UNIVERSITY OF PIRAEUS

SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGIES

DEPARTMENT OF INFORMATICS

PhD Thesis

Thesis Title:	(English) Identification, modeling and assessment of IoT-enabled, cyber-physical attack paths against critical infrastructures and services
	(Greek) Αναγνώριση, μοντελοποίηση και αξιολόγηση κυβερνο-φυσικών μονοπατιών επίθεσης προερχόμενα από το Διαδίκτυο των Πραγμάτων, κατά κρίσιμων υποδομών και υπηρεσιών
Student’s name-surname:	Ioannis Stellios
Student’s ID No:	ΠΛΔ 1608
Main Supervisor:	Kotzanikolaou Panayiotis, Associate Professor

June 2022

PhD Thesis was prepared during the Programme of Doctoral Studies of the Department of Informatics of the School of Information and Communication Technologies of the University of Piraeus for the degree of Doctor of Philosophy

Supervising Committee

Panayiotis Kotzanikolaou
Associate Professor
University of Piraeus
School of Information and
Communication Technologies
Department of Informatics
(Main supervisor)

Cristina Alcaraz
Associate Professor
University of Malaga
Computer Science Department
(Co-supervisor)

Mihalis Psarakis
Associate Professor
University of Piraeus
School of Information and
Communication Technologies
Department of Informatics
(Co-supervisor)

Evaluation Committee

Cristina Alcaraz
Associate Professor
University of Malaga
Computer Science Department

Christos Douligeris
Professor
University of Piraeus
School of Information and
Communication Technologies
Department of Informatics

Panayiotis Kotzanikolaou
Associate Professor
University of Piraeus
School of Information and
Communication Technologies
Department of Informatics

Javier Lopez
Professor
University of Malaga
Computer Science Department

Emmanouil Magkos
Associate Professor
Ionian University
Department of Informatics

Despina Polemi
Professor
University of Piraeus
School of Information and
Communication Technologies
Department of Informatics

Mihalis Psarakis
Associate Professor
University of Piraeus
School of Information and
Communication Technologies
Department of Informatics

PhD Thesis was presented before the Evaluation Committee and was approved on June 6th, 2022

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my advisor, Associate Prof. Panayiotis Kotzanikolaou, for his mentoring and patience, as well as to my co-supervisors Associate Prof. Mihalis Psarakis and Associate Prof. Cristina Alcaraz for their mentoring and support throughout this work. In addition, I would like to express my appreciation to the members of the Evaluation Committee, Prof. Christos Douligeris, Prof. Javier Lopez, Associate Prof. Emmanouil Magkos and Prof. Despina Polemi, who honored me with their participation. I would also like to give credit to Christos Grigoriadis and Kostas Mokos for their collaboration and their contribution in research papers related with the results presented in this PhD Thesis. Finally, many thanks to all of my friends and my family for their encouragement and support during this challenging task!

Abstract

Critical Infrastructures (CIs) play a vital role to the well-being of our society, as their disruption would have a significant effect on the security, safety, economy and public health at a national or even international level. Power grids, communication networks, industry infrastructures, transportation networks, health services, financial services, agriculture as well as urban environments can be considered as the most important CI sectors. In the last few decades the growth of Information and Communication Technologies (ICT) have introduced Industrial Control Systems (ICS) which, in turn, play a vital role on most CIs environments. Unfortunately, cyber-physical threats evolved to fit this new environment. Attacks that formerly required physical access to be triggered, have not become cyber-enabled: A remote adversary could disrupt the operations of a CI just by attacking the corresponding ICS systems.

Furthermore, the introduction of Industry 4.0 as well as the Internet of Things (IoT) related technologies have further transformed the CIs. Enabling features such as system automation and operating efficiency, remote management, command & control, production programming and optimization, human error as well as production cost reduction became the norm to otherwise isolated complex cyber-physical systems.

But all this interconnectivity, interoperability and physical proximity transformed the threat landscape by introducing complex and hard-to-identify attack vectors against Cyber-Physical Systems (CPS) that used to be isolated systems. In addition, the lack of up-to-date security controls and frameworks, the use of commercial, off-the-shelf IoT devices in manufacturing and industrial facilities, the plethora of vulnerabilities found in both hardware and software, the adoption of insecure wireless network protocols and the copious cyber-physical capabilities of IoT-devices, have enabled remote adversaries to extend their reach from cyber to cyber-physical thus resulting in complex, subliminal attack scenarios. Most of these attacks can be considered as *IoT-enabled*: The attacker initially exploits some vulnerable IoT technology as a first step towards compromising a critical system that is connected with it, in some way.

Unfortunately, existing Risk Assessment (RA) methodologies cannot address these new threat types. In the literature, there is a lack of risk assessment methodologies targeted in identifying, modelling and assessing such complex cyber-physical attack vectors. The main research goal of this thesis is to contribute in understanding, identifying and assessing these novel IoT-enabled, cyber-physical attacks paths against critical infrastructures and services.

The thesis is structured in five sections, each of which includes a number of chapters. In Section I the foundations (Chapter 1) and the related work (Chapter 2) is introduced, to assist the reader in understanding the current state-of-the-art and the open research challenges related with the identification and assessment of IoT-enabled, cyber-physical attacks. Section II (Chapters 3-4) analyzes the relevant threat landscape. In particular, in Chapter 3 we review recent, Proof-of-Concept (PoC) as well as real incidents of IoT-enabled attacks on critical infrastructures and services whereas in Chapter 4 we dive into a deeper analysis of high-profile attacks presented in the previous chapter.

Section III (Chapters 5-6) introduces the novel risk assessment methodologies introduced in this thesis. Specifically, in Chapter 5 we propose a high-level framework in order to assess the criticality of the attack scenarios presented in Chapters 3 and 4. Then, in Chapter 6, we develop a low-level, detailed RA methodology to identify, model and assess complex, IoT-enabled cyber-physical attacks.

Section IV (Chapters 7-9) focuses on the validation of the methodologies presented in Chapters 5 and 6. Particularly, in Chapter 7 we apply the framework presented in Chapter 5 on the cyber-physical attacks presented in Chapters 3 and 4, considering a worst-case scenario approach. Then, we test the low-level RA methodology presented in Chapter 6, in two different cases: a smart city scenario (Chapter 8) and a healthcare scenario (Chapter 9).

Finally, Section V (Chapters 10-11) summarizes the results of this thesis that are related with the mitigation of IoT-enabled attack paths, along with open research challenges that require additional future work respectively. In Chapter 10 state-of-the-art mitigation controls are proposed for specific domains. In particular, countermeasures that aim at reducing the threat and/or the vulnerability level, in the context of the attack scenarios presented in Chapter 10.

Additionally, mitigation strategies based on the results of our low-level methodology are presented for the e-healthcare PoC scenario. Finally, Chapter 11 concludes this thesis by providing an overview of the proposed methodologies, along with their limitations and the future research challenges that have been identified.

Περίληψη

Οι Κρίσιμες Υποδομές (ΚΥ) διαδραματίζουν ζωτικό ρόλο για την ευημερία της κοινωνίας μας, ενώ οποιαδήποτε διαταραχή των υπηρεσιών που προσφέρουν έχει σημαντικές επιπτώσεις στην ασφάλεια καθώς και στην οικονομική και κοινωνική ευημερία των πολιτών τόσο σε εθνικό αλλά και σε διεθνές επίπεδο. Δίκτυα ενέργειας, μεταφορών και επικοινωνιών, βιομηχανικές υποδομές, υπηρεσίες υγείας, χρηματοοικονομικές υπηρεσίες, πρωτογενής παραγωγή τροφίμων καθώς και το αστικό περιβάλλον θεωρούνται ως οι κύριοι τομείς των ΚΥ.

Τις τελευταίες δεκαετίες η εξέλιξη των Τεχνολογιών Πληροφορικής και Επικοινωνιών (ΤΠΕ) έχουν ως αποτέλεσμα την ενσωμάτωση των παραπάνω σε Συστήματα Βιομηχανικού Ελέγχου (ΣΒΕ) τα οποία υποστηρίζουν πολλούς τομείς των ΚΥ με αποτέλεσμα την δραστική αλλαγή του πεδίου των απειλών. Πιο συγκεκριμένα, επιθέσεις που προηγουμένως απαιτούσαν φυσική πρόσβαση προκειμένου να πραγματοποιηθούν, ήταν πλέον δυνατό να πραγματοποιηθούν απομακρυσμένα μέσω του κυβερνοχώρου και να διαταράξουν την λειτουργία κυβερνο-φυσικών συστημάτων ζωτικής σημασίας για μια ΚΥ. Επιπρόσθετα, η υιοθέτηση του βιομηχανικού προτύπου "Industry 4.0" καθώς και οι τεχνολογίες που σχετίζονται με το Διαδίκτυο των Πραγμάτων (ΔτΠ) έχουν μεταμορφώσει το περιβάλλον στο οποίο λειτουργούν τα εν λόγω κρίσιμα κυβερνο-φυσικά συστήματα. Οι τεχνολογίες που αφορούν το ΔτΠ προσφέρουν, μεταξύ άλλων, την βελτιστοποίηση των παραγωγικών διαδικασιών, την μείωση των μηχανικών βλαβών και του κόστους παραγωγής καθώς και τον περιορισμό των λαθών που προέρχονται από τον ανθρώπινο παράγοντα, μέσω των υπηρεσιών της απομακρυσμένης διαχείρισης, ελέγχου και ανάλυσης των λειτουργικών δεδομένων, της διασυνδεσιμότητας και διαλειτουργικότητας των συστημάτων αλλά και την υιοθέτηση τεχνικών μηχανικής μάθησης και τεχνητής νοημοσύνης.

Τα παραπάνω πλεονεκτήματα έχουν όμως και ένα σημαντικό κόστος: Η συνδεσιμότητα, διαλειτουργικότητα καθώς και η φυσική εγγύτητα μεταξύ ετερογενών συστημάτων, πρωτοκόλλων δικτύου και εφαρμογών έχει ως αποτέλεσμα την περαιτέρω μετεξέλιξη του πεδίου των απειλών σε σύνθετα διανύσματα αλληλεπιδράσεων, τα οποία έως σήμερα, δεν είναι εύκολο να εντοπιστούν και να αξιολογηθούν ως προς το βαθμό της επικινδυνότητάς τους, μέσω των υφιστάμενων μεθοδολογιών αποτίμησης κινδύνου.

Συνδυάζοντας όλα τα παραπάνω και λαμβάνοντας υπόψιν το μεγάλο αριθμό των ευπαθειών που εντοπίζονται σε καθημερινή βάση στο επίπεδο εφαρμογών και δικτύων των "έξυπνων" συσκευών και συστημάτων αυτοματισμού, την πληθώρα των κυβερνοφυσικών δυνατοτήτων των εν λόγω συσκευών (π.χ. ασύρματες διεπαφές δικτύου, αισθητήρες φωτός, ήχου, εγγύτητας), την έλλειψη ελέγχων για τον εντοπισμό ευπαθειών αλλά και την απουσία αναγνωρισμένων και εξειδικευμένων προτύπων ασφαλείας, γίνεται αντιληπτό ότι κακόβουλοι πράκτορες δύναται να αλληλεπιδράσουν με κρίσιμα

συστήματα από απόσταση ακόμα και με φυσικό τρόπο χωρίς να εντοπιστούν. Οι εν λόγω επιθέσεις χαρακτηρίζονται ως προερχόμενες από το ΔτΠ: Ο εισβολέας αρχικά εκμεταλλεύεται κάποια ευπάθεια μιας συσκευής του ΔτΠ ή/και αντίστοιχης υπηρεσίας προκειμένου στην συνέχεια να επιτεθεί σε ένα κρίσιμο σύστημα που είτε συνδέεται άμεσα ή έμμεσα ή/και βρίσκεται σε φυσική εγγύτητα με αυτό.

Τόσο οι υφιστάμενες μεθοδολογίες ανάλυσης επικινδυνότητας όσο και η μέχρι τώρα έρευνα δεν έχουν καταφέρει να αντιμετωπίσουν αποτελεσματικά τις νέες προκλήσεις που σχετίζονται με σύνθετα μονοπάτια κυβερνο-φυσικών επιθέσεων. Ο κύριος στόχος της παρούσας διδακτορικής διατριβής είναι να συμβάλει στην κατανόηση, ανάλυση και αξιολόγηση της επικινδυνότητας σύνθετων σεναρίων κυβερνο-φυσικών μονοπατιών επίθεσης ενάντια σε κυβερνο-φυσικά συστήματα που παρέχουν υπηρεσίες ζωτικής σημασίας για την κοινωνία, και τα οποία βρίσκονται εγκατεστημένα σε διάφορους τομείς των ΚΥ.

Η παρούσα διατριβή αποτελείται από πέντε θεματικές ενότητες, καθεμία από τις οποίες αποτελείται από μια σειρά από επιμέρους κεφάλαια. Πιο συγκεκριμένα, στην ενότητα I παρατίθεται το απαραίτητο εννοιολογικό πλαίσιο για την κατανόηση των τεχνικών όρων και του περιβάλλοντος των κρίσιμων υποδομών και των μεθοδολογιών αξιολόγησης της επικινδυνότητυας συμπεριλαμβανομένων, μεταξύ άλλων, την εξέλιξη του πεδίου των απειλών, καθώς και τις υφιστάμενες προκλήσεις που σχετίζονται με τον εντοπισμό, μοντελοποίηση και την ανάλυση της επικινδυνότητας σύνθετων, κυβερνο-φυσικών απειλών. Στην ενότητα II παρουσιάζονται, κατηγοριοποιούνται και αναλύονται πραγματικές καθώς και θεωρητικού επιπέδου κυβερνο-φυσικές επιθέσεις σε ΚΥ. Στην ενότητα III παρουσιάζονται μία υψηλού επιπέδου μεθοδολογία εκτίμησης ρίσκου με σκοπό την ανάλυση της κρισιμότητας των επιθέσεων που αναφέρονται στην ενότητα II καθώς και μία χαμηλού επιπέδου μεθοδολογία ανάλυσης, μοντελοποίησης και εκτίμησης της επικινδυνότητας σύνθετων κυβερνο-φυσικών απειλών από το πεδίο του ΔτΠ. Στην ενότητα IV, ελέγχουμε την αποτελεσματικότητα των μεθοδολογιών που αναπτύχθηκαν στην προηγούμενη θεματική ενότητα. Πιο συγκεκριμένα, αναλύουμε την επικινδυνότητα των πραγματικών αλλά και την χειρότερη δυνατή εκδοχή των θεωρητικών, κυβερνο-φυσικών σεναρίων επίθεσης, που παρουσιάστηκαν στην ενότητα II, εφαρμόζοντας την υψηλού επιπέδου μεθοδολογία που αναπτύξαμε στην ενότητα III. Επιπλέον, εφαρμόζουμε την μεθοδολογία χαμηλού επιπέδου στα διαφορετικούς τομείς μιας "έξυπνης" πόλης και σε ένα σύγχρονο περιβάλλον υπηρεσιών υγείας.

Τέλος, στην ενότητα V προτείνουμε μέτρα ασφάλειας τα οποία περιλαμβάνουν τόσο βέλτιστες πρακτικές, όσο και τεχνολογίες αιχμής αντιμέτρων ασφάλειας για κάθε κατηγορία και τύπο απειλής, καθώς και διαφορετικές στρατηγικές αντιμετώπισης κυβερνο-φυσικών απειλών, βασιζόμενοι στα αποτελέσματα της χαμηλού επιπέδου μεθοδολογίας μας για το σενάριο που αφορά της υπηρεσίες υγείας. Η διατριβή ολοκληρώνεται αναφέροντας τους περιορισμούς αλλά και τις μελλοντικές επεκτάσεις των μεθοδολογιών εκτίμησης επικινδυνότητας που παρουσιάστηκαν.

TABLE OF CONTENTS

CHAPTER PAGE

LIST OF FIGURES

FIGURE PAGE

GLOSSARY

AC Attack Complexity (CVSS metric): Low (L) and High (H). 100, 110–113, 119, 139–141, 145, 153

ACARS Aircraft Communications Addressing and Reporting System. 35, 41

ADS-B Automatic Dependent Surveillance Broadcast. 35, 41

AIS Automatic Identification System. 36, 42

AMD Active Medical Device. 44, 45, 49

AMI Advanced Metering Infrastructure. 29, 30, 32

AP Attack Path. 13, 14, 99, 102, 104, 115, 116, 118–120, 122, 134, 143, 144, 152–156, 165, 166

API Application Programming Interface. 4, 22, 27, 55, 63, 64, 88, 133, 139, 140, 162, 163

APT Advanced Persistence Threat. 8, 25, 66, 68–70, 74, 77, 79

AR Access Rules/Capabilities Topology. 105, 107, 108

ATG Automated Tank Gauge. 27

AV Attack Vector (CVSS metric): Network (N), Adjacent Network (A), Local (L) and Physical (P). 100, 101, 109–111, 113, 118, 119, 132, 134, 135, 138, 140, 141, 145, 151, 153, 169

CAN Controlled Area Network. 34, 37, 38

ChainedCVSS A Chained Vulnerability Vector. 113, 115, 138

CI Critical Infrastructure. 1–3, 5–8, 15, 16, 18, 20, 21, 24, 66, 84, 89, 95

CoAP Constrained Application Protocol. 4, 22, 51, 88, 162

CPS Cyber-Physical System. 74, 84, 95–97

CVE Common Vulnerabilities and Exposures. 100, 132

CVSS Common Vulnerability Scoring System. 11, 13, 100–102, 104, 109–111, 113, 114, 117–121, 132, 138, 140, 141

CVV Cumulative Vulnerability Vector of an interaction / attack path. 105, 109, 111, 114–120, 132, 134, 143, 144, 152

DAB Digital Audio Broadcast. 37, 38

DDoS Distributed Denial-of-Service. 12, 27, 32, 53, 56, 64, 90, 131

DLL Dynamic Link Library. 70, 82, 83

DoS Denial-of-Service. 30, 39, 41, 46, 47, 52, 54, 56, 58, 75, 80, 82, 102, 133, 139, 155, 162

EMR Electronic Medical Record. 45, 48, 170

FDIA False Data Injection Attacks. 29, 30, 32

FTP File Transfer Protocol. 28, 75

HMI Human-Machine Interface. 11, 22, 23, 26, 33, 66–68, 79

HTTP HyperText Transfer Protocol. 4, 27, 33, 42, 51, 57, 61, 63, 69, 74, 81, 140

HVAC Heating, Ventilation, and Air Conditioning. 4, 14, 149, 159

HW Hardware. 2, 87, 92, 94, 95

ICD Implantable Cardioverter Defibrillator. 44, 46, 47

ICS Industrial Control Systems. 3, 7, 16, 19, 22, 24, 25, 31, 72, 73, 79, 80, 83

IED Intelligent Electronic Device. 20, 22, 24

IFE In-Flight Entertainment. 35, 41, 42

IMD Implantable Medical Device. 44, 46

IntCVSS The base/environmental vulnerability vector of an interaction. 109–115, 138, 139, 141

IoT Internet-of-Things. 3–15, 84–99, 101, 103, 110, 123, 131–134, 136, 138, 139, 144, 148–151, 155–157, 159–165, 167, 169–171

IT Information Technology. 12, 24, 25, 27, 42–45, 49, 134, 148, 164

ITS Intelligent Transportation Systems. 33, 34, 37, 39

JTAG Joint Test Action Group. 47, 53, 61

LAN Local Area Network. 6, 22, 28, 37, 38, 72

MiTM Man-in-the-Middle. 42, 52, 57, 62, 63

NT Network Topology. 105, 107, 108

OLE Object Linking and Embedding. 3, 23, 82

OPC Object linking and embedding for Process Control. 3, 31, 68, 73, 80, 82

PID Public Information Display. 35, 40

PLC Programmable Logic Controller. 3, 26, 71, 73, 76

PMD Passive Medical Device. 44, 45, 49

PoC Proof-of-Concept. 7–10, 12, 32, 37–39, 41, 42, 47–49, 53, 55, 56, 76, 77, 95, 122, 123, 133, 136, 137, 140, 141, 151, 169, 170

PR Privilege Requirements (CVSS metric): None (N), Low (L) and High (H). 100, 110, 111, 113, 119, 139–141, 145, 153

PT Physical Topology. 105, 107, 108

RA Risk Assessment. 7–9, 13–15, 84, 95, 96, 98, 169, 170

RFID Radio Frequency Identification. 103

RTUs Remote Terminal Units. 22, 24, 31, 82, 83

SCADA Supervisory Control and Data Acquisition. 3, 12, 14, 22–31, 36, 39, 66, 68, 70, 76, 77, 79–81, 123, 131, 169

SingleCVSS A Single Vulnerability Vector. 113, 115, 138

SQL Structured Query Language. 26, 42, 43, 56, 68, 83, 139

SSL Secure Sockets Layer. 40, 42, 63, 159

SW Software. 2, 47, 87, 88, 92, 94, 100, 121, 171

TCP Transmission Control Protocol. 22, 27, 54, 56, 73, 81, 82

TLS Transport Layer Security. 42, 63, 159

UART Universal Asynchronous Receiver/Transmitter. 47, 56, 60

UI User Interaction (CVSS metric): None (N) and Required (R). 100, 110, 111, 113, 119, 139–141, 145, 153

UPnP Universal Plug and Play. 64, 136, 140

USB Universal Serial Bus. 41, 42, 47, 70, 71, 96

V2G Vehicle-to-Grid. 29, 33

VPN Virtual Private Network. 31, 33, 73, 74, 136, 159

WAN Wide Area Network. 22, 28, 29

WMD Wearable Medical Devices. 44, 46, 47

WPAN Wireless Personal Area Network. 51

ZLL ZigBee Light Link. 51, 57, 58, 61, 64

SECTION I
Foundations and Related Work

CHAPTER 1
INTRODUCTION

1.1 Critical Infrastructures

Critical Infrastructures (CIs) are considered as the fundamental pillars that human society depends on, in order to sustain its normality and well-being. Although people in nowadays tend to consider amenities such as healthcare, electricity, telecommunications and clean water as standard, it is not hard to imagine the impact that a disruption on just one of them could have to a modernized economy/nation and/or the ecosystem as well.

According to European Council[1] the term 'Critical Infrastructures' is defined as "*an asset, system or part thereof located in Member States which is essential for the maintenance of vital societal functions, 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*". In October of 1997, the United States of America (USA) published for the first time the booklet "Critical Foundation, Protecting America's Infrastructures" [132] whereas Cybersecurity and Infrastructure Security Agency (CISA) defines PATRIOT Act of 2001[2] the term of 'critical infrastructures 'as' *systems and assets, whether physical or virtual, so vital to the United States that the incapacity or destruction of such systems and assets would have a debilitating impact on security, national economic security, national public health or safety, or any combination of those matters*'.

US Presidential Policy Directive (PPD) 21[3] CIs defined the CIs for USA: Chemical, Commercial Facilities, Communications, Critical Manufacturing, Dams, Defense Industrial Base, Emergency Services, Energy, Financial Services, Food and Agriculture, Government Facilities, Healthcare and Public Health, Information Technology, Nuclear Reactors, Materials and Waste Transportation Systems as well as Water and Wastewater Systems.

In a similar approach European classification of European CIs (COM 2004 - 702 final) are categorized as follows:

- **Energy installations and networks** (e.g. electrical power, oil and gas production, storage facilities and refineries, transmission and distribution system)
- **Communications and information technology**
- **Finance** (banking, securities and investment)

[1] https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32008L0114&from=en

[2] https://www.congress.gov/107/plaws/publ56/PLAW-107publ56.pdf

[3] https://www.cisa.gov/publication/isc-ppd-21-implementation-white-paper

- **Healthcare** (e.g. hospitals, health care and blood supply facilities, laboratories and pharmaceuticals, search and rescue, emergency services)
- **Food** (e.g. safety, production means, wholesale distribution and food industry)
- **Water** (dams, storage, treatment and networks)
- **Transport** (airports, ports, intermodal facilities, railway and mass transit networks and traffic control systems)
- **Production, storage and transport of dangerous goods** (e.g. chemical, biological, radiological and nuclear materials)
- **Government** (e.g. critical services, facilities, information networks, assets and key national sites and monuments)

Europe together with USA have established several directives, frameworks [25] and sector specific guides in order to improve the resilience of CIs against natural disasters (e.g. earthquakes, hurricanes, floods, draught), accidental events (e.g. power outage, improper maintenance, unauthorized Hardware (HW) and/or Software (SW) modification/error/failure) and intentional actions such as tampering, theft and espionage. In particular, the European Commission (EC) proposed the Network and Information Security (NIS) directive[4] as part of the European Union (EU) cybersecurity strategy that aims in coordinating all member states to national supervision of the cybersecurity of critical market operators such as energy, transport, water, health, digital infrastructure and finance sector as well as digital service providers such as online market places and cloud and online search engines. Furthermore, a newer version of NIS directive (NIS 2 - COM/2020/823) is being developed by the EC, in order to address the new challenges and threats (e.g. to apply security on supply chains and supplier relationships):

- It broadens the scope of the existing NIS Directive to apply to "important sectors," such as waste management, postal services, chemicals, food, medical device manufacturers, digital providers and producers of electronics.
- It imposes new, specific cybersecurity requirements relating to incident response, supply chain security, encryption and vulnerability disclosure obligations.
- It aims to establish better cooperation and information sharing between EU Member States, and create a common European vulnerability database.

[4] https://eur-lex.europa.eu/eli/dir/2016/1148/oj

1.2 CIs, Industrial Control Systems (ICS) and IoT

Industrial revolutions changed completely the working habits and lifestyle of mankind. The 1st industrial revolution took place in the 18th century and managed to harvest the power of steam-engines in order to increase productivity and the transportation of goods to great distances. The 2nd began in the 19th century and included novelties such as the discovery of electricity and oil as well as internal combustion engines and the assembly production lines that help cut down the fabrication costs significantly. The 3rd revolution began in the early late 60s of the 20th century and included, among others, the introduction of nuclear energy and the fully automation of the production processes via electronics and Information and Communication Technology (ICT). In particular, during the last few decades, most CIs rely heavily on industrial control systems that include Supervisory Control and Data Acquisition (SCADA) and/or other types of control systems.

Traditional ICS follow a hierarchical design, have limited data storage capabilities, use a standardised communication interface named Object Linking and Embedding (OLE) / OLE for Process Control (OPC) which is based on the Distributed Component Object Model (DCOM) [225] with restricted interoperability features and until recently, were isolated from the Internet. Field devices such as sensors and actuators are usually connected to a Programmable Logic Controller (PLC) via a wired interface (e.g. serial), system software is mainly based on proprietary technology developed by each manufacturer with limited scalability capabilities whereas data collection and storage is restricted to the basics and takes place mostly on each installation site premises.

Launched in the early 2000s, 4th Industrial Revolution (4IR or *Industry 4.0*) [158] promotes interconnectivity and smart automation in order to transform legacy systems (e.g. electric power generators, heavy industrial machinery) into complex cyber-physical entities, which, aside physical interfaces, include advanced cyber characteristics such as embedded software, network interfaces and sensing capabilities. All this transformation is supported by the rapid evolution of hardware, including computing power and energy efficiency, software, wireless technologies (e.g. 5G, ZigBee, WiFi, Z-Wave) and network communication protocols. By incorporating these characteristics systems are becoming capable to communicate one another, interoperate, self adjust and being remotely accessed/managed. The main design principles in Industry 4.0 [123] can be identified as the following:

1. *Interconnection*: The connectivity characteristics that enable machines, devices, sensors and people to connect over the IoT and the Internet of People (IoP) to form the Internet of Everything (IoE).

2. *Information transparency*: Information gathered from physical (e.g. via sensors) and the digital world (e.g. office documents) must be accumulated via assistance systems and be available to all of the IoE participants.
3. *Decentralized decisions*: All IoE participants must perform their tasks as autonomously as possible by utilizing both local and global information that must be available in order to assist the former to make better decisions and to increase their overall productivity.
4. *Technical Assistance*: The assistance systems, must be capable of aggregating and presenting information understandably in order to facilitate humans to make informed decisions. Furthermore, the human role shifts from just an operator to a strategic decision-maker and a flexible problem solver that is able to collaborate with machines via machine-to-human interfaces.

On the other hand, IoT enabling technologies have inaugurated a plethora of interconnected "smart objects" that are present in all aspects of our everyday life. Internet connectivity is incorporated in a diversity of IoT products, installed in various domains, ranging from wearables (e.g smart watches, near-patience medical devices), home appliances, lighting, Heating, Ventilation, and Air Conditioning (HVAC) systems, up to Internet-enabled robotic machinery, water storage, supply and waste treatment systems, transportation (e.g. traffic control lighting systems), supply chain, in-hospital medical devices, industrial equipment and smart power grid automation systems and services. These systems are capable of communicating with heterogeneous types of devices and services, mainly due to the utilization of well-established or recently introduced network/application protocols such as the HyperText Transfer Protocol (HTTP/HTTPS), the Message Queuing Telemetry Transport (MQTT) - suitable for low-bandwidth, high latency, unreliable networks [249], the Extensible Messaging and Presence Protocol (XMPP) - for near real-time exchange of messages and presence notifications via the Extensible Markup Language (XML) streams, the Constrained Application Protocol (CoAP) - for use with limited-resourced nodes and networks and the Representational State Transfer (REST) Application Programming Interfaces (API) for interacting with Web services (RESTfull).

Regarding data analytics, IoT ecosystem favors long-term data retention that can be stored in cloud infrastructure and can be used with machine learning techniques for enabling automated responses, detecting anomalies, preventing equipment failures thus decreasing the downtime and extending the component's life. Stored data can be used also to improve decision making and introducing artificial intelligence features (e.g. self-driving cars, facial recognition technologies) to a wide area of everyday and mission-critical services whereas cross-platform, open standards like MQTT promote interoperability characteristics that facilitate IoT-enabled devices to to collect and exchange data from a variety of sources and different software platforms.

But all these new features and capabilities resulted in the rapid diversification of the threat landscape. New types of attack vectors emerged: A new era was about to begin.

1.3 Introducing the CI threat landscape: From physical to cyber-physical attacks

Attacks against critical infrastructures has been a continuously increasing concern, during the past decades. Until the '90s most concerns were related to environmental threats such as regional outages resulting from earthquakes, storms, and floods as well as man-made physical threats such as arson, sabotage and bombing.

But from the early '90s, the vast adoption of information technology in CIs had a serious effect on the existing threat landscape: Additionally to the increasing concerns due to terrorism as well as new attack vectors named 'Cyber threats', resulted from the expanding universe of Internet and the adoption of ICT-related technology, were raised. Terrorist attack issues were brought about mostly due to the proliferation of mass destruction weapons including chemical, biological, and even nuclear weapons. Furthermore, international terrorism, narcotics, trafficking, and transnational economic could be considered as some of the new challenges of the new threat landscape. In a US technical report [20] risks coming from recent terrorism attacks against the energy supply chain, production and distribution sector including sabotage of oil and gas transportation/storage systems, power lines, fuel tanker trucks and power substations were presented. Additionally, a report [229] published by the Presidential Commission on Critical Infrastructure Protection, showcased the potential impact due to the increasing mutual dependence and the interconnectedness among critical infrastructures. The authors emphasized on the increasing dependencies of critical infrastructures with the Public Telecommunications Network (PTN), the Internet and telecommunications industry. Concerning the threats from the Internet, the report referred as newly potential attack vectors the readily available information on sensitive topics (e.g. disclose to a terrorist the best place to set explosive charges for maximum disruptive effects), as well as the exposure of cyber vulnerabilities that can result in attacks on information and communication systems. The report also incorporated the results of a successful, three-month, black-box penetration testing on computers and networks of the US Department of Defense, privately owned energy companies and telecommunications service providers that showcased several high-severity vulnerabilities. The report also focused on how these changes affect the threat landscape and the CIs: The number of personal computers with access to the World-Wide Web (WWW) increased from just a few thousands in the early 80's to over 400 million in 1996. Similarly, at the same time-period, Local Area Networks (LANs) grow from thousands to millions, public Internet networks went from just a few hundreds to thousands whereas, experts and specialists on telecommunications systems control software equipped with the technical skills fit for a cyber attack scenario, spread worldwide. Moreover, researchers such as Furnell et. al. [93], demonstrated concerns regarding how information technology vulnerabilities can be used by cyber terrorists with devastating impact to the society. As of late '90s, the security community was also began to interested in potential cyber vulnerabilities found on CIs' industrial control systems. For example, in [56] authors present the potential impact of a biological warfare terrorist attack scenario on a water treatment facility.

The new millennium began with the discovery of Stuxnet worm [156]: The first high-profile, cyber-physical documented attack in which malicious code had been utilized in order to inflict physical damage to mission-critical industrial machinery. "Someone" unleashed a digital weapon on computers in Iran, a software so unique it would be named as the mankind's first cyberweapon thus announcing the age of digital warfare. The name of the worm comes out from the subset of the names of the driver files *.**stub*** and *mr**xnet**.sys*.

Today's black market is brimming from zero-day vulnerabilities, credential/company secrets harvesting tools, several payloads that weaponize exploits, ransomware kits and massive numbers of zombie computers and IoT devices for botnets. Bounty programs offered by software vendors like Google, Amazon, Microsoft and several antivirus companies also exist since, software companies are now willing to pay for new vulnerabilities found in their products. But with digital arm dealers and defence contractors driving the prices for zero-day vulnerabilities extremely high, security researchers and white hat hackers are discouraged from participating to bug-bounty programs [190].

The discovery of the Stuxnet was a turning point since, people realized the real potential of cyber warfare and the impact that may have on our society and to the physical world. Nowadays, the introduction of IoT-related technologies to CIs have further widen the threat surface, since, attacks can now be also IoT-enabled: The attacker initially exploits some vulnerable IoT technology in order to propagate to the actual target that is somehow connected and/or in physical proximity with the objective of the attack.

1.4 Research gaps, Thesis contribution and structure

1.4.1 Research gaps

Nowadays, well-established security standards (e.g. [38, 134, 235]) exist in order to manage and assess the risks related with ICT systems and services. Furthermore, several sector-specific security standards, frameworks and guidelines co-exist in order to help implement security countermeasures in specific environments.

But the evolution of the threat landscape due to the emerge of the IoT ecosystem enable adversaries to extend their reach by exploiting the interconnectivity, proximity and/or functionality features of IoT devices thus enabling them to create complex, subliminal, cyber-physical, attack paths. Identification and modelling of such diverse, stealthy attack vectors can be a daunting task especially in large-scale environments such as the one of a CI where diverse ICT technologies and mission-critical, cyber-physical ICS systems coexist in proximity and/or connectivity with off-the-shelf IoT devices.

Moreover, limited research work has been done so far regarding emerging threats from IoT-enabling technologies (e.g. [3, 17, 60, 65, 97, 150]). Additionally, even fewer research works that showcase risks and/or threats that originate from composite, cyber-physical, attack vectors ([2, 10, 60, 195]) exist. Some of the latter cannot even be considered as full RA methodologies whereas other ([2, 10]) come with several limitations, when applied to large-scale industrial environments mainly due the exponential increase of interactions/attack paths and the high percentage of false-positive results.

1.4.2 Thesis contribution

In order to address these open challenges and research gaps we first depict this new threat landscape by surveying, taxonomizing and analyzing recent real and PoC IoT-enabled attacks in popular CI sectors such as in industry, energy, transportation, healthcare as well as home environments.

Then, in order to showcase the potential impact (criticality) of such attacks, we propose a high level risk assessment framework that we then apply on the aforementioned attack scenarios, in a worst-case approach for each domain.

Moreover, to address the research gaps on identifying, modelling and assessing IoT-enabled complex, cyber-physical, attack vectors in large-scale environments, we develop a target-oriented, source-driven, low-level risk assessment methodology which we also test via realistic PoC scenarios on e-healthcare and urban environments. Finally, for all of the above, we propose mitigation strategies to address these risks.

1.4.3 Thesis structure

In the following paragraphs we describe in high-level the layout of the thesis. For better understanding, we have structured our work in eleven chapters that are grouped in five main sections. In particular:

Section	Chapter	Title	Corresponding Paper(s)	Contribution
I - Foundations and Related Work	**1**	Introduction		Description of the threat landscape evolution and research gaps
	2	Related work and overview of security organizations and standards	[258]	Survey of existing CI and IoT-related security standards as well as an analysis of research gaps
II - Introducing the threat landscape	**3**	IoT-enabled attacks on Cyber-Physical systems	[261, 262]	Present and taxonomize IoT-enabled attacks to depict current threat landscape
	4	Advanced Persistent Threats (APTs) in Industrial IoT ecosystem	[259]	Analyze high profile real and PoC attack scenarios for further understanding of complex cyber-physical attack vectors
III - Proposed Methodologies	**5**	A high-level, RA methodology for IoT-enabled attacks	[260, 261]	Create a high-level RA methodology that can be used to assess the criticality of ioT-enabled atatck scenarios
	6	A low-level, risk assessment methodology for complex, IoT-enabled, cyber-physical attack paths	[257]	Develop a low-level RA methodology that is capable of identifying, model and assess composite, IoT-enabled, cyber-physical attack vectors
IV - Methodology Validation	**7**	Assessing the criticality of IoT-enabled Attacks	[261]	Apply the high-level methodology on surveyed attacks
	8	Smart city PoC validation scenario	[262, 263]	Validate the methodology presented in Chapter 6 on a simplistic, yet realistic PoC scenario on urban environment.
	9	E-healthcare PoC validation scenario	[257]	Validate the efficacy of the methodology resented in Chapter 6 on E-healthcare sector.
V - Mitigation and Future Work	**10**	Risk mitigation of IoT-enabled attacks	[257, 261, 263]	Propose mitigation strategies for IoT enabled attacks
	11	Conclusions - Future work	[112]	In this chapter we conlude our work and present thesis's limitations and future work

Table 1.1: Thesis structure with the corresponding published research papers

- Section I: Foundations and related work
  - In Chapter 1 we delineate the definition of critical infrastructure, its corresponding sectors, the evolution of the threat landscape and thesis contribution.
  - In Chapter 2 we overview CI-related security standards, frameworks and related research work [258].
- Section II: Introducing the threat landscape
  - In Chapter 3, we review and taxonomize recent, both real as well as PoC, IoT-enabled, cyber/cyber-physical attack paths in various critical sectors including industrial control systems, Smart Grids, transportation, e-healthcare as well as smart home environments [261].
  - In Chapter 4, we further analyze advanced exploitation techniques and tactics of high-profile, real and PoC IoT-enabled attacks regarding industrial and energy sectors [259].

- Section III: Proposed frameworks and methodologies
  - In Chapter 5, we propose a high-level risk assessment methodology [260, 261].
  - In Chapter 6 [257], we develop a novel, low-level, risk assessment methodology for assessing IoT-enabled, cyber-physical attack paths against critical systems, based on the work presented in Chapter 5.
- Section IV: Methodology validation
  - In Chapter 7, we apply the high-level methodology proposed in [260, 261] to the attack scenarios presented in Chapter 3.
  - In Chapter 8, we identify, and assess smart light enabled attacks on several popular sectors of urban infrastructures and services [263] based on vulnerabilities found on a hands-on approach of popular smart lighting system [262].
  - Similarly, in Chapter 9 we proceed on validating the efficacy of the methodology described in Chapter 6 via a PoC scenario in e-healthcare.
- Section V: Mitigation strategies and future work
  - In Chapter 10, we propose several well-known and state-of-the-art mitigation strategies for the attack path scenarios presented in Chapter 3. Additionally, we demonstrate how the RA methodology in Chapter 6 can help prioritize mitigation actions based on the findings stressed in Chapter 9.
  - Finally in Chapter 11, we conclude our research and show the current limitations and highlight future challenges.

CHAPTER 2
RELATED WORK AND OVERVIEW OF SECURITY ORGANIZATIONS AND STANDARDS

2.1 Literature review on IoT-enabled attack path security assessment

In this Chapter, we are going to review relative research work in order to: (i) Provide a concrete description and categorization of relative efforts and (ii) further highlight the research gaps and therefore the contribution of this work.

Abie and Balasingham [1] have developed a risk-based adaptive security framework for IoT enabled e-Health Cyber-Physical (CP) systems that is used to estimate risk damages and future benefits using game theory and machine learning techniques which, in turn, enables the security mechanisms to adjust their security decisions accordingly. Liu *et al.* [173] propose a dynamical risk assessment method for complicated and constant changing IoT environments by adopting features from an *Artificial Immune System* such as the distributed and parallel treatment, diversity, self-organization, self-adaptation, robustness etc. Through packet inspection from agents, deployed in IoT systems, the proposed method can be used to identify abnormal behavior and responds by appropriately adjusting a predefined risk value.

Atamli and Martin [17] display use cases of IoT enabled attack scenarios (power management, smart car and healthcare) so as to identify potential threat sources and classes of attack vectors. Additionally, impact assessment applicable in device types such as Radio Frequency Identifiers (RFIDs), actuators and sensors as well as networking technologies is included. They also propose specific countermeasures that can reduce the risks evolved mainly in security and privacy. A management framework for IoT devices, called Model-based Security Toolkit (SecKit), used to evaluate security policies that protect user's privacy, is presented in [209]. Seckit has been integrated in a framework, proposed by the *iCore* project, which enables usage control and protection of user data. Then research includes a PoC in a smart home environment.

Corno *et al.* [60] present a design-time verification formal methodology for smart environments that takes as inputs user behavior, device/environment/context modelling in order to verify the desired behavior of IoT-enabled environments. IoTSAT, a framework for security analysis regarding of IoT networks is presented in [195]. Researchers, model the generic behavior of IoT based on device configuration, network topology, user policies and IoT-specific attack surface. The model is then used to measure system's resilience against potential attacks and identify threat vectors and specific attack techniques.

Kott *et al.* [150] describe Mission Impact Assessments (MIAs) in an effort to bridge the gap between operational decision makers and cyber-defenders. They managed to set a test-bed (*Panoptesec*) which is capable of emulating cyber/physical systems of an Italian water and energy distribution company as well as a prototype simulation platform named *Analyzing Mission Impacts of Cyber Actions* (AMICA) that simulate a military's air operations center. Among others, they managed to discover high number of hidden network dependencies that weren't identified by human operators, unnecessary large volume communications between Human-Machine Interfaces (HMIs) and field devices as well as attacks against specific nodes of the network that, when used in a timely manner could lead to devastating results. The researchers proposed an abstractive threat modeling (e.g. [151]), for both adversaries and defenders, and emphasized on the challenges involved when modeling large scale, diverse and complex networks.

A recent approach [65] about Medical Internet of Things (MIoT) points out difficulties, that traditional risk assessment methodologies face when used in non-stable environments, such as the MIoT, where devices maybe added, removed or have their configuration altered. For assessing and managing threats the researchers adopt HMG IS1 and ISO/IEC 27033 standards and an existing threat analysis from the Technology Integrated Health Management (TIHM) project. They taxonomize threats according to the severity level ranging from *very low* to *very high*, as well as the risk that emerges from IoT devices against other MIoT devices. In addition, for each MIoT device connected to the hub a multicheck process is proposed. Researchers in [18] proposed a risk-based access control model for IoT technologies. Real-time data from IoT devices are utilized to dynamically estimate security risks through an risk estimation algorithm. The proposed model is capable of monitoring and analyzing user behavior in order to detect abnormal action from authorized users.

Recent methodologies [97, 300] that utilize the Common Vulnerability Scoring System (CVSS) have been also proposed by a similar group of researchers. In [97], a framework for modeling and assessing the security of the IoT ecosystem is proposed. The framework consists of five phases: (1) Data processing, (2) security model generation, (3) security visualization, (4) security analysis, and (5) model updates. In phase one system information and security metrics are introduced in order to construct the IoT network which is then used (phase two) to construct the extended Hierarchical Attack Representation Model (HARM) [125] and calculate all possible attack paths in the IoT network. During phase three, attack graphs (in low, upper and middle layer) are utilized to visualize the IoT network whereas in four a security analysis, that takes into consideration e.g. nodes or vulnerabilities, is constructed and fed into an analytic modeling and evaluation tool (*Symbolic Hierarchical Automated Reliability and Performance Evaluator - SHARPE* [238]) for further security analysis. Finally in phase five proper defense strategies are decided. The researchers present scenarios such as a Sinkhole attack [184] in a smart home environment, wearable healthcare and environmental monitoring. According to the researchers the limitations in presented attack scenarios include the difficulty to depict all diverse connectivity paths, no-connectivity attack scenarios (e.g. Distributed Denial-of-Service - DDoS) heterogeneity on communication protocols and static network topology.

Dorsemaine *et al.* in [73] assess the risks introduced to a legacy Information System (IS) due to the integration with the IoT infrastructure. A practical example is then presented regarding a smart lighting system in a company's Information Technology (IT) systems. The authors divide the IS into local environment, transportation, storage, mining and provision sectors. Then, they define security properties for the IoT systems of each IS sector, by focusing mainly on aspects such as confidentiality, integrity, availability, usability and auditability while also introduce additional properties for IoT components including energy, communication, functional attributes, local user interface and hardware/software resources. Finally they present the potential threats and the impact in all of the aforementioned attributes for an IS and IoT infrastructure.

Agadakos *et al.* [2] proposed a methodology for modeling cyber-physical attack paths in IoT. In particular, they developed a framework that allowed the identification of IoT device types, interaction channels, as well as security and proximity features. Using the proposed framework they managed to simulate a home network that consisted of several home IoT devices. By using techniques such as passive sniffing for host discovery, they managed to discover attack scenarios that utilized hidden connectivity/interaction paths, security degradation (e.g. from authenticated to unauthenticated communication channels) and violations of transitions and states.

A vulnerability-based risk assessment regarding *edge computing* and IoT was presented in [98]. Authors proposed a multi-attacker, multi-target graphical model that included attackers, targets, vulnerability relations in the network in order to assess the risk at the edge computing devices and apply the corresponding mitigation strategies. Ghazo *et al* presented a tool for automatic attack graph generation for computer and SCADA networks [3]. The authors tested their proposed algorithm in a PoC scenario regarding a water treatment facility.

Sequeiros *et al* [246] present related work on attack and threat modelling for IoT systems and cloud mobile applications whereas in [10] the authors present *IotCom*, an approach to discover hidden threats. In particular, the researchers analyzed multi-app coordination threats that can trigger infinity activation loops or chain coordination events that can lead to race conditions and physical wear of a device. Via their platform they were able to perform static analysis of multiple IoT applications and detect several events of safety violations.

2.2 Analysis and comparison of related work

Technology evolution has introduced new cyber/physical attack vectors that are hard to identify and assess. The entanglement of IoT-enabling technologies with air-gaped, legacy, cyber/physical systems, especially in large-scale, complex environments, such as in critical infrastructures, has made the task of assessing the risk in one of these domains a daunting task to begin with, even when using well-established risk assessment methodologies (e.g. [38]). In order to showcase the research gaps, we first define specific individual characteristics that an low-level risk assessment methodology must incorporate in order to be able to cope with the current trends. The attributes were chosen based on individual factors that are presented in each of the aforementioned research work in Section 2.1 and well established methodologies. These characteristics are presented in the following Table (2.1).

In order to showcase the research gaps of all related work presented in Section 2.1 we apply the characteristics presented in Table 2.1 (see Table 2.2).

Characteristic	**Description**
Type	The type of the research work as defined by the researchers: Framework, methodology/method or model
Sector Specific (SSp)	Applicable to a specific sector (e.g. Healthcare) or sector-agnostic
Case Study (CS)	If there exists a case study scenario in order to prove its efficacy
Tool	Corresponding software for implementing the methodology/framework/model
Level (LvL)	Whether the RA is high level or incorporates detailed information
Interaction Modelling (IntM)	Feature for modelling custom interaction types
Cyber / Cyber-Physical (C/CP)	The types of interactions supported (cyber, physical)
Interaction Assessment (IntA)	Support for validity detection mechanisms of each individual interaction prior constructing complex attack vectors in order to reduce false-positives, complexity and therefore computational cost
Vulnerability Chaining (VCh)	Combine more than one vulnerabilities in order to perform a composite vulnerability CVSS vector in order to maximize impact on target (e.g. Section 3.4 of [85])
Vulnerability Assessment (VA)	Identify and assess the severity of all individual/chained vulnerabilities for all devices and attack paths in scope
Security Controls (SC)	Adjust vulnerabilities foound by taking into consideration existing physical/network/application security controls
Threat modelling (TM)	Applicable threat actors' types along with their corresponding characteristics (e.g. skillset, resources, access, environment)
Impact Assessment (ImpA)	Assess the businesswise impact for each target
Threat assessment (TA)	Assess who of the applicable threat actors correspond to which attack scenario (e.g. based on skillset, access, environment) as well as its businesswise likelihood
Attack Paths (AP)	Construct complex attack path scenarios
Risk assessment	Calculation of risk for each attack path scenario
Risk Treatment (RT)	Prirotize security countermeasures for all identified risks

Table 2.1: Individual characteristics of an RA methodology/model/framework

By examining Table 2.2, we can observe that most of the resent works are focused on identifying and assessing cyber interactions due to interconnectivity among IoT

Ref.	SSp	Sector	CS	Tool	LvL	Type	C/CP	IntM	IntA	TM	VA	VCh	ImpA	SC	AP	RA	RT
[1]	✓	e-health	✗	✗	H	FrmWrk	C	✗	✗	✓	✗	✗	✗	✗	✗	✓	✗
[17]	✗	✗	✓	✗	H	Model	C	✗	✗	✓	✗	✗	✓	✓	✗	✓	✗
[209]	✗	✗	✓	✓	D	FrmWrk	C	✓	✗	✓	✓	✗	✗	✓	✗	✗	✗
[65]	✓	e-health	✓	✗	H	Method	C	✗	✗	✓	✓	✗	✗	✗	✗	✗	✗
[18]	✗	✗	✗	✗	H	Model	C	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗
[97]	✗	✗	✓	✓	D	FrmWrk	C	✓	✓	✓	✓	✗	✓	✓	✓	✗	✗
[300]	✗	✗	✓	✓	D	FrmWrk	C	✓	✓	✗	✓	✗	✗	✓	✓	✗	✗
[73]	✗	✗	✓	✗	H	Method	C	✗	✗	✓	✗	✗	✓	✗	✗	✗	✗
[173]	✗	✗	✗	✗	H	Method	C	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗
[150]	✗	✗	✓	✗	H	Method	C	✓	✗	✓	✗	✗	✓	✗	✓	✗	✗
[2]	✗	✗	✓	✓	D	Model	**CP**	✓	✗	✓	✗	✗	✗	✓	✓	✗	✗
[195]	✗	✗	✓	✓	D	FrmWrk	**CP**	✓	✗	✓	✗	✗	✗	✗	✓	✗	✗
[60]	✗	✗	✓	✓	D	Model	**CP**	✓	✗	✗	✗	✗	✗	✗	✓	✗	✗
[10]	✗	✗	✓	✓	D	Model	**CP**	✓	✗	✓	✗	✗	✗	✓	✓	✗	✗
[98]	✗	✗	✓	✗	D	Model	C	✓	✓	✗	✓	✗	✓	✓	✓	✓	✓
[3]	✓	SCADA	✓	✓	D	Model	C	✓	✓	✗	✓	✗	✗	✓	✓	✗	✗

Table 2.2: Analyis of the related work, using the characteristics defined in Table 2.1

devices and legacy ICT systems. Most of the examined cyber-oriented RA methodologies do not examine existing vulnerabilities and/or logical access among IoT devices or how they can be combined in order to escalate an attack. Additionally, only in [1, 17, 18, 98, 173] the overall risk is calculated. Moreover, from the aforementioned works only the following ([2, 10, 60, 195]) take into consideration except from cyber, interactions with the physical world as well. In particular, the majority of the latter are not defined as complete risk assessment methodologies but as models ([2, 10, 60] or frameworks ([195]) since, each individual one focuses on specific areas (e.g. operational safety [60]) and cannot be considered as a full RA methodology.

In particular, the proposed design time verification methodology presented in [60] is mainly designed in order to be used during the implementation phase of an IoT-enabled environment. Although it takes into consideration cyber - physical inputs/outpust (e.g. proximity sensors) and device states (e.g. door open - closed) it can not be perceived as an RA methodology since its main purpose and is to verify the correctness, reliability, safety, security and desired behavior of IoT-enabled systems installed in smart environments. Additionally, the IoTSAT framework in [195] focuses only on identifying IoT-related threats based on high-level, functional dependencies, network topology and environment couplings regarding network configurations in order to calculate the threat-resilience (i.e. the minimum number of devices and links required to be compromised for the attack to succeed), the potential threat vectors and the specific attack techniques that can be used in order to achieve adversary's end goal. This approach can identify a chain of events (complex attack vectors) based on cyber (network connectivity), as well as physical functionality features (e.g. the ability of a controller to turn the HVAC *on* but does not validate the feasibility of each individual attack (interaction) neither does examine how the existing vulnerabilities/security controls affect the overall attack path scenario (it assumes that the attacker is capable specific subset of devices and that all devices are adequately hardened so they cannot be directly compromised). Additionally, this approach cannot be identified as a RA methodology since it does not calculate the risk involved for each attack scenario.

Limitations stated by the authors in [2], included the false positives results that their model may introduce, since it does not filters unrealistic attack scenarios and that their methodology cannot be implemented in large scale networks due to complexity restrains. Finally, in the most recent work ([10]) authors are mainly focusing on detecting safety and security violations due to interaction of multiple IoT applications and their embodying physical environment that can lead to race conditions and physical wear of an IoT device.

2.3 An overview of Standards, Frameworks and Guidelines for CI protection

Cybersecurity organizations both in Europe and the US begin to recognize the security challenges involved in the IoT ecosystem, especially when these technologies are used in critical infrastructures. In the next paragraphs we present the most important organizations as well as overview the most prominent security standards. Existing security risk and threat assessment methodologies focus on a series of factors such as: (i) The assets that need to be protected, (ii) the threats and vulnerabilities that correspond to these assets, (iii) their value to the organization under assessment, (iv) the consequences (or impact) in case of security violations against the identified assets and (v) security controls that can reduce/eliminate the potential damage. The main goal of a risk assessment methodology is to provide guidance to an organization in order to minimize the risk and maximize the level of confidentiality, integrity and availability of its systems. Implementing the appropriate security measures must be done in respect of each organization's needs while, at the same time, guarantee a satisfactory level of functionality. Additionally, except from the sector-agnostic, several well-established , IoT-specific RA methodologies have been developed in the last few years due to the ever growing risk that stems from the IoT systems and services. In the next paragraphs we are going to briefly present existing security organizations and relative standards, guidelines and RA methods.

European Reference Network for Critical Infrastructure Protection (ERN-CIP) aims at providing a framework within which experimental facilities and laboratories that will share knowledge and expertise in order to harmonize test protocols throughout Europe, leading to better protection of CIs against all types of threats and hazards[1]. It is comprised of thematic groups such as *Chemical and Biological (CB) Risks to Drinking Water*, *Detection of Explosives and Weapons at Secure Locations*, *Detection of Indoor Airborne Chemical-Biological Agents*, *IACS Components Cybersecurity Certification Scheme*, *Radiological and Nuclear Threats to Critical Infrastructure* and *Resistance of Structures to Explosion Effects*. ERNCIP also publishes several reports regarding security such as hot to implement Industrial Automation and Control Systems Components Cybersecurity Certification Scheme (ICCS), guidance on the production of a water security plan for drinking water supply and many more.

European Network and Information Security Agency (ENISA) is considered to be the center of network and information security expertise for the EU, its member states, the private sector and Europe's citizens[2]. The organization cooperates with these groups to advice and recommend good practices in information security. It also assists EU member states in implementing relevant EU legislation and works to improve the resilience of Europe's critical information infrastructure and networks.

The organization has published several guidelines regarding ECIs protection[3] including *Smart Grid Security Recommendations* (2012), good practices for Computer Emergency Readiness Team (CERTs) in the field of industrial control systems (2013), security measures and communication network interdependencies in respect of Smart Grids (2013/2016), cybersecurity and resilience for Smart Hospitals (2016/2020/2021), good practices for cybersecurity or cyber risk management in the maritime sector (2019/2020), railway cybersecurity guidelines (2021) and achieving cyber Resilience in the Finance Sector (2021).

Department of Homeland Security (US) works towards improving security of the United States[4]. It's work includes customs, border, and immigration enforcement, response to natural and manmade disasters, antiterrorism, and cybersecurity. Particularly, it evaluates national capabilities, opportunities, and challenges regarding the protection of CIs, analyzes threats, vulnerabilities, and potential consequences of critical infrastructure and identifies security and resilience functions that are necessary for effective public-private engagement with all critical infrastructure sectors. Publications include a guidebook that provides information for enhancing the security of ICS including administrative controls, architecture design, and security technology.

[1] https://erncip-project.jrc.ec.europa.eu/european-reference-network-critical-infrastructure-protection

[2] https://www.enisa.europa.eu/

[3] https://www.enisa.europa.eu/topics/critical-information-infrastructures-and-services

[4] https://www.dhs.gov/

Cybersecurity and Infrastructure Security Agency (CISA) A United States federal agency[5], under Department of Homeland Security oversight. Its activities are a continuation of the National Protection and Programs Directorate (NPPD). NPPD was formed in 2007 as a component of the United States Department of Homeland Security. CISA was established on November 16, 2018 via the the Cybersecurity and Infrastructure Security Agency Act of 2018. It publishes emergency directives and alerts[6] and sector-specific security plans including, among others, chemical, commercial facilities, communications, critical manufacturing, dams, defense industrial, emergency services, energy, financial services, food and agriculture, healthcare and information technology. Other services and tools may include the development of cybersecurity exercises, incident response, information sharing, risk assessment and even remote penetration tests.

International Organization for Standardization (ISO) ISO is an independent, non-governmental organization[7] that develops and publishes worldwide technical, industrial and commercial standards. It is headquartered in Geneva, Switzerland and its standards are adopted among 165 countries. It is affiliated with the International Electrotechnical Commission (IEC) via a Joint Technical Committee (JTC) in order to develop standards relating to information technology. The organization also publishes several technical reports, specifications and guides. In order for a standard to considered as final it must passes several stages. In particular, each standard first undergoes the *Proposal stage* (including preliminary work), then moves on to the *Preparatory stage* and after that the *Committee stage* where draft versions of the standard are produced. Furthermore, the standard enters the *Enquiry stage* which is followed by the *Approval stage* (final draft) and finally the *Publication stage* at which the standard is officialized. Each standard is the reviewed at regular intervals and can be withdrawn if no longer needed or superseded by another standard.

The International Electrotechnical Commission[8] is an international standards organization that prepares and publishes international standards for all electrical, electronic and related technologies. IEC standards cover a vast range of technologies from power generation, transmission and distribution to home appliances and office equipment, semiconductors, fiber optics, batteries, solar energy, nanotechnology and marine energy as well as many others. IEC/ISA99, industrial automation and control systems security committee's[9] purpose is to improve the confidentiality, integrity, and

[5] https://www.cisa.gov/

[6] https://www.cisa.gov/uscert/ncas/alerts/aa22-011a

[7] https://www.iso.org/home.html

[8] https://iec.ch/homepage

[9] https://www.isa.org/standards-and-publications/isa-standards/isa-standards-committees/isa99

availability of components or systems used for industrial automation and control and provide criteria for procuring and implementing secure control systems. Compliance with the Committee's guidance will improve system electronic security, and will help identify vulnerabilities and address them, thereby reducing the risk of compromising confidential information or causing degradation or failure of the equipment or process under control.

ISO/IEC are considered among (if not the most) popular standards regarding information security. The organization publishes several ISO/IEC (60) guidelines, vocabularies, codes of practise and internal documents that aim to address a variety of security-related issues, such as network/application security (27033/34 series), incident management (27035 series), supplier relationships (27036 series) and/or sector-specific guidelines such as 27017/8 (cloud infrastructure) and 27011 for telecommunication companies. ISO/IEC 27001:2013 specifies the requirements for implementing an Information Security Management System (ISMS) that involves processes, documents, technology and human resources in order to manage, monitor, audit and improve an organisation's information security posture. ISMS can assist an corporation to protect different types of information including digital (on-premise, cloud), paper (e.g. printed documents) and intellectual property. In addition, ISO 27002 is designed to be used as a reference to ANNEX A of 27001 during the selection of applicable security controls related to specific areas described in 27001. Furthermore, ISO/IEC 27005:2018 can be utilized to assist the organization in the area of information security risk management in compliance with ISO 27001. It contains a range of guidelines regarding the formal identification, assessment, evaluation, and treatment of information security vulnerabilities thus ensuring that organisations plan, execute, administer, monitor, and manage their information security controls with their corresponding information security risks.

Other standards regarding CI security are *IEC 62351* (security in energy management systems an associated data exchange) and a series of standards *ISA99/IEC 62443* that includes several technical reports to secure Industrial Automation and Control Systems (IACS). IEC 62443 addresses not only the technology that comprises a control system, but also the work processes, countermeasures, and employees. The standard takes a holistic approach since not all the risks are technology-based: The staff responsible for an IACS must have the required training, knowledge and skills to ensure security. The standard is divided to 4 parts: General, terminology, concepts and models, policies and procedures, security requirements at the system level and detailed security requirements for IACS products.

National Institute for Standards and Technology (NIST) NIST's[10] main goal is to promote U.S. innovation and industrial competitiveness by advancing measurement science, standards, and technology in ways that enhance economic security and improve our quality of life. It maintains ongoing contact with a broad spectrum of users through a variety of means, including, but not limited to, public meetings, public workshops, individual contacts, and formal and informal collaborations and partnerships, in order to ensure that the information it disseminates continues to remain relevant. NIST attends and holds public workshops, conferences, and meetings to gather input about what types of information would be useful to industry; universities; other not-for-profit entities; and Federal, state, and local governments; and maintains memberships in many industry groups for the purpose of facilitating such discussions.

The President of US issued the Executive Order (EO) 13636 "Improving Critical Infrastructure Cybersecurity" in February 2013 via which NIST was obliged to work with stakeholders to develop a cybersecurity framework – based on existing standards, guidelines, and practices - for reducing cyber risks to critical infrastructure. The constructed framework consists of *Functions* (Identify, Protect, Detect, Respond, and Recover), *Categories* (e.g. Asset Management, Access Control), *Subcategories* (e.g. External information systems are catalogued, Data-at-rest is protected) and *Informative References* (specific sections of standards, guidelines, and practices).

Similarly to the ISO/IEC 27000-series, NIST proposes several security standards and guidelines including the widely adopted standard SP 800-31 Rev1 named as '*Guide for Conducting Risk Assessments*', SP 800-39 - '*Managing Information Security Risk: Organization, Mission, and Information System View*', SP 800-53 Rev. 5 - '*Security and Privacy Controls for Information Systems and Organizations*', SP 800-82 Rev.2 - '*Guide to industrial control systems security*' as well as sector-specific guidelines such as NISTIR 7628 Rev. 1 and NISTIR 7176 -2004 for Smart Grid and ICS respectively.

Institute of Electrical and Electronics Engineers (IEEE) IEEE is a professional organization[11] that is responsible for almost one-third of the technical literature in the world each year in areas such as electrical engineering, computer science and electronics with purpose to advance technological innovation and excellence. IEEE 1402-2000 a *"guide for electric power substation physical and electronic security"* describes recommended practices for the physical security of electric power substations and is designed to address a number of threats, including unauthorized access to

[10] https://www.nist.gov/

[11] https://www.ieee.org/

substation facilities, theft of material, and vandalism. The IEEE 1686 standard for Intelligent Electronic Devices (IEDs) addresses security regarding the access, operation, configuration, firmware revision, and data retrieval including the encryption for the secure transmission of data to the IED.

Centre for the Protection of National Infrastructure (CPNI) The CPNI is established in the United Kingdom (UK)[12] and aims in protecting national security by providing advice to the organizations that make up the UK's national infrastructure. It covers physical, personnel and cyber security aspects and collaborates with The security service and the government communications headquarters. It publishes several guidance and regulation documents, a catalogue of security equipment (e.g. access control, detection and tracking systems), current threat landscape and several security projects and initiatives.

North American Electric Reliability Corporation (NERC) NERC is a not-profit international regulatory authority with its mission the effectiveness and efficiency of minimization of risks for establishing the reliability and security of the grid. It develops and enforces reliability standards, educates, trains, and certifies industry personnel. It's area of responsibility includes US, Canada and the northern portion of Baja California, Mexico. It is considered as the Electric Reliability Organization (ERO) for North America, subject to oversight by the Federal Energy Regulatory Commission (FERC) and governmental authorities in Canada (over 400 million people aprox.).

NERC has issued several security standards regarding Smart Grid's protection including CIP-002-5.1 for '*Bulk Electric System (BES) Cyber System Categorization*', CIP-003-8 regarding '*Security Management Controls* and the management of security systems (CIP-007-6), security in the area of personnel training (CIP-004-6), CIP-005-6 for '*Electronic Security Perimeter(s)*', CIP-006-6 - '*Physical Security of Bulk Electric System (BES) cyber systems*', incident reporting and response planning (CIP-008-6), CIP-009-6 regarding '*Recovery Plans for BES Cyber Systems*', configuration change management and vulnerability assessments (CIP-010-3), information protection (CIP-011-2), CIP-013-1 that focuses on the supply chain risk management as well as CIP-014-2 that addresses the challenges in the area of physical security.

Greece's organizations regarding CI protection In Greece, the COUNCIL DIRECTIVE 2008/114/EC[13] was incorporated to the national legislation framework

[12] https://www.cpni.gov.uk/

[13] https://eur-lex.europa.eu/legal-content/en/TXT/?uri=CELEX:32008L0114

via Presidential directive 39/2011. Initially, the Center for the Security Studies (KEMEA)[14] was considered as the only research and consulting body of the Ministry of Citizen Protection (former Ministry of Public Order and Citizen Protection) regarding CI Protection as well as the contact point with the European Commission relevant authorities and the EU Member-States. NIS directive[15] was incorporated to the national legislation framework via 4577/2018 law. Additionally, the Greek Cybersecurity Authority[16], responsible for the enforcement of NIS directive regarding the Greece's critical infrastructure, was established in the Ministry of Digital Governance via the 40/2020 Presidential directive.

[14] http://www.kemea.gr/en/kemea/critical-infrastructures

[15] https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex:32016L1148

[16] https://mindigital.gr/kyvernoasfaleia

SECTION II
Introducing the threat landscape

CHAPTER 3

IOT-ENABLED ATTACKS ON CYBER-PHYSICAL SYSTEMS

3.1 Industrial control systems overview

Industrial control systems usually usually come with with high-availability requirements that depend on a variety of endpoint devices in order to determine the current status of a production process whereas SCADA systems are considered as ICS for monitor and control of such processes. The latter range from small-scale, single-sited systems (e.g. a small manufacturing plan) up to large-scale distributed systems spanning in a large geographic area that referred as Distributed Control Systems (DCS) [36].

A typical SCADA architecture consists of one or more distributed supervisory computers, also called Command and Control (C&C) centers and a number of IEDs, such as PLCs and Remote Terminal Units (RTUs) connected in an hierarchical model (see Figure 3.1). Intelligent electronic devices are used to supervise and control the industry plant through a diverse set of field devices, e.g., sensors, actuators, motors, drives and robotics. In the upper level, the C&C centers consist of Master Terminal Units (MTUs) and Personal Computer (PC) type workstations which gather and process data from the IEDs in order to send commands to the field devices. Operators monitor and control the system through HMI displays, distributed in the C&C center. Other computers may exist in the SCADA network, such as application and database servers for data storage and processing [5].

Given that the geographical area of a SCADA system may significantly vary, from the premises of a small factory up nation state and/or worldwide geographic areas (e.g. a country's electricity transmission infrastructure), the SCADA systems may interconnect via LAN and/or Wide Area Networks (WANs). The communication infrastructure can incorporate frame relay networks, satellite, radiowaves, dedicated lines, power lines or any combination of the above. To overcome network heterogeneity issues, various communication protocols have been adopted in SCADA networks, including Ethernet/IP, Modbus/Transmission Control Protocol (TCP) [267], Distributed Network Protocol 3 (DNP3) [57], IEC-104, DeviceNET [30], ControlNET [40], Highway Addressable Remote Transducer (HART)and WirelessHART [216] and ISA100.11a. Newly introduced network protocols have also been introduced in traditional SCADA ecosystem including MQTT, Routing Protocol for Low-Power and Lossy Networks (RPL) [288], IEEE 802.15.4x, IPv6 over Low -Power Wireless Personal Area Networks (6LowPAN) [201] as well as application-layer protocols like the CoAP and RESTful API Modeling Language (RAML) [163].

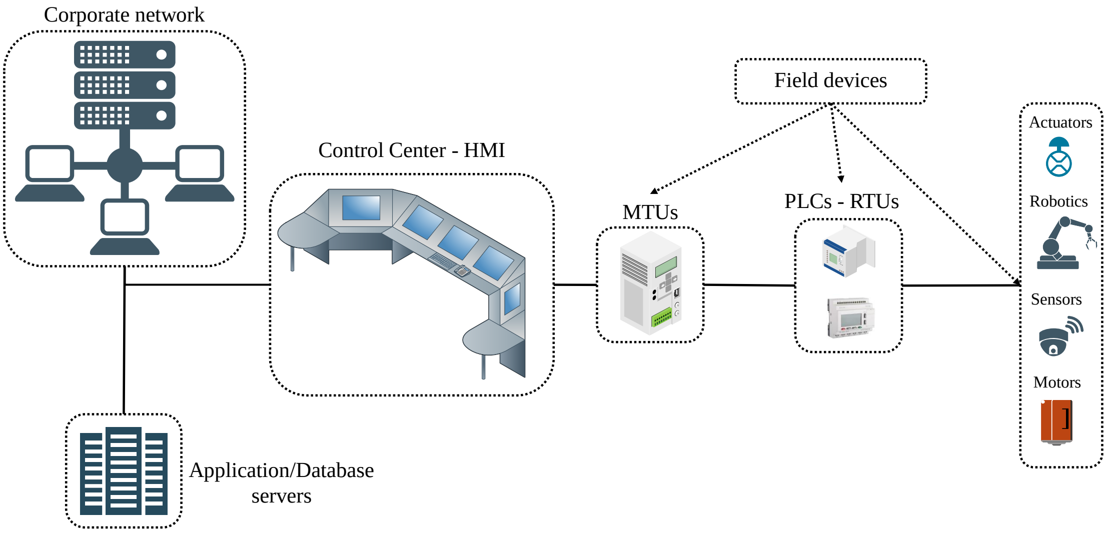


Figure 3.1: A typical SCADA architecture [259]

3.2 Industrial Internet of things

IoT enabling technologies in industry (IIoT) is revolutionising SCADA in terms of standardisation and openness. Most prominent traditional communication schemes such as COM, DCOM and OLE [175], that were mainly utilized for exchanging data between field devices of different manufacturers, come with several limitations regarding the collection and/or exchange of real time data among devices. Newly developed standards such as the Open Platform Communications Unified Architecture (OPC-UA) [103] promote a platform-independent, scalable, object-oriented, client-server architecture suitable for remote, bi-directional communications between diverse types of devices such as PLCs and Programmable Automation Controllers (PAC).

IIoT is also act as enabler for interoperability, scalability and Data Analytics, thus helping to further reduce hardware and software costs for businesses by eliminating the need of software licensing and upgrading via Internet connectivity and cloud services. IIoT related technologies enable users to easily add appliances at ease, are capable to process large amounts of data that are remotely accessible via Internet-facing HMIs throughout the world. This, in turn, enables the utilization of technologies such as big data analytics for equipment efficiency improvement as well as product cost and equipment maintenance/downtime reduction.

In addition, integration of SCADA and IoT related technologies have taken away the need for humans to constantly monitor and intervene with industrial processes by promoting Machine-to-Machine (M2M) communications and software that incorporates machine learning algorithms with artificial intelligence features, which, in turn, field devices can utilize for self-adjustment, error correction and adaptation.

The IoT ecosystem involves sensors and actuators that communicate with physical systems, in order to improve and optimize real-time operations in every aspect of our daily life. This may involve everyday objects, such as home appliances that are controlled through mobile smartphones, up to large-scale infrastructures, like power grids and industrial systems [8, 63] that may be managed through Internet-connected control systems.

3.3 IoT-enabled attacks on industrial control systems

As presented in Section 1.3 cyber as well as cyber-physical attacks have become the norm against CIs nowadays. This new era raises new security challenges. For example, SCADA systems, that until recently were isolated from the cyber world, are now becoming a part of it [7]; at the same time, existing security technologies are inadequate to protect these infrastructures in this fast-evolving threat landscape. The annual reports published by ENISA [77] and the ICS Cyber Emergency Response Team (ICS-CERT) [129], clearly underline the current vulnerabilities of the heterogeneous communication systems in charge of controlling and supervising critical infrastructures. Adversaries, on the other hand, find this new opportunity as a means to inflict maximum damage with minimal effort [44] since, they can stealthily exploit cyber-physical systems of strategic importance in diverse sectors, including energy [104, 160], supply chain management [26] and smart cities [58, 263].

Attacks that target resource-constrained IoT devices have multiplied over the last years [208]. Security vulnerabilities are continuously being discovered in IoT technologies used in both industrial (e.g. sensors and actuators) and home environments (e.g. home appliances, implantable medical devices, etc). Defects and misconfiguration in software applications [84], faulty hardware chips [233] and easy to tamper with devices [49, 262] are making the present situation even more dramatic. In the following sections we taxonomize IoT-enabled attacks on SCADA systems according to the target attack surface. This can be accomplished by either targeting directly the Internet-connected SCADA control devices, e.g. IEDs, PLCs, RTUs, or by first compromising a workstation of the upper SCADA layers, e.g. corporate IT network, control center, and then using that machine as backdoor into the control network. In another case, especially for industrial systems that include IoT-enabled intelligent field devices, the attacker may attempt to directly compromise the end devices. Thus, the attacks are categorized on those that target: (i) The corporate IT network or SCADA control center, (ii) the Internet- or IoT-enabled SCADA PLCs and (iii) the IoT-enabled field devices (see Figure 3.2).

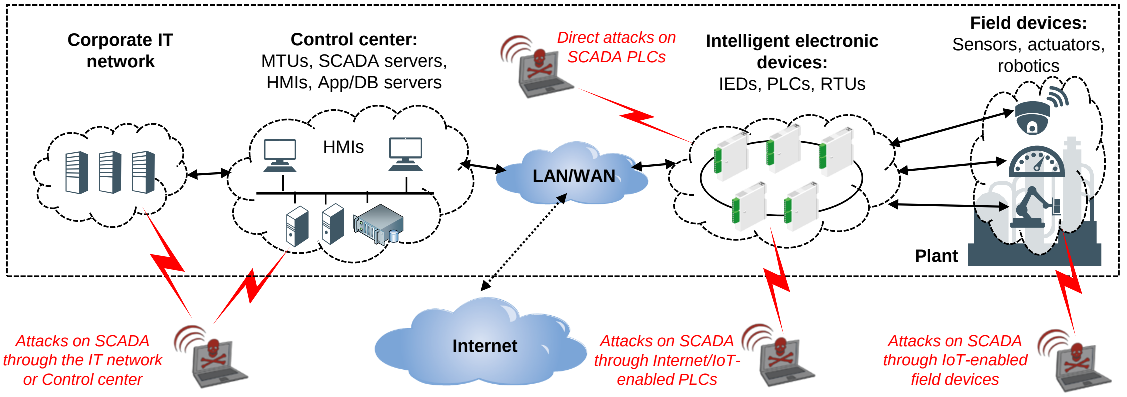

Corporate IT network
Control center:
MTUs, SCADA servers, HMIs, App/DB servers
HMIs
Direct attacks on SCADA PLCs
Intelligent electronic devices:
IEDs, PLCs, RTUs
Field devices:
Sensors, actuators, robotics
LAN/WAN
Plant
Internet
Attacks on SCADA through the IT network or Control center
Attacks on SCADA through Internet/IoT-enabled PLCs
Attacks on SCADA through IoT-enabled field devices

Figure 3.2: IoT-enabled attacks on industrial SCADA systems. Internet connectivity and the interconnection of IoT-enabled PLCs and field devices extend the attack surface [261]

3.3.1 Attacks through the IT network or the control center

Stuxnet worm, reported on June 2010, caused perhaps the most famous cyber-physical attack against critical industrial SCADA systems [81, 156]. The 500-kilobyte computer worm, infected the software of at least 14 industrial sites in Iran, including an uranium enrichment plant, as well as over 200,000 computers globally causing 1000 machines to physically degrade. The attack vector mainly consists of three stages. The malware was introduced to the IT network, probably through spear phishing techniques or through physical access. Then the worm exploited various Windows vulnerabilities and repeatedly replicated itself, seeking for its target software named Siemens Step7, a Windows-based application that is used to program PLCs. Finally, after compromising the PLCs that control the centrifuges, it slightly increased their spinning speed, leading slowly to their complete brake-down. The attack preparation is estimated to a few years and required very high expertise and resources. Stuxnet is considered as an Advanced Persistent Threat (APT) [252, 259] and will be further analyzed in Chapter 4.

In 2013, a security company Trend Micro deployed an ICS-like network of *Honeypots*, i.e. virtual systems that mimic actual ICS systems, in eight different countries in order to gather data of real attacks [287]. From March to June 2013 they observed 74 attacks originating from 16 countries (about 58% of these originated from Russia) with 11 attacks considered as critical. Most critical attacks were identified by alerts triggered when an unauthorized Modbus client attempted to read or write to PLC devices. Most of these attacks gained access to the Modbus by first compromising components of the C&C center. The HMIs were proven to be the gateway to the SCADA systems in several cases. Attackers attempted to exploit HMIs through typical Web attacks like Structured Query Language (SQL) injection, CSRF (cross-site request forgery) and dictionary attacks [285]. Since Modbus protocol does not require authentication [170], a compromised HMI can be used to send valid commands to the PLCs. Note that in most cases the reconnaissance of the honeypot was achieved via an online search through the Shodan IoT search engine[1]. One of the most interesting attacks, against a Japanese honeypot, is analyzed and assessed in Table 7.1.

3.3.2 Attacks through IoT-enabled PLCs

In [90] a research group created a self-spreading ransomware worm, named *LogicLocker*, that could infect three popular Internet-connected PLCs (Modicon M221, an Allen Bradley MicroLogix 1400, and a Schneider Modicon M241). More than 1500 devices of the PLC models, that were proven susceptible to this specific ransomware attack, were discovered through the Shodan search engine. The infected PLC was used as backdoor into the internal SCADA network and was able to infect with ransomware other PLCs of the same vendor. Except from the initial infection, various techniques were used to prevent quick restoration such as PLC access locking and PLC program encryption. Then, a small scale ransomware attack was demonstrated: In a simulated environment of a city water treatment plant [29] a malicious actor compromises the control PLCs and threatens to release large amounts of chlorine into the water unless the ransom is paid.

Alternatively, the ransomware worm can be propagated vertically through the SCADA layers to infect the control network and the corporate servers. In [253] the authors demonstrated a self-spreading worm that can be spread in a SCADA network just by introducing an infected PLC (Siemens SIMATIC S7-1200). It first checks if the target is already infected; if no infection is detected, the worm stops the execution of the installed program, transfer its own code, reboots the PLC and propagates itself to the next target. The worm was designed to survive reboot and power-off procedures, utilizing only the PLC resources, in order to function and spread. These characteristics make it hard to be traced and ideal to be used by an adversary as an attack amplifier. Although this attack cannot be launched from the Internet, it utilizes IoT interconnectivity in order for the worm to spread from one infected PLC

[1] https://www.shodan.io/

to another. Similar ransomware attacks can be accomplished in the opposite direction by first compromising workstations located in the corporate network. In this case, the ransomware attacks belong to the previous category.

3.3.3 Attacks on IoT-enabled field devices

Automated Tank Gauges (ATGs) are small-scale SCADA systems that are used to monitor fuel tank inventory levels and raise alarms (e.g. fuel spill). Most ATGs can be controlled and monitored through a built-in serial interface. Many operators choose to map the serial port to a TCP port that is accessible through the Internet, in order to enable remote control services. According to a technical report published by the security company Rapid7 [121, 136], approximately 5800 ATGs were discovered to be exposed to the Internet through port 10001/TCP, which could be accessed without even requiring a password or utilizing any other authentication mechanism. Through Internet facing TCP port, an adversary can remotely prevent the use of the fuel tank by changing its access settings, simulating false conditions or triggering a manual shutdown. In a similar large-scale security experiment, Trend Micro presented in 2015 a honeypot [157]: fully functional virtualized tank-monitoring systems were created so as to mimic real systems. The virtual ATGs were distributed among eight countries and were visible from search engines such as Shodan. During the experimental period most of the attacks (44%) occurred in the ATGs that where deployed in the USA including a 2-day, 2Gbps DDoS attack, that utilized the *Low-Orbit Ion Cannon* (LOIC) tool [206], against a virtual ATG located in Washington DC.

3.3.4 Attacks on industrial robots

Industrial robots are computerized mechanical multi-axis "arms" used in modern smart factories for automating various operations such as welding, packaging, food processing, etc. Newest models come with advanced programming and networking capabilities that fully integrate them to the factory IT ecosystem. For example, ABB's robots are equipped with a so-called *Robot Web Service* which accept HTTP requests, or support easy-to-use APIs that enable remote control via smarthphones. However, the ever increasing complexity and interconnectivity of industrial control systems and robotics bring a broader attack surface, where different attack types may be combined. Recent studies [180, 223] demonstrated attack scenarios on actual IoT-enabled industrial robots in a controlled environment. Using search engines, like Shodan, ZoomEye and Censys, security researchers managed to discover industrial robots exposed directly to the Internet via File Transfer Protocol (FTP) services or through industrial routers. From a total number of 83.673 robots discovered, 5.105 required no authentication, 59 had embedded known vulnerabilities whereas new vulnerabilities were identified in 6 robots. Their findings included outdated software components (e.g. application-level libraries, compiler, kernel), poor authentication schemes, insecure Web interfaces, obsolete open source code, poor software protection (e.g. unstripped binaries), publicly accessible firmware images, documentation and relative software, WAN access to unfirewalled LAN ports, wireless (GSM or WAN) access to remote service facilities.

The attack scenario presented in Table 7.1 was demonstrated on an ABB's six-axis IRB140 industrial robot. The scenario exploits vulnerabilities of two robot components exposed in the Internet, the main computer and the FlexPendant (a handheld operator unit). Initially static/default FTP credentials were used to access the command driver and permanently disable User Authentication System (UAS). Then, by triggering a reboot, crafted *.NET* Dynamic-Link Libraries (DLLs) were uploaded and executed to the controller, thus enabling them to take control of the robot remotely. The researchers demonstrated five classes of robot-specific attacks that violate the basic operational requirements (accuracy, safety, integrity) of industrial robots: (i) Control-loop parameters alteration, (ii) user-perceived robot state alteration, (iii) actual robot state alteration, (iv) calibration parameters tampering and (v) production logic tampering. Potential impact of these attacks include defective or modified products, robot damages, operator injuries, sensitive data exfiltration (e.g. industrial secrets) and/or ransomware attacks on altered products.

3.4 IoT-enabled attacks on smart power grids

Smart power grids are the modern versions of the energy generation, transmission, distribution and consumption systems. They can be considered as system-of-systems consisting of several SCADA systems and communication networks. The integration of digital monitoring, control and measurement capabilities into the traditional energy systems provide significant benefits to the relative stakeholders such as energy producers, providers and consumers [141]. On the other hand, the distributed intelligence and broadband capabilities of Smart Grids increase the cyber-security risks. A Smart Grid is divided in three main domains: Generation, transmission and distribution of electricity as shown in Figure 3.3. The electricity is generated in power plants and carried along the transmission systems to the distribution systems where electric power is delivered to the end customers, domestic or industrial. These physical systems are interconnected through transmission lines and substations. Energy Man-

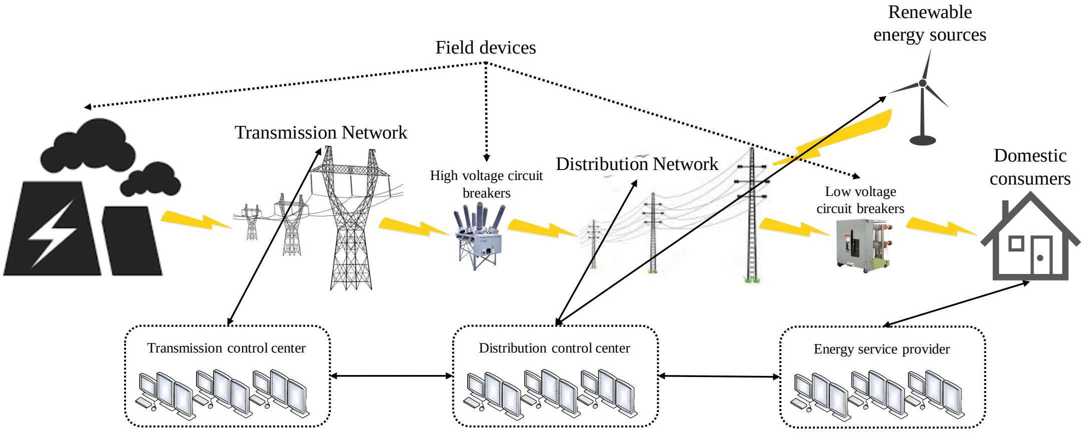


Figure 3.3: A typical Smart Grid architecture [259]

agement Systems (EMS) located at the control centers monitor, control and optimize the grid operations through SCADA systems. On top of these systems, independent system operators coordinate the electricity flow and data exchange among service providers and customers [196].

From the cyber-security viewpoint, the key components of a smart power grid are: (i) The SCADA systems and (ii) the Advanced Metering Infrastructure (AMI) [82, 284]. SCADA systems monitor and control at real-time the power delivery systems based on several communication networks. AMI measures, collects and analyzes the energy usage by the consumers. It mainly consists of smart meters, Data Management Systems (DMS) and several communication networks. Smart meters send measurements towards the DMS through the Home Area Network (HAN). Multiple HANs are connected together to form a Neighbor Area Network (NAN) under each substation, while a WAN is used to connect distributed NANs [118].

Another relative recent Smart Grid feature is the Vehicle-to-Grid (V2G) network [205]. It is based on the concept that the batteries of electric vehicles can be utilized to assist the stabilization of the electricity network [137, 282, 292, 306]. Depending on the power needs, the grid operator may require the batteries of the connected electric vehicles to either return electricity to the grid or throttle their charging rate. However, wireless communication networks between Battery Vehicles (BVs) and the Smart Grid introduce new security challenges [241].

In the following paragraphs we analyze some representative examples of cyber-attacks against Smart Grid components at the generation and transmission domains as well as False Data Injection attacks (FDIA). Moreover, we emphasize on IoT-enabled attacks that usually target customer-side components, such as smart meters, end-user generation systems (solar panels, wind turbines) and electric vehicles connected to the grid. We classify the attacks on Smart Grids according to the target domain: (i) Attacks on generation systems [240, 255, 256], (ii) attacks on transmission systems including interdiction, substation, load redistribution [83, 177, 237, 291, 295], (iii) attacks on distribution/customer side systems - AMI, like energy theft, information and privacy leakages as well as Denial-of-Service (DoS) [183, 251, 297, 298]. We separately examine FDIA, since they affect all domains [169, 269, 290] (see Figure 3.4).

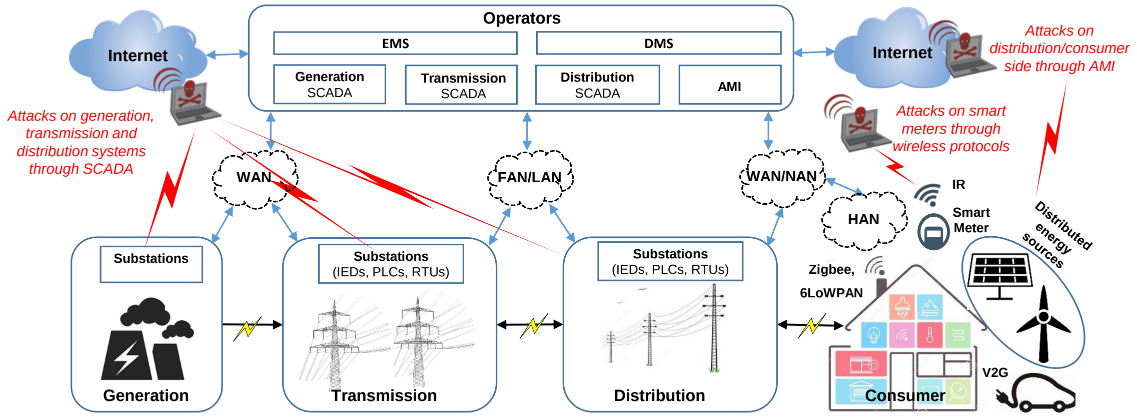

Operators
EMS
DMS
Generation SCADA
Transmission SCADA
Distribution SCADA
AMI
Internet
Internet
Attacks on generation, transmission and distribution systems through SCADA
Attacks on distribution/consumer side through AMI
Attacks on smart meters through wireless protocols
WAN
FAN/LAN
WAN/NAN
HAN
IR
Smart Meter
Distributed energy sources
Zigbee,
6LoWPAN
Substations
Substations (IEDs, PLCs, RTUs)
Substations (IEDs, PLCs, RTUs)
Generation
Transmission
Distribution
Consumer
V2G

Figure 3.4: IoT-enabled attacks on Smart Grid. Wide-area, heterogeneous SCADA systems, AMI infrastructure and home appliances extend the attack surface [261]

3.4.1 Attacks on generation systems

The most famous security testing experiment on electric power generators is the Aurora attack. Demonstrated in 2007 at the Idaho US National Labs [256, 302], the attack forces one or more circuit breakers to open and close in a very fast rate (e.g. every 0.25 sec), resulting in the desynchonization of the power generator and ultimately in its physical damage [302]. The impact of such an attack may range from a short-term power outage to a long-term generation deficiency. The Aurora attack can be performed by compromising the associated PLCs through command injection. An potential attack described in [256] that exploits both cyber and physical system vulnerabilities to control circuit breakers, is an ample example of an Aurora-like attack scenario.

In [110] several real security breaches against power plants in the United States are reported, including a nuclear power plant in Kansas. Despite the suspicions that this incident is connected with the attacks in Ukrainian Smart Grids [14,104,160,228], no digital fingerprints were detected. Although hackers managed to penetrate the corporate networks of operators, no operational impact to the power plants were reported, due to the fact that the industrial computer systems were completely separate from the corporate network. Experts warn that despite that the attacks did not reach any of the critical generation systems, they could be used as preliminary reconnaissance steps in order to collect valuable information.

3.4.2 Attacks on transmission systems

Among the most known attacks in Smart Grids are those regarding the Ukrainian energy industry [14, 104, 160, 228]. In December 2015, a region in Ukraine suffered a massive power outage affecting almost 230,000 customers [160]. Well-known malware, named *BlackEnergy* and *KillDisk*, were sent wrapped up in a word document attached in a phishing email impersonating a message from the Ukrainian parliament. Opening the attachment resulted in executing the malicious payload that planted the BlackEnergy malware. Then the worm spread throughout the power company's networks and managed to retrieve credentials of a Virtual Private Network (VPN) used to access remotely SCADA systems for maintenance. Using the VPN credential enabled them to trip the interconnected circuit breakers in several distribution stations thus causing outages in entire regions. In addition to that, they managed to permanently prevent the legitimate operators from restoring the power by replacing the legitimate firmware of the substation's Serial-to-Ethernet converters, used to connect the older circuit brakers to the network. As their final act, they disabled the battery backup system of the control stations and run *KillDisk* malware to erase information stored on company's compromised workstations.

Next year, a similar, yet much stealthier, cyber attack occurred targeting Kiev transmission station [104]. This time, the central station under attack was of a magnitude of 200 megawatt, thus superseding the total power of all the stations knocked out in the previous-year attack. The adversaries used the same approach and planted the malware *CrashOverride* [74] / *Win32/Industroyer* through spear phishing campaigns. The malware remained stealth until it was triggered by the adversaries. It included a framework that incorporates modules for numerous ICS protocol stacks, such as IEC 101, IEC 104, IEC 61850, and OPC, a wiper to delete files and processes as well as modules to open circuit breakers on RTUs and force them into an infinite loop. A malware analysis by security company Essential Security against Evolving Threats (ESET) [54] revealed that the worm could be programmed to scan the victim's network, discover potential targets and open circuit breakers autonomously, with no intervention of the adversaries. The attacks on the Ukrainian Smart Grid are also further analyzed in Section 4.3.2.

3.4.3 False Data Injection Attacks

State Estimation (SE) plays an important role in Smart Grid operation. It calculates the current state of every circuit and transfers raw measurements from Smart Grid components to the operation control center. In order to affect the SE process an adversary may inject falsified state estimation data so as to disrupt the operation and control of EMS. Recent studies [169] examine the potential impact of FDIA in three main categories: (i) *Electricity market*: They are mainly focused on the economic aspect of FDIA [269]. An adversary can potentially gain a substantial profit by acquiring virtual electric power at a lower node price and sell it at a higher node price; (ii) *System operation*: Their goal is to manipulate the quantity of energy supply and response as well as the link state information. Energy deceiving attacks may deregulate the balance between power supply and demand thus leading to a disruption of the electricity and significant cost increase; and (iii) *Distributed energy routing*: For example, load redistribution attacks [290] target the *security-constrained economic dispatch*, used for minimizing the overall cost. Injects falsified data may drive the system in a unoptimized operating state and may potentially destabilize a large segment of the distribution network.

3.4.4 Attacks on renewable energy and distribution/customer side systems (AMI)

Real as well as PoC attacks depict the threat landscape on AMI (e.g. in the smart meters [172, 183, 187, 251]). Security researchers have presented potential impact scenarios originated from connecting vulnerable smart meters to a home network and analyzed the insecurity features of hardware, embedded software and networks of the AMI. In 2010, an FBI's report analyzed the Puerto Rico's case [154] where a fraud against an electric utility was disclosed. Adversaries (former company's employees) were tampering smart meters and modifying measurement and billing data, using an infrared communication port. As reported, the estimated financial loss could reach up to $400 million. In 2016, a security researcher presented a command injection vulnerability (ICSA-16-231-01) that allows hackers to remotely control vulnerable smart solar meters (Locus Energy) [254] and spoof power level reports or perform DDoS. With almost 100K devices in the wild the company released an updated firmware version to address the issue.

Renewable energy systems, such as wind turbines and solar panels interact directly with the distribution power network and, in most cases, are connected directly to the Internet. In 2016, a security researcher pentested his own solar panel management unit (Tigo Energy MMU) [37] to discover an open access point for remote control as well as a permanent connection through VPN tunnel from his device to the vendor's premises. Using *Wigle.net* engine, he was able to detect almost 10,000 similar systems exposed to the Internet, of which, 160 constantly connected. Their Web interfaces were vulnerable to remote code execution, utilized unencrypted HTTP interfaces and used easy to guess/default credentials (e.g. admin/support). In 2015, another security researcher identified numerous flaws in clean energy systems [274] such as the XZERES 442SR Wind Turbine, the Sinapsi eSolar Light and the RLE Nova-Wind Turbine. These vulnerabilities have been reported to ICS-CERT (ICSA-15-160-02, ICSA-15-342-01B/C, ICSA-15-162-01/A) and include, among others, passwords stored in plaintext files and/or the use of Cross-Site Request Forgery (CSRF) vulnerability to change the Web interface administrator password. For all three devices examined, the researchers could perform various control actions, such as alter wind vane correction or change the network settings to make a Web interface inaccessible. The attack scenario from [274] presented in Table 7.1 is related to the RLE Nova-Wind Turbine HMI vulnerability (ICSA-15-162-01A).

Vulnerable V2G communications are considered to be another way to attack the power distribution network as previously stated. Although hacking smart cars has been proven to be feasible, to our knowledge no attack to Smart Grids through V2G network has been reported in the past. However, recent works [199], [308] indicate security concerns and challenges related with V2G power and communications interactions. In [199], the authors have proposed a model that jointly optimizes security risks and equipment availability in the interdependent power and electric vehicle infrastructure. In [308], a context-aware authentication solution for V2G communications in the Smart Grid has been presented and several open security issues of V2G networks have been discussed.

3.5 IoT-enabled attacks on Intelligent Transportation Systems

Intelligent Transportation Systems (ITS) [69] involve smart cars and road infrastructures, railway control systems, air traffic control systems, and smart maritime surface vessels (see Figure 3.5). Cyber attacks in ITS may lead to severe consequences not only on the transportation operations, but also on other sectors or even on the safety of citizens. A recent study published by ENISA [164] reports that there is currently no EU policy on cyber security for intelligent public transport, the awareness level is low and it is difficult for operators to dedicate budget to this specific objective of cyber security.

3.5.1 ITS architecture and related IoT technologies

We briefly describe the main IoT technologies that are utilized in the ITS ecosystem, as depicted in Figure 3.5.

Smart cars and road infrastructures

Modern cars can be considered as "computers on wheels". Dozens of tiny computers, aka *Electronic Control Units* (ECUs) in automotive terminology, are used to manage traditional mechanical and electrical subsystems, such as breaking, transmission, locking and airbags, as well as modern systems like the infotainment, emergency call or cooperative cruise control [162]. Initially, a dedicated point-to-point connection was used to connect all subsystems to ECUs. In order to reduce car wiring costs, in the mid 80's the dedicated connections were replaced by the Controlled Area Network (CAN) bus. An *On Board Diagnostics Socket* (ODB) was also introduced to provide physical access to the the whole system. Being a 30-year old standard, the CAN bus does not include any security mechanisms making it vulnerable various attack types, such as passive sniffing and command injections [176]. Cheap, off-the-shelf software, such as CANdo [211] by Netronics allows a novice user to control a car via a graphical user interface, sniff, inject or decode CAN bus messages. Despite the advances in car bus technologies [153], a large fraction of the car fleet worldwide relies on CAN.

Smart cars integrate various IoT technologies. Internet connectivity is implemented via cellular data Subscriber Identity Module (SIM) cards, while in-car WiFi is also supported. Internet connectivity enables various services, such as on-line infotainment services, remotely updating the car's software, emergency "e-call" services, and navigation services with real-time traffic data [127]. Various smart control and assisting systems, such as Autonomous Driving Systems (ADS), Adaptive Cruise Control (ACC), collision avoidance, automatic speed enforcement and emergency vehicle notification systems, are based the data collected by on-board sensors [186]. These sensors may use diverse wireless technologies to communicate with each other and with ECUs through the CAN bus, to send the data to other cars (*Vehicle-to-Vehicle* – V2V communications) or to communicate with traffic infrastructures installed in roads (*Vehicle-to-Infrastructure* – V2I communications) [4]. An typical example of V2I/I2V service are smart traffic signals that provide adaptive traffic management and variable speed limit enforcement. Another example involves sensors installed inside the roadways in order to create in-ground induction loops with the metal bodies of the cars, for example, to detect vehicles at intersections. These sensors may also communicate with other infrastructures (I2I communications) such as traffic signals.

Smart railway systems

Modern train control and railway signaling systems have become fully autonomous. With the assistance of the Communication-Based Train Control (CBTC) system [215], a train can determine its position and speed, based on data received from onboard sensors (e.g. tachometer) as well as from the Absolute Position Reference (APR) beacons located on the track. These data are then send to a sideways system through a radio-based communication link, which in turn forwards the data to the central Automatic Train Supervision (ATS) system at the operations control center. Zone controllers that process these data are used to determine the train's Limit of Movement Authority (LMA) – the total distance until the next obstacle. Each train is under the control of a zone controller whereas Automatic Train Protection (ATP) and Automatic Train Operation (ATO) systems [72] associate the LMA information with local train data, to issue appropriate train control commands to the train, typically through some Driver Machine Interface (DMI). Finally Public Information Display (PID) systems are used to inform the commuters in real-time for delays and other incidents and to advise them for alternative means of transportation, through on-site screens, websites and mobile applications.

Aircrafts and civilian air traffic systems

Several air traffic control and support systems are used nowadays to increase the connectivity and "openness" of modern aircrafts. Some of these systems which heavily rely on wireless technologies, thus increasing their exposure to new security threats, are briefly described in the following. Automatic Dependent Surveillance Broadcast (ADS-B) [266] system enables an aircraft to determine, through satellite navigation, and broadcast its position for tracking purposes. ADS-B is expected to replace radar systems as a primary means of tracking. Other wireless supporting systems include the Aircraft Communications Addressing and Reporting System (ACARS) [236] and the Traffic Collision Avoidance System (TCAS) [178].

Another category of aircrafts systems that have been proved in practice to induce serious security risks are the In-Flight Entertainment (IFE) systems. IFE have evolved to sophisticated seat-back computers that provide Internet connectivity to passengers' smartphones or tablets and other services such as stream content, interactive maps and surround-sound audio.

Maritime surface vessel and port control systems

Maritime control and navigation systems include the Automatic Identification System (AIS), the Vessel Traffic Service (VTS), and the Electronic Chart Display Information System (ECDIS) [71, 283]. The interconnection of all these control systems creates a port-specific SCADA system. AIS is an automatic tracking system mainly used for collision avoidance. It transmits safety related information like course, speed, type of vessel, type of cargo, at-anchor or underway status. VTS is a marine traffic monitoring system, similar those used in airports, established by port authorities. ECDIS is a navigational chart display that receives data by other control systems, (AIS, GPS, and radars), to allow an officer on deck to navigate the ship. At the port side, the Port Management System (PMS) has a central role; it receives information from the Terminal Operating System (TOS), essential for supply chain management. TOS monitors the location of containers and handling equipment (cranes) through Optical Character Recognition (OCR), Radio Frequency Identification Devices (RFIDs) and GPS systems.

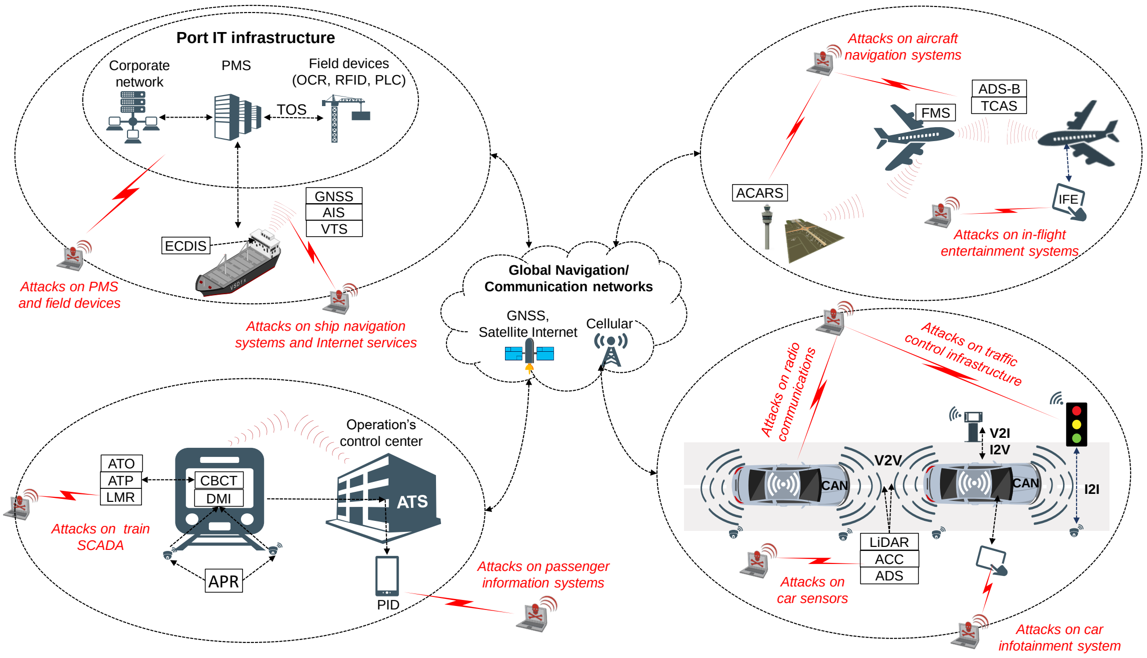

Port IT infrastructure
Corporate network
PMS
Field devices (OCR, RFID, PLC)
TOS
GNSS
AIS
VTS
ECDIS
Attacks on PMS and field devices
Attacks on ship navigation systems and Internet services
Attacks on aircraft navigation systems
ADS-B
TCAS
FMS
ACARS
IFE
Attacks on in-flight entertainment systems
Global Navigation/ Communication networks
GNSS, Satellite Internet
Cellular
ATO
ATP
LMR
CBCT
DMI
APR
Operation's control center
ATS
PID
Attacks on train SCADA
Attacks on passenger information systems
Attacks on radio communications
Attacks on traffic control infrastructure
V2V
V2I
I2V
I2I
CAN
CAN
LiDAR
ACC
ADS
Attacks on car sensors
Attacks on car infotainment system

Figure 3.5: Intelligent transportation systems' architecture and corresponding IoT-enabled attacks [261]

3.5.2 IoT-enabled attacks on ITS systems

In this subsection, we analyze IoT-enabled attacks for all the transportation subsectors. Depending on the adversary's placement inside the relevant infrastructure perimeter, then the attack requires physical proximity with the target (e.g. attacks on car sensors [217, 294]). If the adversary is outside the perimeter then the attacks are executed from the Internet (e.g. attacks on car infotainment systems [193, 279]). Finally, if the adversary of an attack is placed at the border, then both nearby and remote scenarios are possible (e.g. attacks on cars' radio communications [188, 203]).

Attacks on smart cars and traffic control infrastructures

Preliminary works [149] demonstrated a plethora of attacks against the CAN bus. By injecting crafted messages to the bus, it is possible to control the display of the speedometer, kill the engine or the car brakes. Miller and Valasek [191, 192] provided detailed analysis of the CAN bus vulnerabilities. However, these attacks required physical tampering of the target vehicle and thus cannot be considered as typical examples of attacks that exploit some IoT technology (e.g. sensors or other interconnected devices).

As described bellow, IoT-enabled attacks against smart cars can be categorized to: (i) Attacks that exploit radio communication protocols used in smart car communications (such as LAN, Digital Audio Broadcast (DAB) and WiFi); (ii) attacks that exploit vulnerabilities of car infotainment systems; and (iii) attacks based on manipulating sensor IoT technologies.

Attacks based on radio communications In [188] a remote attack based on low-cost radio equipment is described. The attack requires physical proximity to the car. Using a $15 radio transmitter, a nearby attacker can exploit CAN network vulnerabilities and software vulnerabilities, to connect and send commands to the CAN bus. In [279] a similar attack shows that it is possible to extend the distance of the attacker from the target vehicle, by setting up a bogus radio station through which the attacker sends crafted DAB messages in order to compromise the infotainment system of the car. Since the infotainment system is directly connected to the CAN bus, the attacker can remotely control a car, provided that the car's infotainment system is tuned to the bogus station. A similar attack that is based on manipulating the Bluetooth or the telematics unit can be found in [50]. In [203] another PoC attack is demonstrated by professional penetration testers, that is based on vulnerabilities of WiFi connectivity. They discovered that the mobile application used to remotely control several car operations in a specific car model, was using the car's WiFi access point, instead of a GSM module. Then, by cracking the (weak) WiFi password and replaying messages from the mobile application, they succeeded to inject modified commands and control various car systems. In general, the attacks of this category either require that the attacker has some physical proximity to the target (in the cases of LAN and Bluetooth protocols) or that the target car has some specific configuration (in the cases of DAB and WiFi protocols).

Attacks based on car infotainment systems Vulnerabilities in the infotainment system have also been exploited in Internet-connected cars (the attack of [279] already described above, also belongs to this category). In [193] Miller and Valasek demonstrated how it is possible to remotely hack a car (jeep Cherokee) by abusing its infotainment system. Initially the researchers discovered an open port in cellular network used by Harman Uconnect infotainment system designed to offer WiFi connectivity, navigation, and several applications. Using the open port they remotely scanned the software and discovered and exploited vulnerabilities in the OMAP chip of the head unit. Then, using the Secure Shell (SSH) service they enabled remote Command Line Interface (CLI) and compromised the U-connect infotainment system. Since the infotainment was directly connected to the CAN, they were able to flash a modified CAN firmware to remotely control the car. Scanning the network revealed 2,695 connected vulnerable vehicles with their initial projected estimations to put the total number to be somewhere between 292,000 and 471,000. After the hack received publicity [109] the car manufacturer was forced to recall 1.4 million vehicles [226] in order to patch the vulnerability. Infotainment system vulnerabilities, especially when combined with network layer vulnerabilities can cause significant damage, since a remote attacker can launch multiple attacks concurrently against vulnerable vehicles thus having a huge potential effect on transportation infrastructure.

Attacks based on car sensors Autonomous driving systems rely on sensor readings in order to continuously provide data to systems like the ACC, collision avoidance or lane keeping assist system. All these systems require extended wireless connectivity, leading to an increased exposure to remote attacks or system failures. The first known death caused by a self-driving car was disclosed by Tesla Motors [293]; due to a system failure the car's sensors failed to distinguish a large white 18-wheel truck and trailer crossing the highway.

Verified attacks in this category include [217,294]. In [217] a low-cost laser is used to "blind" the camera of the target car. Then by exploiting the lack of authentication in *Light Detection And Ranging* (LiDAR) messages, older messages are replayed to produce false artifacts and confuse the system. A similar PoC is presented in [294].

These attacks demonstrate that the wireless intelligent support systems of modern cars need further security assessment. Other attacks, such as relay station and amplification attacks, demonstrate weaknesses in the Remote Keyless Entry (RKE) systems [94]. Although the above attacks require physical proximity to directly attack sensors' communications at the data-link layer, we must bear in mind that the control is gradually being taken away from the driver and placed under the supervision of embedded autonomous control systems in order to automate the driving process. Therefore, protecting car sensors from Internet adversaries should also be considered in the near-future threat landscape.

Attacks on traffic control infrastructures PoC attacks against IoT-enabled traffic control infrastructures have been recently demonstrated [45, 100]. These attacks are mainly due to vulnerabilities in the radio communications of traffic control systems. In [45], the feasibility of various attacks against real on-road wireless sensors and repeaters was proved for first time. These attacks are due to vulnerabilities in the link-layer radio communications. By creating a portable access point with off-the-shelf hardware and by eavesdropping the messages and then injecting unauthenticated commands to the ITS network, the researcher was able to adjust traffic control systems that could be used to cause traffic jams, and accidents and block emergency services. The most warring evidence is that the attack can be amplified by using a *self-spreading* firmware update, in order to compromise a large number of sensors and repeaters that are installed in many countries world-wide. Another study [100] showed that with the appropriate radio equipment, an adversary could take control of the traffic infrastructure thus enabling DoS attacks, cripple the traffic flow in a city, or cause congestion at intersections by modifying light timings.

Attacks on railway control systems

Real incidents against train control systems, such as [13,55,165,303], come as warning for the worst case scenarios to become true if proper actions are not taken. Verified IoT-enabled attacks against railway systems include: (i) Direct attacks on connected railway SCADA systems and (ii) subliminal attacks that are based on manipulating non-critical passenger information systems.

Attacks against IoT-enabled railway SCADA systems In [107] (see also [152]) a research team named *SCADA Strangelove*, presented a 3-year assessment on actual SCADA train control systems full of high-level security and safety issues. They discovered train switches that needed constantly Internet access to operate and computer-based interlocking systems, which were installed in places with poor physical security and were using outdated and discontinued operating systems (like Windows XP/2000). Furthermore, various network-layer vulnerabilities were found including weak authentication schemes, lack of encryption, integrity and authorization controls as well as design and embedded vulnerabilities, such as internal architecture design issues, port access rules, password policies and more. Through Shodan search engine, they managed to discover publicly accessible network equipment in mission critical systems with default passwords.

A security analysis in communication channels such as GSM-R SIM cards used in Germany, revealed that an adversary with low-cost, off-the-shelf equipment could jam the GSM communications of a moving train thus forcing it to a complete halt. Modems, used to connect train systems and services to the Internet through cellular network, were found to be susceptible to attacks such as the ones described in [299]. By initiating a firmware Over-The-Air (OTA) update an adversary could compromise the modem as well as the connected host machine, thus enabling the remote control of mission critical systems of the train.

Attacks based on passenger information systems A recent security analysis on urban railway systems [51] showed that even attacks against non-critical systems may have severe consequences, due to subliminal (hidden) cyber-physical attack paths. For example, compromised PID systems of railway stations may be used to amplify the impact of a physical attack. Since PID systems send real-time data to mobile users, an adversary that has compromised the PID system may inject fake arrival times to overcrowd train platforms. Then, in a worst-case scenario, terrorists could launch a bombing attack on the targeted platforms with severe consequences. Such combined cyber-physical attacks, that abuse IoT systems, may prove to be critical despite the fact that the exploited IoT system/service (e.g. PID) is not connected to a mission critical system. Although in a particular attack scenario physical access to the PID system is required, an adversary could potentially triggered the attack from a remote location by exploiting direct/indirect attack paths to the PID server. Attacks that belong to this category point out the difficulty in identifying high risk, subliminal attack paths when IoT-enabling technologies are used alongside with traditional cyber-physical systems and services.

Entertainment/infotainment systems, IP surveillance cameras and wireless access points may also induce serious risk when they operate without proper network segmentation. A security analysis [107] concerning devices used in railway communication systems from various vendors revealed hardcoded private Secure Sockets Layer (SSL) keys embedded in their firmware. Other attack scenarios described in [51] with potentially severe consequences, include manipulation of data from installed sensors in the train odometry system, gaining access to the signaling network and jamming or manipulating commands through fake wireless transmitters.

IoT-enabled attacks on aircrafts

Airplanes and air traffic control systems are complex, sophisticated and highly interconnected systems that are subject to various security threats. Recent cyber attacks that have been reported, include shutting down passport control systems [166] and causing DoS to systems used to issue flight plans [268]. Although the aforementioned attacks cannot be classified as IoT-enabled, recent incidents have demonstrated the risk of integrating IoT technologies in aircrafts and air navigation systems. IoT components like air navigation and ground control systems are indirectly connected with auxiliary systems, that may enable hackers to gain unauthorized remote access to critical components. Examples of IoT-enabled attacks in this sector include: (i) Attacks based on vulnerabilities of wireless air traffic surveillance systems and (ii) attacks that exploit vulnerabilities of IFE systems.

Attacks based on aircraft electronic navigation systems In [61,144,272] PoC attacks against the ADS-B system of airplanes are presented. A series of such attacks, that inject bogus messages in the ADS-B network by first eavesdropping unencrypted and unauthenticated communications, were presented in [61]. In a similar work [272] it was claimed that it is possible to take control of the Honeywell NZ-2000 Flight Management System (FMS), through an Android application called *PlaneSploit*. This PoC attack utilized simulation software and parts that are used to control an airplane available on eBay. Using the Android application and ADS-B and ACARS systems the researcher was able to inject bogus messages to FMS system and take full control of the airplane. The Federal Aviation Administration (FAA), however, stated that this attack could not be actually realized, since, the hardware used in the demo attack were not identical to the ones used in real airplanes. A later work [144] analyzed an aircraft's control systems and suggested that ACARS and ADS-B systems are vulnerable to attacks that could potentially affect the autopilot operation, but could not allow a remote attacker to effectively take over the critical navigation systems.

Attacks based on vulnerable in-flight entertainment systems. In [239] a security expert demonstrated a series of attacks that exploit vulnerabilities of the IFE system in order to hijack several mission critical plane subsystems. This demo attack revealed vulnerabilities of the widely used Panasonic Avionics IFE system and was based in real data collected by the researcher while he was in flight. Using an exposed Universal Serial Bus (USB) port the researcher managed to retrieve debug information which then used to discover on-line publicly available firmware updates for multiple airline companies. After some information gathering and reverse engineering, the researcher could finally connect with a USB keyboard to the IFE system and commence attacks. He managed to bypass credit card check, have arbitrary file access as well as perform SQL injections and gain access to credit card details and personal information. Other feasible attack scenarios included flight information spoofing (altitude or speed), introduction of bogus route messages on the interactive map, or tampering the *CrewApp* unit that controls the public address system, lighting and actuators. In a worst-case scenario in which the vulnerable IFE system is indirectly connected to airplane's mission critical control systems, a terrorist could hijack the aircraft from a passenger's seat with devastating consequences.

Attacks on maritime surface vessels

Published incidents against maritime cyber systems that are not IoT-specific can be found in [230, 275]. Again, we will categorize IoT-enabled attacks in this sub-sector.

Attacks on maritime electronic navigation systems and Internet services In [21] attacks against the AIS of existing vessels were presented. In particular by using Man-in-the-Middle (MiTM) attacks, an adversary could hijack and take over AIS communications, tamper with the major online tracking providers and eventually spoof the position of the vessel. Global Navigation Satellite System (GNSS) signals are used even for vessels actively piloted by human operators. But as surface crafts become more autonomous, autopilot systems and dynamic positioning systems are designed under the assumption that GNSS signals are usually available and trustworthy. In a PoC attack presented in [70] researchers from University of Texas managed to deviate a maritime surface vessel from its original course, by broadcasting counterfeit civil GPS signals. In order to remain covert, the spoofed signals were slightly altered.

By using search engines like Shodan, a security company named *PenTestPartners* [202] discovered vulnerable Web interfaces of ship's mission critical systems (e.g. electronic navigation systems). Most of them used weak default passwords, allowed unencrypted HTTP connection without enforcing standard SSL/ Transport Layer Security (TLS) and/or were vulnerable to known Web attacks like SQL injection. Various attack scenarios include remotely exploitation of several IT systems of the ship in order to reveal sensitive information about the ship or the crew and even take control over the ship.

Other vulnerabilities found, include a vulnerable on-board mail client (named AmosConnect by Immarsat Solutions) [22], that could allow unauthenticated attackers to perform blind SQL injection and recover usernames and passwords. Then, with the use of the retrieved credentials, an adversary can remotely execute arbitrary commands with system privileges on the remote system by abusing the Task Manager of the mail client.

Attacks on IoT-enabled PMS and field devices The number of containers shipped world wide have increased over 200% from 1996 to 2014. In a recent study [28] security researchers present an exhaustive analysis of threats and attacks scenarios that include the entire supply chain management such as attacks on Internet-connected port's systems, field devices (TOS, OCR, RFIDs), PLCs and motors that are found mainly installed in yard cranes (ICSA-16-348-05B). In a real attack incident [26], an international drug dealer group used hacking techniques that involved the exploitation of the IT systems and services that controlled the movement and location of containers, in order to illicitly transfer drugs through the port of Antwerp over a two year period.

3.6 IoT-enabled attacks on e-healthcare

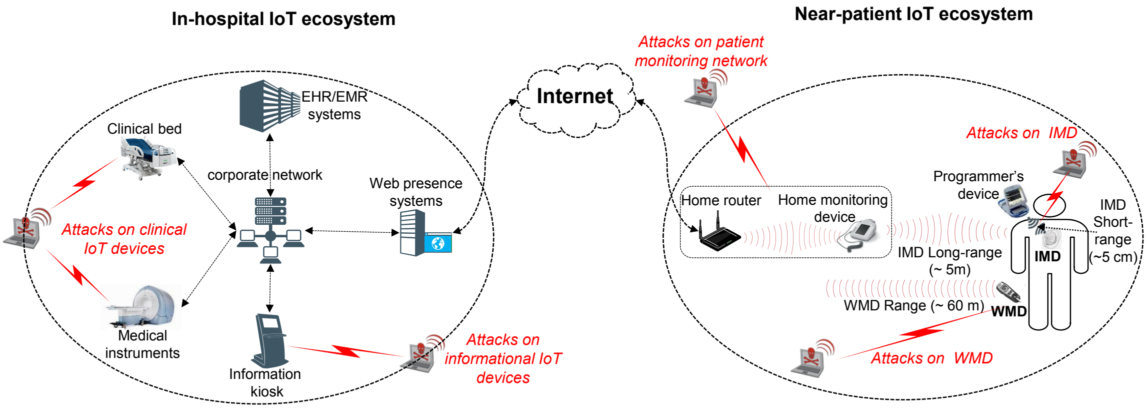


Figure 3.6: A high-level architecture of *near-patient* and *in-hospital* IoT ecosystem and relevant IoT-enabled attacks [261]

Near-patient and *in-hospital* IoT technologies have been used in e-health services to provide timely monitoring of clinical events, reduce routine patient follow-up and transportation costs and increase patient's quality of life. In this segment, we provide a brief description of the medical IoT technologies and then we review IoT-enabled attacks in the medical sector. Figure 3.6 describes a general architecture of the medical IoT ecosystem as well as a high-level description of the relevant IoT-enabled attacks.

3.6.1 Architecture of medical IoT systems

Medical IoT devices can be categorized to active and passive. Active Medical Devices (AMDs) are used to interact directly with a patient in order to dynamically adjust a medical treatment. Examples of AMDs are Implantable Medical Devices (IMDs) (e.g. heart defibrillators) and Wearable Medical Devices (WMDs) - (e.g. insulin pumps) [133]. These can be considered as near-patient technologies, although they can also be used during in-hospital treatment. Other AMD technologies, such as radiation oncology systems, may only reside inside hospitals. Passive Medical Devices (PMDs) monitor, gather and report data related with the patient's physical condition to medical IT systems. Such devices may reside inside in both hospital (e.g. a smart clinical bed) as well as near patient (e.g. a home monitoring device).

Near-patient medical IoT

IMDs and WMDs can be considered as the most common near-patient active IoT technologies. Programmable IMDs consist of a battery-powered embedded device that is surgically implanted under a patient's skin. Via radio communications, IMDs provide continuous and real-time diagnosis and treatment for patients outside the hospital, such as monitoring long-term diseases and remotely applying prescribed therapies. Instances of wireless re-programmable IMDs are smart pacemakers, neurostimulators, and implantable drug pumps [66]. Likewise, latest versions of Implantable Cardioverter Defibrillators (ICDs) support wireless communications for both device reprogramming, through an external device operated by a physician, as well as remote patient monitoring [243]. A home monitoring device may be used to collect patient data through wireless interfaces and transmit them via the Internet to healthcare specialists. A similar but less complicated architecture is used for WMDs, such as mobile insulin pumps that use a Continuous Glucose Monitor (CGM) device to monitor and adjust the sugar level in the, blood of diabetic patients [33]. A wireless interface that utilizes proprietary network protocols (e.g. 916.50 MHz with on-off-keying modulation) is used to configure device settings [224].

IMDs communicate by utilizing two wireless communication channels. Short-range channels (up to 5cm) are used to configure the IMD through the physician's programming device, while "long"-range ones (up to 5m) are used to communicate with a home monitoring device [182]. The WMDs utilize a single wireless communication channel having a broader range, up to 60m, based on the findings of recent attacks [224]. The locally collected data are transmitted via the patient's home network, to back-end hospital IT systems in order to be stored and processed by medical personnel.

In-hospital IoT devices

At the hospital premises, *Electronic Medical/Health Record* (EMR/EHR) systems are critical IT systems that store and process health data collected through various sources. Although EMR/EHR systems are not considered typical IoT systems, they communicate and interact with various IoT-enabled systems. A typical example is the external patients' monitoring networks that provide real-time medical data to healthcare providers in order for them to be able to react promptly to emergencies.

In addition, EMR/EHR systems also communicate with various in-hospital IoT-enabled AMDs. Modern medical instruments that used to be isolated, are now equipped with communication capabilities. For example, oncology radiation or flouroscopy systems are considered to be AMDs that are now able to exchange sensitive data with EMR/EHR systems. These devices are under strict technical specifications and manufacturer restrictions that prevent the hospital's IT security stuff to examine the device for vulnerabilities or install antivirus software. Furthermore, in most cases, such devices come with rich networking capabilities while running on outdated and/or unpatched software which results in a increase on their exposure to security threats. In many real incidents, the use of outdated operating systems in medical devices or in Internet-connected in-hospital IT systems, act as an enabler for the cyber criminals (e.g. to introduce ransomware [92] or steal EMR/EHR data).

In-hospital interconnected smart PMDs, such as patient monitoring systems (e.g. smart clinical beds), can also be used as entry point in order to pivot to critical EMR/EHR systems since they suffer from the same vulnerabilities such as AMDs. Additionally, informational in-hospital kiosks also introduce risks; although they do not fall into the PMDs/AMDs categories, in most cases they are connected to the hospital's internal networks thus creating hard-to-detect, subliminal attack paths towards hospital's critical IT systems.

3.6.2 IoT-enabled attacks on medical systems

Attacks on IoT-enabled medical equipment, IT systems and services may include, among others: Treatment denial or modification, device functionality misuse or abuse (e.g. to deliberately increase the radiation level of an X-ray device or to induce an electric shock to a patient's heart through a heart defibrillator), patient's EMR extraction/modification, medicine loss/destruction, medicine/organ/blood inventory list alternation, surgery schedule alternation, report of false information/medical events, medical event/information concealment, DoS attacks (e.g. battery exhaustion) (x) patient's physical sample(s) loss/destruction, climate controlled transport/storage environment alternation and many more. Bellow, we describe IoT-enabled attacks on medical devices or systems, while in Table 7.5 we present an assessment of these attacks based on realistic scenarios.

Attacks on near-patient medical IoT devices

These attacks are based on vulnerabilities of: (i) The IMD/WMD devices or (ii) the patient's home monitoring network [31]. The impact of such attacks may be high, since motivated cyber criminals may physically harm patients from a short distance or steal health data. Recently, the ICS-CERT issued an advisory (ICSMA-17-241-01) for Abbott Laboratories' pacemakers which affects, only in the US, approximately 65,000 patients. According to the advisory, patients must visit their doctors in order to update the embedded firmware due to security reasons [278].

Attacks based on IMD/WMD devices The security of wearable and implantable medical devices has been studied in various works in the past [101, 142, 143, 167, 171, 305]. Here, we examine some characteristic examples that demonstrate IoT-enabled attack scenarios that usually exploit the short and/or the long-range proprietary IoT communication protocols of the devices in order to inject commands, leak data, brick the devices or introduce spoofed network messages [120, 182, 224].

Halperin *et al.* [120] presented a security analysis of such devices on communication protocols, physical tampering and reverse-engineering techniques of the radio frequency modulation schemes used in short-range proprietary protocols. Due to the lack of cryptographic protection and tamper resistance mechanisms they were able to extract, modify and reinstall a modified firmware image in order to take control of the device from a short distance. Universal Software Radio Peripheral[2] (USRP) and open-source radio libraries were used in order to eavesdrop and examine the (unencrypted) low-range communications between the ICD and the programming device. Finally, an attack scenario was demonstrated in which a nearby attacker could intercept patient data and inject bogus messages to modify the existing therapy.

A similar security assessment was presented by Marin *et al.* [182] in a black-box analysis on an ICD. They demonstrated that attacks, which had been presented by security researchers in the past [120], were still possible. By reverse engineering the proprietary network protocols, they managed to perform passive or active eavesdropping, spoofing and replaying attacks as well as to exploit the functionality of

[2] https://www.ettus.com/

the short-range via the long-range communication protocol. This enabled them to extend the radius of the attack from a few centimeters up to 5 meters. Using inexpensive equipment, the researchers were able to drain the ICD's battery (DoS), recover sensitive patient data (e.g. patient's name or medical history), track, locate or identify patients via ICD's serial number and even send arbitrary commands (spoofing attacks) to the device.

In [224] Radcliffe presented PoC attacks on WMDs, such as insulin pumps. The author demonstrated that through signal jamming, an attacker could launch replay attacks and send falsified readings of glucose levels to the device, or use a wireless peripheral device to change the configuration settings of a insulin pump with potential deadly effects on the patient. As described in the previous scenario, an attack could be launched using cheap and easy to find equipment from a distance up to 60 meters.

Attacks based on patient monitoring networks Rios and Butts, [31] from WhiteScope security company, performed an exhaustive security evaluation of patient home network devices, such as physician programming and home monitoring devices of four major ICD vendors. The security evaluation revealed a large number of potential security risks stemming from underlying protocols of the subsystem communications, hardware and embedded software. In particular, the commercial microprocessors used in most devices were found to be susceptible to reverse engineering due to their open chip architecture and instruction coding. Most devices were found to have at least one easily accessible embedded debug port (Joint Test Action Group (JTAG), Universal Asynchronous Receiver/Transmitter (UART), USB or serial), from which, extraction of the firmware and privileged access to the device was possible. In addition, there was no established anti-reverse engineering techniques, such as firmware packing, code obfuscation and data encryption as well as no authentication or control for digitally signed firmware mechanisms during the OTA update. Furthermore, several bad practices were identified, which in turn can potentially help an attacker to compromise the device: The usage of American Standard Code for Information Interchange (ASCII) text for function names and release versions, hardcoded, clear-text credentials and infrastructure data (e.g. phone numbers and IP addresses of the authentication servers) on home monitoring devices, unencrypted sensitive patient data (patient names, physicians, phone numbers, social security numbers and treatment data) on the programmer's hard drive as well as an extended use of third-party outdated SW libraries. Notably, over 3,700 well-known vulnerabilities were discovered in the embedded software of the physician programming devices under evaluation. Although the study of [31] does not describe any actual PoC attack, it lists numerous, high-severity vulnerabilities on real devices that if exploited, can lead to a full compromisation of a patient device remotely.

Attacks on in-hospital IoT devices

Real cyber attacks against hospitals, such as [92, 96], have increased by 63% during 2016 [140]. Here, we focus on those attacks that rely on IoT-enabling technologies within hospital facilities. These attacks exploit vulnerabilities of either in-hospital medical IoT devices (both passive and active) [131, 277], or other non-medical IoT devices that may reside within hospital premises [131]. Usually, the adversary uses such vulnerable IoT devices as a point-of-entry, in order to pivot and attack other critical EMR/EHR systems that have some indirect connection with the vulnerable IoT devices. In particular, successful attacks against in-hospital IoT devices may be used as "building blocks" of a broader attack. Exploiting in-hospital IoT devices, an adversary may deny critical medical services by launching ransomware campaigns or exfiltrate sensitive medical data with severe consequences.

Attacks based on clinical IoT devices A technical report released by TrapX Research Labs [277] based on in-depth security assessments, revealed real attacks that took place in three hospitals. The assessors installed within the hospitals' facilities a custom-made software called *DeceptionGrid* that emulates medical devices (Virtual Medical Devices – VMDs) in order to attract, trap, and engage attacker software tools. Then, a custom security platform was used to monitor malicious activities in the hospitals' network and reveal potentially hidden attacks. In a relatively short time period after the deployment of the VMDs, they documented various attacks that occurred.

In the first hospital one Virtual Medical Device (VMD) was attacked by a variant of an old worm (MS08-067), which had been repackaged and embedded in a sophisticated way to avoid being detected by any anti-virus software. Since it is common that actual medical devices run outdated operating systems, such as Windows XP and 7, the assessors concluded that the attack had also affected other real in-hospital medical IoT devices. The researchers were able to track the malware back to its source to discover that it had originated from a compromised radiation oncology system running Windows XP. Four VMDs in separate networks also raised alerts. Tracking back the malware indicated a compromised fluoroscopy workstation.

In the second hospital, the introduced VMDs were installed on all internal networks and servers within a Picture Archive and Communication System (PACS) [126] used to exchange medical data between devices, such as X-ray, Computed Tomography (CT-scan) and Magnetic Resonance Imaging (MRI). After one day the

VMDs captured malicious activity that originated from a compromised medical device (MRI), resided in a different network segment. The back-door used by the malware included a sophisticated worm, able to move between different segments of the network and communicate to a C&C server of an external botnet. After analyzing the malware it became clear that the attackers' main target were upatched Windows 7 and outdated Windows XP OS that allowed them to upload a Remote Access Trojan (RAT) in order to download sophisticated malicious software. The compromised MRI was installed within urgent care and the remediation process took several weeks since the infected device had to be replaced by a new one.

Finally, in the third hospital, an attack which originated from an X-ray device running again an outdated operating system (Windows NT), occurred within 20 minutes after the deception grid was installed. The malware, a computer worm, escaped from the detection of the hospital's IT stuff. As in previous cases, the attackers used wrappers with sophisticated package techniques, able to bypass up-to-date antivirus software, whereas the actual payload targeted vulnerabilities that exist only in upatched/discontinued versions of operating systems. In all cases, the IT stuff of the healthcare institutions were unaware that malicious activity had been occurred in their internal networks.

Independent security evaluators conducted a two-year security assessment [131] that included PoC attacks on twelve healthcare facilities with AMDs/PMDs. In some attack scenarios, vulnerable Web applications, that were also connected to the internal hospital network, were used as a initial point-of-entry: Pivoting through the unprotected corporate network the attackers compromised active and passive medical devices in order to achieve their initial goal and retrieve sensitive patient data. In a PoC attack scenario [131], vulnerable PMDs were used in order to disrupt various in-hospital operations. In this attack vector, the first step was to compromise a Web server in order to get initial access to the internal network of the hospital. Then, using network scanning/pivoting techniques vulnerable PMDs were discovered (in this case patient monitors) on various network segments. Finally, after bypassing their authentication mechanisms, they were able to launch a series of attacks, such as enable fake sound alarms or display incorrect patient vital information. The potential impact of such attacks could be very high, since they could be used to affect the treatment received by patients inside hospitals. The assessment revealed that the majority of the PMDs examined were vulnerable and easy to exploit with.

Attacks based on informational IoT devices Another PoC attack scenario [131] demonstrated that non-medical IoT devices connected in the hospital network, may also enable attacks affecting important medical services. This attack was based on a vulnerable vendor information kiosk located inside the hospital's premises that was connected to the hospital's internal network. The first step was to bypass access security controls in order to gain physical access to the kiosk. Then, by exploiting software vulnerabilities, the attackers were able to compromise the kiosk and scan the internal network, since, the device was not on a restricted network zone. They located numerous mobile computer stations in emergency and hospital rooms, one of which, was vulnerable. From the compromised computer the attackers gained access to the medicine and bloodwork barcode scanning device [41]. Through these systems one could view patients' personal data and control the results of the barcode scanning device. In a worst case scenario, this attack could be used to modify patient's therapy by printing falsified labels, contaminating blood samples and/or administer an inappropriate treatment.

3.7 IoT-enabled attacks on Smart home/automation systems and services

Home automation technologies allow users to remotely manage, control and interact with home appliances through their mobile devices, for example, to remotely adjust their air condition, schedule their TV recorder, or monitor their home surveillance system status [289]. Being affordable and readily available to consumers, home automation IoT devices are very popular, by far outreaching all other IoT sectors. Typical devices include smart thermostats, energy management devices, light bulbs, security alarms, locks, smoke detectors, surveillance cameras, home appliances (e.g. smart fridges, coffee makers) and entertainment systems (smart TVs and set-top boxes). Notably, most of the aforementioned home automation systems, are not used only in residential environments, but may also be installed inside critical infrastructure premises, such as factories, hospitals, military, government, financial and transportation facilities. In many occasions smart home systems are able to interact directly/indirectly with critical infrastructures' components, e.g. in the case of smart meters [172, 183, 187, 251].

3.7.1 Architecture

Home IoT devices use various protocols to communicate with each other and/or with the Internet, as briefly described in Figure 3.7. With the absence of a single standard protocol and architecture, many different wired (e.g. Ethernet, Powerline),

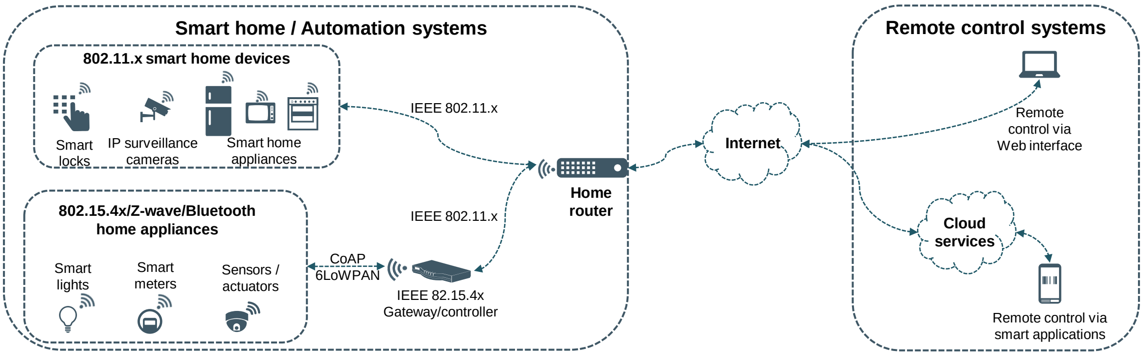

Smart home / Automation systems
802.11.x smart home devices
Smart locks
IP surveillance cameras
Smart home appliances
IEEE 802.11.x
Home router
IEEE 802.11.x
802.15.4x/Z-wave/Bluetooth home appliances
Smart lights
Smart meters
Sensors / actuators
CoAP
6LoWPAN
IEEE 82.15.4x Gateway/controller
Internet
Remote control systems
Remote control via Web interface
Cloud services
Remote control via smart applications

Figure 3.7: Architecture of *home/automation* IoT ecosystem [261]

and more usually wireless (e.g. WiFi, Z-Wave, ZigBee and Bluetooth) technologies are used [102, 276].

Some home devices, such as smart TVs, printers or IP cameras, are usually directly connected to the home router via WiFi connection. On the other hand, resource constrained devices such as smart light bulbs or temperature sensors, usually access the Internet via a low-energy wireless communication interface. Because the IEEE 802.15.4x [197] is suitable for low-rate Wireless Personal Area Networks (WPANs), it is used as the basis for higher-layer protocols, such as Zigbee, 6LoWPan (IPv6 over Low-Power WPAN) or CoAP [35]. ZigBee is a popular low-power wireless mesh networking standard built on top of IEEE 802.15.4. 6LoWPAN [155] is an adaptation layer protocol allowing to transport IPv6 packets over 802.15.4 links, whereas CoAP [35] is an application layer protocol designed to support easy Web integration through an HTTP interface. Only same-profile Zigbee devices can communicate with each other, while bridging between ZigBee and non-ZigBee networks requires a complex IP conversion process. On the contrary, 6LoWPAN offers interoperability with other 802.15.4 devices as well as with devices on any other IP network via a simple bridging device.

Choosing the most appropriate network architecture for an IoT-enabled automation system should take into consideration various criteria, such as device type, cost, power supply and consumption, interoperability, range and bandwidth. For example, Bluetooth, WiFi, ZigBee Light Link (ZLL) Touchlink and Z-Wave are considered to be some of the most prominent wireless network technologies available today for smart lighting applications. ZLL [281] is an industry standard aiming to increase the interoperability between lighting and control products. The ZLL Touchlink protocol allows smart LEDs and control systems to establish WPANs. To secure their communication, ZLL is based on a common *ZLL master key*, embedded in all ZLL certified devices. Unsurprisingly, the master key was leaked during 2015 [59].

Outside the home network, the users can remotely interact and control these devices, either by directly connecting to the them through a Web interface, or through cloud services that enable users to control their devices via smartphone applications provided by the vendors.

3.7.2 Attacks on smart home/automation IoT systems

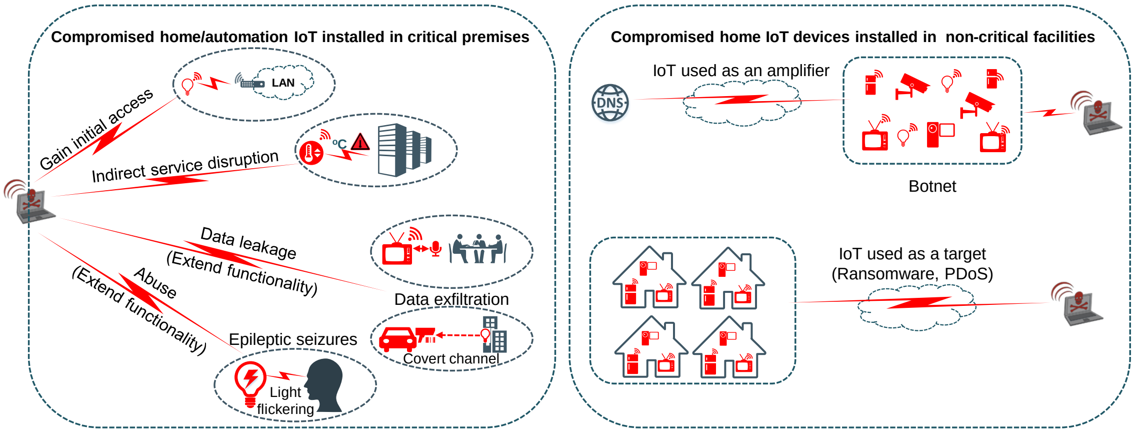


Figure 3.8: Attacks based on smart home devices that may be triggered either by devices that are physically installed near critical systems or by devices installed in non-critical facilities [261]

An analysis of 50 actual home IoT devices in 2015 from security company Symantec [23] identified several common vulnerabilities found in smart home appliances, including weak authentication schemes (e.g., use of weak embedded passwords without even applying "lock out" policies), unauthenticated firmware update process and the use of unencrypted communications. In addition, various Web vulnerabilities were found in many of the applications used to remotely control the devices, or in the relative IoT cloud platforms.

Numerous security researchers [19, 91, 179, 185, 213] have pinpointed security flaws in various wireless protocols used in home IoT devices such as WiFi, ZigBee and Z-Wave. For example, O' Flynn *et al.* [213] presented pulse denial DoS attacks (i.e. block the entire radio frequency spectrum by sending pulses to all channels), node-specific DoS (i.e. detecting and jamming a target node) and interception MiTM attacks (i.e. intercept network traffic and selectively jam communications between nodes to spoof targeted messages) in IEEE 802.15.4 networks.

In a recent disclosure [106] security researchers have revealed a list of default login credentials that correspond to a large number of home routers and more than 1,700 IoT devices. The latter used on just 144 unique username-password pairs for their telnet services authentication. In a report about botnets (e.g. Mirai), that mainly consist of home IoT devices, based on real data collected between January and June 2017, F5 Labs [34] discovered a massive (280%) increase of telnet-based attacks against IoT devices. Intuitively, attacks on IoT devices installed in home environment seem less important than attacks on IoT devices that are used in critical sectors, such as Smart Grids, transportation or hospitals. Note, however, that automation devices used in smart homes may also be installed in the premises of critical infrastructures (e.g. a smart thermostat installed in a data center, or smart lamps installed in a hospital). Although they are only used for secondary and supporting operations, their *physical proximity* with critical systems, may trigger indirect attack paths. Even when they are installed in non-critical, home environment, they can still be used to enable subliminal attacks that may result in high impact (e.g. numerous Internet-connected home IoT devices controlled by botnets in a DDoS attack against a mission critical system).

Bellow we will review real and verified attacks for both cases. Since devices of this category are only used for supporting operations and not as part of a critical control system, we will categorize the attacks based on their actual goal and not based on the underlying system architecture as in the previous sectors. Figure 3.8 provides an overview of possible attacks based on smart home devices installed in both critical and non critical facilities.

3.7.3 Attacks based on devices installed in critical premises

Real and PoC attacks based on home or automation IoT devices installed in critical environments can be classified into the following categories as shown in Figure 3.8: (i) Gain initial access, (ii) indirect disruption or denial of critical services, (iii) data leakage, and (iv) system misuse or abuse attacks. These attacks are usually accomplished by extending the functionality of the devices in unexpected ways. In the following paragraphs we overview such attacks, while in Table 7.6 we analyze the attack vectors and we assess the most characteristic cases, based on real incidents or realistic scenarios.

Gain initial access to an internal network In [49] Chapman demonstrated a series of attacks against WiFi enabled light bulbs. Initially, the firmware of the device was extracted, by using an open source hardware JTAG debugger called *BusBlaster*.

Then, after reverse engineering the firmware, it was possible to retrieve various credentials that were stored in plaintext (unencrypted) form. One of these credentials was a pre-shared cryptographic key that was common for all the lamps of the same model. The key was extracted with the help of a free Advanced Encryption Standard (AES) decryption program. Having access to this key, it was easy to decrypt the WiFi credentials and gain access to the WiFi network that the smart light bulbs are connected to.

In another incident [95], a security expert managed to control various systems of a hotel, by connecting his tablet to an exposed Ethernet socket in his hotel room. Then, after some passive eavesdropping and with the use of a python program available in Github he managed to remotely control the lights, turn the TV on/off and move the curtains of his room. The lack of network security mechanisms (e.g. proper network isolation, use of insecure network protocols - Modbus over TCP) enabled him to seize control of both former and/or other IoT-enabled systems throughout the hotel. Although, in this attack scenario, an adversary needs to be inside hotel's premises, she could potentially affect other resident's safety, violate their privacy, cause discomfort and/or accidents.

Indirect disruption/denial of critical services Fernades *et al.* [124] presented in BlackHat 2014 an attack scenario concerning an IoT-enabled thermostat (Nest) that is designed to remotely control central air conditioning units through the owner's WiFi network. The device can also communicate with other Nest devices via Zigbee and connect to the Nest cloud service to upload usage statistics, that can be used by energy providers to improve energy efficiency. By exploiting embedded communication interfaces and vulnerabilities in the boot process, they managed to install their custom rootkit and Linux kernel, thus ensuring persistence and remote control over the device even after a firmware update. In a worst case scenario where a compromised smart thermostat is installed in a critical infrastructure such as a data center room, a DoS attack could be launched just by altering the room temperature which, in turn, would force the servers to malfunction and/or shutdown.

Data leakage (covert channels) On March 2017, Wiki-Leaks published documents that revealed a CIA project named *Weeping Angel* [286]. Based on the leak, the program included various hacking capabilities that allowed breaking into various devices connected to the Internet such as smart TVs and smartphones. Of a particular interest for our case is the ability to use the microphone of some smart TV models connected to the Internet, to create covert channels. The document describes that it is possible to place a target TV in a *fake-off* mode. Then, by having the owner to falsely believe that the smart TV is off, the microphone can be used to record conversations in the room and then send them over the Internet to a covert server. The attack exploited several known and unknown software and network vulnerabilities. Obviously, such attacks could be used by agencies or nation state adversaries to leak data from very sensitive environments that host vulnerable smart TVs.

Ronen and Shamir [232] demonstrated various PoC attacks based on smart LEDs. One of the attacks exploits the lack of encryption and integrity protection in the communication between the controller and the smart LEDs in order to create a covert channel. Since the controller's API did not enforce *input validation* on the commands, the researchers were able to extend the functionality of the device. Through a customized payload they were able to modify the PWM (Pulse Width Modulation) signals, a function available for dimming the LEDs. By controlling the PWM signals, the researchers were able to cause the bulbs to produce an accurately timed, unnoticeable to human eye, increase/decrease in the brightness level (flickering). Then, by using a laptop, a light sensor, an Arduino board and telescope, they managed to convert these slight brightness changes into usable data from a distance up to 100 meters. Now consider the following scenario: An adversary remotely controls a similar vulnerable smart lighting system [233], indirectly connected (e.g. through the WiFi controller) to a mission critical system which she has already compromised. By extending the functionality of the light bulbs (flickering) she can then create a covert channel and exfiltrate sensitive data, *without being detected* by any computer security system.

System misuse/abuse attacks In the same work [232], Ronen and Shamir describe a second attack scenario where an adversary could exploit LED flickering in order to cause *epileptic seizures*. Strobes of light at specific frequency ranges are known to affect people suffering from photosensitive epilepsy. In a worst case scenario, a similar attack against numerous vulnerable smart lighting systems, installed in hospitals and/or public places, could have a severe impact on public confidence, safety and health.

3.7.4 Attacks based on devices installed in non-critical facilities

IoT devices, that are installed in non-critical facilities (e.g. homes, offices), may still be used as an attack enabler. We classify these attacks into two categories as shown in Figure 3.8: (i) Attacks that use a large number of home IoT devices to amplify an attack against a critical system and (ii) attacks whose actual target are home

IoT devices, but at very large numbers. Table 7.7 provides a detailed analysis of the attacks presented bellow.

Home IoT used as an amplifier This category usually includes DDoS attacks that exploit the availability of many unsecured IoT devices to create a botnet and amplify the attack against the actual target. In 2014, a security service provider (Proofpoint), reported a cyberattack incident that involved thousands of smart home devices [220]. The global attack campaign involved more than 750,000 malicious email communications, typically sent in bursts of 100,000 three times per day, targeting enterprises and individuals worldwide. The attack involved more than 100,000 everyday consumer gadgets such as home-networking routers, connected multimedia centers, TVs and refrigerators.

Another incident was realized on October 2016 [58] [111]. A coordinated DDoS attack against the DYN Domain Name System (DNS) service, at rate that exceeded 600 Gbps, paralyzed the Internet. The attack prevented customers from reaching more than 1,200 domains, including major domains like Amazon, Twitter, GitHub, Spotify, PayPal, Verizon, and Comcast. The attack originated from a botnet named *Mirai* [212] which included approximately 100,000 of infected IoT-enabled digital devices, such as home routers, surveillance cameras and DVRs. The attack was implemented mainly based on "old-fashioned" TCP SYN flood requests as well as *subdomain* attacks [204] that aimed directly at the port 53 of DYN DNS servers. Most of the infected home IoT-enabled devices had password vulnerabilities (use of default or weak passwords) and/or operating system vulnerabilities.

Various attack scenarios against Belkin's WiFi-based products (over 1.5 million sold) and cloud platform for smart home, named *WeMo*, have been recently presented [67, 271]. In these PoC attacks, the researchers managed to execute arbitrary code through SQL injection and take over the device(s) remotely, bypass local authentication mechanisms by connecting to the UART interface of the device and exploit vulnerabilities found in the WeMo app.

Home IoT used as a target (concurrent attacks) The actual target of this category are the IoT devices themselves. The importance of such attacks comes from their massiveness, e.g. concurrently threaten a huge number of such devices with Permanent DoS (PDoS) or ransomware.

In [233] Ronen *et al.* demonstrated how an adversary can take-over a smart lamp and self-propagate the attack in a worm-like manner. The basic idea was to bypass the proximity check mechanism that smart lights use when joining a network, fool them to join to a malicious network and, through the OTA update process, install a modified firmware to take control of the device. To bypass the proximity check a flaw in Atmel's BitCloud Touchlink implementation was used. In order to retrieve the embedded hardware key, differential [147] and correlation [146] power analysis techniques were used. Then, the researchers utilized the recovered key so as to authenticate a firmware file which had previously infected with malicious code. This enabled them to perform various attacks, such as permanently bricking the devices (PDoS) or use them to jam [213] nearby wireless networks that operate in the same band. Notably, the 2.4 GHz license-free band (IEEE 802.15.4x), is also used in other sectors (industrial, medical) and various protocols (WiFi, WirelessHART, MiWi, ISA 100.11a, 6LoWPAN, Nest Weave, JenNet and Threat).

For interoperability, the ZLL protocol allows non-ZLL devices under application control to join a ZLL network without any proximity check [200]. This is allowed only when the device is in *"Factory new"* state which can be achieved by sending a unicast *"Reset to Factory new"* request to the smart light. The device is then forced to scan for nearby ZigBee networks. By sending a ZigBee beacon message, an adversary can fool the device to join a network. To launch a self-propagating attack, factory reset messages were initially sent through the primary channels of the 802.15.4 wireless network whereas for beacon and association messages the secondary channels were used. In that way, devices that had already joined the attackers' network did not respond to any new factory reset messages. Through this technique the infection could spread to all nearby devices of the same type just from a single infected lamp.

Although an attack scenario involving smart lighting systems may seem of low importance, one may want to consider the potential impact of an attack that concurrently bricks numerous smart lighting systems installed throughout a Smart City. The researchers proved that such a scenario is realistic via techniques, such as *war driving* or *war flying* that enabled them to launch the attack from distances up to 350 meters.

In [84] Fernades *et al.* presented a thorough analysis of vulnerabilities and attack scenarios against 499 smart home control applications and 132 device handlers. Using static code analysis techniques, the researchers discovered that more than 55% of the examined applications were over-privileged and lacked of basic protection mechanisms for sensitive data such as door lock codes. Then, they demonstrated possible attack scenarios on an IoT-enabled home surveillance system which included door lock codes' theft/alternation, disable of the *vacation mode* as well as issuing fake fire alarms.

Several researchers [42, 86] have conducted security tests on smart TVs. They discovered that through MiTM attacks, an attacker could redirect unauthenticated, unencrypted (HTTP) requests (e.g. in the case of downloading firmware/applications) to malicious sites and gain control over the devices. In [242], Scheel demonstrated an attack in which, an adversary is able to remotely take over a plethora of smart

TVs by sending specially crafted TV stream DVB-T signals (HbbTV commands) to gain root access. The attack utilizes two known security flaws of the embedded Web browsers and applies to 90% of smart TVs, sold in the last few years.

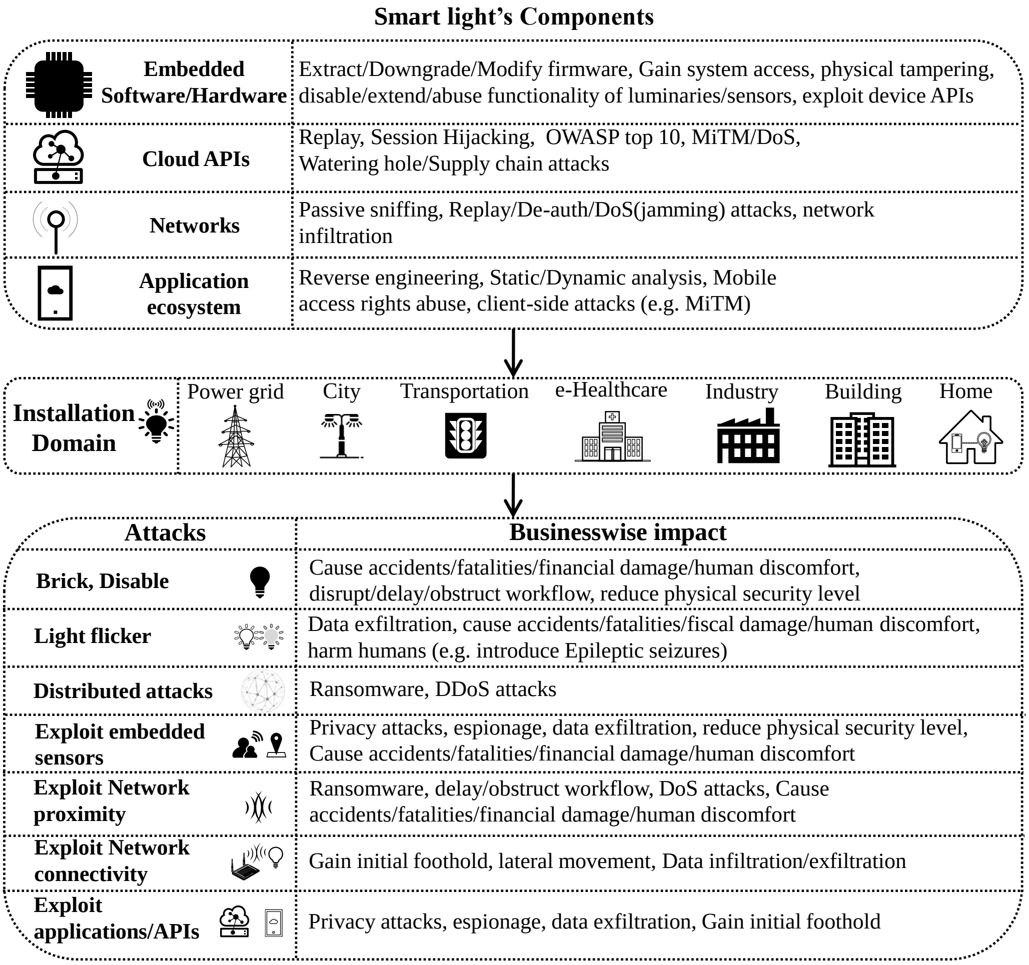
Morgner *et al.* [198] presented a series of attacks based on known vulnerabilities of the ZLL protocol. The attacks were distinguished in two main categories: These that do not require any use of cryptographic protocols (blink, reset, DoS) and those that require access to the ZLL master key (hijack, network key extraction and command injection). The target systems included popular lighting models, such as Philips Hue, Osram Lightify and GE Link. Their goal was to demonstrate a series of attacks against the ZLL protocol, by utilizing its master key vulnerability [59] and the unsecured Inter-PAN frames, used for the communication between different personal area networks (PANs). Other security reports, which involve home IoT devices, include attacks on home robots [48] and on home cameras (privacy violations) [214].

3.8 Security evaluation of a popular smart lighting system: A hands-on approach

Smart lighting systems combine the state-of-the-art lighting technology including Light Emitting Diode (LED) and/or Organic LED (OLED) with sensors (e.g. ambient light, acoustic, ultrasonic, infrared, location), wireless network interfaces, (e.g. Ethernet, WiFi, Z-Wave, ZLL), vendor-specific application software, as well as cloud services (e.g. If-This-Then-That – IFTTT) [189], in order to enable remote control, interoperability and autonomous operation. Via these features they achieve optimization of energy consumption, visual comfort, safety, remote control and adaptability in various environments. Their vast adoption has lead to a significant production cost reduction which, in turn, resulted in making them one of the most wide-spread IoT technologies.

Smart lighting systems can be remotely managed via smartphone applications that utilize local and/or remote connectivity through cloud services. Popular smart lighting systems utilize Apple's HomeKit (Siri), Amazon's Echo (Alexa) and Google Home, in order to enable remote control via voice commands. Near-future advanced intelligent lighting technologies may include, real-time luminosity and spectrum self-adjustment capabilities.

As previously described (see subsection 3.7.2), depending on the installation site, the attacks on smart lighting systems can attract diverse types of attackers, ranging from security enthusiasts up to highly skilled and motivated adversaries such as nation-state and organized cybercrime. An adversary can take advantage of existing vulnerabilities and characteristics of an IoT-enabled automation system such as smart lights to abuse and/or extend its functionality (e.g. sensing capabilities, network connectivity, wireless adaptor's operating frequency and available luminosity levels) and launch a variety of cyber-physical attacks against nearby critical systems or even people [232, 233]. For example, existing vulnerabilities in smart lighting systems (e.g. CVE-2020-6007) may lead to a network infiltration attack, which in turn, may have a significant impact on organizations such as banks or pharmaceutical companies. In Figure 3.9 we depict potential attack vectors and the corresponding businesswise impact on popular installation domains of a rural environment.



Smart light's Components

Embedded Software/Hardware	Extract/Downgrade/Modify firmware, Gain system access, physical tampering, disable/extend/abuse functionality of luminaries/sensors, exploit device APIs
Cloud APIs	Replay, Session Hijacking, OWASP top 10, MiTM/DoS, Watering hole/Supply chain attacks
Networks	Passive sniffing, Replay/De-auth/DoS(jamming) attacks, network infiltration
Application ecosystem	Reverse engineering, Static/Dynamic analysis, Mobile access rights abuse, client-side attacks (e.g. MiTM)

Installation Domain	Power grid	City	Transportation	e-Healthcare	Industry	Building	Home

Attacks	Businesswise impact
Brick, Disable	Cause accidents/fatalities/financial damage/human discomfort, disrupt/delay/obstruct workflow, reduce physical security level
Light flicker	Data exfiltration, cause accidents/fatalities/fiscal damage/human discomfort, harm humans (e.g. introduce Epileptic seizures)
Distributed attacks	Ransomware, DDoS attacks
Exploit embedded sensors	Privacy attacks, espionage, data exfiltration, reduce physical security level, Cause accidents/fatalities/financial damage/human discomfort
Exploit Network proximity	Ransomware, delay/obstruct workflow, DoS attacks, Cause accidents/fatalities/financial damage/human discomfort
Exploit Network connectivity	Gain initial foothold, lateral movement, Data infiltration/exfiltration
Exploit applications/APIs	Privacy attacks, espionage, data exfiltration, Gain initial foothold

Figure 3.9: Potential businesswise impact from attacks against smart lighting systems installed on critical domains [262]

Although several security vulnerabilities for smart lighting devices have been identified in the recent literature (e.g. [174, 270]), still various systems and components have not been tested. In addition, most security researchers and bounty hunters are focused on individual components of each smart lighting device or service rather than follow a holistic approach and assess vulnerabilities found on the smart lighting ecosystem and particular in hardware, embedded software, radio networks, applications and cloud services.

The examined smart lighting system comprises of a smart light controller and a light bulb with the former to have the following features and specifications:

- A Software restore push button, a region specific alternating current plug and a LED to indicate power and WiFi status.
- An internal IEEE 802.11 b/g/n WiFi radio 2.4 GHz antenna for communicating with the local network and the Internet.
- An internal IEEE 802.14.5x b/g/n ZigBee radio 2.4 GHz antenna Home Automation 1.2 Certified for communicating with the light bulb.
- One Spatial stream.
- Works with *If-This-Then-That* Web platform that is used to connect to other Web applications.

Hardware components Disassembling the smart light controller enabled us to access its main circuit board. In particular, we managed to locate: the flash memory chip (*winbond 25Q128FVSG 1603*) with 16MB serial NOR flash memory that communicates over a Serial Peripheral Interface (SPI); the *winbond W9825G6KH-61 1513P 643803400ZU* chip that contains the Electrically Erasable Programmable Read-Only Memory (EEPROM) of the device; the *Ralink RT5350F TP4KW33609 1601STA1* WiFi controller; the *EM357 1536A00MB8 TM ARM (e3)* 802.15.4x/ZigBee controller; and finally a UART debugging interface, which enables communication with any device equipped with a universal bus interface such as *Bus Pirate*, accessible via a three pin layout.

Smart lighting system mobile application and cloud services.

The mobile application is available for both Android and IOS platforms. It is used for setting up, remote control and firmware update of the smart light controller and light bulb(s). During our research we conducted both dynamic as well as static analysis of the application. Via static analysis, hardcoded information such as domain names, emails, passwords, encryption and/or verification keys were retrieved. Via dynamic analysis, the interaction of the application with the devices and/or the cloud servers can be further examined.

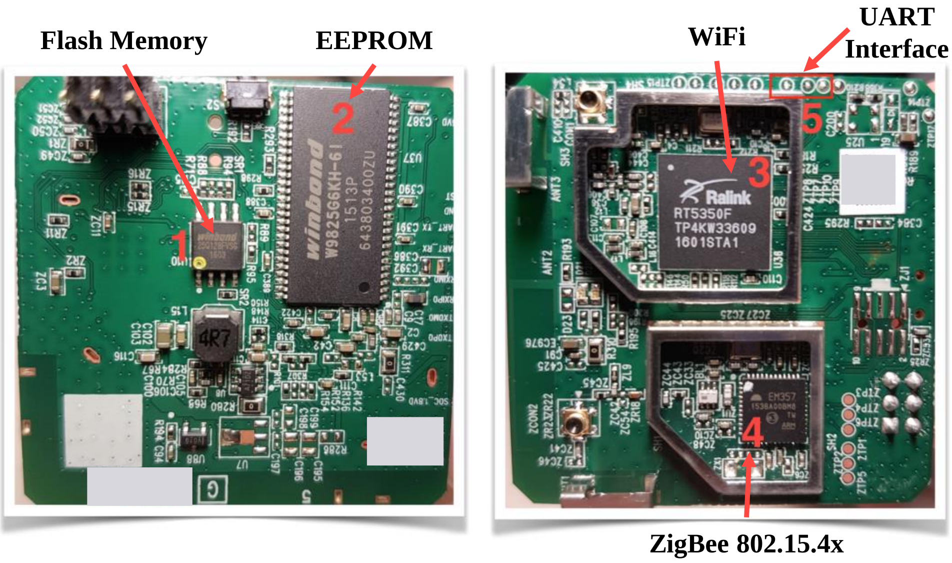


Figure 3.10: Printed circuit of the tested control device and its main components [262]

Radio communications. By utilizing open source tools and a HackRF radio antenna, we analyzed the network traffic between the mobile application, the cloud servers and the control device. In particular, we examined both 802.15.4.x (ZLL) and 802.11.b/g/n, 2.4 GHz, network interfaces that the control device utilizes to communicate with the smart light bulbs and the mobile application/cloud services respectively.

3.8.1 Embedded software vulnerabilities

Physical access to the device is required in most of the security testing cases, whereas, attacks such as firmware extraction where easy to perform since there were no anti-tampering mechanisms (e.g. [75]). Furthermore, firmware modification is considered plausible since the U-Boot partition resides on the same Electrically Erasable Programmable Read-Only Memory (EEPROM) (*winbond W9825G6KH-61 1513P 643803400ZU*) chip which contains the firmware of the device. In addition via the available JTAG connector access to the system software is feasible. During the tests it was discovered that downgrade attacks can be potentially take place via local network and/or the Internet, due to the fact that the communication with the servers is done via plain HTTP and the existing update script does not implement any logical checks on version numbering.

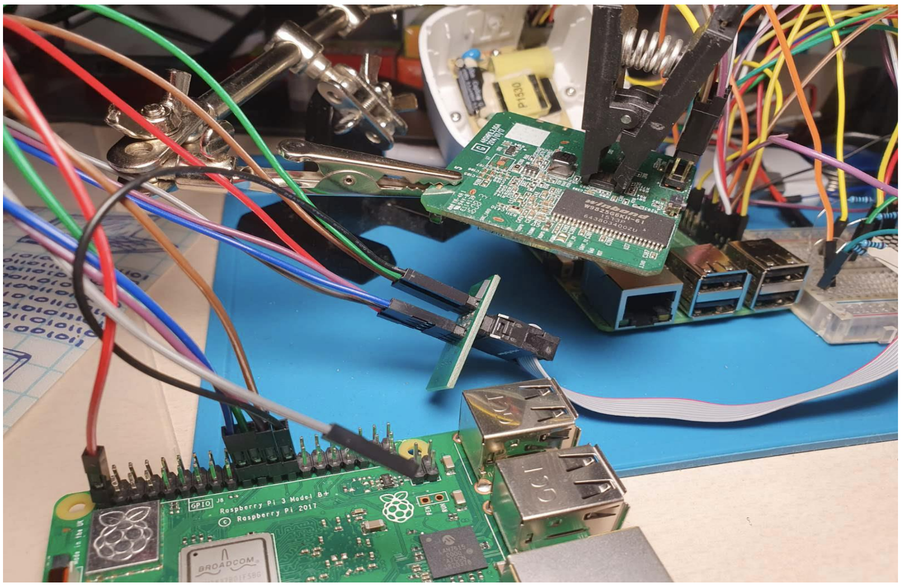


Figure 3.11: Lab setup for firmware extraction [262]

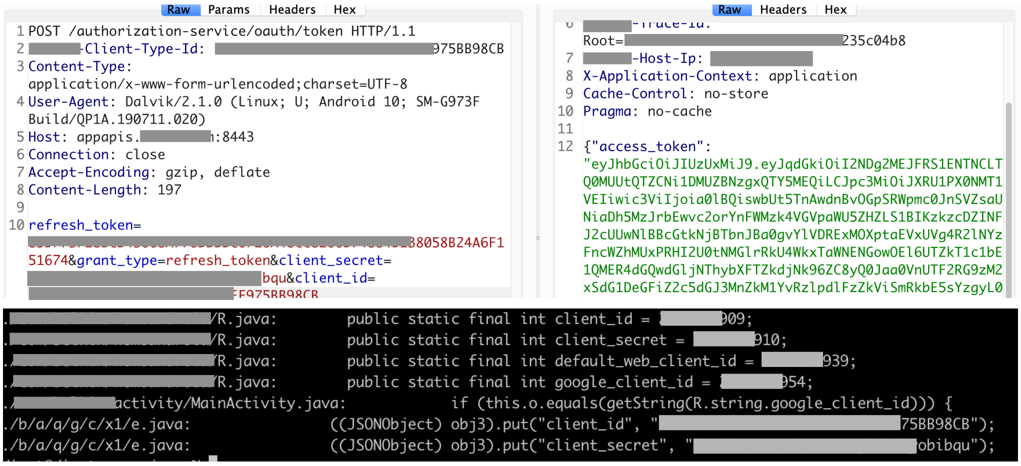
3.8.2 Mobile application security analysis

The manufacturer supports Both Android and IOS platforms. The mobile application can be used in order to configure, control and perform firmware updates of the controller and light bulbs. In order to discover as much information as possible we submitted the application in dynamic as well as static analysis (android version).

Static analysis revealed hardcoded information such as firmware update domains, email addresses and hardcoded secret IDs. Although some level of obfuscation in the source code prevented from determining all functions and secrets, modification and recompile the reversed source code was possible thus allowing an adversary to perform dynamic analysis techniques including MiTM attacks with the cloud servers. Dynamic analysis also revealed that during the authentication procedure with the cloud servers hardcoded secrets in the application were used, which in turn, can potentially compromised the whole authentication process. In addition, misconfigurations including an extended (3-day) validity period of the authentication cookie and a clear text transmission of the password during the reset process can be exploited in various occasions.

3.8.3 Assessing the security of Cloud Servers/APIs

MiTM attacks during the application's firmware update functionality revealed the location of the firmware update server. The latter discovered to be publicly available from the Internet, thus leaking several (aprox. 1400) unencrypted update firmware files of several types of IoT devices including smart light bulbs, light switches, control-links, air purifiers, dimmers, relays and coffee maker machines dating back to 2016. Via custom scripts and open source tools extraction of the filesystems of most of the firmware files was possible. An analysis of the finding give away common root passwords across devices and firmware versions (just four different passwords for all firmware files), whereas, in most cases, the root was the only user in the system. Furthermore, the operating system was mostly based on an obsolete version 10.03 when, the latest release is 21.02.0-rc4[3]). Web servers were serving their content via plain HTTP and/or obolete versions of HTTPS such as Secure Sockets Layer - SSL 3.0 and Transport Layer Security - TLS 1.0 with several vulnerabilities. Web application software was outdated which resulted in, among others, a potential Remote Code Execution (RCE) vulnerability (CVE-2019-0232) and a vulnerability with publicly available exploit [4]. Web servers were also susceptible to Cross Site Scripting (XSS) and data injection attacks. All of the above can act as an enabler for a remote attacker to launch potentially stealthy, high-impact attacks (e.g. supply chain/waterhole) with minimum effort.



```
POST /authorization-service/oauth/token HTTP/1.1
[redacted]-Client-Type-Id: [redacted]975BB98CB
Content-Type:
application/x-www-form-urlencoded;charset=UTF-8
User-Agent: Dalvik/2.1.0 (Linux; U; Android 10; SM-G973F
Build/QP1A.190711.020)
Host: appapis.[redacted]:8443
Connection: close
Accept-Encoding: gzip, deflate
Content-Length: 197

refresh_token=
[redacted]38058B24A6F1
51674&grant_type=refresh_token&client_secret=
[redacted]bqu&client_id=
[redacted]FF975BB98CB
```

```
[redacted]-Trace-Id:
Root=[redacted]235c04b8
[redacted]-Host-Ip: [redacted]
X-Application-Context: application
Cache-Control: no-store
Pragma: no-cache

{"access_token":
"eyJhbGciOiJIUzUxMiJ9.eyJqdGkiOiI2NDg2MEJFRS1ENTNCLT
Q0MUUtQTZCNi1DMUZBNzgxQTY5MEQiLCJpc3MiOiJXRU1PX0NMT1
VEIiwic3ViIjoia0lBQiswbUt5TnAwdnBvOGpSRWpmc0JnSVZsaU
NiaDh5MzJrbEwvc2orYnFWMzk4VGVpaWU5ZHZLS1BIKzkzcDZINF
J2cUUwNlBBcGtkNjBTbnJBa0gvYlVDRExMOXptaEVxUVg4R2lNYz
FncWZhMUxPRHI2U0tNMGlrRkU4WkxTaWNENGowOEl6UTZkT1c1bE
1QMER4dGQwdGljNThybXFTZkdjNk96ZC8yQ0Jaa0VnUTF2RG9zM2
xSdG1DeGFiZ2c5dGJ3MnZkM1YvRzlpdlFzZkViSmRkbE5sYzgyL0
```

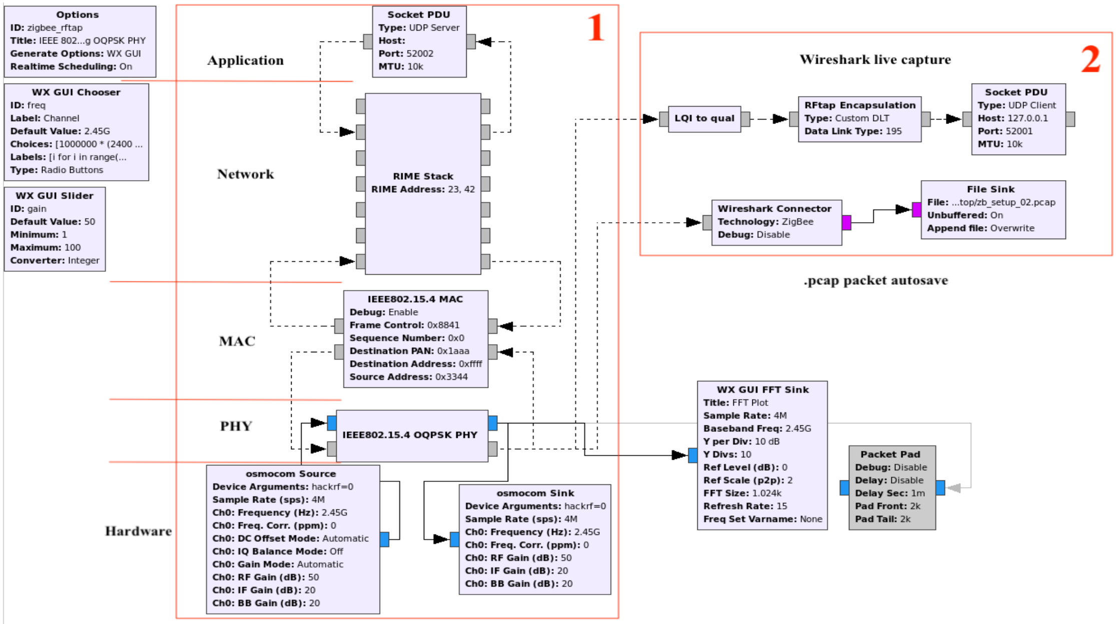
```
.[redacted]/R.java:        public static final int client_id = [redacted]909;
.[redacted]/R.java:        public static final int client_secret = [redacted]910;
.[redacted]R.java:        public static final int default_web_client_id = [redacted]939;
.[redacted]R.java:        public static final int google_client_id = [redacted]954;
.[redacted]activity/MainActivity.java:            if (this.o.equals(getString(R.string.google_client_id))) {
./b/a/q/g/c/x1/e.java:                ((JSONObject) obj3).put("client_id", "[redacted]75BB98CB");
./b/a/q/g/c/x1/e.java:                ((JSONObject) obj3).put("client_secret", "[redacted]obibqu");
```

Figure 3.12: Intercepted network packets during the authentication process with the cloud servers [262]

[3] https://downloads.openwrt.org/releases/

[4] https://www.exploit-db.com/exploits/47073

3.8.4 Network vulnerabilities



Options
ID: zigbee_rftap
Title: IEEE 802...g OQPSK PHY
Generate Options: WX GUI
Realtime Scheduling: On

WX GUI Chooser
ID: freq
Label: Channel
Default Value: 2.45G
Choices: [1000000 * (2400 ...
Labels: [i for i in range(...
Type: Radio Buttons

WX GUI Slider
ID: gain
Default Value: 50
Minimum: 1
Maximum: 100
Converter: Integer

Socket PDU
Type: UDP Server
Host:
Port: 52002
MTU: 10k

RIME Stack
RIME Address: 23, 42

IEEE802.15.4 MAC
Debug: Enable
Frame Control: 0x8841
Sequence Number: 0x0
Destination PAN: 0x1aaa
Destination Address: 0xffff
Source Address: 0x3344

IEEE802.15.4 OQPSK PHY

osmocom Source
Device Arguments: hackrf=0
Sample Rate (sps): 4M
Ch0: Frequency (Hz): 2.45G
Ch0: Freq. Corr. (ppm): 0
Ch0: DC Offset Mode: Automatic
Ch0: IQ Balance Mode: Off
Ch0: Gain Mode: Automatic
Ch0: RF Gain (dB): 50
Ch0: IF Gain (dB): 20
Ch0: BB Gain (dB): 20

osmocom Sink
Device Arguments: hackrf=0
Sample Rate (sps): 4M
Ch0: Frequency (Hz): 2.45G
Ch0: Freq. Corr. (ppm): 0
Ch0: RF Gain (dB): 50
Ch0: IF Gain (dB): 20
Ch0: BB Gain (dB): 20

LQI to qual

RFtap Encapsulation
Type: Custom DLT
Data Link Type: 195

Socket PDU
Type: UDP Client
Host: 127.0.0.1
Port: 52001
MTU: 10k

Wireshark Connector
Technology: ZigBee
Debug: Disable

File Sink
File: ...top/zb_setup_02.pcap
Unbuffered: On
Append file: Overwrite

WX GUI FFT Sink
Title: FFT Plot
Sample Rate: 4M
Baseband Freq: 2.45G
Y per Div: 10 dB
Y Divs: 10
Ref Level (dB): 0
Ref Scale (p2p): 2
FFT Size: 1.024k
Refresh Rate: 15
Freq Set Varname: None

Packet Pad
Debug: Disable
Delay: Disable
Delay Sec: 1m
Pad Front: 2k
Pad Tail: 2k

Figure 3.13: Flowgraph of ZigBee network protocol (1) and Wireshark connectors with live capture (2) and pcap autosave [262]

Since most of not all wireless networks are prone to deauthentication attacks we managed to successfully exploit this feature in both ZLL and WiFi network interfaces of the control device. Via a HackRF antenna and by utilizing open source software[5,6,7] we managed to effectively jam both ZLL as well as WiFi signals which resulted in an extensive period of unresponsiveness among the control device and smart light bulbs and/or the connection in the local WiFi. In addition, during the initialization of the device's WiFi no security measures (e.g. encryption) are enforced. Furthermore, since the the ZLL master key was leaked [59] the network key can be retrieved by intercepting the network pairing process thus enabling an adversary to gain access to both networks and launch a series of passive and active (sniffing and replay) attacks. Several other attack vectors may include signal interference (jamming), unauthorized network commissioning as well as DDoS attacks.

In order to enable remote management and control the specific IoT device utilizes Universal Plug and Play (UPnP) protocol to communicate with the cloud APIs. Via

[5] https://osmocom.org/

[6] https://github.com/bastibl/gr-foo

[7] https://rftap.github.io/

SHODAN search engine[8], we managed to locate several IoT devices and retrieve potential useful information such as Medium Access Control (MAC) address and firmware version.

[8] https://www.shodan.io/

CHAPTER 4
ADVANCED PERSISTENT THREATS (APT) IN INDUSTRIAL IOT ECOSYSTEM

In this chapter, we are going to further analyze IoT-enabled attacks, most of which are presented in Chapter 3 (e.g. [81, 156]), in order to understand to a greater extent the techniques and tactics that high-profile adversaries utilize in order to achieve their malicious goals. In particular, we are going to focus on industrial and energy sector since, they usually attract well-funded, high-skilled and strongly motivated adversaries that seek to gain substantial economic profit (e.g., cybercriminals) or to disrupt a nation's CIs (e.g., nation state adversaries). These attacks are considered of high impact due to the effect that SCADA systems have on our every day life and are mainly distinguished for their sophisticated tactics and advanced exploitation techniques. Such types of attacks can enable attackers to remotely infiltrate and exploit air-gaped systems with proprietary technologies, as well as to remain undetected for a long period of time, even with strong security countermeasures in place.

4.1 Zero-Day vulnerabilities on Human-Machine Interface applications

HMI software is considered to be the most critical application in Industrial IoT ecosystem since it its main purpose is to administer mission-critical SCADA systems. Compromising an HMI system may lead to a series of attacks ranging from information gathering, deactivation of notification systems (e.g. alarms, notifications to operators) up to physically damage industrial equipment. To make things worse, HMI vendors do not always enforce security best practices on the controlling software, thus focusing only on the managed devices. In this Section we present the findings of an extensive research conducted by the Zero Day Initiative (ZDI) team of Trend Micro security company throughout a two-year period (2015-16), which successfully identified 250 *zero-day* vulnerabilities on HMI applications [39]. During the disclosure process, researchers observed that the average time period required by the vendors to release a corresponding patch of a *zero-day* exploit averaged to 150 days. This actually meant that mission-critical SCADA systems were vulnerable for almost five months before a patch was available from software vendors. The exploitation techniques were classified into 4 main categories: (i) Memory corruption, (ii) credential harvesting, (iii) insecure installation, authentication and authorization procedures and (iv) code injection. These exploitation techniques which can be used in various APT attack scenarios are described in detail in the following sections.

4.1.1 Memory corruption

Memory corruption issues accounted for the 20% of the total number of vulnerabilities found. The majority were stack/heap-based buffer overflows [62] and out-of-bounds read/write vulnerabilities. In a particular vendor, the software Advantech WebAccess HMI Solution was proved to have a vulnerable `sprintf` function and no protection mechanisms such as stack cookies, Address Space Layout Randomisation (ASLR) [247] and SafeSEH [115]. Due to the absence of ASLR protection an adversary needs only to overwrite the return address to a controlled Return Oriented Programming (ROP) chain, in order to execute malicious code with elevated privileges. Even though the vendor issued a large number of patches these corrected only specific issues and did not address the problem globally or replaced other problematic functions.

4.1.2 Credential harvesting

Vulnerabilities found in credential management represented the 19% of the overall vulnerabilities found. These included the use of hard-coded passwords as well as insecure storage and/or protection of passwords (e.g., stored clear text/with reversible encryption algorithms). Furthermore, in a particular case study of General Electric (GE) MDS PulseNET[1], a software that is used to monitor industrial equipment and communication networks deployed in energy, water, and waste water sectors globally, they managed to identify an embedded account with full privileges apart from the administrator and user account (CVE-2015-6456 [12]). By utilizing *HeidiSQL* tool they managed to extract the `ge_support` account as well as the password's MD5 hash value (`Pu1seNET`). Notably, even after a successful logging process of the discovered account its username did not appear in the user management screen.

4.1.3 Insecure installation, authentication and authorization procedures

This category represents the 23% of the total vulnerabilities found, including unencrypted communications, such as the transmission in plaintext of sensitive information (e.g., usernames or passwords), as well as vulnerable ActiveX controls which where marked as 'safe'. In another case study concerning Siemens SINEMA[2] Server, a network management software for monitoring and diagnostics, a misconfiguration

[1] https://www.gegridsolutions.com/communications/pulsenet.htm

[2] https://new.siemens.com/global/en/products/automation/industrial-communication/industrial-remote-communication/remote-networks/sinema-remote-connect-access-service.html

allowed standard authenticated users to have full access to Windows sensitive system folders (CVE-2016-6486). In addition, the binary code used to start the SINEMA service run at local system level, thus allowing an adversary with local access to the workstation, to replace the legitimate binary code with a malicious one. Then, triggering a reboot allowed the adversary to execute the malicious code with system privileges.

In another case study considering Advantech WebAccess[3], a cross-platform user interface management based in HTML5, an authenticated user was able to retrieve the passwords of other platform's users including the administrator.

4.1.4 Code injection

Although SQL and Operating System (OS) command injections occupy a small fraction (9%) of the overall vulnerabilities discovered, the impact of such threats on HMI systems is considered to be very high, especially those injections that apply to domain-specific languages for SCADA software solutions. In a particular case study, 'Cogent DataHub', a real time visualization software for complex SCADA systems, was evaluated. The application incorporates *Gamma*, a domain-specific script language that contains built-in features and functions for SCADA systems. Cogent DataHub also includes a database, that resides in server's memory providing interchange of data for OPC and other Windows applications. Researchers discovered that it is possible for an attacker to take advantage a flaw in the `EvalExpression` method of *Gamma* script language and enable the insecure processing mode in the Asynchronous JavaScript and XML (AJAX) Web server, resulting in a remote code execution on the server.

4.2 APT attack scenarios on Industrial IoT field devices

APTs' attack vectors usually include the following basic phases:

- **Reconnaissance/data gathering and host discovery phase:** Gathering valuable information regarding a corporation's employees and executives, enumerating its Web presence and compromising corporate email accounts to launch a series of spear phishing campaigns [43, 105, 210] are considered to be the most prevailing methods in the early stages of an APT attack scenario. In addition, Web search engines (e.g., Shodan[4]) can be also used to locate Web exposed industrial equipment that then can be enumerated for vulnerabilities before the exploitation/ initial infection phase begins [39, 90, 181, 253].

[3] https://www.advantech.com/industrial-automation/webaccess

[4] https://www.shodan.io/

- **Initial infection phase:** Since corporate users must communicate with the outside world and, at the same time, are usually connected (directly or indirectly) to mission critical industrial control systems are considered to be the prime target for adversaries. This is usually accomplished by launching spear-phishing campaigns, which include the process of sending malware infected, office documents and malicious Web links from hijacked corporate/legitimate accounts (e.g., [43, 105]). Another more direct approach is to exploit the Web interfaces of modern industrial equipment, that utilize IoT enabling technologies, in order to be able to be operated, managed and updated remotely (e.g., [39, 90, 181, 253]). In addition, it is common practice for manufacturers as well as companies that provide technical support to industrial equipment, to distribute essential software components and/or updates (e.g., IIoT devices' firmware and relative management software) via vulnerable websites and unsecured methods (e.g. HTTP), with devastating consequences on IIoT ecosystem [210]. Finally, off-line exploiting techniques can be also used, as presented in [81, 156].
- **Establish and maintain remote access:** Asynchronous communication, data masquerade and encryption, Intrusion Prevention/Detection System (IDS/IPS) evasion and privilege escalation are some of the techniques used in order to achieve stealthiness. To ensure access persistence, payloads are made so as to withstand power loss/reboot processes and equipped with auxiliary communication modules for redundancy.
- **Lateral movement and propagation phase:** In APT attack scenarios, adversaries utilize several enumeration and pivoting techniques (e.g., probing nearby systems for open ports, connect to default drive shares, spread to different network segments) in order to locate and exploit other mission critical vulnerable ICT equipment such as control rooms' workstations and IIoT devices.
- **Remote control and device manipulation:** Attackers must incorporate a series of well established and new industrial network protocols in order to remotely communicate and ultimately take control the IIoT device(s). The payloads installed on IIoT devices must be able to run with minimum resources and hide their code so as to avoid detection from machine operators.

Functionality plays an essential role when designing payloads that target industrial equipment, since, adversaries must be able to issue arbitrary commands and even control all functions and features of the IIoT device/system. The latter enables adversaries to lock out legitimate operators thus preventing them from responding to the threat accordingly [43]. In many cases of APT attack scenarios the adversaries include payloads that are used to renter the devices and systems affected unusable and/or hide their footprints (e.g., [43, 105, 210]).

4.2.1 Stuxnet

The most well-known APT attack against SCADA systems, that managed to infect the software of at least 14 industrial sites in Iran, including a uranium enrichment plant, is considered to be *Stuxnet* [81, 156]. First discovered in an Iranian computer the week of June 24, 2010 by Sergey Ulasen, a security researcher of a small antivirus company VirusBlokAda [138], this sophisticated piece of malware had been in the wild since at least 2009. After the initial discovery by Sergey Ulasen, a security researcher Pierre-Marc Bureau of ESET security company found further digital evidence that linked back to the initial findings from VirusBlokAda and Stuxnet: The worm had evolved to bypass the new security controls imposed by the initial findings.

By examining the malicious code the researchers managed to pinpoint several versions of the virus thus revealing that the adversaries had launched the attack at least three different waves, one in June of 2009, then in March and April of 2010, changing the malicious code in the process to evade detection. The fact that the attackers had such sophisticated knowledge, access to hard-to-find resources including software manufacturers' digital certificates and that the campaign was not targeting any banking or other sector lead the researchers to believe that the attackers were nation-state actors. In addition, the main Stuxnet's binary was unusually large - 500 KiloBytes (KB) when compared with other viruses of that time. For example, *Confilcker* worm, that managed to infect more than 6 million computers, was just 35 KB in size. This belief was further solidified when a German security researcher, Frank Boldewin, discovered that the attackers were only interested in attacking Siemens SIMATIC Step 7 software and/or the SIMATIC WinCC program. The reverse engineering of the code revealed, among others, a large encrypted Dynamic Link Library (DLL) file, a configuration file with settings that enable the attackers to fine tune the payload by adjusting the URL of the Stuxnet's C&C servers or the number of computers that the malware should infect via a USB flash drive before shutdown and even an infection stop date (June 24, 2012) [304].

Symantec security researchers after analyzing data of real Stuxnet network traffic sent over to C&C centers they initially managed to discover that from the 38 thousand infected workstations worldwide, the 22 thousand were stationed at Iran and 6.700 Indonesia. That made them suspect of a targeted cyberattack focused on the Islamic Republic. At that time the Iran was planning on operating a nuclear reactor at Bushehr, a source of great tension with Israel and the US for some years. But even more controversial was the uranium enrichment plant in Natanz: United Nations had voted for sanctions against Iran over the plant threatening for an air strike. After collecting more than 3.280 copies of Stuxnet from infected machines by various antivirus firms, and based on the data found in each malware's deployment log files, the Symantec's security researchers were able to discover that the initial targets were five companies in Iran territory although they never reveal their actual names. By further investigating all the aspects of the malware's source code the security researchers managed to identify the attack's main purpose: Sabotage centrifuges installed in the power facilities of Natanz in order to stop or delay the Iranian nuclear program. The attack was accomplished by reprogramming the PLCs that managed the centrifuges thus making them work out of their specified boundaries ending up disabling them. The malware escaped its initial targets and spread beyond Iran national boundaries. Until September 2009 the worm had escaped the boundaries of Iran and managed to infect 100.000 hosts approximately in 155 countries, US included [81].

In particular the vector of the attack can be described as follows:

1. **Reconnaissance phase:** Adversaries create malware infected USB drives which, then, place in strategically chosen sites (e.g., at the Iran's industrial sites' entrances) so as to allure industrial workers to plug them to their computers.
2. **Initial infection phase:** The worm is designed to infect Windows operating machines by taking advantage of auto-execution features in removable drives (Microsoft Windows Shortcut 'LNK/PIF' Files Automatic File Execution Vulnerability - Bugtraq ID 41732). Then, it takes advantage of two *zero-day* Windows vulnerabilities (MS10-073 and MS10-092) to perform privilege escalation. In order to avoid detection, it utilizes a rootkit to hide its binaries so as to evade antivirus products.
3. **Lateral movement and propagation phase:** Module *Export 22* was the main payload responsible for network communications and propagation. In particular:
    - Infects any newly inserted removable drives.
    - Utilizes peer-to-peer networks in order to connect to C&C servers.
    - Uses hardcoded credentials to infect WinCC devices [250].
    - Connects to all available default network shares.
    - Exploits a *zero-day* vulnerability (MS10-061) in Microsoft Windows print spooler service.

- Exploits MS08-067 Windows Server Service Vulnerability.

4. **Establish and maintain remote access:** Adversaries utilize an remote access and update mechanism via a peer-to-peer mechanism within a LAN for communicating to C&C centers, whereas during the final infection process the malware was designed to hide its code on PLCs using a specially crafted rootkit.
5. **Remote control and device manipulation:** The adversaries were able to remotely adjust the spinning rate of the network enabled centrifuges and, at the same time, falsify the information sent back to the operators. The latter enabled them to increase the spinning rate at a level where centrifuges started to fail without anyone noticing.

4.2.2 Dragonfly

A group of well-funded, highly-skilled adversaries launched a cyber-espionage campaign, the first advanced attack after Stuxnet that targeted ICS equipment [210]. The group behind the attack was named 'Dragonfly' by Symantec or 'Energetic Bear' by other security firms. Initially, the targeted systems were aviation and defense industries located in the US/Canada but afterwards the attacker's group showed interest for industries of the energy sector. Using the *watering hole* attack technique [52] the adversaries managed to infect with malware several company networks. Furthermore, they managed to inject malicious payloads on available ICS vendor software found on official websites. The attack was staged in three phases: Firstly, spear-phishing campaigns were launched and remote access was established via a RAT horse. Then, Havex software was used in *watering hole* attacks against official vendor websites thus redirecting users to servers with malware infected ICS software.

1. **Reconnaissance phase:** Retrieval of corporate information from aviation, defense and energy industries' Web presence.
2. **Initial infection phase:** Via spear-phishing techniques 'Dragonfly' group infected employees' workstations with HAVEX malware. Initially, the malware harvested data, such as emails, contact lists and documents.
3. **Establish and maintain remote access:** HAVEX malware served as a means for installation of other malware sent from Dragonfly servers (e.g., Karagany RAT, password stealer module, etc). It consisted of a remote access Trojan and a server module written in Hypertext Preprocessor (PHP). After installation, the malware communicated with C&C server in order to download and execute

other malicious payloads, such as an OPC scanning module, that utilized specific TCP ports used by Siemens and Rockwell automation systems, to retrieve information for ICS equipment.

4. **Propagation phase:** By exploiting vulnerabilities in the vendors' websites the Dragonfly group was able to place its payloads in three major ICS vendor websites. In the case of the first vendor's website (eWon) the adversaries managed to change a download link so as to point to a modified package of a VPN application (Talk2Me) that provided access to PLCs. The second compromised website belonged to a European manufacturer of PLC devices whereas the third website was owned by a company that manufactured ICS for energy sector, including wind turbines. None of the websites affected, enforced any authorization mechanisms for accessing the ICS software.

5. **Remote control and device manipulation:** Havex's main target was ICS communication interfaces and especially OPC information. In all three cases described the attackers successfully managed to inject malicious code into the vendor's driver package. Security researchers were able to identify 88 different versions of Havex, 146 C&C centers (mainly vulnerable blog websites) and 1,500 IP addresses of potential victims, most of which, in Europe.

Although Dragonfly attack did not disturb any industrial control process or lead to a severe energy outage, the adversaries manage to collect a large amount of valuable information that could potentially assist them in launching future attacks [210]. Furthermore, the OPC scanning module could be used to compromise ICS maintenance suppliers' services such as eWon, which utilizes approximately a million remote connections in order to provide remote support on ICS equipment. Based on later investigations, it was discovered that the Dragonfly group had also targeted the pharmaceutical industry aiming at stealing valuable information such as medicine recipes, batch production sequence steps as well as manufacturing plant volumes and capabilities.

4.2.3 Attacking industrial robots

3 million industrial robots operating in factories around the world – an increase of 10

According to the International Federation of Robotics report[5], 3 million industrial robots operate in factories around the world at the end of 2021 with almost 384,000 new units shipped globally in 2020, despite the global pandemic. Robots

[5] https://ifr.org/ifr-press-releases/news/robot-sales-rise-again

are used in almost every critical industrial sector such as automotive, aerospace, defense, plastics, electronics and electrical, metal fabrication, pharmaceutical, railway and many more. Several security firms and researchers have pointed out vulnerabilities in both domestic and industrial robots [47, 181]. The latter are usually of large volume used in complex manufacturing processes and play an essential role in production lines. Industrial robots are exceptionally complex CPS that include actuators, sensors, human-robot interfaces and are constantly connected to computer networks primarily for operation, programming and maintenance purposes. In [181] researchers mainly focused on industrial robots by analyzing protocols and relative software. They utilized an actual robot (ABB six-axis IRB140) in order to demonstrate a series of attacks such as alter or introduce minor defects in the manufactured products, physically damage the robot, steal industrial secrets and/or cause human injuries.The impact of a single software vulnerability could have serious consequences, since, it could enable an adversary to inflict a massive financial damage and/or even threat human lives. After the Industry 4.0 [158] was introduced, almost all new models of industrial robots tend to incorporate IoT technologies such as connectivity and operational features that expose them to a much broader attack surface.

Moreover, using well known search engines (Shodan, ZoomEye and Censys), they managed to discover multiple industrial robots' network interfaces connected directly to the Internet. As of late March of 2017, researchers discovered approximately 84,000 of industrial robots that were exposed to the Internet, 5,105 of which did not require any authentication, 59 had known vulnerabilities whereas the researchers were able to identify 6 totally new (*zero-day*) ones. These included the usage of a self-signed certificate for multiple devices, network service banners that disclosed sensitive information (vendor's name, MAC address, firmware version, CPU model, CPU frequency, etc), outdated software components (application and cryptography libraries, compilers, kernel), default credentials or no/poor authentication mechanisms, static VPN private keys on publicly available firmware images, adoption of symmetric cryptography schemes in VPNs, the use of plain HTTP Web interfaces with no/poor input sanitization, default 'as is' use of open software (e.g., REST layer in PHP) and publicly available unstripped firmware images. A realistic APT attack vector against industrial robots includes the following steps:

1. **Reconnaissance phase:** Adversaries use search engines to discover and enumerate Internet-exposed robot interfaces by searching for specific strings in the HTTP header (e.g., 'eWON', 'Westermo', etc). Then, they manage to locate several software vulnerabilities by reading freely available technical documentation, reverse engineering publicly available software (e.g., firmware files,

controller software) and even run exploitation tests using available simulation software (e.g., ABB's software suite).

2. **Initial infection phase:** Using the vulnerabilities found in previous phases adversaries establish a connection with the device (e.g., authentication bypass in ABB's eWON industrial cellular router, FTP static credentials to access the command driver, memory errors found in the RobAPI). Since no security mechanisms are present and the Internet interface is used, the attack will remain undetected from any IDS/IPS equipment installed in the internal network.
3. **Establish and maintain remote access:** Through FTP access, attackers upload custom, malicious software and trigger a reboot using the command shell reboot FTP function. The malicious files are executed and all robot features are now remotely controlled via a C&C center.
4. **Propagation phase:** Utilizing connectivity features installed in robot's main computer (e.g., FlexPendant, RobotStudio) attackers manage to discover and attack other robot network interfaces that are connected on the company's internal network.
5. **Remote control and device manipulation:** Adversaries are able to launch a series of attacks, which the researchers categorized into five classes, evaluating the potential impact of each one individually. The categorization was made under the assumption that a robot must be able to at least read accurately from its sensors and execute its control logic, perform precise movements, and not harm humans in any circumstance. In particular:

   (a) **Altering the Control-Loop Parameters:** This attack includes the modification of the configuration control closed/open loop parameters used to control robot movements. Implications of such attack can lead to safety boundary violation and even breakage of robot parts.

   (b) **Tampering with Calibration Parameters:** Repeatedly manipulation of the controller's calibration parameters at runtime could lead to DoS attacks.

   (c) **Tampering with the Production Logic:** In the case where the controller does not enforce *end-to-end* integrity checks a program task could be altered thus leading to the manufacturing of defective products or fully compromising a factory's manufacturing process.

   (d) **Altering the User-Perceived Robot State:** In this case the robot's user interface is manipulated in order to hide or misinform the operator of the true robot status so as to fool him/her into making wrong risk evaluations. This kind of attack can put operators at risk and even lead to human injuries.

(e) **Altering the Robot State:** Changing the robot's true state may have major impact especially when combined with other attacks (e.g., manufacture a large amount of defective products).

Realistic threat attack scenarios may include sabotage of an entire production line via product's characteristics alteration followed by a ransomware campaign in order to reveal which product batch was affected, physical damage to industrial equipment, human injuries and/or the use of the device as a means to exfiltrate sensitive industrial data (e.g., industrial secrets such as calibration parameters).

4.2.4 PLC ransomware: LogicLocker

In 2017, researchers of Georgia Institute of Technology [90] presented a hypothetical ransomware attack scenario in which, adversaries target network connected PLCs located in a water treatment plant. The targeted PLCs were used to control the valves which, in turn, control the amount of chlorine that is added into the water. In particular, they developed a framework named 'Logiclocker' that then used to attack some of the most popular PLCs in the market such as Schneider Modicon M221[6], Allen Bradley MicroLogix 1400[7], and Schneider Modicon M241[8]. The phases described in order to launch a successful ransomware campaign included initial infection, lateral movement within internal SCADA networks, reconnaissance and target discovery, locking and encrypting process and finally the negotiation for the ransom. In their PoC attack the researchers managed to retrieve the device's credential (in this case, Modicon M241) either by stealing or using brute force attack techniques. A typical ransomware attack scenario consists of the following phases:

1. **Reconnaissance phase:** Adversaries utilize search engines like Shodan to locate Internet facing PLCs.
2. **Initial infection phase:** Using stealing, brute force and dictionary attack techniques they manage to recover authentication information (e.g., user/system credentials) from the discovered Internet facing PLCs.

[6] https://www.se.com/ww/en/product-range/62128-modicon-m221/

[7] https://www.rockwellautomation.com/en-us/products/hardware/allen-bradley/programmable-controllers/micro-controllers/micrologix-family/micrologix-1400-controllers.html

[8] https://www.se.com/ww/en/product-range/62129-modicon-m241/

3. **Propagation phase:** Embedded payloads enable the malware to scan the internal SCADA networks of the water treatment plant in order to infect other vulnerable PLCs.
4. **Establish and maintain remote access:** Adversaries remotely reprogram the infected PLCs with new passwords thus locking the legitimate operators out.
5. **Remote control and device manipulation:** The attackers remotely encrypt the PLCs' software using well known encryption algorithms (e.g., AES) with a newly generated key.
6. **Ransomware phase:** Via the LogicLocker framework an email is sent to the water treatment plant that threatens to release chlorine in the water and cause massive human fatalities.

4.3 APT attacks on Smart Grid SCADA networks and field devices

Power grid systems deliver electricity from power suppliers via interconnected networks to customers. Smart grids rely on ICT such as SCADA systems to achieve the on-time delivery of the required amount of electricity to the end-users. Disruptions of the normal operation of such systems can lead to major economic losses, electrical blackouts or even human fatalities in a worst case scenario. This in turn, makes them an appealing target to high-profile adversaries who are able to utilize advanced exploitation techniques in order to achieve their goals. In order to further understand this rapidly evolving threat landscape, we are going to analyze recent, high-profile PoC and real cyberattacks on generation, distribution and transmission systems

4.3.1 Attacks on generation systems: The Aurora attack

In 2007, an attack scenario that targeted electric power generators, was demonstrated at the Idaho US National Labs [256, 302]. Network enabled PLCs (circuit breakers) were forced to open and close in a very fast rate (4 times per second) in order to force the affected power generator to desynchronize thus resulting in its physical destruction. In a potential attack scenario described in [302] an attacker compromises the company's corporate network to propagate to the facility's main control center and take advantage of an existing communication link that is used to remotely administer the PLCs.

In order to launch an Aurora-like attack, the attacker would have to overcome intentional delays in switching on and off and synchronization checks that exist to ensure the smooth operation of the system. Assuming that the attacker has compromised a sufficient number of devices, it is possible to inject falsified commands to trip and reclose a circuit breaker in a rapid repetitive way. In particular, an hypothetical Aurora attack scenario can be described as follows:

1. **Reconnaissance phase:** Adversaries manage to collect corporate information (e.g., email accounts) that then use to launch a spear-phishing campaigns
2. **Initial infection phase:** Using known and *zero-day* exploits they manage to elevate privileges and install a RAT tool in order to control the infected workstations remotely.
3. **Establish and maintain remote access:** Using network pivoting techniques they manage to navigate the facility's internal network and infect a workstation located in the control center. Moreover, using similar exploitation techniques they establish remote access to the workstation and through it to the target PLCs.
4. **Lateral movement and propagation phase:** Using Nmap[9] or similar tools they fingerprint the relays' brand name and model (Ethernet and/or Modbus). Then, via passive eavesdropping and vulnerability exploitation techniques (e.g., false data injection attacks [168]) they manage to remotely control the circuit breakers and bypass protection relays.
5. **Remote control and device manipulation:**
    - **Step 1:** The circuit breaker(s) are opened isolating the generator from the grid.
    - **Step 2:** The generator starts to speed up and the frequency of the generator increases.
    - **Step 3:** The frequency difference between the grid and the generator increases.
    - **Step 4:** After a particular amount of time the circuit breakers are closed, thus connecting back the generator to the grid.
    - **Step 5:** The generator is forced into synchronization with "*out-of-sync*" conditions thus causing substantial electrical and mechanical transients.

[9] https://nmap.org/

- **Step 6:** Steps 1 to 5 are repeated in a timely manner until the generator is permanently damaged.

Adversaries (e.g. terrorists, nation state) could launch concurrent attacks against multiple generators, in order to destabilize large areas of a country's Smart Grid, thus maximizing the potential impact of the attack.

4.3.2 Attack on the Ukraine's distribution network (2015)

One of highest impact, highly coordinated, stealthy APT attack against the Smart Grid is considered to be the one that took place on December 23, 2015 against an Ukraine regional electricity company named 'Kyivoblenergo' . The attack resulted in massive outages that affected approximately 225,000 customers for several hours [43], whereas substation control (e.g., circuit brakers) was switched to manual for weeks.

The adversaries utilized a variety of attack techniques including the use of spear-phishing campaigns (they impersonated an email message from the Ukrainian parliament), variants of *BlackEnergy* 3 and *KillDisk* malware as well as the manipulation of Microsoft Office documents in order to gain an initial foothold to the company's internal network. The attackers possessed specialized knowledge of ICS network connected devices such as Uninterruptible Power Supplies (UPSs), HMI interfaces, credential harvesting techniques, and SCADA client software. The attack vector can be described as follows:

1. **Reconnaissance phase:** Nation-state adversaries launched a spear-phishing campaign with malware-infected Microsoft Office documents against corporate users.
2. **Initial infection phase:** By exploiting Windows well known and *zero-day* vulnerabilities they managed to install key-loggers and retrieve user credentials.
3. **Lateral movement and propagation phase:** Initially, the adversaries performed a reconnaissance of internal SCADA network and devices. Then, pivoting throughout different network segments enabled them to locate and infect SCADA dispatch workstations and servers. In particular, they managed to gain access to operators' workstations, located in control rooms, that run HMI software.
4. **Establish and maintain remote access:** Using existing, legitimate remote administration tools, installed on operators' workstations, they managed to remotely connect to the aforementioned workstations and lock the legitimate operators out. In addition they uploaded malicious firmware in field communication devices to prevent any recovery attempts.

5. **Remote control and device manipulation:** In order to magnify the impact of the attack the adversaries proceeded with the following actions:

   (a) Remotely opened multiple circuit breakers to cause massive outages. (attack's main target)

   (b) Reconfigured uninterruptible power supply systems to cause outages in company's buildings.

   (c) Launched a remote telephonic denial of service on the energy company's call center to frustrate the impacted customers.

   (d) Utilized a modified version of *KillDisk* malware to destroy forensic evidence and render workstations inoperable.

4.3.3 Attack on the Ukraine's Kiev transmission station (2016)

In December 2016, the Ukrainian's Smart Grid SCADA systems were targeted for a second year in a row [105]. The target of the attack was a 200 Megawatt transmission station located near the city of Kiev. Similar to the previous attack, the adversaries launched spear phishing campaigns in which they wrapped in a word document attachment the malware *CrashOverride/Win32/Industroyer* [159] in order to infect the employees' workstations. This time the attack techniques used were far more sophisticated and stealthier than the first attack. The malicious code was capable of being preprogrammed to launch an attack against multiple targets, at a future time, without any intervention from the attackers. The malware was modular and included, among others, the main program that ensured communications with C&C centers and IIoT equipment, four different malicious payloads that correspond to industrial control protocols IEC 101, IEC 104, IEC 61850, OPC Data Access (OPC DA) and a DoS tool that targeted a particular family of protection relays (Siemens SIPROTEC). Figure 4.1 depicts the basic functionality of the malware. A more detailed description of the software components as well as a walkthrough of the attack [53] is presented here:

1. **Reconnaissance phase:** Using publicly available information found on the Internet (e.g., YouTube) the adversaries were able to enumerate substation's ICS. Having selected their target, then they launched a spear-phishing campaign (July 2016) against corporate users.
2. **Initial infection phase:** Using advanced exploitation techniques they managed to gain a foothold to the substation's internal network. In particular, after

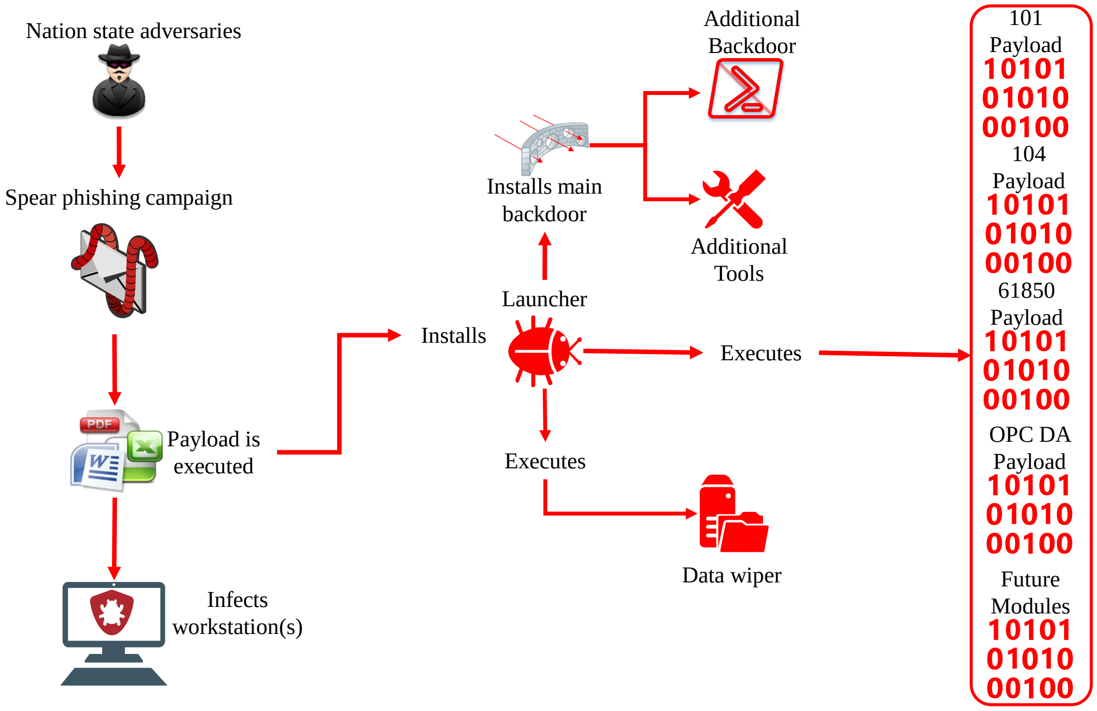


Figure 4.1: Attack vector of the malware (CrashOverride - Win32 - Industroyer) on Ukraine's Smart Grid (December 2016) [259]

they managed to infect corporate workstations and/or servers, the malware installed the main backdoor program responsible also to control all other SCADA modules. The latter could be programmed to communicate with the attackers at a specific time every day via C&C servers (active TOR nodes). Initially, it authenticated with a local proxy (TCP port 3128) and then utilized an HTTPS channel to connect to external C&C servers. After a successful privilege escalation process, the backdoor was masqueraded as a legitimate windows service program to avoid any detection.

3. **Lateral movement and propagation phase:** The adversaries incorporated highly customized, sophisticated SCADA communication modules in order to interact with IIoT equipment. The purpose of the SCADA communication modules was twofold. Initially, they were used in the enumeration and propagation phase, in which specific commands were issued to fingerprint IIoT devices, and as a means of launching the main attack by issuing the necessary control commands.

   - **IEC 60870-5-101 module:** It utilized the file 101.DLL to implement the IEC 101 protocol so as to communicate with compatible RTUs. Upon execution, the payload located and terminated the legitimate process used to communicate with IEC 101 devices. Then, a new process was started in order to take over the control of the RTUs.
   - **IEC 60870-5-104 module:** Since IEC 104 extends the IEC 101, the module utilized TCP/IP network as its main communication channel. It also supported a configuration file for customization and operated in a similar way as the IEC 101 payload.
   - **IEC 61850 module:** Unlike the previous modules this one consisted of both an executable file (61850.exe) as well as a DLL file. When executed, the malicious program enumerated all IP addresses and tried to connect to TCP port 102. Then, Manufacturing Message Specification (MMS) commands were used to enumerate and control all discovered devices, such as circuit breakers.
   - **OPC DA module:** OLE, Component Object Model (COM) and Distributed Component Object Mode (DCOM) are Microsoft technologies that are used for real-time data exchange, based on a client-server model. Similar to IEC 61850 payload, the malicious program consisted of a .EXE and a .DLL file that, incorporated both 61850 and OPC DA functionalities. Upon execution, enumeration of all OPC servers and devices was performed (ABB solutions). Then, the OPC's state was altered using the `IOPCSyncIO` interface.
   - **Port scanner and DoS tools:** Additionally, a custom-made port scanning program and a DoS tool were included in the malware. The latter could be used against SIPROTEC Siemens devices by utilizing a known vulnerability (CVE-2015-5374).

4. **Establish and maintain remote access:** Aside the main backdoor, the attackers utilized a trojanized version of the Windows notepad application, to serve as a back-up persistence mechanism, in order to regain access in the case of the main backdoor was found and disabled. To avoid detection, the embedded malicious code was heavily obfuscated and utilized different C&C servers than the one used from the main backdoor program.
5. **Remote control and device manipulation:** To launch the attack, the adversaries utilized the 'Launch' module in which they had embedded specific time and dates (17th and 20th December). Once one of the dates was reached the module was programmed to execute two processes in high priority. In particular:

- **Payload.DLL:** The actual name of the DLL file that contained the main payload was not hardcoded into the module but had to be supplied from the adversaries along with a configuration file. Upon execution, the payload used the functionality embedded in aforementioned modules to issue commands to located RTUs and PLCs, such as turn the device off or change their status (e.g., open/closed).
- **Data wiper module:** This payload was scheduled to launch with a delay of 1 - 2 hours from the first payload. It included the file `haslo.exe/dat` that when executed, it modified the registry value `ImagePath` with an empty string thus rendering the system unusable. In addition, it deleted specific files by overwriting them twice and terminated all running process in order to make the system crash. The list with the file extensions for deletion included, among others, Windows binaries as well as MS SQL server and ICS configuration (ABB PCM600) files.

SECTION III
Proposed Methodologies

CHAPTER 5

A HIGH-LEVEL, RA METHODOLOGY FOR IOT-ENABLED ATTACKS

5.1 Modeling IoT-enabled Cyber Attacks

Although IoT technologies favor the interoperability and remote management of various CPS, including CIs [6], at the same time, they increase the exposure of those systems to cyber attacks. The inter-connectivity capabilities of IoT technologies, along with their inherent computational constraints [122], are unfortunately sufficient conditions that enable various attack vectors against critical systems and services.

An attack vector describes the necessary steps that an attacker will undertake in order to realize a threat. In order to model IoT-enabled attack vectors against critical systems and services, we are going to examine the main entities involved in such attacks, as well as the interaction among them. From a high level view, the interaction between these entities will capture all possible IoT-enabled attacks. Figure 5.1 describes this model.

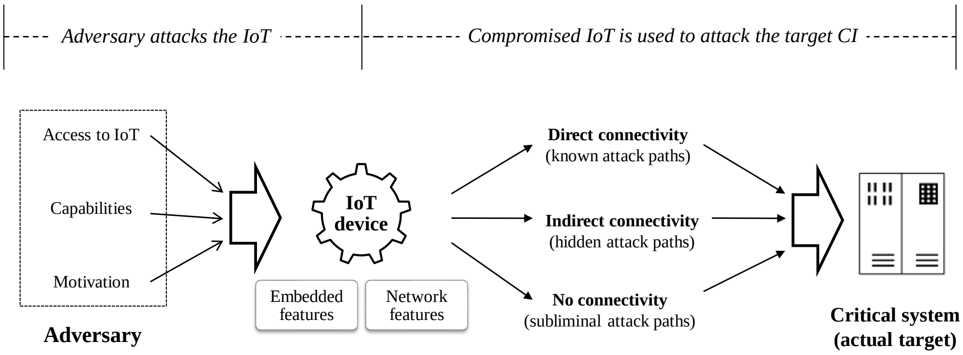


Figure 5.1: A high-level description of IoT-enabled critical attack vectors [261]

- *The adversary:* It represents the actor of the attack. It is the entity whose actual goal is to cause damage to a target system. Attacks that can be realized by "powerful" adversaries are usually less possible to happen and vise versa. We model the characteristics of potential adversaries based on their access level, capabilities and motivation.
- *The IoT device:* In our model, the IoT device is the enabler (or in some cases the amplifier) of the attack. Being in most cases the weakest link in the security

chain, it will usually be used by the adversary as an initial entry point, to gain access to critical services. This can be accomplished by exploiting inherent vulnerabilities, such as lack of embedded security mechanisms or network layer vulnerabilities.

- *The actual target:* Mission-critical systems of high importance are usually the actual targets of real-life, cyber attacks. An adversary with sufficient capabilities and motivation will attempt to abuse existing paths between the vulnerable IoT and a critical system. IoT devices, that are directly connected with a critical infrastructure, create obvious attack paths which in turn attract potential adversaries. Since the target is a system of high significance for the well being of the citizens, if it gets compromised then the consequences for its users will be of high impact. Unfortunately, vulnerable IoT devices may also be connected in less obvious, indirect and hidden ways with critical systems. For example, infotainment systems in smart vehicles may be indirectly connected with mission critical systems of the vehicle [192, 193]. Passive medical IoT devices such as smart clinical beds may be indirectly connected to in-hospital critical systems [131]. In some cases, even the physical proximity of vulnerable IoT devices with a critical system suffices to create such a hidden attack path. For example, [232] describes how vulnerable smart lamps may be used to exfiltrate sensitive data from systems that reside in highly secured premises. Furthermore, it is possible to use IoT devices that are not connected to any critical system, in order to amplify an attack and cause serious damage to critical services, therefore creating subliminal attack paths [212].

5.1.1 Characteristics of the adversary

As shown in Figure 5.1, the adversaries of IoT-enabled attacks can be modeled using three main characteristics: Their access to the IoT device, their capabilities and their motivation. As shown in Figure 5.1, the adversaries of IoT-enabled attacks can be modeled using three main characteristics: Their access to the IoT device, their capabilities and their motivation.

Required access to the IoT

This characteristic examines what type (physical and/or logical) and level of access to the IoT device is required, in order to trigger the attack. In some cases remote logical access is sufficient, while other attacks may require to physically tamper the target device.

- *Physical access:* We distinguish two access levels. An insider is an adversary that has direct physical access to the target IoT device. Since in IoT communication protocols physical proximity with a device may be sufficient to launch an attack, we will consider an adversary with physical proximity to the IoT device as an insider. An outsider has no direct physical access or proximity to the target IoT device, but may try to gain knowledge by tampering another IoT device of the same type (e.g. extract a common pre-shared key from one device to attack the actual target device). In general, if an attack can be realized only by insiders, it is less likely to happen than an attack that could also be triggered by outsiders.
- *Logical access:* Again we distinguish two access levels. Privileged access adversaries are allowed to logically connect to the IoT device through an available interface. Unprivileged adversaries does not have a priori logical access to the target device. In general, attacks that require privileged logical access to the IoT device are less likely to happen, since the adversary will have to bypass authorization controls, e.g. through privilege escalation. On the other hand, attacks that do not require privileged access are more likely to happen, e.g. inject commands to a device without prior authorization.

Required capabilities

This characteristic models the skills and resources required by an adversary to successfully attack the target system.

- *Technical Skills.* Attacks that can only be implemented by technical *experts* are less likely to happen, in comparison with attacks that can be triggered by *novice* adversaries. In the middle, some attacks may require *moderate* technical skills.
- *Recourses.* Similarly, attacks that can be implemented only by adversaries with *high resources* such as very expensive, specialized or hard to find equipment, are less likely to happen, in contrast to attacks that require, for example, cheap off-the-shelf equipment only.

Required motivation

Motivation may be seen as an alternative way to describe the potential gain that an adversary would benefit from a successful attack, in combination with the expected penalty for an adversary being traced. Espionage, financial profit, cyber-terrorism and hacktivism are some of the main categories of adversary's incentives. For example, an on-line banking system may be seen as a potential target for financially-motivated adversaries. On the other hand, in the case of a cyber-terrorist or black hacker, a water treatment facility may look a much more attractive target. Attacks that can attract adversaries having even a *weak motivation* are more likely to happen. In contrast, attacks that would be triggered only by *strongly motivated* adversaries, e.g. ones that may risk being traced in favor of a high expected gain, are less possible.

5.1.2 Vulnerabilities of the IoT device

Since the IoT device is the enabler/amplifier of the attack, an adversary shall discover and exploit existing vulnerabilities associated with one or more layers of the IoT device in order to succeed. We categorize IoT vulnerabilities in two main categories: Embedded vulnerabilities and network vulnerabilities.

Embedded system vulnerabilities

This category involves design and implementation flaws at the IoT HW and the SW layers.

HW layer: Due to their cost and resource constraints, IoT devices may suffer from various HW vulnerabilities.

- *Lack of tamper resistance*: Most IoT devices do not implement HW security controls that may prevent/detect physical tampering attacks, e.g. key extraction attacks.
- *Weak embedded crypto algorithms*: IoT devices may come with embedded implementations of weak encryption algorithms, e.g. algorithms of small key size [27].
- *Weak hardware implementations*: Untested HW implementations may leak sensitive information, such as stored keys used to authenticate the firmware of the device, e.g. through *Side-Channel* attacks (Differential/Correlation Power Analysis – CPA/DPA) [233].

SW layer This includes vulnerabilities, bugs and flaws that can be introduced during the design, implementation and testing of the software developed for the *Firmware* (FW), OS or the application layer of IoT devices.

- *FW layer*: If the firmware is not integrity protected, then an adversary who has gained access to the full FW image (e.g. due to hardware vulnerabilities) may *modify* and re-install it in the device, or may *reverse engineer* it [233] to recover the stored credentials.

- *OS*: Various OS vulnerabilities may allow the adversary to gain unauthorized access, e.g. through *privilege escalation.* A secure architecture should enforce the principle of the *least privilege*, which dictates that only the minimal access required to perform a function should be authorized, in order to minimize the effectiveness of any breach of security.
- *Application layer*: Due to the costs involved, in many cases IoT applications are not audited (penetration tested) prior to their deployment. The API of any IoT application-layer SW should be tested for potential flows that may allow unauthorized *execution*, *injection* or *manipulation* of commands. Techniques such as *input filtering*, command *integrity checks* and other controls applied in secure software development should be applied.

Network vulnerabilities

This category examines vulnerabilities in the network protocols and the supporting mechanisms of IoT communications.

Communication protocols Remote adversaries commonly scan for network-layer vulnerabilities, in order to exploit an IoT device.

- *Link- and network-layer protocol vulnerabilities.* Wireless network protocol families and the relative protocol implementations used in IoT communications, such as IEEE 802.15.4x (e.g. ZigBee, WirelessHART, MiWi) and IEEE 802.11.x (e.g. WiFi) incorporate several security flaws that will be further analyzed in the next sections. Such errors may enable an adversary to *inject*, *modify* or *read* exchanged messages. For example, if the encryption scheme at the network layer does not ensure *semantic security* an adversary may recover encrypted data that are transmitted through the network [49].
- *Application-layer protocol vulnerabilities.* Misconfiguration and implementation flaws in application layer protocols (e.g. CoAP) may have a major impact, especially if the IoT device is a part of, or is connected in some way, with a critical system [232].
- *Network design flaws.* Although these cannot be considered as vulnerabilities of the IoT device only, in many cases the specifications of the IoT device allow such miss-configurations. For example, if IoT devices that do not support any network-layer security are installed, they are completely exposed to network attacks [191]. Another case is IoT devices that are installed in networks with poor or no network segmentation.

Key Management: Proper key management mechanisms are required to enable strong cryptographic mechanisms for data confidentiality, integrity and entity authentication.

- *No support of public key exchange*: Due to hardware, energy and application constraints, strong key management schemes, such as those based on public keys, are difficult to implement in IoT devices.
- *Easily extractable communication keys*: The constraints of many IoT devices may lead to easily exploitable key management schemes, e.g. keys that can be easily retrieved or extracted [76].
- *Use of common (or no) key*: In many cases, key management relies on a common key embedded to all the devices of the same model [233]. An adversary who succeeds to compromise the key from one device, can use it to attack all the devices. In other cases, the use of encryption keys may be optional or not available at all.

5.1.3 Connectivity between the IoT and the target device

By embodying networking capabilities, the IoT devices are able to interconnect with other systems, in ways that cannot be easily perceptible. Protocols like the 6LowPAN, allow IoT devices to directly connect to the Internet thus enabling the remote management of other control systems. An adversary may abuse these connectivity paths to attack CIs and systems.

Direct connectivity with a critical system: In this case, the IoT device is physically and/or logically connected with a critical system. In general, IoT devices that are directly connected with critical systems create attack vectors that are easy to identify and therefore to assess their potential impact.

- *Direct physical connection*: A physical connection usually implies that the IoT device is installed inside a secured physical perimeter; for example a system actuator installed inside the CI premises [90, 253].
- *Direct logical connection*: A logical connection may refer to IoT devices that are either inside or outside the CI premises (e.g. temperature sensor).

Indirect connectivity with a critical system: IoT devices that are connected with a critical system in an indirect and non-obvious way, have been used to attack the system. Such attacks usually exploit the short-range communication protocols of the IoT devices. They can be very dangerous, mostly because they are overlooked and therefore underestimated; if such indirect connections are not identified, then a threat with a potentially high impact will be neglected.

This situation may be aggravated in the future since contemporary working environments apply policies such as *Bring Your Own Device* (BYOD) or *Bring Your Own Phone* (BYOP) which allow untested end-user IoT devices to gain physical proximity and potentially an indirect logical connection with critical systems, thus creating new attack vectors.

- *Physical proximity*: An auxiliary and usually low-importance IoT device that resides near a critical system, may be used to create a hidden attack path. For example a smart lighting system installed in a highly secure facility, or an employee's wireless body area network.
- *Indirect logical connectivity*: IoT devices may be connected to an auxiliary system that is logically connected to a critical system; e.g. the car's infotainment system that may be indirectly connected with critical car control systems through a shared communication bus.

No connectivity with a critical infrastructure: Smart IoT devices that are not connected, even indirectly, with critical systems have also been used to attack critical systems and services. Again, physical proximity may trigger attacks against nearby critical systems. In other cases, the key issue is the quantity of vulnerable IoT devices that are Internet-connected and therefore available to cyber attackers.

- *IoT used as an amplifier*: An adversary can exploit built-in vulnerabilities in a plethora of end-user IoT devices to control them and create a botnet, to ultimately attack a critical system. In recent real attacks, large numbers of low-cost and insecure consumer IoT devices were exploited and used to launch DDoS attacks against critical services [58, 111].
- *IoT as the target (concurrent attacks)*: The attack is actually targeting against a large number of end-user IoT devices. Although such devices are not actually part of a critical service, the massiveness of the attack may lead to very important consequences. A possible attack scenario may include a versatile attacker who is able to remotely infect thousands of smart TVs with a ransomware [86]. The attacker may then cause significant financial losses to the end users and reputation loss to the device manufacturers.

5.2 Assessing IoT-enabled Cyber Attacks

In order to assess IoT-enabled cyber attacks in terms of their severity, we will define a generic risk based methodology. The methodology will utilize the attack model defined in Section 5.1 and the related criteria, as described in this Section.

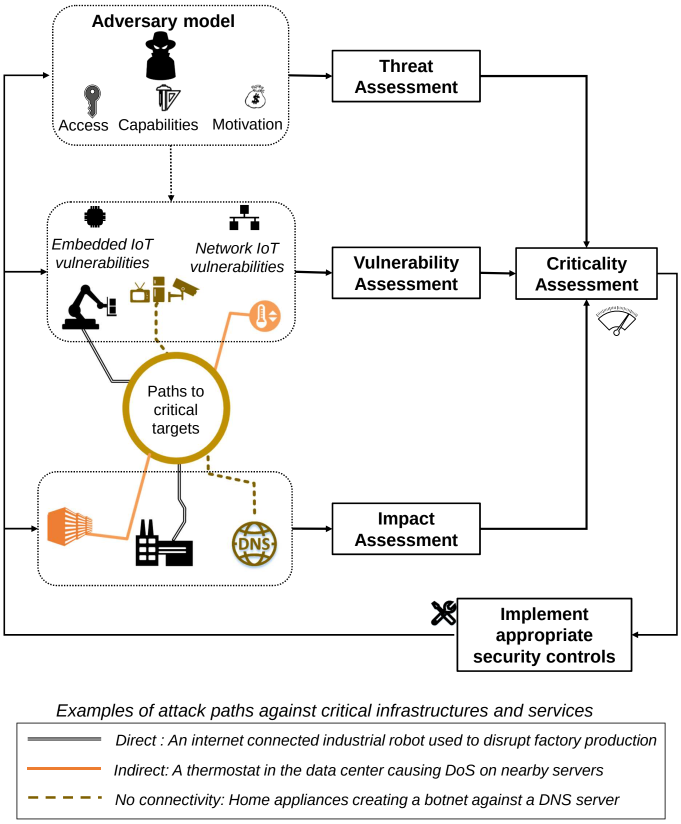
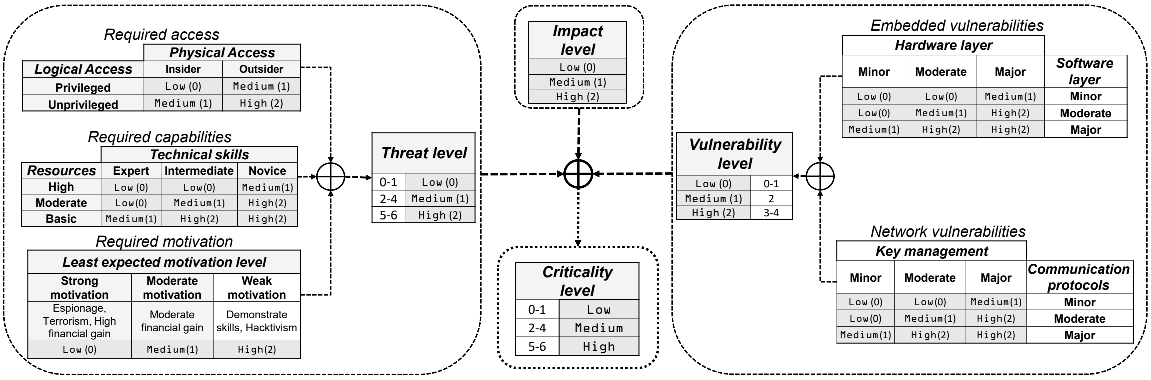

Adversary model
Access
Capabilities
Motivation
Threat Assessment
Embedded IoT vulnerabilities
Network IoT vulnerabilities
Vulnerability Assessment
Criticality Assessment
Paths to critical targets
DNS
Impact Assessment
Implement appropriate security controls
Examples of attack paths against critical infrastructures and services
Direct : An internet connected industrial robot used to disrupt factory production
Indirect: A thermostat in the data center causing DoS on nearby servers
No connectivity: Home appliances creating a botnet against a DNS server

Figure 5.2: A high-level view of the methodology. It represents the attack (threat) model, IoT vulnerabilities, impact and critically of IoT-enabled cyber attacks [261]

5.2.1 Risk-based approach: A high level description

Although various security standards [38, 134, 235] provide slightly varying definitions, in general a security attack can be assessed based on the security *risk* that it may cause to a target system. In turn, the security risk is a metric of the following risk factors: (i) The *threat level*, measuring the extent to which a system is threatened



Required access

Logical Access	Physical Access Insider	 Outsider
Privileged	Low (0)	Medium (1)
Unprivileged	Medium (1)	High (2)

Required capabilities

Resources	Technical skills Expert	 Intermediate	 Novice
High	Low (0)	Low(0)	Medium(1)
Moderate	Low(0)	Medium(1)	High(2)
Basic	Medium(1)	High(2)	High(2)

Required motivation

Least expected motivation level		
Strong motivation	Moderate motivation	Weak motivation
Espionage, Terrorism, High financial gain	Moderate financial gain	Demonstrate skills, Hacktivism
Low (0)	Medium(1)	High(2)

Threat level	
0-1	Low (0)
2-4	Medium (1)
5-6	High (2)

Impact level
Low (0)
Medium (1)
High (2)

Criticality level	
0-1	Low
2-4	Medium
5-6	High

Vulnerability level	
Low (0)	0-1
Medium (1)	2
High (2)	3-4

Embedded vulnerabilities

Hardware layer Minor	 Moderate	 Major	Software layer
Low (0)	Low(0)	Medium(1)	Minor
Low(0)	Medium(1)	High(2)	Moderate
Medium(1)	High(2)	High(2)	Major

Network vulnerabilities

Key management Minor	 Moderate	 Major	Communication protocols
Low (0)	Low(0)	Medium(1)	Minor
Low(0)	Medium(1)	High(2)	Moderate
Medium(1)	High(2)	High(2)	Major

Figure 5.3: A qualitative criticality level of IoT-enabled cyber attacks based on the characteristics of the attack model of Section 5.1 [261]

by the attack. (ii) The *vulnerability level*,[1] which measures the weaknesses that may be exploited by an adversary in order to realize the attack, and (iii) the *impact level*, which represents the potential damage that would be caused by the attack.

To be consistent with well established risk assessment standards (such as the ISO 27005 [134] and the NIST SP800-30 [235]), we define a a risk-based methodology to assess the criticality of IoT-enabled cyber attacks, based on these risk factors (see Figure 5.2). In order to methodologically assess the risk factors, we will utilize the criteria defined in the attack model defined above. In particular, the adversarial model (see also Section 5.1.1) will be used to assess threat level of an attack, while the IoT vulnerability criteria (see also Section 5.1.2) will be used for the vulnerability assessment.

As for the impact factor, when assessing the impact of an attack we will consider realistic scenarios that may cover all the connectivity attack paths described in Section 5.1.3.

5.2.2 Methodology limitations and expected outcome

We stress out that this risk-like categorization of IoT-enabled attacks does not substitute the need for an actual risk assessment on any real system. As information risk assessment standards (e.g. in [134]) suggest, risk evaluations cannot be generalized from one system to another, since risk factors depend on the specific characteristics (services, people, HW, SW and data) of a system under examination. In addition, we

[1] In various security standards the threat and the vulnerability levels are combined in some way to define the *likelihood* of an attack.

do not claim that the examined criteria are complete. For example, since the assessment of a threat is generic, it does not capture system-specific factors related with the threat level, such as the countermeasures that may be already installed. Similarly, vulnerabilities that are non-technical and organization-specific are not captured.

However, our goal here is not to assess particular systems, but to provide a useful insight about the *risk profile* of various IoT systems and services. For a critical system operator, it is very important to understand the risk profile of its IoT systems, even if these are not directly connected to the critical system. Although it is possible that the same cyber attack would exhibit a different risk level in a different system, it is still worthy to identify which IoT-enabled cyber attack vectors are in general more easy to implement against critical systems, which IoT devices have (or can) actually been exploited and in what ways and how severe the potential impact could be.

5.2.3 Defining scales for the risk factors

For each risk factor (threat, vulnerability and impact level) and eventually for their combined outcome, the criticality level, we will use a three level qualitative scale [`Low`, `Medium`, `High`], where each level is also assigned to an arithmetic value in the range [`0`, `1`, `2`]. These arithmetic values are used in order to quantify the various criteria utilized in each risk factor and eventually calculate the criticality level of the examined attack. The meaning of each level is different for each risk factor, as described in Table 5.1. The use of a three level scale is deliberately chosen for simplicity and is compliant with risk assessment standards like [134, 207, 235]. Although more fine-grained scales can be defined (e.g. for multi-layer analysis [273]), our goal is to demonstrate generic risk profiles for IoT-enabled cyber attacks and not to assess specific systems, thus a simplified scale suffices for this goal and is compatible with risk assessment standards. Similar scales have been used in related works, such as the impact assessment of attacks on Smart Grids [148]. Figure 5.3 demonstrates the evaluation of the criticality level of IoT-enabled cyber attacks, based on our risk-based methodology, which is further described bellow.

5.2.4 Threat assessment

When examining the threat level for an attack, the assessor must examine the likelihood for an attack to happen. The adversarial model defined in Section 5.1.1 is used for this purpose, since the probability of realizing an attack depends on the existence of capable and motivated adversaries, with sufficient access [235].

Values	Threat scale	Vulnerability scale	Impact scale	Criticality
`Low` (0)	Attack requires adversaries having full access to IoT, advanced capabilities and motivation	The involved IoT devices have (at most) minor embedded (HW, SW) and network-layer vulnerabilities	Attack may cause limited damages for any possible attack path	Attacks of low importance and priority
`Medium` (1)	Attack requires adversaries having some access to IoT, moderate capabilities and motivation	The involved IoT devices have moderate embedded (HW, SW) or network-layer vulnerabilities that are exploitable	Attack may exploit known, hidden or subliminal paths to cause at most moderate damages	Attacks that should be considered with medium priority
`High` (2)	Attack may be realized by adversaries with no access to IoT, low capabilities and motivation	Highly exploitable HW, SW and network-layer vulnerabilities	Attack may exploit known, hidden or subliminal paths to cause severe damages to a critical system	Highly important attacks that require immediate mitigation

Table 5.1: Summary of the risk factors and their corresponding scales [261]

Figure 5.3 demonstrates how these characteristics are combined to output the threat level, using a simple "*addition-and-reduction*" rule (see the left part of the figure). According to the logical and physical access required to realize the attack, the required access is assigned to one value in the scale [`Low`(0), `Medium`(1), `High`(2)]. Then, the technical skills and other resources required to launch an attack are combined in a similar manner to output a value in the same scale. Finally, the motivation that is expected by an adversary to initiate the attack is also assessed. If an attack is expected to be triggered only by an attacker with a strong motivation, then this attack is less likely to happen, in comparison with an attack that is likely to be triggered by a weakly motivated adversary. Then, a simple addition operation is used on the above partial results, leading to a a threat level in the range [0-6]. This arithmetic value is then reduced (mapped) in the [`Low`(0), `Medium`(1), `High`(2)] scale, as shown in the figure, to output the threat level.

5.2.5 Vulnerability assessment

Since we focus on IoT-enabled attacks, the IoT device is the most vulnerable entry point for an attack. The vulnerability level will be assessed based on the various technical vulnerabilities at all the layers of the IoT device. For each layer, the vulnerabilities described in Section 5.1.2 are examined, in order to determine if the vulnerabilities in the particular layer can be considered as very important (*Major*), *Moderate* or low priority vulnerabilities (*Minor*) (see the right part in Figure 5.3).

When characterizing the vulnerability in each layer, the general rule is to examine if known vulnerabilities have been identified in this layer and if these are easily exploitable. For example, if a device has no tamper resistance and is susceptible to side channel attacks, then it is can be considered as a device with major HW layer vulnerabilities. The identified vulnerabilities from all the layers are combined, to output the vulnerability level in the [`Low`(0), `Medium`(1), `High`(2)] scale, based again on the simple "addition-and-reduction" rule.

5.2.6 Impact assessment

Since the impact level of an attack highly depends on the specific characteristics and services of the target system, it is not easy to define a general impact level for an attack. In order to assess the potential impact for each examined threat we will use input from the real security incidents that we will examine. In addition, when examining the impact of an attack that has been verified as a PoC attack we will consider realistic scenarios not only for obvious and known attack paths, but also for IoT-enabled attack paths that may be hidden or subliminal, as discussed in Section 5.1.3. Again, the impact scale defined in Table 5.1 will be used. As it is the usual practice in risk assessment we will follow a *worst-case scenario approach* when assessing the potential impact.

5.2.7 Criticality assessment

The final step is to combine all the partial risk factors as defined above (Sections 5.2.4 to 5.2.6) to output the overall criticality level of an examined IoT-enabled cyber attack (see Figure 5.3). The three level criticality scale defined in Table 5.1 will again be used to categorize an attack as one of `High` importance that requires immediate mitigation, as a `Medium` importance attack that requires mitigation in a lower priority, or as a `Low` importance attack. In Chapter 7, we apply our high-level framework on both recent, real cyberattacks as well as PoC attack scenarios against CIs and services.

5.3 A high-level RA algorithm for cyber-physical attack paths

Based on the threat model described above, we can describe a high-level risk assessment algorithm, whose goal is to identify and assess *hidden risks* in CPS that stem from the IoT interaction. Following the RA standards, the calculation of the risks will be based on three basic phases: Threat, vulnerability and impact assessment. Finally, the security risk of each identified attack path can be assessed. In the proposed high-level algorithm, one can use of typical Likert scales, to define the threat, vulnerability, impact and risk scales. This is common to most general purpose RA methodologies (e.g. in [235]), although each methodology may define a different scales for each risk factor. During this process all the steps performed in the previous paragraphs will be combined to methodologically output all the related risks. Also, it is crucial, when defining the IoT devices/technologies, to take into consideration mobile devices such as smartphones and laptops (related to BYOD), since, they are equipped with multiple inputs, outputs and wireless network interfaces, can be directly/indirectly (via the corporate network) connected to the Internet, and sometimes are in proximity of mission critical systems.The basic steps of this process are as follows:

1. Identify all IoT devices and enabling technologies.
2. Repeat for each of IoT device (say device i):
   (a) **Access paths:** Identify all applicable access paths (physical, proximity, remote) to the IoT device:
      i. If the device can be physically accessed define all of embedded device's input, output and wired network interfaces (e.g. USB, Ethernet and Serial ports, sensors, speakers)
      ii. Proximity access paths: Define all of input, output and wireless network interfaces characteristics (frequency and active range of ZigBee, Bluetooth, WiFi, Z-Wave, microphone range and sensitivity, etc.).
      iii. Remote access paths: Define all enabled layer-3 network interfaces. Then for each network interface define all possible access paths (directly, indirectly), especially the ones that lead to Internet connectivity (e.g. Ethernet → Control Room → Corporate network → Web server).
   (b) **Attack paths:** Identify all possible attack paths against any affected CPS:
      i. *Direct connectivity attack paths:* Identify all direct attack paths between the IoT device and any other system.
      ii. *Indirect connectivity attack paths:* Identify all systems that are indirectly connected to the IoT using any network interfaces (wired or wireless).
      iii. Identify attack paths against any affected CPS, related with IoT *extended/misused functionality*:
         A. *Physical proximity:* Identify systems that are in physical proximity in respect with the IoT device's wireless network interfaces (e.g.

protocols that use the same bandwidth, devices that are in line of sight etc. [213]).

B. *Potential covert channels:* Examine devices for possible ways to create hidden covert channels (e.g. smart lamp systems have used as a covert exfiltration channels [232]; smart TVs/cameras for espionage [286]).

C. *Other potential misuse:* Examine devices for any other possible misuse against other CPS. Examples of such misuse include abusing smart lamp systems installed in hospitals to cause epileptic seizures [232]; alter the functionality of IoT-enabled industrial robots to affect the production line [48, 180]; manipulate the functionality of thermostats to disrupt the operations of the data center [124]).

(c) **Calculate risk:** For each identified attack path k (with target system j):

i. For each corresponding access path (attack vector):

A. Assess the threat level of the relative attack vector, denoted as T_{ijk}.

B. Assess the vulnerability of the IoT device, for the examined attack, denoted as V_{ijk}.

C. Assess the impact of the *actual target system* of the attack path, denoted as I_{ijk}.

D. Assess the risk of each examined attack path k that is triggered by IoT device i against the target system j as follows:

$$R_{ijk} = T_{ijk}\, V_{ijk}\, I_{ijk} \tag{5.1}$$

As a final step we propose the construction of a table with the calculated risk values of all IoT devices/enabling technologies in respect of the affected systems for all applicable paths. Metrics such as total risk ($R_{i_{total}}$) and Maximum Risk ($R_{i_{maxj}}$) per affected system, can be used in order to assess the criticality of each IoT device/technology i, and help prioritize the implementation of the appropriate security controls, so as to effectively reduce the organization's risk levels under a desirable threshold.

CHAPTER 6
A LOW-LEVEL, RA METHODOLOGY FOR COMPLEX, CYBER-PHYSICAL ATTACK PATHS

In this chapter, we utilize the high-level approach presented in Chapter 5 in order to delve deeper into identifying, modelling and assessing complex, IoT-enabled, attack paths. In order to discover meaningful/profitable attack path scenarios for adversaries, we follow an attack tree approach that is target oriented and source driven: each critical system is considered as a potential target node for adversaries. Then, based on the identified cyber and physical interactions, a recursive algorithm is used to construct all the potential cyber-physical attack paths towards the target node. The exploitability of those attack paths is assessed for various adversarial scenarios with respect to: (i) The exposure of the initial (source) node of each attack path against different adversaries, and (ii) the cumulative vulnerability of all the interacting nodes within each attack path. This allows us to filter out those interactions that are not "mature enough" to be exploited by adversaries and thus to reduce the number of the generated assessed attack paths by focusing only on attack paths that are more likely to be exploited based on their current exposure status. Ultimately, by using a properly modified risk formula the risk of the identified attack paths against realistic threat agents is calculated, thus providing 'ready-to-use' information for applying cost-efficient mitigation controls. By developing a proof-of-concept implementation and by testing a realistic scenario, we validate the proposed methodology and we demonstrate that it can effectively discover hidden and underestimated complex cyber-physical attack paths of high impact and risk.

6.1 Risk calculation formulas

According to RA standards, risk calculation can be defined based on five different risk class types, as defined in [113, 301], which rely on threat, vulnerability and impact factors. In our methodology, risk calculation properly combines 'Type 1' with 'Type 4' risk classes as follows. In 'Type 1' methods [113, 301] risk is analysed in relation to a threat and an asset, (or a group of similar assets). The calculation combines the likelihood of a threat, the 'combined' vulnerability of the asset(s) involved, and the impact of the threat in the (group of) asset(s), as shown in Equation (6.1):

$$Risk(Threat, Asset) = Likelihood(Threat) \otimes Vuln(Threat, Asset) \otimes Impact(Threat, Asset) \quad (6.1)$$

The operator $\otimes$ denotes a combination between the risk factors (this can be implemented through a discrete risk matrix). In 'Type 4' methods, risk is analysed with respect to an asset that has previously been categorized as critical. The risk in relation to a threat combines the vulnerability of the critical asset only and the potential impact of the threat against the critical asset, i.e.:

$$Risk(Threat, Crit.Asset) = Vuln(Crit.Asset) \otimes Impact(Threat, Crit.Asset) \tag{6.2}$$

Since our goal is to assess the risk of attack paths of interacting nodes towards a critical target, we properly combine Equation (6.1) and Equation (6.2) as follows. Let $\mathcal{T}$ denote the critical target system and let $\mathcal{D}$ denote the set of all the assets (devices) in scope. Note that $\mathcal{D}$ contains both typical ICT systems, as well as cyber-physical and IoT or IoT-enabled components that may be directly or indirectly interconnected with $\mathcal{T}$. Let $\mathcal{AP} = (d_n \rightarrow \cdots \rightarrow d_1 \rightarrow \mathcal{T})$, $d_i \in \mathcal{D}$ denote an attack path of interacting nodes, where the threat is triggered in node d_n and the actual target of the attack is the critical target $\mathcal{T}$. Then, the risk for such and attack path is defined as follows:

$$Risk(Threat, \mathcal{AP}) = Likelihood(Threat, \mathcal{AP}) \otimes Vuln(Threat, \mathcal{AP}) \otimes Impact(Threat, \mathcal{T}) \tag{6.3}$$

The reason for combining Type 1 with Type 4 risk classes was to allow for fine-grained threat and vulnerability input from open sources (as supported by Type 1), and at the same time focus on the input of the critical target system (as supported by Type 4 risk formulas). Since the proposed methodology is source driven and target oriented, our goal is to assess the risk for various threat agents that may trigger an attack at the source node of an attack path, in order to eventually affect the critical target system. In our model, asset is replaced by an attack path $\mathcal{AP}$ of multiple interacting assets, where the destination of the path is the critical target system $\mathcal{T}$. The impact is assessed based on the consequences of the critical target $\mathcal{T}$. This is reasonable since the ultimate goal of the adversary is to harm the critical asset; the other systems in the path are used in order to extend the attack vector. However, the likelihood and the vulnerability assessment take into consideration the whole attack path, since the adversary is expected to combine any capability having on the interacting node, in order to gradually exploit all vulnerabilities within an attack path. Obviously, the optimal adversarial strategy is to combine vulnerabilities found at the entry point system d_n with vulnerabilities found in the whole chain, to pivot (horizontally or laterally) to the ultimate target $\mathcal{T}$.

6.2 Building Blocks: CVSS and CVE

Common Vulnerabilities and Exposures[1] (CVE), developed by MITRE, is a list of uniquely identifiable vulnerabilities, and is a 'de facto' standard for numerous SW products. CVSS [78] is an open framework that incorporates risk characteristics to assess the severity of CVE software vulnerabilities. CVSS in its latest version consists of three metric groups: *Base Score*, *Temporal*, and *Environmental* metrics. The Base Score includes the Exploitability and the Impact Metrics. The exploitability metrics include the Attack Vector (AV), Attack Complexity (AC), Privileges Required (PR), User Interaction (UI) and the Scope (S) whereas the Impact Metrics include the Confidentiality (C), the Integrity (I) and the Availability (A). The base score produces a score ranging from 0 (lowest) to 10 (most severe). A CVSS vulnerability is represented as a vector string, a compressed textual representation of the values used to derive the score. The corresponding values for each exploitability and impact metric are depicted in Table 6.1

The base score can be modified with temporal and environmental metrics, to fine-tune the vulnerability level. Temporal metrics contribute to the final score by taking into consideration the current state of the vulnerability, e.g. whether a full patch exists or not. The environmental metrics modify the base score to each custom environment; for example, the implementation of network-layer security controls, relevant to the particular vulnerability. Depending on the organization under assessment it may be possible to apply temporal and/or environmental metrics "en masse" for specific device/interaction types.

In our methodology, we adopt and make use of the CVSS v3.1 scoring system and its notation, in order to assess the vulnerability of the interactions between the nodes for both cyber as well as physical interactions. The reader is referred to [85] for detailed analysis of CVSS.

Exploitability metrics	**Values**
Attack Vector (AV):	(N)etwork, (A)djacent network, (L)ocal, (P)hysical
Attack Complexity (AC):	(L)ow, (H)igh
Privileges Required (PR):	(N)one, (L)ow, (H)igh
User Interaction (UI):	(N)one, (R)equired
Scope (S):	(C)hanged, (U)nchanged
Impact metrics	**Values**
Confidentiality (C):	(N)one, (L)ow, (H)igh
Integrity (I):	(N)one, (L)ow, (H)igh
Availability (A):	(N)one, (L)ow, (H)igh

Table 6.1: CVSS base exploitability and impact metrics with their corresponding values

[1] https://cve.mitre.org/

6.3 Terminology and Definitions

In order to assist the reader, we will first define the basic terminology. Then, before describing the methodology in detail, we provide a high-level description.

Interactions: We define as an *Interaction* between two systems (nodes), called the *source* node, say x and the *destination* node y and we denote as (x, y, type) the directional action or 'influence' that x may cause to y, due to their proximity and/or connectivity. We define two categories of interactions (each having detailed types): *physical* and *cyber* interactions.

Cyber interactions: They include all the actions that may be triggered by the source towards the destination node, due to their cyber connectivity. In order to model cyber interactions, we make use of two characteristics: the network connectivity level and the logical access level. Concerning the network connectivity level x and y may either reside to the same network or they may be connected via different network segments and/or technologies. Concerning the logical access of x to y we distinguish three access levels, none, low and high. None implies that x has no logical access at all at y; low corresponds to user-level access whereas high corresponds to privileged (e.g. root/admin) access. Table 6.2 summarizes the cyber interaction types.

	Logical Access		
Connectivity	None (no explicit access)	Low (user-level)	High (admin-level)
L2 (Local) Network	**C1**	**C2**	**C3**
L3 (Remote) Network	**C4**	**C5**	**C6**

Table 6.2: Cyber interaction types: A cyber interaction ($x \rightarrow y$) may belong to type C1–C6, based on the connectivity and the logical access of x to y [257]

Physical Interactions These include all the actions that may be triggered by x to y due to their physical proximity. The physical Attack Vector (`AV:P`) described in CVSS [85], is applied for M2M interactions that are capable to physically reach each another. In addition, `AV:A` is considered appropriate for physical interaction types P2 and P3, since Adjacent network access is adequate for physical interactions that require network proximity. We define three types of physical interactions, as shown in Table 6.3. Type P1 describes cases where devices equipped with moving parts or moving capabilities (e.g. IoT-enabled industrial robotic arm, a robot vacuum cleaner) are in proximity with the target system. Adversaries may exploit network/software vulnerabilities of the device to extend their physical reach and cause physical damage and/or gain physical proximity with the target system [180]. Type P2 describes I/O proximity for specific interfaces (e.g. infrared, proximity, luminosity). Such I/O interfaces can be vulnerable against nearby adversaries. For example, line-of-sight

Type	Description	Interface	Examples	Common attack patterns
P1	**Physical proximity** (x may use a moving part and/or moving capabilities to physically reach y)	Remotely controlled moving parts or devices	Robotic arm, crane, wheeled device, drone	Cause destruction/obstruction
P2	**Wireless I/O proximity** (x is in range with a wireless I/O interface of y)	Audio, Visual, Optical interfaces	Line-of-sight (LiDAR, IR), audio / video interfaces	I/O suppression/manipulation (e.g. introduce artifacts in optical sensors) Side-channel attacks (covert channels for data exfiltration)
P3	**Networks' proximity** (x and y at *different* networks that are in range)	Different, but shared-band wireless interfaces	e.g 802.11.x and 802.15.x operate at 2.4 GHz	DoS (jamming) - Packet injection attacks

Table 6.3: Physical interaction types based on the proximity between devices [257]

interfaces such as optical sensors of collision avoidance systems may be abused by introducing artifacts [217]. Other examples include the abuse of line-of-sight interfaces for creating covert channels to exfiltrate data [119, 232]. Furthermore, audio or video I/O interfaces have been proved to leak information as described in [16]. Finally P3 types are based on the fact that it is possible to cause jamming or even integrity attacks, when wireless interfaces that operate on the same bandwidth (even if they are running different protocols) are physically in range (e.g. [213, 217]). Typical characteristics (frequency/band, range) of wireless/infrared interfaces are presented in Tables 6.4 and 6.5 respectively.

Attack Paths Let $\mathcal{T}$ denote the critical target system and let $\mathcal{D}$ denote the set of all the assets (devices) in scope. We define as an *Attack Path* against a target system $\mathcal{T}$ and we denote as $\mathcal{AP} = (d_n \rightarrow \cdots \rightarrow d_1 \rightarrow \mathcal{T})$, $d_i \in \mathcal{D}$ a chain of interactions, where the threat is triggered in node d_n (the entry-point system) and the actual target of the attack is the critical system $\mathcal{T}$. We stress out that for systems that directly interact with the target, we will examine both cyber and physical interactions, since any direct interaction may be exploited by the adversary to harm the target system. For systems that are indirectly connected with the target, we only model their cyber dependencies, since they may be utilized in order to extend the attack vector, by successively compromising a chain of interactions towards the target system.

Cumulative Vulnerability Vector of an Interaction: $CVV\big((x, y, \text{type})\big)$ This is a CVSS-like vector representing the *combined* vulnerability level of an interaction. It has a central role in our methodology and is described in Section 6.6.

technology	Band	Range
ZigBee	2.4 GHz	10-20 m
Z-Wave	915 MHz USA - 868 MHz Europe	Up to 100m
WiFi	2.4 and/or 5 GHz	Up to 100m or greater
Bluetooth - BLE	2.4 GHz	Class 1 100m - Class 2 10m - Class 3 ¡ 10m
Low Frequency (LF) RFID	30 KHz - 300 KHz (typically 125/134 KHz)	Up to 10 cm
High-Frequency (HF) RFID	3 to 30 MHz (13,56)	Up to 1 m
Ultra-high frequency (UHF) RFID	860 - 960 MHz (900-915 in most countries)	Up to 12 m (passive), Up to 100 m (active)
SigFox	915 MHz USA - 868 MHz Europe - 433MHz Asia	30-50km (rural) 3 - 10 Km (Urban)
LoRaWAN	915 MHz USA /Australia- 868 MHz Europe - 433MHz Asia	2-3 Km (urban), 5 - 7 Km (rural)
Ingenu	2.4 GHz	15 Km (urban)
Weightless-P	169/433/470/780/ 868/915/923 MHz	Up to 2 Km (urban)
ANT	2457MHz	100 ft (ANT+)
DigiMesh	2.4 GHz	Up to 40 miles or more
MiWi	868/915 MHz and 2.4 GHz	From 30m (indoors) up to 120m (Outdoors)
EnOcean	868 MHz (Europe/China)- 902 MHz (North America) - 928 MHz for JAPAN	Up to 30m (Indoors)
DASH7 Alliance Protocol (D7A)	433/868/915 MHz	Up to 2 Km
Narrow Band Internet of Things (NB-IoT)	Licensed LTE frequency bands	Up to 15 Km (Urban) - 50Km (Rural)
WirelessHART	2.4 GHz	Up to 225m
KNX-RF	868 MHz	From 30m (indoors) up to 150m (Outdoors)

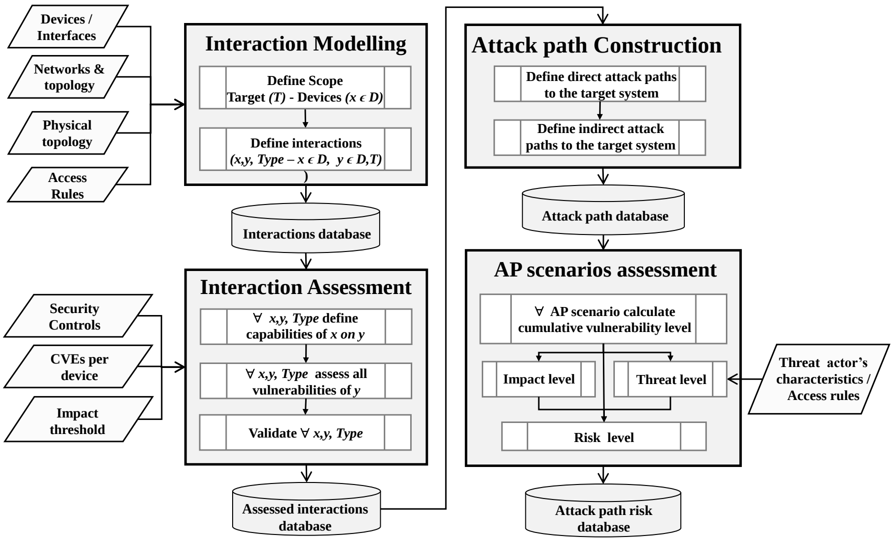
Table 6.4: Typical range and operational frequency of popular wireless technologies

Technology	Band	Range
IrDA-SIR	Wavelength: 850 up to 900 nm	At least 1m, 15 up to 30 degree angle (transmitter)
IrDA-MIR	Wavelength: 850 up to 900 nm	At least 1m, 15 up to 30 degree angle (transmitter)
IrDA-FIR	Wavelength: 850 up to 900 nm	At least 1m, 15 up to 30 degree angle (transmitter)
Night vision CCTV cameras	Wavelength: 700 up to 1000 nm	Depends on the location

Table 6.5: Typical operating distances and degree angles for infrared inputs/outputs

Cumulative Vulnerability Vector of an Attack Path: $CVV\big(\mathcal{AP}, AV\big)$ Similarly, it denotes a CVSS-like vector that represents the combined vulnerability level of an $\mathcal{AP}$ consisting of several interactions. Its computation is described in detail in Section 6.8.

6.4 A high-level description



Devices / Interfaces
Networks & topology
Physical topology
Access Rules
Interaction Modelling
Define Scope Target (T) - Devices (x ϵ D)
Define interactions (x,y, Type – x ϵ D, y ϵ D,T)
Interactions database
Interaction Assessment
∀ x,y, Type define capabilities of x on y
∀ x,y, Type assess all vulnerabilities of y
Validate ∀ x,y, Type
Security Controls
CVEs per device
Impact threshold
Assessed interactions database
Attack path Construction
Define direct attack paths to the target system
Define indirect attack paths to the target system
Attack path database
AP scenarios assessment
∀ AP scenario calculate cumulative vulnerability level
Impact level
Threat level
Risk level
Threat actor's characteristics / Access rules
Attack path risk database

Figure 6.1: High level description of the risk assessment methodology [257]

The proposed methodology will utilize CVSS information in order to construct CVSS-like vectors that will enable the assessment of the exploitability of the identified interactions, and ultimately the vulnerability level of attack paths against adversaries. In this way we will assess the implied capabilities of the source to the target node resulting from their interaction, in order to exclude those interactions that are not "mature enough" to be exploited by adversaries. Then, by combining the validated interactions, we will generate and assess attack paths that are more likely to happen, based on the current exposure status of their interactions. The proposed methodology, shown in Figure 6.1, consists of the following phases:

- **Phase 1 - Interaction modelling:** The goal of this phase is to model all potential cyber and physical interactions between the target $\mathcal{T}$ and all the devices in $\mathcal{D}$, as well as between devices themselves. It combines information such as a device's I/O and network interfaces, moving parts and their active ranges, devices' physical location, available networks with their cyber/physical characteristics and logical/physical access rules, to construct lists of interactions.

- **Phase 2 - Interaction vulnerability assessment:** The main goal in this phase is to assess all the potential interactions identified in Phase 1 by defining the cumulative vulnerability level of each interaction, based on existing CVEs per device as well as on environmental information. Essentially, this phase filters out those interactions that are not 'mature enough' to be exploited by potential adversaries in their current state.
- **Phase 3 - Attack Path Construction.** The goal of this phase is to construct all the attack paths against the target system, by exhaustively combining all the assessed interaction tuples provided by Phase 2. Attack paths may vary in length, by involving one or more interactions.
- **Phase 4 - Attack Path Assessment:** Finally all the attack paths defined is Phase 3 are assessed so as to calculate their risk level, based on a practical implementation of Equation (6.3). For each attack path the Cumulative Vulnerability Vector (CVV) of each interaction tuple is combined with the vulnerabilities of the initial node and the characteristics of various adversaries, to calculate the risk level for various attack path scenarios.

Figure 6.2 presents a graphical representation of the first three phases. As shown, the first modeled and assessed based on the input information, in order to construct valid attack paths to eventually be assessed.

6.5 Phase 1: Interaction modelling

During this phase we utilize all available information regarding device's physical characteristics, I/O interfaces and network connectivity, to construct all their cyber and physical interactions as defined in Section 6.3. Algorithm 1 implements interaction modeling. Let PT the input for Physical Topology related information, NT for Network Topology and AR for access capabilities respectively. The algorithm takes as input all of the described information, and outputs a set of lists (denoted as $InteractionLists[]$) containing all the direct and indirect interactions between the critical target $\mathcal{T}$ and any device $x \in \mathcal{D}$ in a structured way.

We define as the level-i interaction list (denoted as $InteractionLists[i] = \mathbb{L}_i$) the set of all the interactions of the form (x, y, type), where the shortest distance of the source node x from the target system $\mathcal{T}$ is i hops. In addition, since for the

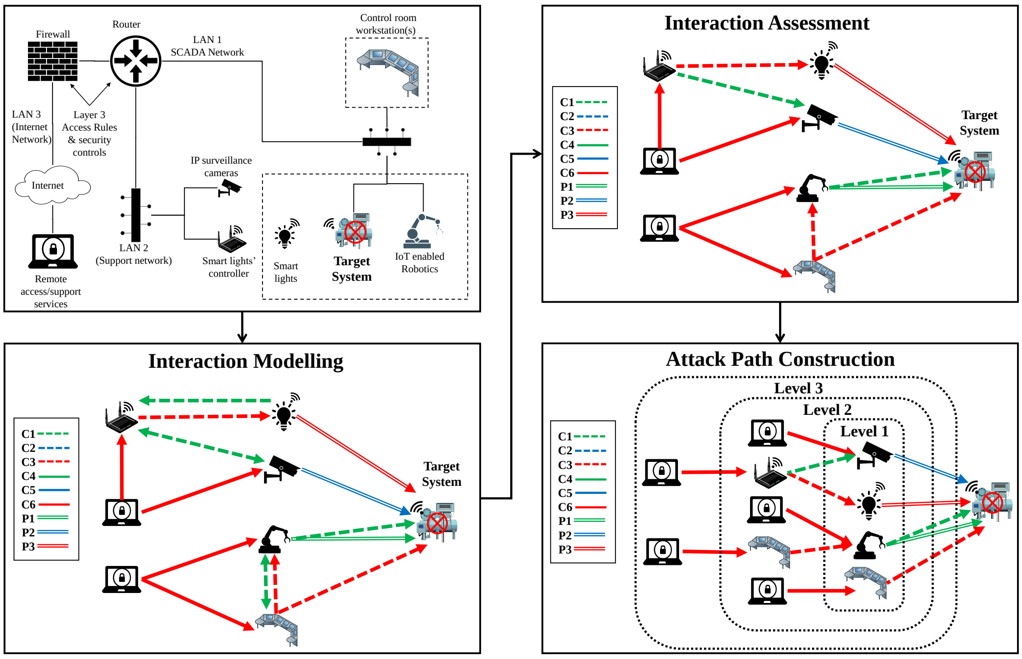


Figure 6.2: A graphical representation of interaction modelling, assessment and attack path construction phases [257]

direct interactions we model both cyber and physical interactions, it holds that if $(x, y, \text{type}) \in \mathbb{L}_1$ then $y \equiv \mathcal{T}$ and type $\in [P1|P2|P3|C1|\ldots|C6]$ (as defined in Tables 6.2 and 6.3). On the other hand, for all indirect interactions $\in \mathbb{L}_2, \ldots, \mathbb{L}_n$, it holds that $y \neq \mathcal{T}$ and type $\in [C1|\ldots|C6]$.

Algorithm 1 works as follows. First, all the direct interactions with the target system are computed to form the list $\mathbb{L}_1$ (see lines 2-3 in Algorithm 1). Then, all the indirect interaction lists $\mathbb{L}_i, i = 2, \ldots, n$ are recursively computed, by exhaustively examining the potential interactions of all the source nodes in level-i interactions, but now as being destination nodes of possible interactions (Lines 4-15). The algorithm avoids duplicating interactions already defined in previous lists, so that each interaction is defined once, in the shortest possible list. The procedure `IdentifyInteractions` is recursively called in the main algorithm. In the first call, since the destination of the interaction will be the target system $\mathcal{T}$, both physical and cyber interactions will be checked. For all other calls, only the cyber interactions will be modeled.

Since each call on `IdentifyInteractions` has computational cost proportional to $|\mathcal{D}|$, it is easy to see that the computational cost of Algorithm 1 will be proportional to $\mathcal{O}(|\mathcal{D}|^n)$ where n is the number of interaction lists. In our implementation,

```
Input  : $\mathcal{T}$=Target system. $\mathcal{D}$=The set of devices in scope and their
         corresponding interfaces. $PT$=Physical Topology. $NT$=Network
         Topology. $AR$=Access Rules.
Output: $InteractionLists[]$ = A set of lists containing all direct interactions with
         the target system ($\equiv \mathbb{L}_1$) as well as the devices themselves
         ($\equiv \mathbb{L}_i, i = 2, 3...n$)
Algorithm ModelInteractions()
    $i \leftarrow 1$ // Compute $InteractionLists[1](\equiv \mathbb{L}_1)$
    $InteractionLists[i] \leftarrow$ IdentifyInteractions($\mathcal{D}, \mathcal{T}, PT, NT, AR$)
    while ($TRUE$) do
        $InteractionLists[i+1] \leftarrow \emptyset$
        // Check all devices in Level-i as ‘target’ of any other
           device, in order to construct Level-(i+1) interactions
        while ( $(x, y, \text{type}) \leftarrow hasNext(InteractionLists[i])$ ) do
            $L_x \leftarrow$ IdentifyInteractions($\mathcal{D}, x, PT, NT, AR$)   // $x$ is a Level-i
             device
            $L_x \leftarrow L_x - \big(L_x \cap InteractionLists[i]\big)$ // Don’t duplicate in
               Level-(i+1), interactions already identified in Level-i.
               Possible if graph has loops
            $InteractionLists[i+1] \leftarrow InteractionLists[i+1] + L_x$
        end
        if $\big(InteractionLists[i+1] = \emptyset\big)$ then
            break // If no Level-(i+1) interactions exist, then exit
        end
        $i \leftarrow i + 1$
    end
return ($InteractionLists[]$) /* Interaction lists for all existing levels
   ($\mathbb{L}_1, \mathbb{L}_2, ..$) */
```

Algorithm 1: Identify and model all potential interactions in $\{\mathcal{D}, \mathcal{T}\}$.

```
/* Checks all devices $x \in \mathcal{X}$ for interactions with device $y$.  It
   outputs a list $\mathbb{L}$ of all such interactions, described as vectors
   $(x, y,$ Interaction_Type).  */
IdentifyInteractions($\mathcal{X}, y, PT, NT, AR$)
$\mathbb{L} \leftarrow \emptyset$
while $\big(x \leftarrow hasNext(\mathcal{X})\big)$ do
    /* Check for cyber dependencies based on network connectivity,
       network topology and access rules */
    if $\big(\ C_i \leftarrow CyberInteraction(x, y, NT, AR)\big) \neq \emptyset$ then
        $add\big(\mathbb{L},\ (x, y, C_i)\big)$ // $C_i \in$ (C1,...,C6) as defined in Table 6.2
    end
    /* Also check for physical dependencies with $\mathcal{T}$, based on physical
       topology information PT (including the interfaces and physical
       capabilities of $x$) */
    if $\big[\ (y = \mathcal{T})$ and $\big(P_i \leftarrow PhysicalInteraction(x, y, PT)\big) \neq \emptyset\ \big]$ then
        $add\big(\mathbb{L},\ (x, y, P_i)\big)$ // $P_i \in$ (P1,P2,P3) as defined in Table 6.3
    end
    $\mathcal{X} \leftarrow \mathcal{X} - x$ // Remove $x$ and continue until $\mathcal{X}$ is empty
end
return $\mathbb{L}$
```

Algorithm 1: (continued) – Procedure *IdentifyInteractions*

various ways are used to optimize the identification of interactions. First, during the identification of the systems $(\mathcal{D}, \mathcal{T})$ the physical characteristics (such as movement capabilities or proximity-based network interfaces and other non-typical interfaces) are identified. Thus, physical dependencies will only be examined for nodes with such capabilities. For example, a device equipped with moving parts/capabilities must be within its operating radius range in order to interact with the target system. Similarly, devices equipped with wireless interfaces are examined for physical interactions with the target system if their interfaces operate in the same frequency (see Table 6.4). This information is assumed in *physical topology* data-set, PT in our algorithm. For the cyber interactions, during the identification of the nodes, each network interface of each node will be assigned to its corresponding network. Cyber interactions will then be identified based on network relations table as well as network access rules.

Furthermore, the logical access level for each interaction is examined, in order to define the level or remote access capabilities for each interaction tuple. This information is assumed in *access rules* data-set denoted as AR in algorithm.

6.6 Phase 2: Interaction assessment

The goal of this phase is to filter out from further processing those interactions that are not 'mature enough' to be exploited by assessing their vulnerability level. For interactions that are considered as valid, their CVV, as defined in Section 6.3, is calculated.

Assessing whether an interaction (x, y, type) is valid or not, is based on the level of the influence that x has on y due to their interaction. Recall that by definition, an interaction characterizes the influence that the source node x has on the destination y, due to their network connectivity or physical proximity. Assume that x has been compromised by the adversary (partially or fully). Then, the adversary can take advantage of all the capabilities of x on y in order to compromise y (partially or fully), as the next step towards the actual target system $\mathcal{T}$. Apart from the explicit access that x has on y due to their interaction, an adversary controlling x may also attempt to extend the control on y, by exploiting the existing vulnerabilities of y.

For example, attempt to escalate the access level of x to y from user-level to admin-level access. *Assessment Strategy:* In order to assess an interaction, we first define their default (implied) impact and attack capabilities. These baseline attack capabilities of an interaction will be modelled using a CVSS-like vector, denoted as $IntCVSS_{base}$. Then, for each particular interaction we will use environmental information to transform the baseline capability interaction vector into a vulnerability vector, by taking into consideration the characteristics of the specific environment. The modified vulnerability interaction vector is denoted as $IntCVSS_{env}$. For example for a cyber interaction, $IntCVSS_{env}$ will take into consideration existing network security controls (if any), miss-configured access lists or context-specific access capabilities. For physical interactions, environmental information such as physical security controls and other context-specific information will be considered (e.g. in Table 6.10). Finally, in order to assess the possible ways that an adversary might exploit to escalate its control on y we will examine the resulting attack capabilities of x on y, with respect to the overall vulnerabilities identified on y.

6.6.1 Defining the capabilities and impact of interactions

As explained above, for each interaction type, we define a baseline CVSS-like vector, $IntCVSS_{base}$, representing the implied attack capabilities of x on y.

Cyber interactions: For the cyber interactions, recall that they have been defined based on the network connectivity and logical access of x to y (see Section 6.3 and Table 6.2). Thus, if x and y are connected at the same local network, we define the attack vector capability of the interaction as 'Adjacent Network' (`AV:A`), while for remote network connectivity, `AV:N` is assumed. Concerning the privileges required metric, we consider the implied logical access of each interaction type. In the case where the interaction type implies no access of x to y (e.g. nodes that only reside in the same or different network – types C1 and C4 respectively), then we set the implied privileges of x to y to 'None' (PR:N). Similarly, the baseline privileges of x to y is set to 'Low', for types implying non-privileged access C2 and C5 (e.g. x is an aggregator that has limited capabilities of to reading/writing and/or execute data on a sensor y). Finally, for types C3 and C6 the privileges metrics are set to 'High' (e.g. x is an e-health Web server that is able to remotely administer critical functions of an IoT-enabled medical infusion pump y). For the rest of the exploitability metrics we set the baseline attack complexity to 'High' and the user interaction to 'None'. The motivation is to assume as default values the most favorable for the adversary (although these values are modified when environmental characteristics are applied). Concerning the impact metrics, we consider that if an interaction does not imply any access privileges of x on y (C1 and C4), no impact can be caused on y by default. For interaction types that consider low level access of x on y (C2 and C5) we set the implied impact on y to 'Low' for all impact metrics (C-I-A), proportionally to the impact of a user access vulnerability. Similarly, for C3 and C6, we set the implied impact to 'High' for all impact metrics.

Table 6.6 presents the $IntCVSS_{base}$ vectors for all cyber interaction types. As presented in Table 6.7, the attack complexity and the impact metrics of $IntCVSS_{base}$ capability vector are modified in order to form the $IntCVSS_{env}$ vulnerability vector, depending on the available environmental information regarding security controls on network and/or application layer. For example, lack of security controls reduces the required `AC` of an $IntCVSS_{base}$ whereas a network security control (e.g. use of latest encryption schemes on network layer) can further reduce the corresponding impact metric (confidentiality).

		Exploitability Metrics					Impact Metrics		
	Type	AV	(M)AC	PR	UI	S*	(M)C	(M)I	(M)A
$IntCVSS_{base}$	C1	A	H	N	N	U	N	N	N
	C2	A	H	L	N	U	L	L	L
	C3	A	H	H	N	U	H	H	H
	C4	N	H	N	N	U	N	N	N
	C5	N	H	L	N	U	L	L	L
	C6	N	H	H	N	U	H	H	H
$IntCVSS_{env}$		(M): These metrics can be environmentally modified (See Table 6.7) *Scope is unchanged (U), for level 1 interactions							

Table 6.6: Defining the implied capabilities for each of cyber interaction type as a CVSS vector [257]

Physical interactions: For physical interactions we also use a similar approach. Due to physical proximity the attack vector is set as the implied access capability of

Network Security Controls	**(M)AC**	**Impact Modifiers**		
		M(C)	**M(I)**	**M(A)**
Not defined/Weak	H → L	No effect	No effect	No effect
Moderate	H	No effect	No effect	No effect
Strong	H	H → L	H → L	H → L
		L → N	L → N	L → N

Table 6.7: Proposed network environmental modifiers for $IntCVSS_{env}$ vector according to the corresponding security control level [257]

x on y (`AV:P/A`) depending on the interaction type (see Section 6.3 and Table 6.3). For the rest of the exploitability metrics we follow a same reasoning as in the case of cyber interactions, allowing the most favorable metrics for an adversary as the default values. The only difference is that for all the types the implied privileges are set to 'None', since a physical interaction does not require any kind of privileges (as defined in CVSS) of x on y.

Finally we consider the scope as 'Unchanged', since, physical interactions are only effect the target system $\mathcal{T}$. Similarly to cyber, the transformation of the baseline capabilities of physical interactions to a vulnerability vector is subject to environmental information. In particular, relevant security controls (see table 6.10) and the amount of damage that the source device's interface is capable of deliver to the target system are both taken into consideration for the final CVV to be calculated. Depending on the target type, several types of security controls may also be applicable (e.g. in [88, 89, 265]). An overview of how environmental security controls affect attack complexity and individual impact metrics of $IntCVSS_{base}$ capability vector is presented in Table 6.9.

		Exploitability Metrics					Impact Metrics		
	Type	AV	(M)AC	PR	(M)UI	S	(M)C	(M)I	(M)A
$IntCVSS_{base}$	P1	P	H	N	N	U	N	L	L
	P2	A	H	N	N	U	L	L	L
	P3	A	H	N	N	U	N	L	L
$IntCVSS_{env}$		(M): Can be modified, based on physical environment (See Table 6.9.)							

Table 6.8: Defining the implied capabilities for physical interactions as a CVSS-like vector [257]

6.6.2 Identifying the vulnerabilities of the destination node

For each target node of an interaction, we examine its existing vulnerabilities (CVEs). In addition, vulnerability chaining of single CVSS vectors is applied in specific cases, to assess the effect of *combined* vulnerabilities (see for example [85]). In any case,

Physical Security Controls	**(M) AC**	**Impact Modifiers**		
		(M)C	**M(I)**	**(M)A**
Not defined/Weak	H → L	No effect	No effect	No effect
Moderate	H	No effect	No effect	No effect
Strong	H	H → L	H → L	H → L
		L → N	L → N	L → N

Table 6.9: Proposed physical environmental modifiers for $IntCVSS_{base}$ vector according to the corresponding security controls for each impact metric [257]

	P1	P2	P3
Physical/combination barriers	✓		
Protective lockable casing	✓		
Proximity alerting systems	✓		
Integrity alerting systems	✓		
I/O redudancy		✓	
I/O filtering		✓	
I/O Protective shaders		✓	
I/O integrity check mechanisms		✓	
I/O logical check mechanisms		✓	
Device state inspection		✓	
Network isolation		✓	
Proactive/Reactive Frequency-Hopping Spread Spectrum (FHSS)			✓
Direct Sequence Spread Spectrum (DSSS)			✓
Hybrid FHSS/DSSS			✓
Antenna Polarization			✓
Ultra Wide Band Technology			✓

Table 6.10: Applicable security controls per physical interaction type (P1-P3) [257]

environmental information (temporal included) must first be applied before the vulnerability assessment and chaining process begins. **Single-vulnerability CVSS vectors:** Depending on the cyber interaction type, CVEs can be considered as possible single (non-chained) vulnerability vectors, if their attack vector is adjacent or remote network for C1-C3 (`AV:A/N`), or `AV:N` for C4-C6 respectively.

$$\forall\, CVE \texttt{ of } d \in \mathcal{D}, \texttt{ if AV:A/N then } CVE \in SingleCVSS \tag{6.4}$$

Chained-vulnerabilities CVSS vectors: Vulnerability chaining is based on the paradigm of [85] which demonstrates serial exploitation of vulnerabilities for privilege escalation, i.e. escalate the attack vector from local access to network or adjacent network (see Section 3.4 of [85]). In particular, we consider the cases where the exploitation of network vulnerabilities on y (`AV:A` or `AV:N`) that result in basic user access or an equivalent impact of `C:L/I:L/A:L` is combined with high-impact vulnerabilities (`AV:L`) to produce a chained vulnerability CVSS vector as described in Equation (6.5)[2]:

$$\mathit{ChainedCVSS} = [\text{AV:[N|A]},\quad \max(\text{AC}),\quad \min(\text{PR}),\quad \max(\text{UI}),\quad \max(\text{S}),\quad \max(\text{C,I,A})] \tag{6.5}$$

Validating CVSS vulnerability vectors: After vulnerability chaining is complete, all of the identified (single and chained) vulnerabilities of y are examined, to verify which of them are exploitable based on the attack capabilities of x on y, as defined in $IntCVSS_{env}$. Equation (6.6) is applied for each vector $CVSS \in \{SingleCVSS\ |ChainedCVSS\}$, i.e.

$$\text{If}\quad IntCVSS_{env}[\mathit{Exploitability}] \geq CVSS[\mathit{Exploitability}] \quad \text{then} \quad CVSS \in ValidCVSS \tag{6.6}$$

In Equation (6.6) the operator $\geq$ has the following meaning for each exploitability metric: AV:A $\geq$ AV:N (i.e., if x is assumed to have adjacent network access to y, then it is capable to exploit vulnerabilities that require either adjacent or remote access); AC:H $\geq$ AC:L (i.e., if node x is capable to trigger attacks against y requiring high complexity, then it is also capable to trigger low complexity ones); PR:H $\geq$ PR:L $\geq$ PR:N (in the same sense is x is already assumed to have high privilege access on y then it will be able to also exploit vulnerabilities on y requiring low privilege access or no logical access at all). For the rest of the exploitability metrics the explanation is straightforward.

[2] Function min/max is based on the following assumptions: `AV:N>A>L>P, AC:H>L, PR:H>L>N, UI:R>N, S:C>U, C/I/A:H>L>N.`

6.6.3 Assessing the vulnerability level of the interaction

Now the cumulative vulnerability level of an interaction $CVV\big(x, y, \text{type}\big)$, defined in Section 6.3, is computed as follows. Recall that $IntCVSS_{env}$ is a cvss-like vector that defines the actual (environmental) capabilities that x has on y due to their interaction, and that $ValidCVSS$ is a set of all valid (i.e. potentially exploitable) vulnerabilities identified on the destination node y, either single or chained ones. The vulnerability vector CVV that characterizes this interaction will be chosen among the above, based on the following procedure.

$$\text{CVV}\big((\text{x,y, type})\big) = \text{V} \in (ValidCVSS_y, IntCVSS_{env})\ \texttt{s.t.:}\ \begin{cases} \texttt{V has max(Impact,Exploitability)} & \text{if } y = \mathcal{T} \\ (C, I, A) \geq L\ \&\ \texttt{V has max(Expl., Impact)} & \text{if } y \neq \mathcal{T} \end{cases} \tag{6.7}$$

For all `level-1` interactions, the primary criterion for choosing the vector to be assigned as $CVV\big(x, \mathcal{T}, \text{type}\big)$ is considered the impact rather than the exploitability sub-score, since we are interested in identifying the maximum possible damage that the target node may exhibit by each interaction.

For `level-i`, $i \geq 2$ interactions, the cumulative vulnerability level $CVV\big(x, y, \text{type}\big)$, $y \neq \mathcal{T}$, is assessed as follows. From the $IntCVSS_{env}$ as well as from all the valid single and chained vulnerability vectors of y, $CVSS \in ValidCVSS$ choose the one that: (i) concerning its impact metrics, it satisfies (C $\geq$ L & I $\geq$ L & A $\geq$ L) and (ii) has the highest exploitability sub-score. If more than one exist that satisfy the above criteria, choose the CVSS vector that has the maximum impact sub-score. The main motivation for this process is to to ensure that interactions will be assigned to the CVV vector that corresponds to at least a partial compromisation on y (assured by the impact threshold) with the minimum required effort (i.e. the higher exploitability sub-score).

In both cases, if there exist more than one valid vulnerability vectors with identical exploitability and impact sub-scores, the single is preferred over the chained (if any). Finally, if no CVSS vector exists that satisfies the required criteria set in Equation (6.7), the interaction is considered as invalid and CVV is set to $\emptyset$. These rules are described in Equation (6.7). Note that the order of the arguments in function max denotes their priority in each case. Algorithm 2 summarizes the interaction vulnerability assessment phase.

6.7 Phase 3: Attack path construction database

In this phase, all possible attack paths against the target system $\mathcal{T}$ are constructed, by exhaustively combining all the assessed interactions, produced in the previous phase.

```
Input  : $InteractionLists[\,]$ ($\equiv \mathbb{L}_i, i = 1, 2, ...n$) : A set of lists containing all
         interactions produced by Algorithm 1.
         $\{CVE_d\}$ : Sets of CVE/CVSS (environmental) vectors $\forall\ d \in \mathcal{D}$.
Output: $AssessedLists[\,]$ ($\equiv \mathbb{AL}_i, i = 2, 3...n$) : A set of lists containing all
         assessed interactions.
AssessInteractions($InteractionLists[\,], \{CVE_d\}$)
for $InteractionLists[i]$, $i : 1 \ldots\ n$ do
    $AssessedLists[i] \leftarrow \emptyset$;   $CVV \leftarrow \emptyset$
    while $\big(\ (x, y, \text{type}) \leftarrow hasNext(InteractionLists[i])\ \big)$ do
        Define $IntCVSS_{base}(x, y, \text{type})$ /* Based on Tables 6.6,6.8 */
        $IntCVSS_{env}(x, y, \text{type}) \leftarrow$ ApplyEnv$\big(IntCVSS_{base}(x, y, \text{type})\big)$ /* As
           defined in Tables 6.7,6.9 */
        if $type \in [C1, \cdots C6]$ /* Chaining cyber interactions */
         then
            for $CVE \in \{CVE_y\}$ do
                $SingleCVSS_y \leftarrow$ SingleCVE($CVE$) // Based on Eq. (6.4)
                $ChainedCVSS_y \leftarrow$ ChainCVE($CVE$) // Based on Eq. (6.5)
                $ValidCVSS_y \leftarrow$ ValidCVE($SingleCVSS_y$, $ChainedCVSS_y$)
                    // Based on Eq. (6.6)
            end
        end
        $CVV \leftarrow$ CalcCVV($ValidCVSS_y$, $IntCVSS_{env}$) /* Calculate
           interaction's $CVV$ as described on Eq. (6.7) */
        $add\big(AssessedLists[i], (x, y, \text{type}, CVV)\big)$
    end
end
return $AssessedLists[i]$,   $i$= 1,. . . ,$n$
```

Algorithm 2: Assess Identified Interactions (*AssessInteractions*)

The attack path construction is described in Algorithm 3. The main algorithm (lines 1–22) works as follows. First, all the assessed level-1 interactions (i.e., direct interactions with the target system $\mathcal{T}$) are defined by default as one-hop attack paths ($\mathbb{AP}_1$). Then all the level-i attack paths $\mathbb{AP}_i$, $i > 1$, are computed recursively using $\mathbb{AP}_{i-1}$ and all the assessed interaction lists up to level-i ($\mathbb{AL}_1, \ldots, \mathbb{AL}_i$), by exhaustively examining if the destination node of a level-i interaction is the initial (source) node in each level-($i-1$) attack path. The final output is a list of lists $AttackPaths[i][j]$, containing all the valid chains of interactions of depth i towards the target system $\mathcal{T}$.

```
Input  : $AssessedLists[i] \equiv \mathbb{AL}_1, \dots \mathbb{AL}_n$. A set of lists containing all the
         assessed interactions between devices themselves $\in \mathcal{D}$ (Level-2,...) and
         against the target system $\mathcal{T}$ (Level-1)
Output: $AttackPaths[i][] \equiv \mathbb{AP}_1, \mathbb{AP}_2, \dots \mathbb{AP}_n$. A list of lists containing chains of
         interactions from an initial node $\in \mathcal{D}$ against $\mathcal{T}$. $\mathbb{AP}_i$ will contain the
         attack paths of depth $i$.
Algorithm ConstructAttackPaths()
    for ($i \leftarrow 1;\ i = n;\ i \leftarrow i + 1$) // Initialize all attack path lists.  $n$:#
       of assessed lists
     do
        $AttackPaths[i][] \leftarrow \emptyset$
    end
    // Define $\mathbb{AP}_1$ first.  By default, all interactions $\in \mathbb{AL}_1$ are
       level-1 Attack Paths.
    $i \leftarrow 1,\ \ j \leftarrow 1$
    while ( $(x, y, Type, CVV) \leftarrow hasNext(AssessedLists[i])$ and CVV$\neq \emptyset$) do
        $add\big(AttackPaths[i][j],\ [(x, y, Type, CVV)]\big)$
        $j \leftarrow j + 1$
    end
    // Recursively compute $\mathbb{AP}_i, i \in 2, \dots, n$ using $\mathbb{AP}_{i-1}$ and $\mathbb{AL}_i$.
    $i \leftarrow i + 1$
    while ( $(x, y, Type, CVV) \leftarrow hasNext(AssessedLists[i])$ and CVV$\neq \emptyset$) do
        $j \leftarrow 1,\ \ k \leftarrow 1$
        while ( $AttackPaths[i-1][j] \leftarrow hasNext(AttackPaths[i-1])$ ) do
            if (isSource($y,\ AttackPaths[i-1][j]$) ) then
                $add\big(AttackPaths[i][k],$ append$((x, y, Type, CVV), AttackPaths[i-$
                  $1][j])\big)$
                $k \leftarrow k + 1$
            end
            $j \leftarrow j + 1$
        end
        $i \leftarrow i + 1$
    end
    return ($AttackPaths[i][]$) /* Attack paths $\mathbb{AP}_1, \mathbb{AP}_2, ..$ */
```

Algorithm 3: Attack Path Construction Algorithm

$IntCVSS_{base}$	A CVSS-like capability vector assigned on the interaction based on the interaction's type, using Table 6.6 (for cyber) or Table 6.8 (for physical interactions)
$IntCVSS_{env}$	The modified $IntCVSS_{base}$ vector based on environmental information for each particular interaction (e.g. see Tables 6.7 and 6.10).
$\{SingleCVSS\}$	A list of all the single CVSS vectors corresponding to vulnerabilities identified in y satisfying Equation (6.6)
$\{ChainedCVSS\}$	A list of all the CVSS vectors of the chained vulnerabilities of y, computed based on Equation (6.5) and satisfying Equation (6.6)
$CVV\big((x, y, \text{type})\big)$	The Cumulative Vulnerability Vector of an interaction as defined on Equation (6.7)

Table 6.11: Summary of all vectors utilized in interaction assessment [257]

```
/* Boolean function that checks if a node $d$ is the source node for a
   given attack path $AttackPaths[i][j] = [(x_1, y_1, Type_1), \ldots, (x_i, y_i, Type_i)]$.
   */
Procedure isSource($d$, $AttackPaths[i][j]$)
    $(x_1, y_1, Type_1) \leftarrow AttackPaths[i][1]$
    if ($d = x_1$) then
        return (TRUE)
    else
        return (FALSE)
    end

/* Takes as input an interaction and an attack path of depth $i$.
   Appends the given interaction at the beginning and returns a new
   attack path of depth $i + 1$.  */
Procedure append($(x_0, y_0, Type_0)$, $[(x_1, y_1, Type_1), \ldots, (x_i, y_i, Type_i)]$)
    for ($k \leftarrow i$; $k$=1; $k \leftarrow k - 1$) do
        $(x_{k+1}, y_{k+1}, Type_{k+1}) \leftarrow (x_k, y_k, Type_k)$
    end
    $(x_1, y_1, Type_1) \leftarrow (x_0, y_0, Type_0)$
    return $\big([(x_1, y_1, Type_1), \ldots, (x_{i+1}, y_{i+i}, Type_{i+1})]\big)$
```

Procedure(*isSource* & *append* – Algorithm 3)

The procedures *isSource* and *append* are described for clarity.

Note that in Algorithm 3 the interaction tuples have been extended to also include their cumulative vulnerability vector, which was defined and assessed in Phase 2. In the case where interactions have null CVV value (recall that this is possible, as described in Section 6.6.3), they are considered as invalid and are excluded from any phase of the attack path construction (lines 7 and 12). It is easy to see that the computational cost of Algorithm 3 will be proportional to the product of the size of all the assessed lists, i.e., $\mathcal{O}(|\mathbb{AL}_1| \cdots |\mathbb{AL}_n|)$.

6.8 Phase 4: Attack path scenarios assessment

The attack paths constructed in the previous phase can now be assessed. The risk of each attack path will be assessed using Equation (6.3), as defined in Section 6.4. Recall that the risk for each attack path, takes into consideration the vulnerability of the whole attack path, the likelihood of a threat against the attack path being realized, and finally the impact on the actual critical target system.

The vulnerability level of each attack path combines the cumulative vulnerability level of all the interactions that form the attack path, i.e. $\{CVV\} \in \mathcal{AP}$, which have been assessed during the second phase (Section 6.6). In addition, we also consider the vulnerabilities of the initial ('entry') node of each attack path, i.e. the source node of the level-n interaction, for each attack path of length n. Recall that for each assessed interaction the CVV calculation has considered the capabilities of the source node and the vulnerabilities of the destination node. Thus, the vulnerabilities of the initial entry node have not been considered.

In order to examine all applicable threat agents against an attack path, for the initial node we first calculate all the applicable CVV vectors, one for each available `AV:N/A/L/P)`. As in Section 6.6.3 each individual CVV must meet the impact threshold criterion. As defined in Section 6.3, we denote as $CVV\big(\mathcal{AP}, \texttt{AV}\big)$ the CVV for a specific attack path and for a specific Attack Vector $\in$ [N|A|L|P]. For example $CVV\big(\mathcal{AP}_1, \mathrm{N}\big)$ denotes the CVV of $\mathcal{AP}_1$ for the attack vector 'Network'. The threat level for each attack path will then be assessed based on threat modeling against each available `AV` of the initial node of the path. Recall that, by definition, this node will be the entry point for an adversary exploiting an attack path. Thus, we will model and assess all the applicable threat agents that are capable of utilizing different attack vectors against the initial node. For each attack vector of an attack path, the corresponding threat level is determined by taking into consideration the relevant $CVV\big(\mathcal{AP}, AV\big)$ exploitability metrics, physical/network characteristics of the initial node, as well as adversarial profiling features including, among others, required resources, motivation and even current threat landscape reports. Finally, the impact level for all attack paths will be based on the actual business impact that the loss of confidentiality, integrity and availability of the target system has on the organization. We utilize the impact metrics (C,I,A) of the level-1 interaction where $\mathcal{T}$ is the destination node, and modify them properly by applying the corresponding *Impact Subscore Modifier* as defined in the CVSS.

6.8.1 $Vuln(Threat, \mathcal{AP})$: Calculating the vulnerability level of attack paths

As discussed above, for an attack path $\mathcal{AP}$, this process will combine the cumulative vulnerability CVV of each interaction involved in $\mathcal{AP}$ along with the vulnerabilities of the initial node of a path, to form, for each attack path, the CVV(s) for all existing attack vectors of the path's entry node, i.e. $CVV\big(\mathcal{AP}, AV\big)$. At first all individual CVEs of the initial entry node are processed to form single and/or chained CVSS vectors.

Similarly to Section 6.6.2, for each possible AV a single or a chained vulnerability with the highest impact and exploitability sub-score is selected to from the CVSS vector. Each of the latter is then combined using Equation (6.8):

$$CVV\big(\mathcal{AP}, AV\big) = [\text{AV : [N|A]}, \max(AC), \max(PR), \max(UI), \max(S), Level_1(C, I, A)] \tag{6.8}$$

6.8.2 $Likelihood(Threat, \mathcal{AP})$: Calculating the threat level of attack paths

After all the relevant $CVV\big(\mathcal{AP}, AV\big)$ have been calculated, the threat likelihood can be defined. In order to calculate the threat level one must first identify all available profiles of threat agents that fit the organization under assessment. Then, the corresponding capabilities for each type of the adversary are defined by utilizing the CVSS exploitability metrics `AV/AC/PR/UI` (see Figure 6.3).

For example, a disgruntled employee is considered as someone with both logical as well as physical access to internal networks/devices (`AV:N/A/L/P`), restricted (user) access (`PR:Low`), basic computer skills (`AC:Low`) and is not relying on any user interaction in order to launch an attack (`UI:None`). On the other hand, cyber criminal groups, mostly attack organizations from external networks (e.g. Internet - `AV:N`), without the need of logical access (`PR:None`), consist of highly skilled adversaries (`AC:High`) and are capable of gaining initial foothold to the organization either by exploiting network vulnerabilities or via spear-phishing campaigns (`UI:Required/None`). In order to define the threat level we adopt the context and scale as described in [235].

For each AV of $CVV\big(\mathcal{AP}, AV\big)$ of the initial node of an attack path the corresponding access level (physical or network) of the adversary is defined. For physical access ($AV \equiv P$), *public* applies to devices which are placed in a public places (e.g. an IP camera in a outside a building), *private* can be considered an area where the access is limited to certain groups of people (e.g. an IP surveillance camera in a corpo-

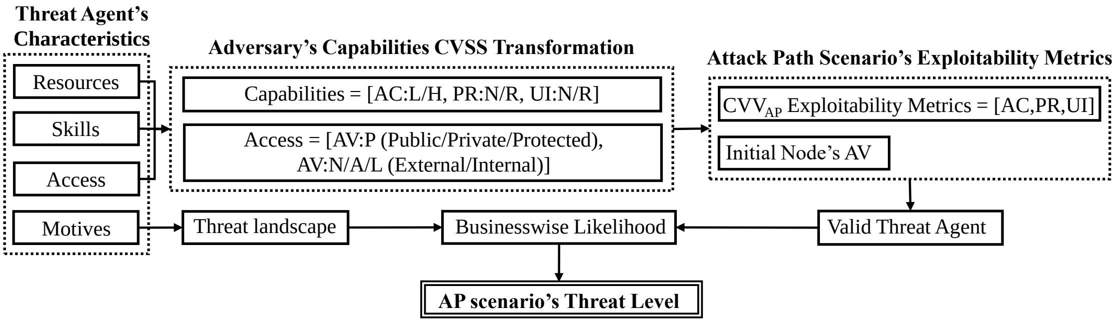

Threat Agent's Characteristics
Resources
Skills
Access
Motives
Adversary's Capabilities CVSS Transformation
Capabilities = [AC:L/H, PR:N/R, UI:N/R]
Access = [AV:P (Public/Private/Protected), AV:N/A/L (External/Internal)]
Attack Path Scenario's Exploitability Metrics
CVV_AP Exploitability Metrics = [AC,PR,UI]
Initial Node's AV
Threat landscape
Businesswise Likelihood
Valid Threat Agent
AP scenario's Threat Level

Figure 6.3: Threat level (likelihood) calculation methodology [257]

rate garage accessible only to employees) whereas *protected* can be considered a place heavily monitored and safeguarded by physical access security systems (e.g. a smart thermostat placed inside a data center). Similarly, for network access ($AV \equiv L, A, N$) we characterize as `internal` networks that are accessible from within the corporate environment whereas `external` are the ones that reside outside the organization's premises, the Internet included.

In order to match all applicable threat agents for each attack path scenario each individual metric of $CVV\big(\mathcal{AP}, AV\big)$ is compared to the corresponding metrics of each attacker profile. Then, for the adversary types that satisfy all individual criteria described in the previous paragraph, the corresponding likelihood for each particular threat agent is applied.

6.8.3 $Impact(Threat, \mathcal{T})$: Calculating the impact level of attack paths

In order to assess the actual impact that the organization suffers from each attack path in terms of CIA, we utilize each individual impact metric of the `Level-1` interaction tuple and apply the appropriate security requirement weights as defined in the CVSS. Guidelines for defining these weights according to the type of the target system can be found in the *CVSS Guide* (see Section 3.11 of [85]), as well as in several other publications such as [24]. For example, the applicable security requirements' weights for a power generator could be set to `High` for integrity and availability and `Low` for confidentiality.

In Table 6.12, we define the values of each individual CIA impact metric after the proper weight is applied. As in threat level, we adopt a [*Very Low... Very High*] scale,

	Security requirements			
CIA metrics	**Low**	**Medium**	**High**	**Not Defined**
None	None	None	None	None
Low	Very Low	Low	Moderate	Low
High	Moderate	High	Very High	High

Table 6.12: Transformation matrix of individual vulnerability impact metrics of level 1 interactions (attack paths) based on the CVSS corresponding security requirements [257]

identical to the context and scale of NIST [235]. Finally the overall impact level can be computed by combining the individual (CIA) impact metrics (see Table 6.13).

Impact Level																																				
Conf.	**Integrity**																																			
	None						**Very Low**						**Low**						**Moderate**						**High**						**Very High**					
	Availability																																			
	N	**VL**	**L**	**M**	**H**	**VH**	**N**	**VL**	**L**	**M**	**H**	**VH**		**VL**	**L**	**M**	**H**	**VH**		**VL**	**L**	**M**	**H**	**VH**		**VL**	**L**	**M**	**H**	**VH**		**VL**	**L**	**M**	**H**	**VH**
N	N	VL	VL	VL	VL	L	VL	VL	VL	VL	L	L	VL	VL	VL	L	L	L	VL	VL	L	L	L	M	VL	L	L	L	M	M	L	L	L	M	M	M
VL	VL	VL	VL	VL	L	L	VL	VL	VL	L	L	L	VL	VL	L	L	L	M	VL	L	L	L	M	M	L	L	L	M	M	M	L	L	M	M	M	H
L	VL	VL	VL	L	L	L	VL	VL	L	L	L	M	VL	L	L	L	M	M	L	L	L	M	M	M	L	L	M	M	M	H	L	M	M	M	H	H
M	VL	VL	L	L	L	M	VL	L	L	L	M	M	L	L	L	M	M	M	L	L	M	M	M	H	L	M	M	M	H	H	M	M	M	H	H	H
H	VL	L	L	L	M	M	L	L	L	M	M	M	L	L	M	M	M	H	L	M	M	M	H	H	M	M	M	H	H	H	M	M	H	H	H	VH
VH	L	L	L	M	M	M	L	L	M	M	M	H	L	M	M	M	H	H	M	M	M	H	H	H	M	M	H	H	H	VH	M	H	H	H	VH	VH

Impact Level: N=No Impact (None), Very Low= VL, Low = L, Moderate =M, High = H, Very High = VH

Table 6.13: Impact level calculation matrix [257]

6.8.4 Attack path risk assessment

By combining all the above information, the risk level of each attack vector for each attack path can be computed, according to our risk assessment formula of Equation (6.3). This is essentially computed using the risk matrix shown in Table 6.14. Similarly to impact the context and scale of risk level is identical with the one described in [235].

6.9 Attack path scenario risk mitigation

Since the implementation of security controls varies, granular security policies can be tested and implemented, e.g. from applying low cost security controls like system patching, medium cost controls like ICT vulnerability patching, up to targeted policies such as SW security hardening on the selected nodes.

Depending on a pre-selected risk threshold, the assessor can identify which attack path scenarios exhibit an unacceptable security risk. Then, the assessor can

Risk Level																									
Vulnerability Level	Impact Level																								
	Very Low					Low					Moderate					High					Very High				
	Threat Level																								
	VL	**L**	**M**	**H**	**VH**	**VL**	**L**	**M**	**H**	**VH**	**VL**	**L**	**M**	**H**	**VH**	**VL**	**L**	**M**	**H**	**VH**	**VL**	**L**	**M**	**H**	**VH**
Low	VL	VL	L	L	M	VL	L	L	M	M	L	L	M	M	M	L	M	M	M	M	M	M	M	M	H
Medium	VL	L	L	M	M	L	L	M	M	M	L	M	M	M	M	M	M	M	M	H	M	M	M	H	H
High	L	L	M	M	M	L	M	M	M	M	M	M	M	M	H	M	M	M	H	H	M	M	H	H	VH
Critical	L	M	M	M	M	M	M	M	M	H	M	M	M	H	H	M	M	H	H	VH	M	H	H	VH	VH

Risk Level: Very Low= VL, Low = L, Moderate =M, High = H, Very High = VH

Table 6.14: Risk calculation matrix for assessing $Risk(Threat, \mathcal{AP})$ by combining $Vuln(Threat, AP)$, $Likelihood(Threat, \mathcal{AP})$ and $Impact(Threat, \mathcal{T})$, as defined in Equation (6.3) [257]

implement the mitigation plan based on the organization's security policies and procedures. In addition, our methodology enables the assessor to add alternative mitigation schemes. For example, if impact is considered of utmost importance the proposed strategy is to apply the appropriate security controls at all the nodes and corresponding networks of `Level-1` interactions. In addition, the assessor may choose to eliminate certain types of adversaries just by focusing on applying the proper security countermeasures on entry nodes. Finally, in situations where security policies and procedures is difficult to implement and/or an intermediate response is needed, the assessor may choose to prioritize the mitigation process by selecting specific devices that have the highest multitude of attack path scenarios and/or are above a predefined risk level. All of the aforementioned mitigation scenarios can be simulated and the most efficient, cost beneficial security policies and procedures can then be selected. Prioritization on such mitigation actions based on the results of our risk assessment methodology is presented in a PoC scenario in Chapter 9.

SECTION IV
Methodology Validation

CHAPTER 7
ASSESSING THE CRITICALITY OF IOT-ENABLED ATTACKS

In order to determine the criticality level on the attacks and corresponding vulnerabilities identified in Chapters 3 and 4, we apply the methodology presented in 5 in order to reproduce realistic attack scenarios[1] that describes the environment of the attack, the adversary and the actual target (See Tables 7.1-7.7) in a worst-case scenario approach. In case of real incidents where such information is available, the attack scenario describes the actual environment/target that the attack was realized. In the case of PoC attacks, we adopt hypothetical, yet realistic attack scenarios, mostly applied in related state-of-the-art research (e.g. [90, 193, 224, 232, 233]), as well as on sector-specific technical reports of major security companies [34, 131, 180]. Then, the attack is assessed based on the attack scenario, using the risk factors, i.e. threat, vulnerability and impact levels.

Especially for the impact factor, each attack scenario is decomposed and assessed on the basis of the connectivity level between the IoT device (the attack enabler) and the target critical system or service. IoT devices of one application domain may also affect other application domains (e.g. use of industrial automation devices such as smart meters in home applications or use of smart lights in industrial environments). As described in Section 5.1, the IoT is not always the actual target; for each attack scenario presented in Chapter 3 we analyze the worst-impact connectivity path, i.e. the one that would affect the most critical target in realistic situations. For instance, an attack against an industrial actuator usually has high impact on SCADA systems directly connected to it, and in this case, we will examine the impact of the direct (known) attack path. In other scenarios, it may be more important to examine the impact of an indirect (hidden) attack path against a target system with an indirect connection with the IoT. Finally, in some scenarios the impact caused to a system that is not even indirectly connected to the IoT may be more significant, and in these cases we assess the impact of the subliminal attack paths.

[1] For some attacks, we may describe more than one attack scenario.

7.1 Attacks on Industrial IoT-enabled SCADA field devices, systems and services

Attack description and attack scenarios				Threat assessment				Vulnerability assessment			Impact assessment			Criticality
Description (Year,Type)	Attack vector	Weaknesses found (* exploited)	Attack scenario	Access [Physical, Logical]	Capabilities [Resources, Tech.Skills]	Motiv.	Thr. level	Embedded [H/W, S/W]	Network [Protocols, Key Manag.]	Vuln. level	Connectivity with critical systems	Potential impact	Imp. level	
[81,156] Sabotage of SCADA systems by reprogramming PLCs - Stuxnet (2010, Real)	1. Infect a Windows PC in the IT network (worm uses a Windows rootkit to remain stealth) 2. Self-replicate to other computers, e.g. through network shares or removable drives 5. Find and abuse Siemens Step7 software to program PLCs and sabotage centrifuges	* Windows 0-day exploits (2 for self-replication and 2 for escalating privileges) * PLC 0-day exploits * Lack of proper network segmentation	Nation state adversaries target a hostile nation's CIs	[Insider, Unpriv.]	[High, Expert]	Strong	**Low**	[Minor, Major]	[Major, Minor]	**High**	**Indirect:** The adversaries attack the IT network in order to pivot to SCADA field devices (PLCs)	It is estimated that the attack destroyed 984 uranium enriching centrifuges, decreasing by at least 30% the enrichment efficiency	**High**	**Medium**
[287] Attacks on SCADA honeypots through HMIs (2013, PoC)	1. Locate the ICS (honeypot) using Shodan 2. Gain access to the secure HMI area 3. Modify device settings (pump pressure and water temperature) changing HMI set-points 4. Schedule a pump shutdown	* Poor IT security policy (e.g. user unawareness, no email A/V) * HMI servers' vulnerabilities, (e.g. SQL injection, XSS bugs, unpatched OS)	Cyber criminals spy a large company's CI and halt the operation of target SCADA devices	[Outsider, Unpriv.]	[Moderate, Expert]	Strong	**Medium**	[Minor, Major]	[Major, Major]	**High**	**Indirect:** The adversaries attack the SCADA C&C servers in order to pivot to the connected devices	The attack can disrupt production line causing major economic/reputation loss and damage the equipment	**High**	**High**
[90] Ransomware attacks on SCADA systems through Internet-facing PLCs (2017, PoC)	1. Locate vulnerable PLC models using Shodan 2. Infect a PLC with ransomware 2. Worm self-spreads horizontally across same-vendor PLCs 3. Worm locks PLCs and sends a ransom note (through PLC email client)	* Easy to locate vulnerable PLCs * Weak authentication in most PLCs * No integrity protection	Terrorists infect the PLCs of a city water treatment plant and threaten to increase the levels of chlorine	[Outsider, Unpriv.]	[Moderate, Expert]	Strong	**Medium**	[Moderate, Major]	[Major, Major]	**High**	**Direct:** PLCs are attacked directly from the Internet and are part of the CI	The attack may affect people safety, lead to loss of public confidence, or cause significant financial loss	**High**	**High**
[253] Attacks on SCADA systems by introducing an infected PLC (2016, PoC)	1. Physically install an infected PLC to the plant 2. Worm spreads to other PLCs through TCP port 3. Worm contacts C&C center and manipulates more PLCs 4. DoS: Timeout period is altered and PLC is entered an endless loop	* No integrity protection * Disabled access protection (by default) * Weak crypto scheme	A malicious employee of the supplier company introduces infected PLCs to the production line	[Insider, Unpriv.]	[Moderate, Expert]	Strong	**Low**	[Moderate, Major]	[Major, Major]	**High**	**Direct:** PLCs are attacked on-site and are part of the CI	Disrupt production line causing major economic/reputation loss and damage equipment	**High**	**Medium**
[121,136] Attacks on automated tank gauges (ATGs) (2015 PoC)	1. Locate vulnerable ATGs through search engines (TCP port 10001). 2. Access to the unprotected serial control/-monitor port 3. Spoof fuel level report causing station shutdown	* Insecure web interface * No credentials or poor authentication	Hacktivists attack and shutdown multiple fueling stations	[Outsider, Unpriv.]	[Basic, Novice]	Weak	**High**	[Moderate, Major]	[Major, Major]	**High**	**Direct:** ATGs are attacked directly from the Internet and are part of the CI	The adversary can shutdown multiple fueling stations creating user discomfort, loss of public confidence, and some financial loss	**Low**	**Medium**
[223] Attacks on industrial robots (2017 PoC)	1. Locate vulnerable industrial robot using Shodan 2. Compromise main computer bypassing authentication (static FTP credentials) and upload malicious payload 3. Cause FlexPendant to auto-execute the malicious code 4. Compromise robot altering its firmware, e.g. PID controller parameters	* Insecure web interface * Default credentials or poor authentication * RobAPI vulnerable to buffer overflow * Vulnerable OS * Missing code signing	Microdefects are injected into a volume of products which escape detection by vendor's checks and end up with the customers	[Outsider, Unpriv.]	[Moderate, Intermed.]	Moderate	**Medium**	[N/A, Major]	[Major, Major]	**High**	**Direct:** Robots are attacked directly from the Internet and are part of the CI	The adversary can sabotage production outcome, threat human safety, and inflict major financial loss	**High**	**High**

7.2 Attacks on smart power grid infrastructure

Attack description and attack scenarios				Threat assessment				Vulnerability assessment			Impact assessment			Criticality
Description (Year,Type)	**Attack vector**	**Weaknesses found (* exploited)**	**Attack scenario**	**Access** [Physical, Logical]	**Capabilities** [Resources, Tech.Skills]	**Motiv.**	**Thr. level**	**Embedded** [H/W, S/W]	**Network** [Protocols, Key Manag.]	**Vuln. level**	**Connectivity with critical systems**	**Potential impact**	**Imp. level**	
[256] Aurora-like attack to smart grid generation system (2013, PoC)	1. Penetrate communication interface of a relay modem 2. Change relay settings to support remote trip/reclose 3. Eavesdrop MODBUS to learn function codes 4. Send trip and reclose commands every 0.25sec	* Exposed communication interfaces * Weak authentication * No encryption	Security testing of power generators based on a coordinated Aurora-like attack	[Outsider, Unpriv.]	[Moderate, Expert]	Strong	**Medium**	[Moderate, Major]	[Major, Minor]	**High**	**Direct:** The power generators are part of the smart grid	Coordinated attacks may cause widespread damage to many power generators that could take months to recover	**High**	**High**
[160] Attack against Ukraine's smart grid regional transmission system (2015, Real)	1. Use spear-phishing to steal credentials of IT servers 2. Use credentials and connect to SCADA through a VPN 3. Command remotely through an HMI and trip breakers	* Weak A/V protection * Use of 0-day exploit * OS vulnerabilities * Unsegmented network * Exposure of ICS SW	Nation state adversaries target Ukraine's smart grid	[Outsider, Unpriv.]	[Moderate, Expert]	Strong	**Medium**	[Moderate, Major]	[Major, Minor]	**High**	**Indirect:** The adversaries exploited corporate servers in order to attack to smart grid's CI	Three distributors were attacked resulting in several outages for a few hours affecting almost 230,000 consumers	**High**	**High**
[104] Stealth attack against Ukraine's smart grid transmission station (2016, Real)	1. Send spear-phishing emails to steal PC credentials 2. Propagate to the ICS network and gain access to RTUs/PLCs 3. Use embedded sophisticated modules to control RTUs/PLCs and cut off electric power.	* Weak A/V protection * Use of 0-day exploit * OS vulnerabilities * Unsegmented network	Nation state adversaries target Ukraine's smart grid	[Outsider, Unpriv.]	[Moderate, Expert]	Strong	**Medium**	[Major, Major]	[Major, Minor]	**High**	**Indirect:** The adversaries exploited PCs that were indirectly connected to smart grid's mission critical systems	The adversaries caused disruption of CI services, causing financial, public confidence loss	**High**	**High**
[154] FBI's investigation on Puerto Rico's utility AMI (2010, Real)	1. Connect an IR-equipped laptop with smart meter (required SW can be downloaded from the Internet) 2. Change settings for recording power consumption.	* Exposed communication interfaces * Easy to guess credentials * Weak authentication	Adversaries target numerous smart meters devices for financial profit (fraud)	[Insider, Unpriv.]	[Moderate, Intermed.]	Moderate	**Medium**	[Major, Major]	[Major, Minor]	**High**	**Direct:** The meters are part of the smart grid	The adversaries caused significant financial ($400M) loss	**High**	**High**
[254] Command injection attacks on vulnerable solar panel meters (2016 Real)	1. Locate vulnerable smart meters through Shodan 2. Exploit PHP vulnerability 3. Modify meter's parameters 4. Read/manipulate metering data	* Direct Internet connection * SW vulnerabilities (PHP script allowed remote code execution) * Use of hard-coded passwords	A cyber criminal hacks many meters to modify power levels reported to the grid	[Outsider, Unpriv.]	[Basic, Novice]	Moderate	**High**	[Moderate, Major]	[Major, Minor]	**High**	**Direct:** The meters send modified data to the smart grid	The adversary can cause some financial loss to the grid operator	**Low**	**Medium**
[37] Attacks on solar panel management systems (2016, PoC)	1. Locate connected devices through Wigle engine 2. Spoof unencrypted communications (use of plain HTTP) 3. Connect and control the device over the Internet.	* Unencrypted network protocols * Weak passwords * Exposed physical interfaces (Uboot, Console)	A hacker exploits many Internet-facing solar panel systems to attack the smart grid	[Outsider, Unpriv.]	[Basic, Intermed.]	Moderate	**High**	[Moderate, Major]	[Major, Minor]	**High**	**Direct:** The devices are part of the smart grid	The adversary can disrupt the smart grid and cause some financial loss	**Low**	**Medium**
[274] Attacks on wind and solar power systems (2015, PoC)	1. Locate vulnerable devices through Shodan 2. Recover a plaintext file with credentials 3. Authenticate and modify settings	* Direct Internet connection * Unsecure password storage * Application-layer vulnerabilities (*e.g.* CSRF)	Financially motivated adversaries target renewable smart grid energy systems	[Outsider, Unpriv.]	[Basic, Intermed.]	Moderate	**High**	[Minor, Major]	[Major, Moderate]	**High**	**Direct:** The devices are part of the smart grid	The adversaries can disrupt the smart grid and cause moderate financial loss	**Medium**	**High**
[269] FDI Attacks on real-time market model and state estimation systems (2017, PoC)	1. Remotely exploit vulnerabilities on AMI and sensor network 2. Introduce falsified data to the Market Management System 3. Purchase and sell virtual power for specific nodes to gain profit	* Direct/indirect Internet connection * Vulnerabilities on smart meters, AMI and sensor networks * Vulnerabilities on real-time market model and state estimation systems	Financially motivated adversaries target smart grid's real-time market model for profit	[Outsider, Unpriv.]	[Moderate, Expert]	Moderate	**Medium**	[Moderate, Major]	[Major, Moderate]	**High**	**Direct:** Real-time market model and state estimation systems are directly connected to the smart grid	The adversaries can disrupt the smart grid's operation and cause substantial financial loss	**High**	**High**

7.3 Attacks on smart cars and road traffic control infrastructure

Attack description and attack scenarios				Threat assessment				Vulnerability assessment			Impact assessment			Criticality
Description (Year,Type)	**Attack vector**	**Weaknesses found (* exploited)**	**Attack scenario**	**Access** [Physical, Logical]	**Capabilities** [Resources, Tech.Skills]	**Motiv.**	**Thr. level**	**Embedded** [H/W, S/W]	**Network** [Protocols, Key Manag.]	**Vuln. level**	**Connectivity with critical systems**	**Potential impact**	**Imp. level**	
[188] Connect to car's LAN from small distance using low cost transmitter (2015, PoC)	1. Exploit network flaws to connect to car's W-LAN 2. Connect to the CAN bus 3. Reverse engineer CAN S/W to control several systems	* Wireless protocol flaws (WiFi, Bluetooth, cellular) * Unauthenticated CAN access * CAN's flat architecture * Reversible CAN S/W	A nearby adversary takes control of a target vehicle (no physical access is required)	[Insider, Unpriv.]	[Basic, Novice]	Moderate	**High**	[Major, Major]	[Major, Minor]	**High**	**Indirect:** The car's WLAN is indirectly connected to the control systems of the car (*e.g.* though the CAN)	Attack control systems of the car (*e.g.* start, lock, breaks) to cause human injuries	**Medium**	**Medium**
[279] Take control of cars by sending crafted Digital Audio Broadcasting (DAB) signals (2015, PoC)	1. Create a bogus radio station 2. Send crafted DAB data to compromise the infotainment system 3. Control various CAN critical systems through the infotainment	* Vulnerable infotainment S/W * CAN's flat architecture (infotainment connected to CAN) * Unsanitized DAB signals * Unauthenticated CAN access	An adversary creates a bogus radio station and concurrently attacks vulnerable cars in range (the radio of target cars must be adjusted to receive signals)	[Outsider, Unpriv.]	[Basic, Expert]	Strong	**Medium**	[Major, Major]	[Major, Minor]	**High**	**Indirect:** The car's digital radio (DAB) is indirectly connected to the control systems of the car (*e.g.* though the CAN)	An adversary may cause accidents in the range covered by the bogus radio station, by disabling critical systems of the affected cars	**High**	**High**
[203] Exploit the WiFi connection between a car and its mobile control app (2016, PoC)	1. Crack the Wi-Fi preshared key 2. Sniff messages sent by the mobile app through the WiFi 3. Decrypt and get old commands 3. Inject old commands to control car's systems	* Predicable WiFi password * Network vulnerabilities (control app to car connection) * Unauthenticated CAN access * WiFi SSID allows geolocation	A nearby adversary takes control of a target vehicle	[Insider, Priv.]	[Basic, Intermed.]	Strong	**Medium**	[N/A, Expl]	[Expl, NP]	**High**	**Indirect:** The car's WiFi is indirectly connected to the control systems of the car (*e.g.* though the CAN)	Attack various systems of the car to cause user discomfort or human injuries	**Medium**	**Medium**
[193] Control cars through the Internet by abusing the infotainment system (2015, PoC)	1. Connect to target's IP port 6667 (open in a network provider) 2. Exploit the OMAP chip of head unit and enable SSH and CLI 3. Exploit the infotainment system to flash modified CAN firmware 4. Control the car through the CLI	* Exposure of D-Bus through cellular/WiFi * Command injection in D-Bus * Reversible CAN firmware * Unprotected update process of the infotainment system * CAN's flat architec-	(a) A remote adversary takes control of a car having physical access to a critical facility	[Outsider, Unpriv.]	[Moderate, Expert]	Strong	**Medium**	[Major, Moderate]	[Major, Minor]	**High**	**Indirect:** The compromised vehicle in not the actual target but has access to a critical facility	Gain an initial entry point (e.g. to a WiFi net) in a physically secured facility and use it to pivot to a CI	**Medium**	**Medium**
			(b) An adversary concurrently attacks many vulnerable cars from the Internet	[Outsider, Unpriv.]	[Moderate, Expert]	Strong	**Medium**	[Major, Moderate]	[Major, Minor]	**High**	**Indirect:** The infotainment is indirectly connected to the control systems of the car	Cause multiple car accidents through the Internet and harm people safety	**High**	**High**
[217] Remote attacks against camera and Light Detection and Raging (LiDAR) system (2015, PoC)	1. Blind the car's camera with a laser to confuse relative controls 2. Replay spoofed signals 3. Produced fake artifacts to confuse the LiDAR	* H/W vulnerabilities of LiDAR (pulse period and modulation, no use of redundancy) * H/W vulnerabilities of the camera (improper lenses, optical filters, no redundancy)	An adversary places the laser equipment in roads to "blind" the cameras of passing cars	[Outsider, Unpriv.]	[Moderate, Intermed.]	Strong	**Medium**	[Major, Minor]	[Moderate, Minor]	**Medium**	**Direct:** The attacked cameras are part of the car's mission critical systems	By causing multiple accidents in selected roads an adversary may disrupt the traffic in critical transportation infrastructures	**High**	**Medium**
[294] Contactless attacks against popular sensors used in Advanced Driver Assistance System (ADAS) (2016, PoC)	1. Jamm the ultrasonic sensors 2. Spoof the sensors to display fake pseudo-obstacles 3. Blind the laser on medium range vehicle's radars	* H/W, S/W flaws on sensors * H/W vulnerabilities of the camera (lenses, optical filters, noise reduction) * ADAS S/W does not distinguish spoofed signals	An adversary launches attacks against vehicles moving in high congestion roads in dense populated areas	[Outsider, Unpriv.]	[Moderate, Intermediate]	Strong	**Medium**	[Major, Major]	[Minor, Minor]	**Medium**	**Direct:** A vulnerable ADAS system may provide false data to other systems used in V2I and V2V communications like the ACC	Attacked cars that propagate false data to nearby vehicles and sideways transport infrastructures can disrupt the traffic or cause accidents	**High**	**Medium**
[45] Attacks on US IoT-enabled traffic control systems (DoS, bricking, flooding, spoofing) (2014,Real)	1. Create portable access point 2. Sniff and analyze wireless communications 3. Create self-spreading firmware 4. Update and remotely control sensors/repeaters	* Insecure wireless network (no encryption/ authentication) * Firmware updates allowed without authentication	Vulnerable traffic control systems are actually deployed in major cities all over the world	[Insider, Unpriv.]	[Basic, Intermed.]	Strong	**Medium**	[Minor, Major]	[Major, Major]	**High**	**Direct:** The traffic control system is part of the transportation infrastructure)	An adversary may disrupt multiple CI services (traffic jams), cause human fatalities (road accidents) and major economic loss	**High**	**High**
[100] Remote at- tacks on IoT-enabled traffic control systems (2014, PoC)	1. Use radio equipment to communicate with traffic controllers 2. Passively eavesdrop the network(s) (900 MHz and 5.8 GHz) 3. Analyze message structure 4. Inject commands to remotely control traffic lights	* Traffic controllers exposed to known network vulnerabilities * Insecure wireless network (no encryption/ authentication) * Lack of physical security	The attack is targeted to traffic control systems placed in critical roads (*e.g.* of high traffic)	[Insider, Unpriv.]	[Basic, Intermed.]	Strong	**Medium**	[Major, Moderate]	[Major, Major]	**High**	**Direct:** The traffic control systems are directly connected to critical transportation infrastructures	An adversary may concurrently attack multiple control systems to cause DoS attack on critical roads, or cause car accidents	**High**	**High**

7.4 Attacks on other intelligent transportation systems

Attack description and attack scenarios				Threat assessment				Vulnerability assessment			Impact assessment			Criticality
Description (Year,Type)	**Attack vector**	**Weaknesses found (* exploited)**	**Attack scenario**	**Access** [Physical, Logical]	**Capabilities** [Resources, Tech.Skills]	**Motiv.**	**Thr. level**	**Embedded** [H/W, S/W]	**Network** [Protocols, Key Manag.]	**Vuln. level**	**Connectivity with critical systems**	**Potential impact**	**Imp. level**	
[107] A 3-year assessment triggered by train companies on SCADA systems used in train control systems (2015, real)	1. Compromise physical security 2. Bypass authentication 3. Compromise outdated systems 4. Locate and attack train communication systems from the Internet (Shodan) 5. Exploit cellular network and entertainment system flaws	* Weak authentication * Lack of encryption/ integrity * Lack of physical security * Outdated S/W * Keys hardcoded in firmware * Vulnerable cellular modems * Flat network architecture	Attacks on IoT-enabled mission critical railway systems from a group of terrorists / hostile nation	[Outsider, Unpriv.]	[Moderate, Expert.]	Strong	**Medium**	[Major, Major]	[Major, Major]	**High**	**Indirect:**The IoT-enabled devices are indirectly connected to mission critical railway systems	Compromised rail- way control systems may be used to cause train collisions, leading to human fatalities as well as economic, public trust and confidence loss	**High**	**High**
[51] Compromise public information systems to mani-pulate passengers' (2015, PoC)	1. Compromise PID server 2. Compromise PID system 3. Send spoofed PID messages to mobile app to overcrowd platforms	* Web vulnerabilities of PID server * Weak authentication * Lack of encryption/ integrity * Lack of physical security	PIDs compromised by terrorists to control passengers' behavior	[Outsider, Priv.]	[Moderate, Intermed.]	Strong	**Medium**	[Minor, Moderate]	[Major, Major]	**High**	**No connectivity:** The PID server is not connected to mission critical systems	Compromised PID systems may be used to amplify the human casualties in a combined cyber-physical attack	**High**	**High**
[61, 144, 272] Demo attacks against plane ADS-B systems (2012-15, PoC)	(Description based on [272]) 1. Build a simulation environment with off-the-shelf H/W 2. Eavesdropp communications 3. Use a mobile app to inject commands and take control of the FMS	* Unencrypted, unauthenticated communications * Unauthenticated commands allow command injection	Terrorists gain control of airplanes in transit by exploiting ADS-B systems	[Outsider, Unpriv.]	[Moderate, Expert]	Strong	**Medium**	[Major, Moderate]	[Major, Minor]	**High**	**Direct:**The ADS-B system is connected to the aircraft's mission critical systems	A plane takeover attack could lead to human fatalities as well as economic, environmental and public trust and confidence loss	**High**	**High**
[239] Demo attacks against IFE system (2016, PoC, based on real data)	1. Reverse engineer firmware files of the Panasonic IFE system 2. Extract hardcoded credentials 3. Perform SQL injection 4. Control how passengers aboard are informed	* Publicy available firmware update files and source code * Easy to reverse binaries * Sensitive hardcoded data (*e.g.* credentials, databases etc) * SQL injection vulnerabilities	Terrorists gain control of mission critical systems of airplanes in transit by exploiting vulnerabilities of the IFE system	[Insider, Unpriv.]	[Moderate, Expert]	Strong	**Medium**	[Moderate, Major]	[Major, Major]	**High**	**Indirect:**The IFE system is connected indirectly to the aircraft's mission critical systems	A plane takeover attack could lead to human fatalities as well as economic, environmental, public trust and confidence loss	**High**	**High**
[21] Demo attacks on AIS vessel tracking system (2013, PoC)	1. Use off-the-shelf H/W to transmit AIVDM messages 2. Inject spoofed messages in the AIS network (Man-In-The-Water, CPA alerting, signal jamming etc) 3. Force ship to follow a path	* Weak authentication * AIVDM messages (received data from other vessels) susceptible to tampering * Lack of integrity controls on data context	Pirates introduce fake AIVDM messages to lead a cargo tanker to shallow waters and/or render it invisible	[Insider, Unpriv.]	[Basic, Intermed.]	Strong	**Medium**	[Minor, Moderate]	[Major, Minor]	**Medium**	**Direct:** The AIS system is directly connected to mission critical systems of a maritime vessel	The attackers can cause human injuries/fatalities as well as major economic and environmental loss	**High**	**Medium**
[22] Remote attacks on a ship's mission-critical systems using vulnerabilities found on *AmosConnect* server (2017, PoC)	1. Search Shodan for exposed web interfaces of vulnerable systems 2. Recover privileged backdoor account and execute commands with system privileges on the remote system 3. Pivot to other segments of the ship's network and locate and takeover ship's mission-critical systems.	* No/weak authentication mechanisms * Vulnerable web interfaces * Exposure of sensitive data * Lack of network segmentation	Cyber criminals take over ship's navigation systems remotely for ransomware	[Outsider, Unpriv.]	[Intermed, Intermed.]	Strong	**High**	[Minor, Major]	[Major, Major]	**High**	**Indirect:** AmosConnect system is indirectly connected to the ship's mission critical systems	The attackers can cause human injuries/fatalities as well as major economic and environmental loss	**High**	**High**
[28] Attacks on a container port's Internet-connected systems and devices (TOS - OCR - RFID) (2017, PoC)	1. Use spear phishing techniques gain access to port's internal network 2. Locate vulnerable systems and devices 3. Exploit network and software vulnerabilities to infect the devices	* No network segmentation/isolation * Vulnerable network protocols * Vulnerable/Outdated OS installed * Lack of security mechanisms (*e.g.* authentication)	Terrorists infiltrate port's internal networks and infect/remotely control port's OCR - GPS and RFIDs systems in order to smuggle weapons	[Out, Non-priv]	[Moderate, Expert]	Strong	**Medium**	[Minor, Major]	[Major, Major]	**High**	**Indirect:** The infected systems and devices are indirectly connected to the Internet through the company's corporate network	In this scenario, the adversaries can harm human lives and cause substantial economic, public trust and confidence loss	**High**	**High**

7.5 Attacks on IoT-enabled healthcare infrastructure, services and devices

Attack description and attack scenarios				Threat assessment				Vulnerability assessment			Impact assessment			Criticality
Description (Year,Type)	**Attack vector**	**Weaknesses found (* exploited)**	**Attack scenario**	**Access** [Physical, Logical]	**Capabilities** [Resources, Tech.Skills]	**Motiv.**	**Thr. level**	**Embedded** [H/W, S/W]	**Network** [Protocols, Key Manag.]	**Vuln. level**	**Connectivity with critical systems**	**Potential impact**	**Imp. level**	
[120, 182] Nearby attacks on pacemakers and IMDs (2008 and 2016, PoC)	1. Reverse-engineer RF communications of the ICD device 2. Intercept patient telemetry data 3. Inject data to modify therapy	* RF modulation scheme * Lack of encryption * Lack of authentication	An adversary tampers an ICD from short distance (5m)	[Insider, Unpriv.]	[Basic, Expert]	Strong	**Medium**	[Major, Moderate]	[Major, Major]	**High**	**Direct:** The device can directly affect the safety of the patient	The attack may harm a target patient from a short (< 5m) distance	**Medium**	**Medium**
[224] Remote attacks on insulin pumps and CGMs (2011, PoC)	1. Intercept RF traffic 2. Exploit configuration S/W 3. Demodulate captured traffic	* Lack of encryption * Lack of authentication * Easy to reverse S/W	An adversary tampers a CGM device from a medium (60m) distance	[Outsider, Unpriv.]	[Basic, Expert]	Strong	**High**	[Minor, Major]	[Major, Major]	**High**	**Direct:** The device can directly affect the safety of the patient	The attack may harm a target patient from a medium (< 60m) distance	**Medium**	**High**
[31] Security evaluation of patient home network devices (2017, PoC)	(Attack based on findings of [31]) 1. Extract F/W using an available (embedded) interface 2. Reverse-engineer F/W 3. Recover hardcoded credentials 4 Upload custom F/W through OTA update to control the device	* Debugging interfaces * No F/W protection (obfuscation, encryption) * Hard-coded credentials * 3rd-party S/W flaws * ASCII function names * Unencrypted stored data	An adversary remotely controls a vulnerable home monitoring device	[Outsider, Unpriv.]	[Basic, Intermed.]	Strong	**High**	[Major, Major]	[Major, Major]	**High**	**Direct:** The home monitoring devices can directly affect the safety of the patient	The attack may cause loss of public confidence for e-health services or even threat human lives	**High**	**High**
[277] A security assessment in three major hospitals based on emulated Virtual Medical Devices, revealed actual stealth attacks (2017, real)	1. Introduce a repackaged old worm to the hospital's internal network (*e.g.* through phishing) 2. The worm commences stealth attacks to control medical devices (*e.g.* radiation oncology system) 3. Self-propagate to other networks 4. Remotely control compromised medical devices	* Outdated OS * Unsegmented networks * Lack of endpoint A/V protection	Financial criminals target healthcare industry to extract sensitive data and/or install ransomware	[Outsider, Unpriv.]	[Moderate, Expert]	Strong	**High**	[Major, Major]	[Major, Moderate]	**High**	**Indirect:** The vulnerable medical devices and systems are indirectly connected to the targeted IT support systems	The attack can cause major economic, reputation, and privacy loss	**High**	**High**
[131] Compromise in-hospital medical devices using web applications (2016, PoC)	1. Compromise hospital's web server to enter the internal network 2. Scan internal network to discover medical devices 3. Compromise vulnerable PMDs 4. Create / disable alarms, display incorrect vital information	* Web vulnerabilities * Unsegmented networks * Weak authentication mechanisms in PMDs	Nation state adversaries attack several hospitals medical equipment	[Outsider, Unpriv.]	[Basic, Intermed.]	Strong	**High**	[Major, Major]	[Minor, Minor]	**High**	**Indirect:** The exploited PMDs were connected to the vulnerable web server	The attack may cause loss of public confidence for e-health services or even threat human lives	**High**	**High**
[131] Use a vulnerable smart information kiosk to control patient's medical station (2016, PoC)	1. Bypass physical security 2. Exploit a kiosk device to access the internal network. 3. Compromise patient's station 4. View and modify treatment data	* No physical security * Vulnerable kiosk device * Unsegmented network	Terrorists attack a large healthcare facility	[Ins, Unpriv.]	[Basic, Intermed.]	Strong	**Medium**	[Major, Major]	[Major, Moderate]	**High**	**Indirect:** The vulnerable IoT devices are indirectly connected to hospital's critical systems	The attack may lead to implementing an inappropriate treatment and affect patient's health condition	**High**	**Medium**

7.6 Attacks on smart home/building systems installed in CIs

Attack description and attack scenarios				Threat assessment				Vulnerability assessment			Impact assessment			Criticality
Description (Year,Type)	**Attack vector**	**Weaknesses found (* exploited)**	**Attack scenario**	**Access** [Physical, Logical]	**Capabilities** [Resources, Tech.Skills]	**Motiv.**	**Thr. level**	**Embedded** [H/W, S/W]	**Network** [Protocols, Key Manag.]	**Vuln. level**	**Connectivity with critical systems**	**Potential impact**	**Imp. level**	
[49] Attack on LIFX smart light bulbs to gain unauthorized WiFi access (2014, PoC)	1. Extract firmware (F/W) from a smart lamp 2. Reverse engineer F/W 3. Extract embedded crypto key (common in all devices) 4. Use the extracted key to decrypt WiFi password of a target device	* H/W easy to tamper * Reversible F/W * Common key embedded in all devices * Unauthenticated, unencrypted commands	Vulnerable smart lights installed in a critical facility (*e.g.* hospital)	[Outsider, Priv.]	[Low, Intermed.]	Strong	**High**	[Major, Moderate]	[Minor, Major]	**High**	**No connectivity:** The smart lighting system is not connected with any critical system but it is WiFi connected	An adversary may use this attack against vulnerable devices to gain an initial access to point (*e.g.* through the WiFi)	**High**	**High**
[95] Hacking a hotel's smart automation systems (2016, real)	1. Connect to hotel's internal network through exposed interface. 2. Monitor all network traffic (Modbus over TCP) 3. Use open-source S/W to access other control systems connected to the Modbus	* Exposed interfaces * Unsegmented network * Lack of network security mechanisms	The smart automation systems are installed in a hotel	[Insider, Unpriv.]	[Low, Intermed.]	Moderate	**Medium**	[Major, Minor]	[Major, Major]	**High**	**Direct:** Directly connected to hotel's safety critical control systems (a/c, water heating, elevators)	An adversary could harm the safety of other residents (water heating), cause user discomfort (elevators) or privacy loss (know if residents are inside)	**Medium**	**Medium**
[124] Attack on a smart Nest thermostat could indirectly affect IT systems' operation (2014, PoC)	1. Initiate a reset of the device 2. Connect USB boot device 3. Execute / install custom kernel and backdoor 4. Remotely control the device	* Easily accessible embedded communication interfaces * Lack of security mechanisms during booting process	A malicious insider attacks a vulnerable thermostat installed in the data center	[Insider, Unpriv.]	[Moderate, Intermed]	Strong	**Medium**	[Major, Major]	[Moderate, Minor]	**Medium**	**No connectivity:** The devices are isolated from other systems but are installed in a data center	In this scenario, the adversary can disrupt mission critical systems and services, for example by causing servers to overheat and shutdown	**Medium**	**Medium**
[286] Use vulnerable smart TVs to create a covert audio channel (2017, Real)	1. Exploit known and/or 0-day S/W vulnerabilities 2. Remote control the device 3. Modify device's characteristics to enable the microphone in a fake-off mode and create a covert audio channel	* S/W vulnerabilities * Use of unsigned F/W updates * Use of insecure (plain http) communication to control the device	Nation state adversaries target smart TVs installed inside highly secure building in order to exfiltrate data and/or spy on selected individuals	[Outsider, Unpriv.]	[High, Expert]	Strong	**Medium**	[Moderate, Major]	[Major, Moderate]	**High**	**No connectivity:** The smart TV is isolated from other systems but it is Internet-enabled and is inside a top secret premise	Nation state adversaries may spy a highly secure facility (*e.g.* Government)	**High**	**High**
[232] Extend the functionality of smart light bulbs to manipulate flickering and: (a) create a covert channel (b) cause epileptic seizures (2016, PoC)	1. Manipulate API to inject customized commands 2. Modify the PWM signals to control flickering (unnoticeable to human eye) 3. Remotely receive flickering changes with a light sensor (scenario a), or 4. Modify flickering to a specific range (scenario b)	* No encryption or integrity check for API commands * Input not sanitized * Lack of network security mechanisms Use of unencrypted WiFi password	Installed in top-secret facility (*e.g.* military)	[Insider, Unpriv.]	[Low, Intermed.]	Strong	**Medium**	[Minor, Major]	[Major, Major]	**High**	**Indirect:** The smart lighting system is indirectly connected to a critical system that is installed nearby	An insider may use the covert channel to exfiltrate highly sensitive data, without being detected by any security system	**High**	**High**
			Installed in a public, crowded place (*e.g.* metro station)	[Outsider, Unpriv.]	[Low, Intermed.]	Strong	**Medium**	[Minor, Major]	[Major, Major]	**High**	**No connectivity:** The smart lighting system is not connected to any critical system	An adversary may abuse flickering to cause epileptic seizures in crowded places and affect people safety	**High**	**High**

7.7 Attacks based on IoT devices installed in non-critical environments

Attack description and attack scenarios				Threat assessment				Vulnerability assessment			Impact assessment			Criticality
Description (Year,Type)	Attack vector	Weaknesses found (* exploited)	Attack scenario	Access [Physical, Logical]	Capabilities [Resources, Tech.Skills]	Motiv.	Thr. level	Embedded [H/W, S/W]	Network [Protocols, Key Manag.]	Vuln. level	Connectivity with critical systems	Potential impact	Imp. level	
[58, 111] An DDoS attack against DNS servers based on home IoT devices (2016, Real)	1. Exploit vulnerabilities on home IoT-enabled devices. 2. Control the devices remotely. 3. Launch DDoS attack against the DYN's DNS servers.	* Weak / default passwords in IoT-enabled devices such as routers, cameras and DVRs * Characteristics of the DNS protocol	The attack targeted the DYN DNS servers	[Outsider, Unpriv.]	[Moderate, Expert]	Strong	**High**	[Minor, Major]	[Major, Major]	**High**	**No connectivity:** The devices are not connected, even indirectly, to the actual target	The attack caused disruption of service to more than 1200 domains, including major sites (Amazon, PayPal and others) in same cases for several hours	**Medium**	**High**
[67, 271] Attacks on WeMo smart home devices, smart apps and platforms (2015/6, PoC)	1. Locate vulnerable devices and services through a search engine (*e.g.* Shodan) 2. Remotely execute commands through SQL injection to get root access 3. Remotely control the device	* Lack of input sanitation * Exposed vulnerable interfaces to the Internet	Cybecriminals may target against vulnerable WeMo smart home devices to remotely control them and create a botnet	[Out, Non-priv]	[Moderate, Intermed.]	Moderate	**Medium**	[Minor, Major]	[Moderate, Moderate]	**Medium**	**No connectivity:** The devices are not connected, even indirectly, to the actual target	In this scenario, the adversary may use compromised devices as part of a botnet to attack a critical service (DDoS attack)	**Medium**	**Medium**
[233] Take control of smart lights from distance, using a custom self-propagating firmware (2016, PoC)	1. Use CPA/DPA analysis to retrieve embedded H/W key 2. Create self-propagating firmware 3. Sign firmware with H/W key 4. Bypass proximity check and force a device to join a network 5. Replace firmware from distance (wardriving, warflying) 6. Self-propagate to massively takeover smart lamp devices	* Common embedded key * H/W easy to tamper * Atmel's "leaky" H/W AES engine * Atmel's proximity check bug Lack of PKC in 802.15.4x networks Leaked master ZLL key	In this scenario vulnerable smart lights are widely installed in non critical places (homes, offices, public places) in a densely populated area	[Outsider, Unpriv.]	[Low, Expert]	Strong	**High**	[Major, Major]	[Major, Major]	**High**	**No connectivity:** The actual target of the attack is the IoT device itself, but in large numbers	Since the attack can be triggered from distance (up to 350m) and is self-propagating, if vulnerable lights are densely installed it may lead to a massive take-over of lighting systems (PDoS and/or ranswomware attacks)	**Medium**	**High**
[84] Attacks on IoT devices based on control app vulnerabilities (2016 PoC)	1. Obtain embedded in the SmartApp client Id and secret key 2. Replace part of the OAuth token with the attacker's domain 3. Invite the victim to a crafted link 4. Inject commands to the WebService SmartApp and plant a backdoor (persistence) 5. Remotely control the lock mechanism of the door	* Overprivileged SmartApp(s) * Unsanitized input strings * Hardcoded credentials * Lack of encryption	Criminals exploit vulnerabilities in SmartApps to break into houses	[Outsider, Priv.]	[Moderate, Intermed.]	Medium	**Medium**	[Minor, Major]	[Major, Major]	**High**	**No connectivity:** The actual target of the attack is the IoT device itself, but in large numbers	By remotely controlling many vulnerable smart locks, the adversary can violate the physical access to all facilities that can be controlled from the vulnerable smart app	**Medium**	**Medium**
[242] Attacks on smart TVs (2017, PoC)	1. Create crafted TV signal 2. Use a drone (warflying) and off-the-shelf equipment to propagate the signal in a densely populated area 3. Take over the vulnerable smart TVs	* Vulnerable to command injection background apps * No sanitation of input data	An adversary uses this attack to install ransomware in all vulnerable TVs in range	[Outsider, Unpriv.]	[Moderate, Intermed.]	Strong	**High**	[Minor, Major]	[Major, Minor]	**Medium**	**No connectivity:** The actual target of the attack is the IoT device itself, but in large numbers	In this scenario, the adversary may concurrently install ransomware to all vulnerable TVs in range and cause economic loss to users and loss of confidence to the manufacturer	**Low**	**Medium**
[198] Exploiting ZigBee prototocol to enable attacks like DoS, hijack and command injection (2016, PoC)	1. Insert spoofed ACK messages to inter-PAN frames 2. Use (leaked) ZLL master key to inject customized commands	* Unauthenticated inter-PAN frames * Use of a common ZLL master key	Installed in non-critical facilities (*e.g.* homes) but in large numbers	[Outsider, Unpriv.]	[Low, Intermed.]	Strong	**High**	[Moderate, Minor]	[Major, Major]	**Medium**	**No connectivity:** The actual target of the attack is the IoT device itself, but in large numbers	An adversary may launch massive DoS attacks and cause user discomfort	**Low**	**Medium**

7.8 Result analysis

From the results presented in Tables 7.1-7.7, we can infer that industrial SCADA and Smart Grids favor the direct connectivity attack path scenarios, since modern field devices provide Web interfaces for remote monitoring and control [90, 136, 223, 254, 256]. However, indirect attack paths may also occur. Since SCADA command and control centers can interact with corporate networks, attacks such as spear phishing [104, 160] have also been realized. In that case, IoT connectivity of field devices may be used as pivoting points, in order to attack mission critical systems [81, 156].

Indirect IoT-enabled attacks are more common in both healthcare and intelligent transportation systems. In the case of smart transportation, vulnerable on-board entertainment, informational and communication systems may enable an adversary to indirectly control mission critical functions [22,107,193,203,239]. Similarly, outdated, interconnected, passive medical devices [131, 277] can be used to attack a hospital's mission critical systems that process valuable data. Direct attacks against medical devices, may also have severe impact, since they may directly affect patients' safety [31].

Smart home automation devices are primary used in no-connectivity attack scenarios. Due to their proliferation and their low security level, such devices are usually easy to compromise. In many cases they have been used by botnets in order to amplify DDoS attacks against critical targets that are not connected, even indirectly, with the IoT devices (e.g. [58, 111, 232, 271]). In other cases home IoT devices may also serve as the actual target of the attack (e.g. ransomware attacks [84, 198, 242]). Finally, smart automation devices, that are installed inside the premises of critical infrastructures, can also be used to indirectly attack their nearby critical systems [49] or even to exfiltrate sensitive data from nearby systems [232].

Interestingly, in all the attacks examined in this paper and regardless of the examined sector, the success of the attack relied in one or more of the following characteristics: (i) The physical proximity of the IoT device with the target, (ii) the exploitation of its communication interfaces (physical or network) and (iii) the extended, and usually unexpected, extension of the functionality provided by the IoT device.

CHAPTER 8
SMART CITY VALIDATION SCENARIO

8.1 Assessing IoT-enabled, cyber-physical attack paths: Implementation tool overview

In order to validate the efficacy of our methodology presented in Chapter 6 an implementation tool was developed in python3, utilizing several libraries. Pandas dataframes were used to structure and analyze the required input and output data of the application. The AST library[1] was used in order to split complex input data from .csv files, so they can be inserted to lists and dataframes. For the vulnerabilities, the CVSS/CVSSlib library was used to calculate the base score (the exploitability and impact sub scores) of the interaction CVSS vectors. The CVEs were collected from the NIST database and were pulled from the json files, based on their CPE identifier. For the implementation of Algorithm 1, the interaction tuples were properly adjusted and extended to also include the network id and the interface id used by the source and destination nodes. This extension aims to raise the complexity of attack paths from n^2 to $n^2 \cdot n_i$, where n is the number of devices and n_i the number of interfaces per device. During the interaction assessment phase (Algorithm 2), rules for network connectivity, physical interactions and security controls were applied. Then, capability along with the CVE/CVSS vectors of the destination node of each interaction, were utilized, along with python libraries CVSSlib/CVSS, for the calculation of the highest scoring vector for each `AV`, the CVV score and the production of the $AssessedLists[i], i = 1, \ldots, n$ lists.

The attack path construction module (Algorithm 3) is an iterative procedure that takes as input the$AssessedLists[i]$, $i = 1, \ldots, n$, along with an extensive CVSS centric rule-set, in order to produce a structured $AttackPaths[i][]$, $i = 1, \ldots, n$ lists.

Finally for the attack path assessment, the CVV vector for each vulnerability `AV` is calculated by utilizing CVSS/CVSSlib[2], based on the available vulnerabilities on the source node of each attack path. The exploitability metrics of the produced vector are then checked with each attacker's capabilities and the physical and/or logical access of the each adversary's profile to the initial node.

[1] https://pub.dev/documentation/analyzer/latest/dart_ast_ast/dart_ast_ast-library.html

[2] https://pypi.org/project/cvsslib/

8.2 IoT-enabled cyberattacks on Smart City infrastructures

As described in Chapter 7, attack scenarios on Smart City environment can have significant impact on our everyday life. Security researchers such as Branden Ghena et. al. [100] presented vulnerabilities found in traffic lighting systems' controllers which enabled them to launch a series of attacks including DoS attacks, gain remote access and cause traffic congestion. Cerrudo in [46], describes security vulnerabilities and potential cyberattack scenarios on Smart City infrastructure, including cyber terrorism, nation state warfare, cybercrime ransomware campaigns and hacktivist movements. Researchers in [307] point out security and privacy issues in Smart City IoT-enabled systems such as privacy leakage in data sensing, privacy and availability in data storage and processing (e.g. cloud infrastructure) as well as control services dependencies and trustworthiness. In [9] researchers present potential attack vectors in Smart City environments. In particular, they define the main areas of a Smart City environment as follows: Governance, economy, people, mobility, living and environment. Then, researchers examine several, IoT-enabled attack vectors and their potential impact including public/private, Internet connected cameras, building management systems, transport management systems, including traffic lights and/or road electronic signs as well as communication networks (e.g. public WiFi spots). For all of the above attacks that may result in privacy attacks, traffic jams, injuries and/or fatalities in a large scale are examined. A comprehensive study of IoT cybersecurity in Smart Cities [11] pointed out as some of the most prominent attack vectors in IoT ecosystem such as the use of hardcoded weak credentials, lack of secure update mechanisms, deprecated software, lack of tamper-resistant hardware, insecure communication APIs and services as well as weak authentication and session management mechanisms.

In Section 3.8, we were able to present several vulnerabilities and misconfigurations in embedded software, cloud APIs, mobile application, and networks of a smart lighting system, that if successfully exploited, may lead to a variety of cyber or cyber-physical attack scenarios.

In order to demonstrate the risks that derive from smart lighting systems and test the efficacy of our methodology presented in Chapter 6, we present a simplistic, yet realistic PoC scenario. In particular, a smart lighting system that shares the same vulnerabilities presented in [262], is installed in popular domains of an urban environment. In Figure 8.1, we present potential installation domains and targets of such a scenario. Since we only focusing on risks that originate from the particular smart lighting system, we only consider the attack path scenarios that are either targeting directly the smart lighting systems or utilize them in order to propagate the attack to another critical system.

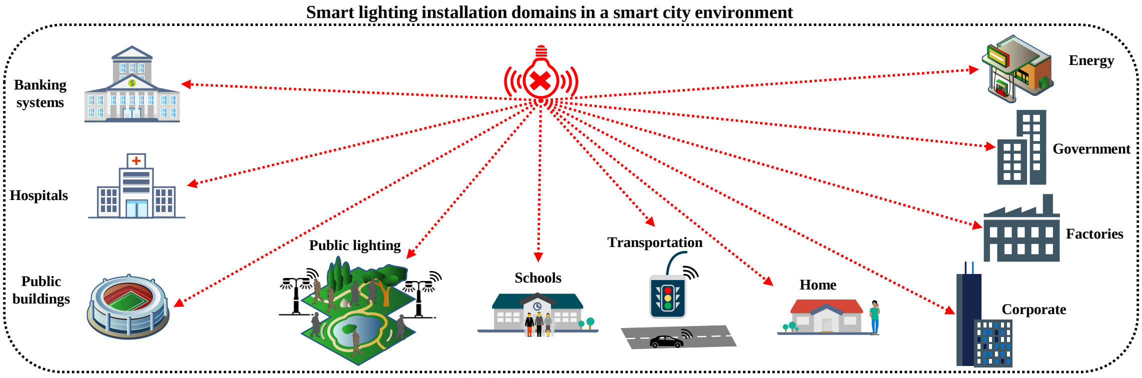


Figure 8.1: A paradigm of potential targets regarding smart lighting systems in a Smart City environment [262]

8.3 Methodology walkthrough

As defined in Chapter 6, the assessment of the cascading IoT-enabled risks is divided in four main phases: The *interaction modelling*, the *interaction assessment*, the *construction of the attack paths* and the *attack path assessment* phase.

During the first phase (see Section 8.3.1), the cyber and physical interactions between all the IoT/IT systems under assessment are identified, based on cyber and physical types of interactions (Tables 6.2 and 6.3). Then, throughout the second phase (Section 8.3.2), all the interactions from the previous phase are assessed in order to examine the level of the combined vulnerabilities found in each couple of interacting systems. The overall vulnerability level of an interaction is defined as the *Interaction Cumulative Vulnerability Vector* (CVV_{int}).

During the third phase (Section 8.3.3), the assessed interactions are utilized to produce attack paths constructed by combining the direct (*Level-1*) interactions in order to calculate two-hop (*Level-2*) attack paths which can be constructed from the assessed interactions (Tables 8.8 and 8.9 respectively). Then, the businesswise impact is calculated by taking into consideration the particularity of each environment (see in Table 8.10).

Finally in the fourth phase (Section 8.3.4), for each attack path the vulnerabilities of the point-of-entry device are examined and the corresponding attack scenarios are created. For each attack path, the vulnerabilities of the entry node are combined with the vulnerability level of all the interactions that exist in the attack path, to define the vulnerability level of the whole attack path, defined as $CVV\big(\mathcal{AP}, AV\big)$ in previous chapters. Note that an attack path may have a different cumulative vulnerability level for attacks of different `AV:N/A/L/P`. In order to define the threat level for each attack path, various threat agents are modeled and the corresponding characteristics of each applicable threat agent are matched with the exploitability characteristics of the newly created cumulative vulnerability vector for each attack vector, i.e. `AV=[N|A|L|P]`. Then, the risk is defined by using the risk calculation matrix Table 6.14.

8.3.1 Phase 1: Interaction Modelling

In order to identify all potential cyber and physical interactions between the target(s) and all the devices in scope one must first gather information such as inputs and outputs, network interfaces, moving parts (if any) with corresponding active ranges. Furthermore, the physical location, the connected networks with their cyber-physical characteristics and logical/physical access rules must also be defined for each device.

ID	Device	Physical		Network/Interface	R/F(*)	Internet Access
		Location	Type			
1	Smart lights	Bank	Internal	Net 1/WiFi/Zigbee	2.4 GHz	✓
2	E-banking servers	Bank	Protected	Net 2/Ethernet	N/A	x
3	Smart lights	Pharmaceutical	Internal	Net 4/WiFi/Zigbee	2.4 GHz	x
4	Corporate server	Pharmaceutical	Protected	Net 5/Ethernet	N/A	x
5	Smart lights	Governmnet	Internal	Net 6/WiFi/Zigbee	2.4 GHz	✓
6	Cloud ICT	Governmnet	Protected	Net 7/Ethernet	N/A	x
7	Smart lights	Home	Internal	Net 8/WiFi/Zigbee	2.4 GHz	✓
8	Workstation	Home	Internal	Net 8/WiFi	2.4 GHz	✓
9	Smart lights	Public Stadium	External	Net 9/WiFi/Zigbee	2.4 GHz	✓
10	Alarm System	Public Stadium	External	Net 9/WiFi	2.4 GHz	✓
11	Smart lights	Public Lighting	External	Net 10/WiFi/Zigbee	2.4 GHz	✓
12	Workstation	Bank	Internal	Net 3/Ethernet	N/A	x

(*): Radio Frequency

Table 8.1: Device list with corresponding physical and network characteristics [262]

NetID	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Int**
1	N/A	x	✓	x	x	x	x	x	x	x	✓
2	x	N/A	x	x	x	x	x	x	x	x	
3	x	✓	N/A	x	x	x	x	x	x	x	
4	x	x	x	N/A	✓	x	x	x	x	x	
5	x	x	x	x	N/A	x	x	x	x	x	
6	x	x	x	x	x	N/A	✓	x	x	x	✓
7	x	x	x	x	x	x	N/A	x	x	x	
8	x	x	x	x	x	x	x	N/A	x	x	✓
9	x	x	x	x	x	x	x	x	N/A	x	✓
10	x	x	x	x	x	x	x	x	x	N/A	✓

N/A: Not Apllicable, Int: Internet access

Table 8.2: Network access rules and Internet connectivity [262]

We consider several installation domains ranging from a home environment to public buildings/areas and corporate/government institutions. An overview of the PoC installation domains, devices, networks and physical location are depicted in Figure 8.2. In particular, it includes several scenarios were the smart lighting system is installed within the premises of the following domains:

1. A systemic monetary institution (e.g. banks): The smart lighting system is installed within the bank's premises and can communicate with the Internet (e.g. via UPnP), the administrators network, but cannot access the bank's server farm network directly.
2. A pharmaceutical company: The smart lighting system is installed within the company's premises, is isolated from the Internet, but can communicate with the server sub-network.
3. A government cloud infrastructure: The smart lighting system is installed within the building, can communicate with the Internet (e.g. via UPnP) and is indirectly connected to the G-Cloud central network.
4. A smart home environment: The smart lighting system can communicate with the Internet (e.g. via UPnP) and is installed in the same network with a mobile workstation that has admin access to G-Cloud infrastructure (remote working scenario).
5. A sports stadium facility: The smart lighting system can communicate with the Internet (e.g. via UPnP) and is installed in the same network with an vulnerable IoT-enabled alerting system.
6. Is a part of the public lighting infrastructure: The smart lighting system is massively installed to several public areas (streets, parks etc.) and is managed remotely (via the Internet).

Regarding network interfaces, we consider that all smart lighting systems that are connected to an internal network have Internet access via UPnP protocol with the exception of the pharmaceutical company environment. Devices' characteristics and network access are defined in Tables 8.1 and 8.2 respectively. Since we aim in showcasing risks that stem out from smart lighting systems, we only examine interactions of smart lighting systems and the predefined targets but not among the devices themselves (as in a full risk assessment scenario), with the exception where logical access between devices exist (interaction types C2, C3, C5 and C6). In addition, we consider that the user in the home environment is an administrator working remotely via VPN service to a government cloud infrastructure (interaction

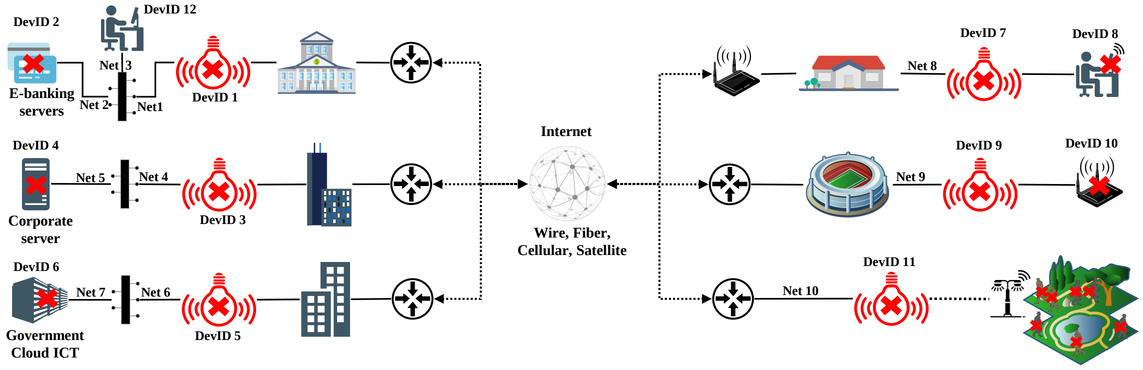


Figure 8.2: PoC scenario regarding devices, networks and physical location [262]

Source*	Target*	IntType**	Source Int	Target Int	Int level
1	Human	P2 (Disable)	Luminaire	Eye	1
1	Human	P2 (Flicker)	Luminaire	Eye/Reciever	1
1	12	C4	WiFi	Ethernet	1
12	2	C6	Ethernet	Ethernet	1
3	Human	P2 (Disable)	Luminaire	Eye	1
3	Human	P2 (Flicker)	Luminaire	Eye/Reciever	1
3	4	C4	WiFi	Ethernet	1
5	Human	P2 (Disable)	Luminaire	Eye	1
5	Human	P2 (Flicker)	Luminaire	Eye/Reciever	1
5	6	C4	WiFi	Ethernet	1
7	Human	P2 (Disable)	Luminaire	Eye	1
7	Human	P2 (Flicker)	Luminaire	Eye	1
8	6	C6	WiFi	Ethernet	1
9	Human	P2 (Disable)	Luminaire	Eye	1
9	Human	P2 (Flicker)	Luminaire	Eye	1
9	10	P3(Jamming)	WiFi	WiFi	1
9	10	P3(Jamming)	ZigBee	WiFi	1
9	10	C1	WiFi	WiFi	1
11	Human	P2 (Disable)	Luminaire	Eye	1
11	Human	P2 (Flicker)	Luminaire	Eye	1
1	12	C4	WiFi	Ethernet	2
7	8	C1	WiFi	WiFi	2

(*):Device ID → Table 8.1, (**): Interaction types → Tables 6.2 and 6.3

Table 8.3: PoC paradigm's interaction tuples [262]

type C6) similarly to a bank executive officer that administers the financial systems of the bank (device ID 12).

Based on the above scenarios and the cyber-physical interaction types defined in [257] (see Tables 6.2 and 6.3), one can now construct all potential interaction tuples (source and target devices or interfaces). The whole process utilizes a recursive algorithm that, in general, first constructs all the 'direct' cyber-physical interactions with the predefined targets and then all the cyber interactions between the devices in scope themselves. All the applicable interaction tuples are defined in Table 8.3.

8.3.2 Phase 2: Interaction Assessment

As defined in [257] an $IntCVSS_{base}$ vector represents the initial (base) capabilities of an adversary for each interaction. More specifically, by assuming that a device x interacts with a device y based on a cyber (C1-C6), or physical (P1-P3) interaction type, the corresponding $IntCVSS_{base}$ capability vector implies the capabilities that device x (or an adversary controlling x) has on device y, due to their interaction type (Tables 6.6 and 6.8). In the work [262], $IntCVSS_{base}$ vectors were constructed based on the discovered vulnerabilities/security misconfigurations of the smart lighting systems, in a CVSS-like structure (see Table 8.4). These vectors are properly modified to depict the minimum required capabilities of an adversary in order for the attack to be successful. The individual impact metrics CIA have been properly adjusted in order to depict the impact (severity) of the vulnerability/security misconfiguration on the device. This in turn, can be used to determine whether an attack may result in a full or partial compromisation of an IoT device, thus allowing the adversary to remotely control the device or use it in order to propagate to the actual target.

In order to assess the vulnerability level of an interaction tuple, the existing vulnerabilities of the target system are firstly utilized to form single ($SingleCVSS$, `AV:N/A`) or combined (chained) vulnerability vectors ($ChainedCVSS$) - see Equation (6.5).

The available environmental information regarding existing security controls on network and physical layer, for each installation domain, is then applied to single, chained as well as $IntCVSS_{base}$ CVSS vectors as defined in Table 6.7. After the environmental transformation of all CVSS vectors is complete, the existing single and chained transformed vulnerability CVSS vectors are utilized in order to form the CVV_{Int}, which represents the overall vulnerability level of each interaction.

By examining the newly discovered findings in Section 3.8 in the work [262]), as presented in $IntCVSS$ vectors in Table 8.4, we can infer that various attack vectors are characterized with `Low` attack complexity. In addition, for the majority of attack scenarios the privilege requirements metric is set to *None* (an unauthenticated adversary can trigger the vulnerability), and attack vector is set as *Network (N)* since, in several vulnerability vectors the attack can be launched from public networks such as the Internet with substantial damage on the target system. For example, a firmware downgrade attack can potentially be launched from the Internet since

	Attack type	Status	$IntCVSS_{base}$							
			AV	**AC**	**PR**	**UI**	**S**	**C**	**I**	**A**
IoT Control device	Firmware Extraction	Confirmed	P	L	N	N	C	H	N	N
	Firmware Downgrade	Plausible	N	L	N	N	C	L-H	L-H	L-H
	Firmware Modification	Plausible	P	H	N	N	C	H	H	H
	Device system access	Plausible	P	L	N	N	C	H	H	H
Cloud API servers	Access sensitive information	Confirmed	N	L	N	N	C	H	N	N
	Waterhole attack	Plausible	N	H	N	N	C	H	H	N
	API manipulation	Plausible	A	ND	N	N	C	L	L	L
	session hijacking	Confirmed	A	L	N	N	C	L-H	L-H	L-H
	API exhaustion	Confirmed	N	L-H	N	N	C	N	N	L-H
	XSS / SQL injection etc.	Plausible	N	L-H	N	N	C	L-H	L-H	L-H
	Server-side Man-in-The-Middle attacks	Plausible	N	L-H	N	N	C	L-H	L-H	N
Device's ZigBee Network	De-Auth attacks	Confirmed	A	L	N	N	C	N	N	H
	Passive Sniffing	Confirmed	A	L	N	N	C	L-H	N	N
	Replay attacks	Confirmed	A	L	N	N	C	L-H	L-H	N-H
	DoS Attacks	Confirmed	A	L	N	N	C	N	L	L
	Gain Network Access	Confirmed	A	L	N	N	C	L	N	N
Device's WiFi Network	De-Auth attacks	Plausible	A	L	N	N	C	N	N	H
	Passive Sniffing	Confirmed	A	L	N	N	C	L	N	N
	Replay attacks	Plausible	A	L-H	N	N	C	L-H	L-H	N-H
	DoS Attacks	Confirmed	A	L	N	N	C	N	L	L
	Gain Network Access	Confirmed	N	L	N	N	C	L	N	N
Device's Mobile Application	Reverse engineering	Confirmed	N	L	N	N	C	L-H	L-H	N
	Dynamic analysis	Confirmed	N	L	N	N	C	L-H	L-H	N
	Application rights abusal	Plausible	L	L-H	L	N	C	L-H	L-H	N-H
	Application modification	Confirmed	N	L	N	N	U	L-H	L-H	N-H
	Client-side Man-in-The-Middle attacks	Confirmed	A	L	N	N	U	L-H	L-H	N

to∆AV: P=Physical, L=Logical access, A=Adjacent/Proximity, N= Remote network access
to∆AC/PR/UI/S/CIA: N=None, L=Low, H=High, ND=Not Defined

Table 8.4: $IntCVSS$ vectors representing the required capabilities and impact metrics for all applicable attack scenarios as defined in [262]

the cloud server that hosts the firmware files utilizes the *HTTP* protocol and the device does not implement any check on the firmware version numbering. In addition, network infiltration can be executed when the adversary is in network proximity with the target system [139] and the Internet. The latter is possible since UPnP protocol, which connects the smart lighting systems to the cloud API servers, is enabled by default[3] [4]. The majority of the attacks do not require any special tools and skill-set, thus enabling a diverse group of adversaries, ranging from activists and disgruntled workers to cybercriminal groups and nation state adversaries to remotely attack critical systems, services, or even people.

By taking into consideration all the above information, we can compute a single Vulnerability CVSS Vector for the specific smart lighting system (Device IDs 1,3,5,7,9,11 in our PoC scenario) that represents a worst-case scenario where an attacker manages to combine several of the aforementioned vulnerabilities - Equation 8.1.

$$[\mathrm{AV:Network, AC:Low, PR:None, UI:None, S:Unchanged/Changed, C:High, I:High, A:High)}] \tag{8.1}$$

The value *S:Changed* represents the cases where a smart lighting component's vulnerability directly affects directly another device-target (e.g. a vulnerability on the device allows to jam the target's communications). Finally, as described, factors *AV*, *AC* and impact (CIA) metrics can be modified depending on the environmental information of each installation domain.

In Table 8.5, we present the base vulnerability CVSS vectors for all devices in the PoC scenario.

Device ID	**CVSS Vulnerability Vectors (Base)**							
	Exploitability					**Impact**		
	AV	**AC**	**PR**	**UI**	**S**	**C**	**I**	**A**
1,3,5,7,9,11	Network	Low	None	None	Unchanged	High	High	High
2	Local	Low	None	None	Unchanged	High	High	High
4	Network	Low	None	None	Unchanged	High	Low	None
6	Adjacent	Low	None	None	Unchanged	High	High	None
8	Adjacent	Low	None	None	Unchanged	High	High	High
10	Adjacent	Low	None	None	Unchanged	Low	Low	Low
12	Network	Low	None	None	Unchanged	Low	Low	Low

Table 8.5: Existing CVSS vectors (vulnerabilities) for each device in the PoC scenario [262]

[3] https://www.checkpoint.com/defense/advisories/public/2020/cpai-2019-1605.html/

[4] https://nvd.nist.gov/vuln/detail/CVE-2020-12695

Depending on the installation site, environmental information such as network security controls and security requirements can alter the individual exploitability and impact metrics of the $IntCVSS_{base}$ vector and each vulnerability of the target device [85]. Table 8.6 depicts the corresponding cyber-physical security level for each network in scope whereas Table 6.7 display the affect on cyber interactions due to network security controls. For physical interaction types we consider the physical security control level as *Low* on all installation sites.

Layer	**Network cyber-physical security control level**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Cyber	L	H	M	M	M	L	M	L	L	L
Physical	L	L	L	L	L	L	L	L	L	L

L:Low, M:Moderate, H:High

Table 8.6: Environmental information regarding PoC network/physical layer security controls [262]

After applying the environmental information, $IntCVSS_{env}$ and the existing vulnerability CVSS vectors can now be combined (chained) appropriately, in order to form the cumulative vulnerability vector for each interaction (CVV_{int} - see Table 8.7).

Int. Tuple	***CVV_{int}* (*Environmental*)**							
	Exploitability					**Impact**		
	AV	**AC**	**PR**	**UI**	**S**	**C**	**I**	**A**
12 → 2	Network	High	None	None	Unchanged	Low	Low	Low
3 → 4	Network	High	None	None	Unchanged	High	Low	None
5 → 6	Adjacent	High	None	None	Unchanged	High	High	None
8 → 6	Network	High	None	None	Unchanged	High	High	High
7 → 8	Adjacent	Low	None	None	Unchanged	Low	Low	Low
9 → 10	Adjacent	Low	None	None	Unchanged	Low	Low	Low
9 → 10*	Adjacent	Low	None	None	Changed	None	Low	Low
9 → 10**	Adjacent	Low	None	None	Changed	None	High	High
1 → 12	Network	Low	None	None	Unchanged	Low	Low	Low

(*): P3 (Jamming) Zigbee → WiFi, (**): P3 (Jamming) WiFi → WiFi

Table 8.7: The computed CVV_{int} for each interaction (environmental) [262]

8.3.3 Phase 3: Attack path construction

In order to calculate all attack paths, all of the assessed (valid) interactions of the previous phase are fed to a recursive algorithm to produce complex attack vectors. In particular, `Level-1` assessed interaction tuples are considered as `Level-1` attack paths. Then, `Level-1` and `Level-2` interaction tuples are utilized with all the `Level-1` attack paths in order to form two-step (`Level-2`) attack paths. This process continues recursively until all the interaction tuples are exhausted. Since we focus on examining the introduced risks from smart lighting systems, we select the subset of those attack paths that the smart lighting system is either the target system or acts as an enabler (point-of-entry device). Tables 8.8 and 8.9 present the applicable `Level-1` and `Level-2` attack paths respectively.

Level-1 attack paths		
Level-1 Int. Tuple	**Source → Target Interface**	**Interaction Type**
1 → human	Luminaire → Eye	P2 (Disable)
1 → human	Luminaire → Eye/Receiver	P2 (Flicker)
3 → human	Luminaire → Eye	P2 (Disable)
3 → human	Luminaire → Eye/Receiver	P2 (Flicker)
3 → 4	WiFi → Ethernet	C4
5 → human	Luminaire → Eye	P2 (Disable)
5 → human	Luminaire → Eye/Receiver	P2 (Flicker)
5 → 6	WiFi → Ethernet	C4
7 → human	Luminaire → Eye	P2 (Disable)
7 → human	Luminaire → Eye	P2 (Flicker)
9 → human	Luminaire → Eye	P2 (Disable)
9 → human	Luminaire → Eye	P2 (Flicker)
9 → 10	WiFi → WiFi	P3(Jamming)
9 → 10	ZigBee → WiFi	P3(Jamming)
9 → 10	WiFi → WiFi	C1
11 → human	Luminaire → Eye	P2 (Disable)
11 → human	Luminaire → Eye	P2 (Flicker)

Table 8.8: Level-1 attack paths [262]

Level-2 Attack Paths		
Level-2 → Level-1 Interaction Tuples	**Source → Target Interface**	**Interaction Types**
1 → 12 → 2	WiFi → Ethernet → Ethernet	C4 → C5
7 → 8 → 6	WiFi → WiFi → Ethernet	C1 → C6

Table 8.9: Level-2 attack paths [262]

Calculating the cumulative vulnerability vector of an attack path:

For each attack path $CVV\big(\mathcal{AP}, AV\big)$ can now be formed for each possible attack vector, as follows: For `Level-1` attack paths the cumulative vulnerability vector of the attack path corresponds to the CVV_{int} as defined in Table 8.7. For `Level-n` $(n \geq 2)$ attack paths, each $CVV\big(\mathcal{AP}, AV\big)$ is calculated by utilizing the individual CVV_{int_n}, that the attack path is comprised of, and by applying Equation (6.8).

Calculating the businesswise impact:

Device Type	**ID**	**Impact**	**Interaction type**			
			P2		**P3**	**C1-C6**
			Disable	**Flicker**	**Jamming**	
Smart lights	**DevID 1**	**BI**	D/I(M)	D/I(M), DE	D(M)	DE,R
		LvL	M	H	L	H
	DevID 3	**Impact**	D/I(M)	D/I(M), DE	D(M)	DE,R
		LvL	M	H	L	H
	DevID 5	**BI**	D/I(M)	D/I(M), DE	D(M)	DE,R
		LvL	M	H	L	VH
	DevID 7	**BI**	D/I	D/I	D	DE,R
		LvL	VL	VL	VL	VH
	DevID 9	**BI**	D/I/F(M)	D/I/F(M)	D(M)	DE,R
		LvL	VH	H	M	-
	DevID 11	**BI**	D/I/F(M)	D/I/F(M)	D(M)	DE,R
		LvL	H	H	L	-

(M):Multiple, D:Discomfort, I:Injury, F:Fatality, DE:Data Exfiltration, R:Ransomware
-:Not Applicable BI: Businesswise Impact, LvL:Impact Level
VH:Very High, H:High, M:Moderate, L:Low, VL:Very Low

Table 8.10: Businesswise impact of smart lighting systems for each installation domain [262]

In order to assess the risk for each attack path the businesswise impact must be identified and calculated. Since, we focus on both cyber and cyber-physical interactions, we consider the impact deriving from cyber interactions (types C1-C6), as well as impact due to physical proximity (P2/P3). In particular, we consider direct impact to humans due to disable/flicker the luminaire (discomfort/injury/fatality), as well as the combined impact on air-gaped systems, due to internal networks connectivity and functionality features of a smart lighting system, that can create a covert channel in order to exfiltrate data, as described in [232]. For example, the impact of installing a smart lighting system in a corporate environment such as a major pharmaceutical company, may range from moderate-impact attack scenarios such as the physical interaction with the employees (disable the smart lighting systems of an entire building to cause injuries/discomfort), up to potential high-impact, data exfiltration scenarios where vaccine formulas are leaked from otherwise air-gaped corporate servers due to internal networks connectivity and smart lights vulnerabilities/functionality features.

Furthermore, installing vulnerable smart lighting systems in crowed infrastructures, such as a sports stadium, may result in catastrophic consequences (multiple deaths/injuries) especially when the attack is combined with other IoT-enabled systems such as a remotely managed alerting system (e.g. trigger the alarm and disable all lights in stadium during the evacuation process). Table 8.10 represents the businesswise impact, based on worst case scenarios, for each smart lighting system for all installation domains. For impact scale, we utilize the scale/context as presented in National Institute of Standards and Technology - NIST [235] *Very Low → Very High.*

8.3.4 Attack path assessment

In order to assess the risk introduced by each attack path the attack path scenarios must be first formed. First, the cumulative vulnerability vectors of the point-of-entry device are calculated for each available attack vector $AV = [N|A|L|P]$, defined in Chapter 6, using the existing vulnerabilities. Then, for each possible attack vector of the examined attack path $CVV\big(\mathcal{AP}, AV\big)$ is calculated, using Equation (6.8). Finally, depending on their capabilities access and resources, the applicable threat agents are matched with each attack path's cumulative vulnerability vector $CVV\big(\mathcal{AP}, AV\big)$, for $AV = [N|A|L|P]$, to form the corresponding attack path scenarios. Only after the scenarios are formed the risk assessment phase can begin.

Calculating threat likelihood level:

In Table 8.11, we present a list of common adversary types and their corresponding characteristics such as capabilities, access, motives and available resources. In addition, available attack vectors for each adversary type and businesswise threat level (likelihood of occurrence) per installation domain are presented in Table 8.12. The latter has been calculated by taking into consideration both generic and/or sector-specific threat intelligence sources such as threat reports from security organizations (e.g. ENISA Threat Report [130], NIST) and security companies [99, 135], as well as cyber security incidents found on relative websites.

Adversaries	**Capabilities**	**Physical Access**	**Motives**	**Resources**
Activist	AV:N,A,P/AC:L/PR:N/UI:N	External	1,2	Limited
Disgruntled Worker	AV:N,A,L,P/AC:L/PR:N,L/UI:N,R	Internal	1,2	Limited
Bussiness Competitor	AV:N/AC:L/PR:N,L/UI:N	External	1,3	Significant
Cyber Criminal Group	AV:N/AC:L,H/PR:N,L,H/UI:N,R	External	3,4,5	High
Cyber Terrorist	AV:N,A,L,P/AC:L,H/PR:N,L,H/UI:N,R	External/Internal	1,2,4,5	High
Nation State	AV:N,A,L,P/AC:L,H/PR:N,L,H/UI:N,R	External/Internal/Protected	1,2,4,5	Very High

Motivation: 1=Harm Reputation, 2=Damage/Disable equipment, 3=Financial Gain, 4=Harm Humans, 5=Extract Information

Table 8.11: Applicable adversary profiles and their corresponding characteristics

Adversary type	**Attack Vector (AV)/Likelihood (L)**											
	Bank		**Pharmaceutical**		**Government**		**Home**		**Stadium**		**Public Lighting**	
	AV	**L**	**AV**	**L**	**AV**	**L**	**AV**	**L**	**AV**	**L**	**AV**	**L**
Activist	N	Low	N	Mod	N	Mod	–	–	N,A	VL	N,A,P	VL
Disgruntled Worker	N,A,L,P	Low	N,A,L,P	Low	N,A,L,P	Low	–	–	N,A,L,P	VL	–	–
Bussiness Competitor	N,A	Mod	N,A	Mod	–	–	–	–	–	–	–	–
Cyber Criminal Group	N,A	VH	N,A	High	N,A	High	N	Mod	N,A	Low	N,A,P	VL
Cyber Terrorist	N,A	Low	N,A	Low	N,A	Mod	N,A	VL	N,A,P	Mod	N,A,P	Mod
Nation State	N,A,L,P	Mod	N,A,L,P	High	N,A,L,P	High	N,A,L,P	VL	N,A,L,P	Low	N,A,L,P	Low

–: Not Applicable, Mod:Moderate, VL:Very Low, VH:Very High

Table 8.12: Businesswise threat level (likelihood of occurrence) and applicable attack vectors for each installation domain per adversary type [262]

Calculating Risk level:

By combining the vulnerability, impact (see Section 8.3.3) and threat level as well as the assessment formula (see Equation 6.3), the risk for each attack path scenario can be defined for all applicable threat agents. This is essentially computed by using the risk matrix as presented in Table 6.14. Similar to impact, the context and scale of risk level is identical with the one described in [235].

8.4 Result analysis

A total of 90, smart light's enabled, attack path scenarios were created during the attack path assessment phase, 24 of which were cyber (27%) and 66 cyber-physical (73%), 38% of which were characterized as *very high* and as 27% *high* impact, due to high-value/criticality of most targets in scope. In addition, *medium* (18%), *low* (11%) and *very low* (7%) impact attack path scenarios correspond mostly to direct cyber-physical attack paths on corporate/home environments (e.g. cause discomfort/injuries to employees/families). Four typical examples of attack path scenarios with *very high* impact are presented in Figure 8.3. Two cyber attack scenarios involving cyber criminals and nation state adversaries against a bank and a government cloud infrastructure respectively, and two cyber-physical ones: A terrorist attack against a crowded stadium and a sophisticated data exfiltration attack against a major pharmaceutical corporation.

Threat actors' likelihood profile included a 19% of *very low* probability and a 39% with *low* probability of attack path scenarios, which, mainly correspond to activists and disgruntled workers whereas *medium* likelihood scenarios were spread evenly among all threat agents. As expected, threat agents such as cyber criminals and nation-state were responsible for 16% of the most likely to happen attack path scenarios (13% *high* - 3% *very high*).

As far as Vulnerability level is concerned 93% of all smart light enabled, attack path scenarios were identified as *high* (55%) or *critical* (38%) even in cases were the environmental network security controls were defined as *high* (bank's server farm internal network) mainly due to the fact of the existing high-severity vulnerabilities/misconfigurations of the smart lights and target systems.

Riskwise, the majority (67%) of assessed scenarios were categorized as of *moderate* risk (51% cyber-physical and 16% cyber), 27% as *high* (16% cyber-physical and 11% cyber) and a 14% as *low* (7% cyber-physical and 7% cyber). All of the above are depicted in Figure 8.4. Further analysis of the results (Figure 8.5) revealed that, regarding the risk profile of each threat agent for both cyber, as well as cyber-physical attack path scenarios, cyber criminals are the adversaries with the highest percentage

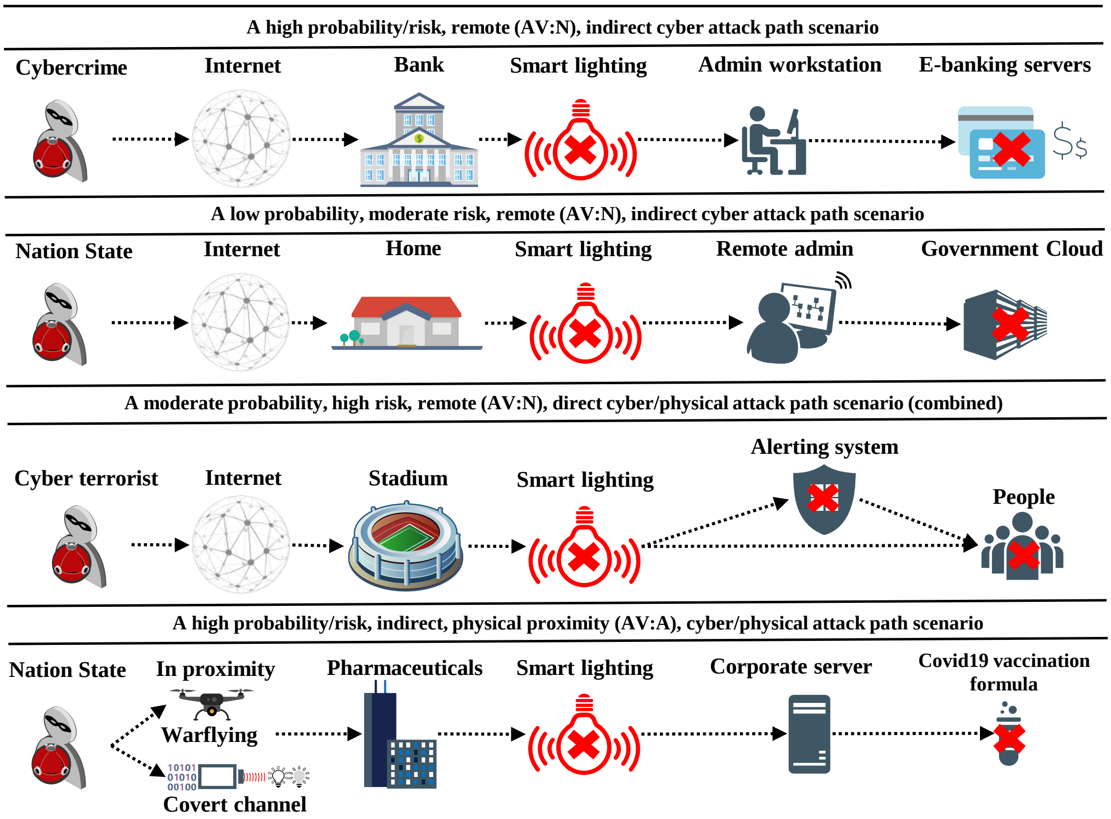

A high probability/risk, remote (AV:N), indirect cyber attack path scenario
Cybercrime
Internet
Bank
Smart lighting
Admin workstation
E-banking servers
A low probability, moderate risk, remote (AV:N), indirect cyber attack path scenario
Nation State
Internet
Home
Smart lighting
Remote admin
Government Cloud
A moderate probability, high risk, remote (AV:N), direct cyber/physical attack path scenario (combined)
Cyber terrorist
Internet
Stadium
Smart lighting
Alerting system
People
A high probability/risk, indirect, physical proximity (AV:A), cyber/physical attack path scenario
Nation State
In proximity
Warflying
Covert channel
Pharmaceuticals
Smart lighting
Corporate server
Covid19 vaccination formula

Figure 8.3: Examples of *Very High* impact, cyber and cyber-physical attack path scenarios [262]

(46%) of the *high* risk attack scenarios followed by nation-state (29%) and cyber terrorists (17%). This is mainly due to the fact that their capabilities satisfy most of exploitability requirements of high-impact attack scenarios and are more likely to attack (higher probability) than the other threat actors, in most domains in scope.

The fact that disgruntled workers were responsible for 17% of *low* and 22% of *Medium* risk attack path scenarios does not mean, in any case, that these are risks that can be easily overlooked, since, in most cases, corresponded to *high*/*very high* impact attack scenarios. For example, a disgruntled public servant that works as an administrator at a government's central ICT infrastructure can have significant impact on several mission-critical applications and services (e.g. taxation/COVID19 vaccination applications), especially if no proper security countermeasures exist and no business continuity plans are in place.

Finally, business competitors and activists were accountable mainly for *medium*/*low* risk and only for a small (4% each) portion of *high* risk attack path scenarios. On

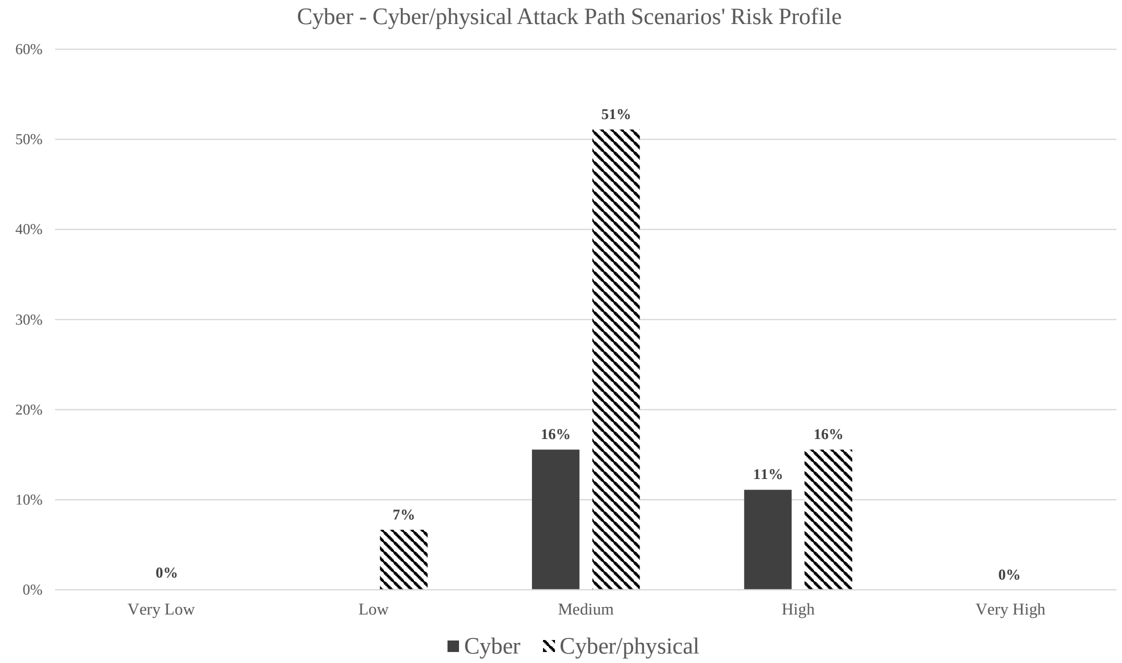


Figure 8.4: The distribution of the risk levels for both cyber as well as cyber-physical attack path scenarios [262]

the other hand, the fact that limited resourced, low-skilled adversaries are capable of triggering *high* risk attack path scenarios can be considered an alarming finding depending on the organization type.

Even though some of the described attack scenarios that are based on network connectivity are relatively easy to identify, assessing the risk of untested IoT equipment installed within critical environments can be tricky. Intuitively, the isolation of the organization's internal networks from the Internet seems to be sufficient to mitigate the risks that stem out from IoT-enabling technologies. Although this is indeed a step towards the right direction, it can also give a false sense of security, since an adversary can exploit both physical proximity (e.g. via wardriving/warflying techniques) and vulnerabilities found on the smart lighting system to extend the functionality [232] and ultimately exfiltrate valuable corporate secrets via a newly created, covert channel. Moreover, recent developments due to the pandemic, has lead many IT administrators to work from home environments thus introducing new, high-impact attack scenarios that may include home IoT devices such as smart lights.

Risks from public lighting systems are usually overlooked since, no real, high-profile attacks exist. But as the deployment of these systems grows, it becomes

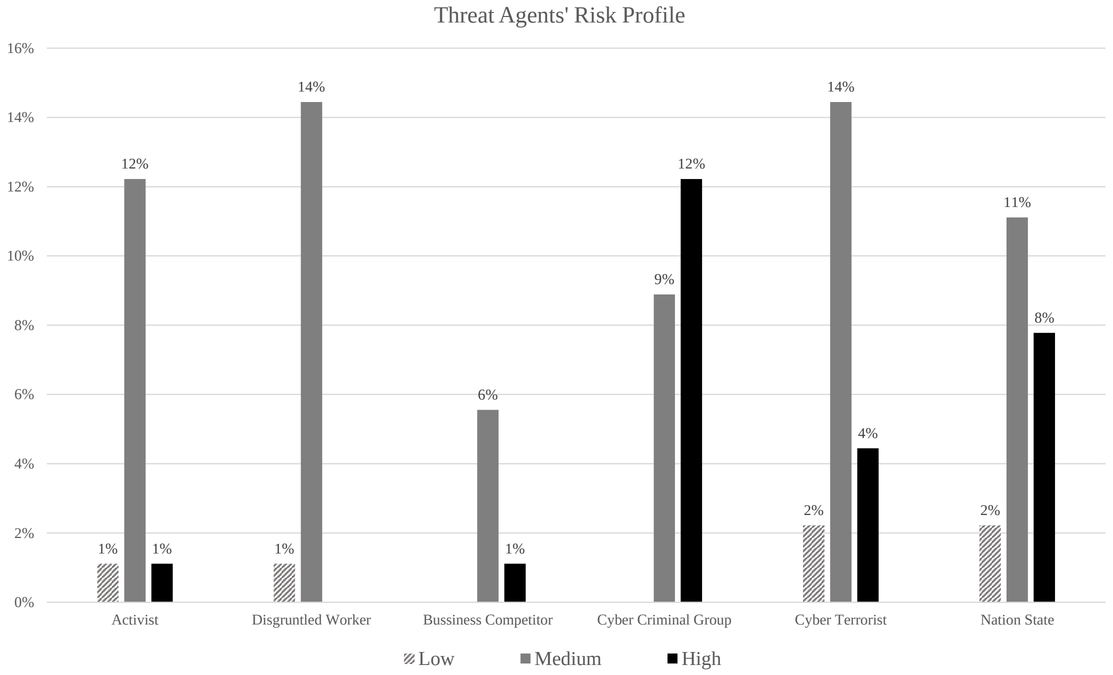


Figure 8.5: Threat agents' risk profile for both cyber as well as cyber-physical attack path scenarios [262]

more crucial and therefore more valuable. As modern cities embracing technology at a rapid pace, smart lighting systems can be expected to become more and more integrated with other building management systems such as HVAC, alerting systems and other IoT-enabled infrastructure automation components. Ultimately, this means that smart lighting systems will be an even more tempting target for a variety of threat agents in the near future.

CHAPTER 9
E-HEALTHCARE POC VALIDATION SCENARIO

9.1 Test scenario

To further validate the efficiency and accuracy of the proposed methodology and to prove its effectiveness across different critical environments, we used as a test case a realistic scenario from the healthcare sector (see Figure 9.1). In particular, we focused on critical systems and services such as on-line remote healthcare services and near-patient infusion pumps. We simulated scenarios where the infusion pump is placed both in a smart home, as well as within a hospital.

In addition, we included various low-importance IoT devices in both environments such as smart lamps, thermostats and IP surveillance cameras, as well as traditional ICT systems such as personal computers, network routers and access points. We defined logical access rules among the devices (e.g. to allow a doctor to monitor and reprogram infusion pumps via e-health services). In addition, for each device several well-known CVEs, or in some cases custom CVEs based on previous research were assigned (e.g. remote takeover of smart lights as described in [233] → CUS-2016-1).

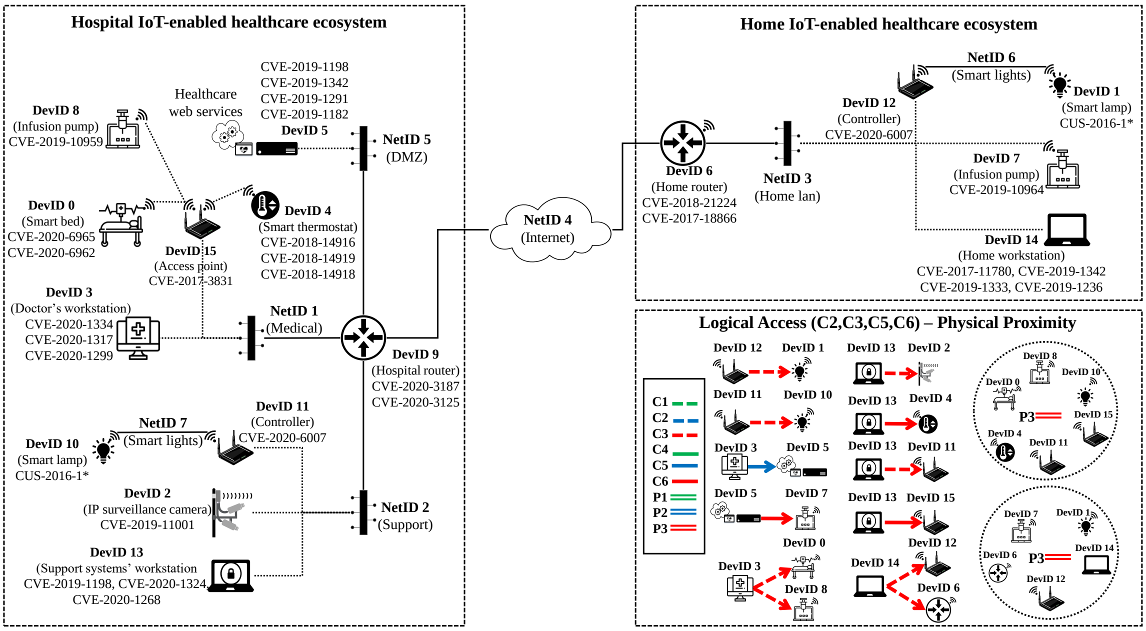


Figure 9.1: The simulated scenario - Network diagram, CVEs, cyber and physical proximity [257]

In order to be as realistic as possible we included popular medical devices and ICT equipment. In particular, we utilized two infusion pumps, one by 'BD Alaris'[1] (near-patient, home) and another one by 'Medtronic' (in-hospital)[2], as well as a patient monitor (Carescape B450 by 'GE healthcare'[3]). In addition, we added IoT devices such as smart lighting systems (Philips), a smart thermostat, an IP-enabled surveillance camera with infrared interface as well as windows server(s) running remote medical services, network equipment by Cisco and D-Link and home/hospital workstations running Windows 10.

For each device specific software version(s) based on the Common Platform Enumeration (CPE) standard IDs (CPEIDs - see Table 9.1), vulnerabilities, cyber-physical interfaces, physical location (hospital/home), each device's interface network and logical access to other devices were assigned. Moreover, network relations (Table 9.2), network access rules among devices as well as relevant security controls on network layer (environmental information) were defined. For each interface type, cyber and physical interaction types, range and type (internal, external) were also identified. For example, an infusion pump is physical located at the hospital (physical location type:internal), has one wireless interface (802.11.x) that is connected to an internal network ((NetID 1), can interact with other devices with interfaces that operate in the same band (e.g. Philips hue smart lamps - interaction type P3) and is remotely managed by e-healthcare software (DevID 5). Except from traditional cyber attack vectors (`AV:N/A/L`) we also included non-traditional attack methods such as those described in [119].

DeviceID	CPEID	DeviceID	CPEID
DevID 1	cpe:2.3:o:gehealthcare:carescape_b450_monitor	DevID 2	1.56.8_r30456 (Philips)
DevID 3	cpe:2.3:h:reolink:rlc-410w	DevID 4	cpe:2.3:o:microsoft:windows_10:1909
DevID 5	cpe:2.3:h:loytec:lgate-902	DevID 6	cpe:2.3:o:microsoft:windows_server_2016:1903
DevID 7	cpe:2.3:o:netgear:wnr2000	DevID 8	cpe:2.3:h:medtronic:minimed_508
DevID 9	cpe:2.3:h:bd:alaris_gs_syringe_pump	DevID 10	cpe:2.3:o:cisco:asa_5512-x_firmware:9.12 \(2.12\)
DevID 11	1.56.8_r30456 (Philips)	DevID 12	cpe:2.3:h:philips:hue_bridge_v2
DevID 13	cpe:2.3:h:philips:hue_bridge_v2	DevID 14	cpe:2.3:o:microsoft:windows_10:1903
DevID 15	cpe:2.3:o:microsoft:windows_10:1703	DevID 16	cpe:2.3:h:cisco:aironet_1800

Table 9.1: CPEIDs for all devices in PoC scenario [257]

Healthcare is an attractive sector for adversaries such as organized cyber crime, due to the great value of proprietary research data (e.g. COVID-19 vaccine) as well as patient's medical information such as Electronic Health Records (EHR) in the black market whereas healthcare organizations such as hospitals are considered as 'profitable business' of ransomware campaigns. In addition, COVID-19 pandemic

[1] https://www.bd.com/en-us/products-and-solutions/products/product-families/bd-alaris-pump-module

[2] https://www.medtronic.com/us-en/healthcare-professionals/products/neurological/drug-infusion-systems.html

[3] https://www.gehealthcare.com/products/patient-monitoring/patient-monitors/carescape-monitor-b450

increased the need for telehealth services and therefore the interest in dark Web mentions increased 144% according to a recent threat report [245]. In our threat analysis we considered several types of realistic threat agents, ranging from highly motivated adversaries such as cyber criminals, to internal, moderately motivated and skilled disgruntled employees. In addition, we defined specific motives known to be applicable to the healthcare sector. Finally, for each motive, we took into consideration past and present threat reports ([87, 221, 222, 245]) including recent reported incidents regarding the healthcare ecosystem such as the one presented in [244], in order to define the likelihood of each adversary type. We also applied different likelihood levels for same adversary types depending on the point-of-entry devices' environment (home/hospital). To test our methodology, we first identify all the cyber and physical interactions, using all devices in scope as possible targets. Then, we calculated the attack paths for the three critical target systems: two medical pumps, one inside the hospital and the other in the home environment (DevIDs `7`, `8`) and also an e-health services Web server (DevID `5`). We assess the relevant interactions and we calculated the risk of the attack paths towards all the three predefined targets.

NetID	**1**	**2**	**3**	**4**	**5**	**6**	**7**
1	N/A	✓		✓	✓		
2	✓	N/A		✓			
3			N/A	✓	✓		✓
4			✓	N/A	✓		
5			✓	✓	N/A		
6						N/A	
7							N/A

Table 9.2: Network access rules [257]

In the attack path assessment phase, we first computed all applicable CVVs of the initial node of each attack path and then we went on calculating the $CVV\big(\mathcal{AP}, AV\big)$ for each attack path scenario. In order to define the applicable threat agents for each attack path, we compare their characteristics shown in Table 9.3 with each of the $CVV\big(\mathcal{AP}, AV\big)$ exploitability metrics.

To calculate the impact level for each attack path we utilize the vulnerability impact metrics of the '`Level-1`' interaction of each attack path and apply the security requirement weights. In particular we defined the latter as `C/I/A:L/M/M` and `C/I/A:M/H/H` for home and hospital infusion pumps as well as `C/I/A:H/H/H` for e-health services. In particular, we consider the impact of exploiting a single infusion pump placed in a home environment to be significant lower than the one of multiple infusion pumps installed in a hospital whereas the e-health Web services is considered as a high impact target.

Adversaries	Capabilities	Physical/Network Access Level	Motives	Resources	LH
Rights Activist	AV:N/AC:L/PR:N/UI:N	External	1	Limited	L
Disgruntled Worker	AV:N,A,L/AC:L/PR:N,L/UI:N,R	Internal (Hospital)	1,2	Limited	L
Disgruntled Administrator	AV:N,A,L,P/AC:H/PR:N,L,H/UI:N,R	Internal/Protected (Hospital)	1,2	Moderate	L
Business Competitor	AV:N/AC:L/PR:N/UI:N,R	External(Internet)	1	Significant	M
Cyber Criminals	AV:N/AC:L,H/PR:N/UI:N,R	External (Internet)	3,4,5	High	VH/L
Cyber Terrorist	AV:N,A,L,P/AC:L,H/PR:N/UI:N,R	External/Internal (Hospital/Home)	1,2,4	High	M/L
Nation State	AV:N,A,L,P/AC:L,H/PR:N/UI:N,R	External/Internal (Hospital/Home)	1,2,4,5	Very High	L

Motivation: 1=Harm Reputation, 2=Damage/Disable equipment, 3=Financial Gain, 4=Harm Patient(s), 5=Steal Patients' Data
LH:Likelihood: Hospital/Home, L:Low, M:Moderate, H:High, VH:Very High

Table 9.3: Adversarial model for healthcare ecosystem [257]

Finally, we utilized Table 6.14 from Section 6.8.4 to define the risk level of each attack path scenario.

9.2 Results analysis

In order to test the performance of the algorithm we first run the simulation for the creation of interaction tuples using each node as the target device. Table 9.4 sums up the required time for computing all possible interactions. Then, we proceeded with the implementation of all of the methodology phases for the three preselected critical targets.

Target Device	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**sum**	**averg**
Time (sec)	1,71	1,46	0,80	1,02	1,04	1,14	1,34	1,40	1,11	0,70	1,19	0,85	1,39	0,84	1,39	1,01	18,40	1,15
Levels	3	6	4	3	3	4	6	5	3	3	4	4	6	4	6	3	N/A	4,19
Interactions	113	142	109	108	76	118	97	75	113	107	124	112	137	109	140	99	1773	120,06

Table 9.4: Interaction modelling calculation time (per target device/total/average)

Interaction modelling and attack path construction phase

	Level 1	Level 2	Level 3	Level 4	Level 5	Level 6
Interactions	23 (9 Phy)	87	87	50	1	0
Assessed Interactions	19 (9 Phy)	65	47	50	1	0
Attack Paths (Cyber)	10	47	154	454	688	478
Attack Paths (Cyber-Physical)	8	24	68	171	1	0
AP Scenarios (Cyber)	46	162	514	1555	2283	1603
AP Scenarios (Cyber-Physical)	16	66	246	682	6	0

Table 9.5: Interactions, attack paths and attack path scenarios per interaction level for all three targets [257]

From Table 9.5, we can infer that our target-oriented approach reduced the multitude of potential interactions of all devices, networks and interfaces for all three targets to 245 cyber-physical interaction tuples in total whereas in the vulnerability assessment of the interaction tuples phase the overall number was further reduced by 27% (182). From the latter, 2103 attack paths were formed, of which 272 cyber-physical, for all three targets. Finally, for all the predefined threat agents, 6163 cyber and 1016 cyber-physical attack path scenarios (mappings of attack paths to applicable threat agents) were formed and assessed.

Risk assessment phase

Risk analysis of the formed attack path scenarios resulted in a variety of risk levels ranging from very low to very high (see Figure 9.2). In particular, 75 (1,2%) of the assessed cyber threat scenarios were characterized as very high whereas the highest risk level of cyber-physical was high (4%).

The adversary risk profiles for the healthcare ecosystem paradigm is depicted in Figure 9.3. By further analyzing the results we defined the AP scenarios that each device participated either as an intermediate node in the attack chain or as a Point-of-Entry. As shown in Table 9.6 the devices with IDs **3**, **0** and **13** are the top three devices that are part of, or act as enablers for an AP scenario. In addition, the aforementioned devices were also the ones with the highest score concerning AP scenarios with risk levels very high or high.

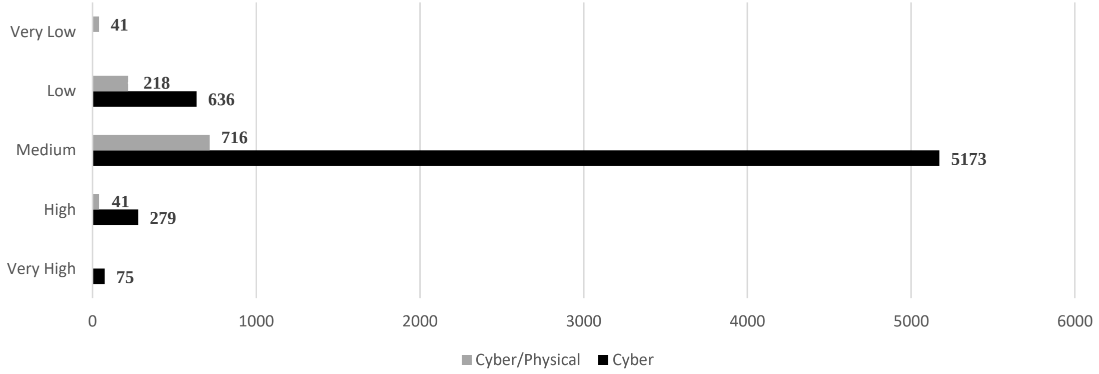


Figure 9.2: Cyber and cyber-physical attack paths scenarios per risk level

TargetID	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
AP Scenarios	4857	32	4574	4927	709	2458	122	0	2002	2568	288	3199	101	4742	92	2138
As point-of-entry	315	11	762	1016	0	9	24	0	465	2568	230	423	11	562	24	759

Table 9.6: Multitude of AP scenarios per node for targetIDs **5**, **7** and **8**

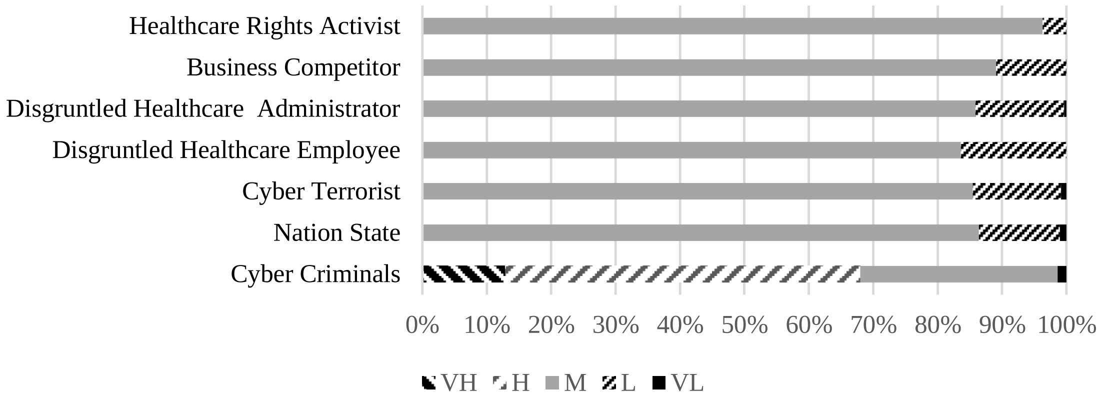


Figure 9.3: Risk profile of each predefined threat agent [257]

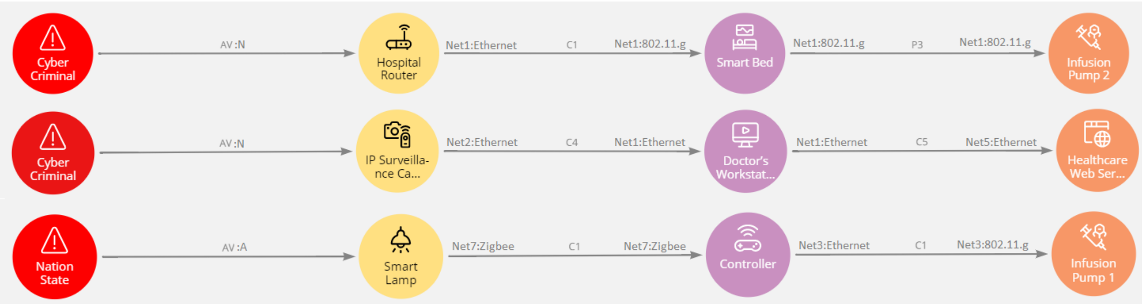


Figure 9.4: High impact, IoT-enabled, stealthy cyber/cyber-physical AP scenarios paradigms from our test scenario [257]

Besides the analysis and ranking of the attack paths and the relevant scenarios, and beyond the 'expected' high-risk paths, our methodology may assist the risk assessor to identify underestimated and/or hidden attack paths. We analyse three characteristic AP scenarios provided by our tool (see Figure 9.4). We deliberately included high impact and low probability scenarios, as those are likely to be overlooked by typical risk assessment methodologies. The first is a stealthy, cyber-physical AP Scenario of high risk. A cyber-criminal takes advantage of software vulnerabilities on the hospital's main router to gain initial foothold, then exploits vulnerabilities found on IoMT devices (patient healthcare monitors and smart beds) and causes DoS to multiple IoT-enabled infusion pumps by exploiting the physical proximity of interfaces working in the same band frequency). Such an attack path could be part of a ransomware campaign.

In the second AP scenario, a remote adversary exploits a critical vulnerability found on an Internet exposed IP surveillance camera; then exploits via lateral movement an IoT-enabled healthcare monitor to ultimately to gain access to hospital's Web services and exfiltrate sensitive patient data. Finally, the third AP scenario is considered as a stealthy, high impact - low likelihood scenario. An highly skilled/resourced adversary (e.g. nation state) targets a home patient: Via war driving techniques she manages to initially infiltrate the patient's home network by exploiting vulnerabilities found in smart light bulbs and their controller; Then, she locates the IoT-enabled infusion pump and by exploiting the device's existing vulnerabilities threatens the patient's life.

SECTION V
Mitigation and future work

CHAPTER 10
RISK MITIGATION OF IOT-ENABLED ATTACKS

10.1 Mitigating the risks of IoT-enabled attack scenarios

From the analysis of the attack scenarios presented in Chapter 7, it is shown that various attack patterns are common to many sectors, while other attacks are specific to a particular domain. Usually, the IoT devices increase the vulnerability level, while the lack of physical and logical access controls exposes critical systems to threats. Their inherent security weaknesses stem from their constrained computing capabilities and their poor security design. These features, combined with their connectivity and functionality capabilities as well as their non-obvious (indirect, subliminal or hidden) interaction with other systems, are the main reasons for this radical change. To be consistent with the risk-based assessment methodology presented in Chapter 5.1, we will examine the security controls according to which risk factors they primarily mitigate. Thus, we present security controls based on whether they mainly reduce the threat, the vulnerability or the impact level. Note, however, that usually a security control may reduce at the same time multiple risk factors. Therefore, a mitigation strategy shall methodologically examine alternative strategies based on various combinations of controls [264] using cost and benefit analysis.

In Table 10.1, we present a detailed mapping of the proposed security controls with all the characteristics they positively affect as well as which of the examined attacks could be mitigated (at least partially) for all attack path scenarios. Likewise, for each security control, we indicate which actors are usually responsible to implement the control: The system *Owner*, the system *Administrator*, the IoT *Manufacturer* or finally a *Regulator* (standardization or governmental body).

By examining Table 10.1, one can infer that some security controls are usually neglected in specific attack path scenarios, and therefore sectors. For example, avoid/-controlling direct Internet access with the IoT are high priority controls for direct attack scenarios. Segmentation of internal networks should be a top priority against indirect attack path scenarios. For no-connectivity scenarios continuous security testing, key management and identifying IoT dependencies are some of the most prominent controls. Las but not least, some controls such as those related with software security, seem to be of high priority for all attack path scenarios.

10.1.1 Reducing the threat level

The goal of these controls is to increase the access, capabilities and motivation threshold required by potential adversaries to trigger an attack. Since the threat level usu-

Table 10.1: A summary of the security controls for IoT-enabled cyber attacks [261]

	Controlling Access				Mitigating IoT vulnerabilities													Examples of affected attack paths			Actor[a]
	Physical		Logical		HW layer			SW layer			Network & Protocols				Key Management						
SECURITY CONTROLS	Ins.	Out.	Priv.	Unpr.	Tampering	Embed. Crypto	Implementation	Firmware	Operat. System	Application	Netw. Design	Link-Layer	Netw.-Layer	Appl.-Layer	No PKC	Common /No Key	Extractable Key	Direct	Indirect	No conn.	
Limit physical access to IoT	✓				✓													[95, 100, 107, 253, 272]	[81, 131, 239]	[51, 124]	A
Monitor physical access to IoT	✓				✓													[37, 224, 253, 272]	[81, 131]	[124]	A
Avoid direct Internet access		✓	✓	✓					✓	✓			✓	✓	✓			[121, 223, 254, 256]	[22]	[58, 242, 271]	A
Enforce proxy-based access		✓	✓	✓					✓	✓				✓				[37, 90, 95, 100, 121, 223, 254, 256, 272, 274]	[193, 287]	[124, 242]	A
Secure remote access		✓	✓	✓									✓	✓				[37, 90, 121, 223, 254, 256, 274]	[22, 193, 287]	[58, 233, 242]	A
Apply security extensions for link-layer protocols	✓		✓	✓								✓						[21, 45, 100, 182, 294]	[188, 217]	[198]	A
Log and monitor access to IoT	✓	✓	✓	✓														[154]	[22, 28]	[58, 271]	A
Audit access to IoT	✓	✓	✓	✓															[22, 28]		A
Tamper resistance mechanisms	✓				✓	✓	✓	✓								✓	✓	[31, 37, 154, 253, 294]	[217]	[84]	M
Secure embedded crypto	✓	✓			✓	✓						✓	✓					[182, 253, 256]		[233]	M
Side-channel attack protection	✓				✓		✓													[232, 233, 286]	M
Firmware protection	✓	✓	✓	✓				✓										[31, 45, 154, 253, 254]	[193, 239]	[233]	M
Secure firmware update	✓							✓										[31, 45, 253]	[193, 239]	[49, 233, 286]	M
Secure OS architecture									✓									[21, 90, 95, 107]	[28, 104, 131, 160, 193, 277]	[58, 124, 286]	M
OS hardening									✓									[21, 107, 121, 182, 272]	[28, 104, 131, 160, 277]	[58, 124, 286]	M
Use of secure API										✓								[31, 239, 254, 256, 274]	[81, 131, 239, 279, 287]	[49, 58, 84, 232, 242, 271]	M
Code auditing										✓								[31, 254, 272, 274]	[81, 131, 203, 239, 287]	[84, 232, 271]	M
Network security protocols												✓	✓	✓				[37, 45, 100, 107, 256, 294]	[104, 160, 193, 203, 217, 279]	[232]	M
Secure key management																✓	✓	[107]	[239]	[49, 84, 198, 233]	M
Secure key exchange															✓	✓		[107]		[49, 233]	A
Device acquiring criteria					✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	[21, 154, 182, 272]		[242]	O
Secure change management						✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	[107, 224]	[131, 277]		A
Continuous security testing					✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	[21]	[104, 160, 277]	[58, 198, 233, 242, 286]	A
Security standards enforcement					✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	[182, 294]	[104, 160, 203, 217]	[58]	R
Identify IoT dependencies	✓	✓	✓	✓														[272]	[28, 239]	[49, 51, 124, 232]	A
Re-examine BYOD policies	✓		✓	✓																[51, 84]	A
Avoid physical proximity		✓	✓	✓															[81, 188]	[49, 124, 232, 286]	A
Segment networks to avoid cascading impact	✓	✓	✓	✓							✓							[95, 224]	[22, 28, 104, 131, 131, 160, 193, 203, 239, 277, 279]	[233]	A
Favor technology diversity	✓	✓	✓	✓							✓							[90]		[233]	O

[a] O: owner; A: administrator; M: manufacturer; R: regulator

ally depends on the specific system environment, the implementation of these controls usually relies on the system operator. In particular:

- **Limit physical access to IoT:** Avoid installing IoT devices in places that are physically accessible to unauthorized users. Physical access rules must be enforced on all internal areas of an organization and proper characterization of security zones must be applied. In particular, state-of-the-art techniques in limiting access to areas with critical systems (e.g. data centers) may include proper control mechanisms via the use of keycard-controlled doors and biometric identification mechanisms. Additionally, restriction policies regarding the entry and exit of personnel/equipment/media must be in place whereas physical isolation should extend to include locations of network equipment and cabling (both power and network), HVAC and any other IoT-enabled systems (e.g. temperature sensors).
- **Monitor physical access to IoT:** Physical access to IoT devices should be monitored (e.g. via closed-circuit television cameras and motion detectors), especially for critical IoT devices that must be installed in places accessible by outsiders.
- **Avoid direct Internet access:** Avoid assigning IoT devices with public IP addresses directly if this in not an absolute necessity. The use of local IPs and indirect Internet access through a gateway/firewall should be preferred.
- **Enforce proxy-based access:** Consider access through proxy systems that provide advanced authentication and authorization capabilities and security policy enforcement, to "encapsulate" vulnerable IoT interfaces.
- **Secure remote access:** Addressing risks due to logical access cyber interaction types is a much more daunting task, to begin with. Remote management of critical systems, which has been on the rise due to the pandemic of COVID19 resulted in a unprecedented rise of cyber attacks[1]. Remote access to IoT devices should be protected with secure authentication and encryption mechanisms. Especially for Internet access, strong authentication, encryption and integrity controls should be applied (e.g. use of SSL/TLS, SSH or VPN protocols), to drastically increase the difficulty for potential adversaries. In any case, privileged remote access to IoT devices and services must be thoroughly assessed and be submitted under exhaustive security evaluation. State-of-the-art security countermeasures may include, among others, Multi-Factor Authentication (MFA) schemes, constant monitoring via the use of Security Information

[1] https://www.swissinfo.ch/eng/jump-in-cyber-attacks-during-covid-19-confinement/45818794

and Event Management (SIEM) systems, application security (e.g. security-by-design - secure software development), 3-tier architecture, least privilege principle enforcement on all layers (application database, Web), strict endpoint security rules and application vulnerability assessments/penetration testing.

- **Apply security extensions for link-layer protocols:** IoT devices that are directly connected to critical systems should be configured with the highest available security level provided by the data link layer protocol used. For example, use of the AES in Galois/Counter Mode (GCM)[2], to ensure data encryption and integrity at the same time (by default IEEE 802.15.4 does not apply any security mode [227]). Another example is the use of security extensions for ad-hoc networks, such as those described in [68, 219], to deal with wormhole and sybil attacks.
- **Log and monitor access to IoT:** Continuously log and monitor access to/from IoT devices. When possible, use IDS and/or IPS to monitor access to IoT devices and prevent attacks, especially from the Internet.
- **Audit access to IoT:** Enforce auditing procedures to trace potential attackers in a timely manner. The last two controls can increase the counter-motivation of potential adversaries, since with proper logging and monitoring, adversaries are more likely to be traced. Therefore, a potential adversary will also consider the potential consequences (e.g. legal), if traced, and not only the potential gain from a successful attack.

10.1.2 Reducing the IoT vulnerability level

The goal of these controls is to reduce the available attack surface of the IoT devices. Since the most of the vulnerabilities are inherent to the devices, usually the manufacturers are the actors that can implement such controls. Regulator bodies can also enforce the implementation of such controls. In some cases a proper configuration of an IoT device by the administrator, may reduce the vulnerability level. In particular:

- **Tamper resistance mechanisms:** IoT devices should implement mechanisms to detect and prevent physical tampering. For example, mechanisms that physically destroy a critical component or that securely delete an embedded crypto key, if physical tampering is detected. Additionally, IoT-devices that are placed unattended, in public areas must be placed into protective cases.

[2] https://en.wikipedia.org/wiki/Galois/Counter_Mode

- **Secure embedded crypto mechanisms:** IoT devices should implement tested and secure crypto algorithms in the proper mode of operation. For example, although AES is secure, implementations in CCM mode have been found vulnerable to cryptanalysis attacks [233].
- **Protection from side-channel attacks:** IoT devices, especially those installed in critical premises, should be implement hardware security controls for protection from side-channel attacks, such as, protection from power analysis attacks that may leak sensitive information [233].
- **Firmware protection mechanisms:** The firmware of IoT devices should be protected from unauthorized access and modification. Techniques like obfuscation, packaging and encryption should be used. In addition, security countermeasures such as trusted execution environments (e.g. via code signing - secure boot process), embedded code obfuscation and Trusted Platform Module (TPM) chips can be also utilized to impede access to sensitive information of an IoT device (e.g. source code, encryption keys, hardcoded passwords). Furthermore, the adoption of integrated instead of separate memory chips, in conjunction with the lack of debugging interfaces, can ensure that attempts aiming at retrieving sensitive information directly from the board components will result in damaging the chip thus destroying any data it contains.
- **Secure firmware update mechanisms:** Mechanisms that prevent updating a device with a tampered firmware should be in place, for example, by allowing only digitally signed firmware to be installed (e.g. via X.509 digital certificates) . Security mechanisms must also be in place to ensure that the firmware installed is the latest one in order to thwart firmware downgrade attacks. Vendors should also make sure that IoT appliances do not run on obsolete software and security updates are delivered timely. Additionally, proper informative signs on the IoT casing and/or systemic alerts should exist in order to ensure that the customer is aware of the exact time-period, after which, the IoT device is no longer supported from the official manufacturer.
- **Secure OS architecture:** Since updating the operating system of IoT devices is not always possible, their OS should be based on tested, minimized architectures that provide the least necessary services, to minimize the exposure to known and future OS vulnerabilities.
- **OS hardening:** The OS of IoT devices should be configured based on security hardening best practices and standards when possible, by enforcing mandatory access control mechanisms and least privilege access. Vendors should avoid reusing hardcoded passwords among different devices and platforms.

- **Use of secure APIs:** When developing application software for IoT devices, the developers should use only secure and tested APIs that provide tested software development libraries and prevent well-known software vulnerabilities (for instance buffer overflows and use of non-sanitized input).
- **Code auditing of application software:** IoT applications should be thoroughly tested by security experts, prior to the commercial deployment of the related IoT devices, using software security best practices. In this way, attacks related with application-layer vulnerabilities, like command injection, would be avoided.
- **Support for network security protocols:** IoT devices should implement at their network stack, at least as optional, network protocols that support security extensions for encryption, integrity and authentication for all wireless interfaces at all layers: At the link layer (e.g. the auxiliary security frame in IEEE 802.15.4), at the network layer (e.g. IPSec) or at the application layer (e.g. CoAP). Anti-DoS mechanisms on network equipment must also be in place (e.g. set custom volumetric thresholds for each network service, apply inbound/outbound packet filtering, monitor for abnormal network activity, utilize outsource available specialized security solutions).
- **Secure key management:** Devices should not rely on insecure key management mechanisms, such as the use of a common key embedded by the manufacturer in all devices of the same type, but only on tested secure key management techniques [231].
- **Secure key exchange protocols:** If key exchanged is based on symmetric cryptography, IoT devices should implement a secure key bootstrapping protocol. Key exchange protocols based on public key cryptography should be preferred. For example, those based on elliptic curve cryptography may be efficient for various IoT devices [248].
- **Device acquiring criteria:** The operators should favor IoT devices and vendors that utilize strong security controls, even if this implies some increase of device acquiring costs.
- **Secure change management:** The administrators should implement a procedure to rapidly integrate and deploy software and firmware updates provided by the IoT vendors.
- **Continuous security testing:** The administrators should integrate security testing of IoT devices in their lifecycle, e.g. vulnerability scanning and penetration testing. In particular, IoT devices must undergo an extensive security

testing in hardware, networks, I/O interfaces as well as application and cloud API services prior to installation to any secure environment whereas security assessments must be repeated at regular intervals.

- **Security standards enforcement:** The regulators and standardization bodies should enforce the use of IoT devices that comply with high security standards, at least for critical infrastructures and systems. USA, has declared the *Internet of Things Cybersecurity Improvement Act of 2017*[3], based on which NIST has defined a set of security guidelines for IoT devices purchased by the federal government [80]. Among others, the act defined minimum security requirements regarding vendors: Support of security patching, rely on industry standard protocols, prohibit hardcoded passwords or have any known security vulnerabilities. Recent sector/device specific guides such as the '*Cyber Security for Lighting Systems*'[4], released from US Department of Energy, can evaluate the cybersecurity risks that are associated with IoT devices including common types of attacks on such systems. Furthermore, a series of standards (UL 2900)[5], that helps improving the security of IoT devices by providing measurable criteria for the testing of network-connected devices that send, store, or transmit data, has been developed from the American National Standard Institute (ANSI)[6]. Among its publications ENISA has also published a tool[7] as well as a *Baseline security recommendations for IoT in the context of Critical Information Infrastructures* report [234] that aims in presenting a baseline of security measures necessary for the secure operation of IoT devices that are installed within critical infrastructures. It focuses on authorization and authentication mechanisms, data protection and compliance, cryptography, secure interfaces and network services, privacy by design as well as Third-Party relationships. In addition, NISTIR 8259 [79] describes basic recommendations to manufacturers, on how to establish cybersecurity features including the necessary security services to customers, for IoT devices that are equipped with at least a transducer (sensor or actuator) and at least one network interface (e.g. Zigbee, WiFi). Furthermore, in NISTIR 8259A [80] authors specify technical baselines in security areas regarding *Device Identification*, *Device Configuration*, *Software Update*, *Data Protection*, *Logical Access to Interfaces* and *Cybersecurity State Awareness.*

[3] https://www.congress.gov/bill/115th-congress/senate-bill/1691

[4] https://www.energy.gov/sites/prod/files/2018/06/f52/cyber_security_lighting.pdf

[5] https://www.shopulstandards.com/Catalog.aspx?UniqueKey=1&Catalog=1

[6] https://ansi.org/

[7] https://www.enisa.europa.eu/topics/iot-and-smart-infrastructures/iot/good-practices-for-iot-and-smart-infrastructures-tool/

10.1.3 Reducing the potential impact of connectivity paths

Since in IoT-enabled attacks the impact is usually related with critical systems that are connected in some way with the IoT device, we examine security controls that target to identify and "cut-off" hidden and subliminal attack paths. In particular:

- **Identify and document IoT dependencies:** The dependencies and interdependencies between IoT devices and critical systems should be identified and documented. For example, how the devices communicate directly with critical systems, or indirectly through aggregation points that are used for monitoring and control.
- **Re-examine "Bring-Your-Own-Device" policies:** Policies like BYOD should be re-examined to assure that potential subliminal attack paths against critical systems are not underestimated by the security policy.
- **Avoid unnecessary physical proximity:** Avoid installing IoT devices physically near critical systems, e.g. a smart thermostat inside the data center. If physical proximity is necessary, assure that the IoT devices do not create indirect and/or hidden attack paths against the critical systems [213]. IoT devices must be installed out-of-range with critical systems that share cyber-physical interfaces with common inputs/output types. In particular, critical systems equipped with wireless network interfaces such as WirelessHART should not co-exist (in-range) with IoT devices that utilize network interfaces with similar radio frequencies (e.g. 802.11.x, 802.15.4). Similarly, IoT devices with infrared outputs and/or IoT-enabled robotic machinery with moving capabilities/parts that can physically destroy/touch a critical component. In cases where the out-of-range criterion can not be applied, security controls must be in place (e.g. see also Table 6.10) as well as TEMPEST[8] shielding and even the use of Faraday cages.
- **Segment networks to avoid cascading impact:** When IoT devices are installed, examine the network design to assure proper network segmentation. For example, passive medical devices within a hospital should not be installed in the same local network with other IT systems. Proper segmentation of networks limits the exposure of mission critical systems, since it prevents threats like malware from easily spreading to mission critical systems. Moreover, it allows fine-tuning of access control and improves monitoring processes. Specific network access rules among different network segments must be defined in order

[8] Telecommunications Electronics Material Protected from Emanating Spurious Transmissions

to ensure that all unused ports are disabled whereas the adoption of Demilitarized Zones (DMZs) inhibits any attackers from create direct connections to the internal industrial networks.

- **Favor technology diversity:** Technology unification in hardware (e.g. processors) and network protocols is a cost efficient policy. However it may also mean that a single self-spreading worm or a hardware vulnerability is applicable to multiple IoT devices and networks thus leading to cascading effects. When possible, operators should consider acquiring diverse (but tested) IoT technologies to reduce this risk. Additionally, the adoption of transparent, open standards rather than proprietary technology, guarantees that any potential vulnerabilities can be identified and addressed on time by the security community.

10.2 Mitigating the risk of IoT-enabled attack paths in e-healthcare paradigm

After calculating the risk for cyber and cyber-physical AP scenarios in Chapter 9, the assessor can proceeded to the risk mitigation phase. Here, we are going to demonstrate how one can utilize our methodology presented in the aforementioned chapter in order to prioritize mitigation actions in the most effective way. In particular, we are going to first simulate a typical patch scenario where an organization would most likely implement in order to mitigate these risks: As the first logical step in a typical threat remediation process is to address the vulnerabilities found at the critical devices (targets). Therefore, the next stage is to patch the ICT equipment such as servers, workstations and crucial network equipment. The last step, which is considered as the most challenging one, is to address the vulnerabilities found on IoT devices.

As depicted in Figure 10.1, after patching all three critical systems there was a significant reduction from a total of 7179 to 4984 AP scenarios (31%). Especially for cyber AP scenarios there was a significant reduction (100% for `Very High`, 25% for `High`, 35% for `Moderate` and `Low`) whereas there was no reduction to cyber-physical ones, since, physical interactions with the target system do not usually rely on software vulnerabilities. During the next stage (ICT patch process), all AP scenarios related with high risk level were mitigated, leading to a significant reduction from 4984 to just 95 (just 6 cyber and 95 cyber-physical) AP scenarios. The numbers of cyber-physical AP scenarios where further reduced to just 10 after IoT devices' vulnerabilities were addressed. The residual risks after the whole patching process was completed, where mainly due to insider threats that corresponded to adversary types such as the *Disgruntled Healthcare Systems' Administrator* that has both logical and physical access to most hopsital's devices. All of the above are depicted in Figure 10.1.

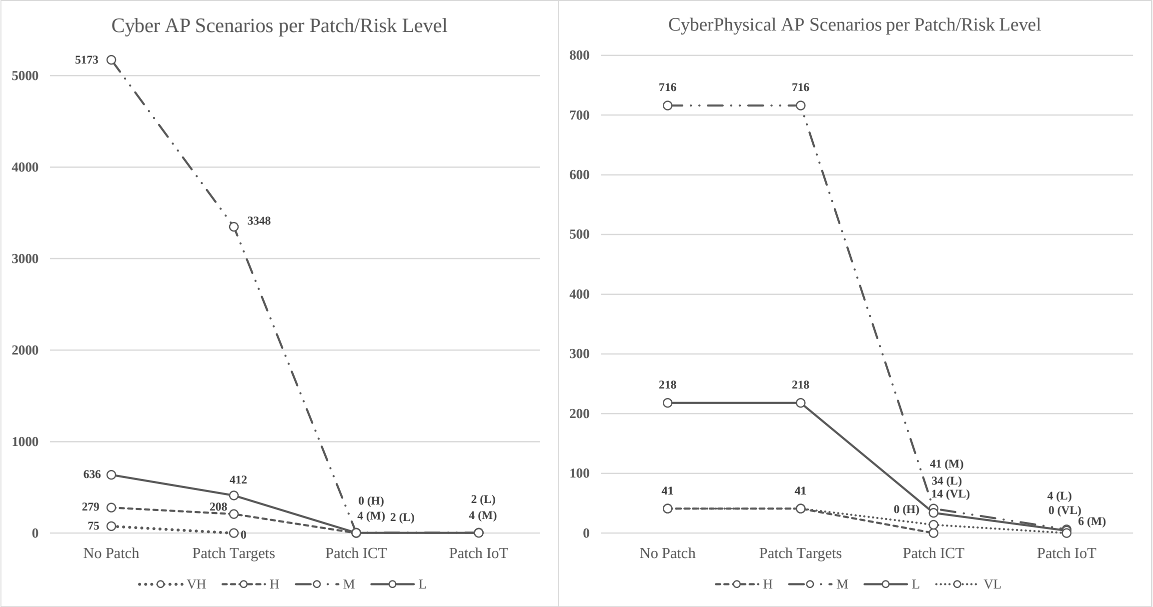


Figure 10.1: Risk level and multitude of attack path scenarios per patch level [260]

In order to further improve the mitigation process we can strategically use the available information from the risk assessment phase. In particular, from the available information in Table 9.6, we can classify the devices based on the multitude of AP scenarios (either as intermediate or entry node). We can then choose the first three devices with the highest score (in this case IDs **3**,**0** and **13**) and apply all security patches. We then run the simulation and discover a total reduction of 94% (from 7179 to just 396) for all AP scenarios (97% for cyber and 78,5% for cyber-physical). This, in turn, makes this approach a far more efficient way to reduce risks in terms of time and effort and can be used when a quick response is of utmost importance. In addition, an assessor may apply the available information to prioritize the security countermeasures against specific types of adversaries and/or specific types of cyber or physical interactions (e.g. mitigate risks from cyber criminals that are able to physically interact with near-patient medical devices from the Internet).

10.3 Research and implementation gap analysis

Based on the analysis of the examined cyber attack scenarios in Chapter 7, we summarize the relative research and implementation gaps, in comparison to the existing state-of-the-art security controls (also see Table 10.2). The inadequate implementation of security controls is usually due to the lack of security policy enforcement, the underestimation of the current threat landscape and budget constraints. Although the available security controls are not always sufficient to mitigate some of the novel advanced threats, the majority of the attack vectors could be properly mitigated if the existing security mechanisms and standards were properly implemented. The lack of regulation that would enforce critical system operators to use security tested, but usually more expensive, IoT devices also contributes to the implementation gaps.

IoT security is nowadays considered as one of the most active and evolving research domain. However, despite the recent state-of-the-art advances (e.g. [64, 114, 116, 280, 309, 310]) research gaps can be still identified in all the layers of IoT platforms [194]. For example, sophisticated attacks such as [232] demonstrate that existing physical proximity testing mechanisms, required for some security sensitive operations like firmware update, can be bypassed. Remote access and control of IoT devices, especially via cloud-based services [145], also require novel technologies like Blockchain [32,128], for distributed monitoring and auditing of IoT access. Hardware layer security research challenges involve, among others, the protection of IoT devices from novel side-channel attacks, which have been proven hard to deal with. At the software layer, trending attacks such as ransomware and botnets demonstrate the challenge for developing novel and effective protection mechanisms.

The constrained environment of IoT devices still requires the design of lightweight and protocol-specific network security mechanisms and protocols [108,218], including the support of efficient public key management, despite the recent advances [248].

Table 10.2: Gap analysis for IoT security: Research and implementation gaps [261]

Group	**Security controls**	**State-of-the-art**	**Ideal state (Research Gaps)**	**State of practice (Implementation Gaps)**
Physical access	- *Limit physical access to IoT* - *Monitor physical access to IoT* - *Avoid physical proximity*	- Standard physical protection and monitoring mechanisms may be applied. - Physical proximity testing may be required by IoT devices for some sensitive operations (e.g. firmware update).	- Recent attacks against current physical proximity testing mechanisms (e.g. [232] demonstrate the need for further research in this area.	- Physical access protection and monitoring is not a common practice.
Logical access	- *Avoid direct Internet access* - *Enforce proxy-based access* - *Secure remote access* - *Log and monitor access to IoT* - *Audit access to IoT*	- Current access control solutions for IoT include proximity-based, proxy-based and biometric solutions, among others. - Adoption of Security frameworks [32, 128] that use Blockchain technology.	- Further research for remote access control of IoT devices in cloud-based services is required. - There is a need for further research in authentication and access control especially for energy constrainted IoT devices.	- Thousands of IoT devices worldwide may be remotely accessed/administered with default passwords, due to lack of user awareness and/or defective policies and procedures. - Administration interfaces of (critical) IoT devices may be directly accessible through the Internet (proxy-based access not enforced).
Hardware	- *Tamper resistance mechanisms* - *Secure embedded crypto* - *Side-channel attack protection*	- Trusted platform modules as well as Physical Unclonable Functions (PUFs) integrated into the circuit, can support embedded hardware-based IoT authentication capabilities [64, 280, 309, 310]. - Resilience against side-channel attacks (e.g. DPA/CPA/Photonic) [114, 116].	- H/W integrity checks [117] represent some of the current active research challenges for tamper resistance and H/W security. - There is a need for novel security mechanisms against side-channel attacks.	- Strong hardware-layer security mechanisms are not a common practice, due to the extra costs.
Software	- *Firmware protection* - *Secure FW update* - *Secure OS architecture and hardening* - *Secure APIs* - *Code auditing*	- State of the art mechanisms include firmware signing, code obfuscation, protected boot process, secure coding and compiling techniques [161] and cloud services security (e.g. [145]).	- There is a need for novel cross-layer SW protection mechanisms of IoT devices (e.g. against ransomware attacks [212, 296]) and platforms [194].	- Software vulnerabilities, especially in low cost IoT devices, are commonly caused by non-tested development APIs. - Existent cross-layer vulnerabilities affect the software layer (e.g. weak tamper resistance may lead to firmware/OS tampering via unprotected debugging ports [31]).
Network	- *Link-layer security extensions* - *Network-layer security protocols* - *Secure key exchange and management* - *Network segmentation and architectures*	- Existing key management schemes and public key primitives for IoT can be found in [231, 248]. - Standard end-to-end security protocols, such as IPSec, may be applied through header compression.	- Securing IoT specific protocols at the routing layer (e.g. RPL) and at the application layer (e.g. CoAP) are open challenges [108, 218]. - There is a need for further research for lightweight cryptography and key management for IoT devices. - Novel network architectures are needed for increased resilience to cascading IoT-enabled attacks.	- Network-layer security mechanisms are commonly not supported by the device and/or not configured by the device operator. - Key management may rely on the use of a common (embedded) key for all the devices of a certain type, or on default keys.
Procedures	- *Secure change management* - *Device acquiring criteria* - *Continuous security testing* - *Security standards enforcement* - *Identify IoT dependencies* - *Re-examine BYOD policies* - *Favor technology diversity*	- Generic IT security management standards (such as [38, 235]) can be applied. - Sector-specific standards are being developed (e.g. [15]).	- Advanced methods for threat modeling based on dependency analysis are required. - There is a need to develop targeted security standards for IoT devices and sectors.	- The lack of standardization and regulation greatly contribute to the increase the security implementation gaps.

CHAPTER 11
CONCLUSIONS, LIMITATIONS AND FUTURE WORK

11.1 Conclusions

The goal of this thesis was to showcase the risks that stem out from the IoT-CI interactions and to address existing research gaps related with the identification and assessment of IoT-enabled, cyber-physical attack paths against critical infrastructures and services. To achieve this, first we identify IoT-enabled attacks by surveying both the relevant literature and real security incidents, as presented in Chapters 3 and 4. Then, in order to model the criticality of such attacks, in Chapter 5 we have developed a high-level security assessment framework, which is applied in the aforementioned attacks (Chapter 7). From the IoT-enabled attacks examined one can infer that direct attack scenarios favor SCADA environments and other target devices with interfaces that have direct Internet access. Indirect attack scenarios are most common in the transport and healthcare sectors. Additionally, attacks due to physical proximity fall into the category of 'direct' attack scenarios, since, the IoT device interacts directly with the target. Off-the-shelf smart home devices are usually responsible for all 'no-connectivity' attack scenarios, since they do not require any indirect or direct access with the target system. All of our assessed attack scenarios were characterized as of `Moderate` (39,1 %) and `High` (60,1 %) critically. The high rating is partly due to the application the highest possible impact in our analysis, as an attempt to capture the worst-case scenarios of such attacks.

In order to address the limitations presented in Subsection 2.2, in Chapter 2, we presented a low-level RA methodology. In particular, the latter takes as inputs IoT/ICT devices/interfaces, networking, location, physical - logical access and proximity in order to construct interaction lists in an source-driven, target-oriented approach. Then, in order to reduce both complexity and false positive results it utilizes existing vulnerabilities and environmental security controls to validate each identified interaction based on predefined criteria. In order to construct complex, cyber-physical attack vectors a recursive algorithm is applied to all assessed interaction lists. Finally, based on the existing vulnerabilities of the initial (*point-of-entry*) node, the business-wise impact of the target device and the threat likelihood of the corresponding threat agents, the overall risk is calculated for all individual attack vectors (`AV`). Additionally, a tool is developed and used to test the efficacy of the proposed RA methodology in IoT-enabled, PoC scenarios in healthcare and Smart City environments (Chapters 9 and 8).

From the analysis of the results presented in Section 8.4, it is possible to infer that the majority (73 %) of the smart-light enabled attacks in the Smart City PoC scenario were cyber-physical mainly due to the proximity of wireless, shared-spectrum, network interfaces ([213]) and functionality attack scenarios presented in [232]. On the other hand, in the healthcare PoC scenario the percentage of the latter represented just the 14,2 % of all attack paths, since, targets in the assessment scenario included and 'conventional' ICT systems such as EMR/EHR servers. The analysis of attacks presented in Chapters 3, 4, 8 and 9 showcased, except from cyber attack scenarios, several stealthy, cyber-physical, high criticality/risk attack paths as well. In particular, a closer look, revealed attacks in which the adversary could physically interact with the target system from a remote location (e.g. via the Internet). This is feasible mainly due to existing vulnerabilities found on the intermediate/initial nodes, the indirect connectivity among IoT/ICT enabling technologies, the proximity of wireless of IoT network interfaces that utilize the same spectrum (e.g. WiFi/ZigBee 2.4 GHz) with critical systems and/or the functionality features of IoT physical interfaces (e.g. by utilizing the available luminosity levels of a smart light bulb).

Finally, in Chapter 10 we apply the taxonomy presented in Chapter 5 to categorize proposed security controls in order to reduce the threat, vulnerability and/or impact level. Additionally, we utilize the results from our PoC scenarios in order to showcase the benefits of our low-level methodology when developing mitigation strategies.

11.2 Research limitations

During the development of our RA methodology we encountered multiple challenges, some of which we managed to address whereas several others can still be considered as open challenges. Limitations of this work may include, among others, the following:

No support of physical interaction types among intermediate nodes: The methodology does not examine how an adversary can exploit physical proximity in order to pivot through IoT/ICT devices, since, physical interaction types are only defined for direct (`Level-1`) ones with the target system.

No support for logical dependencies among IoT/ICT systems: Logical dependencies among traditional ICT systems and glsIoT devices during the interaction modelling and assessment phase are not examined.

Lack of software security control assessment: Although environmental information is taken into consideration during the evaluation process, it is mainly focused to network security controls and does not takes into consideration software security controls that could limit the amount of damage (impact) each vulnerability delivers to the corresponding device.

Limited vulnerability chaining: Serial exploitation of existing vulnerabilities is supported only for a specific set of vulnerabilities.

No support for reverse attack paths: During the construction of attack paths the recursive algorithm presented in Subsection 6.7 does not calculate the reverse interaction attack paths(`Level n-1` interactions → `Level-n` attack paths).

11.3 Future work

Future work should include the extension of the methodology in order to include logical interdependencies among IoT/ICT systems as well as physical interactions during the intermediate stages (propagation phase) of an attack scenario. Additionally, more actions must take place in order to further improve the automation of the initial process of collecting the required information via the use of software tools, since, even partially manual construction of the lists of assets and their corresponding interfaces, software, networks and proximity characteristics can be proved quite challenging in large, complex corporate environments.

Moreover, research is required in order to further automate the interaction assessment process regarding the identified SW and its corresponding vulnerabilities. To address this issue, we plan to develop our tool so as to include such feature in the near future. Regarding the vulnerability assessment, more vulnerability combination and chaining techniques must also be added to the methodology. For example, to include more realistic attack techniques, the Adversarial Tactics, Techniques, and Common Knowledge (ATT&CK) could be used to enrich the attack methods regarding the simultaneous exploitation of different vulnerability vectors. To address limitations in threat modelling identification and automation process, we have recently proposed an ontology that can unify information from an extensive collection of known cybersecurity datasets, semi-structured or unstructured (text) data from public security reports and environmental security information. The latter is gathered from network security tools and can be applied to networks and systems under assessment including information about threat actors and valid users of existing infrastructures [112].

BIBLIOGRAPHY

[1] Abie, H., and Balasingham, I. Risk-based adaptive security for smart iot in ehealth. In *Proceedings of the 7th International Conference on Body Area Networks* (2012), ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), pp. 269–275.

[2] Agadakos, I., Chen, C.-Y., Campanelli, M., Anantharaman, P., Hasan, M., Copos, B., Lepoint, T., Locasto, M., Ciocarlie, G. F., and Lindqvist, U. Jumping the air gap: Modeling cyber-physical attack paths in the internet-of-things. In *Proceedings of the 2017 Workshop on Cyber-Physical Systems Security and PrivaCy* (2017), ACM, pp. 37–48.

[3] Al Ghazo, A. T., Ibrahim, M., Ren, H., and Kumar, R. A2G2V: Automatic attack graph generation and visualization and its applications to computer and scada networks. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* (2019).

[4] Al-Sultan, S., Al-Doori, M. M., Al-Bayatti, A. H., and Zedan, H. A comprehensive survey on vehicular ad hoc network. *Journal of network and computer applications 37* (2014), 380–392.

[5] Alcaraz, C., Fernandez, G., and Carvajal, F. Security aspects of SCADA and DCS environments. *Critical Infrastructure Protection* (2012), 120–149.

[6] Alcaraz, C., and Lopez, J. Secure interoperability in cyber-physical systems. In *Security Solutions and Applied Cryptography in Smart Grid Communications, IGI Global, USA*. IGI Global, USA, 2017, ch. 8, pp. 137–158.

[7] Alcaraz, C., and Zeadally, S. Critical control system protection in the 21st century: Threats and solutions. *IEEE Computer 46*, 10 (2013 2013), 74 – 83.

[8] Alcaraz, C., and Zeadally, S. Critical infrastructure protection: Requirements and challenges for the 21st century. *International Journal of Critical Infrastructure Protection (IJCIP) 8* (01/2015 2015), 53–66.

[9] AlDairi, A., et al. Cyber security attacks on smart cities and associated mobile technologies. *Procedia Computer Science 109* (2017), 1086–1091.

[10] Alhanahnah, M., Stevens, C., and Bagheri, H. Scalable analysis of interaction threats in iot systems. In *Proceedings of the 29th ACM SIGSOFT International Symposium on Software Testing and Analysis* (2020), pp. 272–285.

[11] Andrade, R. O., Yoo, S. G., Tello-Oquendo, L., and Ortiz-Garcés, I. A comprehensive study of the iot cybersecurity in smart cities. *IEEE Access 8* (2020), 228922–228941.

[12] Andrea, M. Ge mds pulsenet hidden support account remote code execution vulnerability, 2015.

[13] Andrew, L. Hackers are holding San Francisco's light-rail system for ransom (The Verge), 2016.

[14] Anton, C. Industroyer: Biggest threat to industrial control systems since Stuxnet, 2017.

[15] Anton, C. Win32/Industroyer: A new threat for Industrial Control systems. Tech. rep., ESET, 2017.

[16] Assange, J. Vault 7: Cia hacking tools revealed. *WikiLeaks.(Mar. 2017). Retrieved Mar 7* (2017), 2017.

[17] Atamli, A. W., and Martin, A. Threat-based security analysis for the internet of things. In *Secure Internet of Things (SIoT), 2014 International Workshop on* (2014), IEEE, pp. 35–43.

[18] Atlam, H. F., Alenezi, A., Walters, R. J., Wills, G. B., and Daniel, J. Developing an adaptive risk-based access control model for the internet of things. In *2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)* (2017), pp. 655–661.

[19] Badenhop, C. W., Graham, S. R., Ramsey, B. W., Mullins, B. E., and Mailloux, L. O. The z-wave routing protocol and its security implications. *Computers & Security 68* (2017), 112–129.

[20] Badolato, E. V. Terrorism and the us energy infrastructure.

[21] Balduzzi, M., Wihoit, K., and Pasta, A. Hey captain, where's your ship? attacking vessel tracking systems for fun and profit. In *Hack in the Box (HITB) Security Conference in Asia* (2013), pp. 1–36.

[22] Ballano, M. AmosConnect: Maritime communications security has its flaws. *IOActive* (2017).

[23] Barcena, M. B., and Wueest, C. Insecurity in the internet of things. *Security Response, Symantec* (2015).

[24] Barker, W., and NIST, S. 800-60, revision 1. *Guide for Mapping Types of Information and Information systems to Security Categories* (2008).

[25] Barrett, M. P., et al. Framework for improving critical infrastructure cybersecurity version 1.1.

[26] Bateman, T. Police warning after drug traffickers' cyber-attack (The BBC), 2013.

[27] Beaulieu, R., Treatman-Clark, S., Shors, D., Weeks, B., Smith, J., and Wingers, L. The SIMON and SPECK lightweight block ciphers. In *Design Automation Conference (DAC), 2015 52nd ACM/EDAC/IEEE* (2015), IEEE, pp. 1–6.

[28] Beaumont, P. Cyber-risks in maritime container ports: An analysis of threats and simulation of impacts.

[29] Beyah, R., and Formby, D. Simulated attack shows ICS weakness (a new form of ransomware to take over control of a simulated water treatment plant), 2017.

[30] Biegacki, S., and VanGompel, D. The application of devicenet in process control. *ISA transactions 35*, 2 (1996), 169–176.

[31] Billy, R., and Jonathan, B. Security evaluation of the implantable cardiac device ecosystem architecture and implementation interdependencies. Tech. rep., WhiteScope, 2017.

[32] Biswas, K., and Muthukkumarasamy, V. Securing smart cities using blockchain technology. In *High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS), 2016 IEEE 18th International Conference on* (2016), IEEE, pp. 1392–1393.

[33] Blauw, H., Van Bon, A., Koops, R., and DeVries, J. Performance and safety of an integrated bihormonal artificial pancreas for fully automated glucose control at home. *Diabetes, Obesity and Metabolism 18*, 7 (2016), 671–677.

[34] Boddy, S., and Shattuck, J. The hunt for IoT: The rise of thinkbots (F5 Labs Technical Report), July 2017.

[35] Bormann, C., Castellani, A. P., and Shelby, Z. CoAP: An application protocol for billions of tiny Internet nodes. *IEEE Internet Computing 16*, 2 (2012), 62.

[36] Boyer, S. A. *SCADA: supervisory control and data acquisition.* International Society of Automation, 2009.

[37] BRET-MOUNET, F. All your solar panels are belong to me. *DEF CON 24* (2016), 4–7.

[38] Brewer, D. *An Introduction to ISO/IEC 27001:2013.* BSI British Standards Institution, 2013.

[39] Brian, G., and Fritz, Sands Team, T. M. Z. D. I. Hacker machine interface: The state of scada hmi vulnerabilities. *White paper, Trend Micro* (2017).

[40] Brooks, P. Ethernet/ip-industrial protocol. In *ETFA 2001. 8th International Conference on Emerging Technologies and Factory Automation. Proceedings (Cat. No. 01TH8597)* (2001), vol. 2, IEEE, pp. 505–514.

[41] Brown, J. E., Smith, N., and Sherfy, B. R. Decreasing mislabeled laboratory specimens using barcode technology and bedside printers. *Journal of nursing care quality 26*, 1 (2011), 13–21.

[42] Candid, W. How my TV got infected with ransomware and what you can learn from it, 2015.

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

[44] Cazorla, L., Alcaraz, C., and Lopez, J. Cyber stealth attacks in critical information infrastructures. *IEEE Systems Journal* (03/2016 2016), 1–15.

[45] Cerrudo, C. Hacking US traffic control systems. Presentation at DEFCON 22, 2014.

[46] Cerrudo, C. An emerging us (and world) threat: Cities wide open to cyber attacks. *Securing Smart Cities 17* (2015), 137–151.

[47] Cerrudo, C., and Apa, L. Hacking robots before skynet1. *IOActive Website* (2017).

[48] Cesar, C., and Lucas, A. Hacking robots before Skynet (IOActive).

[49] Chapman, A. Hacking into internet connected light bulbs, 2014.

[50] Checkoway, S., McCoy, D., Kantor, B., Anderson, D., Shacham, H., Savage, S., Koscher, K., Czeskis, A., Roesner, F., Kohno, T., et al. Comprehensive experimental analyses of automotive attack surfaces. In *USENIX Security Symposium* (2011), San Francisco, pp. 77–92.

[51] Chen, B., Schmittner, C., Ma, Z., Temple, W. G., Dong, X., Jones, D. L., and Sanders, W. H. Security analysis of urban railway systems: the

need for a cyber-physical perspective. In *International Conference on Computer Safety, Reliability & Security* (2015), Springer, pp. 277–290.

[52] Chen, P., Desmet, L., and Huygens, C. A study on advanced persistent threats. In *IFIP International Conference on Communications and Multimedia Security* (2014), Springer, pp. 63–72.

[53] Cherepanov, A. Win32/industroyer: A new threat for industrial control systems. *White paper, ESET (June 2017)* (2017).

[54] Cherepanov, A., and Lipovsky, R. Industroyer: Biggest threat to industrial control systems since stuxnet. *WeLiveSecurity, ESET 12* (2017).

[55] Chris, G. Cyber attack hits German train stations as hackers target Deutsche Bahn (The Telegraph), 2017.

[56] Clark, R. M., and Deininger, R. A. Protecting the nation's critical infrastructure: the vulnerability of us water supply systems. *Journal of contingencies and crisis management 8*, 2 (2000), 73–80.

[57] Clarke, G., Reynders, D., and Wright, E. *Practical modern SCADA protocols: DNP3, 60870.5 and related systems.* Newnes, 2004.

[58] Cobb, S. 10 things to know about the October 21 IoT DDoS attacks, 2016.

[59] Colin, O. A lightbulb worm? Details of the Philips Hue smart lighting design (Black Hat USA 2016 White Paper).

[60] Corno, F., and Sanaullah, M. Modeling and formal verification of smart environments. *Security and Communication Networks 7*, 10 (2014), 1582–1598.

[61] Costin, A., and Francillon, A. Ghost is in the air (traffic). *Black Hat USA (July 2012)* (2012), 1–9.

[62] Cowan, C., Wagle, F., Pu, C., Beattie, S., and Walpole, J. Buffer overflows: Attacks and defenses for the vulnerability of the decade. In *DARPA Information Survivability Conference and Exposition, 2000. DISCEX'00. Proceedings* (2000), vol. 2, IEEE, pp. 119–129.

[63] Da Xu, L., He, W., and Li, S. Internet of things in industries: A survey. *IEEE Transactions on Industrial Informatics 10*, 4 (2014), 2233–2243.

[64] Danese, A., Pravadelli, G., and Bertacco, V. DOVE: pinpointing firmware security vulnerabilities via symbolic control flow assertion mining (work-in-progress). In *Proceedings of the Twelfth IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis Companion* (2017), ACM, p. 9.

[65] Darwish, S., Nouretdinov, I., and Wolthusen, S. D. Towards composable threat assessment for medical iot (miot). *Procedia Computer Science 113* (2017), 627–632.

[66] Demosthenous, A. Advances in microelectronics for implantable medical devices. *Advances in Electronics 2014* (2014).

[67] Dhanjani, N. *Abusing the internet of things: blackouts, freakouts, and stakeouts.* " O'Reilly Media, Inc.", 2015.

[68] Dhyani, I., Goel, N., Sharma, G., and Mallick, B. A reliable tactic for detecting black hole attack in vehicular Ad Hoc networks. In *Advances in Computer and Computational Sciences*. Springer, 2017, pp. 333–343.

[69] Dimitrakopoulos, G., and Demestichas, P. Intelligent transportation systems. *IEEE Vehicular Technology Magazine 5*, 1 (2010), 77–84.

[70] DiRenzo, J., Goward, D. A., and Roberts, F. S. The little-known challenge of maritime cyber security. In *Information, Intelligence, Systems and Applications (IISA), 2015 6th International Conference on* (2015), IEEE, pp. 1–5.

[71] Donderi, D. C., Mercer, R., Hong, M. B., and Skinner, D. Simulated navigation performance with marine electronic chart and information display systems (ecdis). *The Journal of Navigation 57*, 2 (2004), 189–202.

[72] Dong, H., Ning, B., Cai, B., and Hou, Z. Automatic train control system development and simulation for high-speed railways. *IEEE circuits and systems magazine 10*, 2 (2010), 6–18.

[73] Dorsemaine, B., Gaulier, J.-P., Wary, J.-P., Kheir, N., and Urien, P. A new threat assessment method for integrating an iot infrastructure in an information system. In *Distributed Computing Systems Workshops (ICDCSW), 2017 IEEE 37th International Conference on* (2017), IEEE, pp. 105–112.

[74] Dragos Inc. Crashoverride analysis of the threat to electric grid operations, 2017.

[75] Dubrova, E. Anti-tamper techniques. *KTH Royal Institute of Technology, Sweden* (2018).

[76] Eden, T. The absolute horror of WiFi light switches, 2016.

[77] ENISA. European Union Agency for Network and Information Security. `https://www.enisa.europa.eu/topics/threat-risk-management/threats-and-trends`, retrieved on January 2022, 2021.

[78] Erdősi, P. M. The common vulnerability scoring system (cvss) generations–usefulness and deficiencies.

[79] Fagan, M., Fagan, M., Megas, K. N., Scarfone, K., and Smith, M. *Foundational cybersecurity activities for iot device manufacturers.* US Department of Commerce, National Institute of Standards and Technology, 2020.

[80] Fagan, M., Marron, J., Brady, K., Cuthill, B., Megas, K., and Herold, R. Iot device cybersecurity guidance for the federal government: Establishing iot device cybersecurity requirements. Tech. rep., National Institute of Standards and Technology, 2020.

[81] Falliere, N., Murchu, L. O., and Chien, E. W32. stuxnet dossier. *White paper, Symantec Corp., Security Response 5*, 6 (2011), 29.

[82] Farhangi, H. The path of the smart grid. *IEEE power and energy magazine 8*, 1 (2010).

[83] Farraj, A. K., and Kundur, D. On using energy storage systems in switching attacks that destabilize smart grid systems. In *Innovative Smart Grid Technologies Conference (ISGT)* (2015), IEEE, pp. 1–5.

[84] Fernandes, E., Jung, J., and Prakash, A. Security analysis of emerging smart home applications. In *Security and Privacy, 2016 IEEE Symposium on* (2016), IEEE, pp. 636–654.

[85] FIRST.Org. *Common Vulnerability Scoring System v3.1: User Guide*, 2019.

[86] Fisher, D. What's on TV tonight? ransomware, 2016.

[87] for Network, E. U. A., and (ENISA), I. S. Main incidents in the eu and worldwide - enisa threat landscape, 2020.

[88] Force, J. T. Security and privacy controls for information systems and organizations. Tech. rep., National Institute of Standards and Technology, 2017.

[89] Force, J. T., and Initiative, T. Security and privacy controls for federal information systems and organizations. *NIST Special Publication 800*, 53 (2013), 8–13.

[90] Formby, D., Durbha, S., and Beyah, R. Out of control: Ransomware for industrial control systems, 2017.

[91] Fouladi, B., and Ghanoun, S. Honey, I'm home!! Hacking Z-Wave home automation systems. *Black Hat USA* (2013), 1–53.

[92] Fox-Brewster, T. Medical devices hit by ransomware for the first time in US hospitals (Forbes), 2017.

[93] Furnell, S. M., and Warren, M. J. Computer hacking and cyber terrorism: The real threats in the new millennium? *Computers & Security 18*, 1 (1999), 28–34.

[94] Garcia, F. D., Oswald, D., Kasper, T., and Pavlidès, P. Lock it and still lose it–on the (in) security of automotive remote keyless entry systems. In *25th USENIX Security Symposium* (2016), pp. 929–944.

[95] Garrett, M. I stayed in a hotel with android lightswitches and it was just as bad as you'd imagine, 2016.

[96] Gayle, D., Topping, A., Sample, I., Marsh, S., and Vikram, D. NHS seeks to recover from global cyber-attack as security concerns resurface (The Guardian), 2017.

[97] Ge, M., Hong, J. B., Guttmann, W., and Kim, D. S. A framework for automating security analysis of the internet of things. *Journal of Network and Computer Applications 83* (2017), 12–27.

[98] George, G., and Thampi, S. M. Vulnerability-based risk assessment and mitigation strategies for edge devices in the internet of things. *Pervasive and Mobile Computing 59* (2019), 101068.

[99] George, K. 2021 global threat report, 2021.

[100] Ghena, B., Beyer, W., Hillaker, A., Pevarnek, J., and Halderman, J. A. Green lights forever: Analyzing the security of traffic infrastructure. *WOOT 14* (2014), 7–7.

[101] Gollakota, S., Hassanieh, H., Ransford, B., Katabi, D., and Fu, K. They can hear your heartbeats: non-invasive security for implantable medical devices. *ACM SIGCOMM Computer Communication Review 41*, 4 (2011), 2–13.

[102] Gomez, C., and Paradells, J. Wireless home automation networks: A survey of architectures and technologies. *IEEE Communications Magazine 48*, 6 (2010).

[103] González, I., Calderón, A. J., Figueiredo, J., and Sousa, J. A literature survey on open platform communications (opc) applied to advanced industrial environments. *Electronics 8*, 5 (2019), 510.

[104] Goodin, D. Hackers trigger yet another power outage in Ukraine, 2017.

[105] Goodin, D. Hackers trigger yet another power outage in Ukraine, 2017.

[106] Goodin, D. Leak of >1,700 valid passwords could make the IoT mess much worse (Ars Technica), 2017.

[107] Gordaychik, S., Timorin, A., and Gritsai, G. T. The great train cyber robbery. Presentation at the Chaos Communication Congress (CCC), 2015.

[108] Granjal, J., Monteiro, E., and Silva, J. S. Security for the internet of things: a survey of existing protocols and open research issues. *IEEE Communications Surveys & Tutorials 17*, 3 (2015), 1294–1312.

[109] Greenberg, A. Hackers remotely kill a jeep on the highway—with me in it, 2015.

[110] Greenberg, A. Hack brief: Hackers targeted a US nuclear plant (but don't panic yet), 2017.

[111] Greene, T. How the Dyn DDoS attack unfolded, 2016.

[112] Grigoriadis, C., Berzovitis, A. M., Stellios, I., and Kotzanikolaou, P. A cybersecurity ontology to support risk information gathering in cyber-physical systems. In *European Symposium on Research in Computer Security* (2021), Springer, pp. 23–39.

[113] Gritzalis, D., Iseppi, G., Mylonas, A., and Stavrou, V. Exiting the risk assessment maze: A meta-survey. *ACM Computing Surveys (CSUR) 51*, 1 (2018), 11.

[114] Gross, H., Mangard, S., and Korak, T. An efficient side-channel protected AES implementation with arbitrary protection order. In *Cryptographers' Track at the RSA Conference* (2017), Springer, pp. 95–112.

[115] Gruber, E. Verifying aslr, dep, and safeseh with powershell. *blog, NetSPI 23* (2014).

[116] Gruss, D., Lettner, J., Schuster, F., Ohrimenko, O., Haller, I., and Costa, M. Strong and efficient cache side-channel protection using hardware transactional memory. In *USENIX Security Symposium* (2017).

[117] Guin, U., Bhunia, S., Forte, D., and Tehranipoor, M. M. Sma: A system-level mutual authentication for protecting electronic hardware and firmware. *IEEE Transactions on Dependable and Secure Computing 14*, 3 (2017), 265–278.

[118] Gungor, V. C., Sahin, D., Kocak, T., Ergut, S., Buccella, C., Cecati, C., and Hancke, G. P. Smart grid technologies: Communication technologies and standards. *IEEE transactions on Industrial informatics 7*, 4 (2011), 529–539.

[119] Guri, M., and Bykhovsky, D. air-jumper: Covert air-gap exfiltration/infiltration via security cameras & infrared (ir). *Computers & Security 82* (2019), 15–29.

[120] Halperin, D., Heydt-Benjamin, T. S., Ransford, B., Clark, S. S., Defend, B., Morgan, W., Fu, K., Kohno, T., and Maisel, W. H. Pacemakers and implantable cardiac defibrillators: Software radio attacks and zero-power defenses. In *2008 IEEE Symposium on Security and Privacy (S&P 2008)* (2008), IEEE, pp. 129–142.

[121] HD, M. R. The internet of gas station tank gauges, 2015.

[122] Heer, T., Garcia-Morchon, O., Hummen, R., Keoh, S. L., Kumar, S. S., and Wehrle, K. Security challenges in the IP-based Internet of Things. *Wireless Personal Communications 61*, 3 (2011), 527–542.

[123] Hermann, M., Pentek, T., and Otto, B. Design principles for industrie 4.0 scenarios. In *2016 49th Hawaii international conference on system sciences (HICSS)* (2016), IEEE, pp. 3928–3937.

[124] Hernandez, G., Arias, O., Buentello, D., and Jin, Y. Smart nest thermostat: A smart spy in your home. *Black Hat USA* (2014), 1–8.

[125] Hong, J., and Kim, D.-S. Harms: Hierarchical attack representation models for network security analysis.

[126] Huang, H. *PACS and imaging informatics: Basic principles and applications.* John Wiley & Sons, 2011.

[127] Huang, S.-C., Chen, B.-H., Chou, S.-K., Hwang, J.-N., and Lee, K.-H. Smart car [application notes]. *IEEE Computational Intelligence Magazine 11*, 4 (2016), 46–58.

[128] Huh, S., Cho, S., and Kim, S. Managing IoT devices using Blockchain platform. In *Advanced Communication Technology (ICACT), 2017 19th International Conference on* (2017), IEEE, pp. 464–467.

[129] ICS-CERT. The Industrial Control Systems Cyber Emergency Response Team. `https://ics-cert.us-cert.gov`, retrieved on January 2022, 2021.

[130] Ifigeneia, L., Marianthi, T., Eleni, T., and Apostolos, M. Enisa threat landscape 2021. *ENISA, Oct* (2021).

[131] Independent Security Evaluators (Technical Report). Securing hospitals: A research study and blueprint. Tech. rep., 2016.

[132] Infrastructures, C. F. P. A. The report of the president's commission on critical infrastructure protection. *Washington, DC* (1997).

[133] Islam, S. R., Kwak, D., Kabir, M. H., Hossain, M., and Kwak, K.-S. The internet of things for health care: a comprehensive survey. *IEEE Access 3* (2015), 678–708.

[134] ISO, E. Iec 27005: 2018 (en) information technology–security techniques–information security risk management switzerland. *ISO/IEC* (2018).

[135] Joe, L. Sophos 2022 threat report: Interrelated threats target an interdependent world, 2021.

[136] Jon, H. R. The internet of gas station tank gauges – take 2, 2015.

[137] Kang, J., Yu, R., Huang, X., Maharjan, S., Zhang, Y., and Hossain, E. Enabling localized peer-to-peer electricity trading among plug-in hybrid electric vehicles using consortium blockchains. *IEEE Transactions on Industrial Informatics 13*, 6 (Dec 2017), 3154–3164.

[138] Kaspersky, E. The man who found stuxnet–sergey ulasen in the spotlight. *Nota Bene 2* (2011).

[139] Kayas, G., Hossain, M., Payton, J., and Islam, S. R. An overview of upnp-based iot security: Threats, vulnerabilities, and prospective solutions.

In *2020 11th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)* (2020), IEEE, pp. 0452–0460.

[140] Kelly, S. Major cyberattacks on healthcare grew 63% in 2016, 2016.

[141] Keyhani, A., and Marwali, M. *Smart power grids.* Springer, 2012.

[142] Khera, M. Think like a hacker insights on the latest attack vectors (and security controls) for medical device applications. *Journal of Diabetes Science and Technology* (2016), 1932296816677576.

[143] Kirk, J. Pacemaker hack can deliver deadly 830-Volt jolt, 2012.

[144] Klein, R. AATS security: Risk assessment to ensure aviation safety: Threat-scenario-based hazard analysis and risk assessment. In *Integrated Communication, Navigation, and Surveillance Conference (ICNS), 2015* (2015), IEEE, pp. 1–15.

[145] Klein, S. Azure event hubs. In *IoT Solutions in Microsoft's Azure IoT Suite.* Springer, 2017, pp. 273–289.

[146] Kocher, P., Jaffe, J., and Jun, B. Differential power analysis. In *Annual International Cryptology Conference* (1999), Springer, pp. 388–397.

[147] Kocher, P., Jaffe, J., Jun, B., and Rohatgi, P. Introduction to differential power analysis. *Journal of Cryptographic Engineering 1*, 1 (2011), 5–27.

[148] Komninos, N., Philippou, E., and Pitsillides, A. Survey in smart grid and smart home security: Issues, challenges and countermeasures. *IEEE Communications Surveys & Tutorials 16*, 4 (2014), 1933–1954.

[149] Koscher, K., Czeskis, A., Roesner, F., Patel, S., Kohno, T., Checkoway, S., McCoy, D., Kantor, B., Anderson, D., Shacham, H., et al. Experimental security analysis of a modern automobile. In *Security and Privacy (SP), 2010 IEEE Symposium on* (2010), IEEE, pp. 447–462.

[150] Kott, A., Ludwig, J., and Lange, M. Assessing mission impact of cyberattacks: toward a model-driven paradigm. *IEEE Security & Privacy*, 5 (2017), 65–74.

[151] Kott, A., Wang, C., and Erbacher, R. F. *Cyber defense and situational awareness*, vol. 62. Springer, 2015.

[152] Kovaks, E. Trains vulnerable to hacker attacks (Securityweek), 2015.

[153] Kraus, D., Leitgeb, E., Plank, T., and Löschnigg, M. Replacement of the controller area network (CAN) protocol for future automotive bus system solutions by substitution via optical networks. In *Transparent Optical Networks (ICTON), 2016 18th International Conference on* (2016), IEEE, pp. 1–8.

[154] KrebsonSecurity. Fbi: Smart meter hacks likely to spread, 2012.

[155] Kushalnagar, N., Montenegro, G., and Schumacher, C. IPv6 over low-power wireless personal area networks (6LoWPANs): overview, assumptions, problem statement, and goals. Tech. rep., 2007.

[156] Kushner, D. The real story of stuxnet. *ieee Spectrum 50*, 3 (2013), 48–53.

[157] Kyle, W., and Stephen, H. The gaspot experiment: Unexamined perils in using gas-tank-monitoring systems. Tech. rep., TrendMicro - TrendLabs, 2015.

[158] Lasi, H., Fettke, P., Kemper, H.-G., Feld, T., and Hoffmann, M. Industry 4.0. *Business & information systems engineering 6*, 4 (2014), 239–242.

[159] Lee, R. Crashoverride: Analysis of the threat to electric grid operations. *Dragos Inc., March* (2017).

[160] Lee, R. M., Assante, M. J., and Conway, T. Analysis of the cyber attack on the Ukrainian power grid. *SANS Industrial Control Systems* (2016).

[161] Lee, Y., Jeong, J., and Son, Y. Design and implementation of the secure compiler and virtual machine for developing secure IoT services. *Future Generation Computer Systems 76* (2017), 350–357.

[162] Leen, G., and Heffernan, D. Expanding automotive electronic systems. *Computer 35*, 1 (2002), 88–93.

[163] Lerche, C., Hartke, K., and Kovatsch, M. Industry adoption of the internet of things: A constrained application protocol survey. In *Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies & Factory Automation (ETFA 2012)* (2012), IEEE, pp. 1–6.

[164] Levy-Bencheton, C., and Darra, E. Cyber security and resilience of intelligent public transport: good practices and recommendations. *ENISA* (December 2015).

[165] Leyden, J. Polish teen derails tram after hacking train network. *The Register 11* (2008).

[166] Leyden, J. Airports' passport controls shut down by 'malware' (The Register), 2013.

[167] Li, C., Zhang, M., Raghunathan, A., and Jha, N. K. Attacking and defending a diabetes therapy system. In *Security and Privacy for Implantable Medical Devices*. Springer, 2014, pp. 175–193.

[168] Liang, G., Weller, S. R., Zhao, J., Luo, F., and Dong, Z. Y. The 2015 ukraine blackout: Implications for false data injection attacks. *IEEE Transactions on Power Systems 32*, 4 (2017), 3317–3318.

[169] Liang, G., Zhao, J., Luo, F., Weller, S., and Dong, Z. Y. A review of false data injection attacks against modern power systems. *IEEE Transactions on Smart Grid* (2017).

[170] Liao, G.-Y., Chen, Y.-J., Lu, W.-C., and Cheng, T.-C. Toward authenticating the master in the MODBUS protocol. *IEEE Transactions on Power Delivery 23*, 4 (2008), 2628–2629.

[171] Liebowitz, J., and Schaller, R. Biological warfare: Tampering with implantable medical devices. *IT Professional 17*, 5 (2015), 70–72.

[172] Lisovich, M. A., Mulligan, D. K., and Wicker, S. B. Inferring personal information from demand-response systems. *IEEE Security & Privacy 8*, 1 (2010).

[173] Liu, C., Zhang, Y., Zeng, J., Peng, L., and Chen, R. Research on dynamical security risk assessment for the internet of things inspired by immunology. In *Natural Computation (ICNC), 2012 Eighth International Conference on* (2012), IEEE, pp. 874–878.

[174] Liu, H., Spink, T., and Patras, P. Uncovering security vulnerabilities in the belkin wemo home automation ecosystem. In *2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)* (2019), IEEE, pp. 894–899.

[175] Liu, J., Lim, K. W., Ho, W. K., Tan, K. C., Tay, A., and Srinivasan, R. Using the opc standard for real-time process monitoring and control. *IEEE software 22*, 6 (2005), 54–59.

[176] Liu, J., Zhang, S., Sun, W., and Shi, Y. In-vehicle network attacks and countermeasures: Challenges and future directions. *IEEE Network 31*, 5 (2017), 50–58.

[177] Liu, X., and Li, Z. Local topology attacks in smart grids. *IEEE Transactions on Smart Grid* (2017).

[178] Livadas, C., Lygeros, J., and Lynch, N. A. High-level modeling and analysis of the traffic alert and collision avoidance system (TCAS). *Proceedings of the IEEE 88*, 7 (2000), 926–948.

[179] Lomas, N. Critical flaw identified in zigbee smart home devices, 2015.

[180] Maggi, F., Quarta, D., Pogliani, M., Polino, M., Zanchettin, A. M., and Zanero, S. Rogue robots: Testing the limits of an industrial robot's security. Tech. rep., Trend Micro, Politecnico di Milano, 2017.

[181] Maggi, F., Quarta, D., Pogliani, M., Polino, M., Zanchettin, A. M., and Zanero, S. Rogue robots: Testing the limits of an industrial robot's security. Tech. rep., Technical report, Trend Micro, Politecnico di Milano, 2017.

[182] Marin, E., Singelée, D., Garcia, F. D., Chothia, T., Willems, R., and Preneel, B. On the (in) security of the latest generation implantable cardiac defibrillators and how to secure them. In *Proceedings of the 32nd Annual Conference on Computer Security Applications* (2016), ACM, pp. 226–236.

[183] Marmol, F. G., Sorge, C., Ugus, O., and Pérez, G. M. Do not snoop my habits: preserving privacy in the smart grid. *IEEE Communications Magazine 50*, 5 (2012).

[184] Martins, D., and Guyennet, H. Wireless sensor network attacks and security mechanisms: A short survey. In *Network-Based Information Systems (NBiS), 2010 13th International Conference on* (2010), IEEE, pp. 313–320.

[185] Mathy, V., and Frank, P. Key reinstallation attacks: Forcing nonce reuse in wpa2. *DistriNet* (2017).

[186] Maurer, M., Gerdes, J. C., Lenz, B., Winner, H., et al. *Autonomous driving*. Springer, 2016.

[187] McLaughlin, S., Podkuiko, D., Miadzvezhanka, S., Delozier, A., and McDaniel, P. Multi-vendor penetration testing in the advanced metering infrastructure. In *Proceedings of the 26th Annual Computer Security Applications Conference* (2010), ACM, pp. 107–116.

[188] Mearian, L. With 15 Dollars in Radio Shack parts, 14-year-old hacks a car, 2015.

[189] Mi, X., Qian, F., Zhang, Y., and Wang, X. An empirical characterization of ifttt: ecosystem, usage, and performance. In *Proceedings of the 2017 Internet Measurement Conference* (2017), pp. 398–404.

[190] Miller, C. The legitimate vulnerability market: the secretive world of 0-day exploit sales. In *WEIS* (2007).

[191] Miller, C., and Valasek, C. Adventures in automotive networks and control units. *Def Con 21* (2013), 260–264.

[192] Miller, C., and Valasek, C. A survey of remote automotive attack surfaces. *black hat USA 2014* (2014), 94.

[193] Miller, C., and Valasek, C. Remote exploitation of an unaltered passenger vehicle. *Black Hat USA* (2015), 1–91.

[194] Mineraud, J., Mazhelis, O., Su, X., and Tarkoma, S. A gap analysis of internet-of-things platforms. *Computer Communications 89* (2016), 5–16.

[195] Mohsin, M., Anwar, Z., Husari, G., Al-Shaer, E., and Rahman, M. A. Iotsat: A formal framework for security analysis of the internet of things (iot). In *2016 IEEE conference on communications and network security (CNS)* (2016), IEEE, pp. 180–188.

[196] Momoh, J. *Smart grid: fundamentals of design and analysis*, vol. 63. John Wiley & Sons, 2012.

[197] Montenegro, G., Kushalnagar, N., Hui, J., and Culler, D. Transmission of IPv6 packets over IEEE 802.15.4 networks. Tech. rep., 2007.

[198] Morgner, P., Mattejat, S., and Benenson, Z. All your bulbs are belong to us: Investigating the current state of security in connected lighting systems. *arXiv preprint arXiv:1608.03732* (2016).

[199] Mousavian, S., Erol-Kantarci, M., Wu, L., and Ortmeyer, T. A risk-based optimization model for electric vehicle infrastructure response to cyber attacks. *IEEE Transactions on Smart Grid PP*, 99 (2017), 1–1.

[200] Müller, C., Armknecht, F., Benenson, Z., and Morgner, P. On the security of the zigbee light link touchlink commissioning procedure. In *Sicherheit* (2016).

[201] Mulligan, G. The 6lowpan architecture. In *Proceedings of the 4th workshop on Embedded networked sensors* (2007), pp. 78–82.

[202] Munro, K. OSINT from ship satcoms, 2017.

[203] Munro, K., and Lodge, D. Hacking the Mitsubishi Outlander PHEV hybrid, 2016.

[204] Musashi, Y., Kumagai, M., Kubota, S., and Sugitani, K. Detection of kaminsky dns cache poisoning attack. In *Intelligent Networks and Intelligent Systems (ICINIS), 2011 4th International Conference on* (2011), IEEE, pp. 121–124.

[205] Mwasilu, F., Justo, J. J., Kim, E.-K., Do, T. D., and Jung, J.-W. Electric vehicles and smart grid interaction: A review on vehicle to grid and renewable energy sources integration. *Renewable and Sustainable Energy Reviews 34* (2014), 501–516.

[206] Nagpal, B., Sharma, P., Chauhan, N., and Panesar, A. DDoS tools: Classification, analysis and comparison. In *Computing for Sustainable Global Development (INDIACom), 2015 2nd International Conference on* (2015), IEEE, pp. 342–346.

[207] National Institute of Standards and Technology (NIST). Standards for security categorization of federal information and information systems, FIPS PUB 199. http://csrc.nist.gov/publications/fips/fips199/FIPS-PUB-199-final.pdf, 2004.

[208] Nawir, M., Amir, A., Yaakob, N., and Lynn, O. B. Internet of things (iot): Taxonomy of security attacks. In *2016 3rd International Conference on Electronic Design (ICED)* (Aug 2016), pp. 321–326.

[209] Neisse, R., Steri, G., Fovino, I. N., and Baldini, G. Seckit: a model-based security toolkit for the internet of things. *Computers & Security 54* (2015), 60–76.

[210] Nelson, N. The impact of dragonfly malware on industrial control systems. *SANS Institute* (2016).

[211] Netronics. Cando: Can bus analyzer, 2021.

[212] Nixon, A., Costello, J., and Wilkholm, Z. An after-action analysis of the Mirai botnet attacks on Dyn, 2016.

[213] O'Flynn, C. P. Message denial and alteration on IEEE 802.15.4 low-power radio networks. In *New Technologies, Mobility and Security (NTMS), 2011 4th IFIP International Conference on* (2011), IEEE, pp. 1–5.

[214] Owens, C. Stranger hacks family's baby monitor and talks to child at night, 2016.

[215] Pascoe, R. D., and Eichorn, T. N. What is communication-based train control? *IEEE Vehicular Technology Magazine 4*, 4 (2009).

[216] Petersen, S., and Carlsen, S. Wirelesshart versus isa100. 11a: The format war hits the factory floor. *IEEE Industrial Electronics Magazine 5*, 4 (2011), 23–34.

[217] Petit, J., Stottelaar, B., Feiri, M., and Kargl, F. Remote attacks on automated vehicles sensors: Experiments on camera and Lidar. *Black Hat Europe 11* (2015), 1–13.

[218] Pongle, P., and Chavan, G. A survey: Attacks on RPL and 6LoWPAN in IoT. In *Pervasive Computing (ICPC), 2015 International Conference on* (2015), IEEE, pp. 1–6.

[219] Ponsam, J. G., and Srinivasan, R. A survey on MANET security challenges, attacks and its countermeasures. *International Journal of Emerging Trends & Technology in Computer Science (IJETTCS) 3*, 1 (2014).

[220] Proofpoint. More than 750,000 phishing and spam emails launched from "thingbots" including televisions, fridge, 2014.

[221] Proofpoint. Protecting patients, providers and payers 2019 healthcare threat report, 2019.

[222] ProofPoint. 2020 healthcare threat landscape, 2020.

[223] Quarta, D., Pogliani, M., Polino, M., Maggi, F., Zanchettin, A. M., and Zanero, S. An experimental security analysis of an industrial robot controller. In *Security and Privacy (SP), 2017 IEEE Symposium on* (2017), IEEE, pp. 268–286.

[224] Radcliffe, J. Hacking medical devices for fun and insulin: Breaking the human SCADA system. *Black Hat USA Conference, White Paper* (2011).

[225] Redmond, F. E. *Dcom: Microsoft Distributed Component Object Model with Cdrom*. IDG Books Worldwide, Inc., 1997.

[226] Reem, N. Chrysler recalls 1.4 million cars after remote hacking of jeep (CNBC), 2015.

[227] Reziouk, A., Laurent, E., and Demay, J.-C. Practical security overview of ieee 802.15. 4. In *2016 International Conference on Engineering & MIS (ICEMIS)* (2016), IEEE, pp. 1–9.

[228] Robert, L., and Anton, C. Blackenergy trojan strikes again: Ukrainian electric power industry, 2016.

[229] Robert, T. M. The report of thepresident's commissionon critical infrastructure protection. *CRITICAL FOUNDATIONS PROTECTING AMERICA'S INFRASTRUCTURES* (1997).

[230] Robinson, M. Terror fears over hundreds of 'ghost ships' turning off their tracking devices (The Dailymail), 2017.

[231] Roman, R., Alcaraz, C., Lopez, J., and Sklavos, N. Key management systems for sensor networks in the context of the internet of things. *Computers & Electrical Engineering 37*, 2 (2011), 147–159.

[232] Ronen, E., and Shamir, A. Extended functionality attacks on IoT devices: The case of smart lights. In *2016 IEEE European Symposium on Security and Privacy (EuroS&P)* (2016), IEEE, pp. 3–12.

[233] Ronen, E., Shamir, A., Weingarten, A.-O., and O'Flynn, C. IoT goes nuclear: Creating a zigbee chain reaction. In *Security and Privacy (SP), 2017 IEEE Symposium on* (2017), IEEE, pp. 195–212.

[234] Ross, M., Hannes, T., and Jara, A. Baseline security recommendations for iot in the context of critical information infrastructures, 2017, 2019.

[235] Ross, R. S. Guide for conducting risk assessments (NIST SP-800-30rev1). *The National Institute of Standards and Technology (NIST), Gaithersburg* (2012).

[236] Roy, A. Secure aircraft communications addressing and reporting system (ACARS), Jan. 13 2004. US Patent 6,677,888.

[237] Roy, S. Denial of service attack on protocols for smart grid communications. In *Security Solutions and Applied Cryptography in Smart Grid Communications*. IGI Global, 2017, pp. 50–67.

[238] Sahner, R. A., Trivedi, K., and Puliafito, A. *Performance and reliability analysis of computer systems: an example-based approach using the SHARPE software package*. Springer Science & Business Media, 2012.

[239] Santamarta, R. In flight hacking system (IOActive Research Labs), 2016.

[240] Sargolzaei, A., Yen, K., and Abdelghani, M. Delayed inputs attack on load frequency control in smart grid. In *Innovative smart grid technologies conference (ISGT)* (2014), IEEE, pp. 1–5.

[241] Saxena, N., Grijalva, S., Chukwuka, V., and Vasilakos, A. V. Network security and privacy challenges in smart vehicle-to-grid. *IEEE Wireless Communications 24*, 4 (2017), 88–98.

[242] Scheel, R. Smart TV hacking. (Oneconsult talk at EBU Media Cyber Security Seminar), 2017.

[243] Schoenfeld, M. H., Compton, S. J., Mead, R. H., Weiss, D. N., Sherfesee, L., Englund, J., and Mongeon, L. R. Remote monitoring of implantable cardioverter defibrillators. *Pacing and clinical electrophysiology 27*, 6p1 (2004), 757–763.

[244] Scroxton, A. German authorities probe ransomware hospital death, 2020.

[245] SecurityScorecard, D. Listening to patient data security: Healthcare industry and telehealth cybersecurity risks, 2020.

[246] Sequeiros, J. B., Chimuco, F. T., Samaila, M. G., Freire, M. M., and Inácio, P. R. Attack and system modeling applied to iot, cloud, and mobile ecosystems: Embedding security by design. *ACM Computing Surveys (CSUR) 53*, 2 (2020), 1–32.

[247] Shacham, H., Page, M., Pfaff, B., Goh, E.-J., Modadugu, N., and Boneh, D. On the effectiveness of address-space randomization. In *Proceedings of the 11th ACM conference on Computer and communications security* (2004), ACM, pp. 298–307.

[248] Shim, K.-A. A survey of public-key cryptographic primitives in wireless sensor networks. *IEEE Communications Surveys & Tutorials 18*, 1 (2016), 577–601.

[249] Shinde, S. A., Nimkar, P. A., Singh, S. P., Salpe, V. D., and Jadhav, Y. R. Mqtt-message queuing telemetry transport protocol. *International Journal of Research 3*, 3 (2016), 240–244.

[250] Siemens, A. Simatic wincc flexible, 2010.

[251] Skopik, F., and Ma, Z. Attack vectors to metering data in smart grids under security constraints. In *Computer Software and Applications Conference Workshops (COMPSACW)* (2012), IEEE, pp. 134–139.

[252] Sood, A. K., and Enbody, R. J. Targeted cyberattacks: A superset of advanced persistent threats. *IEEE security & privacy 11*, 1 (2013), 54–61.

[253] Spenneberg, R., Brüggemann, M., and Schwartke, H. PLC-blaster: A worm living solely in the PLC. *Black Hat Asia, Marina Bay Sands, Singapore* (2016), 1–16.

[254] Spring, T. Solar power firm patches meters vulnerable to command injection attacks, 2016.

[255] Sridhar, S., and Govindarasu, M. Model-based attack detection and mitigation for automatic generation control. *IEEE Transactions on Smart Grid 5*, 2 (2014), 580–591.

[256] Srivastava, A., Morris, T., Ernster, T., Vellaithurai, C., Pan, S., and Adhikari, U. Modeling cyber-physical vulnerability of the smart grid with incomplete information. *IEEE Transactions on Smart Grid 4*, 1 (2013), 235–244.

[257] Stellios, I., Kotzanikolaou, P., and Grigoriadis, C. Assessing iot enabled cyber-physical attack paths against critical systems. *Computers & Security 107* (2021), 102316.

[258] Stellios, I., Kotzanikolaou, P., Polemi, N., and Douligeris, C. Adversarial models and behaviors. In *Information and Systems on Cyberspace*. New Tech Pub, 2021, pp. 129–151.

[259] Stellios, I., Kotzanikolaou, P., and Psarakis, M. Advanced persistent threats and zero-day exploits in industrial internet of things. In *Security and Privacy Trends in the Industrial Internet of Things*. Springer, 2019, pp. 47–68.

[260] Stellios, I., Kotzanikolaou, P., Psarakis, M., and Alcaraz, C. Risk assessment for iot-enabled cyber-physical systems. In *Advances in Core Computer Science-Based Technologies*. Springer, 2021, pp. 157–173.

[261] Stellios, I., Kotzanikolaou, P., Psarakis, M., Alcaraz, C., and Lopez, J. A survey of iot-enabled cyberattacks: Assessing attack paths to critical infrastructures and services. *IEEE Communications Surveys & Tutorials 20*, 4 (2018), 3453–3495.

[262] Stellios, I., Mokos, K., and Kotzanikolaou, P. Assessing vulnerabilities and iot-enabled attacks on smart lighting systems. In *European Symposium on Research in Computer Security* (2021), Springer, pp. 199–217.

[263] Stellios, I., Mokos, K., and Kotzanikolaou, P. Assessing smart light enabled cyber-physical attack paths on urban infrastructures and services. *Connection Science 34*, 1 (2022), 1401–1429.

[264] Stergiopoulos, G., Kotzanikolaou, P., Theocharidou, M., and Gritzalis, D. Risk mitigation strategies for critical infrastructures based on graph centrality analysis. *International Journal of Critical Infrastructure Protection 10* (2015), 34–44.

[265] Stouffer, K., Falco, J., and Scarfone, K. Guide to industrial control systems (ics) security. *NIST special publication 800*, 82 (2011), 16–16.

[266] Strohmeier, M., Schafer, M., Lenders, V., and Martinovic, I. Realities and challenges of nextgen air traffic management: the case of ADS-B. *IEEE Communications Magazine 52*, 5 (2014), 111–118.

[267] Swales, A., et al. Open modbus/tcp specification. *Schneider Electric 29* (1999), 3–19.

[268] Szary, W., and Auchard, E. Polish airline, hit by cyber attack, says all carriers are at risk (The Reuters), 2015.

[269] Tajer, A. False Data Injection Attacks in Electricity Markets by Limited Adversaries: Stochastic Robustness. *IEEE Transactions on Smart Grid* (2017).

[270] Tanen, J. Breaking bhad: Getting local root on the belkin wemo switch.

[271] Tenaglia, S., and Tanen, J. Breaking BHAD: Abusing Belkin home automation devices. *Black Hat Europe* (2016), 1–46.

[272] Teso, H. Aircraft hacking: Practical aero series. *Hack In The Box* (2013), 1–44.

[273] Theoharidou, M., Kotzanikolaou, P., and Gritzalis, D. A multi-layer criticality assessment methodology based on interdependencies. *Computers & Security 29*, 6 (2010), 643–658.

[274] Thomas, F.-B. Hundreds of wind turbines and solar systems wide open to easy exploits (forbes), 2015.

[275] Torbati, Y., and Saul, J. Iran's top cargo shipping line says sanctions damage mounting (The Reuters), 2012.

[276] Toschi, G. M., Campos, L. B., and Cugnasca, C. E. Home automation networks: A survey. *Computer Standards & Interfaces 50* (2017), 42–54.

[277] TrapX Research, Labs. Anatomy of Attack: MEDJACK.2 – Hospitals Under Siege. TrapX Investigative Report, 2016.

[278] U.S. Food and Drug Administration. Firmware Update to Address Cybersecurity Vulnerabilities Identified in Abbott's (formerly St. Jude Medical's) Implantable Cardiac Pacemakers. FDA Safety Communication, August 2017.

[279] Vallance, C. Car hack uses digital-radio broadcasts to seize control (BBC), 2015.

[280] Wachsmann, C., and Sadeghi, A.-R. Physically unclonable functions (PUFs): Applications, models, and future directions. *Synthesis Lectures on Information Security, Privacy, & Trust 5*, 3 (2014), 1–91.

[281] Wang, J. Zigbee light link and its applicationss. *IEEE Wireless Communications 20*, 4 (2013), 6–7.

[282] Wang, Q., Liu, X., Du, J., and Kong, F. Smart charging for electric vehicles: A survey from the algorithmic perspective. *IEEE Communications Surveys Tutorials 18*, 2 (2016), 1500–1517.

[283] Ward, R., Roberts, C., and Furness, R. Electronic chart display and information systems (ecdis): State-of-the-art in nautical charting. *Marine and Coastal Geographical Information Systems* (2000), 149–161.

[284] Wenpeng, L. Advanced metering infrastructure. *Southern Power System Technology 3*, 2 (2009), 6–10.

[285] Wichers, D. Owasp top-10 2013. *OWASP, February* (2013).

[286] Wikileaks. Vault 7: CIA Hacking Tools Revealed - CIA malware targets iPhone, Android, smart TVs, 2017.

[287] Wilhoit, K. The SCADA that didn't cry wolf. *Trend Micro Inc., White Paper* (2013).

[288] Winter, T., Thubert, P., Brandt, A., Hui, J., and Kelsey, R. Routing protocol for low-power and lossy networks. *Technical report, rfc 6550, 6551, 6552. IETF* (2012).

[289] Withanage, C., Ashok, R., Yuen, C., and Otto, K. A comparison of the popular home automation technologies. In *Innovative Smart Grid Technologies-Asia (ISGT Asia), 2014 IEEE* (2014), IEEE, pp. 600–605.

[290] Xiang, Y., Ding, Z., Zhang, Y., and Wang, L. Power system reliability evaluation considering load redistribution attacks. *IEEE Transactions on Smart Grid 8*, 2 (2017), 889–901.

[291] Xie, S., Yang, J., Xie, K., Liu, Y., and He, Z. Low-sparsity unobservable attacks against smart grid: Attack exposure analysis and a data-driven attack scheme. *IEEE Access 5* (2017), 8183–8193.

[292] Xie, S., Zhong, W., Xie, K., Yu, R., and Zhang, Y. Fair energy scheduling for vehicle-to-grid networks using adaptive dynamic programming. *IEEE Transactions on Neural Networks and Learning Systems 27*, 8 (Aug 2016), 1697–1707.

[293] Yadron, D., and Tynan, D. Tesla driver dies in first fatal crash while using autopilot mode (The Guardian), 2016.

[294] Yan, C., Wenyuan, X., and Liu, J. Can you trust autonomous vehicles: Contactless attacks against sensors of self-driving vehicle. *DEF CON 24* (2016), 1–50.

[295] Yan, J., Tang, Y., Zhu, Y., He, H., and Sun, Y. Smart grid vulnerability under cascade-based sequential line-switching attacks. In *Global Communications Conference (GLOBECOM)* (2015), IEEE, pp. 1–7.

[296] Yaqoob, I., Ahmed, E., ur Rehman, M. H., Ahmed, A. I. A., Al-garadi, M. A., Imran, M., and Guizani, M. The rise of ransomware and emerging security challenges in the internet of things. *Computer Networks 129* (2017), 444–458.

[297] Yi, P., Zhu, T., Zhang, Q., Wu, Y., and Li, J. A denial of service attack in advanced metering infrastructure network. In *Communications (ICC), 2014 IEEE International Conference on* (2014), IEEE, pp. 1029–1034.

[298] Yi, P., Zhu, T., Zhang, Q., Wu, Y., and Pan, L. Puppet attack: a denial of service attack in advanced metering infrastructure network. *Journal of Network and Computer Applications 59* (2016), 325–332.

[299] Yunusov, T. Critical vulnerabilities in 3G/4G modems or how to build big brother, 2015.

[300] Yusuf, S. E., Ge, M., Hong, J. B., Kim, H. K., Kim, P., and Kim, D. S. Security modelling and analysis of dynamic enterprise networks. In *Computer and Information Technology (CIT), 2016 IEEE International Conference on* (2016), IEEE, pp. 249–256.

[301] Zambon, E., Etalle, S., Wieringa, R. J., and Hartel, P. Model-based qualitative risk assessment for availability of it infrastructures. *Software & Systems Modeling 10*, 4 (2011), 553–580.

[302] Zeller, M. Myth or reality: Does the aurora vulnerability pose a risk to my generator? In *Protective Relay Engineers, 2011 64th Annual Conference* (2011), IEEE, pp. 130–136.

[303] Zetter, K. Hackers breached railway network, disrupted service (The Wired), 2012.

[304] Zetter, K. *Countdown to Zero Day: Stuxnet and the launch of the world's first digital weapon.* Broadway books, 2014.

[305] Zetter, K. Hacker can send fatal dose to hospital drug pumps, 2015.

[306] Zhang, H., Hu, Z., Xu, Z., and Song, Y. Evaluation of achievable vehicle-to-grid capacity using aggregate PEV model. *IEEE Transactions on Power Systems 32*, 1 (2017), 784–794.

[307] Zhang, K., Ni, J., Yang, K., Liang, X., Ren, J., and Shen, X. S. Security and privacy in smart city applications: Challenges and solutions. *IEEE Communications Magazine 55*, 1 (2017), 122–129.

[308] Zhang, Y., Gjessing, S., Liu, H., Ning, H., Yang, L. T., and Guizani, M. Securing vehicle-to-grid communications in the smart grid. *IEEE Wireless Communications 20*, 6 (December 2013), 66–73.

[309] Zheng, J. X., Xu, T., and Potkonjak, M. Securing embedded systems and their IPs with digital reconfigurable PUFs. In *Power and Timing Modeling, Optimization and Simulation (PATMOS), 2016 26th International Workshop on* (2016), IEEE, pp. 169–176.

[310] Zimmer, V., Sun, J., Jones, M., and Reinauer, S. *Embedded Firmware Solutions: Development Best Practices for the Internet of Things*. Apress, 2015.