I. Stellios, P. Kotzanikolaou, M. Psarakis, C. Alcaraz and J. Lopez, "Survey of IoT-enabled Cyberattacks: Assessing Attack Paths to Critical Infrastructures and Services", *IEEE Communications Surveys and Tutorials*, vol. 20, pp. 3453-3495, 2018.
http://doi.org/10.1109/COMST.2018.2855563
NICS Lab. Publications: https://www.nics.uma.es/publications

A Survey of IoT-enabled Cyberattacks: Assessing Attack Paths to Critical Infrastructures and Services

[1]Ioannis Stellios [1]Panayiotis Kotzanikolaou, [1]Mihalis Psarakis, [2]Cristina Alcaraz, and [2]Javier Lopez

[1]DEspedard hay un error. Hay un mept. of Informatics, University of Piraeus,

85 Karaoli & Dimitriou, GR-18534, Piraeus, Greece

[2]Computer Science Department,University of Malaga,

Campus de Teatinos s/n, 29071, Malaga, Spain

Abstract

As the deployment of Internet of Things (IoT) is experiencing an exponential growth, it is no surprise that many recent cyber attacks are *IoT-enabled*: The attacker initially exploits some vulnerable IoT technology as a first step towards compromising a critical system that is connected, in some way, with the IoT. For some sectors, like industry, smart grids, transportation and medical services, the significance of such attacks is obvious, since IoT technologies are part of critical back-end systems. However, in sectors where IoT is usually at the end-user side, like smart homes, such attacks can be underestimated, since not all possible attack paths are examined. In this paper we survey IoT-enabled cyber attacks, found in all application domains since 2010. For each sector, we emphasize on the latest, *verified* IoT-enabled attacks, based on known real-world incidents and published proof-of-concept attacks. We methodologically analyze representative attacks that demonstrate direct, indirect and subliminal attack paths against critical targets. Our goal is threefold: (i) To assess IoT-enabled cyber attacks in a risk-like approach, in order to demonstrate their current threat landscape; (ii) To identify hidden and subliminal IoT-enabled attack paths against critical infrastructures and services, and (iii) To examine mitigation strategies for all application domains.

Keywords: Internet of Things, Cyber Attacks, Smart Grids, SCADA, Intelligent Transportation Systems, Smart Medical Systems, Smart Home, Critical Infrastructures.

1 Introduction

In the last two decades we have experienced significant advances in both computing and communication technologies that have generated a plethora of new smart appliances. Internet of Things (IoT) technologies mainly consist of computationally constraint devices that extended the connectivity of systems and users in domain-specific applications. The number of connected "things" is estimated to grow exponentially and is expected to reach up to 50 billion by 2020 [1].

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 [2, 3] that may be managed through Internet-connected control systems. Since the services provided by such large systems are vital for the well-being of the society, they have been recognized as *Critical Infrastructures* (CIs) by national and international bodies [4] and their resilience against cyber attacks has been recognized as a primary concern. Indeed, attacks that could compromise, degrade or lead to loss of their services, would result in severe consequences, in terms of public safety and order, economic or environmental loss.

Apparently, this new interconnected world of devices raises new security challenges. For example, Supervisory Control and Data Acquisition (SCADA) systems, that until recently were isolated from the cyber world, are now becoming a part of it [5]; at the same time, existing security technologies are inadequate to protect these infrastructures in this fast-evolving threat landscape. The annual reports published by the European Agency for Network and Information Security (ENISA) [6] and the Industrial Control Systems Cyber Emergency Response Team (ICS-CERT) [7], clearly underline the current vulnerabilities and exploitations of the heterogeneous communication systems in charge of controlling and supervising critical infrastructures. From the cyber criminals' perception, this is a new opportunity to inflict maximum damage with minimal effort [8]. They can apply many existing techniques to stealthily exploit cyber-physical systems of strategic importance in diverse sectors, such as energy [9, 10], supply chain management [11], smart cities [12] and others.

Attacks that target resource-constrained IoT devices have multiplied over the last years [13]. 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 [14], faulty hardware chips [15] and easy to tamper with devices [16] are making the present situation even more dramatic.

Motivation Although security attacks against cyber-physical systems [17] and IoT devices [13] have received considerable attention during the last years, the significance of IoT-enabled attacks is not always fully assessed. In the case

of IoT devices installed in "back-end", large-scale systems it is clear that attacks against them directly affect critical systems and services. Typical examples are Wireless Sensor Networks (WSNs) that monitor and control electrical pylons, industrial machineries, traffic lights and healthcare systems [1, 18], or smart power meters that interact with generation and distribution systems, in order to preserve power supply according to the real demand [19].

On the other hand, in the case of attacks against "end-user" IoT devices, the potential consequences are usually underestimated. Intuitively this seems to be correct, since such attacks involve smart home appliances, personal devices, vehicles or body area networks which are not directly connected with CIs. However, as recent real-world incidents have shown [12, 20], in many cases the actual target *was not* the IoT system itself, but some external, somehow connected and far more critical system or service. Unfortunately, the potential interconnections of IoT devices may not always be directly identifiable. To make things even worse, the interdependencies between CIs [21] combined with the lack of proper security controls in IoT devices create a wide attack surface; a skillful adversary may abuse, or even extend the capabilities of IoT devices in unpredictable ways [22], with potentially high impact.

To secure this highly connected cyber-physical world we must first examine in depth, the known incidents of critical IoT-enabled attacks. In the case of attacks on back-end IoT devices (directly connected with CIs), we must analyze the IoT vulnerabilities that were exploited to ultimately compromise the CIs, in order to properly secure them. In the case of attacks based on end-user IoT devices (not directly connected with critical systems), we must additionally study the attack vectors in order to learn from the creativity of the attackers, re-assess the underestimated attack paths, and prevent such subliminal threats from being realized in the future.

Contribution In this paper we survey recent *IoT-enabled* cyber attacks, *i.e.* attack vectors that exploit vulnerabilities in IoT technologies (devices, applications and communications), in order to enable and/or amplify an attack against a target system. The actual target of these attacks however, may not be the IoT device itself. We focus on *critical* IoT-enabled attacks, *i.e.* those whose actual goal is to affect some critical system or service, interconnected in some way with the IoT device. We focus on *verified* attacks, *i.e.* either real-world incidents, or attacks that have been implemented and published by researchers. Attacks that are theoretically possible but have not been verified at least by a proof of concept (PoC), are not included in this survey, in order to provide a taxonomy of realistic attack vectors. To the best of our knowledge, this is the first systematic approach to review and assess verified IoT-enabled attacks for all application domains.

Structure The rest of this paper is structured as follows (see Fig. 1 for a pictorial view). In Section 2 we review existing surveys on IoT security. Besides from having a merit on its own, it enables us to clarify the differences with

our work and the novelty of this survey. In Section 3, we describe a threat model for IoT-enabled attacks, that aims to capture all possible attack paths (direct, indirect and subliminal) against critical systems and services. Based on this model, in Section 4 we define a targeted risk-based criticality assessment methodology for IoT-enabled cyber attacks. This methodology is then used in Sections 5, throughout 9, to analyze verified cyber attacks in various IoT sectors. Based on the analysis of these attacks, in Section 10 we summarize security controls that can effectively mitigate such threats, either as short-term or long-term solutions. Finally, Section 11 indicates research and implementation gaps and concludes the paper.

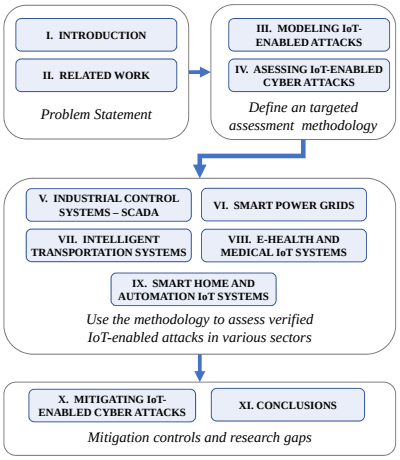

Figure 1: A pictorial view of the paper structure.

2 Related Work: Surveys on IoT security

Several surveys in the literature exist that address many diverse aspects of security in the IoT ecosystem [2, 17, 23–38]. We briefly review these works in order to: (a) Provide a concrete description and categorization of relative efforts and (b) highlight the differences and therefore the actual contribution of this work.

Surveys on IoT security and threat modeling Several works describe different approaches and viewpoints to model IoT security, based on various IoT components and layers. Babar *et al.* [23] proposed one of the first IoT security models, based on a 3-axis generic categorization: Security, privacy and trust requirements. The proposed taxonomy captures threats related with identity management, communications, storage, embedded security and physical threats. However it does not model IoT-enabled attacks against interconnected systems. Roman *et al.* [25] describe the security challenges of highly connected objects.

They identify the need for a holistic view of security that takes into consideration all IoT elements and layers, such as protocol and network security, identity management, data privacy, trust management and fault tolerance. The security challenges defined in [25] were examined later by the same authors [28] in distributed IoT paradigms with various degrees of centralization, collaboration and connectivity. They analyze various internal and external attacker models; since the definition of a 'perimeter' is not easy in IoT, they examine fuzzy internal/external attackers based on a threat categorization that includes Denial-of-Service (DoS), physical threats, eavesdropping, node capture and compromisation. Humayed [17] *et al.* describe a framework to model cyber-physical security. Their approach is based on three perspectives. The security perspective examines threats, vulnerabilities and security controls. The cyber-physical perspective examines physical, cyber and mixed components. Finally the systems perspective examined security threats in various IoT application domains.

Mosenia and Jha [39] provide a comprehensive security study for IoT technologies, with an emphasis on RFID and sensors. Their security model uses a three level approach, based on: (i) edge nodes (*e.g.* RFIDs, sensors), (ii) edge (fog) computing and (ii) communications. Then, for each level, the relative vulnerabilities and countermeasures are presented and mapped to security threats. Yaqoob [40] *et al.* present recent work as well as key requirements and future research challenges concerning the IoT architecture. Parameters such as applications, enabling technologies, architectural requirements, network topologies are described whereas future IoT architectures are proposed through real paradigms. Yaqoob [41] *et al.* discuss the rise of ransomware attacks. After a meticulous categorization of ransomware software they present the state of the art research on IoT security in different domains such as ad hoc and sensor networks, smart home, healthcare, smart cities, end-user, and RFIDs. Then they devise an threat taxonomy that is based on parameters which include threats, requirements, IEEE standards, deployment levels and Technologies. Finally they pinpoint open research challenges including the need for data integrity, lightweight security mechanisms, physical protection, privacy and trust.

Surveys on IoT communications security Other surveys focus on the security aspects of the IoT communication protocols. Granjal *et al.* [31] examine in detail the security of existing IoT protocols for various network layers. They analyze the 802.15.4x security mechanisms at the different network layers, such as the MAC layer. They also compare existing key management techniques for network-layer IoT protocols, such as the Routing Protocol for Low-Power and Lossy Networks (RPL) and for application-layer protocols, like the Constrained Application Protocol (CoAP) [42]. Furthermore, they review end-to-end security technologies for IoT, such as IPSec or VPN with compressed security headers. Other security issues discussed in [31] include 6LoWPAN [43] security enhancements such as protocol modifications to confront fragmentation attacks.Pongle and Chavan [34] examine a number of possible attacks on RPL routing protocol topology, such as: "On-path" attacks (selective forwarding,

alteration), availability attacks (sinkhole, hello flooding, wormhole, blackhole, DoS), impersonation attacks (Sybil, clone ID) and spoofing attacks. Attacks on 6LoWPAN (fragmentation, authentication, and confidentiality attacks) are also discussed. Furthermore, they compare Intrusion Detection Systems (IDS) that are suitable for resource constrained devices running under 6LoWPAN protocol and classify them in event-based, signature-based, host-based and specification-based IDS.

Airehrour *et al.* [35], to the contrary, emphasize on secure routing for IoT and analyze existing routing protocols and mechanisms. The proposed recommendations for secure routing can be categorized in secure route establishment, self-stabilization and malicious node identification, while maintaining lightweight computations. Finally, they summarize various trust models for secure routing used in sensor networks, such as Bayesian, game theory, entropy, fuzzy, probability, neural network, swarm intelligence, directed and undirected graph, arithmetic/weighting and Markov chain, than can be considered for IoT as well. Finally, Sonar and Upadhyay [36] emphasize their survey on Distributed DoS attacks in IoT, such as jamming, "killing", de-synchronizing and flooding attacks.

Domain-specific surveys on IoT security Many works survey security issues of particular areas (domains of use) of IoT technologies such as WSNs. Alcaraz *et al.* [24] examine the security challenges related with the integration of WSNs in the Internet, using two different integration approaches, (protocol) stack-based and topology-based. Granjal *et al.* [32] analyze the challenges of integrating low-power WSNs with the Internet. This survey provides a concrete review of *Machine-to-Machine* technologies. Furthermore, it provides a critical view of the existing security mechanisms for various network layers of WSNs, such as 6LoWPAN and application-layer mechanisms.

In a more specific context, Roman *et al.* [44] examine the applicability of existing key management schemes, either public-key or symmetric key ones, in WSNs. They analyze the suitability of various key management frameworks, like key pool frameworks and mathematical frameworks. Shim [38] provides a thorough review of the applicability of Public Key Cryptography (PKC) in IoT devices in WSNs. The goal of this survey is to investigate under what conditions PKC is viable for resource-constraint sensors. It examines implementations of known asymmetric algorithms such as RSA, ECC, L-PKC MQ-PKC on popular IEEE 802.15.4-compliant micro-controllers, in terms of execution time, energy consumption, communication overhead and resource occupations.

Industrial control and SCADA systems is another domain with particular security challenges. For example, Alcaraz *et al.* [27] provide a thorough analysis of the security requirements for industrial substations based on smart sensor networks and on specific communication protocols, such as the Highway Addressable Remote Transducer (HART), WirelessHART [45] or ISA100.11a [46], all of which are compatible with 6LowPAN [47]. They examine various integration strategies into the Internet to determine the connectivity degree, based on

the operational limitations of the control systems, the criticality of the application context and the constraints of the Internet-enabled control devices. Similarly, Xu, He and Li [2] review the key enabling technologies of IoT in a variety of industry sectors, such as health-care, food supply chain, mining, transportation logistics and firefighting, and analyze the research challenges and future trends. Security and privacy challenges in industrial IoT are also examined by the National Institute of Standards and Technology (NIST) special publication 800-82 [30] whereas Miller and Rowe [26] also provide a review specific to industrial systems. Their analysis focus on the analysis of real cyber-security incidents against SCADA systems. They classify the attacks based on various criteria, such as the source and the target sector of the attack, the method of operations and the impact of the attack. The goal is to provide a taxonomy that will be used in order to compare current and future SCADA incidents. This is one of the very few works that provide a description and classification of real-world security incidents in SCADA and critical infrastructures, although the attack vectors of the examined incidents do not always involve IoT technologies. Sadeghi *et al.* [33] also examine a number of actual attacks against industrial systems. Based on an analysis of various attack surfaces of industrial IoT systems, they define generic security goals and requirements.

The security of smart grids has been studied in various recent surveys, such as [48–51]. For example, Komninos *et al.* [50] survey security challenges and countermeasures in smart grid and smart home and evaluate the security impact of related attacks. Also the three volumes given by the NIST 7628 [52] provide an extensive analysis of security recommendations and good practices for smart grid security. Another IoT domain that has drawn considerable attention from a security point of view during the last years is intelligent transportations. A recent study of the ENISA [53] reports that there is currently no EU policy on cyber security for transport. Miller and Valasek [29] provide an excellent survey of the remote automotive attack surfaces. They review the automotive network architecture, providing one of the first publicly available detailed resources for the automobile network topology. They describe the remote attack surfaces for various car models and discuss possible mitigation controls.

Comparison with this work Some of the surveys discussed above present various interesting security perspectives, models and categorizations for IoT security [23,25,28,39,41], or for related areas like cyber-physical systems security [17]. However, none of them provides a taxonomy based on an assessment of real-world attacks. In addition, the above models mainly aim to identify attacks against the IoT systems themselves, and not indirect IoT-enabled attack paths against other systems. Other works [31,34,35] emphasize on the security analysis of IoT-specific network protocols, while others examine specific types of attacks, like Distributed Denial-of-Service (DDoS) attacks [36].

The strategy followed in our work is conceptually closer to [26], [33] and [29]. Our analysis will also be based on the study of verified attacks only (real or PoC). However, our scope will be much broader than industrial IoT (as in [26, 33],

smart vehicles (as in [29]), smart grid (as in [50]), or any other sector-specific analysis. In addition, we will define a risk-based model in order to assess the criticality of the examined attacks, taking into consideration all possible attack paths, including *non-obvious* (indirect or hidden) attack vectors that may use IoT devices not as the target of the attack, but mainly as the enabler or the amplifier of an attack against interconnected critical systems.

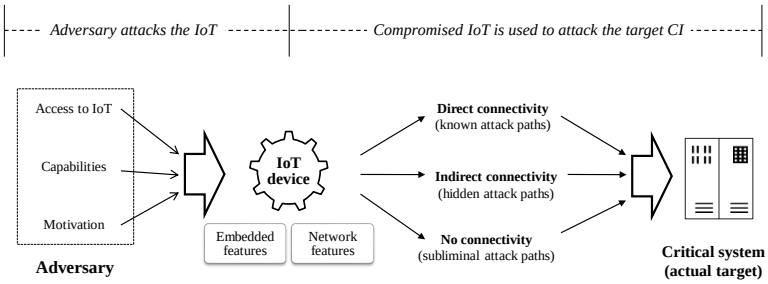

Figure 2: A high-level description of IoT-enabled critical attack vectors. A motivated adversary will use any potential access to the IoT device and his/her skills in order to compromise the device. Then, by exploiting all the connectivity paths of the IoT device with other systems, he/she will eventually attack the critical system. The connectivity of the IoT device with the critical system may not be obvious.

3 Modeling IoT-enabled Cyber Attacks

IoT technologies favor the interoperability and remote management of various cyber-physical systems, including CIs [54], but 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 [55], are unfortunately sufficient conditions that enable various *attack vectors* against critical systems and services. An attack vector describes the steps that an attacker will undertake in order to realize a threat [6]. In order to model IoT-enabled attack vectors against critical systems and services, we will 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 2 describes this model.

The adversary. It represents the actor of the attack. It is the entity whose actual goal is to cause damage to a target system. If an attack can be realized by "powerful" adversaries only, then such an attack is usually less possible to happen and vise versa. We model the power of potential adversaries based on their access level, capabilities and motivation (Section 3.1).

The IoT device. In our attack 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. A highly vulnerable IoT device usually means a higher exposure (attack surface) for all systems connected with it. Section 3.2 describes how IoT vulnerabilities are modeled.

The actual target. Usually, critical systems of high importance are the actual targets of attacks. An adversary with sufficient capabilities and motivation will attempt to abuse existing paths between the vulnerable IoT and a critical system. 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; IoT devices directly connected with a critical infrastructure create obvious attack paths that are very attractive for potential adversaries and for that reason their security should be a top priority. 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 [29,56]. Passive medical IoT devices such as smart clinical beds may be indirectly connected to in-hospital critical systems [57]. 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, [22] describes how vulnerable smart lamps may be used to exfiltrate sensitive data from systems that reside in highly secured premises. Even worse, 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. In Section 3.3 we describe these connectivity paths in detail.

3.1 Characteristics of the adversary

As shown in Figure 2, the adversaries of IoT-enabled attacks can be modeled using three main characteristics: Their access to the IoT device, their capabilities and their motivation.

3.1.1 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.

a) 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.

b) 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.

3.1.2 Required capabilities

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

a) 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.

b) 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.

3.1.3 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.

3.2 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.

3.2.1 Embedded system vulnerabilities

This category involves design and implementation flaws at the IoT hardware (HW) and the software (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 [58].
- *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) [15].

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), *Operating System* (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 [15] to recover the stored credentials.
- *Operating system.* 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 Application Programming Interface (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.

3.2.2 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 [16].

- *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 [22].

- *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 [59]. 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 [60].

- *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 [15]. 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.

3.3 Connectivity between the IoT device and the actual target

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.

3.3.1 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 [61, 62].

Direct logical connection A logical connection may refer to IoT devices that are either inside or outside the CI premises (*e.g.* temperature sensor).

3.3.2 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.

3.3.3 No connection 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 [12, 20].

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 [63]. The attacker may then cause significant financial losses to the end users and reputation loss to the device manufacturers.

4 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 3 and the related criteria, as described in this section.

4.1 Risk-based approach: A high level description

Although various security standards [64–67] 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 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 [65] and the NIST SP800-30 [66]), we define a a risk-based methodology to assess the criticality of IoT-enabled cyber attacks, based on these risk factors (see Fig.3). In order to methodologically assess the risk factors, we will

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

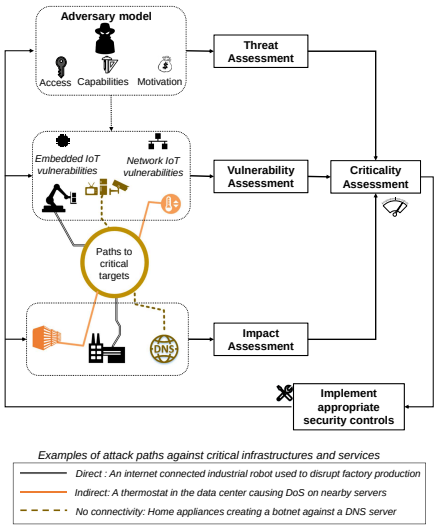

Figure 3: A high-level view of the methodology. The attack model of Section 3 is applied to assess threats, IoT vulnerabilities and impact for potential attack paths and eventually the criticality of IoT-enabled cyber attacks. Security controls may then be implemented (see section 10) to reduce the risk factors.

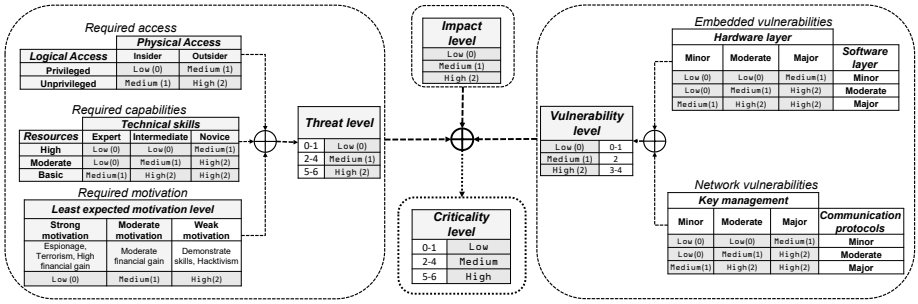

Figure 4: The methodology utilizes the characteristics of the attack model of Section 3 to output a qualitative criticality level of IoT-enabled cyber attacks.

utilize the criteria defined in the attack model defined above. In particular, the adversarial model (Section 3.1) will be used to assess threat level of an attack, while the IoT vulnerability criteria (Section 3.2) 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 3.3.

4.2 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 suggest [65], 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 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.

4.3 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 1. The use of a three level scale is deliberately chosen for simplicity and is compliant with risk assessment standards like [65, 66, 68]. Although more fine-grained scales can be defined (*e.g.* for multi-layer analysis [69]), 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 [50]. Figure 4 demonstrates the evaluation of the criticality level of IoT-enabled cyber attacks, based on our risk-based methodology, which is further described bellow.

4.4 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 3.1 is used for this purpose, since the probability of realizing an attack depends on

Table 1: Summary of the risk factors and their corresponding scales

Value	**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

the existence of capable and motivated adversaries, with sufficient access [66]. Figure 4 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.

4.5 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 3.2 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 4). 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.

4.6 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 3.3. Again, the impact scale defined in Table 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.

4.7 Criticality assessment

The final step is to combine all the partial risk factors as defined above (Sections 4.4 to 4.6) to output the overall criticality level of an examined IoT-enabled cyber attack (see Figure 4). The three level criticality scale defined in Table 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.

4.8 Revisiting IoT-enabled Cyber Attacks

Based on the assessment methodology described, we will analyze and assess IoT-enabled cyber attacks in various sectors. We emphasize on attacks with the following characteristics:

- *Verified* attacks. As explained above, we examine either real-world incidents or attacks that have been actually implemented in controlled envi-

ronments by researchers.

- *IoT-enabled* attacks. We examine attack vectors that involve compromising/abusing vulnerable IoT devices, as part of an extended attack against a target system that is not necessarily the IoT system itself, but usually another important system that is interconnected in some way.

- *Critical* attacks, *i.e.* attacks that may result in potentially high consequences.

For each examined attack, we describe a scenario[2] that describes the environment of the attack, the adversary and the actual target. 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.* [15,22,56,62,70]), as well as on sector-specific technical reports of major security companies [57, 71, 72]. Then, the attack is assessed based on the attack scenario, using the risk factors, *i.e.* threat, vulnerability and impact levels and the risk-based methodology.

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. As described in Section 3, the IoT is not always the actual target; for each attack scenario 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.

Characteristic examples of verified attacks in various IoT application domains are analyzed in this survey: Industrial control systems, smart grids, intelligent transportation systems, medical systems as well as smart home devices. However this categorization is only for practical purposes, since 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). For clarity and in order to assist the reader, we divide the analysis in multiple sub-domains through sections 5 to 9 so as to provide a brief description of the underlying systems and architectures. Then we taxonomize the IoT-enabled attacks that were found in each sub-domain. Detailed analysis of the attacks is provided in the corresponding tables (see Tables 2 to 8), where we describe the implementation of each attack (attack vector and attack scenario), and we assess ther criticality level, based on the examined scenarios.

[2]For some attacks we may describe more than one attack scenario

5 Industrial Control Systems - SCADA

Industrial Control Systems (ICS) are mission critical applications with a high-availability requirement, fully or partially automated, that gather information from a variety of endpoint devices about the current status of a production process. SCADA systems are ICS which are used to monitor and control distribution systems spanning a large geographic area such as gas pipeline, electric power transmission and water distribution systems. SCADA systems are also used to control single sited facilities such as refinery and heavy machinery manufacturing [73]. Given the fact the SCADA is the most well-established ICS technology, this section focuses on IoT-enabled attacks on industrial SCADA systems. In the following sections we also present IoT-enabled attacks that refer to SCADA systems used in other sectors and infrastructures such as smart grids and transportation systems.

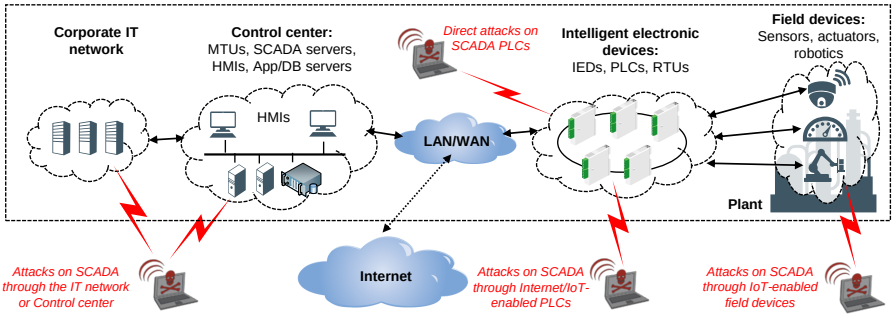

Figure 5: A typical architecture of industrial SCADA systems and relative IoT-enabled attacks. Internet connectivity and the interconnection of IoT-enabled PLCs and field devices extend the attack surface. Some attacks are only possible by adversaries with physical proximity with the target (*e.g.* direct attacks on PLCs). Others may also be initiated by remote adversaries (*e.g.* attacks on IoT-enabled PLCs or field devices).

Cyber security was not among the major concerns of the early SCADA systems, which used to operate in protected and isolated environments. But since the SCADA systems became connected to corporate IT networks and to the Internet, they have become exposed to a wider attack surface. Cyber attacks associated with the Internet connectivity of SCADA systems have been extensively studied in the past [26, 74–77]. Most reported attacks exploit the connectivity of the SCADA components with the corporate IT servers or the SCADA servers, in order to pivot to the field devices. As the IoT technology increasingly spreads in the industry sector, more SCADA components become interconnected using IP-based protocols, and thus extending the attack surface.

Obviously, industrial IoT devices are in most cases directly connected with critical control systems. Therefore, by compromising an industrial IoT device,

an adversary can usually establish a direct attack vector to compromise a critical system. Thus, such attacks paths are more or less known and easy to identify. From this point of view, although IoT-enabled attacks on industrial SCADA are the most obvious example of attacks against critical infrastructures, they are not on the focal point of the current survey because they do not demonstrate hidden or underestimated paths. Thus, in this section we do not present a detailed list of cyber attacks in SCADA systems but some representative examples for survey completeness. Sections 7 - 9 that examine transportation, medical devices and home appliances respectively, present more illustrative examples that activate such invisible attack paths.

5.1 SCADA architecture

Figure 5 depicts a typical SCADA architecture. It consists of one or more distributed supervisory computers, also called Command and Control (C&C) centers and a number of Intelligent Electronics Devices (IEDs), such as PLCs and Remote Terminal Units (RTUs) connected in an hierarchical model. Intelligent electronics 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 and send commands to the field devices. Operators monitor and control the system through Human Machine Interface (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 [78].

Given that the geographical area of a SCADA system may significantly vary, from the premises of a small factory up to a large city area when the SCADA controls the power grid of a smart city, the SCADA systems may use Local (LAN) or Wide Area Networks (WAN). The communication infrastructure maybe frame relay network, 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/TCP, Distributed Network Protocol 3 (DNP3), IEC-104, DeviceNET, ControlNET, and many more. This network diversity raises more security issues.

5.2 IoT-enabled attacks on SCADA systems

Securing SCADA systems is a daunting challenge when compared to classic IT infrastructure. Several vulnerability issues have been reported for SCADA systems [79, 80]: The lack of feedback from system operators, the extensive lifetime of SCADA products, the fact that availability is considered more important than confidentiality or integrity, the use of unsecured protocols (*e.g.* Hyper Text Transfer Protocol - HTTP/File Transfer Protocol - FTP), the lack of authentication and message integrity mechanisms in the existing network protocols,

the embedded vulnerabilities of SCADA networks such as Modbus, DNP3 and ICCP [81]. Several attacks on SCADA systems have been reported the last years that exploit the aforementioned vulnerabilities.

In this survey, we classify IoT-enabled attacks on SCADA systems according to the target attack surface. The ultimate goal of most attacks is to affect the SCADA field devices. 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. The first category cannot be considered as IoT-enabled attack, even though it takes advantage of the connectivity of the SCADA components to infiltrate into the system. However, it is presented here for survey completeness. Next, we describe shortly some representative examples of these categories, while in Table 2 we present realistic attack scenarios and assess them using the risk-based methodology described in Section 4. Finally, we present a more complex case, where different attack scenarios are combined to compromise industrial robots.

5.2.1 Attacks through the IT network or the control center

The Stuxnet worm, reported on June 2010, caused perhaps the most famous cyber-physical attack against critical industrial SCADA systems [82, 83]. 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 an instance of what is now known as Advanced Persistent Threats (APTs) [84].

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 [85]. 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 SQL injection, CSRF (cross-site request forgery) and dictionary attacks [86]. Since Modbus protocol does not require authentication [87], 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 [88]. One of the most interesting attacks, against a Japanese honeypot, is analyzed in Table 2.

5.2.2 Attacks through IoT-enabled PLCs.

In [62] 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 [92] 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 [61] 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 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.

5.2.3 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

Table 2: 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** ×	×
[82, 83] 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**
[85] 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**
[62] 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**
[61] 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**
[89, 90] 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**
[91] 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**

the Internet, in order to enable remote control services. According to a technical report published by the security company Rapid7 [89, 90], 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 [93]: 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 [94], against a virtual ATG located in Washington DC.

5.2.4 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 [72, 91] 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 FTP services or through industrial routers. From a total number of 83673 robots discovered, 5105 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 2 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: (a) Control-loop parameters alteration, (b) user-perceived robot state alteration, (c) actual robot state alteration, (d) calibration parameters tampering and (e) 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.

6 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 [95]. On the other hand, the distributed intelligence and broadband capabilities of smart grids increase the cyber-security risks.

Although the smart grids could be considered as special, large-scale SCADA systems, we further analyze them separately, due to their importance as CIs. As in the previous domain, we do not aim to analyze in depth the security risks of smart grids; too many surveys are available in the literature (*e.g.* [48–51]). 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 (FDIAs). 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.

6.1 Smart grid architecture

A smart grid is divided in three main domains: Generation, transmission and distribution of electricity as shown in Fig.6. 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 deployed in a wide area. Energy Management 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 [96].

From the cyber-security viewpoint, the key components of a smart power grid are: (a) The SCADA systems and (b) the Advanced Metering Infrastructure (AMI) [97, 98]. SCADA systems monitor and control at real-time the power delivery systems based on several communication networks. AMI measures,

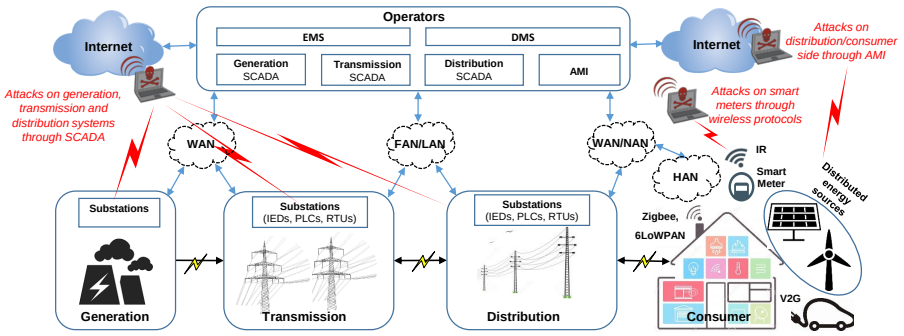

Figure 6: A typical smart power grid architecture. Wide-area, heterogeneous SCADA systems, AMI infrastructure and home appliances extend the attack surface. For the cases where the adversary is located on the cloud, the attack is from the Internet (*e.g.* attack on the transmission system through the IT network of the operators). Otherwise, the attack requires physical proximity with the target (*e.g.* attack on smart meters through infrared port).

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 Wide Area Network (WAN) is used to connect distributed NANs [99].

Another smart grid feature is the Vehicle-to-Grid (V2G) network [100]. It is based on the concept that the batteries of electric vehicles can be utilized to assist the stabilization of the electricity network [101–104]. 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 [105].

6.2 IoT-enabled attacks on smart grids

The cyber security of smart power grids have been extensively studied in the past. Several surveys [48–51] present and categorize cyber-physical attacks on different assets of the smart grid and propose effective security countermeasures. Recent research papers [106–111] describe potential PoC attacks on smart grids. Cyber criminals, terrorists and nation state adversaries [10] may attempt to disrupt smart grid services. The consequences of successful attacks may include large-scale blackouts, human safety threats, significant economic loss for companies and less severe such as small-scale outages or consumer equipment damage.

Many smart grid vulnerabilities are associated with networking and communications. Due to the heterogeneity, the diversity and the ever increasing

complexity of the grid networks [112], new security and privacy issues arise. Moreover, the integration of low-cost, low-power wireless protocols, *e.g.* ZigBee or 6LoWPAN, into the smart grid components, especially in the distribution system (AMI), introduces new vulnerabilities. Even though more secure protocols are employed, e.g. WiFi, in a HAN or a NAN network, a strongly motivated attacker can crack the supported encryption scheme and perform *Man-in-the-Middle* (MiTM) attacks [33, 113].

Since smart grids are mainly controlled by SCADA systems, they are prone to similar security problems. Alike SCADA, indirect connectivity paths, that exist between the smart grid components and corporate IT networks such as grid operators and service providers, extend the attack surface of smart grid SCADA systems. For example, since utility companies must be constantly aware of real-time data concerning energy usage to make smart trades, financial-driven attackers can take advantage existing or zero day vulnerabilities to exploit the economics of the energy industry [111, 114–117]. An on-growing plethora of smart objects such as home appliances and electric vehicles, interact directly with smart meters and through them with the distribution and transmission domain as well. Smart meters collect a large amount of data and transport it, in many cases through wireless communications, to the utility company and service providers. In addition to that, grids must integrate and regulate a large number of distributed small-scale renewable energy sources, such as wind turbines, solar panels or biomass power plants, as part of the generation domain. Since all these cyber-physical systems heavily rely on IoT interconnectivity in order to interact and self regulate one another, cascading failures are imminent. Just by exploiting vulnerabilities of smart meters, an adversary can obtain private consumer information [118], monitor consumer's activities, permanently disable smart electrical appliances or use them as an amplifier (*e.g.* for DDoS attacks [119, 120]). In [51] attacks on electricity market systems are also surveyed.

In this survey, we classify the attacks on smart grids according to the target domain: (i) Attacks on generation systems [121–123], (ii) attacks on transmission systems including interdiction, substation, load redistribution [107–109,124, 125], (iii) attacks on distribution/customer side systems - AMI, like energy theft, information/privacy leakages/DoS [119, 120, 126, 127]. We separately examine False Data Injection Attacks (FDIA), since they affect all domains [111,128,129]. Table 2 reports realistic scenarios and assesses the criticality of these attacks.

6.2.1 Attacks on generation systems

One of the first security testing experiments on electric power generators is the Aurora attack, demonstrated in 2007 at the Idaho US National Labs [121, 130]. An Aurora 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 [130]. 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 attack scenario described

in [121] and presented in Table 3, that exploits both cyber and physical system vulnerabilities to control circuit breakers, is an ample example of an Aurora-like attack.

In [131] 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 [9, 10, 132, 133], 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.

6.2.2 Attacks on transmission systems

Among the most known attacks in smart grids are those in the Ukrainian energy industry [9, 10, 132, 133], analyzed in Table 3. In December 2015, a region in Ukraine suffered a massive power outage affecting almost 230,000 customers [9]. 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 [10]. 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* [134] / *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 ESSET [135] 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.

6.2.3 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 [128] examine the potential impact of FDIA in three main categories: (i) *Electricity market*: They are mainly focused on the economic aspect of FDIA [111]. 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 [129] 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.

6.2.4 Attacks on renewable energy & distribution/customer side systems (AMI)

Real as well as PoC attacks depict the threat landscape on AMI (*e.g.* in the smart meters [126, 127, 139, 140]). 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 [136] 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) [137] 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) [110] 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

Table 3: 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** ×	× ×
[121] 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**
[9] 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**
[10] 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**
[136] 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**
[137] 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 hardcoded 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**
[110] 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**
[138] 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**
[111] 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**

identified numerous flaws in clean energy systems [138] 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 [138] presented in Table 2 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 [141], [142] indicate security concerns and challenges related with V2G power and communications interactions. In [141], the authors have proposed a model that jointly optimizes security risks and equipment availability in the interdependent power and electric vehicle infrastructure. In [142], 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.

7 Intelligent Transportation Systems

Intelligent transportation systems (ITS) [148] involve smart cars and road infrastructures, railway control systems, air traffic control systems, and smart maritime surface vessels (see Fig.7). 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 [53] 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.

7.1 ITS architecture and related IoT technologies

We briefly describe the main IoT technologies that are utilized in the ITS ecosystem, as depicted in Figure 7.

7.1.1 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 [149]. Initially, a dedicated point-to-point connection was used to connect all subsystems to ECUs.

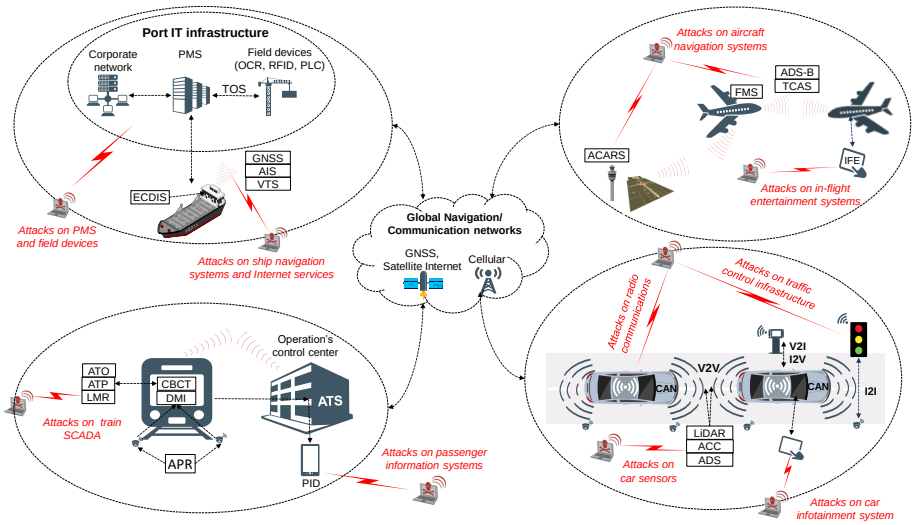

Figure 7: Intelligent transportation systems architectures and relevant IoT-enabled attacks. If the adversary is placed inside the relevant infrastructure perimeter, then the attack requires physical proximity with the target (*e.g.* attacks on car sensors [143,144]). If the adversary is outside the perimeter then the attacks are executed from the Internet (*e.g.* attacks on car infotainment systems [56,145]). 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 [146,147]).

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 [150]. Low-cost off-the-shelf software, such as CANdo [151] 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 [152], a large fraction of the car fleet worldwide relies on CAN.

Smart cars integrate various IoT technologies. Internet connectivity is implemented via cellular data 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 [153]. 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 [154]. 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) [155]. 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.

7.1.2 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 [156], 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 [157] 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.

7.1.3 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) [158] 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) [159] and the Traffic Collision Avoidance System (TCAS) [160].

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.

7.1.4 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) [161, 162]. 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.

7.2 IoT-enabled attacks on ITS systems

We analyze IoT-enabled attacks for all the transportation subsectors.

7.2.1 Attacks on smart cars and traffic control infrastructures

Preliminary works [165] 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 [29, 59] 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, DAB and WiFi); (ii) attacks that exploit vulnerabilities of car infotainment systems; and (iii) attacks based on manipulating sensor IoT technologies. In Table 4 we describe in detail the most characteristic attack vectors and we assess them based on realistic scenarios.

Attacks based on radio communications In [147] 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

Table 4: ATTACKS ON SMART CARS AND ROAD TRAFFIC CONTROL INFRASTRUCTURES

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** ×	× ×
[147] 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**
[145] 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**
[146] Exploit the WiFi connection between a car and its mobile control app (2016, PoC)	1. Crack the Wi-Fi pre-shared key 2. Sniff messages sent by the mobile app through the WiFi 2. 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**
[56] 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	* 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 architecture	(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**
[143] 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**
[144] 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**
[163] 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**
[164] Remote attacks 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**

commands to the CAN bus. In [145] 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 Digital Audio Broadcast (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 [166]. In [146] 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) – and in that sense we characterize the attacker as an "insider" in our assessment – 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 [145] already described above, also belongs to this category). In [56] 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 Wi-Fi 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 [167] the car manufacturer was forced to recall 1.4 million vehicles [168] 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 [169]; 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 [143,144]. In [143] 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 [144]. 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 [170]. 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 [163,164]. These attacks are mainly due to vulnerabilities in the radio communications of traffic control systems. In [163] 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 [164] 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.

Apart from smart cars, IoT-enabled attacks can be found in all other transportation sub-sectors. In the following paragraphs we summarize these attacks, while Table 5 presents an analysis of the related attack vectors.

7.2.2 Attacks on railway control systems

Real incidents against train control systems, such as [180–183], 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

Table 5: Attacks on other Intelligent Transportation Systems (aircrafts, trains and maritime vessels)

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** ×	× ×
[171] 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**
[172] 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**
[173–175] Demo attacks against plane ADS-B systems (2012-15, PoC)	(Description based on [174]) 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**
[176] 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**
[177] 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**
[178] 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**
[179] 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**

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 [171] (see also [184]) a research team named SCADA Strangelove, presented the results of their 3-year assessment on actual SCADA train control systems, utilized be many train operators. They found a number of high-level security and safety issues. First, some digital train switches need constantly Internet access to operate. In addition to that, computer-based interlocking systems where installed in places with poor physical security 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 discovered 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 [185]. 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 [172] 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 [172] 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 in railway transportation systems

when they operate without proper network segmentation. A security analysis [171] concerning devices used in railway communication systems from various vendors revealed hardcoded private SSL keys embedded in their firmware. Other attack scenarios described in [172] 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.

7.2.3 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 [186] and causing DoS to systems used to issue flight plans [187]. 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 phenomenically less important 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 [173–175] 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 [173]. In a similar work [174] 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 [175] 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 (IFE) systems. In [176] 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 Pana-

sonic Avionics IFE system and was based in real data collected by the researcher while he was in flight. Using an exposed 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.

7.2.4 Attacks on maritime surface vessels

Published incidents against maritime cyber systems that are not IoT-specific can be found in [188,189]. Again, we will categorize IoT-enabled attacks in this sub-sector.

Attacks on maritime electronic navigation systems and Internet services In [177] attacks against the AIS of existing vessels were presented. In particular by using 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 [190] 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 *PenTest-Partners* [191] 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/TLS security 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) [178], 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 exe-

cute arbitrary commands with system privileges on the remote system by abusing the Task Manager of the mail client.

Attacks on IoT-enabled Port Management Systems (PMS) and field devices The number of containers shipped world wide have increased over 200% from 1996 to 2014. In a recent study [179] 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 (OTS, OCR, RFIDs), PLCs and motors that are found mainly installed in yard cranes (ICSA-16-348-05B). In a real attack incident [11], 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.

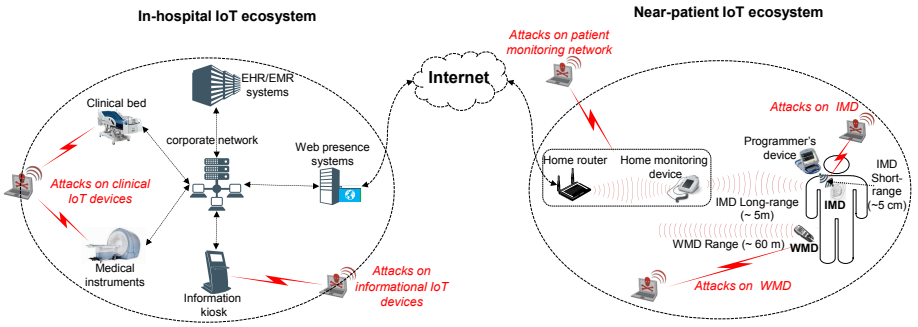

Figure 8: A general architecture of *near-patient* and *in-hospital* IoT devices and relevant IoT-enabled attacks. As in the previous sectors, the placement of the adversary inside the perimeter implies nearby attacks, whereas if the adversary is placed outside the perimeter then the attack can be triggered remotely. Attacks where the adversary is placed at the border imply that this category involves both nearby and remote attack scenarios.

8 E-health and medical IoT systems

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. First we provide a brief description of the medical IoT technologies and then we will review IoT-enabled attacks in the medical sector. Figure 8 describes a general architecture of the medical IoT ecosystem as well as a high-level description of the relevant IoT-enabled attacks. A description of the attack vectors and an assessment of these attacks is provided in Table 6.

8.1 Architecture of medical IoT systems

Medical IoT devices can be categorized to active and passive. Active Medical Devices (AMD) directly interact 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) [192]. These are basically near-patient technologies, although they can also be used during in-hospital treatment. Other AMD technologies can only reside inside hospitals, such as radiation oncology systems. Passive Medical Devices (PMDs) monitor, gather and report data related with the patient's physical condition to medical IT systems. Again those devices may reside inside the hospital (*e.g.* a smart clinical bed) or near the patient (*e.g.* a home monitoring device).

8.1.1 Near-patient medical IoT

IMDs and WMDs are the most common near-patient active IoT technologies [193]. 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 treating patients with automatic therapies. Instances of wireless re-programmable IMDs are smart pacemakers, neurostimulators, and implantable drug pumps [194]. Likewise, latest versions of Implantable Cardioverter Defibrillators (ICDs) support wireless communications for both device re-programming, through an external device operated by a physician and remote patient monitoring [195]. 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 [196]. A wireless interface that utilizes proprietary network protocols (*e.g.* 916.50 MHz with On-Off-Keying modulation) is used to configure device settings [70].

IMDs communicate by utilizing two wireless communication channels. Short-range channels (up to 5cm) are used to program the IMD through the physician's programming device, while "long"-range ones (up to 5m) are used to communicate with a home monitoring device [197]. The WMDs utilize a single wireless communication channel having a broader range, up to 60m, based on the findings of recent attacks [70]. The locally collected data may then be transmitted through an IP patient's monitoring network, to be stored and processed by back-end hospital IT systems.

8.1.2 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 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 A/V software. Furthermore, in most cases, such devices come with rich networking capabilities while running on outdated and 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 [198] 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. 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.

8.2 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/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/systems, while in Table 6 we present an assessment of these attacks based on realistic scenarios.

8.2.1 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 [199]. 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 [200].

Attacks based on IMD/WMD devices The security of wearable and implantable medical devices has been studied in various works in the past [201–206]. 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 [70, 197, 207].

Halperin *et al.* [207] presented a security analysis of such devices on communication protocols, physical tampering and reverse-engineering techniques to 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 (USRP) and open-source radio libraries were used in order to eavesdrop and examine the (unencrypted) low-range communications between the ICD and its programming device. Finally, an attack scenario was demonstrated in which a nearby attacker could intercept patient data and inject bogus messages to modify the preconfigured therapy.

A similar security assessment was presented by Marin *et al.* [197] in a black-box analysis on an ICD. They demonstrated that attacks, which have been presented by security researchers in the past [207], were still possible. Through reverse engineering techniques on the proprietary network protocols, they managed to perform passive/active eavesdropping, spoofing and replaying attacks as well as to exploit the functionality of 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 [70] 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, from WhiteScope security company, performed an exhaustive security evaluation [199] 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 (JTAG, UART, USB or serial), from which, extraction of the firmware and privileged access to the device were possible. Furthermore, there was a lack of well established anti-reverse engineering techniques, such as firmware packing, code obfuscation and data encryption. Moreover, no authentication or control for digitally signed firmware mechanisms were found during the updating process of an OTA update. In addition, several bad practices were discovered that can potentially help an attacker to compromise the device: Use of ASCII text for function names and release versions, clear-text hardcoded credentials on home monitoring devices, hardcoded infrastructure data (*e.g.* phone numbers and IP addresses of the authentication servers), unencrypted sensitive patient data (patient names, physicians, phone numbers, social security numbers and treatment data) on the programmer's hard drive as well as extended use of third-party outdated SW libraries. Notably, over 3,700 known vulnerabilities were discovered in the embedded software of the physician programming devices under evaluation.

Although the study of [199] does not describe any actual PoC attack, it lists numerous vulnerabilities that have been verified on real devices. By exploiting these vulnerabilities, an adversary may compromise any vulnerable patient network device remotely as described in the corresponding attack vector of Table 6.

8.2.2 Attacks on in-hospital IoT devices

Real cyber attacks against hospitals, such as [198, 209], have increased by 63% during 2016 [210]. Here, we focus on those attacks that rely on IoT technologies within hospital facilities. These attacks are based on vulnerabilities of either in-hospital medical IoT devices (both passive and active) [57, 208], or other non-medical IoT devices that may reside within hospital premises [57]. Usually, the adversary uses such vulnerable IoT devices as a point-of-entry, in order to pivot and attack other critical EHR/EMR 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 [208] based on in-depth security assessments, revealed real attacks that took place in three hospitals. The assessors installed within

Table 6: 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** ×	× ×
[197,207] 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**
[70] 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**
[199] Security evaluation of patient home network devices (2017, PoC)	(Attack based on findings of [199]) 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**
[208] 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**
[57] 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**
[57] 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**

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 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) [211] 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 was a computer worm [212] and the hospital's IT stuff had no knowledge of its existence. 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 [57] 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 another PoC attack scenario [57], 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 [57] 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 [213]. 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.

9 Smart Home and automation IoT systems

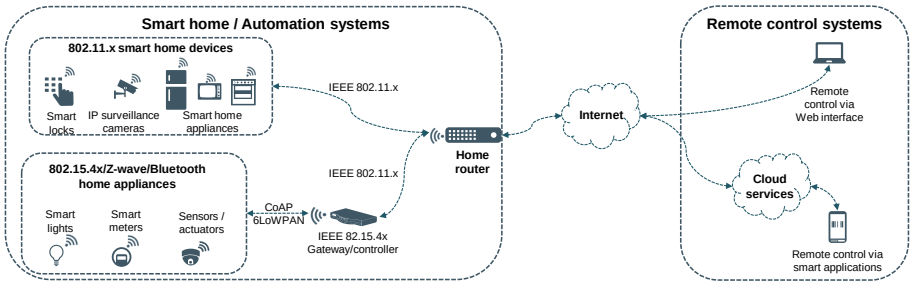

Figure 9: Architecture of *home/automation* IoT ecosystem.

Home automation IoT 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 [214]. 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), entertainment systems (smart TVs and set-top boxes) and smart office devices like printers. 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 many smart home systems are able to interact directly/indirectly with critical infrastructures' components, *e.g.* in the case of smart meters [126, 127, 139, 140].

9.1 Smart home/automation IoT architecture

Home IoT devices use various protocols to communicate with each other and/or with the Internet, as briefly described in Figure 9. With the absence of a single standard protocol and architecture, many different wired (*e.g.* Ethernet, Powerline), and more usually wireless (*e.g.* WiFi, Z-Wave, ZigBee and Bluetooth) technologies are used [215, 216].

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 [217] 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 [42]. ZigBee is a popular low-power wireless mesh networking standard built on top of IEEE 802.15.4. 6LoWPAN [43] is an adaptation layer protocol allowing to transport IPv6 packets over 802.15.4 links, whereas CoAP [42] 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 [218] 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 [219].

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.

9.2 Attacks on smart home/automation IoT systems

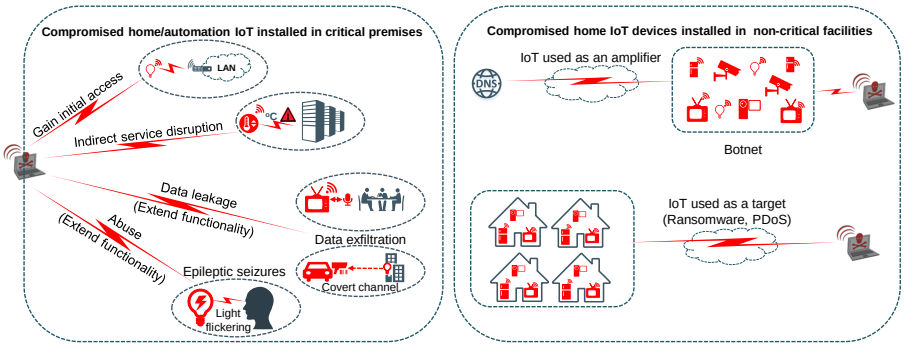

Figure 10: Attacks based on smart home devices. Such attacks may be triggered either by devices that are physically installed near critical systems or by devices installed in non-critical facilities. In the second case a large number of vulnerable IoT devices are used to amplify the consequences of the attack

Based on an analysis of 50 actual home IoT devices, Symantec reported in 2015 [220] a list of common vulnerabilities found in smart home appliances. These included 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 [221–225] have discovered security flaws in various wireless protocols used in home IoT devices such as WiFi, ZigBee and Z-Wave. For example O' Flynn *et al.* [221] presented pulse denial DoS attacks (*i.e.* block the entire RF 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 [226] 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 their latest 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 [71] 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 10 provides an overview of possible attacks based on smart home devices installed in both critical and non critical facilities.

9.2.1 Attacks based on devices installed in critical premises

Real and PoC attacks based on home/automation IoT devices installed in critical environments can be classified into the following categories as shown in Figure 10: (i) Gain initial access, (ii) indirect disruption/denial of critical services, (iii) data leakage, and (iv) system misuse/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 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 [16] 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 [227], a security expert managed to control various systems of a hotel, by connecting his tablet to an exposed Ethernet socket in his

Table 7: ANALYSIS OF IoT-ENABLED ATTACKS ON SMART HOME/BUILDING AUTOMATION SYSTEMS INSTALLED IN CRITICAL PREMISES

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** ×	× ×
[16] 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**
[227] 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**
[228] 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**
[229] 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**
[22] 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**

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.* [228] 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* [229]. 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 [22] 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 [15], 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 [22], 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.

9.2.2 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 10: (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 8 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 [234]. 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 [12] [20]. 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* [235] 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 [236] that

Table 8: ANALYIS OF IoT-ENABLED ATTACKS BASED ON DEVICES INSTALLED IN NON-CRITICAL FACILITES (SMART HOMES)

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** ×	× ×
[12, 20] 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**
[230, 231] 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**
[15] 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**
[14] 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**
[232] 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**
[233] 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**

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 [230, 231]. 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 [15] 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 [237] and correlation [238] 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 [221] 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 [239]. 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 tech-

niques, such as *war driving* or *war flying* that enabled them to launch the attack from distances up to 350 meters.

In [14] 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 [63, 240] 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 [232], 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.* [233] 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 [219] 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 [241] and on home cameras (privacy violations) [242].

10 Mitigating IoT-enabled cyber attacks

From the analysis of the attacks presented above, 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. To be consistent with our risk-based assessment methodology, 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 [243] using cost/benefit analysis.

In Table 9 we present a detailed mapping of the proposed security controls

with all the characteristics they positively affect. Table 9 also shows which of the examined attacks could be mitigated (at least partially) by each security control, for all attack paths. Finally, 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).

10.1 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 usually depends on the specific system environment, the implementation of these controls usually relies on the system operator.

Limit physical access to IoT Avoid installing IoT devices in places that are physically accessible to unauthorized users. Otherwise, apply suitable physical protection controls, (*e.g.* install the IoT device in a locked cabinet).

Monitor physical access to IoT Physical access to IoT devices should be monitored (*e.g.* with surveillance controls), 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 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.

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 GCM mode, to ensure data encryption and integrity at the same time (by default IEEE 802.15.4 does not apply any security mode). Another example is the use of security extensions for AdHoc networks, such as those described in [244, 245], to deal with wormhole and sybil attacks.

Table 9: A summary of the security controls for IoT-enabled cyber attacks. For each control we indicate the main threat and/or vulnerability characteristics that are mitigated (at least partially) by the control. We also indicate which of the examined attacks would require this security control, for all attack paths

×	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. ×	Tamp-ering	Embed. Crypto	Implem-entation	Firm-ware	Operat. System	Applic-ation	Netw. Design	Link-Layer	Netw.-Layer	Appl.-Layer	No PKC	Common /No Key	Extract-able Key	Direct ×	Indirect ×	No conn. ×	
Limit physical access to IoT	✓				✓													[61, 164, 171, 174, 227]	[57, 82, 176]	[172, 228]	A
Monitor physical access to IoT	✓				✓													[61, 70, 110, 174]	[57, 82]	[228]	A
Avoid direct Internet access		✓	✓	✓					✓	✓			✓	✓	✓			[89, 91, 121, 137]	[178]	[12, 230, 232]	A
Enforce proxy-based access		✓	✓	✓					✓	✓				✓				[62, 89, 91, 110, 121, 137, 138, 164, 174, 227]	[56, 85]	[228, 232]	A
Secure remote access		✓	✓	✓									✓	✓				[62, 89, 91, 110, 121, 137, 138]	[56, 85, 178]	[?, 12, 232]	A
Apply security extensions for link-layer protocols	✓		✓	✓								✓						[144, 163, 164, 177, 197]	[143, 147]	[233]	A
Log and monitor access to IoT	✓	✓	✓	✓														[136]	[178, 179]	[12, 230]	A
Audit access to IoT	✓	✓	✓	✓															[178, 179]		A
Tamper resistance mechanisms	✓				✓	✓	✓	✓								✓	✓	[61, 110, 136, 144, 199]	[143]	[14]	M
Secure embedded crypto	✓	✓			✓	✓						✓	✓					[61, 121, 197]		[?]	M
Side-channel attack protection	✓				✓		✓													[?, 22, 229]	M
Firmware protection	✓	✓	✓	✓				✓										[61, 136, 137, 163, 199]	[56, 176]	[?]	M
Secure firmware update	✓							✓										[61, 163, 199]	[56, 176]	[?, 16, 229]	M
Secure OS architecture									✓									[62, 171, 177, 227]	[9, 10, 56, 57, 179, 208]	[12, 228, 229]	M
OS hardening									✓									[89, 171, 174, 177, 197]	[9, 10, 57, 179, 208]	[12, 228, 229]	M
Use of secure API										✓								[121, 137, 138, 176, 199]	[57, 82, 85, 145, 176]	[12, 14, 16, 22, 230, 232]	M
Code auditing										✓								[137, 138, 174, 199]	[57, 82, 85, 146, 176]	[14, 22, 230]	M
Network security protocols												✓	✓	✓				[110, 121, 144, 163, 164, 171]	[9, 10, 56, 143, 145, 146]	[22]	M
Secure key management																✓	✓	[171]	[176]	[?, 14, 16, 233]	M
Secure key exchange															✓	✓		[171]		[?, 16]	A
Device acquiring criteria					✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	[136, 174, 177, 197]		[232]	O
Secure change management						✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	[70, 171]	[57, 208]		A
Continuous security testing					✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	[177]	[9, 10, 208]	[?, 12, 229, 232, 233]	A
Security standards enforcement					✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	[144, 197]	[9, 10, 143, 146]	[12]	R
Identify IoT dependencies	✓	✓	✓	✓														[174]	[176, 179]	[16, 22, 172, 228]	A
Re-examine BYOD policies	✓		✓	✓																[14, 172]	A
Avoid physical proximity		✓	✓	✓															[82, 147]	[16, 22, 228, 229]	A
Segment networks to avoid cascading impact	✓	✓	✓	✓							✓							[70, 227]	[9, 10, 56, 57, 57, 145, 146, 176, 178, 179, 208]	[?]	A
Favor technology diversity	✓	✓	✓	✓							✓							[62]		[?]	O

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

Log and monitor access to IoT Continuously log and monitor access to/from IoT devices. When possible, use Intrusion Detection/Prevention Systems (IDS/IPS) to monitor access to IoT devices, 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.

10.2 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.

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.

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 [15] and should be avoided.

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 [15].

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.

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 firmwares to be installed.

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.

Use of secure APIs When developing application software for IoT devices, the developers should use only secure and tested Application Programming Interfaces (API) that provide tested software development libraries and prevent well-known software vulnerabilities (like 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).

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 [44].

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 [38].

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.

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.

10.3 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.

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 hidden/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 [221].

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.

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.

By examining Table 9 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. Finally, some controls, such as those related with software security, seem to be of high priority for all attack path scenarios.

11 Conclusions

From the analysis of the recent IoT-enabled cyber attacks, it is obvious that IoT enabling technologies are radically changing the threat landscape on any sector they are applied. 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.

11.1 Gap Analysis

Based on the analysis of the examined cyber attacks, we summarize the relative research and implementation gaps, in comparison to the existing state-of-the-art security controls (see Table 10). 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.*, [248–253]) research gaps can be still identified in all the layers of IoT platforms [257]. For example, sophisticated attacks such as [22] 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 [256], also require novel technologies (*e.g.* Blockchain [246, 247]) 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 [31, 34], including the support of efficient public key management, despite the recent advances [38].

11.2 Discussion

In this paper we have reviewed and assessed verified IoT-enabled attacks in various application domains, in a risk-based approach. The goal of this survey is to point out the significance of underestimated attack paths in various IoT sectors and to provide a useful insight, both for security researchers and for critical system operators.

Industrial SCADA and smart grids favor the direct connectivity attack path scenarios, since modern field devices provide web interfaces for remote monitoring and control [62, 90, 91, 121, 137]. However, indirect attack paths may also occur. Since SCADA command and control centers can interact with corporate networks, attacks such as spear phishing [9, 10] 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 [82, 83].

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 [56,146,171,176,178]. Similarly, outdated, interconnected, passive medical devices [57,208] 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 [199].

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.* [12, 20, 22, 230]). In other cases home IoT devices may also serve as the actual target of the attack (*e.g.* ransomware attacks [14, 232, 233]). Finally, smart automation devices, that are installed inside the premises of critical infrastructures, can also be used to indirectly attack their nearby critical systems [16] or even to exfiltrate sensitive data from nearby systems [22].

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.

Various sector-specific standardization efforts, like Industry 4.0 [258], attempt to incorporate this continuously changing threat landscape. In addition, recent ongoing legislation efforts [259] attempt to cover this gap and to enforce the use of IoT with a high level of security in critical domains. It is worth to mention that various IoT security testbed labs (*e.g.* [260–262]) are also in the direction of helping the IoT industry to ensure a high security level.

References

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[2] L. Da Xu, W. He, and S. Li, "Internet of things in industries: A survey," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2233–2243, 2014.

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

[4] T. G. Lewis, *Critical infrastructure protection in homeland security: defending a networked nation.* John Wiley & Sons, 2014.

[5] C. Alcaraz and S. Zeadally, "Critical control system protection in the 21st century: Threats and solutions," *IEEE Computer*, vol. 46, no. 10, pp. 74 – 83, 2013 2013.

[6] ENISA, "European Union Agency for Network and Information Security," https://www.enisa.europa.eu/topics/threat-risk-management/threats-and-trends, retrieved on February 2017, 2017.

[7] ICS-CERT, "The Industrial Control Systems Cyber Emergency Response Team," https://ics-cert.us-cert.gov, retrieved on February 2017, 2017.

[8] L. Cazorla, C. Alcaraz, and J. Lopez, "Cyber stealth attacks in critical information infrastructures," *IEEE Systems Journal*, pp. 1–15, 03/2016 2016.

[9] R. M. Lee, M. J. Assante, and T. Conway, "Analysis of the cyber attack on the Ukrainian power grid," *SANS Industrial Control Systems*, 2016.

[10] D. Goodin. (2017) Hackers trigger yet another power outage in Ukraine. [Online]. Available: https://arstechnica.com/security/2017/01/the-new-normal-yet-another-hacker-caused-power-outage-hits-ukraine/

[11] T. Bateman. (2013) Police warning after drug traffickers' cyber-attack (The BBC). [Online]. Available: http://www.bbc.com/news/world-europe-24539417

[12] S. Cobb. (2016) 10 things to know about the October 21 IoT DDoS attacks. [Online]. Available: http://www.welivesecurity.com/2016/10/24/10-things-know-october-21-iot-ddos-attacks/

[13] M. Nawir, A. Amir, N. Yaakob, and O. B. Lynn, "Internet of things (iot): Taxonomy of security attacks," in *2016 3rd International Conference on Electronic Design (ICED)*, Aug 2016, pp. 321–326.

[14] E. Fernandes, J. Jung, and A. Prakash, "Security analysis of emerging smart home applications," in *Security and Privacy, 2016 IEEE Symposium on*. IEEE, 2016, pp. 636–654.

[15] E. Ronen, A. Shamir, A.-O. Weingarten, and C. O'Flynn, "IoT goes nuclear: Creating a zigbee chain reaction," in *Security and Privacy (SP), 2017 IEEE Symposium on*. IEEE, 2017, pp. 195–212.

[16] A. Chapman. (2014) Hacking into internet connected light bulbs. [Online]. Available: https://www.contextis.com//resources/blog/hacking-internet-connected-light-bulbs/

[17] A. Humayed, J. Lin, F. Li, and B. Luo, "Cyber-physical systems security–a survey," *IEEE Internet of Things Journal (pre-print version)*, 2017.

[18] L. Buttyan, D. Gessner, A. Hessler, and P. Langendoerfer, "Application of wireless sensor networks in critical infrastructure protection: challenges and design options [security and privacy in emerging wireless networks]," *IEEE Wireless Communications*, vol. 17, no. 5, pp. 44–49, October 2010.

[19] C. Alcaraz, J. Lopez, and S. Wolthusen, "OCPP protocol: Security threats and challenges," *IEEE Transactions on Smart Grid*, vol. PP, no. 99, pp. 1–1, 2017.

[20] T. Greene. (2016) How the Dyn DDoS attack unfolded. [Online]. Available: http://www.networkworld.com/article/3134057/security/how-the-dyn-ddos-attack-unfolded.html

[21] P. Kotzanikolaou, M. Theoharidou, and D. Gritzalis, "Assessing n-order dependencies between critical infrastructures," *International Journal of Critical Infrastructures 6*, vol. 9, no. 1-2, pp. 93–110, 2013.

[22] E. Ronen and A. Shamir, "Extended functionality attacks on IoT devices: The case of smart lights," in *2016 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, 2016, pp. 3–12.

[23] S. Babar, P. Mahalle, A. Stango, N. Prasad, and R. Prasad, "Proposed security model and threat taxonomy for the internet of things (IoT)," in *International Conference on Network Security and Applications*. Springer, 2010, pp. 420–429.

[24] C. Alcaraz, P. Najera, J. Lopez, and R. Roman, "Wireless sensor networks and the internet of things: Do we need a complete integration?" in *1st International Workshop on the Security of the Internet of Things (SecIoT'10)*, 2010, pp. 1–8.

[25] R. Roman, P. Najera, and J. Lopez, "Securing the internet of things," *Computer*, vol. 44, no. 9, pp. 51–58, 2011.

[26] B. Miller and D. Rowe, "A survey of SCADA and critical infrastructure incidents," in *Proceedings of the 1st Annual Conference on Research in Information Technology*. ACM, 2012, pp. 51–56.

[27] C. Alcaraz, R. Roman, P. Najera, and J. Lopez, "Security of industrial sensor network-based remote substations in the context of the internet of things," *Ad Hoc Networks*, vol. 11, no. 3, pp. 1091–1104, 2013.

[28] R. Roman, J. Zhou, and J. Lopez, "On the features and challenges of security and privacy in distributed internet of things," *Computer Networks*, vol. 57, no. 10, pp. 2266–2279, 2013.

[29] C. Miller and C. Valasek, "A survey of remote automotive attack surfaces," *black hat USA*, vol. 2014, p. 94, 2014.

[30] K. Stouffer, S. Lightman, V. Pillitteri, M. Abrams, and A. Hahn, "NIST special publication 800-82, revision 2: Guide to industrial control systems (ICS) security," *Gaithersburg, MD, USA: National Institute of Standards and Technology*, 2014.

[31] J. Granjal, E. Monteiro, and J. S. Silva, "Security for the internet of things: a survey of existing protocols and open research issues," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1294–1312, 2015.

[32] ——, "Security in the integration of low-power wireless sensor networks with the internet: A survey," *Ad Hoc Networks*, vol. 24, pp. 264–287, 2015.

[33] A.-R. Sadeghi, C. Wachsmann, and M. Waidner, "Security and privacy challenges in industrial internet of things," in *Proceedings of the 52nd Annual Design Automation Conference*. ACM, 2015, p. 54.

[34] P. Pongle and G. Chavan, "A survey: Attacks on RPL and 6LoWPAN in IoT," in *Pervasive Computing (ICPC), 2015 International Conference on*. IEEE, 2015, pp. 1–6.

[35] D. Airehrour, J. Gutierrez, and S. K. Ray, "Secure routing for internet of things: A survey," *Journal of Network and Computer Applications*, vol. 66, pp. 198–213, 2016.

[36] K. Sonar and H. Upadhyay, "A survey: DDoS attack on internet of things," *International Journal of Engineering Research and Development*, vol. 10, no. 11, pp. 58–63, 2014.

[37] R. H. Weber, "Internet of things–new security and privacy challenges," *Computer law & security review*, vol. 26, no. 1, pp. 23–30, 2010.

[38] K.-A. Shim, "A survey of public-key cryptographic primitives in wireless sensor networks," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 577–601, 2016.

[39] A. Mosenia and N. K. Jha, "A comprehensive study of security of internet-of-things," *IEEE Transactions on Emerging Topics in Computing*, vol. 5, no. 4, pp. 586–602, 2017.

[40] I. Yaqoob, E. Ahmed, I. A. T. Hashem, A. I. A. Ahmed, A. Gani, M. Imran, and M. Guizani, "Internet of things architecture: Recent advances, taxonomy, requirements, and open challenges," *IEEE wireless communications*, vol. 24, no. 3, pp. 10–16, 2017.

[41] I. Yaqoob, E. Ahmed, M. H. ur Rehman, A. I. A. Ahmed, M. A. Al-garadi, M. Imran, and M. Guizani, "The rise of ransomware and emerging security challenges in the internet of things," *Computer Networks*, vol. 129, pp. 444–458, 2017.

[42] C. Bormann, A. P. Castellani, and Z. Shelby, "CoAP: An application protocol for billions of tiny Internet nodes," *IEEE Internet Computing*, vol. 16, no. 2, p. 62, 2012.

[43] N. Kushalnagar, G. Montenegro, and C. Schumacher, "IPv6 over low-power wireless personal area networks (6LoWPANs): overview, assumptions, problem statement, and goals," Tech. Rep., 2007.

[44] R. Roman, C. Alcaraz, J. Lopez, and N. Sklavos, "Key management systems for sensor networks in the context of the internet of things," *Computers & Electrical Engineering*, vol. 37, no. 2, pp. 147–159, 2011.

[45] J. Song, S. Han, X. Zhu, A. K. Mok, D. Chen, and M. Nixon, "A complete wirelessHART network," in *Proceedings of the 6th ACM conference on Embedded network sensor systems*. ACM, 2008, pp. 381–382.

[46] M. Nixon and T. Round Rock, "A comparison of WirelessHART and ISA100.11a," *Emerson Process Management*, pp. 1–36, 2012.

[47] Z. Shelby and C. Bormann, *6LoWPAN: The wireless embedded Internet*. John Wiley & Sons, 2011, vol. 43.

[48] Y. Yan, Y. Qian, H. Sharif, and D. Tipper, "A survey on cyber security for smart grid communications," *IEEE Communications Surveys & Tutorials*, 2012.

[49] Z. A. Baig and A.-R. Amoudi, "An analysis of smart grid attacks and countermeasures," *Journal of Communications*, vol. 8, no. 8, pp. 473–479, 2013.

[50] N. Komninos, E. Philippou, and A. Pitsillides, "Survey in smart grid and smart home security: Issues, challenges and countermeasures," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 4, pp. 1933–1954, 2014.

[51] H. He and J. Yan, "Cyber-physical attacks and defences in the smart grid: a survey," *IET Cyber-Physical Systems: Theory & Applications*, vol. 1, no. 1, pp. 13–27, 2016.

[52] NIST SGIP, "Guidelines for smart grid cybersecurity vol. 1 – Smart Grid cybersecurity strategy, architecture, and high-level requirements," NISTIR 7628 Revision 1, 2014.

[53] C. Levy-Bencheton and E. Darra, "Cyber security and resilience of intelligent public transport: good practices and recommendations," *ENISA*, December 2015.

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

[55] T. Heer, O. Garcia-Morchon, R. Hummen, S. L. Keoh, S. S. Kumar, and K. Wehrle, "Security challenges in the IP-based Internet of Things," *Wireless Personal Communications*, vol. 61, no. 3, pp. 527–542, 2011.

[56] C. Miller and C. Valasek, "Remote exploitation of an unaltered passenger vehicle," *Black Hat USA*, pp. 1–91, 2015.

[57] Independent Security Evaluators (Technical Report), "Securing hospitals: A research study and blueprint," Tech. Rep., 2016. [Online]. Available: https://www.securityevaluators.com/wp-content/uploads/2017/07/securing_hospitals.pdf

[58] R. Beaulieu, S. Treatman-Clark, D. Shors, B. Weeks, J. Smith, and L. Wingers, "The SIMON and SPECK lightweight block ciphers," in *Design Automation Conference (DAC), 2015 52nd ACM/EDAC/IEEE*. IEEE, 2015, pp. 1–6.

[59] C. Miller and C. Valasek, "Adventures in automotive networks and control units," *Def Con*, vol. 21, pp. 260–264, 2013.

[60] T. Eden. (2016) The absolute horror of WiFi light switches. [Online]. Available: https://shkspr.mobi/blog/2016/03/the-absolute-horror-of-wifi-light-switches/

[61] R. Spenneberg, M. Brüggemann, and H. Schwartke, "PLC-blaster: A worm living solely in the PLC," *Black Hat Asia, Marina Bay Sands, Singapore*, pp. 1–16, 2016.

[62] D. Formby, S. Durbha, and R. Beyah. (2017) Out of control: Ransomware for industrial control systems. [Online]. Available: http://www.cap.gatech.edu/plcransomware.pdf

[63] D. Fisher. (2016) What's on TV tonight? ransomware. [Online]. Available: https://www.onthewire.io/whats-on-tv-tonight-ransomware/

[64] D. Brewer, *An Introduction to ISO/IEC 27001: 2013*. BSI British Standards Institution, 2013.

[65] E. ISO, "Iec 27005: 2011 (en) information technology–security techniques–information security risk management switzerland," *ISO/IEC*, 2011.

[66] R. S. Ross, "Guide for conducting risk assessments (NIST SP-800-30rev1)," *The National Institute of Standards and Technology (NIST), Gaithersburg*, 2012.

[67] P. Bowen, J. Hash, and M. Wilson, "Information security handbook: a guide for managers," in *NIST special publication 800-100, National Institute of Standards and Technology*. Citeseer, 2007.

[68] 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.

[69] M. Theoharidou, P. Kotzanikolaou, and D. Gritzalis, "A multi-layer criticality assessment methodology based on interdependencies," *Computers & Security*, vol. 29, no. 6, pp. 643–658, 2010.

[70] J. Radcliffe, "Hacking medical devices for fun and insulin: Breaking the human SCADA system," *Black Hat USA Conference, White Paper*. [Online]. Available: https://media.blackhat.com/bh-us-11/Radcliffe/BH_US_11_Radcliffe_Hacking_Medical_Devices_WP.pdf

[71] S. Boddy and J. Shattuck, "The hunt for IoT: The rise of thinkbots (F5 Labs Technical Report)," July 2017.

[72] F. Maggi, D. Quarta, M. Pogliani, M. Polino, A. M. Zanchettin, and S. Zanero, "Rogue robots: Testing the limits of an industrial robot's security," Trend Micro, Politecnico di Milano, Tech. Rep., 2017.

[73] S. A. Boyer, *SCADA: supervisory control and data acquisition*. International Society of Automation, 2009.

[74] B. Zhu, A. Joseph, and S. Sastry, "A taxonomy of cyber attacks on SCADA systems," in *Internet of things (iThings/CPSCom), 2011 international conference on and 4th international conference on cyber, physical and social computing*. IEEE, 2011, pp. 380–388.

[75] R. E. Johnson, "Survey of SCADA security challenges and potential attack vectors," in *Internet Technology and Secured Transactions (ICITST), 2010 International Conference for*. IEEE, 2010, pp. 1–5.

[76] I. N. Fovino, A. Carcano, M. Masera, and A. Trombetta, "An experimental investigation of malware attacks on SCADA systems," *International Journal of Critical Infrastructure Protection*, vol. 2, no. 4, pp. 139–145, 2009.

[77] V. M. Igure, S. A. Laughter, and R. D. Williams, "Security issues in SCADA networks," *Computers & Security*, vol. 25, no. 7, pp. 498–506, 2006.

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

[79] J. D. Fernandez and A. E. Fernandez, "SCADA systems: vulnerabilities and remediation," *Journal of Computing Sciences in Colleges*, vol. 20, no. 4, pp. 160–168, 2005.

[80] I. N. Fovino, M. Masera, L. Guidi, and G. Carpi, "An experimental platform for assessing SCADA vulnerabilities and countermeasures in power plants," in *Human System Interactions (HSI), 2010 3rd Conference on*. IEEE, 2010, pp. 679–686.

[81] G. P. Devarajan, "Unraveling SCADA protocols: Using sulley fuzzer," *DEF CON*, vol. 15.

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

[83] D. Kushner, "The real story of Stuxnet," *IEEE Spectrum*, vol. 50, no. 3, pp. 48–53, 2013.

[84] A. K. Sood and R. J. Enbody, "Targeted cyberattacks: A superset of advanced persistent threats," *IEEE security & privacy*, vol. 11, no. 1, pp. 54–61, 2013.

[85] K. Wilhoit, "The SCADA that didn't cry wolf," *Trend Micro Inc., White Paper*, 2013. [Online]. Available: https://www.trendmicro.de/cloud-content/us/pdfs/security-intelligence/white-papers/wp-the-scada-that-didnt-cry-wolf.pdf

[86] D. Wichers, "Owasp top-10 2013," *OWASP, February*, 2013.

[87] G.-Y. Liao, Y.-J. Chen, W.-C. Lu, and T.-C. Cheng, "Toward authenticating the master in the MODBUS protocol," *IEEE Transactions on Power Delivery*, vol. 23, no. 4, pp. 2628–2629, 2008.

[88] J. Matherly, "Complete guide to Shodan," 2017. [Online]. Available: https://leanpub.com/shodan

[89] M. R. HD. (2015) The internet of gas station tank gauges. [Online]. Available: https://blog.rapid7.com/2015/01/22/the-internet-of-gas-station-tank-gauges/

[90] H. R. Jon. (2015) The internet of gas station tank gauges – take 2. [Online]. Available: https://blog.rapid7.com/2015/11/18/the-internet-of-gas-station-tank-gauges-take-2/

[91] D. Quarta, M. Pogliani, M. Polino, F. Maggi, A. M. Zanchettin, and S. Zanero, "An experimental security analysis of an industrial robot controller," in *Security and Privacy (SP), 2017 IEEE Symposium on*. IEEE, 2017, pp. 268–286.

[92] R. Beyah and D. Formby. (2017) Simulated attack shows ICS weakness (a new form of ransomware to take over control of a simulated water treatment plant). [Online]. Available: http://www.isssource.com/simulated-attack-shows-ics-weakness/

[93] W. Kyle and H. Stephen, "The gaspot experiment: Unexamined perils in using gas-tank-monitoring systems," TrendMicro - TrendLabs, Tech. Rep., 2015. [Online]. Available: http://www.trendmicro.it/media/wp/the-gaspot-experiment-wp-en.pdf

[94] B. Nagpal, P. Sharma, N. Chauhan, and A. Panesar, "DDoS tools: Classification, analysis and comparison," in *Computing for Sustainable Global Development (INDIACom), 2015 2nd International Conference on*. IEEE, 2015, pp. 342–346.

[95] A. Keyhani and M. Marwali, *Smart power grids*. Springer, 2012.

[96] J. Momoh, *Smart grid: fundamentals of design and analysis*. John Wiley & Sons, 2012, vol. 63.

[97] H. Farhangi, "The path of the smart grid," *IEEE power and energy magazine*, vol. 8, no. 1, 2010.

[98] L. Wenpeng, "Advanced metering infrastructure," *Southern Power System Technology*, vol. 3, no. 2, pp. 6–10, 2009.

[99] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, and G. P. Hancke, "Smart grid technologies: Communication technologies and standards," *IEEE transactions on Industrial informatics*, vol. 7, no. 4, pp. 529–539, 2011.

[100] F. Mwasilu, J. J. Justo, E.-K. Kim, T. D. Do, and J.-W. Jung, "Electric vehicles and smart grid interaction: A review on vehicle to grid and renewable energy sources integration," *Renewable and Sustainable Energy Reviews*, vol. 34, pp. 501–516, 2014.

[101] Q. Wang, X. Liu, J. Du, and F. Kong, "Smart charging for electric vehicles: A survey from the algorithmic perspective," *IEEE Communications Surveys Tutorials*, vol. 18, no. 2, pp. 1500–1517, 2016.

[102] H. Zhang, Z. Hu, Z. Xu, and Y. Song, "Evaluation of achievable vehicle-to-grid capacity using aggregate PEV model," *IEEE Transactions on Power Systems*, vol. 32, no. 1, pp. 784–794, 2017.

[103] S. Xie, W. Zhong, K. Xie, R. Yu, and Y. Zhang, "Fair energy scheduling for vehicle-to-grid networks using adaptive dynamic programming," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 8, pp. 1697–1707, Aug 2016.

[104] J. Kang, R. Yu, X. Huang, S. Maharjan, Y. Zhang, and E. Hossain, "Enabling localized peer-to-peer electricity trading among plug-in hybrid electric vehicles using consortium blockchains," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 6, pp. 3154–3164, Dec 2017.

[105] N. Saxena, S. Grijalva, V. Chukwuka, and A. V. Vasilakos, "Network security and privacy challenges in smart vehicle-to-grid," *IEEE Wireless Communications*, vol. 24, no. 4, pp. 88–98, 2017.

[106] D. Deka, R. Baldick, and S. Vishwanath, "Jamming aided generalized data attacks: exposing vulnerabilities in secure estimation," in *System Sciences (HICSS), 2016 49th Hawaii International Conference on*. IEEE, 2016, pp. 2556–2565.

[107] S. Roy, "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.

[108] X. Liu and Z. Li, "Local topology attacks in smart grids," *IEEE Transactions on Smart Grid*, 2017.

[109] S. Xie, J. Yang, K. Xie, Y. Liu, and Z. He, "Low-sparsity unobservable attacks against smart grid: Attack exposure analysis and a data-driven attack scheme," *IEEE Access*, vol. 5, pp. 8183–8193, 2017.

[110] F. BRET-MOUNET, "All your solar panels are belong to me," *DEF CON*, vol. 24, pp. 4–7, 2016.

[111] A. Tajer, "False Data Injection Attacks in Electricity Markets by Limited Adversaries: Stochastic Robustness," *IEEE Transactions on Smart Grid*, 2017.

[112] J. C. Stephens, E. J. Wilson, and T. R. Peterson, *Smart grid (R) evolution*. Cambridge University Press, 2015.

[113] I. Stojmenovic and S. Wen, "The fog computing paradigm: Scenarios and security issues," in *Computer Science and Information Systems (FedCSIS), 2014 Federated Conference on*. IEEE, 2014, pp. 1–8.

[114] J. Duan, W. Zeng, and M.-Y. Chow, "Economic impact of data integrity attacks on distributed dc optimal power flow algorithm," in *North American Power Symposium (NAPS), 2015*. IEEE, 2015, pp. 1–7.

[115] R. Tan, V. B. Krishna, D. K. Yau, and Z. Kalbarczyk, "Integrity attacks on real-time pricing in electric power grids," *ACM Transactions on Information and System Security (TISSEC)*, vol. 18, no. 2, p. 5, 2015.

[116] H. Ye, Y. Ge, X. Liu, and Z. Li, "Transmission line rating attack in two-settlement electricity markets," *IEEE Transactions on Smart Grid*, vol. 7, no. 3, pp. 1346–1355, 2016.

[117] J. Giraldo, A. Cárdenas, and N. Quijano, "Integrity attacks on real-time pricing in smart grids: impact and countermeasures," *IEEE Transactions on Smart Grid*, 2017.

[118] J. Fan, Q. Li, and G. Cao, "Privacy disclosure through smart meters: Reactive power based attack and defense," in *Dependable Systems and Networks (DSN), 2017 47th Annual IEEE/IFIP International Conference on*. IEEE, 2017, pp. 13–24.

[119] P. Yi, T. Zhu, Q. Zhang, Y. Wu, and J. Li, "A denial of service attack in advanced metering infrastructure network," in *Communications (ICC), 2014 IEEE International Conference on*. IEEE, 2014, pp. 1029–1034.

[120] P. Yi, T. Zhu, Q. Zhang, Y. Wu, and L. Pan, "Puppet attack: a denial of service attack in advanced metering infrastructure network," *Journal of Network and Computer Applications*, vol. 59, pp. 325–332, 2016.

[121] A. Srivastava, T. Morris, T. Ernster, C. Vellaithurai, S. Pan, and U. Adhikari, "Modeling cyber-physical vulnerability of the smart grid with incomplete information," *IEEE Transactions on Smart Grid*, vol. 4, no. 1, pp. 235–244, 2013.

[122] S. Sridhar and M. Govindarasu, "Model-based attack detection and mitigation for automatic generation control," *IEEE Transactions on Smart Grid*, vol. 5, no. 2, pp. 580–591, 2014.

[123] A. Sargolzaei, K. Yen, and M. Abdelghani, "Delayed inputs attack on load frequency control in smart grid," in *Innovative smart grid technologies conference (ISGT)*. IEEE, 2014, pp. 1–5.

[124] J. Yan, Y. Tang, Y. Zhu, H. He, and Y. Sun, "Smart grid vulnerability under cascade-based sequential line-switching attacks," in *Global Communications Conference (GLOBECOM)*. IEEE, 2015, pp. 1–7.

[125] A. K. Farraj and D. Kundur, "On using energy storage systems in switching attacks that destabilize smart grid systems," in *Innovative Smart Grid Technologies Conference (ISGT)*. IEEE, 2015, pp. 1–5.

[126] F. Skopik and Z. Ma, "Attack vectors to metering data in smart grids under security constraints," in *Computer Software and Applications Conference Workshops (COMPSACW)*. IEEE, 2012, pp. 134–139.

[127] F. G. Marmol, C. Sorge, O. Ugus, and G. M. Pérez, "Do not snoop my habits: preserving privacy in the smart grid," *IEEE Communications Magazine*, vol. 50, no. 5, 2012.

[128] G. Liang, J. Zhao, F. Luo, S. Weller, and Z. Y. Dong, "A review of false data injection attacks against modern power systems," *IEEE Transactions on Smart Grid*, 2017.

[129] Y. Xiang, Z. Ding, Y. Zhang, and L. Wang, "Power system reliability evaluation considering load redistribution attacks," *IEEE Transactions on Smart Grid*, vol. 8, no. 2, pp. 889–901, 2017.

[130] M. Zeller, "Myth or reality: Does the aurora vulnerability pose a risk to my generator?" in *Protective Relay Engineers, 2011 64th Annual Conference for*. IEEE, 2011, pp. 130–136.

[131] A. Greenberg, "Hack brief: Hackers targeted a US nuclear plant (but don't panic yet)," 2017. [Online]. Available: https://www.wired.com/story/hack-brief-us-nuclear-power-breach/

[132] L. Robert and C. Anton. (2016) Blackenergy trojan strikes again: Ukrainian electric power industry. [Online]. Available: https://www.welivesecurity.com/2016/01/04/blackenergy$-trojan-strikes-again-attacks-ukrainian-electric-power-industry/

[133] C. Anton. (2017) Industroyer: Biggest threat to industrial control systems since Stuxnet. [Online]. Available: https://www.welivesecurity.com/2017/06/12/industroyer-biggest-threat-industrial-control-systems-since-stuxnet/

[134] Dragos Inc., "Crashoverride analysis of the threat to electric grid operations," 2017. [Online]. Available: https://dragos.com/blog/crashoverride/CrashOverride-01.pdf

[135] C. Anton, "Win32/Industroyer: A new threat for Industrial Control systems," ESET, Tech. Rep., 2017. [Online]. Available: https://www.welivesecurity.com/wpcontent/uploads/2017/06/Win32_Industroyer.pdf

[136] KrebsonSecurity. (2012) Fbi: Smart meter hacks likely to spread. [Online]. Available: https://krebsonsecurity.com/2012/04/fbi-smart-meter-hacks-likely-to-spread/

[137] T. Spring. (2016) Solar power firm patches meters vulnerable to command injection attacks. [Online]. Available: https://threatpost.com/solar-power-firm-patches-meters-vulnerable-to-command-injection-attacks/122324/

[138] F.-B. Thomas. (2015) Hundreds of wind turbines and solar systems wide open to easy exploits (forbes). [Online]. Available: https://www.forbes.com/sites/thomasbrewster/2015/06/12/hacking-wind-solar-systems-is-easy/#9c91c7a4d5c5

[139] M. A. Lisovich, D. K. Mulligan, and S. B. Wicker, "Inferring personal information from demand-response systems," *IEEE Security & Privacy*, vol. 8, no. 1, 2010.

[140] S. McLaughlin, D. Podkuiko, S. Miadzvezhanka, A. Delozier, and P. McDaniel, "Multi-vendor penetration testing in the advanced metering infrastructure," in *Proceedings of the 26th Annual Computer Security Applications Conference*. ACM, 2010, pp. 107–116.

[141] S. Mousavian, M. Erol-Kantarci, L. Wu, and T. Ortmeyer, "A risk-based optimization model for electric vehicle infrastructure response to cyber attacks," *IEEE Transactions on Smart Grid*, vol. PP, no. 99, pp. 1–1, 2017.

[142] Y. Zhang, S. Gjessing, H. Liu, H. Ning, L. T. Yang, and M. Guizani, "Securing vehicle-to-grid communications in the smart grid," *IEEE Wireless Communications*, vol. 20, no. 6, pp. 66–73, December 2013.

[143] J. Petit, B. Stottelaar, M. Feiri, and F. Kargl, "Remote attacks on automated vehicles sensors: Experiments on camera and Lidar," *Black Hat Europe*, vol. 11, pp. 1–13, 2015.

[144] C. Yan, X. Wenyuan, and J. Liu, "Can you trust autonomous vehicles: Contactless attacks against sensors of self-driving vehicle," *DEF CON*, vol. 24, pp. 1–50, 2016.

[145] C. Vallance. (2015) Car hack uses digital-radio broadcasts to seize control (BBC). [Online]. Available: http://www.bbc.com/news/technology-33622298

[146] K. Munro and D. Lodge. (2016) Hacking the Mitsubishi Outlander PHEV hybrid. [Online]. Available: https://www.pentestpartners.com/blog/hacking-the-mitsubishi-outlander-phev-hybrid-suv/

[147] L. Mearian. (2015) With 15 Dollars in Radio Shack parts, 14-year-old hacks a car. [Online]. Available: http://www.computerworld.com/article/2886830/with-15-in-radio-shack-parts-14-year-old-hacks-a-car.html

[148] G. Dimitrakopoulos and P. Demestichas, "Intelligent transportation systems," *IEEE Vehicular Technology Magazine*, vol. 5, no. 1, pp. 77–84, 2010.

[149] G. Leen and D. Heffernan, "Expanding automotive electronic systems," *Computer*, vol. 35, no. 1, pp. 88–93, 2002.

[150] J. Liu, S. Zhang, W. Sun, and Y. Shi, "In-vehicle network attacks and countermeasures: Challenges and future directions," *IEEE Network*, vol. 31, no. 5, pp. 50–58, 2017.

[151] Netronics. (2017) Cando: Can bus analyzer. [Online]. Available: http://www.cananalyser.co.uk/index.html

[152] D. Kraus, E. Leitgeb, T. Plank, and M. Löschnigg, "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*. IEEE, 2016, pp. 1–8.

[153] S.-C. Huang, B.-H. Chen, S.-K. Chou, J.-N. Hwang, and K.-H. Lee, "Smart car [application notes]," *IEEE Computational Intelligence Magazine*, vol. 11, no. 4, pp. 46–58, 2016.

[154] M. Maurer, J. C. Gerdes, B. Lenz, H. Winner *et al.*, *Autonomous driving*. Springer, 2016.

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

[156] R. D. Pascoe and T. N. Eichorn, "What is communication-based train control?" *IEEE Vehicular Technology Magazine*, vol. 4, no. 4, 2009.

[157] H. Dong, B. Ning, B. Cai, and Z. Hou, "Automatic train control system development and simulation for high-speed railways," *IEEE circuits and systems magazine*, vol. 10, no. 2, pp. 6–18, 2010.

[158] M. Strohmeier, M. Schafer, V. Lenders, and I. Martinovic, "Realities and challenges of nextgen air traffic management: the case of ADS-B," *IEEE Communications Magazine*, vol. 52, no. 5, pp. 111–118, 2014.

[159] A. Roy, "Secure aircraft communications addressing and reporting system (ACARS)," Jan. 13 2004, uS Patent 6,677,888.

[160] C. Livadas, J. Lygeros, and N. A. Lynch, "High-level modeling and analysis of the traffic alert and collision avoidance system (TCAS)," *Proceedings of the IEEE*, vol. 88, no. 7, pp. 926–948, 2000.

[161] R. Ward, C. Roberts, and R. Furness, "Electronic chart display and information systems (ecdis): State-of-the-art in nautical charting," *Marine and Coastal Geographical Information Systems*, pp. 149–161, 2000.

[162] D. C. Donderi, R. Mercer, M. B. Hong, and D. Skinner, "Simulated navigation performance with marine electronic chart and information display systems (ecdis)," *The Journal of Navigation*, vol. 57, no. 2, pp. 189–202, 2004.

[163] C. Cerrudo, "Hacking US traffic control systems," Presentation at DEFCON 22, 2014.

[164] B. Ghena, W. Beyer, A. Hillaker, J. Pevarnek, and J. A. Halderman, "Green lights forever: Analyzing the security of traffic infrastructure." *WOOT*, vol. 14, pp. 7–7, 2014.

[165] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham *et al.*, "Experimental security analysis of a modern automobile," in *Security and Privacy (SP), 2010 IEEE Symposium on*. IEEE, 2010, pp. 447–462.

[166] S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner, T. Kohno *et al.*, "Comprehensive experimental analyses of automotive attack surfaces." in *USENIX Security Symposium*. San Francisco, 2011, pp. 77–92.

[167] A. Greenberg. (2015) Hackers remotely kill a jeep on the highway—with me in it. [Online]. Available: https://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/

[168] N. Reem. (2015) Chrysler recalls 1.4 million cars after remote hacking of jeep (CNBC). [Online]. Available: http://www.nbcnews.com/tech/tech-news/chrysler-recalls-1-4-million-cars-after-remote-hacking-jeep-n397851

[169] D. Yadron and D. Tynan. (2016) Tesla driver dies in first fatal crash while using autopilot mode (The Guardian). [Online]. Available: https://www.theguardian.com/technology/2016/jun/30/tesla-autopilot-death-self-driving-car-elon-musk

[170] F. D. Garcia, D. Oswald, T. Kasper, and P. Pavlidès, "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.

[171] S. Gordaychik, A. Timorin, and G. T. Gritsai. (2015) The great train cyber robbery. Presentation at the Chaos Communication Congress (CCC). [Online]. Available: https://media.ccc.de/v/32c3-7490-the_great_train_cyber_robbery

[172] B. Chen, C. Schmittner, Z. Ma, W. G. Temple, X. Dong, D. L. Jones, and W. H. Sanders, "Security analysis of urban railway systems: the need for a cyber-physical perspective," in *International Conference on Computer Safety, Reliability & Security*. Springer, 2015, pp. 277–290.

[173] A. Costin and A. Francillon, "Ghost is in the air (traffic)," *Black Hat USA (July 2012)*, pp. 1–9, 2012.

[174] H. Teso, "Aircraft hacking: Practical aero series," *Hack In The Box*, pp. 1–44, 2013.

[175] R. Klein, "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*. IEEE, 2015, pp. 1–15.

[176] R. Santamarta. (2016) In flight hacking system (IOActive Research Labs). [Online]. Available: http://blog.ioactive.com/2016/12/in-flight-hacking-system.html

[177] M. Balduzzi, K. Wihoit, and A. Pasta, "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.

[178] M. Ballano, "AmosConnect: Maritime communications security has its flaws," *IOActive*, 2017. [Online]. Available: http://blog.ioactive.com/2017/10/amosconnect-maritime-communications.html

[179] P. Beaumont, "Cyber-risks in maritime container ports: An analysis of threats and simulation of impacts," 2017.

[180] J. Leyden, "Polish teen derails tram after hacking train network," *The Register*, vol. 11, 2008.

[181] K. Zetter. (2012) Hackers breached railway network, disrupted service (The Wired). [Online]. Available: https://www.wired.com/2012/01/railyway-hack/

[182] L. Andrew. (2016) Hackers are holding San Francisco's light-rail system for ransom (The Verge). [Online]. Available: https://www.theverge.com/2016/11/27/13758412/hackers-san-francisco-light-rail-system-ransomware-cybersecurity-muni

[183] G. Chris. (2017) Cyber attack hits German train stations as hackers target Deutsche Bahn (The Telegraph). [Online]. Available: http://www.telegraph.co.uk/news/2017/05/13/cyber-attack-hits-german-train-stations-hackers-target-deutsche/

[184] E. Kovaks. (2015) Trains vulnerable to hacker attacks (Securityweek). [Online]. Available: http://www.securityweek.com/trains-vulnerable-hacker-attacks-researchers

[185] T. Yunusov. (2015) Critical vulnerabilities in 3G/4G modems or how to build big brother. [Online]. Available: http://blog.ptsecurity.com/2015/12/critical-vulnerabilities-in-3g4g-modems.html

[186] J. Leyden. (2013) Airports' passport controls shut down by 'malware' (The Register). [Online]. Available: https://www.theregister.co.uk/2013/07/31/istanbul_airport_chaos_malware_blamed/

[187] W. Szary and E. Auchard. (2015) Polish airline, hit by cyber attack, says all carriers are at risk (The Reuters). [Online]. Available: http://www.reuters.com/article/us-poland-lot-cybercrime-idUSKBN0P21DC20150622

[188] M. Robinson. (2017) Terror fears over hundreds of 'ghost ships' turning off their tracking devices (The Dailymail). [Online]. Available: http://www.dailymail.co.uk/news/article-4300170/Terror-fears-hundreds-ghost-ships-turning-GPS.html

[189] Y. Torbati and J. Saul. (2012) Iran's top cargo shipping line says sanctions damage mounting (The Reuters). [Online]. Available: http://www.reuters.com/article/us-iran-sanctions-shipping-idUSBRE89L10X20121022

[190] J. DiRenzo, D. A. Goward, and F. S. Roberts, "The little-known challenge of maritime cyber security," in *Information, Intelligence, Systems and Applications (IISA), 2015 6th International Conference on.* IEEE, 2015, pp. 1–5.

[191] K. Munro. (2017) OSINT from ship satcoms. [Online]. Available: https://www.pentestpartners.com/security-blog/osint-from-ship-satcoms/

[192] S. R. Islam, D. Kwak, M. H. Kabir, M. Hossain, and K.-S. Kwak, "The internet of things for health care: a comprehensive survey," *IEEE Access*, vol. 3, pp. 678–708, 2015.

[193] P. J. Soh, G. A. Vandenbosch, M. Mercuri, and D. M.-P. Schreurs, "Wearable wireless health monitoring: Current developments, challenges, and future trends," *IEEE Microwave Magazine*, vol. 16, no. 4, pp. 55–70, 2015.

[194] A. Demosthenous, "Advances in microelectronics for implantable medical devices," *Advances in Electronics*, vol. 2014, 2014.

[195] M. H. Schoenfeld, S. J. Compton, R. H. Mead, D. N. Weiss, L. Sherfesee, J. Englund, and L. R. Mongeon, "Remote monitoring of implantable cardioverter defibrillators," *Pacing and clinical electrophysiology*, vol. 27, no. 6p1, pp. 757–763, 2004.

[196] H. Blauw, A. Van Bon, R. Koops, and J. DeVries, "Performance and safety of an integrated bihormonal artificial pancreas for fully automated glucose control at home," *Diabetes, Obesity and Metabolism*, vol. 18, no. 7, pp. 671–677, 2016.

[197] E. Marin, D. Singelée, F. D. Garcia, T. Chothia, R. Willems, and B. Preneel, "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*. ACM, 2016, pp. 226–236.

[198] T. Fox-Brewster. (2017) Medical devices hit by ransomware for the first time in US hospitals (Forbes). [Online]. Available: https://www.forbes.com/sites/thomasbrewster/2017/05/17/wannacry-ransomware-hit-real-medical-devices/#3ecc0f42425c

[199] R. Billy and B. Jonathan, "Security evaluation of the implantable cardiac device ecosystem architecture and implementation interdependencies." WhiteScope, Tech. Rep., 2017. [Online]. Available: https://www.a51.nl/sites/default/files/pdf/PacemakerEcosystemEvaluation.pdf

[200] 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. [Online]. Available: https://www.fda.gov/MedicalDevices/Safety/AlertsandNotices/ucm573669.htm

[201] S. Gollakota, H. Hassanieh, B. Ransford, D. Katabi, and K. Fu, "They can hear your heartbeats: non-invasive security for implantable medical devices," *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 4, pp. 2–13, 2011.

[202] C. Li, M. Zhang, A. Raghunathan, and N. K. Jha, "Attacking and defending a diabetes therapy system," in *Security and Privacy for Implantable Medical Devices*. Springer, 2014, pp. 175–193.

[203] J. Liebowitz and R. Schaller, "Biological warfare: Tampering with implantable medical devices," *IT Professional*, vol. 17, no. 5, pp. 70–72, 2015.

[204] K. Zetter. (2015) Hacker can send fatal dose to hospital drug pumps. [Online]. Available: https://www.wired.com/2015/06/hackers-can-send-fatal-doses-hospital-drug-pumps/

[205] M. Khera, "Think like a hacker insights on the latest attack vectors (and security controls) for medical device applications," *Journal of Diabetes Science and Technology*, p. 1932296816677576, 2016.

[206] J. Kirk. (2012) Pacemaker hack can deliver deadly 830-Volt jolt. [Online]. Available: http://www.computerworld.com/article/2492453/malware-vulnerabilities/pacemaker-hack-can-deliver-deadly-830-volt-jolt.html

[207] D. Halperin, T. S. Heydt-Benjamin, B. Ransford, S. S. Clark, B. Defend, W. Morgan, K. Fu, T. Kohno, and W. H. Maisel, "Pacemakers and implantable cardiac defibrillators: Software radio attacks and zero-power defenses," in *2008 IEEE Symposium on Security and Privacy (S&P 2008)*. IEEE, 2008, pp. 129–142.

[208] TrapX Research, Labs, "Anatomy of Attack: MEDJACK.2 – Hospitals Under Siege," TrapX Investigative Report, 2016. [Online]. Available: https://trapx.com/wp-content/uploads/2017/08/AOA_Report_TrapX_MEDJACK.2.pdf

[209] D. Gayle, A. Topping, I. Sample, S. Marsh, and D. Vikram. (2017) NHS seeks to recover from global cyber-attack as security concerns resurface (The Guardian). [Online]. Available: https://www.theguardian.com/society/2017/may/12/hospitals-across-england-hit-by-large-scale-cyber-attack

[210] S. Kelly. (2016) Major cyberattacks on healthcare grew 63% in 2016. [Online]. Available: http://www.darkreading.com/attacks-breaches/major-cyberattacks-on-healthcare-grew-63--in-2016/d/d-id/1327779

[211] H. Huang, *PACS and imaging informatics: Basic principles and applications*. John Wiley & Sons, 2011.

[212] N. Weaver, V. Paxson, S. Staniford, and R. Cunningham, "A taxonomy of computer worms," in *Proceedings of the 2003 ACM workshop on Rapid malcode*. ACM, 2003, pp. 11–18.

[213] J. E. Brown, N. Smith, and B. R. Sherfy, "Decreasing mislabeled laboratory specimens using barcode technology and bedside printers," *Journal of nursing care quality*, vol. 26, no. 1, pp. 13–21, 2011.

[214] C. Withanage, R. Ashok, C. Yuen, and K. Otto, "A comparison of the popular home automation technologies," in *Innovative Smart Grid Technologies-Asia (ISGT Asia), 2014 IEEE*. IEEE, 2014, pp. 600–605.

[215] C. Gomez and J. Paradells, "Wireless home automation networks: A survey of architectures and technologies," *IEEE Communications Magazine*, vol. 48, no. 6, 2010.

[216] G. M. Toschi, L. B. Campos, and C. E. Cugnasca, "Home automation networks: A survey," *Computer Standards & Interfaces*, vol. 50, pp. 42–54, 2017.

[217] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, "Transmission of IPv6 packets over IEEE 802.15.4 networks," Tech. Rep., 2007. [Online]. Available: https://tools.ietf.org/html/rfc8137

[218] J. Wang, "Zigbee light link and its applicationss," *IEEE Wireless Communications*, vol. 20, no. 4, pp. 6–7, 2013.

[219] O. Colin, "A lightbulb worm? Details of the Philips Hue smart lighting design (Black Hat USA 2016 White Paper)," 2016.

[220] M. B. Barcena and C. Wueest, "Insecurity in the internet of things," *Security Response, Symantec*, 2015.

[221] C. P. O'Flynn, "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*. IEEE, 2011, pp. 1–5.

[222] B. Fouladi and S. Ghanoun, "Honey, I'm home!! Hacking Z-Wave home automation systems," *Black Hat USA*, pp. 1–53, 2013.

[223] N. Lomas, "Critical flaw identified in zigbee smart home devices," 2015.

[224] C. W. Badenhop, S. R. Graham, B. W. Ramsey, B. E. Mullins, and L. O. Mailloux, "The z-wave routing protocol and its security implications," *Computers & Security*, vol. 68, pp. 112–129, 2017.

[225] V. Mathy and P. Frank, "Key reinstallation attacks: Forcing nonce reuse in wpa2," *DistriNet*, 2017.

[226] D. Goodin. (2017) Leak of >1,700 valid passwords could make the IoT mess much worse (Ars Technica). [Online]. Available: https://arstechnica.com/information-technology/2017/08/leak-of-1700-valid-passwords-could-make-the-iot-mess-much-worse/

[227] M. Garrett. (2016) I stayed in a hotel with android lightswitches and it was just as bad as you'd imagine. [Online]. Available: https://mjg59.dreamwidth.org/40505.html

[228] G. Hernandez, O. Arias, D. Buentello, and Y. Jin, "Smart nest thermostat: A smart spy in your home," *Black Hat USA*, pp. 1–8, 2014.

[229] Wikileaks. (2017) Vault 7: CIA Hacking Tools Revealed - CIA malware targets iPhone, Android, smart TVs. [Online]. Available: https://wikileaks.org/ciav7p1/

[230] S. Tenaglia and J. Tanen, "Breaking BHAD: Abusing Belkin home automation devices," *Black Hat Europe*, pp. 1–46, 2016.

[231] N. Dhanjani, *Abusing the internet of things: blackouts, freakouts, and stakeouts.* " O'Reilly Media, Inc.", 2015.

[232] R. Scheel. (2017) Smart TV hacking. (Oneconsult talk at EBU Media Cyber Security Seminar). [Online]. Available: https://www.oneconsult.com/en/smart-tv-hacking/

[233] P. Morgner, S. Mattejat, and Z. Benenson, "All your bulbs are belong to us: Investigating the current state of security in connected lighting systems," *arXiv preprint arXiv:1608.03732*, 2016.

[234] Proofpoint. (2014) More than 750,000 phishing and spam emails launched from "thingbots" including televisions, fridge. [Online]. Available: http://investors.proofpoint.com/releasedetail.cfm?releaseid=819799

[235] A. Nixon, J. Costello, and Z. Wilkholm, "An after-action analysis of the Mirai botnet attacks on Dyn," 2016.

[236] Y. Musashi, M. Kumagai, S. Kubota, and K. Sugitani, "Detection of kaminsky dns cache poisoning attack," in *Intelligent Networks and Intelligent Systems (ICINIS), 2011 4th International Conference on.* IEEE, 2011, pp. 121–124.

[237] P. Kocher, J. Jaffe, B. Jun, and P. Rohatgi, "Introduction to differential power analysis," *Journal of Cryptographic Engineering*, vol. 1, no. 1, pp. 5–27, 2011.

[238] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Annual International Cryptology Conference.* Springer, 1999, pp. 388–397.

[239] C. Müller, F. Armknecht, Z. Benenson, and P. Morgner, "On the security of the zigbee light link touchlink commissioning procedure," in *Sicherheit*, 2016.

[240] W. Candid. (2015) How my TV got infected with ransomware and what you can learn from it. [Online]. Available: https://www.symantec.com/connect/blogs/how-my-tv-got-infected-ransomware-and-what-you-can-learn-it

[241] C. Cesar and A. Lucas, "Hacking robots before Skynet (IOActive)," 2017. [Online]. Available: https://ioactive.com/pdfs/Hacking-Robots-Before-Skynet.pdf

[242] C. Owens. (2016) Stranger hacks family's baby monitor and talks to child at night. [Online]. Available: http://sfglobe.com/2016/01/06/stranger-hacks-familys-baby-monitor-and-talks-to-child-at-night/

[243] G. Stergiopoulos, P. Kotzanikolaou, M. Theocharidou, and D. Gritzalis, "Risk mitigation strategies for critical infrastructures based on graph centrality analysis," *International Journal of Critical Infrastructure Protection*, vol. 10, pp. 34–44, 2015.

[244] I. Dhyani, N. Goel, G. Sharma, and B. Mallick, "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.

[245] J. G. Ponsam and R. Srinivasan, "A survey on MANET security challenges, attacks and its countermeasures," *International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)*, vol. 3, no. 1, 2014.

[246] K. Biswas and V. Muthukkumarasamy, "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*. IEEE, 2016, pp. 1392–1393.

[247] S. Huh, S. Cho, and S. Kim, "Managing IoT devices using Blockchain platform," in *Advanced Communication Technology (ICACT), 2017 19th International Conference on*. IEEE, 2017, pp. 464–467.

[248] C. Wachsmann and A.-R. Sadeghi, "Physically unclonable functions (PUFs): Applications, models, and future directions," *Synthesis Lectures on Information Security, Privacy, & Trust*, vol. 5, no. 3, pp. 1–91, 2014.

[249] J. X. Zheng, T. Xu, and M. Potkonjak, "Securing embedded systems and their IPs with digital reconfigurable PUFs," in *Power and Timing Modeling, Optimization and Simulation (PATMOS), 2016 26th International Workshop on*. IEEE, 2016, pp. 169–176.

[250] V. Zimmer, J. Sun, M. Jones, and S. Reinauer, *Embedded Firmware Solutions: Development Best Practices for the Internet of Things*. Apress, 2015.

[251] A. Danese, G. Pravadelli, and V. Bertacco, "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*. ACM, 2017, p. 9.

[252] D. Gruss, J. Lettner, F. Schuster, O. Ohrimenko, I. Haller, and M. Costa, "Strong and efficient cache side-channel protection using hardware transactional memory," in *USENIX Security Symposium*, 2017.

[253] H. Gross, S. Mangard, and T. Korak, "An efficient side-channel protected AES implementation with arbitrary protection order," in *Cryptographers' Track at the RSA Conference*. Springer, 2017, pp. 95–112.

[254] U. Guin, S. Bhunia, D. Forte, and M. M. Tehranipoor, "Sma: A system-level mutual authentication for protecting electronic hardware and firmware," *IEEE Transactions on Dependable and Secure Computing*, vol. 14, no. 3, pp. 265–278, 2017.

[255] Y. Lee, J. Jeong, and Y. Son, "Design and implementation of the secure compiler and virtual machine for developing secure IoT services," *Future Generation Computer Systems*, vol. 76, pp. 350–357, 2017.

[256] S. Klein, "Azure event hubs," in *IoT Solutions in Microsoft's Azure IoT Suite*. Springer, 2017, pp. 273–289.

[257] J. Mineraud, O. Mazhelis, X. Su, and S. Tarkoma, "A gap analysis of internet-of-things platforms," *Computer Communications*, vol. 89, pp. 5–16, 2016.

[258] N. Jazdi, "Cyber physical systems in the context of industry 4.0," in *Automation, Quality and Testing, Robotics, 2014 IEEE International Conference on*. IEEE, 2014, pp. 1–4.

[259] M. Warner, C. Gardner, R. Wyden, and S. Daines, "S.1691 - Internet of Things (IoT) Cybersecurity Improvement Act of 2017 (Introduced to US Congress 08/01/2017)." [Online]. Available: https://www.congress.gov/bill/115th-congress/senate-bill/1691/text?format=txt

[260] iTrust Centre for Research in Cyber Security, Singapore University of Technology and Design. (2015) Internet of things automatic security testbed. [Online]. Available: https://itrust.sutd.edu.sg/research/testbeds/internet-of-things/

[261] A. P. Mathur and N. O. Tippenhauer, "SWaT: A water treatment testbed for research and training on ICS security," in *Cyber-physical Systems for Smart Water Networks (CySWater), 2016 International Workshop on*. IEEE, 2016, pp. 31–36.

[262] A. S. Aragó, E. R. Martínez, and S. S. Clares, "SCADA laboratory and test-bed as a service for critical infrastructure protection," in *2nd International Symposium on ICS & SCADA Cyber Security Research 2014*, ser. ICS-CSR 2014. BCS, 2014, pp. 25–29.

He has received various certifications in information security.

He is a member of the IEEE and the IEEE Computer Society and a member of ACM.

Table 10: Gap analysis for IoT security: Research and implementation gaps

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 & 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.* [22] demonstrate the need for further research in this area.	- Physical access protection & 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 [246, 247] that utilize 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 [248–251]. - Resilience against side-channel attacks (*e.g.* DPA/CPA/Photonic) [252, 253].	- H/W integrity checks [254] 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 & hardening* *Secure APIs* *Code auditing*	- State of the art mechanisms include firmware signing, code obfuscation, protected boot process, secure coding & compiling techniques [255] and cloud services security (*e.g.* [256]).	- There is a need for novel cross-layer SW protection mechanisms of IoT devices (*e.g.* against ransomware attacks [41, 235]) and platforms [257].	- 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