) testing is an effective test case generation technique

[7] https://en.wikipedia.org/wiki/System_testing.

that is based on the observation that most faults are caused by interactions of at most two factors. Pairwise-generated test suites cover all combinations of two; therefore, they are much smaller than exhaustive ones while being very effective in finding defects.

We reviewed the source code of the sender. The test case execution order of the sender did not affect the current results of the test. Figure 4b represents the state transition of the sender during execution. If we performed all test cases for the six states in Fig. 4b, we could have covered all possible cases of the sender. The total number of test cases was 570 (=95 (number of possible digital signal line inputs) $\times$ 6 (state number)).

Testing result. The testing procedure is outlined below:

1. Setting the status of the sender
2. Saving the snapshot the initial status of the sender
3. Executing a test case
4. Saving a snapshot the final status of the sender
5. Comparing the initial and final snapshots.

We performed a total of 570 tests. No result violated Requirement 1 or 2. During the tests, we identified a software bug of the sender; i.e., the four test cases killed the sender processes. However, this bug cannot cause security problem related to Requirement 1 or 2. For stress testing, we performed random testing of TOS in the test environment over three days. Once again, no result violated Requirement 1 or 2. Thus, we conclude that the reverse channel does not cause security problems related to Requirement 1 or 2.

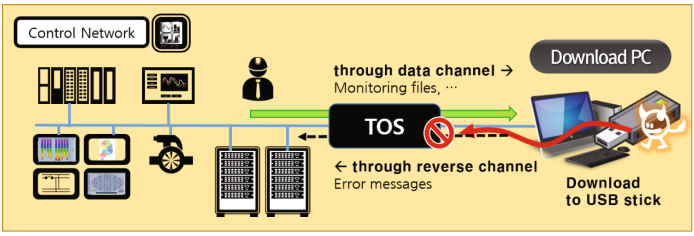
5 Applications

If a reverse channel can be securely used, the data diode with a reverse channel can provide a more secure method for data flow control than software-based security solutions as shown in Fig. 7.

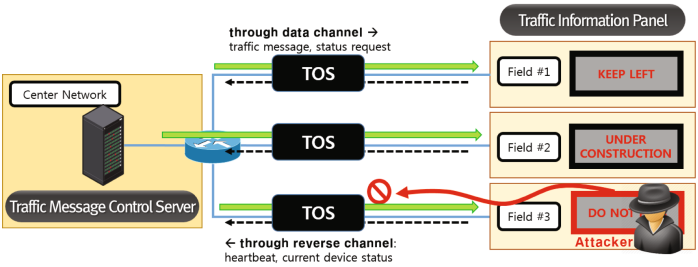
5.1 Field Experiment: Safe Usages of USB Memory Stick

The control system operator should occasionally move monitoring information files of a control system to business network for several reasons. Portable storage (e.g. an USB memory stick) is a convenient utility to move the files. To safely use portable storage, we usually install media control solutions. However, these solutions cannot be installed on some specific devices and servers in the control systems.

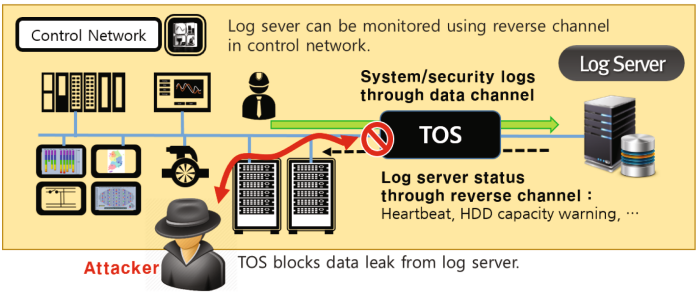
We thus built a system that safely downloads internal files on portable storage using TOS. The entire structure is shown in Fig. 7a. Files are sent to the download PC through TOS. At the PC users can download the files into portable storage. Although the download PC may be compromised, the negative effect cannot be propagated in the control network on account of TOS. We installed this system on two sites in October 2015 and ran it for more than eight months. Field operators gave the system a good rate because it conveniently enhances security.



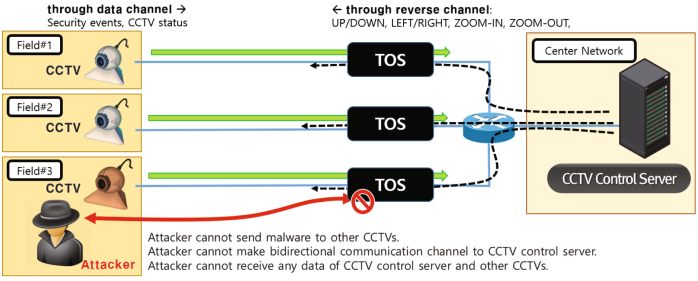
(a) File download into a portable storage through TOS



(b) Variable message sign (VMS) System



(c) Logging system



(d) CCTV monitoring system

Fig. 7. Applications of TOS

5.2 Suggestion

The application shown in Fig. 7b is similar to those in Fig. 7a. A traffic message control server can safely send data to end systems through TOS. TOS directs data flow and blocks all reverse data transmission. If necessary, TOS can allow limited requests and status checking of end systems using the reverse channel. Although one of end system is compromised, it cannot attack the central server and compromise other end systems. In a similar manner TOS can be applied to a patch management system.

TOS can be applied for security enhancement of data receivers, as shown in Fig. 7c and d. TOS blocks data leak from log server in Fig. 7c. Using the reverse channel, a system manager can remotely check the status of a log server. In case of Fig. 7d, CCTV control server can safely gather security information and send limited orders to CCTV devices using the reverse channel. Although an attacker can control one unprotected CCTV device, the attacker cannot connect and steal information from other systems.

6 Conclusion

In this paper, we proposed security criteria based on an application environment for a data diode with a reverse channel. We implemented a data diode with a reverse channel using digital signal lines, and we validated its security by unit/integration/system testing based on our security criteria. Our research can be a starting point for expanding application areas of the data diode for security enhancement.

References

1. Forsberg, K., Mooz, H.: The relationship of system engineering to the project cycle. In: Proceedings of the First Annual Symposium of National Council on System Engineering, pp. 57–65, October 1991
2. Cai, J., Chen, C.: FEC-based video streaming over packet loss networks with pre-interleaving. In: Proceeding of International Conference on Information Technology: Coding and Computing, pp. 10–14 (2001)
3. Namioka, Y., Miyao, T.: Data communication method and information processing apparatus for acknowledging signal reception by using low-layer protocol. Hitachi Ltd., U.S. Patent 20060026292 A1, 2 February 2006
4. Kim, K., Na, E., Kim, I.: Gateway device of physically unidirectional communication capable of re-transmitting data, as single device, and method of transferring data using the same. NNSP, Korea Patent 1015623120000, 15 October 2015
5. Kim, K., Chang, Y., Kim, H., Yun, J., Kim, W.: Reply-type based agent generation of legacy service on one-way data transfer system. KIISC **23**(2), 299–305 (2013)
6. Wallace, D.R., Kuhn, D.R.: Failure modes in medical device software: an analysis of 15 years of recall data. Int. J. Reliab. Qual. Saf. Eng. **8**(4), 351–371 (2001)

Towards a Cybersecurity Game: Operation Digital Chameleon

Andreas Rieb(✉) and Ulrike Lechner

Universität der Bundeswehr München, Werner-Heisenberg-Weg 39, 85577 Neubiberg, Germany
{Andreas.Rieb,Ulrike.Lechner}@unibw.de

Abstract. In the Serious Game "Operation Digital Chameleon" red and blue teams develop attack and defense strategies to explore IT-Security of Critical Infrastructures as part of an IT-Security training. This paper presents the game design and selected results from the evaluation of the gaming experience, an analysis of attack vectors and defense strategies developed in gaming and take outs of game participants. Participants enjoy the experience, develop APTs with realistic complexity and even innovations and take out the need for more information, more awareness training and cross-functional teams in IT-Security.

Keywords: Serious gaming · IT-Security of Critical Infrastructures
Cyberwargaming · IT-Security Awareness

1 Introduction

"Operation Digital Chamaeleon" is a Serious Game designed to train IT-Security professionals in IT-Security Awareness. Its format is an adoption of Wargaming to the domain of IT-Security: Red teams develop attack strategies, blue teams design IT-Security countermeasures and the white team decides who wins.

The game "Operation Digital Chameleon" aims to train participants to deal with Advanced Persistent Threats (APTs) as they are characteristic for the domain of IT-Security of Critical Infrastructures. APTs are cyber operations that are sophisticated, persist over a long period and covered [1]. Symantec describes that "An advanced persistent threat (APT) uses multiple phases to break into a network, avoid detection, and harvest valuable information over the long term."and assumes that "fraud – or worse" are part of APTs [2, 3]. To get the APT covered, attackers modify their attack vectors and update their malicious code frequently [4]. Besides innovations of existing attack vectors, the domain of IT-Security of Critical Infrastructures is concerned with so-called black swans, i.e., high-profile, hard-to-predict, and rare events with extreme impact that are beyond the realm of normal expectations [5]. The infamous Stuxnet is considered such a black swan as both its purpose, namely to destroy production infrastructures as well as its level of sophistication. This level was beyond what was considered to be thinkable in 2009 when it was detected and published.

"Operation Digital Chameleon" is designed to address the training of IT-Security professionals in a playful way. The method Wargaming, which we adopt for "Operation

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 283–295, 2017.
https://doi.org/10.1007/978-3-319-71368-7_24

Digital Chameleon", has a long tradition. In the literature, Wargaming is being dated back to the 19th century, when Baron von Reisswitz used this method for preparing army commanders for taking better decisions – especially in dynamic and unpredictable situations [6]. Geilhardt traces Wargaming back to 1000 BC whereas Geuting suggests that Wargaming was done as early as 600 AC [7, 8]. This method is known to be effective in training as well as in exploring new threats. Wargaming however is a rather new method in the domain of IT-Security. In 2012, ENISA compiled an overview of 85 Cybersecurity exercises [9]. Most of these exercises are table-top exercises "to validate plans and integration of procedures prior to moving on to more complex, team-based activities" [9]. This report refers various goals of IT-Security exercises: To build IT-Security Awareness about cyber threats; identify roles, responsibilities and authorities for responding; test decision-making; assess cyber security emergency readiness; build trust among nation states [9]. In 2015, the report of ENISA [10] extends this compilation to more than 200 exercises. ENISA singles out methods like red teaming, discussion based games, capture the flag, seminars, simulation, and others. Note that term "Wargaming" is neither used in the report of 2012 nor in 2015. According to ENISA [10], just a fraction (11%) of Cybersecurity exercises uses the format of opponent teams. Here we make a contribution with "Operation Digital Chamaeleon" with a playful format of opposing teams. It aims to raise awareness in the domain of IT-Security in which threat innovations occur regularly. With the adoption of the Wargaming method we hope to unlock a creative potential to find attack vector and IT-Security measure innovations.

The paper at hand presents the design of the game "Operation Digital Chameleon", first results of four games and an analysis of these games. This submission extends a previous publication on game design with results of one game and an analysis of the outcomes from one game from an innovation point of view [11].

2 Method

The overall research follows a design oriented research approach as described by Hevner et al. [12]. We use an iterative approach in the design and evaluation of "Operation Digital Chameleon" and the paper at hand describes the state of the design after pretest and first games done in 2015 and 2016.

The main influences in the iterative design process came from the Game Master's observation as well as participants' ideas. E.g. from game #1 the discussion in the phase of debriefing was improved and more time was allocated as it turned out that the discussion was a major learning experience that all participants seemed to enjoy – given the active involvement in the discussion and the ideas that all participants contributed to the discussion. We observed that despite the competitive setting, all participants were motivated in building and understanding good attack vectors and their relevance for an organizations' IT-Security management.

For data generation and data collection, the first author who acted as Game Master on the basis of ten years + experience in IT-Security consulting and awareness training took notes during the game, collected the gaming material (cards, game board with notes), and summarized observations in a structured way. The empirical basis of this

analysis (done in April/May 2016) comprises 4 games with IT-/IT-Security-Experts of Critical Infrastructures within Europe (see Table 1). The Business Sector classification is done according to the Federal Ministry of the Interior Germany [13].

Table 1. Games of "Operation Digital Chameleon"

#	Date	Duration training	Duration game	Participants	Business sector	Country
1	10/2015	3d	6 h	11	Government (Police, Justice)	Germany
2	01/2016	5d	9 h	9	Transport and logistics (Aviation)	France
3	03/2016	2d	6 h	10	Government (Police)	Germany
4	03/2016	2d	10.5 h	19	Government (Military)	Germany

"Operation Digital Chameleon" #1 was played as a pretest and as the game worked out well its data is included in the analysis. Game #2 and #3 were played in "standard mode" with one red and one blue team. Game #4 explored a mode with one Team Blue to defend a Critical Infrastructure against three Red Teams.

3 The Game Design of "Operation Digital Chameleon"

"Operation Digital Chameleon" is designed as a table-top Cyberwargame (i.e. a game without IT support). The game board represents the IT-Infrastructure of a Critical Infrastructure. Participants play in Team Red as attackers and in Team Blue as IT-Security defenders. The Game Master (Team White) moderates.

3.1 The Target Group

The game is created the target group of: IT-Security professionals with experience and expertise on an operational, conceptual or strategic level. The target group comprises Chief Risk Officers, IT-Security Management, IT- and IT-Security Administrators, SCADA Operators, Business Continuity Management or Security Guards.

The underlying assumption is that a wide range of experience and expertise allows developing comprehensive strategies. This view is also met by the German Federal Office for Information Security: IT-Security, especially for Critical Infrastructures cannot be sufficiently achieved by a sole consideration of physical security. It can only be achieved by a cooperation of specialized teams and all responsible actors like experts of Business Continuity Management or Crisis Management [14].

3.2 The Board and the Teams

The game board (cf. Fig. 1) depicts the IT-Infrastructure of a Critical Infrastructure – without any IT-Security instruments. This IT-Infrastructure comprises (mobile) clients, various servers, a database and with reference to Critical Infrastructures, an industrial

network with a Human Machine Interface, an industrial system and an industrial Ethernet switch. Note that these elements are normally used in factories, harbors, energy systems, water power plants, and others [15]. Other elements of the board are designed to resemble industrial systems. Industrial systems and their subcomponents are designed for long lifetimes and there are many systems still in operation that run Windows NT and Windows XP [16]. For this reason, the game board does also include older systems for which patches are no longer available.

Fig. 1. The board of "Operation Digital Chameleon"

This IT-Infrastructure with networks and connections has to be accepted by Team Blue and cannot be rebuilt or disconnected. e.g., it is a constraint to have the industrial network connected to the Internet because of remote maintenance purposes [17]. The board is handed out to all teams.

"Operation Digital Chameleon's" games are managed by a Game Master, who is labelled white. His job is to prepare the game and to supervise the teams during the game. The Game Master ensures that teams comply with the rules of the game, assesses the solutions, determines the winner and manages the reflection on the game. Additionally, the white team can include observers.

Team Red's job is to develop an attack against the Critical Infrastructure protected by Team Blue. The captain of Team Red chooses the one out of five threat actors, the team has to keep during the game. These threat actors are Nation States, Script Kiddies, Hacktivists, Employees, and Cyber Criminals. These roles come with a description of a profile with motivation, intention and capabilities. Note that the five chosen different threat actors are taken from the threat actor classification of Hald and Pedersen [18]. These five threat actors coincide with Robinson's identified actors who are threatening Critical Infrastructures [19].

Team Blue's intension is to protect – according to a predefined strategy – the assets. Team Blue knows that they will be attacked in future, but they neither know the intention of Team Red nor their attack vectors.

To establish teams "Operation Digital Chameleon" uses a lottery. A team typically consists of 3 to 6 participant, with a team captain and a recorder as special roles.

3.3 The Rules and the Game

All teams are instructed by the Game Master and specific rules of the game, which are handed out to the team captains.

"Everything is possible, nothing is necessary."

The teams are instructed to develop their solutions based on the game material, the teams' material and based on their knowledge, and professional experience.

Team Red is instructed that it has to maintain the chosen role and that the attack needs to be plausible for the chosen threat actor's role such that it fits to motivation, skills and capabilities. e.g. if Team Red is playing Script Kiddies, it is not plausible to develop zero-day exploits or activate the hacking community for a Distributed Denial of Service-Attack.

Team Red is encouraged to work with attack trees. The term attack tree was coined by Schneier. He describes attack trees as follows: "Attack trees provide a formal, methodical way of describing the security of systems, based on varying attacks. Basically, you represent attacks against a system in a tree structure, with the goal as the root node and different ways of achieving that goal as leaf nodes." [20].

The Game Master evaluates Team Blue's and Team Red's plans and determines in how far the strategies have been successful. Plans and underlying assumptions have to be feasible and plausible for all – especially for the opposing team. Note that this needs to be done in a competent, transparent way.

At the beginning of the game, each team starts with a mission, handed out by the Game Master. Whereas Team Blue's intension is, to prevent the fictive organization from being attacked, it is the intension of Team Red to find a way into the company.

The later missions are designed to facilitate an elaboration of the attack and defense strategies, i.e. to find inconsistencies, implicit, explicit or possibly vulnerable assumptions in the team's strategies, to define alternative courses of actions, and to facilitate dealing with (partial) strategy disclosure to the opposing team. Teams are encouraged to use assumption based planning [21].

The typical course of the game bases on four subsequent missions. All missions are handed out by the Game Master according to a predefined schedule.

For playing the game, several rooms are needed. The main room is used for briefing, presentation and debriefing and was equipped with flipchart, several whiteboards, and the Game Master's notebook including a projector. The Game Master moves between rooms, provides clarifications to rules or scenario and observes, keeps time and hands out missions.

After the briefing is done, the team captains get the first mission and work with the teams in separate rooms. Rooms are equipped with a large table, so all team members can see the board, which is printed in DIN A0 format. It is the team captain's job to organize the time including breaks and to find an appropriate working atmosphere.

3.4 The Debriefing

The debriefing aims to solicit emotions, proposals on improvements of the gaming experience and a self-assessment of IT-Security Awareness levels. A discussion of attack and defense strategies is part of the debriefing.

The debriefing in "Operation Digital Chamaeleon" is structured according to the six-phases model of Thiagarajan and guided by the following questions: "How Do You Feel?" (Phase 1), "What Happened?" (Phase 2), "What Did You Learn?" (Phase 3), "How Does This Relate To The Real World?" (Phase 4), "What If?" (Phase 5), and "What Next?" (Phase 6) [22]. The six phases are accompanied by one or more key questions. The Game Master decides on the methods to be used in debriefing: Moderated group discussion (1), questionnaire (2), and presentation of written down answers by participants (3).

4 Results of "Operation Digital Chameleons" First Four Games

This section reflects the games and their results with a focus on the attack vectors and the participants' experiences. In the after-action review of the four games, the observation of the Game Master and the notes have been reviewed and reflected. From the Game Master's perspective, all four games were a success:

- The results of the questionnaires, the participants' statements on fun factor, method and the game material in the debriefing are mainly positive, no participant dropped out. Observations coincide with results of the questionnaires (cf. Sect. 4.1).
- The attack vectors and the countermeasures are non-trivial and demonstrate intensive and individual engagement in a creative solution development process (cf. Sects. 4.2 and 4.3).
- Game participants' responses in the questionnaire on insights and risk perception indicate that the participants benefit from the game (cf. Sect. 4.4).

The next four subsections present the analysis of the results of the four games.

4.1 The Gaming Experience

The "fun factor" exemplifies our idea on the game. According to McConigal, having fun while playing is an important prerequisite and increases the participants motivation and the quality of results [23]. The Game Master observes the working atmosphere and looks, e.g., for outbursts of loud laughing. Also in all four games, no game participant dropped out. The "fun factor" was either queried in the moderated group discussions or in a questionnaire. Table 2 depicts participants' answers of game #4 with reference to the question/statement "Das Cyberwargame hat mir Spaß gemacht".

Table 2. Result regarding the fun factor in game #4

Statement	False (−)	False (−)	Neutral	True (+)	True (++)
The cyberwargame was a lot of fun	0	2	4	10	3

Most of the participants stated that the game was a fun. This result is also confirmed by the observations, the Game Master made in all four games.

4.2 Attack Vectors – The Attack Strategies

The development of attack vectors and countermeasures in a playful atmosphere is the core gaming activity. Non-trivial attack vectors and IT-Security measures with relevance and esprit are the results that we hope for. This section presents an analysis of attack vectors, set up by the red teams (cf. Table 3).

In debriefing, the discussion on attack vectors went on and here two attack vector innovations emerged. In debriefing of game #2, a participant brought up the idea, that an attacker could interrupt the flight simulator. The flight simulators are located on the premises of an airport and some components are connected via Wi-Fi. An attacker could use a drone that carries a Wi-Fi jammer to disturb the flight simulator. The discussion confirmed that such an attack scenario had not been considered by the IT-Security personnel although it occurred to be realistic to all participants.

Within the debriefing phase of game #3, a participant brought up the idea, that the wire of a keyboard could be used to inject keystrokes that can compromise the host, the keyboard is connected with. Neither the participants, nor the Game Master knew about such an attack vector. So an attack vector "CableJack" (inspired by MouseJack, an attack vector published in 2016 [24]) was developed. An online research and 7 interviews with IT-Security experts found no evidence that such an attack vector had been published – only later more research pointed to a publication of a similar attack vector [25] (For a more detailed analysis cf. [11]). We argue that "CableJack" is an attack vector innovation. It took significant research to identify a publication at a conference and we found an attack vector, which is similar, but not identical.

Table 3. Games of "Operation Digital Chameleon"

Game	Goal team red	Attack vectors	Winner
#1	Theft of data to get fame and glory	Usage of compromised USB-Sticks (zero-day exploit and dropper) Compromising the domain controller Compromising clients with rootkits via updating-mechanisms Gaining employees support for installation of keylogger	Team Blue
#2	Creating a backdoor for other attackers to get fame and glory	Infiltrate attackers at maintenance company Getting credentials by shoulder surfing Manual installation of malware at maintenance company Using update mechanism to install malware Distributed denial of service to the firewall Exploiting systems Usage of manipulated computer mice Sniffing WLAN data and attacking the WLAN hotspot Phishing Getting access to employee's data by burglary for blackmailing	Tie game
#3	Disclosure of internal data to get fame and glory	Corruption of cleaning personnel Infiltration of attackers as a trainee Installation of keylogger and micro-cameras Creation of an rogue access point Social engineering Extortion of employees	Team Blue
#4	Spying out mission critical data (Team Red #1)	Hijacking VPN-connection; Hijacking remote maintenance-connection Remote installation of malware Corruption of cleaning personnel Compromised USB sticks Denial of service by mail bombing	Team Red #1
	Theft of data for reselling (Team Red #2)	Corruption of cleaning personnel and IT-Personnel Social engineering	Team Blue
	Sabotage (Team Red #3)	Social engineering Theft of digital certificates Obfuscation by deleting logfiles Installation of hardware-keylogger	Tie game

4.3 IT-Security Measures – the Defense Strategies

Similar to the attack vectors, the IT-Security measures developed in the process were non-trivial and reflected the game participants experiences and professional knowhow. In all games, Team Blue set up countermeasures with elements of three categories:

Human Factor (1), Factor Organization (2), Technical Factor (3). To give some examples from four games, the Human Factor often contained measures like IT-Security Awareness trainings or instructions. Factor Organization contained measures like how to do security management for visitors (ID-cards, escort service) or how to dispose hardware. Protecting the IT-Infrastructure by endpoint protection, hardening server or port security are some examples of the Technical Factor.

Note that additionally, in game #1 to #4, countermeasures came up, that do not come naturally to mind as typical measures for preventing cyberattacks. To give an example, in game #2, Team Blue used "recognition of one's achievement" to motivate IT-Administrators and make them more resistant to corruption. Note that such a measure eventually has strong influence to positive satisfaction, arising from intrinsic conditions of the job itself (cf., e.g., Herzberg's Two-Factor Theory [26]).

4.4 Learnings, Insights and Planned Behavioral Changes

In debriefing, participants write their new or important insights onto cards, which are collected on a wall and discussed. E.g., in game #3, the participants' learnings included types of VPN-connections, shortcomings of device control, and new cyberattack scenarios. To give an example: Participants of game #3 discussed that threats to IT-Infrastructure are not only limited by technical attacks from outside, but also given by attacks, performed by humans inside the organization. E.g., a disgruntled employee or the cleaning personnel could be a severe threat to the organization and this is hard to control by technical measures.

In debriefing, the Game Master asks which changes in work practices the participants plan. The empirical basis comprises 85 statements from game #1 to #4 that were collected by questionnaire and notes from moderated group discussion. To analyze the statements we did a qualitative content analysis according to Mayring [27]. We coded each statement with one category. Statements containing several doings like "To think and work more interdisciplinary" were weighted against the best improvement and effect to IT-Security and mapped with the corresponding subcategory.

A first analysis shows from the 85 statements only one "resigned" – the others indicated that participants in fact plan to improve IT-Security. None of the statements indicated "wishful thinking", "radically different approaches to IT-Security" or a search for "black swans", i.e., highly unlikely, high impact events that are hardly possible to detect [5]. Instead statements referred to reasonable IT-Security measures with practical relevance. The range of statements includes "physical security" as, e.g., locking ones' office door and novel IT-Security approaches, as e.g., using information repositories like Twitter or Shodan for up-to-date information. We argue that this indicates that the game raises awareness for sensible IT-Security measures. Table 4 presents a selection of statements.

Table 4. Selected statements of participants with reference to behavioral change

Individual	Organization	IT-Infrastructure
To lock the door when not in office	To make IT-security more transparent	To random check for external IT-Devices
To increase knowledge by reading more IT-security related reports, news and blogs	To improve IT-security awareness-training for employees	To look for old devices or overlooked test settings
To spend more time in IT-Security	To think up attack scenarios in cross functional teams	To do penetration testing more frequently; To use a testbed before installing updates
To use Shodan more frequently	To distribute competencies among personnel	

The second analysis distinguishes statements that refer to behavioral changes for Individuals (28 statements), Organizations (49 statements), and changes in IT-Infrastructure (8 statements). This indicates that the game addresses organizational IT-Security rather than "shopping for new IT-Security technology".

In a third round of analysis, we consider IT-Security topics. We identify in a qualitative content analysis 14 categories of IT-Security actions to be taken (cf. Table 5).

Table 5. Categories and number of statements

Subcategory	Number of statements
(Organizational) IT-security awareness training	16
Cross-functional teams	12
(Individual) IT-security awareness training	10
Monitoring	9
Information	8
Open-minded way of thinking	5
Existing IT-security concepts	5
Penetration testing	5
IT-Security operations	4
Audit	4
Information sharing	4
Attention	1
Resignation	1
Deterrence	1

16 statements refer to the need to do awareness training in the organization. In category "(Individual) Awarenesstraining" 10 statements refer to more IT-Security Awarenesstraining for the individual (who did the training). In category "(Individual) IT-Security Awarenesstraining", the analysis subsumed statements that participants see the need or usefulness to attend further trainings.

Code "Cross-functional teams" the content analysis subsumes 12 statements that emphasize the need for cross functional collaboration and on re-organizing collaboration in IT-Security within an organization. To give an example the participants planned to

do future risk analysis with experts from other domains (e.g. organizational leadership) or departments because other perspectives would be very helpful.

The need for better monitoring/supervision of technology, staff and service providers or trainees is subsumed in the 9 statements of "Monitoring". "Information" (8 statements) comprises all statements on the use or more information sources or different information repositories and better processing of this information. Such statements refer to Twitter, technical blogs or reports.

The games in particular trigger to have interdisciplinary ways of working (code "Cross-functional teams"). This corresponds to the Game Masters' notes: Cross-functional Red or Blue Teams developed very good results because non-IT measures were included. On the red side, corruption or blackmailing can improve traditional attack vectors, on the blue side, offering motivators (Herzberg's Two-Factor Theory) can prevent employees from becoming an internal offender.

On the technical level, there was no need for more IT-Security instruments. However some participants stated they want to look for vulnerabilities like misconfigurations or outdated technologies. Looking for vulnerabilities and eliminating them was coded as "Penetration testing" (5 statements).

To summarize the analysis, with exception to one statement (resignation), all participants liked the game, were motivated to change their working behavior to improve IT-Security on the individual, organizational and IT-Infrastructural level.

5 Limitations, Discussion and Conclusion

In this article we present the design of "Operation Digital Chameleon" and the selected results of the four games #1 to #4. Game participants liked the game experience, the attack and defense strategies developed are sophisticated and realistic with respect to the level of sophistication of current APTs. Game participants took out ideas for their day-to-day work. These ideas demonstrate that "Operation Digital Chameleon" prepares game participants for the challenges of IT-Security: Game participants plan to solicit more information on Cybersecurity, plan to have more awareness trainings both for themselves as well as for their organization and they see the need to reorganize IT-Security in cross-functional teams.

Our study has several limitations, some limitations are inherent to design oriented research: The game design is the result of a creative search process, the development of the game is ongoing, the Cybersecurity domain is undergoing rapid changes due to innovations and driven by legislation and regulation, and the results of a game depend on social and cognitive abilities of participants. In [28], Hofmann gives examples like learning effects.

According to Hevner – there is no "best solution" for such wicked problems in unstable contexts. Our approach claims to be useful for the domain of IT-Security. It is however inherently difficult to establish the link between an awareness training as a game, IT-Security awareness in day-to-day work and IT-Security levels. A limitation of "Operation Digital Chameleon" is that it is designed for IT-Security experts and small groups.

For the second half of 2016, more games are planned. We plan to do variations with game boards and missions that are specific for Critical Infrastructure sectors. We also plan to address more explorative formats that allow looking for black swans.

Acknowledgments. We would like to acknowledge the funding from BMBF for project "Vernetzte IT-Sicherheit Kritischer Infrastrukturen" (FKZ: 16KIS0213). We thank all participants for making "Operation Digital Chameleon" a success, Marko Hofmann and Alexander Laux for their contributions in the design of "Operation Digital Chameleon".

References

1. McAfee: Combating Advanced Persistent Threats, Santa Clara (2011)
2. Symantec: Advanced Persistent Threats: How They Work. http://www.symantec.com/theme.jsp?themeid=apt-infographic-1
3. Rowney, K.: What We Talk About When We Talk About APT. http://www.symantec.com/connect/blogs/what-we-talk-about-when-we-talk-about-apt#!
4. Rouse, M.: advanced persistent threat (APT). http://searchsecurity.techtarget.com/definition/advanced-persistent-threat-APT
5. Suárez-Lledó, J.: The black swan: the impact of the highly improbable. Acad. Manag. Perspect. **25**, 87–90 (2011)
6. Perla, P.P.: The Art of Wargaming: A Guide for Professionals and Hobbyists. US Naval Institute Press (1990)
7. Geilhardt, T., Mühlbrandt, T.: Planspiele im Personal- und Organisationsmanagement. Hogrefe Publishing Göttingen (1995)
8. Geuting, M.: Planspiel und soziale Simulation im Bildungsbereich (Studien zur Pädagogik, Andragogik und Gerontagogik/Studies in Pedagogy, Andragogy, and Gerontagogy). Lang, Peter Frankfurt (1992)
9. ENISA: On National and International Cyber Security Exercises. Europäische Agentur für Netz- und Informationssicherheit (ENISA), Heraklion (2012)
10. ENISA: The 2015 Report on National and International Cyber Security Exercises. Europäische Agentur für Netz- und Informationssicherheit (ENISA), Athen (2015)
11. Rieb, A., Lechner, U.: Operation digital chameleon – towards an open cybersecurity method. In: Proceedings of the 12th International Symposium on Open Collaboration (OpenSym 2016), Berlin, pp. 1–10 (2016)
12. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design science in information systems research. MIS Q. **28**, 75–105 (2004)
13. BMI: Definition "Kritische Infrastrukturen" (2009). http://www.bmi.bund.de/SharedDocs/Downloads/DE/Themen/Sicherheit/BevoelkerungKrisen/Sektoreneinteilung.pdf?__blob=publicationFile\nBundesministeriumdesInnern2009-DefinitionKritischeInfrastrukturen.pdf
14. UPKRITIS: UP KRITIS Öffentlich-Private Partnerschaft zum Schutz Kritischer Infrastrukturen., Bonn (2014)
15. Kamath, M.: Hackers can remotely take over Nuclear Power Plants by exploiting vulnerability in IES. http://www.techworm.net/2015/08/security-flaws-in-industrial-ethernet-switches.html
16. Neitzel, L., Huba, B.: Top ten differences between ICS and IT cybersecurity (2014). http://www.isa.org/standards-and-publications/isa-publications/intech-magazine/2014/may-jun/features/cover-story-top-ten-differences-between-ics-and-it-cybersecurity/
17. Erswell, D.: The SCADA Internet - What to Look Out for, pp. 1–5 (2015)

18. Hald, S., Pedersen, J.: An updated taxonomy for characterizing hackers according to their threat properties. In: 2012 14th International Conference on 2012 Advanced Communication Technology (ICACT), pp. 81–86 (2012)
19. Robinson, M.: The SCADA threat landscape. In: 1st International Symposium on ICS & SCADA Cyber Security Research 2013 (ICS-CSR 2013), pp. 30–41 (2013)
20. Schneier, B.: Attack Trees - Modeling security threats. Dr. Dobb's J. (1999)
21. Dewar, J.A.: Assumption-Based Planning - A Tool for Reducing Avoidable Surprises. The Press Syndicate of the University of Cambridge, Cambridge (2002)
22. Thiagarajan, S.: How to maximize transfer from simulation games through systematic debriefing. Simul. Gaming Yearb. **1993**, 45–52 (1993)
23. McConigal, J.: Besser als die Wirklichkeit!: Warum wir von Computerspielen profitieren und wie sie die Welt verändern. Heyne Verlag, München (2012)
24. Newlin, M.: MouseJack Injecting Keystrokes into Wireless Mice (2016)
25. Spill, D.: USBProxy - an open and affordable USB man in the middle device. In: 2014 ShmooCon Proceedings (2014)
26. Herzberg, F., Mausner, B., Snyderman, B.B.: The motivation to work. Transaction publishers, Piscataway (1959)
27. Mayring, P.: Qualitative Inhaltsanalyse. Grundlagen und Techniken. (2008)
28. Hofmann, M.: Abschlussbericht taktisches Wargaming. ITIS, München

Cyber Security Investment in the Context of Disruptive Technologies: Extension of the Gordon-Loeb Model and Application to Critical Infrastructure Protection

Dimitri Percia David[1,2(✉)], Marcus Matthias Keupp[1,3], Solange Ghernaouti[2], and Alain Mermoud[1,2]

[1] Swiss Cybersecurity Advisory and Research Group (SCARG), University of Lausanne, 1015 Lausanne, Switzerland
dimitri.davidpercia@unil.ch
[2] Department of Defense Management, Military Academy at ETH Zurich, 8903 Birmensdorf, Switzerland
[3] Institute of Technology Management, University of St. Gallen, Dufourstrasse 40a, 9000 St., Gallen, Switzerland

Abstract. We propose an extension of the Gordon-Loeb model by considering multi-periods and relaxing the assumption of a continuous security breach probability function. Such adaptations allow capturing dynamic aspects of information security investment such as the advent of a disruptive technology and its consequences. In this paper, the case of big data analytics (BDA) and its disruptive effects on information security investment is theoretically investigated. Our analysis suggests a substantive decrease in such investment due to a technological shift. While we believe this case should be generalizable across the information security milieu, we illustrate our approach in the context of critical infrastructure protection (CIP) in which security cost reduction is of prior importance since potential losses reach unaffordable dimensions. Moreover, despite BDA has been considered as a promising method for CIP, its concrete effects have been discussed little.

1 Introduction

The Gordon-Loeb (GL) model has established a general setup for determining the optimal level of information security investment[1] [8,9]. Nevertheless, its framework evades dynamic issues such as perverse economic incentives[2] and the

Dimitri Percia David—Short paper submitted to the 2016 CRITIS conference under topic 1: Technologies: Innovative responses for the protection of cyber-physical systems.

[1] The terms *cyber security* and *information security* are considered as synonyms and therefore substitutable in this paper.

[2] E.g. *externalities* arising when decisions of one party affects those of others [2].

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 296–301, 2017.
https://doi.org/10.1007/978-3-319-71368-7_25

advent of a disruptive information security technology.[3] Yet, accounting for such dynamic issues could significantly enrich the initial model by giving supplementary time-related insights of investment in information security.

By extending the original GL model to a multi-periods setup, and by relaxing the assumption of a continuously twice-differentiable security breach probability function, this paper aims to give additional insights on information security investment by theoretically investigating the dynamic impact of a disruptive technology. The case of big data analytics (BDA)[4] is assessed.

The remainder of this paper is structured as follows. Section 2 contains our suggested GL model extensions that are needed in order to capture the dynamic consequences of disruptive technologies on information security investment. Section 3 calls for empirical investigation in the field of critical infrastructure protection (CIP), while the last section concludes.

2 Extending the GL Model

By focusing on costs and benefits associated with cyber security, the GL model defines that each organization's concern is to maximize its expected net benefit coming from information security expenditures. This corresponds to minimizing the total expected cost, equivalent to the addition of the expected loss due to information security breaches and the cost of information security activities undertaken in order to counter such breaches.[5]

The cost of information security activities undertaken in order to counter cyber security breaches is intrinsically related to the efficiency of those activities. While organizations in the last three decades have developed numerous technical means to increase cyber security centered on *signature*-based tools – such as access control mechanisms, intrusion detection systems, encryption techniques, firewalls and anti-virus software, etc. – the success of these conventional measures has been limited [3,6]. These latter aim to detect threats by examining incoming traffic against *bad signatures*. Yet, such an approach is becoming more

[3] In economic terms, the notion of *disruptive technology* [7] refers to a radically innovative technology that significantly disrupts existing economic structures, markets and value networks, displacing established leading products and services.

[4] In this paper, the term big data refers to data whose complexity impedes it from being processed (mined, managed, queried and analyzed) through conventional data processing technologies [10,11]. The complexity of big data is defined by three aspects: 1°, the volume (terabytes, petabytes, or even exabytes (1000^6 bytes); 2°, the velocity (referring to the fast paced data generation); and 3°, the variety (referring to the combination of structured and unstructured data) [10,11]. The field of BDA is related to the extraction of value from big data – i.e., insights that are non-trivial, previously unknown, implicit and potentially useful [11]. BDA extracts patterns of actions, occurrences, and behaviors from big data by fitting statistical models on those patterns through different data mining techniques (predictive analytics, cluster analysis, association rule mining, and prescriptive analytics) [5,12].

[5] The model description and its assumptions have been previously explained in details by [8,9].

and more ineffective as it only detects threats that have been already witnessed in the past, creating a lag between the development of *signatures* and the rapid expansion of cyber threats. Moreover, *zero-day vulnerabilities* cannot be caught by such an approach. Hence, the above-mentioned conventional cyber security techniques can be easily rendered ineffective by cyber criminals [11]. Such a scenario is even amplified in the era of big data as up to exabytes of information are being transferred daily, giving cyber criminals the possibility to accessing networks, hiding their presence and inflicting damage efficiently. The emphasis of information security is thus shifting from a conventional approach of detecting *bad signatures* – by monitoring Internet traffic and networks – to an innovative examination for detecting *bad actions* [11]. Security analytics embrace these changes by suggesting a disruptive approach for producing cyber security [11]. It employs procedures from BDA to derive relevant security information for preventing cyber breaches [12]. Targeted methodologies based on anticipation and forecast abilities such as *real-time analytics*, *early warning* and *dynamic detection* are likely to become the next generation of technologies implicating superior returns on investment *vis-à-vis* conventional measures [4,12]. Accordingly:

Proposition 1. *If a disruptive technology is employed for producing cyber security, time series of investment in cyber security witness a statistical structural break due to a greater efficiency of the mentioned disruptive technology compared to conventional tools.*

In order to capture such dynamic aspects, we propose that this presumed time-related investment level differential calls for an adaptation of the GL model in two important ways:

Firstly, in order to capture the advent of a disruptive technology such as BDA and its dynamic consequences on information security investment, a temporal setup has to be implemented. Since the GL model is constructed on a single-period, it excludes the fundamental temporal dimension for analyzing the technological shifts' dynamics induced by efficiency improvements. Hence, the extension of the original single-period model to a multi-periods setup might bring significant insights for understanding dynamic aspects of information security investment that were originally evaded. Specifically, adapting from [8], the maximization of the expected net benefits in information security (ENBIS) at the end of the specified horizon $[1, n]$ is:

$$Max \;\; ENBIS(z_i) \left\{ \sum_{i=1}^{n} \left[v_i - S_i(z_i, v_i) \right] L_i - z_i \right\}$$

where for each period i, v_i is the organization's inherent vulnerability to information security breaches; S_i is the organization's security breach function, defined as the probability that an information security breach would occur; z_i is the organization's investment in cyber security; and L_i is the potential loss associated with the security breach.

Secondly, the presumed technological shift induced by superior efficiency of a disruptive technology challenges the assumption of a continuously twice-

differentiable security breach probability function. The original model defines continuously decreasing but positive returns to scale of cyber security investment. Yet, this continuously twice-differentiable setup leaves no room for a discrete emergence of a technological shift brought by a disruptive and more efficient technology.[6] In such a theoretical framework, the elasticity of protection of cyber security activities evades radical technological progress. However, a technological progress induced by the implementation of a disruptive technology such as BDA might considerably reduce information security investment by bringing suggestively greater returns on investment.[7] Hence, the investor realizes a *Pareto* improvement by either obtaining a better level of protection for the same investment, or obtaining the same level of protection at a lower cost (since less resources such as time and human labor – that can be largely substituted by algorithms and automation – might be used). As a result, with the implementation of BDA, information security investment might be significantly reduced by disruption. BDA would introduce a discontinuity in the security breach probability function, modifying the original GL model assumption of continuity. Accordingly, adapting from S^I [8], we claim that the following security breach probability function $S_i^{I'}$ captures the advent of a disruptive technology by introducing discontinuity through the parameter d_i:

$$S_i^{I'}(z_i, v_i) = \frac{v_i}{(\alpha_i z_i + 1)^{\beta_i + d_i}}$$

where for each period i, α_i and β_i represent productivity parameters of a given cyber security technology, and d_i represents a dummy variable that takes the value 0 when no disruptive technology is used, and 1 otherwise.

[6] [8] explicitly acknowledge that they «abstract from reality and assume that postulated functions are sufficiently smooth and well behaved », and therefore creating favorable conditions for applying basic differential calculus, simplifying the optimization problem of the security investment phenomenon. Although a smooth approximation of the security investment phenomenon done by [8] is a reasonable first approach in order to deliver insights concerning the problem of determining an optimal level of cyber security investment, such an approach lacks of realism. As explicitly mentioned by [8] themselves: "[. . .] in reality, discrete investment in new security technologies is often necessary to get incremental result. Such discrete investment results in discontinuities ".

[7] The following cases illustrate this claim. In BDA, an extremely large, fast paced and complex amount of information can be processed with significantly shortened timeframes and – once the fixed cost of the systems and algorithm for investigating threat patterns is invested – at almost zero marginal cost per additional unit of information [13]. Furthermore, the *real-time analytics* provided by big data algorithms are likely to neutralize any attacker's information advantage, such that the probability of a cyber breach should be reduced. For example, an attacker can exploit *zero-day vulnerabilities* by knowing where to attack, while the defender does not know and hence has to protect all potential entry spots. As *real-time analytics* reveals both the time and the position of the attack as it happens, the defender can react precisely in the attacked spot only and thus saves any unnecessary investment in the protection of spots, which, eventually, are never attacked.

While the multi-periods setup and the suggested security breach probability function $S_i^{I'}$ constitute the main theoretical contributions in order to extend the original GL model, their application to a concrete context is necessary in order to exemplify and demonstrate their relevancy, as well as for empirically testing them in a further research.

3 Application to CIP

While we believe that the above-mentioned reasoning should be generalizable across any kind of cyber security concerns, we illustrate our approach in the context of CIP in which a cyber security cost reduction is of prior importance since potential losses reach unaffordable dimensions [9]. Moreover, despite BDA has been considered as a promising method for CIP, its concrete implications have been discussed little.

Critical infrastructures (CIs) are systems, assets or services which are so vital to the society that any extended disturbance or destruction of them would strongly affect the functioning of security, economic system, public health or safety, or any combination of the above [4]. A cyber security breach inflicted to CIs generates massive negative externalities, especially because of the increasing interdependency and technical interconnectedness of different CIs. Particularly, the cascading effect of failures among CIs could pose a serious threat to the society's crucial functions [1,9]. Cyber attacks attempting to exploit the interconnectedness of CIs are the main and most dangerous asymmetric threat CIs have to face today and in the future [4]; as a result, cyber security issues are the main challenge for CIP. Thus, the issue of information security investment seems highly relevant in the context of CIP, and particularly so from a social welfare perspective [9].

The application of our extension to the context of CIP should provide us with a relevant and seminal ground on which hypothesis can be formally modeled and simulated. In the case of human-processed information and defense tactics, these issues would probably make the optimal level of protection difficult to attain or even impossible to finance since the expected loss would be extreme in the case of cascading CIs failure, and hence the resulting cyber security investment would also have to reach extreme levels [9]. However, with the effects BDA technology may have on investment in information security, investment needs for CIs protection may decrease due to superior efficiency. To analyze the extent to which (if at all) this is the case, further research should propose to simulate a multi-players and multi-periods game that models the cyber security of CIs in the era of big data.

4 Concluding Comments

In this paper, we proposed an extension of the GL model by adapting its original theoretical framework to capture dynamic aspects of investment. Two important contributions have been proposed. First, we argued that a single-period model

is not adapted to capture dynamic aspects of information security investment such as the advent of a disruptive technology. The extension to a multi-periods model is indeed necessary. Second, the security breach probability function of the original GL model could not be considered as continuously differentiable in the context of the introduction of a discrete radically innovative information security technology. These two arguments enrich the initial model by giving supplementary insights potential on information security investment. While we believe that this reasoning is generalizable across a wide range of cyber security concerns, we illustrated our approach in the context of CIP in which cyber security breaches inflict unaffordable social costs that urge to be reduced.

Further research should propose to simulate a multi-players and multi-periods game that models the cyber security of CIs in the era of big data. Such a research – by collecting simulated data and quantitatively analyzing them – would contribute to complement this theoretical paper in order to test if the theory's intuition is observable in a simulated setup.

References

1. Alcaraz, C., Zeadally, S.: Critical infrastructure protection: Requirements and challenges for the 21st century. Int. J. Crit. Infrastruct. Prot. **8**, 53–66 (2015)
2. Anderson, R.: Why information security is hard - an economic perspective, pp. 358–365. IEEE Comput. Soc (2001)
3. Anderson, R., Moore, T.: The economics of information security. Science **314**(5799), 610–613 (2006)
4. Anderson, R., Fuloria, S.: Security economics and critical national infrastructure. In: Moore, T., Pym, D., Ioannidis, C. (eds.) Economics of Information Security and Privacy, pp. 55–66. Springer, US (2010). https://doi.org/10.1007/978-1-4419-6967-5_4
5. Cardenas, A.A., Manadhata, P.K., Rajan, S.P.: Big data analytics for security. IEEE Secur. Priv. **11**(6), 74–76 (2013)
6. Chen, H., Chiang, R.H., Storey, V.C.: Business intelligence and analytics: from big data to big impact. MIS Q. **36**(4), 1165–1188 (2012)
7. Christensen, C., Raynor, M.E., McDonald, R.: What Is Disruptive Innovation? Harvard Business Review, Boston (2015)
8. Gordon, L.A., Loeb, M.P.: The economics of information security investment. ACM Trans. Inf. Syst. Secur. (TISSEC) **5**(4), 438–457 (2002)
9. Gordon, L.A., Loeb, M.P., Lucyshyn, W., Zhou, L., et al.: others: Externalities and the magnitude of cyber security underinvestment by private sector firms: a modification of the Gordon-Loeb model. J. Inf. Secur. **6**(01), 24 (2014)
10. Laney, D.: 3D data management: Controlling data volume, velocity and variety. META Group Research Note 6, 70 (2001)
11. Mahmood, T., Afzal, U.: Security analytics: big data analytics for cybersecurity: a review of trends, techniques and tools. In: 2013 2nd National Conference on Information Assurance (NCIA), pp. 129–134 (2013)
12. Sathi, A.: Big Data Analytics: Disruptive Technologies for Changing the Game. Mc Press, Los Angeles (2012)
13. Sowa, J.F.: Conceptual Structures: Information Processing in Mind and Machine. Addison-Wesley Pub., Reading (1983)

Behavioral Intentions and Threat Perception During Terrorist, Fire and Earthquake Scenarios

Simona A. Popușoi[1(✉)], Cornelia Măirean[1], and Grigore M. Havârneanu[2]

[1] Alexandru Ioan Cuza University of Iasi, Iasi, Romania
simona.popusoi@yahoo.com, cornelia.mairean@uaic.ro
[2] Security Division, International Union of Railways (UIC), Paris, France
havarneanu@uic.org

Abstract. The aim of this study is to assess the determinants of behavioral intention and threat perception in three types of crisis situations (fire, earthquake, and terrorist attack). We considered both individual factors (locus of control, illusion of control, optimism bias, knowledge about crisis management, and institutional trust) and situational ones (the presence vs. absence of significant others). A sample of 249 students was included in the study. The crisis type and the presence of significant others were manipulated through scenarios. The participants were randomly assigned to one of the experimental conditions and filled in self-report scales which assessed individual factors, behavioral intention and threat perception. The results showed that individuals prefer an affiliative behavioral response in all crisis types. Institutional trust, locus of control, and the level of knowledge predicted the affiliative behavior. The implications for crisis situation management of crowded places and risk communication are discussed.

Keywords: Behavioral intention · Threat perception · Crisis · Human factor

1 Introduction

Critical Infrastructures (CIs) are those physical and information technology facilities, networks, services and assets which, if disrupted or destroyed, would have a serious impact on the health, safety, security or economic well-being of citizens or the effective functioning of governments in European Union (EU) countries [1]. Events affecting CIs continuity not only come from natural disasters such as flooding, earthquakes, hurricanes, tsunamis, etc., but also result from accidental man-made incidents (e.g. fire) or from willful malicious acts (i.e. threats). Natural disasters differ from threats due to the absence of malicious intent. Additionally, accidents occur without intent and are caused by human errors or technical failures [2].

In the last years the terrorist threat has increased in several EU countries spreading uncontrolled fear, distrust, and other economic and psychological effects with a heavy impact on society [3]. The recent European attacks have proven that public crowded places like transport hubs, stadiums, concert halls or shopping centers have become new targets for terrorist attacks. During disasters, major accidents or terrorist attacks, citizen response usually involves an alarm, an acute and a recovery stage [4]. The little existing

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 302–307, 2017.
https://doi.org/10.1007/978-3-319-71368-7_26

research has shown that during the acute stage lay citizens react more effectively than we would intuitively expect, and often respond as effective as well-trained emergency personnel [4]. While fear is the dominant emotion across different types of disasters [5], it appears that in most cases panic does not take over the rational behavior [4]. In fact the typical response to a variety of physical threats is neither fight nor flight but affiliation – that is seeking the proximity of familiar persons and places, even though this may involve approaching or remaining in a situation of danger [6]. People's responses may depend on one's ability to recognize and to make sense of cues to life-threatening stimuli but there have also been insights that people tend to underestimate such cues [5]. The EU Getaway Project has shown that individuals tend to evacuate as a group and those who do not group usually decide to investigate the critical incident or to warn others [7]. Further recent research conducted in the EU IMPROVER project suggests that public expectations about service continuity after disaster is high, and points out the importance of the social/societal dimension of resilience [8]. Yet, the ongoing challenge is to find solutions to raise citizen awareness and improve their preparedness, also by studying their psychological characteristics.

Locus of control (LOC) is an interactive expression of the relation between an individual and its environment [9]. Internal LOC refers to the belief that people control their own destiny. On the other hand, individuals with external LOC consider fate or powerful others to be in control over the outcomes of their behavior and shift their own responsibility to external agents such as other persons or institutions. Previous findings showed that individuals with internal LOC tend to be more proactive in taking precautionary measures against security breaches [10]. Additional individual factors may cause maladaptive decisions when facing a threat. For example, optimism bias relates to the perception of personal invulnerability, representing the underestimation of the likelihood of experiencing negative events [11]. It represents a defensive distortion that could undermine preventive actions [12], interfere with precautionary behavior and ultimately aggravate individual's risk-seeking tendency [13]. Another factor is the illusion of control, namely the people's tendency to perceive that they have more control over their own behavior or over the environment than they can actually have [14]. Whereas optimism refers to a generalized expectancy for positive outcomes independent of the source of the outcomes, the illusion of control locates the source of the expected outcome in terms of personal control [15]. Applied social studies have shown that people overestimate personal control in situations that are heavily or completely chance-determined, but they also under-estimate their personal control when they have indeed a great deal of objective control [16].

These individual psychological factors are highly understudied in relation to major disasters or security threats. This paper aims to explore if psychological factors (knowledge level about crisis management, LOC, illusion of control, institutional trust, and optimism bias) are likely to influence the threat perception and the behavioral intentions during three different types of crisis situations. The group membership was also manipulated by describing the crisis situation as an in-group (the person is among close friends) or an out-group condition (the person is among strangers).

2 Method

The sample included 249 Psychology students (78.7% females) who volunteered to participate in the study. The mean age of the participants was 22.35 years (SD = 5.47 years), ranging from 18 to 48 years. Initially, the participants provided information about their age and gender and filled in the pre-experimental questionnaire measuring emergency knowledge, locus of control, and optimism bias. *Behaviour, Security, and Culture Survivor Questionnaire* [17] was used to measure the emergency knowledge using a 6-point Likert scale ranging from 0 to 5 (0 = not at all, 5 = extremely). *Locus of control scale* [18] was used to measure the extent to which a person perceives events as being a consequence of his or her own behavior. The 17-items were assessed with a 6-point scale, ranging from 0 (totally disagree) to 5 (totally agree). A total score was computed, where high scores indicate externality ($\alpha = .81$). *Optimism bias* was measured with 5 items assessing the probability that a specific event can occur in the future, in the participant's life. The participants could choose the probability that a certain event will occur, using a number from 0 to 100. There were six experimental conditions, based on scenarios where we manipulated 3 types of crisis situations (terrorist bomb attack, fire, and earthquake) and the presence of other people: close friends (in-group) and strangers (out-group). Participants were randomly assigned to one of the conditions (terrorist attack, in-group, N = 39; terrorist attack, out-group, N = 41; fire, in-group, N = 41; fire, out-group, N = 42; earthquake, in-group, N = 43; earthquake, out-group, N = 43). They were asked to read the scenario and to imagine themselves instead of the main character. All six scenarios described a flash disaster in a crowded public building (i.e. shopping center). This location was chosen because it is the most popular local commercial and spare time area that all people know very well. In order to measure behavioral intentions we used the same instrument as presented above [17]. Items referred to either an affiliative behavior or fight behavior. *Threat perception* was measured through two items using a 6-point Likert scale ranging from 0 to 5 (0 = not at all, 5 = extremely). *Perceived control* was assessed through one item using the same presented scale. *Institutional trust* was measured through one item using the same scale.

3 Results

The mixed analysis of variance (ANOVA) with type of scenarios as a between-subjects factor and type of behavioral intention (affiliative, flight, or fight behavior) as a within factor was conducted in order to examine the effect of the experimental manipulation on behavioral intentions. Overall, regardless of the crisis type (terrorist bomb attack, fire, and earthquake) or group condition (ingroup and outgroup), individuals reported higher levels of affiliative behavior ($M = 3.86$; $SD = .05$) compared to the flight ($M = 2.18$; $SD = .06$) or fight behavior ($M = 2.05$; $SD = .05$) [$F(2, 480) = 342.33$; $p < .001$; $\eta_p^2 = .588$]. Moreover, there is a combined effect of behavioral intentions (affiliative, flight, or fight) and scenario type (all six scenario conditions) [$F\ (10, 470) = 2.29$; $p = .01$; $\eta_p^2 = .047$], indicating that in all six conditions individuals preferred the affiliative behavior in contrast with fight or flight behavior.

We used stepwise linear regressions to compute prediction model for affiliative behavior separately for the three crisis types (bomb attack, fire, and earthquake) having as predictors the group type (ingroup vs outgroup), LOC, optimism bias, knowledge level, perceived control, and institutional trust. The *affiliative behavior* during a bomb attack was predicted by institutional trust ($\beta = .45$; $p < .001$) and LOC ($\beta = .23$; $p = .02$), both of them predicting 24.6% of the affiliative behavior ($R^2_{aj} = .246$; $p < .001$). This means that the people with higher level of institutional trust and external LOC will have the tendency to affiliate during a bomb attack. Moreover, during the fire scenario, institutional trust ($\beta = .42$; $p < .001$) and the group type ($\beta = .38$; $p < .001$) predicted 37.7% of the affiliative behavior ($R^2_{aj} = .377$; $p < .001$), meaning that when being among friends and with higher levels of institutional trust, individuals are likely to prefer the affiliative behavior. Lastly, during the earthquake scenario, the individuals with low levels of knowledge ($\beta = -.265$; $p = .01$) and high levels of institutional trust ($\beta = .30$; $p = .005$) had the tendency to affiliate ($R^2_{aj} = .013$; $p = .001$).

4 Discussion

First of all, our results indicated that individuals would prefer to affiliate, regardless of the crisis type. Our results support previous findings suggesting that panic does not take over the rational behavior individuals [4]. The tendency to seek the proximity of familiar persons or places, even though it might imply approaching or remaining in a situation of danger, was stronger than the tendency to run [6]. Therefore, the current study brings further support for the "social attachment" model of collective behavior under threat, which sustains the fundamental social nature of humans and the primacy of attachments [19]. For crisis management, the fact that people typically chose affiliation might represent a challenging issue, since this response can delay or fail the take appropriate evacuation actions.

Secondly, individuals tended to affiliate more during a fire crisis and when being among friends. We suppose that the affiliative behavior during a fire crisis was more salient in the mind of the participants due to an event that occurred in Romania in November 2015 (i.e. a massive fire during a concert in Bucharest, which led to more than 60 fatalities and many more injuries). Another explanation may be that emergency trials developed in schools focus on correct actions during fires. The public knowledge about these disasters or the individual past experiences may determine the affiliation response, considered most common [6]. Moreover, individuals perceived a higher level of threat during a bomb attack, suggesting the fact that higher levels of information from the media may lead to a more accurate view over threat which may be linked to the fact that this was the only scenario involving a typical security threat compared to the other two which did not include a malicious intent.

Institutional trust is present in all three crisis types, suggesting that it can lead to a more adaptive response during a crisis. A more consistent and constant way of institutions to deal with emergency situation may lead to higher levels of trust of individuals and therefore increase the chances of survival. Future risk communication campaigns are needed in order to increase the citizen knowledge about crisis, since this factor could improve the threat perception by making it more accurate. As previous research

suggested, survivors' responses may depend on their ability to perceive cues to life-threatening stimuli [5]. Both previous information and experience can determine the level of risk perception [5], therefore public campaigns should provide more information about these disasters, in order to increase the level of knowledge. Emergency trials should also be implemented. Although these trials cannot replace the actual experience with threats and disasters, they allow people to create memory schemas of actions in crisis situation, which can help them choose appropriate response during real-life disasters [20].

The limitations of this study have to be taken into account when interpreting the results. First, we studied only self-reported behavioral intentions on a limited sample of respondents, thus the results do not claim that the responses of the participants are representative for all populations who experience these types of disasters. Second, it is well known in psychology studies that there can be notable gaps between the intended behavior and the actual behavior of people. Third, the participants' past experience with disasters was not considered in this study. Despite these limitations, the current study offers an overview of the determinants of the most common threat perceptions and behavioral responses to different natural and man-made disasters occurring in public settings. Institutional trust appeared to be a universal determinant factor of behavioral intention in all three crisis situations. Further investigation of the different situational and individual determinants of responses in crisis situation, as well as their interaction, is needed. Moreover, further studies should analyze in connection different types of responses, such as behaviors, emotions, and cognitions, in order to determine their intercorrelations. The present results have practical implications for public communication of risks affecting crowded public spaces. Based on our results risk communication should first develop or consolidate the trust of people in the responsible institutions and their emergency plans. Next, public security trainings should be tailored to specific threats or disasters, instructing people according to the unique characteristics of different events. This instruction will help them quickly recognize the threat cues and react appropriately, but may be very challenging to implement in practice. Since the human factor is an important element in crisis situations and emergency plans, partnership work should develop to better include the human aspects within CIP research. Social science can shed more light on how people perceive and accept risks, and can reveal their needs in terms of well-being during a disaster management. This requires close collaboration between scientists with different backgrounds: engineers, computer scientists, security experts, human factor specialists, psychologists, etc.

Acknowledgment. This work was supported by a grant of the Romanian National Authority for Scientific Research and Innovation, CNCS – UEFISCDI, project number PN-II-RU-TE-2014-4-2872

References

1. European Commission. Communication from the Commission to the Council and the European Parliament – Critical Infrastructure Protection in the fight against terrorism COM/2004/0702 final

2. Reason, J.: Human Error. Cambridge University Press, New York (1990)
3. Lazari, A.: European Critical Infrastructure Protection. Springer, Cham (2014)
4. Helsloot, I., Ruitenberg, A.: Citizen response to disasters: a survey of literature and some practical implications. J. Contingencies Crisis Manage. **12**(3), 98–111 (2004)
5. Grimm, A., Hulse, L., Preiss, M., Schmidt, S.: Behavioural, emotional, and cognitive responses in European disasters: results of survivor interviews. Disasters **38**(1), 62–83 (2014)
6. Mawson, A.R.: Understanding mass panic and other collective responses to threat and disaster. Psychiatry **68**(2), 95–113 (2005)
7. Burroughs, M., Galea, E.R.: Real time, real fire, real response: an analysis of response behaviour in housing for vulnerable people. In: Proceedings of the 6th International Symposium on Human Behaviour in Fire, vol. 6, pp. 477–488. Interscience Communications Ltd (2015)
8. Petersen, L., Fallou, L., Reilly, P., Serafinelli, E., Carreira, E., Utkin, A.: Social resilience criteria for critical infrastructures during crises. Deliverable 4.1 for IMPROVER (2016)
9. Rotter, J.B.: Generalized expectancies for internal versus external control of reinforcement. Psychol. Monogr. Gen. Appl. **80**(1), 1 (1966)
10. Tu, Z., Yuan, Y., Archer, N.: Understanding user behaviour in coping with security threats of mobile device loss and theft. Int. J. Mob. Commun. **12**(6), 603–623 (2014)
11. Weinstein, N.D., Klein, W.M.: Unrealistic optimism: present and future. J. Soc. Clin. Psychol. **15**, 1–8 (1996)
12. Schwarzer, R.: Optimism, vulnerability, and self-beliefs as health related cognitions: a systematic overview. Psych. Health **9**, 161–180 (1994)
13. Erenberg, E:. What type of disputes are best suited for alternative dispute resolutiondan analysis in the space of the odds of litigation. In: Proceedings of the Fourth Annual Meetings of Israeli Law & Economics Association (ILEA) 2005
14. Langer, E.J.: The illusion of control. J. Pers. Soc. Psychol. **32**, 311–328 (1975)
15. McKenna, F.P.: It won't happen to me: unrealistic optimism or illusion of control? Br. J. Psychol. **84**, 39–50 (1993)
16. Gino, F., Sharek, Z., Moore, D.A.: Keeping the illusion of control under control: Ceilings, floors, and imperfect calibration. Organ. Behav. Hum. Decis. Process. **114**(2), 104–114 (2011)
17. Knuth, D., Kehl, D., Galea, E., Hulse, L., Sans, J., Vallès, L., Roiha, M., Seidler, F., Diebe, E., Kecklund, L., Petterson, S.: BeSeCu-S–a self-report instrument for emergency survivors. J. Risk Res. **17**(5), 601–620 (2014)
18. Craig, A.R., Franklin, G.A.: A scale to measure locus of control of behavior. Br. J. Med. Psychol. **57**, 173–180 (1984)
19. Mawson, A.R.: Is the concept of panic useful for study purposes. Behavior in fires [NBS Report NBSIR-802070]. US Department of Commerce, Washington, DC (1980)
20. Leach, J.: Why people "freeze" in an emergency: temporal and cognitive constraints on survival responses. Aviat. Space Environ. Med. **75**(6), 539–542 (2004)

An Operator-Driven Approach for Modeling Interdependencies in Critical Infrastructures Based on Critical Services and Sectors

Elisa Canzani[1(✉)], Helmut Kaufmann[2], and Ulrike Lechner[1]

[1] Department of Computer Science, Universität der Bundeswehr München, Werner-Heisenberg-Weg 39, 85577 Neubiberg, Germany
{Elisa.Canzani,Ulrike.Lechner}@unibw.de

[2] Cybersecurity Research Lab, Airbus Aerospace and Defense, Willy-Messerschmidt-Straße 1, 85521 Ottobrunn, Germany
Helmut.Kaufmann@airbus.com

Abstract. To trigger disruptive cascading effects among Critical Infrastructures (CIs), advanced cyber attacks take advantage of dependences among organizations. CIs are highly interconnected due to services and products they deliver one another to guarantee correct operational processes in such complex system-of-systems. Consequently, proper countermeasures in case of threats to CIs must consider interdependencies between them. The strategic use of information systems to coordinate response efforts of CI operators at national and international levels is a major objective towards more resilient societies. As relevant contribution to the development of a cyber incident early warning system for CI operators, this paper presents a System Dynamics (SD) interdependency model based on critical services that different operators must provide to guarantee the correct functioning of a CI. We explain model requirements and characteristics, and demonstrate how it can be used to gain situational awareness in the context of European CIs.

Keywords: Critical Infrastructures · Interdependency modeling · System Dynamics · Early warning system · Incident response coordination

1 Introduction

Modern infrastructures have become increasingly interconnected due to the increasing dependence on Information Technology (IT) for business operations. However, existing security measures to prevent and protect a Critical Infrastructure (CI) from threats and cyber attacks do not usually cross the organization's boundaries [1]. Understanding complex dynamics of interdependencies to coordinate mitigation strategies and response efforts would help to prevent CI networks from potential catastrophic cascading effects [2]. In particular, information sharing has become essential in cyber defense to combat Advanced Persistent Threats (APTs) that leverage on such interdependencies to attack one specific CI by intruding multiple dependent CIs [3].

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 308–320, 2017.
https://doi.org/10.1007/978-3-319-71368-7_27

Surveys conducted by the European Union Agency for Network and Information Security (ENISA) reveal that, in Europe, a significant number of member states present a low level of maturity and lack of a structured approach regarding identification of CIs and establishment of coordination plans. As a first step towards unifying CI protection programs among the European member states, the ENISA has recently issued guidelines to identify CI assets and services [4]. On the basis of definitions and reference lists provided by the ENISA, we move one step further by proposing an operator-driven approach to model and analyse CI dependencies and related critical services.

To develop the interdependency model, we apply the block building methodology based on System Dynamics (SD) of Canzani [5]. This approach allows breaking down the overall system complexity by iteratively developing and assembling together blocks of models to replicate relevant dynamics of disruptions in interdependent CIs. In line with system-of-systems modeling principles [6], we extend the three building blocks developed in [5] to highlight the relevance of critical sectors, CIs, and services. Mainly, we adopt a perspective of CI operators and introduce three different types of dependencies: dependencies within a single CI, intra-sector interdependencies, and cross-sector interdependencies.

This work pulls together synergies of two emergent fields such as crisis management and cyber security research toward improving Critical Infrastructure Protection (CIP). We present an application of the crisis management approach of the NITIMesr project to scenarios of the ECOSSIAN project to contribute to the design of a real-time Early Warning System (EWS) and incident information analysis for gaining situational awareness in a European control system security network. With respect to the ECOSSIAN framework [7], we demonstrate how to develop and use the operator-driven SD model to support coordination and response of CI operators in case of an attack. The crucial role of our interdependency model is emphasized at large as well as with specific examples.

The paper is organized as follows. In Sect. 2, we contextualize the interdependency model within the ECOSSIAN framework for developing a pan-European early warning system. In accordance with ENISA guidelines, Sect. 3 introduces our operator-driven approach based on critical services and sectors. Main characteristics of the interdependency model and its implementation with SD tools are then discussed in Sects. 4 and 5 respectively. In Sect. 6, we give a small example of scenario generation using the SD model. Section 7 wraps up our work with concluding remarks and future research directions.

2 Early Warning and Incident Response System for Operators

At present, government agencies, private companies, and academic communities are engaged in the development of effective Early Warning Systems (EWSs) for CIP. An effective EWS should support prevention and mitigation actions in case of disruptive events by monitoring operations and sharing relevant incident information. The main goal is to identify impacts and cascading effects among CIs in time to permit an effective incident response that reduces or avoids potential breakdowns of CI networks. In this

context, the crucial role of information sharing between CI operators is clear: comprehensive knowledge of the current threat state of the networked system of CIs facilitates both detection of large-scale attacks and coordination of response strategies among stakeholders.

However, CIs usually adopt security measures that only make use of information collected from their own systems. For instance, a review of EWSs for the safeguard of public water supplies is [8]. Beyond securing one CI as independent system, insights for a network-based EWS that consider interdependent CIs are given [9].

In this paper, we specifically refer to the framework proposed by the ECOSSIAN project for the development of a real-time EWS and impact analysis for gaining situational awareness in a European control system security network. The ECOSSIAN ecosystem proposes a layered security approach with incident information sharing through Security Operations Centres (SOCs), which have specific responsibilities at operator level (O-SOC), national level (N-SOC), and European level (E-SOC). See [10] for more details on ECOSSIAN and its ecosystem. Figure 1 describes how the interdependency model extends the ECOSSIAN framework.

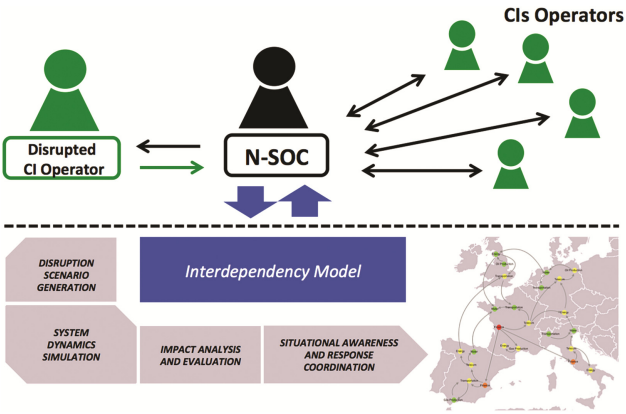


Fig. 1. Use of the interdependency model in the ECOSSIAN scenario (Color figure online)

The green arrow indicates that the disrupted operator (i.e. O-SOC according to the ECOSSIAN terminology) must immediately inform the national entity (N-SOC) about damages occurred in its own operational facilities. The CI operator is asked to characterise severity and time components of disruptive event and expected recovery. This data is used as input parameters for the interdependency model such that the N-SOC can generate the specific disruption scenario. Note that a list of general information about all CI operators is assumed to be available, so that the N-SOC can identify and characterize the disrupted operator when a warning is reported. The ECOSSIAN scenario suggests that once the model setting is completed, damage effects in the interdependent system of CIs are calculated and disruption impact analysis is conducted by the N-SOC to gain situational awareness of national CIs and all CIs over Europe (through the E-SOC). The final ECOSSIAN

attempt is to help the SOCs to timely coordinate mitigation actions and to establish recovery priorities among CI operators (black arrows in Fig. 1).

The simulation model that we present in this paper aims at capturing and analysing the dynamics of potential consequences in such comprehensive picture.

3 Identification of Critical Sector and Services

This section describes the approach and specific concepts we use in modeling on the basis of ENISA definitions [4]. In accordance with the ECOSSIAN objective to promote the use of information sharing to improve cybersecurity, the work of ENISA leverages on the role of communication networks to ensure the correct functioning of every CI. A cyber attack affecting these assets, commonly referred to as Critical Information Infrastructures (CIIs), could lead to large-scale cascading effects.

Figure 2 illustrates a classification of approaches to identify CI/CIIs assets. In approaches that do not rely on critical services, data network analysis of national infrastructures is required for mapping and protecting network components. However, the identification of all components that are critical to CI operations is costly and often prohibitive. Approaches dependent on critical services are based on consequences of impacts that a critical service disruption may have on the society. Also, we can differentiate such approaches depending on who has the leading role for the identification of critical services: government agencies (State-driven process) or CI operators (Operator-driven process).

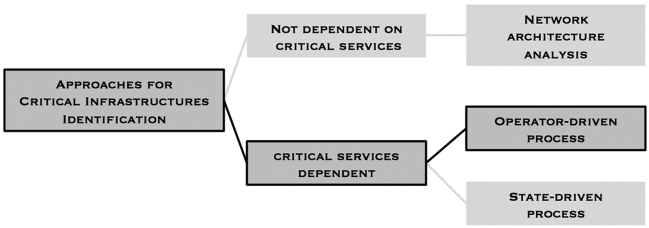


Fig. 2. Methodological approaches for CI identification

We build the interdependency model accounting for an operator-driven approach based on critical services and sectors. This choice fits the ECOSSIAN mission to provide a layered security approach with specific focus on operators and critical services that they deliver for a correct CI functioning. In fact, information sharing in this holistic EWS includes specific responsibilities of CI operators for coordination purposes (see Sect. 2).

In line with the ENISA, we distinguish between critical sector, critical infrastructure (ENISA also refers to it as "subsector"), and critical services as it is shown in Fig. 3 (cf. [4] for the complete list).

Critical Sector	Critical Infrastructure	Critical Services
Energy	Electricity	• Generation • Transmission/Distribution • Electricity Market
	Petroleum	• Extraction • Refinement • Transport • Storage
	Natural Gas	• Generation • Transport/Distribution • Storage
Transportation	Train Transport	• Railway Management • Train Transport Facilities
	Road Transport	• Road Network Maintenance • Bus/Tram Services
	Maritime Transport	• Shipping traffic Management • Ice-breaking Operations
	Aviation	• Airport operations • Air Navigation Services
Water	…	• …
	…	• …
…	…	• …

Fig. 3. Example of list of critical sectors, CIs and related critical services

This reference list helps to understand our complex scenario as system-of-systems. Each critical sector (e.g. Energy) corresponds to a group of CIs (e.g. Electricity, Petroleum, Natural Gas), which in turn are able (or not) to provide respective final services and products according to internal operations. The correct functioning of such operational processes depends on critical services provided by CI operators, which contribute to the complete chain value of each CI.

Thus, the classification in Fig. 3 is used as backbone structure of the interdependency model for assessing interdependencies within and across CIs.

4 Interdependency Modeling

4.1 Background and Related Work

After pioneering works that focus on qualitative aspects of the interdependency problem (e.g. [11]), the understanding of CIs as "system of systems" [6] and the "network of networks" approach [12] led to deeper quantitative investigations of dependencies within a CI and interdependencies across CIs. Adetoye et al. [13] propose an analytical framework for reasoning about CI dependencies. A review of modeling and simulation approaches to interdependent CI systems is in [14].

This research paper adopts the block building approach based on SD modeling of Canzani [5] to support CI operators in crisis management processes. The author distinguishes between two dimensions of system resilience: operational state and service level. The main criteria to identify interdependencies is that every CI produces commodities to satisfy the demand while needs products and services from other CIs in order to operate normally. We extend previous modeling work by a perspective of CI operators for the supply of critical services.

4.2 Model Structure Overview

In accordance with the ENISA classification (Fig. 3), we propose a layered structure based on critical sectors, critical infrastructures, and critical services. Sectors are groups of CIs whose assets, systems, and networks are considered so vital to health, safety, security, economic or social well-being of people and the disruption or destruction of which would have a significant impact in a member state as a result of the failure to maintain those functions [15]. In turn, each CI has an internal operational dynamics based on different critical services. Only if CI operators can adequately provide all critical services, the complete value chain of the CI is preserved. This structure is the basis to identify three types of dependency:

- **dependencies within a CI** (if critical services of a CI depend on the final product/service of the CI itself),
- **intra-sector interdependencies** (if two CIs belong to the same sector and a critical service of one of them depends on resources and final services of the other CI),
- **cross-sector interdependencies** (if two CIs belong to different sectors and a critical service of one of them depends on resources and final services of the other CI).

Figure 4 clarifies layered components of our interdependency model as well as the three types of dependencies and interdependencies listed above.

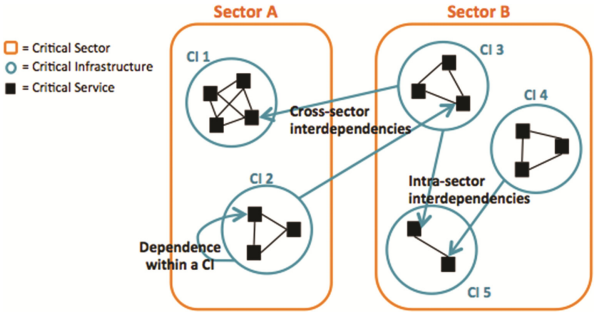


Fig. 4. Layered structure of the interdependency model

5 System Dynamics Implementation

Adopting the block building modeling approach of Canzani [5], this section demonstrates how to develop and implement our interdependency model with SD tools. The main intent is to qualitatively describe model characteristics keeping in mind a wider audience of CI experts and operators. Thus notation and level of detail is aimed at a broad range of researchers, and not limited to SD modelers. For more details on SD theory, we refer the reader to the seminal book of Sterman [16].

5.1 Disruption Characterization

The disruption is modelled as a pulse function starting at the time in which the disruption occurs (*Disruption Time*) and lasting a certain period (*Disruption Duration*). *Disruption Magnitude* varies from 0 to 10 as a result of the product of two input parameters that characterise the disruption by an operator perspective, i.e.

- *Damage Estimation,* which is a value from 0 to 10 given by the disrupted operator to assess magnitude of disruption effects in its own organization;
- *Operator Importance,* which is a factor that scales the disruption damage with respect to the importance of the disrupted operator in providing a specific critical service to the CI. It ranges from 0 to 1 according to the piece of market segment the operator owns among critical service providers.

Note that *Operator Importance* is part of the general information that must be available to the N-SOC at the moment of the warning report in the ECOSSIAN scenario (see Sect. 2). When reporting relevant incident information, the disrupted operator (O-SOC) should be able to assess *Expected Time to Recovery* and inform about eventual delays with which the recovery started (*Delay Recovery*). This is implemented with SD tools using a step function that considers time lags between *Disruption Time* and *Start Recovery* (i.e. the time in which response actions start).

Moreover, model capabilities allow accounting for human, environmental, economic and other impacts that influence cascade effects and magnitudes over time. See our recent work [17] for more details on disruption characterization.

5.2 Single Critical Infrastructure Dynamics

Canzani [5] describes the single CI with a system of differential equations that captures the dynamics of running, down and recovered operations over time. CI operational state changes according to principles of the SIRS epidemic model. Here, we consider two operational states of the CI, i.e. running and down operations, in a modified version of the SIS epidemic model. The goal is to have an emphasis on identification of criticality of critical services to understand the CI dynamics. Mathematical insights on epidemic modeling can be found in [18].

Given a general infrastructure i, we consider the stocks of *Running Operations* $OP^i_{run}(t)$ and *Down Operations* $OP^i_{down}(t)$, and we split the flow of operation breakdown into different flows $s^i_1(t), \ldots, s^i_n(t)$ corresponding to breakdown rates of critical services that must be provided to the CI for its correct operational functioning. All critical services are needed to ensure the normal operational state of the CI, but some of them are more critical than others. For instance, railway management and availability of train facilities are critical services of the Train Transport CI (see Fig. 3). If trains are out of services due to a cyber attack manipulating their control systems, operations to maintain the railway can be pursued anyway. If rail signal upgrades are hacked to cause crashes, trains cannot run safely. In both cases the final service of the Train Transport CI cannot be fully provided.

Therefore, we assign a *criticality factor* to each flow $s^i_j(t), j = 1, \ldots, n$, to assess criticality of each critical service with respect to other critical services of the CI. In the initial model setting, criticality factors are constant parameters $c^i_1, \ldots, c^i_n$ such that $c^i_1 + \ldots + c^i_n = 1$.

Let us now define a recovery rate $r^i(t)$ and the total number of CI operations n^i_{OP} (also denoted as maximum capability of the CI, see [5]). In line with epidemic models, we assume $OP^i_{run}(t) + OP^i_{down}(t) = n^i_{OP}$ at any time t. Mathematically we have:

$$\begin{cases} \dfrac{d}{dt} OP^i_{run}(t) = -\sum_{j=1}^{n} c^i_j s^i_j(t) \left(\dfrac{OP^i_{run}(t)}{n^i_{OP}} \right) + r^i(t) OP^i_{down} \\ \dfrac{d}{dt} OP^i_{down}(t) = \sum_{j=1}^{n} c^i_j s^i_j(t) \left(\dfrac{OP^i_{run}(t)}{n^i_{OP}} \right) - r^i(t) OP^i_{down} \end{cases} \tag{1}$$

As the dynamic behavior starts at the moment of time when the disruption occurs, breakdown rate of a general critical service $s^i_j(t)$ depends on disruptive events affecting a provider of that critical service as well as ability of other CIs to deliver services over time (i.e. interdependencies).

Further mathematical details about relationship among CI operations, capabilities, service provided, and average demand for such service are in [5].

5.3 Interdependencies Assessment

On the basis of the ENISA reference list in Fig. 3, we build the *connection matrix* (or *interdependency matrix*) to assess the three types of interdependencies defined in Sect. 4.2. As shown in Fig. 5, the matrix identifies CI services needed to each operational process (i.e. operator that provides that critical service to the CI). The objective is to disaggregate the interdependencies at the critical service level, so that specific links and relationships will emerge from the complex system of CIs.

The connection matrix can be *Boolean*, indicating whether or not there is a dependence; or *weighted* to quantify magnitudes of effects of a CI if another CI would be non-operational for a certain time period. Canzani [5] uses the results of a latest survey of CIs experts from several countries [19] to set weights on a scale of 0 to 5.

			Energy			Transportation				....		
SECTOR	CI	CRITICAL SERVICES	Electricity	Petroleum	Natural Gas	Aviation	Road	Train	Maritime	....	....	
Energy	Electricity	Generation										
		Distribution										
		Electricity Market										
	Petroleum	Extraction										
		Refinement										
		Transport										
		Storage										
	Natural Gas	Extraction										
		Transport/Distribution										
		Storage										
Transportation	Aviation	Air Navigation Services										
		Airport Operations										
	Road	Road Network Maintenance										
		Bus/Tram Services										
	Train	Railway Transport Services										
		Public Railway Maintenance										
	Maritime	Shipping Traffic Management										
		Ice-Breaking Operation										
....	....	....										
	....	....										
		....										

Fig. 5. Connection matrix to assess interdependencies among CIs based on critical services

6 Scenario Example

To give an idea of how the CIs interdependency model can be applied to ECOSSIAN scenarios with system dynamics tools, we demonstrate how to build the model using Vensim DDS software package (Ventana Systems, Inc.).

Figure 6 illustrates how building blocks are integrated together in a scenario with three critical infrastructures:

– Petroleum CI (within the Energy Sector),
– Electricity CI (within the Energy Sector),
– Train Transport CI (within the Transportation Sector).

Critical services are identified and respective breakdown rates modeled for each block representing the CIs (dotted blue boxes). The figure shows interdependencies of different types, e.g.:

- *Cross-sectors interdependencies*. Trains may need fuel or electric power to run and therefore final products and services of Petroleum and Electricity CIs are needed to train facilities, which is identified as a critical service of the Train Transport CI. Also, trains are needed to transport oil barrels.
- *Intra-sector interdependencies*. Oil resources are crucial for power plants in order to generate electricity. Within the Energy Sector, the functioning of the Petroleum CI is vital for generation service providers in the Electricity CI.

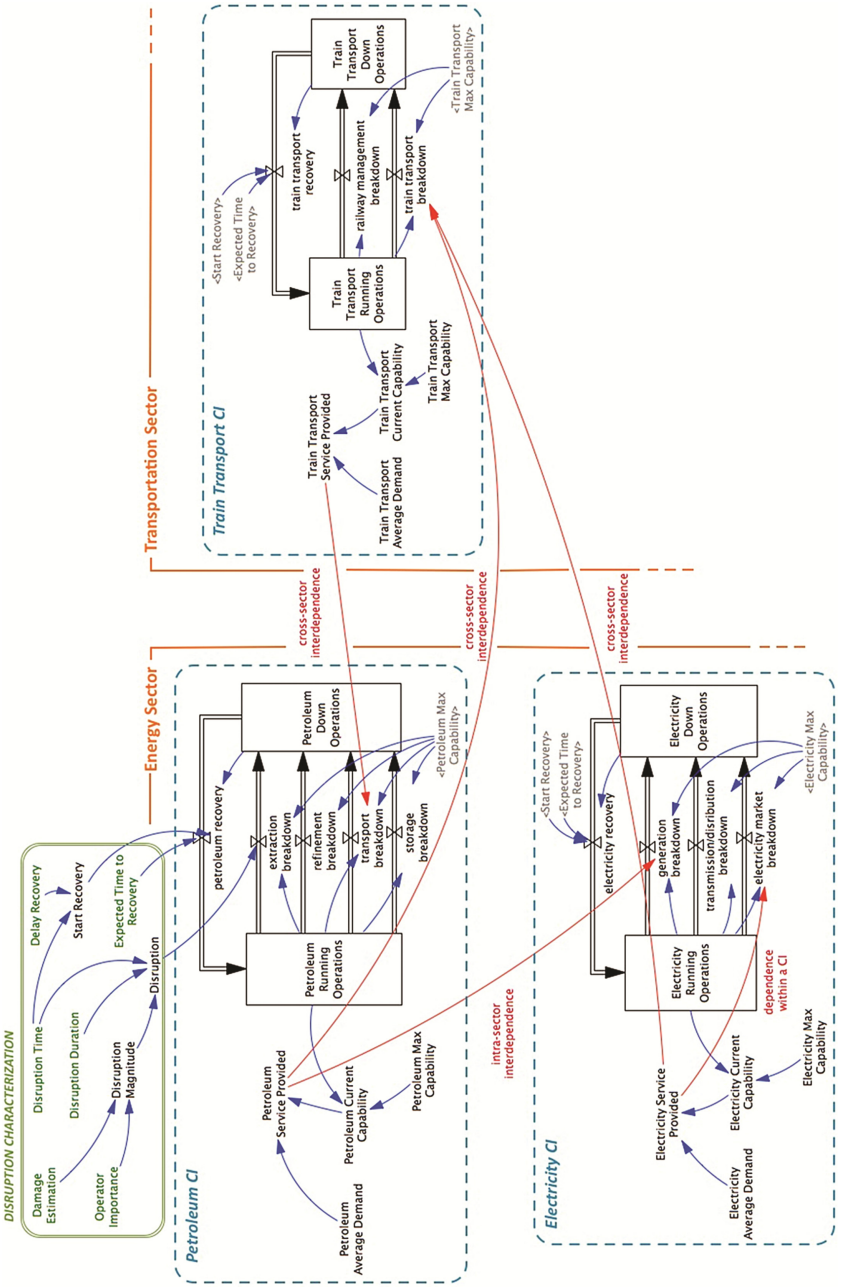


Fig. 6. Example of integrated SD building blocks (Color figure online)

- *Dependencies within a CI*. Price perturbations in the electricity market may occur if electricity cannot be provided; therefore a critical service of the Electricity CI depends on service breakdowns of the CI itself.

Our goal here is to show how to apply the interdependency model to support early warning system and coordination of response among CI operators. Incident information sharing mainly refers to the description of disruptive events. Therefore, we focus on the block of disruption characterization in Fig. 6 (double-line green box). The scenario describes a disruption of an operator providing petroleum extraction as critical service for the correct functioning of the Petroleum CI. In the ECOSSIAN scenario (see Fig. 1), the disrupted operator (O-SOC) must inform the national entity (N-SOC) about the moment of time when the disruption occurs, disruption duration, the expected time to recover and eventual time delays with which the recovery started. The O-SOC should also estimate the damages occurred in its operational facilities so that the N-SOC can assess impacts on other CIs based on the "a priori" operator importance respect to the petroleum extraction service.

The N-SOC uses incident information provided by the disrupted operator (green variables in Fig. 6) as input parameters of the interdependency model. The simulation-based impact analysis leads to a better understanding of national and European scenarios to support crisis management processes of coordination and response among CI operators.

Note that the SD model in Fig. 6 is not exhaustive, and other interdependencies can be identified to characterize the network of CIs. We remark that this example also serves to clarify the analytical description of variables given in Sect. 5.

7 Discussion and Concluding Remarks

This work contributes to the relevant fields of crisis management and cyber security research by proposing an operator-driven approach to understand interdependencies among Critical Infrastructures (CIs). A System Dynamics (SD) model is developed to capture the dynamics of CI operations and critical services by a perspective of CI operators. We demonstrate how to embed the interdependency model in a cyber incident analysis system to improve situational awareness and support coordination of CI operators at national and European levels.

Our aim is to contribute to the improvement of the current state of CI protection plans in Europe to be ready for future threats. The ECOSSIAN scenarios and use cases [7] developed are discussed and extended through the use of the interdependency model to show that a proper understanding of CI interdependencies is key to the design of an early warning system for CI operators. In modeling we build on the ENISA guidelines for the identification of critical services and assets [4]. We use the block building methodology of [5] to structure the modeling process. This approach facilitates further model extensions to emphasize different aspects of cyber security of critical infrastructures according to requirements of the EWS and/or availability of data. For instance, in [20] we have developed a building block to capture cyber attacker-defender dynamics using game theory.

We took publicly available data from ENISA survey results of CI operators. Naturally, more data of CI experts and operators would help to better quantify magnitudes of dynamic dependencies among CIs. Next steps in our work include validation of model and assumptions and an attempt to use our model to solicit relevant data from operators. Of our interest is also to extend model features by considering human, economic, environmental and other impact factors to characterize the disruption and assess interdependencies in the final ECOSSIAN demonstrator.

Acknowledgments. Elisa Canzani PhD research is funded within the Marie Curie Research & Innovation Actions by the European Union FP7/2007-2013, NITIMesr (317382). This work is partly funded by the European Union FP7 project ECOSSIAN (607577).

References

1. Luiijf, H.A.M., Besseling, K., Spoelstra, M., de Graaf, P.: Ten national cyber security strategies: a comparison. In: Bologna, S., Hämmerli, B., Gritzalis, D., Wolthusen, S. (eds.) CRITIS 2011. LNCS, vol. 6983, pp. 1–17. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-41476-3_1
2. Buldyrev, S.V., Parshani, R., Paul, G., Stanley, H.E., Havlin, S.: Catastrophic cascade of failures in interdependent networks. Nature **464**, 1025–1028 (2010)
3. Juuso, A., Takanen, A.: Proactive Cyber Security: Stay Ahead of Advanced Persistent Threats (APTs). Codenomicon WP (2012)
4. Mattioli, R., Levy-Benchton, C.: Methodologies for the identification of Critical Information Infrastructure assets and services (2014)
5. Canzani, E.: Modeling dynamics of disruptive events for impact analysis in networked critical infrastructures. In: 13th International Conference on Information Systems for Crisis Response and Management, ISCRAM (2016)
6. Eusgeld, I., Nan, C., Dietz, S.: System-of-systems approach for interdependent critical infrastructures. Reliab. Eng. Syst. Saf. **96**, 679–686 (2011)
7. Settanni, G., Skopik, F., Shovgenya, Y., Fiedler, R., Kaufmann, H., Gebhardt, T., Ponchel, C.: A blueprint for a pan-European cyber incident analysis system. In: 3rd International Symposium for ICS and SCADA Cyber Security Research 2015, pp. 84–88 (2015)
8. Hasan, J., States, S., Deininger, R.: Safeguarding the security of public water supplies using early warning systems: a brief review. J. Contemp. Water Res. Educ. **129**, 27–33 (2004)
9. Bsufka, K., Kroll-Peters, O., Albayrak, S.: Intelligent network-based early warning systems. In: Lopez, J. (ed.) CRITIS 2006. LNCS, vol. 4347, pp. 103–111. Springer, Heidelberg (2006). https://doi.org/10.1007/11962977_9
10. Kaufmann, H., Hutter, R., Skopik, F., Mantere, M.: A structural design for a pan-European early warning system for critical infrastructures. Elektrotechnik und Informationstechnik **132**, 117–121 (2014). Springer, Vienna
11. Rinaldi, S.M., Peerenboom, J.P., Kelly, T.K.: Identifying, understanding, and analyzing critical infrastructure interdependencies. IEEE Control Syst. Mag. **21**, 11–25 (2001)
12. Gao, J., Li, D., Havlin, S.: From a single network to a network of networks. Natl. Sci. Rev. **1**, 346–356 (2014)

13. Adetoye, Adedayo O., Goldsmith, M., Creese, S.: Analysis of dependencies in critical infrastructures. In: Bologna, S., Hämmerli, B., Gritzalis, D., Wolthusen, S. (eds.) CRITIS 2011. LNCS, vol. 6983, pp. 18–29. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-41476-3_2
14. Ouyang, M.: Review on modeling and simulation of interdependent critical infrastructure systems. Reliab. Eng. Syst. Saf. **121**, 43–60 (2014)
15. 2008/114/EC-Council Directive: Identification and designation of European Critical Infrastructures and the assessment of the need to improve their protection. Off. J. Eur. Union **51**, 75–82 (2008)
16. Sterman, J.D.: Business Dynamics: Systems Thinking and Modeling for a Complex World. Irwin/McGraw-Hill, Boston (2000)
17. Canzani, E., Kaufmann, H., Lechner, U.: Characterizing disruptive events to model cascade failures in critical infrastructures. In: 4th International Symposium for ICS and SCADA Cyber Security Research 2016 (2016)
18. Canzani, E., Lechner, U.: Insights from modeling epidemics of infectious diseases – a literature review. In: 12th International Conference on Information Systems for Crisis Response and Management, ISCRAM (2015)
19. Laugé, A., Hernantes, J., Sarriegi, J.M.: Critical infrastructure dependencies: a holistic, dynamic and quantitative approach. Int. J. Crit. Infrastruct. Prot. **8**, 16–23 (2015)
20. Canzani, E., Pickl, S.: Cyber epidemics: modeling attacker-defender dynamics in critical infrastructure systems. In: 7th International Conference on Applied Human Factors and Ergonomics, AHFE (2016)

Domain Specific Stateful Filtering with Worst-Case Bandwidth

Maxime Puys(✉), Jean-Louis Roch, and Marie-Laure Potet

Verimag, University Grenoble Alpes/Grenoble-INP, Gières, France
{maxime.puys,jean-louis.roch,marie-laure.potet}@univ-grenoble-alpes.fr

Abstract. Industrial systems are publicly the target of cyberattacks since Stuxnet. Nowadays they are increasingly communicating over insecure media such as Internet. Due to their interaction with the real world, it is crucial to ensure their security. In this paper, we propose a domain specific stateful filtering that keeps track of the value of predetermined variables. Such filter allows to express rules depending on the context of the system. Moreover, it must guarantee bounded memory and execution time to be resilient against malicious adversaries. Our approach is illustrated on an example.

1 Introduction

Industrial systems also called SCADA (*Supervisory Control And Data Acquisition*) are the target of cyberattacks since the Stuxnet worm [1] in 2010. Nowadays, these systems control nuclear power plants, water purification or power distribution. Due to the criticality of their interaction with the real world, they can potentially be really harmful for humans and environment. The frequency of attacks against these systems is increasing to become one of the priorities for government agencies, *e.g.:* [2] from the French *Agence Nationale de la Sécurité des Systèmes d'Information* (ANSSI).

State-of-the-art. To face such adversaries, industrial systems can be protected by intrusion detection systems [3–6] which will only detect the attack and do not circumvent it. Intrusion protection systems [7,8] also exist and are able to block a malicious message when it arrives. Those kind of filters are usually *stateless*, meaning that the legitimacy of a message is only based on the message itself but not on the context. However, attacks may occur because a sequence of messages is received in a certain order or in a certain amount of time, each message being legitimate on its own. Such attack has been demonstrated through the Aurora project [9], lead by the US National Idaho Laboratory in 2007 (and classified until 2014). In order to test a diesel generator against cyberattacks, researchers rapidly sent opening and closing commands to circuit breakers. The frequency

This work has been partially supported by the LabEx PERSYVAL-Lab (ANR-11-LABX-0025) and the project *PIA ARAMIS (P3342-146798).*

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 321–327, 2017.
https://doi.org/10.1007/978-3-319-71368-7_28

of orders being to high, it caused the generator to explode. Electrical disconnectors also require to be managed by commands in a precise order. If any electric current runs through a disconnector while it is manipulated, an electric arc will appear, harming humans around and damaging equipment. To answer this problematic, stateful filtering mechanisms were first proposed by Schneider in 2000 [10]. Those contributions lead to many researches on *runtime-enforcement* to ensure at execution time that a system meets a desirable behavior and circumvent property violations. In 2010, Falcone *et al.* [11] proposed a detailed survey on this topic. In 2014, Chen *et al.* [12] detailed a similar monitoring technique applied to SCADA networks. However, their approach seems limited to the MODBUS protocol. Finally in 2015, Stergiopoulos *et al.* [13] described a method for predicting failures in industrial software source codes.

Contributions: We propose a protocol-independent language to describe a stateful type of domain specific filtering. Such filter is able to keep track of the value of predetermined variables. While filtering messages, the values of some variables are saved when they go through. This is a tedious task since the filter must be the single point of passage of all commands to not miss any. However, having a single point of passage for commands also means a single point of failure. Thus, to be resilient against malicious adversaries, we designed our filtering process to guarantee worst-case bandwidth and memory.

Outline: First, Sect. 2 explains more deeply stateful filtering and its pros and cons. Then Sect. 3 describes our filtering model and Sect. 4 illustrates it on an example. Finally, Sect. 5 concludes.

2 Classical Stateful Filtering

In this Section, we discuss what is stateful filtering and its shortcomings. Stateful filtering consists in keeping track of the value of predetermined variables of servers. The filter saves their values when they go through. As we said in Sect. 1 this supposes the filter to be the single point of passage of all messages. It implies that the filter must be hardened to resist against attacks. It also requires it to run in bounded memory and execution time to not delay real time message or overfill the memory of the filter when processing a memory-worst-case message. Moreover, no decision can be taken for a variable if it has not yet been seen before. For this sake, one might want to use three values logic such as Kleene's logic. This also holds if the server can update variables on his own (such as temperature, pressure, etc.) and they are not read frequently enough. Three values logics introduce a value neither true or false, called *unknown* or *irrelevant* and extend classic logic operators to handle such value. Thus a default policy is needed when the filter is not able to take a decision.

Two major concerns in filtering are (1) the time intervals between successive messages and (2) the ordering of messages. As the language we present in Sect. 3 does not rely on the time between messages, we are not concern by the first one. The second is obviously important since two different message orderings may

lead in two different filtering decisions. Thus as the filter handles messages in the order they arrive, it is crucial that client and servers communicating have deterministic behavior when ordering messages (which they shall do since this matter particularly applies to industrial communications).

Attacker model. We consider any client-side attacker who has access to the rules configured for the filter and their implementations but cannot change any of them. Such attacker is able to intercept, modify, replay legitimate traffic or forge his own messages. The attacker is considered as any client sending (possibly malicious) commands to a server situated on the other side of the filter. Thus every client including the attacker has to send commands complying with the rules configured.

Filtering and Safety properties. For each command message received, the filter decides whether to accept or reject it, based on its state. This decision has to be computed in statically bounded time and memory space. Only accepted command are transmit to the server that returns an acknowledgment; rejected commands are logged and the corresponding input channel is closed (until a reset). The filter behaves as a classical safety run time monitor. Following [14], a property is a set of finite or infinite traces; a safety property P is a behavioral property which, once violated, cannot be satisfied anymore; thus P is prefix-closed: if $w.u \in P$, then $w \in P$. The requirement to ensure safety property in bounded time and memory space is equivalent for the filter to implement a finite state automaton. For the sake of simplicity and without restriction, this automaton can be defined by a finite number of state variables with values in finite domains and a function transition ϕ. ϕ is a finite set of pairwise exclusive Boolean conditions C_i, each related to an action A_i (atomic update of state variables). The Boolean conditions are evaluated from both the input command and the state variables value. Either none is verified and the command is rejected; or exactly one condition C_i is verified and the command is accepted and its corresponding action A_i is performed before checking the next input command. Such rule system is usually known as *Event-Condition-Action* (EC) and XACML is an example [15].

3 Towards SCADA Specific Filtering

In this section, we explain how we restrict general stateful mechanisms explained in Sect. 2 in order to guarantee a worst-case bandwidth. The filters manages local state variables (acting as local copy of server variables) and rules.

Server variables: Variables present on a server and used to define safety property are known by filter where they are matched to local state variables. Thus a variable represented by a numerical identifier is associated to a server (associated to a protocol), a data type and the path on the server to access it (*e.g.:* a MODBUS address or an OPC-UA node). Variables can also have a sequence of

dimensions (*e.g.:* the length of an array or the dimensions of a matrix). Their definition is shown in Listing 1.

```
# A MODBUS server
Declare Server 1 Protocol Modbus Addr 10.0.0.1 Port 502
# A MODBUS coil (read/write Boolean)
Declare Variable 1 Server 1 Type Boolean Addr coils :0x1000
# An OPC-UA server
Declare Server 2 Protocol OpcUa Addr 10.0.0.2 Port 48010
# An OPC-UA unsigned integer 5× 10 matrix
Declare Variable 2 Server 2 Type UInt32 Addr numeric:5000 Dims 5 10
```

Listing 1. Variable definition example

Local state variable: some commands on a server variable (especially write requests or read results) provide information of the variable value; the *LocalVal* declaration enforces the filter to store this value in a local state variable that acts as a delayed copy of the server variable. Yet, when such a command is accepted, the default action is to update the value of the state variable. To prevent space overhead in case of multidimensional variables, this only applies to one cell; we use the *Index* keyword followed by corresponding valid array keys to obtain a scalar value. Such constraint can be lifted if and only if the size and dimensions of a variable cannot be modified once set. The value of a local variable shall be updated when a message containing the value goes through the filter. In a traditional ECA rule system, updating a local variable should be specified as actions to do when a condition is met. In the case of SCADA filtering, we can easily keep such action implicit due to the restricted number of event able to update local variables (it mainly applies to read responses and write requests). Moreover, updates on write requests must be reversible since the request can possibly be rejected by the server. An example of the definition of local variables is shown in Listing 2.

```
# A local variable on a MODBUS coil
Declare LocalVal 1 Variable 1
# A local var. on a cell of an OPC-UA unsigned integer 5× 10 matrix
Declare LocalVal 2 Variable 2 Index 3 4
```

Listing 2. Local variable definition example

Rules: Finally, rules can be set on variables using the previously declared local variables: conditions are evaluated from the local values of state variables; actions implicitly update those values. They can target either a whole variable or a subrange when multidimensional. They take the form of Boolean functions taking two arguments, separated by AND and OR operators. These functions implement Boolean conditions such as equality, integer relations, etc. Arguments of these predicates can either be: (i) constant numbers, (ii) *NewVal* designating the value to be written in a write request or (iii) *LocalVal* designating a previously defined local variable by its identifier. A rule can be either an assertion that will block a

message when violated or a warning that will authorize the message but log the violation for later event analysis. An example of the definition of rules is shown in Listing 3.

```
# Variable 1 should never been set to its current value
# (e.g.: opening a currently opened circuit breaker)
Declare Rule Variable 1 Assert NotEqual(NewVal, LocalVal[1])

# The first three rows of variable 2 must remain between 0 and 100.
Declare Rule Variable 2 Range 0-5 0-2 Warning \
    GreaterThan(NewVal,0) AND LessThan(NewVal,100)
```

Listing 3. Rule definition example

To ensure constant processing time and memory, both conditions and actions have to be processed in constant time. In our language, all conditions are Boolean conditions that can be verified in $\mathcal{O}(1)$ complexity due to the fact that arguments are restricted scalar values (constants, local variables, etc.). Moreover, actions are limited to: (i) either block or transmit the message (ii) log information, (iii) update a local variable and all of them also are in constant time. Thus processing one command only depends on the number of rules. In the worst case, a message would be checked against all predicates (for example in the case of a legitimate message). Thus if we associate a constant processing time τ_i to each predicate P_i appearing n_i times total in all the rules, we can compute the worst case processing time T of a message as: $T = \sum \tau_i n_i$.

4 Use-Case Example: An Electrical Disconnector

To illustrate our stateful filtering process, we propose the following simple example. An electrical disconnector D separates three electrical networks such as networks 2 and 3 are connected to the same input of D. As we told in Sect. 1, a disconnector cannot be manipulated while current is passing to avoid the creation of an electric arc. To ensure safety, three circuit breakers B_1, B_2 and B_3 are placed between D and each electrical network. Figure 1 describes this setup.

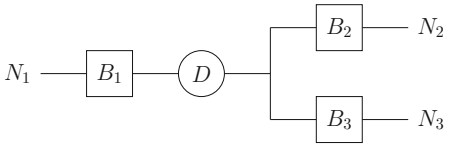


Fig. 1. Example infrastructure

Within a MODBUS server, D, B_1, B_2 and B_3 can be represented as coils (i.e.: read/write Booleans) with opened state represented by *False*. In this example, D can be manipulated if and only if either B_1 is opened or if both B_2 and B_3 are open. Thus the configuration presented in Listing 4 is enough to describe this rule. Note that in the rule definition, the AND operator has priority on the OR operator.

```
Declare Server 1 Protocol Modbus Addr 10.0.0.1 Port 502

Declare Variable 1 Server 1 Type Boolean Addr coils :0x1001 # B1
Declare Variable 2 Server 1 Type Boolean Addr coils :0x1002 # B2
Declare Variable 3 Server 1 Type Boolean Addr coils :0x1003 # B3
Declare Variable 4 Server 1 Type Boolean Addr coils :0x1004 # D

Declare LocalVal 1 Variable 1 # Local variable on B1
Declare LocalVal 2 Variable 2 # Local variable on B2
Declare LocalVal 3 Variable 3 # Local variable on B3

Declare Rule Variable 4 Assert    \  # Rule on variable 4 = D
    Equal(LocalVal[1], False) OR \  # Using local variable on B1, B2, B3
    Equal(LocalVal[2], False) AND Equal(LocalVal[3], False)
```

Listing 4. Example configuration

Thus, any sequence of messages violating the rule will be blocked ensuring the safety of the disconnector.

5 Conclusion

In this paper we present a language to describe a stateful type of domain specific filtering able to keep track of the value of predetermined variables. It guarantees bounded memory space and execution time to be resilient against malicious adversaries since processing one command only depends on the number of rules and memory to store monitor is controlled by only monitoring scalar variables or cells. In the future, we plan on extending the Boolean predicates to handle more complex arithmetic such has "*Equal(2*NewVal + 1, LocalVal[1]**2)*". Such verification are still performed in constant time since we are only evaluating the expression with concrete values. We would also be able to specify rules to avoid *Denial-of-service*. Such rules would limit the number of access to a certain variable within a period of time (*e.g.:* no more than 10 Read commands per minute) while keeping our bounded time and memory properties.

References

1. Langner, R.: Stuxnet: dissecting a cyberwarfare weapon. IEEE Secur. Priv. **9**(3), 49–51 (2011)
2. ANSSI. Managing cybersecurity for ICS, June 2012
3. Verba, J., Milvich, M.: Idaho national laboratory supervisory control and data acquisition intrusion detection system (scada ids). In: THS 2008 (2008)
4. Paxson, V.: Bro: a system for detecting network intruders in real-time. Comput. Netw. **31**(23), 2435–2463 (1999)
5. OISF. Suricata: Open source ids / ips / nsm engine, April 2016. http://suricata-ids.org/
6. Snort Team. Snort: Open source network intrusion prevention system, April 2016 https://www.snort.org

7. EDF R&D SINETICS. Dispositif d'échange sécurisé d'informations sans interconnexion réseau. Agence nationale de la sécurité des systèmes d'information, April 2010
8. SECLAB-FR. Dz-network. Agence nationale de la sécurité des systèmes d'information, June 2014
9. United States Department of Homeland Security. Foia response documents, July 2014. http://s3.documentcloud.org/documents/1212530/14f00304-documents.pdf
10. Schneider, F.B.: Enforceable security policies. ACM Trans. Inf. Syst. Secur. (TISSEC) **3**(1), 30–50 (2000)
11. Falcone, Y., Fernandez, J.-C., Mounier, L.: What can you verify and enforce at runtime? Technical report TR-2010-5, Verimag Research Report (2010)
12. Chen, Q., Abdelwahed, S.: A model-based approach to self-protection in scada systems. In: IWFC 2014, Philadelphia, PA, June 2014
13. Stergiopoulos, G., Theocharidou, M., Gritzalis, D.: Using logical error detection in software controlling remote-terminal units to predict critical information infrastructures failures. In: Tryfonas, T., Askoxylakis, I. (eds.) HAS 2015. LNCS, vol. 9190, pp. 672–683. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-20376-8_60
14. Roşu, G.: On safety properties and their monitoring. Sci. Ann. Comput. Sci. **22**(2), 327–365 (2012)
15. Lorch, M., Proctor, S., Lepro, R., Kafura, D., Shah, S.: First experiences using XACML for access control in distributed systems. In: XML Security 2003 (2003)

Securing SCADA Critical Network Against Internal and External Threats

Short Paper

Mounia El Anbal[1,2(✉)], Anas Abou El Kalam[3], Siham Benhadou[1], Fouad Moutaouakkil[1], and Hicham Medromi[1]

[1] Systems architectures Team, Hassan II University, ENSEM Casablanca, Casablanca, Morocco
mounia.el1@gmail.com,
benhadou.siham@gmail.com, fmoutaouakkil@hotmail.com,
hmedromi@yahoo.fr
[2] IPI Paris, IGS Group, Paris, France
elkalam@hotmail.fr
[3] ENSA - UCA, Marrakesh, Morocco

Abstract. Supervisory control and data acquisition systems (SCADA) constitute the sensitive part of the critical infrastructures. Any successful malicious incident could cause material, human and economic damages. Thus, the security of the SCADA networks became an emergency requirement to keep the continuity of services against hostile and cyber terrorist security risks. Several studies were conducted to secure SCADA networks against internal or external threats. In this paper, we focused on protection against both internal and external threats by adopting security mechanisms as access control, availability, authentication and integrity using a secure communication protocol ModbusSec and an intelligent firewall . We adopt also the self-healing and the intrusion tolerance techniques so in case of an intrusion in the system; it will have no impact on the continuity of service and the network safety.

Keywords: SCADA · Threat · Malicious · Abnormal · Authentication Availability · Access control · Integrity · Protection · Intrusion tolerance Self-healing

1 Introduction

Supervisory control and data acquisition systems provide an automated process for gathering real-time data, controlling industrial processes, and monitoring physically dispersed industrial equipment. Critical infrastructure (CI) such as utility companies and various industries have used industrial automation systems for decades to automate natural gas, hydro, water, nuclear, and manufacturing facilities.

Consisting of sensors, actuators, and control software, SCADA networks automate industrial processes while providing real time data to human operators. However, they are faced to several attack vectors that may cause software, hardware, human and economic

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 328–339, 2017.
https://doi.org/10.1007/978-3-319-71368-7_29

damages. Then, according to several research projects, the various features of the different levels that make up a SCADA system, whether it was the field, command, or control levels, are exposed to several cybercriminal threats. Moreover, the communication protocols between these levels could also be vulnerable. Furthermore, this is often due to the lack of security control, which was not considered during the SCADA systems conception.

Our work purpose is to provide a new approach that does not change either the internal architecture or the SCADA devices configuration in a critical network. It however aims to protect Critical Infrastructures (CI) by detecting hostile activities led by internal nodes. As well as protection against the intrusions by intrusion-tolerant devices with the possibility of periodic or forced recovery in case of the detection of malicious activity coming from a WAN, competing network or devices that are mutually monitored.

The SCADA architecture consists of three levels: the field level, the control level and the supervision level (Fig. 1). Often approaches offer the protection measure of one or two levels and leave others vulnerable. Therefore, the protection is incomplete, and an in-depth protection seems necessary for critical infrastructures (CI).

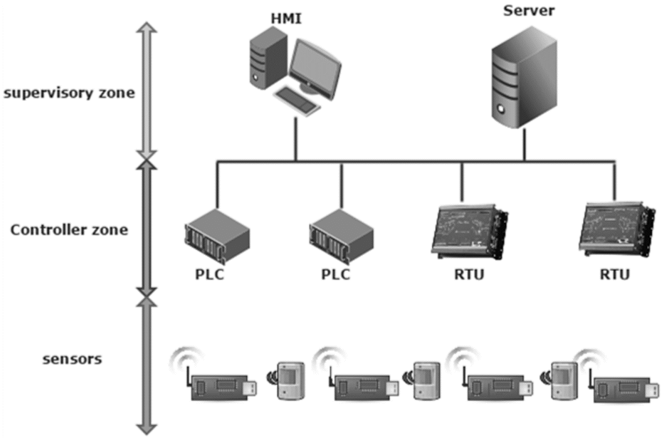


Fig. 1. An example of SCADA network

In the next section, we present existing approaches for protecting CI against internal as well as external threats, then we present mechanisms used for ensuring CI self-healing. Afterwards, in Sect. 3, we propose a new architecture and mechanisms for an in-depth protection of the Critical Infrastructures.

2 Related Work

2.1 Protecting Against Internal Threats

In [6], the authors proposed an approach that detects malicious activities within a SCADA system using the data collection, log management and inter neighbors monitoring to filter information from logs in order to detect internal suspicious nodes. This

approach only concerns the sensors zone (field level). However, other levels still vulnerable to malicious threats and they need security and recovery measure.

Besides that, a heavy focus was put on architecture protection but the communication protocols can mitigate common SCADA protocol attack vectors such as message spoofing, modification, replay attacks; denial of service vulnerabilities and man-in-the-middle attacks. In [8], the author states that currently there is no Modbus feasible secure implementation exists. Furthermore, Modbus dependence on TCP as a transport mechanism has many inherent security risks (e.g. increased susceptibility to denial of service attacks) which has several consequences on the continuity of services in a SCADA network. As a solution, he proposed a an alternative of Modbus TCP protocol which is the ModbusSec. ModbusSec uses the SCTP (stream transmission control protocol) with HMAC (Hash of codes for Message Authentication) to secure Modbus transactions by ensuring the availability, integrity and authenticity of Modbus messages.

In [4], the authors present a technique for detecting the abnormal behavior on Modbus/TCP transactions that is performed by learning the traffic patterns between supervisory and control systems. Therefore, they proceed for detecting the abnormal behavior beyond normal traffic patterns by using normal traffic pattern learning. The detection of abnormal behavior in the control levels complete the approach of the detection of malicious activities in the devices field. However, the Modbus protocol have no inherent security controls. Therefore, it can be easily attacked [17]. For example, attackers can easily collect inside information of control systems in a critical infrastructure and send malicious commands through a simple packet manipulation [18].

Nevertheless, they ensure the detection of internal malicious behaviors. The external threat could be detected using the Intrusion tolerant devices presented in [7]. The authors of [7] point of view is that interference and attacks start at the level of the macroscopic data flows between SCADA systems, internal and external networks. Protecting these data flows using proper access control policies is thus a fundamental step towards security. Furthermore, highly dependable devices, the CRUCIAL Information Switches (CIS), which are designed with a range of different levels of intrusion-tolerance and self-healing to serve distinct resilience requirements, must enforce these policies.

2.2 Protecting Against External Threats

The Intrusion-Tolerant CIS with Proactive and Reactive Recovery (ITCIS-PRR) [7] is replicated in a set of computers in order to mask intrusions in some of its components. More precisely, the CIS access control functionality is replicated across $2f + 1$ machines to tolerate up to f malicious or accidental faults. A replica in which there is an intrusion or a crash is said to be faulty. It has a self-healing capability. This capability is implemented by recovering (or rejuvenating) periodically each replica to remove the effects of any intrusion that might have occurred. Besides the periodic rejuvenation of replicas, each replica monitors the behavior of all others (for example, by looking at the voting decisions and the packets forwarded by the leader). If a set of replicas discover that another one is misbehaving, they force the recovery of this replica.

In other work, The SCADA Intelligent Gateway (SIG) [12] was proposed for securing SCADA systems, which are inherently insecure because SCADA was not

designed to be connected to the internet. As SCADA devices are now internet-connected, it is paramount that the system is protected against external attacks that could lead to serious inconveniences and safety issues.

SCADA inelegant gateway (SIG) is able to prevent DOS (Denial of Service) attacks to the SCADA system. The goal of the SIG is to create an intelligent security appliance that can be dropped into multiple locations on an existing SCADA network. Conceptually it consists of a Master SIG and one or more Perimeter SIGs. Each SIG both Master and Perimeter) monitors SCADA traffic on the network segment for which it is responsible.

The SIG would protect SCADA systems by only allowing recognized behavior on the system by verifying the origin of traffic flowing through the system and by capturing and buffering suspect traffic found on the system, along with testing their bandwidth allowance. These afore-mentioned functions support integrity and availability. The SIG also protects confidentiality by encrypting traffic in systems, which do not use encryption.

2.3 Self-healing Technique

Ghosh et al. provide a definition of self-healing systems: "… a self-healing system should recover from the abnormal (or "unhealthy") state and return to the normative ("healthy") state, and function as it was prior to disruption" [11].

The ITCIS-PRR applies the self-healing capability in the extremity of SCADA networks. However, inside the network-critical, we must restore any incident by using the self-healing loop depicted in the Fig. 2 loop with the data-flow among the three stages and the environmental interfaces:

- Detecting: Filters any suspicious status information received from samples and reports detected degradations to diagnosis.
- Diagnosing: Includes root cause analysis and calculates an appropriate recovery plan with the help of a policy base.
- Recovery: Carefully applies the planned adaptations meeting the constraints of the system capabilities and avoids any unpredictable side effects.

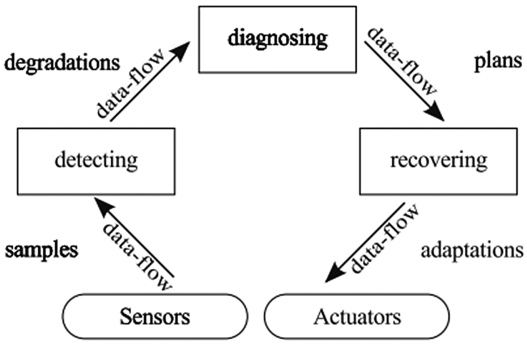


Fig. 2. Staged loop of self-healing

3 Proposed Secure SCADA Network Against Internal and External Threats

In our architecture, the first protection mechanism is to place (at the border of a SCADA system) an intelligent firewall implementing the security policy rules such as those proposed in [13–15]. The firewall should be intelligent and take into account tolerance and self-healing mechanisms. The goal is to protect infrastructure against external threats from either the Internet or intranet. The second protection mechanism is a hybrid approach to protect different levels the SCADA network by detection abnormal traffic and activities and response to a many type of malicious internal threats.

3.1 Proposed Intelligent Firewall to Secure Against External Threats

Our intelligent firewalls can be used in a redundant way, enforcing the access control policies in different points of the network. The concept is akin to use firewalls to protect hosts instead of only network borders, and is especially useful for critical information infrastructures given their complexity and criticality.

Also for the approach [6], the HMI nodes are not protected from external attacks in our approach we propose to apply the masking technique by using redundancy. The application servers and the HMI are sufficient to ensure a satisfactory response time on a nominal request rate in a given redundancy scheme (Fig. 3). The servers are isolated from the Internet by "the intelligent firewall", itself composed of a preliminary access control firewall and diverse computers, but driven by a specifically developed software. The requests from the Internet, filtered by the firewall using PolyOrBAC access control technique [13–15] are taken into account by one of the mandatories who plays leading role. The leader distributes the requests from the Internet to the servers and check their answers before sending them to the request sender. Rescue mandatory monitors the operation of the leader and observes network firewalls/mandatary and mandatary/servers. In case of the leader failure, elect among themselves a new leader. Every mandatory computer contains an IDS (intrusion detection system). The mandatories also

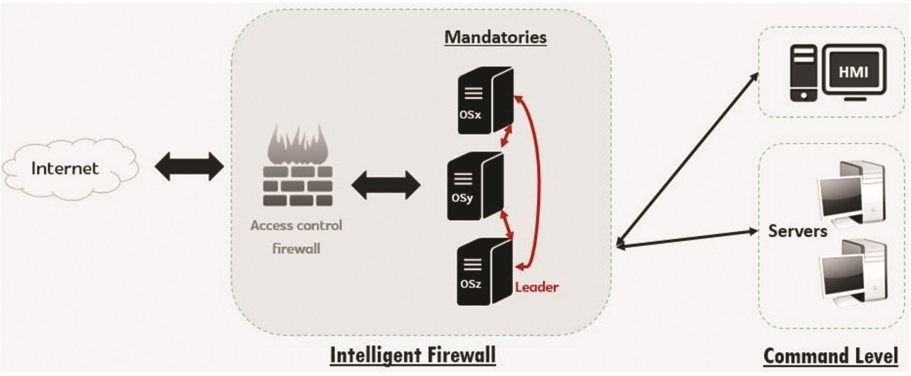


Fig. 3. Intelligent firewall architecture

handle alarms from an intrusion detection sensors installed on both supervisory and mandatories networks. This technique allows us to add the intrusion tolerance, by the masking technique applied in our intelligent firewall, to the SCADA network. Therefore, an intrusion, into a part of the system, has no impact in safety.

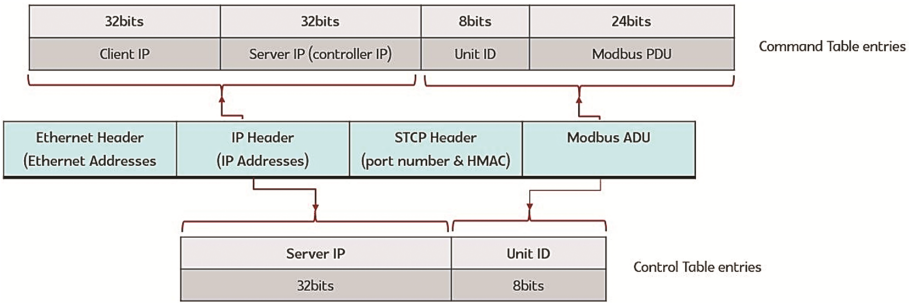


Fig. 4. Communication pattern repository

3.2 Proposed Approach to Secure Against Internal Threats

For the internal threats, we propose a hybrid approach that combines several security mechanisms. In particular, we propose using the ModbusSec packets supervisory as instead of the Modbus TCP supervisory as shown in the Fig. 5. The mean goal is to implement the supervision of the ModbusSec protocol, to detect anomalies, and to generate alerts in case of grant (Fig. 6). The use of ModbusSec who has as protocol of the transport layer the SCTP that guarantee a reliable communication and ensure a high level of availability. The availability is guarantee by the multi-homing technique which

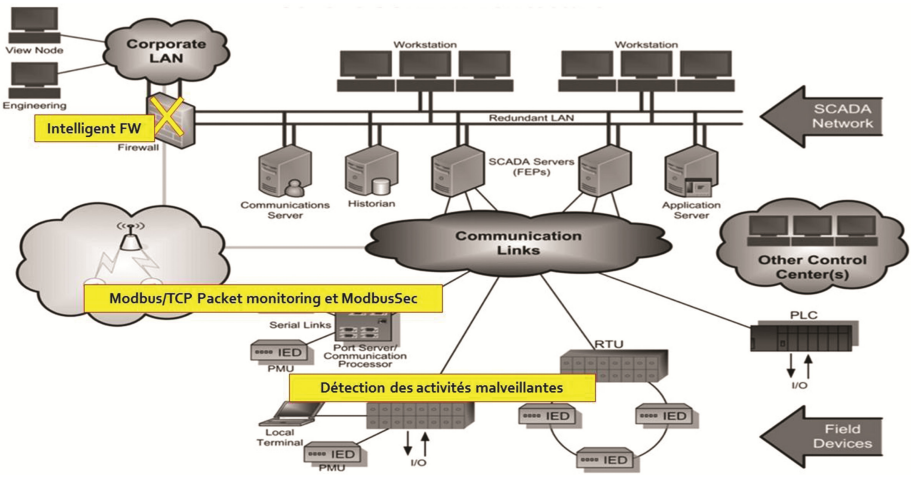


Fig. 5. Our proposed secured architecture for SCADA network

we could use to apply a redundancy of the devices of the control level as a measure of intrusion tolerance.

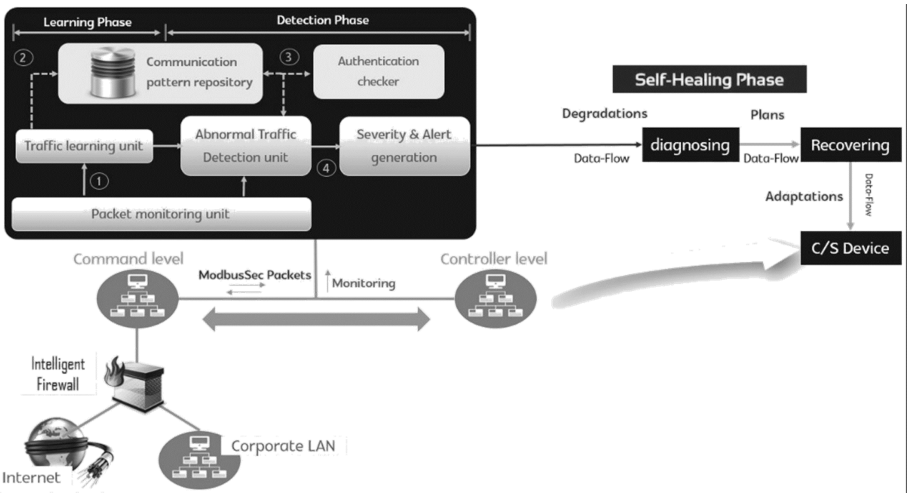


Fig. 6. Securing SCADA critical network against internal and external threats

After defining the protocol used for communication between the control level and the control level. We proceed to the supervision of the latter using the following procedure:

- The packet-monitoring unit: supervises all packets transaction in the area between the control and command level. The ModbusSec messages filtered will be transmitted to the learning block unit.
- The traffic-learning unit: listens to the network traffic and generates ModbusSec communication patterns during learning phase. The generated patterns are used as normal patterns for detecting abnormal behavior after learning phase
- Abnormal traffic detection unit: analyzes network packets one by one. When the content of one packet is different from learned normal patterns, it will specify the severity of the abnormal detection. For instance if the packet has a different content and is from an unknown devices the alert will be with high severity than it is from a known device with a different content.
- Communication pattern repository: consists of the command table and the control table. They contain information collected from only ModbusSec packets. Figure 4 shows the entries of each table that are consulted by the abnormal traffic detection unit. The command table is used for detecting abnormal command that can be derived from a machine or a device of the command level that is not registered in the repository. Then, the control table is used for detection abnormal behavior in the control level, such as an invalid ModbusSec message form controller and a message to unknown controller. In addition, Modbus PDU can store up to 3 bytes for identifying

each command exactly, and Unit ID is used for identifying the controllers in serial line.

– Authentication checker: we do not also forget the security mechanism included in the ModbusSec package that is the HMAC that ensures the authentication of the different devices of both control and command levels. The devices authentication is verified at the authentication checker. If it detect an unknown device, the abnormal traffic unit raises an alert.

Through the monitoring of the ModbusSec packets, we will control and protect the communication between control and command levels.

For the field level, we adopt malicious activities detection application to deal with all internal threats coming from compromised nodes (sensors and/or actuators). This approach detect and response for several attacks as: Jamming attack in which an adversary keeps sending useless signals and interferes with the communication of the sensor nodes present in the network. It is a kind of DOS attack in which adversary attempts to disrupt or damage the network and makes nodes unable to communicate. In addition, Sybil attack that is a particularly harmful attack against sensor networks where a malicious device illegitimately takes on multiple identities.

For the alerts generated by the abnormal traffic and activities detection techniques used in the field, control and command levels, we propose to filter suspicious data. Then, we will send it to determine the cause in the diagnosing block (using diagnosing tools like NetDecoder and Direct-Link SNMP Management Suite (SNMP MS)). Moreover, according to the cause of the incidents, the recovering block calculates an appropriate recovery plan according to the severity score of the incident. Consequently, it will be applied to restore the system, isolate the compromised devices, add the suspicious data pattern to the auto-dropped packets and the compromised devices to the packet monitoring unit black list.

3.3 Advantages

The proposed approach implements an in-depth defense including protection, warning, measurement and response to cyber incidents. In addition, the application of the intrusion tolerance mechanisms and self-healing for which there is, so far, no research project involving them with SCADA systems. The intrusion tolerance is insured in the intelligent firewall by adopting the masking technique defined at [16]. Moreover, the self-healing is applied in the case of failure of one mandatory (or many mandatories); it will execute a recovery system. In addition, the number of mandatories that can be restored at the same time define the number of replicas to be placed in the intelligent firewall and always having a working mandatory.

As benefits of our vision, we will say that the various existing solutions in the literature are only designed to secure one of the levels of SCADA architecture. In our approach, in addition of ensuring secure levels we also respect the fact of keeping the topology as it is.

On the other hand, there are approaches using encryption but it annihilates service availability. The aim is ensuring a high level of availability, resilience, access control, and detection of malicious activities and the transparent cover of the system. Therefore, we provide a robust internal and external protection of SCADA architectures.

Table 1 shows a comparison between our approach and other approaches mentioned in the related work section. In the table, we can observe that our approach covers the protection of all SCADA network levels with the response to malicious nodes at the field level. In addition, a complete supervision in the area between the command and control levels by ensuring the authentication, integrity and availability by using the ModbusSec as a communication protocol as well as the self-healing process. On the area between Internet and the control level, we use an intelligent firewall with intrusion tolerance mechanisms and self-healing. We find that the approach characteristics are not found together at any of the approaches [4, 6–8, 12] which lead as to consider that our approach with an in-depth defense.

Table 1. Comparison between the proposed approach and other ones

Approaches	Ours	[6]	[4]	[8]	[7]	[12]
Detection	☑	☑	☑	☒	☑	☑
Response	☑	☑	☒	☒	☑	☑
Field level	☑	☑	☒	☒	☒	☑
Control level	☑	☒	☑	☑	☒	☑
Command level	☑	☒	☑	☑	☒	☑
External threats	☑	☒	☑	☑	☑	☑
Internal threats	☑	☒	☒	☒	☑	☒
Self-healing	☑	☒	☒	☒	☑	☒
Intrusion tolerance	☑	☒	☒	☒	☑	☒

The proposed approach protect SCADA systems against several attacks as follows in the Table 2:

Table 2. Attacks protected against them due to the proposed approach

Attack	Caused by	Solution	Zone
Any attacks	Internet	Intelligent firewall	Internet and command level
Message spoofing	Lack device authentication	ModbusSec: HMAC	Command and control level
Replay attacks			
Denial of service			
Man-in-the-middle			
Doorknob rattling attack	Access control	The occurrence of at least six failed logins in the log within 30 s	
Selective forwarding and black hole attack	Compromised nodes	The mechanism uses acknowledgement (ACK)	Control and field level
Sybil attack	Malicious sensors	Our log records the location with their Ids when a node sends data to its destination. If two identities are recorded from the same location, log infers that it is a malicious node	
Jamming attack	Jammer nodes	If the traffic from the same node identity repeats for above or equal to a threshold value it may be from adversaries to cause jamming in the network	

4 Concluding and Future Works

According to the approach, a high availability is guaranteed by using intrusion tolerance in the network border, self-healing, and SCTP as transport protocol in the area between the control and the command levels. In addition, we guarantee an access control at the incoming data to the SCADA network due to of the intelligent firewall and a check of the devices authentication and the data integrity in the area between the control and the command levels by the HMAC field included in the ModbusSec message. As future work, we will create a platform to evaluate our proposed approach and to come with results in a future publication. Moreover, we will consider an internal access control application by applying an access control policy at supervisory level by restricting access to HMIs in other future work. In addition, it is expected to employ a mechanism of

tolerance to intrusion at supervisory level and keeping availability as the most critical factor in the SCADA system.

References

1. Gao, J., Liu, J., Rajan, B., Nori, R., Fu, B., Xiao, Y., Philip Chen, C.L.: SCADA communication and security issues. Sec. Commun. Netw. **7**(1), 175–194 (2014)
2. Psaier, H., Dustdar, S.: A survey on self-healing systems: approaches and systems. Computing **91**(1), 43–73 (2011)
3. Shahzad, A., Musa, S., Aborujilah, A., Irfan, M.: Secure cryptography testbed implementation for SCADA protocols security. In: 2013 International Conference on Advanced Computer Science Applications and Technologies (ACSAT), pp. 315–320. IEEE, December 2013
4. Shahzad, A., Xiong, N., Irfan, M., Lee, M., Hussain, S., Khaltar, B.: A SCADA intermediate simulation platform to enhance the system security. In: 2015 17th International Conference on Advanced Communication Technology (ICACT), pp. 368–373. IEEE, July 2015
5. Kim, B.K., Kang, D.H., Na, J.C., Chung, T.M.: Detecting abnormal behavior in SCADA networks using normal traffic pattern learning. In: Park, J., Stojmenovic, I., Jeong, H., Yi, G. (eds.) Computer Science and its Applications. Lecture Notes in Electrical Engineering, vol. 330, pp. 121–126. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-45402-2_18
6. Pramod, T.C., Sunitha, N.R.: An approach to detect malicious activities in SCADA systems. In: 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT), pp. 1–7. IEEE, July 2013
7. Sousa, P., Bessani, A.N., Dantas, W.S., Souto, F., Correia, M., Neves, N.F.: Intrusion-tolerant self-healing devices for critical infrastructure protection. In: IEEE/IFIP International Conference on Dependable Systems & Networks, DSN 2009, pp. 217–222. IEEE, June 2009
8. Hayes, G., El-Khatib, K.: Securing modbus transactions using hash-based message authentication codes and stream transmission control protocol. In: 2013 Third International Conference on Communications and Information Technology (ICCIT), pp. 179–184. IEEE, June 2013
9. Chen, Q., Abdelwahed, S.: Towards realizing self-protecting SCADA systems. In: Proceedings of the 9th Annual Cyber and Information Security Research Conference, pp. 105–108. ACM, April 2014
10. Blangenois, J., Guemkam, G., Feltus, C., Khadraoui, D.: Organizational security architecture for critical infrastructure. In: 2013 Eighth International Conference on Availability, Reliability and Security (ARES), pp. 316–323. IEEE, September 2013
11. Ghosh, D., Sharman, R., Raghav Rao, H., Upadhyaya, S.: Self-healing systems—survey and synthesis. Decis. Support Syst. **42**(4), 2164–2185 (2007)
12. Panja, B., Oros, J., Britton, J., Meharia, P., Pati, S.: Intelligent gateway for SCADA system security: a multi-layer attack prevention approach. In: 2015 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA), pp. 1–6. IEEE, June 2015
13. Ameziane El Hassani, A., Abou El Kalam, A., Bouhoula, A., Abbassi, R., Ait Ouahman, A.: Integrity-OrBAC: a new model to preserve critical infrastructures integrity. Int. J. Inf. Secur. **14**(4), 369–385 (2014). https://doi.org/10.1007/s10207-014-0254-9
14. Abou El Kalam, A., Baina, A., Deswarte, Y., Kaaniche, M.: PolyOrBAC: a security framework for critical infrastructures. Int. J. Crit. Infrastruct. Prot. (IJCIP) **2**(4), 154–169 (2009). https://doi.org/10.1016/j.ijcip.2009.08.005

15. Veríssimo, P., Neves, Nuno F., Correia, M., Deswarte, Y., Abou El Kalam, A., Bondavalli, A., Daidone, A.: The CRUTIAL architecture for critical information infrastructures. In: de Lemos, R., Di Giandomenico, F., Gacek, C., Muccini, H., Vieira, M. (eds.) WADS 2007. LNCS, vol. 5135, pp. 1–27. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-85571-2_1
16. Deswarte, Y.: Comment peut-on tolérer les Intrusions sur Internet? Revue de l'électricité et de l'électronique **8**, 83–90 (2003)
17. Huitsing, P., Chandia, R., Papa, M., Shenoi, S.: Attack taxonomies for the modbus protocol. Int. J. Crit. Infrastruc. Prot. **1**, 37–44 (2008)
18. Bhatia, S., et al.: Practical modbus flooding attack and detection. In: 2014 Proceedings of the ACSW-AISC, pp. 20–13 (2014)

Simulation of Cascading Outages in (Inter)-Dependent Services and Estimate of Their Societal Consequences

Short Paper

Antonio Di Pietro(✉), Luigi La Porta, Luisa Lavalle, Maurizio Pollino, Vittorio Rosato, and Alberto Tofani

Laboratory for the Analysis and Protection of Critical Infrastructures, ENEA, Via Anguillarese 301, 00123 Rome, Italy
{antonio.dipietro,luigi.laporta,luisa.lavalle,maurizio.pollino, vittorio.rosato,alberto.tofani}@enea.it
http://www.enea.it

Abstract. We present RecSIM, a specific application which is part of the CIPCast tool, a Decision Support System (DSS) for risk analysis of (inter)-dependent Critical Infrastructures (CI) under development within the EU FP7 CIPRNet Project. Electrical and the telecommunication networks play a central role in systems of CI and are also strongly inter-dependent: the latter allows tele-control operations on the former which, in turn, supplies energy to the latter. Services outages of those CI can generate cascading effects also on other systems, thus producing large societal consequences. The RecSIM tool is a discrete-time simulator able to model the dependencies between the electric and the telecommunication systems, to simulate the cascading effects in the two networks and to reproduce the results of the actions taken by operators to reconfigure the electrical network. Starting from the prediction of damages on the network topology, RecSIM reproduces the whole crisis and estimates the resulting outages (outage start and outage end) in all the involved CI elements. RecSIM, moreover, through the Consequence Analysis module, estimates the effects of the degradation (or the complete loss) of the electric and the telecommunication services, via the estimate of specific indices useful to assess the severity of the crisis in terms of its societal impact.

Keywords: Critical infrastructures · Simulator · Electrical network Telecommunication network · Wealth index

1 Introduction

Electrical and telecommunication networks are at the heart of modern cities. Frequently, mobile communication networks provide tele-control capabilities to the electrical distribution (Medium Voltage, MV) networks via Base Transceiver

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, pp. 340–345, 2017.
https://doi.org/10.1007/978-3-319-71368-7_30

Stations (BTS). These stations, in turns, rely on energy supplied (in most of the cases) by the same electrical operators to whom they ensure tele-controllability. This is the origin of the possible feedback loops which could be triggered by the fault of elements in one or the other infrastructure.

RecSIM (Reconfiguration actions SIMulation in power grids [1]) is a discrete-time simulator developed in Java. Starting from the modelling of the electrical-telecommunication network and of their dependencies, it emulates the sequence of events which produces when one (or more) CI elements of one of the two networks fail and the actions performed by the electric operator to restore the network. RecSIM has been integrated into the CIPCast Decision Support System to provide the *Impact Estimation* capability i.e. to assess a real-time risk analysis of inter-dependent critical infrastructures subjected to natural hazards. In fact, CIPCast firstly predicts the faults on CI elements in the networks and then, through the RecSIM module, transforms the physical damages into reduction (or loss) of services. The result of RecSIM consists thus of the outage time profile of each electric and telecommunication components which are directly damaged by the events or failed by some cascading effects. The outage time profile is then input to a further CIPCast module called *Consequence Analysis* that estimates (through the use of appropriate metrics) the ultimate effects of the reduction (or loss) of services on citizens, on the economy sectors and on the environment.

2 RecSIM Model

The main components of the RecSIM model include:[1] (i) Primary Substations (PS) containing HV-MV transformers; (ii) Secondary Substations (SS) containing MV-LV transformers; (iii) Feeders holding the set of secondary substations; (iv) Switches which section each SS and that can be operated to perform reconfiguration actions; (v) the number and the initial position in the area of interest of crew teams and power generators required to re-energize disconnected electric substations; and (vi) BTS which provide tele-control capability to those SS that are remotely controlled. The RecSIM procedure identifies those SS that, due to the loss of tele-control capability, require manual intervention and those that can be reconnected via SCADA system. Then, considering the average time required to technical crews to reach specific SS and reconnect the relative users (using Power Generators), the procedure estimates the $T_i(t)$, the function expressing the outage profile of each one of the $i = [1, M]$ SS involved in the outage over the time of interest T (usually some hours). It is clear that, according to the sequence of manual actions executed by the crews, there might be different impact outcomes. In fact, some SS can supply many more households w.r.t. other ones or the reactivation of some of them could be preparatory for other restoration and/or enabling some actions to be performed more rapidly.

[1] HV = High Voltage, MV = Medium Voltage, LV = Low Voltage.

3 Consequence Analysis

The Consequence Analysis (CA) module identifies the sectors of the societal life to be considered to describe the consequences inflicted by a crisis of CI services and, for each sector, the metric which better measures the extent of the consequences. CA allows to estimate the Consequences by using two approaches (i.e. metrics): the *Service Continuity Index*, a specific metric used by the Italian operators and authorities and the *Service Wealth Indices*, a new metric introduced to improve the consequence estimation by projecting their effects on the different societal sectors. Consequences are estimated (for each selected metric) by assuming an "expected" Wealth[2] (W_{exp}), assumed when all CI elements are correctly functioning) and an "effective" Wealth (W_{eff}), estimated by inserted in the Wealth metric the predicted outages. Consequence will be thus estimated as

$$Consequence = \Delta W = W_{exp} - W_{eff} \tag{1}$$

If $\Delta W = 0$ there will be no consequences; if, in turn, W_{eff} is lower than W_{exp} the weight of consequences is higher and the predicted crisis more severe.

3.1 Service Continuity Index

The *Service Continuity Index* (*kmin*) is estimated by multiplying the time duration of the outage T_i experienced by each of the CI elements involved in the electrical crisis and the number of electrical users (supplied by the CI elements, u_i) involved. If , for a number M of affected SS the expected Wealth on a period T would be $W_{exp} = \sum_{i=1}^{M} u_i T$, when some outage will occur on the same number of CI elements the effective Wealth would be $W_{eff} = \sum_{i=1}^{M} u_i T_i$. Thus

$$\Delta W = kmin = \sum_{i=1}^{M} u_i (T - T_i) \tag{2}$$

The index *kmin* is thus expressed in minutes and provides a prompt estimate of the crisis consequences to users. Larger the value of *kmin*, lower the Service Continuity and larger the societal consequences. A short blackout in a highly populated area may produce a high kilominutes crisis w.r.t. a longer outage in a less populated area. All customers, both people and economic activities, are considered equally weighted.

3.2 Service Access Wealth Metric

Whereas the Service Continuity metric is only focussed on the electrical service, the *Service Access Wealth* (SAW) metric attempts to perform a finer grain analysis, firstly by discriminating among different societal sectors (citizens, industrial

[2] When dealing with a metric which measures the well-being of a system (i.e., a state of normal functioning where all the expectations are fulfilled) we will refer to it as the "Wealth" of that system.

activities and the environment) and then, within a given sector, by differently treating the different classes of a given sector. The SAW metric considers a complex landscape, where the consequences of the outages of each service are "modulated" with respect to specific indices (called SAW Indices) describing the relevance of a specific service for the Wealth fulfillment of a societal sector element t_{ij}. In the SAW metric, society is decomposed into societal sectors (citizens, public services, economic activities, environment) labelled by t_i and each Sector further decomposed in Sector element (thus t_{ij} will indicate the j-th element of the i-th sector). For each t_{ij} we could define the expected Wealth $W_{exp}(t_{ij})$ i.e., the level of well-being of that sector element as resulting from the availability of the $N = 5$ services (electricity, gas, water, telecommunication, public transportation), each being identified by their outage profile $T_{kl}(t)$ as follows (k labels the service, l the element of that service):

$$W_{exp}(t_{ij}) = M(t_{ij}) \tag{3}$$

$$W_{eff}(t_{ij}) = M(t_{ij}) \sum_{k=1}^{N} \sum_{l=1}^{M} \int_0^T r_k(t_{ij}) T_{kl}(\tau) d\tau \tag{4}$$

Thus

$$\Delta W = M(t_{ij})[1 - \sum_{k=1}^{N} \sum_{j=1}^{M} \int_0^T r_k(t_{ij}) T_{kl}(\tau) d\tau] \tag{5}$$

where $r_k(t_{ij})$ are the SAW indices. The terms $T_{kl}(t)$ are the time outage profiles of each element of each CI involved in the outage, which are estimated by RecSIM.

For each societal sector elements, a wise recognition in available Open Data has allowed to estimate the "relevance" of a specific service availability to determine the Wealth of a given societal sector element. For some elements of the industrial sector, for instance, the availability of the electrical service could be more relevant than for another where, in turn, the telecommunication service could be more relevant to produce the usual turnover. Citizens of a given class of age (elderly people) might be more vulnerable to the absence of electricity that of telecommunication, more of water supply and less of public transportation. SAW Indices attempt to keep these differences and turn them into values $(0 < r_k(t_{ij}) < 1)$.

The estimate of $r_k(t_{ij})$ for citizens has been carried out by inferring their value from available statistical data: the more a service is used, the more it is considered as relevant for the Wealth achievement of a given sector element. Using such a working hypothesis, we have estimated the SAW indices for the different sectors elements coherently with independent studies and measurement campaigns made by CI operators for electricity, gas and water [4]. SAW indices related to telecommunication services, for instance, have been inferred from data about telecommunication usage provided by ISTAT [5]. SAW indices for the different elements of Sector "Citizens" are reported in Table 1. Although a general Consequence Analysis implies the identification of time-dependent SAW indices

Table 1. SAW indices for Consequence Analysis about citizens.

Sector Elements	Electricity	Telecom	Water	Gas
Citizens (0–5) e_1	0.234	0	0.181	0.095
Citizens (18–64) e_2	0.288	0.145	0.212	0.097
Citizens (65 or older) e_3	0.398	0.126	0.343	0.134

(Services relevance has a time variation during the course of the day) for the sake of simplicity we have performed SAW indices estimate in a time-independent way, by summing up all the usages in all the periods of the day and normalising to the highest value.

When dealing with the sector of Economical activities, the relevance of each service has been related to the effect that its unavailability would have in terms of economic losses: consequences are estimated as the difference between the expected and the effective turnover produced in a given period of time. In order to elicit the $r_k(t_{ij})$, still keeping a statistical approach, we have used the input-output tables [3], $n \times n$ matrices representing the mutual relations between the various economical activities, showing which and how goods and services produced (output) by each activity are used by others as inputs in their production processes. These data are usually released by Statistical Offices and acknowledged in the National Accounts of many countries. With the basic assumption that all industrial costs contribute to the production and thus to the turnout - we grouped all industries (with different NACE codes) in belonging to the Primary, Secondary and Tertiary sectors; for each sector, the percentage of the whole budget spent for the different CI related services has been subsequently estimated (Table 2). Gas service relevance is not available as in the input-output matrices Electricity and Gas are considered in the same primary service. It is worth noticing that services relevance are estimated in "normal" situation; we are aware, however, that such relevances could change over an emergency. Figure 1 shows consequences of an outage in a specific area of the city of Rome (Italy) both in terms of *kmin* and by using the SAW metric. As it can be seen, the SAW metric allow to differentiate the different city areas whose consequences appear to be similar when considering the metric defined by using *kmin*.

Table 2. SAW indices for Consequence Analysis about economy.

Sectors	Electricity	Telecom	Water	Gas	Mobility
Primary t_{31}	0.4	0.06	0.186	N/A	0.408
Secondary t_{32}	0.3	0.058	0.23	N/A	0.411
Tertiary t_{33}	0.197	0.248	0.185	N/A	0.37

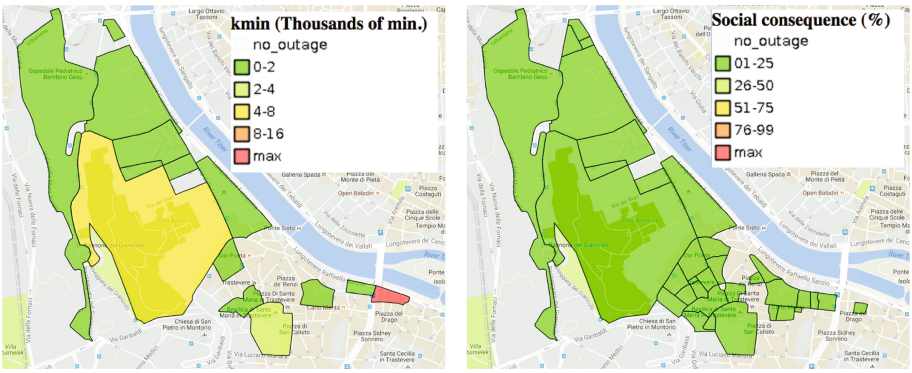


Fig. 1. An example of kmin and SAW index for citizens for some census areas of Rome.

4 Conclusions

In this work, we have shown the basic features of the *Consequence Analysis* module of CIPCast Decision Support System able to perform a "quantitative" estimate of the consequences that the reduction (or loss) of services (caused by the damage of CI elements and the resulting cascading effects) might produce on the different societal sectors. This information to CI operators and Emergency managers for perceiving the ultimate social consequences that a coming crisis scenario could produce. Future work will focus on the extension of the approach to consider additional CI services (e.g., water, gas) and the application of optimisation techniques to reduce the actions to be taken with the aim of minimising the consequences for specific contexts (e.g., citizens).

Acknowledgments. This work was developed from the FP7 Network of Excellence CIPRNet, which is being partly funded by the European Commission under grant number FP7-312450-CIPRNet. The European Commissions support is gratefully acknowledged.

References

1. Di Pietro, A., Lavalle, L., Pollino, M., Rosato, V., Tofani, A.: Supporting decision makers in crisis scenarios involving interdependent physical systems. The International Emergency Management Society (TIEMS) (2015)
2. Smits, J., Steendijk, R.: The International Wealth Index, NiCE Working Paper 12-107, Nijmegen Center for Economics (NiCE) Institute for Management Research, Radboud University, Nijmegen (The Netherlands), December 2013
3. Leontief, W.: Input-output analysis. In: Input- Output Economics, p. 1940 (1986)
4. Di Pietro, A., Lavalle, L., La Porta, L., Pollino, M., Tofani, A., Rosato, V.: Design of DSS for supporting preparedness to and management of anomalous situations in complex scenarios. In: Setola, R., Rosato, V., Kyriakides, E., Rome, E. (eds.) Managing the Complexity of Critical Infrastructures. SSDC, vol. 90, pp. 195–232. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-51043-9_9
5. Italian National Institute of Statistics. http://www.istat.it/it/prodotti/microdati

Erratum to: Human Vulnerability Mapping Facing Critical Service Disruptions for Crisis Managers

Amélie Grangeat[1(✉)], Julie Sina[2], Vittorio Rosato[3], Aurélia Bony[2], and Marianthi Theocharidou[4]

[1] CEA/GRAMAT, 46500 Gramat, France
amelie.grangeat@orange.fr
[2] Institut des Sciences des Risques – Centre LGEI, Ecole des mines d'Alès, 30100 Alès, France
julie.sina@hotmail.fr, aurelia.bony-dandrieux@mines-ales.fr
[3] ENEA Casaccia Research Centre, Rome, Italy
vittorio.rosato@enea.it
[4] European Commission, Joint Research Centre (JRC), Via E. Fermi 2749, 21027 Ispra, VA, Italy
marianthi.theocharidou@ec.europa.eu

Erratum to:
Chapter "Human Vulnerability Mapping Facing Critical Service Disruptions for Crisis Managers" in:
G. Havarneanu et al. (Eds.): *Critical Information Infrastructures Security*, LNCS 10242,
https://doi.org/10.1007/978-3-319-71368-7_9

By mistake, the originally published version of the paper did not acknowledge that Marianthi Theocharidou is an employee of the European Union and that therefore the copyright on her part of the work belongs to the European Union. This has been corrected in the updated version.

The updated online version of this chapter can be found at https://doi.org/10.1007/978-3-319-71368-7_9

G. Havarneanu et al. (Eds.): CRITIS 2016, LNCS 10242, p. E1, 2017.
https://doi.org/10.1007/978-3-319-71368-7_31

Author Index