SN Computer Science (2022) 3:173
https://doi.org/10.1007/s42979-022-01061-9

ORIGINAL RESEARCH

A Secure Method for Industrial IoT Development

Sebastiao Beethoven Brandao Filho[1] · **Cecilia de Azevedo Castro Cesar**[1]

Received: 25 November 2021 / Accepted: 9 February 2022 / Published online: 25 February 2022

Abstract
Due to the wide scope of technical issues involved, IIoT projects require Systems Development Methods (SDMs) to deal with the entire complex process. With so many issues to solve, security is often left until the main functional pillars have already been established. Thus, the security standards for IIoT and IoT SDMs are not fully integrated, which leads to projects that do not meet the security-by-Design requirement. In this article, we present IgniteSec, a new method that combines the Ignite System Development Method with the security standard for industrial systems NIST 800-82. The case study carried out with IgniteSec shows that, by uniting these two fronts from the first stages, the project arrives at the software design stage with the important functional and security elements well integrated into the design and in compliance with security standard.

Keywords Industrial Internet of Things · System Development Methods · Security-by-Design · Security norms · NIST 800-82

Introduction

The large number of currently connected devices, which is almost 10 times the world population, is justified due to the extremely positive results of Internet of Things (IoT) projects and the real, measurable impact they have had. A market analysis forecast from Gartner company predicts that, in 2024, the IoT platform market will grow to $7.6 billion with a 31% CAGR [1]. As the processes have evolved, the complexity of the processes has also grown. More and more, in an Industrial IoT (IIoT) project, multiple enabling technologies must be integrated with each other and multiple stakeholders must be involved, without losing sight of the business focus. It is not easy to satisfy all interested parties, deal with different areas of knowledge, and meet the project's development timeline without neglecting any important details. The vast scope of the currently multidisciplinary project and the complexity of its life cycle justify some structuring of the developmental design steps, using Systems Development Methods (SDMs). In the beginning of the so-called digital transformation, designers used Software Development Methods exclusively, but the cyber-physical nature of IoT projects requires different modeling, with the inclusion of physical systems that encompass hardware, software, and communication components. SDMs cover the entire project life cycle [2] while the traditional reference models and architectures only define the structure of the system at a conceptual level [3]. Formalization is even more necessary in the IoT environment than in the classic computing environment, since the IoT system incorporates physical things, sensors, actuators, connected devices, and many stakeholders. The benefits of using SDMs for IoT include ways of expressing thoughts, formalizing procedures, developing systems in a systematic way that encompass functional and non-functional requirements, and reducing errors in the development process [4]. Despite the use of SDMs, the question of where and when to address security concerns remains. Security is a top concern for IoT as identified in several surveys [5]. One difficulty is that cybersecurity in IoT is not limited to a single aspect; it covers the protection of data flows, cyber-physical components, and all the security gaps that can be exploited for the purposes of industrial espionage, or just for ransomware purposes. Another difficulty in securing IoT is that cryptography approaches are not always used because they demand too much computing power and memory space, which are highly limited resources in IoT devices [6]. There is growing evidence,

✉ Cecilia de Azevedo Castro Cesar
ceciliaccesar@gmail.com

Sebastiao Beethoven Brandao Filho
beethovenbrandaofilho@gmail.com

[1] ITA: Instituto Tecnologico de Aeronautica, Sao José dos Campos, Brazil

however, that a delay in designing security into the IoT system is short-sided and, ultimately, not cost-effective. The CCDCOE (NATO Cooperative Cyber Defense Centre of Excellence) argues that the typical approach of proposing high-level policy principles is not always sufficient to incorporate all the requirements defined by the technology space into the policy design process. The environment is complex and dynamic, and policymakers are not tech experts and technologists are not policy experts [7]. So, there needs to be a way to integrate policies and technologies systematically. In the last 9 years, more than 2,232 CVEs (Common Vulnerability and Exposure) were reported in Industrial Control Systems [8]. The numerous attacks reported can destroy productive efforts as well as ruin the industry's reputation. Customers and consumers are increasingly aware that they need to purchase reliable products, which puts pressure on the industry in the pursuit of this excellence. Consequently, the security issue has been increasingly considered in Models, Architectures, and SDMs. The design manager is well aware of the need for Security-by-Design; but while this manager knows what to do, he does not necessarily know how to do it, or when to apply each step of the security process. What we have seen in the past is that the security issue has been incorporated into more advanced stages of the project causing patches that do not suit either the functional team or the security team. Without a systematic approach that defines what should be done and when, the teams compete, defending their own point of view, harming the entire project. If security is not one of the design criteria, the software engineers do not address it! If the security issue is already present at the idea stage, the security aspect is incorporated in the creation of the prototype—and is upheld through all production stages. We consider that it would not be a good strategy to propose, a series of steps that contemplate security within a development methodology without compatibility with the already established norms. So, the path adopted was to analyze Industrial Security norms and select one solid, widely used norm that was still suitable for insertion in a SDM. We studied SDMs and security norms and selected a pair, integrating them to create a reliable framework for IoT development. As a SDM, we chose the Ignite Method because of the completeness and relevance of the method, which is based on real-world experience [9]. As a security norm, we chose NIST 800-82, because it has gained particular recognition in the Industrial Control Systems field for being very clear, concise, largely used in Industry and available for free [10]. The choice of Ignite and NIST 800-82 is well supported by our review of the literature presented below. Our main objective is to generate a Systems Development Method for the industrial environment that, in addition to guiding the important steps of the development process, includes a well-established security standard. Our proposal is called IgniteSec.

The rest of the paper is organized as follows: "A Review of the Literature" presents a review of the literature regarding IoT SDMs and Security norms; "Background—Ignite and NIST 800-82" briefly describes the Ignite Method and the NIST 800-82 that form the basis of the proposed system. "IgniteSec" presents IgniteSec. Our solution is applied to a road accident alert system in "Case Study". Finally, the conclusions are in "Conclusion".

A Review of the Literature

Existing SDMs for IoT have flaws: some are missing steps, some are exhaustive and produce too many documents, and some neglect important aspects, such as security. We analyzed some of the most relevant and most cited methods. The GSEM-IoT method [11] adapts concepts from traditional software development and applies them to the IoT domain, adopting key abstractions for IoT systems engineering. These abstractions can be the basis for a general IoT-oriented software engineering discipline, but they need many additions to cover all the aspects of a design for a more complex system, for example, the business model or the hardware design. The IoT-AD method [12] covers more aspects, offering a conceptual framework and a development framework. It is particularly good for the initial stages, where the method seeks to normalize the vocabulary. Another strong point of the method is its capacity to assign responsibilities to the different agents. Its drawback is that the artifacts do not address the development of IoT services or the hardware and communication components. ELDAMeth [13] also devotes much effort to software development. It proposes 3 phases: Modeling, Simulation, and Implementation. The modeling phase does not consider business aspects; nor does it contemplate engineering requirements. A strong point is the inclusion of simulations that advance and solve many problems without having developed the platform-specific code yet. A new approach for IoT development is presented by [14] based on metamodels, which are the construction of a collection of "concepts" within a given domain. A Metamodel is proposed for each development phase of Analysis, Design, and Implementation, capturing the level of abstraction in each phase. But this approach remains only within the scope of software engineering, without a systemic view of the project. To adapt methods to a specific context, Giray and Tekinerdogan [15] propose Situational Methods Engineering (SME) to build SDMs, where relevant aspects of the specific situation are compiled from a base method to create a new method that is adapted to the actual case. This seems like a good idea, but it requires prior knowledge of the ISO/IEC 24744 standard and RUP (Rational Unified Process) taxonomy, from which method fragments and situational factors are extracted, before applying the technique

to generated the new method. IoT-Methodology (IOTM) is a method for developing IoT projects in different areas whose creators also provide tools to follow the different stages of development [16]. IoTM covers several important stages of the project but not all and not fully integrated, such as functional design and technical design. Also, it does not address security. Agile methods are among the most used in organizations although they are not specific to the IoT ecosystem. They promote interactions between parties and development in an ever-changing environment, which is common in IoT projects. The Scrum method is one of the oldest and most used, but it has some limitations for IoT, such as not incorporating hardware design and lack of a holistic view [17]. Another agile style method that has been used in IoT is the Scaled Agile Framework (SAFe) method which has the benefit of a fast learning curve and the size of the admitted team which can be large. SAFe covers more aspects than Scrum, but does not cover the IoT system view as a whole [18]. We found alternative models emphasizing some specific aspect of the project, like [19]. The authors propose a model to increase efficiency at the network layer, capturing the complexity of transactions and dynamically assigning the traffic flow in the network devices. They benefit from the SDN approach to making decisions centrally, with a clear IIoT architecture. Nevertheless, they do not include modeling the entire ecosystem in their approach, leaving out aspects, such as stakeholder discussions, business considerations, and hardware design. Studies [20] and [4] compare SDMs used in IoT. Merzouk et al. [20] evaluate the Ignite, Scrum, Kanban, Scaled Agile Framework, and IoT-Methodology methods by comparing them according to criteria that provide an overview of each method. The Ignite and IoT-Methodology methods were identified as the first two methods designed for IoT Domain. The authors considered all the methods for the IoT scenario inappropriate in some respect. In [4], evaluation criteria to analyze existing IoT SDMs are presented. The criteria include artifacts defined, process steps, support for life cycle activities, coverage of IoT system elements, metrics, and rigidity of the method. The methods evaluated are Ignite, IoT Methodology, IoT-AD, ELDAMeth, SPLP-IoT, and GSEM-IoT. The study points out that none of the identified IoT SDMs cover all the necessary phases for developing IoT systems.

In regards to security, we identified main standards used in industry, in general, all over the world: the ISO /IEC 2700 family of standards, the IEC 62443, and the NIST 800-82. These norms are defined for an Industrial environment, since they define requirements for IT suppliers, System integrators, and network operators. Other standards appeared in our search, but they are only recommendations, or guidance like EN 303 645 for consumer IoT devices, or NERC CIP defined exclusively for the electrical sector. With its focus on Industrial Automation and Control System (IACS), the ISA/IEC 62443 standard [21] has good relative security coverage, relevance for manufacturers, and a great number of details for the operation. This norm has a comprehensive scope, promoting the collaboration for the Information Technology (IT) and the Operational Technology (OT) departments. Its strong point is that cybersecurity is treated as an ongoing process and not a goal that can be achieved, which is in line with SDMs. The problem with this norm is that the set of documents is expensive and extensive; and there is little free information. For this reason, we did not consider it the best standard to be used in projects for small- and medium-sized companies, or for academic purposes. The family of norms ISO 27000 [22] has good relative security coverage, a great number of details for the operation, widespread in IT, but of little relevance for manufacturing since it does not cover OT issues. Finally, we chose the NIST 800-82 [23] for our project, because of its completeness, wide scope, easy access, and great acceptance.

An IoT project must start with the idea generation phase involving stakeholders, go through a business model, manage the identified opportunity, outline a solution, move on to the functional project where the identified components are integrated and finally reach the technical project where the software architecture, hardware design and technical infrastructure are defined. Given all these required steps, the gaps shown in our survey are clear: (1) some of the methods focus on the software design stage (GSEM-IoT , ELDAMeth, Scrum, SAFe); (2) some of the methods cover more steps beyond software design, but leave some important steps aside (IoT-AD, IOTM); (3) some of them require a lot of prior knowledge and are difficult to use in practice (SME, Metamodel); (4) few methods consider security and do so superficially (only Ignite explicitly includes the security theme); (5) none of the methods is integrated with security standards. After studying and applying some of these methods, we chose Ignite to incorporate the security standard in its stages. The choice is due to the completeness of Ignite and for being used in practice in projects in the industry. Corroborating our decision, the authors of [4] conclude: "Ignite presents a more holistic view of developing IoT systems than the rest of SDMs", p. 159.

Background—Ignite and NIST 800-82

The fundamentals used in the project will be briefly explained in this section. The first subsection presents the basics of Ignite, the SDM used, and the second subsection presents the standard used - NIST 800-82. "IgniteSec" will present our solution that unites them.

Ignite

The Ignite Method [24] foresees two well-marked stages in the development of the project. The first is IoT Strategy Execution to define the organization's strategy, where topics related to elements of a high hierarchical level are covered. The second is IoT Solution Execution, which involves planning the project, as well as building and executing the IoT system. The next two subsections detail these stages. Figure 1 shows IoT Strategy Execution and IoT Solution Execution with a summary of the activities of each step. This figure summarizes the process flow of Ignite using Business Process Model Notation (BPMN) [25]. This notation uses the symbols:

- **X** : only one of the branches can be traversed;
- **+** : the tasks can be performed in parallel.

IoT Strategy Execution

It is not easy to deal with disruptive paradigms. In this first stage, Ignite seeks a better understanding of the transformation roadmap and how to manage a portfolio of opportunities. This stage is dedicated to the following activities: discussion of ideas, strategic visions and objectives, development of a business model, consideration of impacts and risks, approval of the IoT opportunity and consideration of partnerships. See the top of Fig. 1. These activities can be adapted to the vertical needs. Bringing the OT and IT areas together from the start is challenging, but it is the key to success, and Ignite integrates these areas.

IoT Solution Execution

In this phase, stakeholders are called upon to act with their skills and interests to be incorporated into the project. In the initial design of the project, the constituent elements, the architecture, and the profile of the technologies to be employed are identified. The project life cycle here is divided into planning, building, and executing IoT solutions. During the planning phase, the initial, smaller team is augmented to form the team that will later build the solution. The initial solution design defined key artifacts that cover activities related to analysis and planning, functional design, and technical design. The activities of the initial project are grouped as:

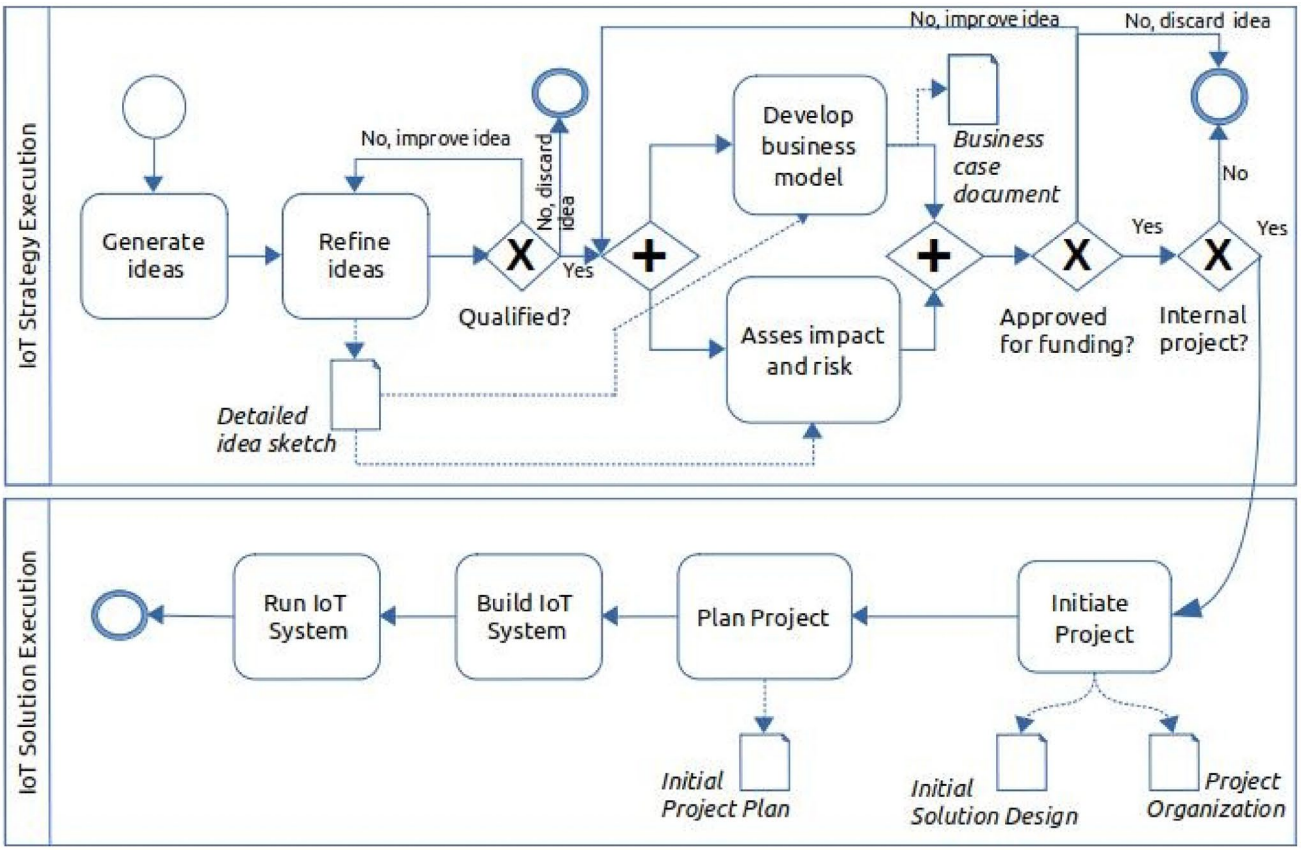

Fig. 1 Ignite process flow. Source [24]

- Analysis, projections and planning, where the team conducts an overview of the proposed solution design, stating the problem and involving stakeholders. This activity includes insights into the project assets and ends with a plan that defines the key milestones of the project.
- Functional design, in which a team proposes a basic design which is sufficient to start the implementation phase. It includes the most important Use Cases, User Interface Mockups, and Domain Model. This activity includes an architectural artifact called Asset Integration Architecture (AIA) to express the integration of the asset and the backend application.
- Technical design, where the team provides software architecture, hardware design, and the whole technical infrastructure.

The Asset Integration Architecture (AIA) from the Functional Design is an innovative addition of Ignite. It is an artifact to show the connections involved between assets and business applications through intermediate layers, identifying the fundamental elements that should be present in the system. In AIA, we can see the whole integrated system in one diagram. Ignite foresees security concepts that are inserted in the Asset Integration Architecture (AIA) in the Functional Design phase, but Ignite lacks detailed steps to design the security AIA. After this initial phase attains a complete vision, the project can be organized by workstreams, which are work packages responsible for key deliverables. For example, a communication workstream can be organized to deal with communication issues. One of the proposed workstreams is cross-cutting, which deals with tasks that depend on other workstreams. For Ignite, security actions are incorporated into the cross-cutting workstream. We argue that dealing with security actions at this stage is too late to comply with security standards. Our approach is shown in "IgniteSec".

NIST 800-82

Dedicated to the security of Industrial Control Systems (ICS), the NIST 800-82 standard [23] begins with an overview of ICS, emphasizing the differences between security in the Information Technology environment and security in the ICS environment. The standard presents a hierarchy that considers the following three levels:

i The organization level defines organizational strategies, policies, guidance, and processes for managing risk.
ii The mission/business process level addresses risk from a mission/business process perspective by developing the business missions/functions defined in the Organization Level.
iii The information system level integrates risk management activities into the system development life cycle of the organizational information systems.

Four planned activities are carried out continuously as the environment changes to address the Risk Management process. The activities are:

1. Framing risk: develops a framework for the risk management; considers existing disaster plans, requirements and makes assumptions about safety and security explicit;
2. Assessing risk: identifies the organization's threats and vulnerabilities, the damage an incident can cause to the organization, and the likelihood that such adverse events may actually occur;
3. Responding to risk: responds to the identification of risk (not of an actual incident);
4. Monitoring risk: monitors risk as an on-going process.

Our priority in this article is the initial steps of the project and, therefore, we will focus on activities 1 and 2 with an aid for activity 3. The standard proposes a Risk Management Framework (RMF) that integrates information security and risk management activities into the system development life cycle. The NIST norm 800-37 [26] addresses the RMF in general, but NIST 800-82 applies this framework to the ICS scenario. To apply RMF, the following sub-activities should be carried out:

(a) Categorizing the information system classifies the information and information system according to the potential level of impact of loss in terms of low, moderate, or high-impact on the security objectives of confidentiality, integrity, and availability.
(b) Selecting security controls involves a minimum of planned or already established security controls; controls are enumerated from eighteen security-related areas.
(c) Implementing security control applies the chosen controls to the project.
(d) Assessing security control ensures that the controls are implemented correctly, operating as intended, and producing the desired outcome.
(e) Authorizing information system accepts the risk explicitly based on the implementation of an agreed-upon set of security controls.
(f) Monitoring security controls continuously tracks changes that may affect security controls.

We believe that the first activities should be carried out as soon as possible in the project, the sooner, the better for the system as a whole. With this premise, the subsequent phases

of the life cycle will be able to apply the recommendations of the respective standard.

IgniteSec

To incorporate the tasks indicated by the security standard in the method of developing IoT systems, care must be taken not to overload a project with excessive controls when the project has barely gotten off the ground. It is important to emphasize that trained cross-functional personnel should be integrated into the first team so that they are discussing IoT Opportunity Management. Information security is a commercial responsibility shared by all members of the company and especially by the leading members of the business, process, and management teams. This justifies the early entry of the security team into the management team. Another premise is that the proposed process is cyclical and iterative. An activity started at a particular stage is not necessarily completed at that stage. It can be improved as the project progresses, as new information is outlined. Waiting for all information to be collected before addressing security would not produce Security-by-Design. The changes suggested by IgniteSec in the original Ignite paradigm are summarized in Fig. 2, where activities related to the NIST 800-82 standard were inserted. Figure 2 is derived from Fig. 1; it expands it, with the following differences:

- The parts of the IgniteSec process that have been changed or inserted in relation to Ignite have been painted gray.
- The middle part of Fig. 2 is dedicated to detailing the Initiation Project box shown in Fig. 1.
- The bottom part of Fig. 2 continues the initiation and shows the rest of the process.

These changes and their context will be explained next. Each step mentioned below can be better understood by following Fig. 2, where the chaining from one step to the next can be visualized. Figure 1 shows the activity "Develop Business models" that deals with scenario planning and uncertainty. It is important to explore those elements of the business model that generate impact and value in the context of a strategy [23]. Now, in IgniteSec, we include a business case for security at this moment that provides the business impact and financial justification for creating an integrated information security program. This is the right time for the group to ask itself, for example, how much downtime would cost due to security issues, what would be the harm of a data breach, or what the biggest beneficial impacts of investing in security would be [27]. Thus, our first insertion in the Ignite method incorporates the security business case when the business model is being elaborated, giving visibility to the risk issue right after the generation and refinement of ideas. And this insertion should be registered in the Business case document. Our second proposition is to insert the top level recommendations in the hierarchy of risk analysis from NIST into the activity "Assess impact and risk" (see Fig. 1). We inserted the risk analysis here at the organizational level and at the mission level as shown by the items (i) and (ii) indicated in "NIST 800-82". Risk assessment also has non-security related activities. For example, at the organizational level, the assessment can provide useful inputs to operational risk determinations and organizational risk determinations, including financial risk, compliance risk, regulatory risk, reputation risk, and multiple-impact risk which includes supply chain risk and risk involving partnerships. We suggest the first security steps to be included here, so that the first referrals are already engaged with the security issue. Organizations may differ in the approaches they prefer for a variety of reasons, but the organizational risk framework must be determined that outlines which risk models will be used and which approaches to risk assessment and analysis will be chosen. If the analysis at the organizational level is mature, it can be adapted from other previous projects, taking advantage of the company's culture. However, to continue the analysis made at the organizational level, the analysis at the mission level should be developed for this ongoing IoT Opportunity. The mission level determines process protection and resiliency requirements, and the allocation of those requirements to the enterprise architecture as part of mission/business segments. And the enterprise architecture is expected to incorporate the information security architecture. The mission level helps guide the allocation of security controls in the third level of the hierarchy, which is the information system level. at the information system level, risk assessment proceeds in the next phase shown in the middle part of Fig. 2. The first macro-activity in the Initiation Project stage is analysis, projections, and planning, where the key parts that make up the system are defined by narrowing down the solution scope. At this stage, when a solution sketch and site survey are done, new assets can emerge. If the initial risk analysis points to stringent security requirements, some innovative hardware designed to support security can be considered as in [28]. The hardware architecture details can be left to the technical design phase (see bottom of Fig. 2). After having contextualized knowledge about the assets and a solution sketch prepared, the project team can identify the critical parts of the project which is precisely the first step of the Risk Management Framework: (a) Categorize the Information System. This ensures that security controls will be in place and even improved during the steps that follow. If security considerations are not included here, the whole plan will fail. At this point, there is still

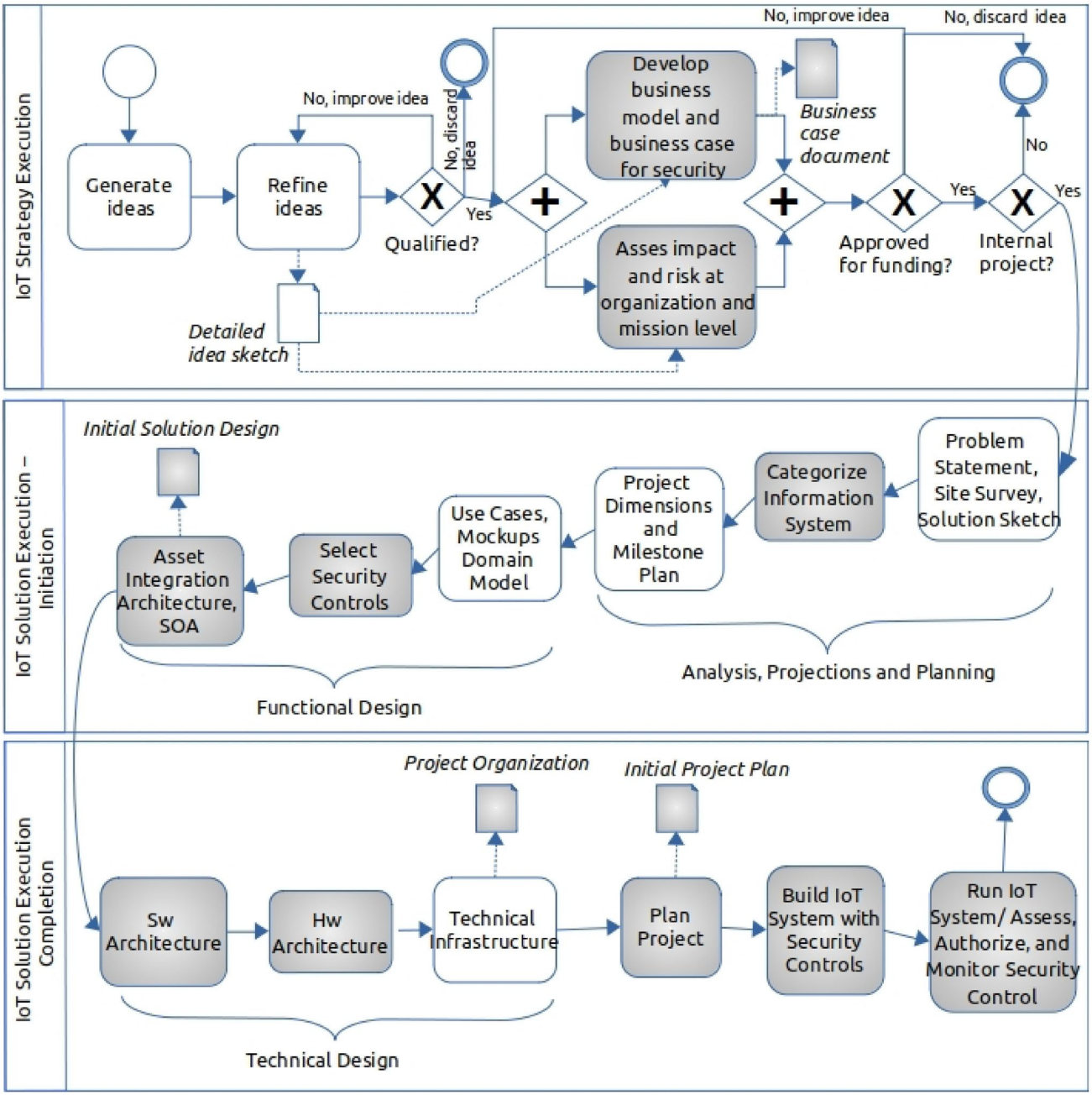

Fig. 2 IgniteSec methodology

no complete view of the solution, but the design teams already have an idea of the amount of effort that must be made to protect critical parts. The next macro activity is Functional Design, which contains the Domain Model among other preparatory activities for the implementation. The Domain Model emphasizes the key entities and their most important relationships, making it possible to select what is needed to protect them. With the knowledge of the Domain Model, the specific security controls for this project can be chosen, thus fulfilling the second step of the Risk Management Framework: (b) selecting the Security Controls. Security controls may involve aspects of policy, supervision, individual actions, or automated mechanisms implemented by information systems/devices. The functional designers must work in cooperation with the security specialists who have knowledge of all security controls suggested by the standard. Together, they select the best controls applicable to their project. As the project moves forward, Ignite creates the diagram called AIA, mentioned in "Background—Ignite and NIST 800-82". In this step,

we suggest an AIA that covers useful security concepts. Since, at this point, the security controls have already been selected, the task of performing a security AIA merely requires positioning the chosen controls on the layers of the diagram. The result of the Security AIA is reflected in the software architecture, the first activity of the Technical Design stage—see the third large block in Fig. 2. The main components of the AIA are transformed into software components. The hardware architecture activity is painted gray in Fig. 2 as it can also present its contribution to the secure solution. Completing the Technical Design phase, the Technical Infrastructure completes the technical project, raising the elements that make up the solution with an emphasis on the network infrastructure. To complete this initial solution design, the organizational structure of the project is set up, by defining workstreams. Ignite suggests a cross-cutting workstream to deal with the tasks that have dependencies in all subsequent workstreams, including security. In our opinion, security is, indeed, a cross-cutting subject, but it deserves a specific security workstream, because it is so complex, with many interfaces throughout the entire project. Even small projects must have dedicated staff and a dedicated workstream to achieve the goal of having a secure IoT system. Project organization is a document containing the result of the Technical Design used as an input to the Plan Project phase (see Fig. 2). As the architecture to integrate these security mechanisms has already been planned and the risk assessment is underway, a detailed plan for implementing the security architecture should be developed now in the Plan Phase, along with a strategy for testing and validating the implemented mechanisms. These plans are registered in a document called Initial Project Plan. In the Build Phase, the third step of the Risk Management Framework is carried out, which is (c) Implementing Security Control. Next, the fourth step of NIST 800-82 to assess implemented controls is completed: (d) Assessing Security Control. In the execution phase, steps (e) and (f) are performed to Authorize and Monitor security controls. Great effort must be devoted to monitoring as it is essential to track the behavior of the system to take action to respond to incidents as soon as possible.

Regarding the complexity of the project, although there is no universal standardization on a measure of complexity of a project, there are studies that quantify the complexity considering numerous factors [29]. In IT projects, the literature points out, among others, the following factors that influence the complexity of a project: human resources, skills, number of interfaces, use of new technologies, the use of incremental or iterative methodologies and the ability to predict risks and have options to minimize risk. In IgniteSec, on the one hand, the complexity of project development increases as security analysis is added, requiring more human resources, skills and more communication between the parties; however, on the other hand, this same analysis reduces the uncertainty factor, also providing actions that minimize risks, thus increasing the chances of project success.

Case Study

The main reference of the Ignite Method presents a case study called eCall [24]. It is a system for detecting and dealing with emergency situations in Smart Transportation. The goal is to reduce the time it takes to direct emergency services to the scene of a car accident, where a Call Center is processing incoming distress calls from vehicles and/or vehicle drivers. Figure 3 illustrates eCall with the elements that make up the solution:

- an on-board telematics control unit (TCU), which has an acceleration sensor integrated into the car's airbag; this unit communicates directly with the Call Center in the event of an accident without human intervention;
- a backend service which is the call center application;
- a telephony management service integrated into the application, the vehicle database and system partners: Public Security Service Station (PSAP), police station, fire department or ambulance service closest to the accident site.

The European Union has adopted legislation in this concern, as immediate information about an accident and the exact location has the potential to reduce the emergency service's response time by up to 60% in urban areas and can save 2,500 lives per year [30]. In fact, the European Union has approved regulations for the eCall system to be in all vehicles produced as of 2018, and, in 2017, the United Nation also proposed regulations for this type of system [31]. When surveying the first studies on eCall, some issues were found that threatened the success of these systems. The following criticisms regarding security were cited:

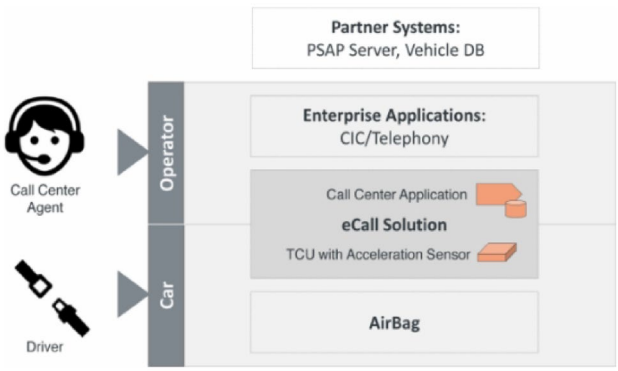

Fig. 3 eCall solution Source [24]

– Even though the accident data are only sent in case of an accident, which happens only occasionally, the communication system is always active, making the system constantly exposed to possible cyber attacks.
– The minimum data to be sent to a call center are not discussed sufficiently. And how these data will be treated is also not part of any of the existing regulations, which could lead to misuse.
– There are several aspects of the eCall system that can be affected by attackers:

  ⋆ Safety: harming users;
  ⋆ Financial: performing unauthorized transactions;
  ⋆ Operational: interfering with the intended operational performance; and
  ⋆ Privacy: gaining unauthorized access to data about the activities and the identity of vehicle owners or drivers [32].

– Finally, the systems can fail for structural reasons, like the loss of signal or the failure of some onboard unit, or for intentional reasons triggered by human activities [33].

These criticisms pose even greater responsibility on the system's conception. We took back the case study done by Slama et al. [24] with just Ignite, and applied IgniteSec to raise security-related suggestions from inception through the technical design phase.

Starting at the executive level, when carrying out the initial stage of the business model in conjunction with the business case for security, it was determined that the financial loss from a single security breach could jeopardize the existence of the business. A generic business case was created based on tangible and intangible costs and benefits that were collected to define how much would be invested and the return that could result from the investment. It was decided to invest in a basic security architecture that would deal seriously with possible violations. In parallel with the business model, we conducted a risk analysis at the organizational and the mission level. It was necessary to rank the impact of the failure of each part of the system. This analysis and ranking indicated that, if a failure or an attack prevented the accident vehicle from communicating with the Call Center, lives could be lost. Thus, the communication of the car controls with the Call Center was deemed the most critical. The next step in the Ignite method was to find a solution sketch. In the eCall system, managed assets were cars and the enterprise was the eCall operator. The main backend service was the call center application that would be integrated with local telephone management, with the Original Equipment Manufacturer (OEM) vehicle database, and with the public-safety answering point (PSAP). Figure 4 shows the AIA of the ecall connecting

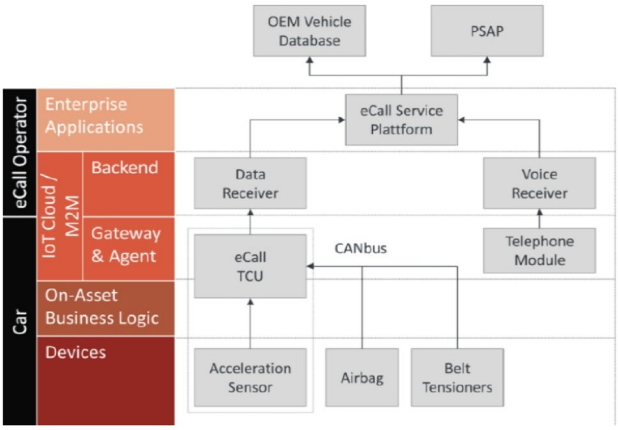

Fig. 4 AIA—eCall service Source [24]

these elements. This diagram from the authors of [24] does not consider security. This figure shows the functional side with all tiers integrated: in the lower side the devices and gateway inside the car and in the upper side the backend and applications in eCall Operator. Note that in the device layer, the Acceleration sensor, Airbag, and Belt tensioner devices integrate with the TCU that acts as a gateway. The TCU transmits the collected data to the Data Receiver service on the Backend tier. A telephone contact via voice service may also be in progress. The two communication routes reach the eCall platform at the Enterprise Applications tier. But, with IgniteSec, after the solution has been sketched, the risk analysis at the level of the information system begins. The first activity is to categorize the information system. In this security analysis done at the beginning of the project, it is possible to restrict our view to the communication system involved in eCall and not analyze the entire car or the security of all its components, because the components were not detailed yet, and so in this stage, the final result of an internal failure or a communication failure is the same: the operator will not be informed of the accident. In a future stage, a new risk analysis should be done, detailing new controls linked to defined components. According to the AIA from Fig. 4, two key communication segments were identified in the mission-level risk analysis:

– from TCU to the Call Center (TCU-CC);
– from the telephone module in the car to the Call-Center (Voice-CC).

We classified the main elements of the system in relation to the security pillars: confidentiality, integrity, and availability. In our context, each pillar refers to an aspect of the system:

– Confidentiality means that only the ecall app should handle the accident;

– Integrity means that no one can change/enter data relating to an accident;
– Availability means that all parts of the system must be available all the time, especially during an accident.

These three security objectives were associated with one of three levels of the potential impact of a security breach: low, moderate, and high impact. The TCU-CC segment cannot fail as it is the center of the system. The impact of confidentiality, integrity, or availability failures in this segment is high because if the sensor's data were not transmitted correctly, accurately, and privately, the entire mission will be compromised. Failures in the integrity and availability of the Voice-CC segment were considered to have a moderate impact, because, even if the driver was unconscious, the information from the TCU-CC segment would trigger the service. The confidentiality of the Voice-CC segment is also of high impact, as the conversation between the driver and the call center must be private without external influence on the service. The combined result of the two segments indicates that the final impact on the three pillars is considered high. The results of the impact analysis can be seen in Table 1. With this result, we selected security controls that could protect the system. Due to the budget reserved for security in the initial stages, the following minimal control families were selected:

– Access control (AC): defines policies and procedures for specifying the use of system resources by only authorized users, programs, processes, or other systems.
– Identification and authentication (IA): identifies potential network users, hosts, applications, services, and resources.
– Incident response (IR): evaluates the effect of an attack and the possible options to respond to limit consequences of incidents.
– System and communications protection (SC): defines mechanisms for protecting both system and data transmission components.
– System and information integrity (SI): assures that sensitive data have not been modified or deleted in an unauthorized and undetected manner.

After selecting the controls in IgniteSec, we were now in a position to produce a security AIA. Figure 5 contains this diagram. Regarding the position of each control, we decided that:

Table 1 Loss impact analysis of eCall systems

Segment	Confidentiality	Integrity	Availability
TCU-CC	High	High	High
Voice-CC	High	Moderate	Moderate
Result	High	High	High

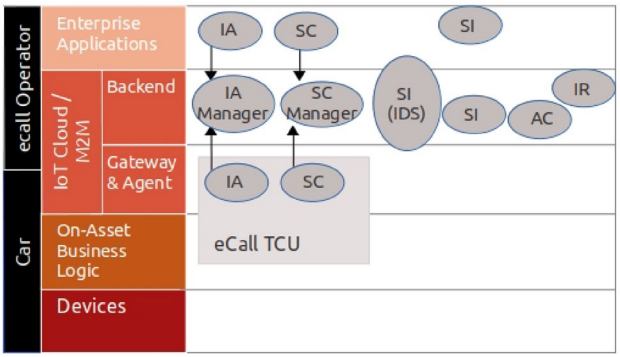

Fig. 5 Security controls in eCall AIA

– Identification and authentication (IA) be inserted in all tiers involved in communication tasks; in the Backend Service, IA Manager was positioned to control the whole process.
– The access control (AC) control the communication capabilities as part of the backend layer.
– Part of the incident response (IR) plan be implemented in the backend layer and provide further support for the decision about whether to send an ambulance to the indicated location. If this component is integrated with access control, it can interrupt certain actions when there is indication of a cyber attack.
– The system and communications protection (SC) control handle encryption. Each of the communicating parties must have a module that encrypts the information being transmitted to the outside.
– For the system and information integrity (SI) control, an antivirus be added to the application layer and to the backend as well as a network-based intrusion detection system on the LAN that links the application to the backend and the gateway to the backend.

These are the security elements that were incorporated into the functional elements presented in the project of Fig. 4. Functional and security elements should also be present in the next stage of the method, which would be the design of the software architecture.

Undoubtedly, it would be possible to further strengthen the entire system; however, the option was to minimally include these basic controls due to the budget set aside for the initial stages. Even with these basic decisions and no details, the project has already indicated which security controls should be incorporated in which portion of the project. The AIA diagram shown in Fig. 5 provides a wealth of additional detail over the diagram in Fig. 4 that

will allow you to guide your software project addressing security issues.

Conclusion

The complexity of the ecosystem involved in IIoT projects requires development methods and current security standards are not easily integrated into the process of developing IoT systems. IgniteSec combining the Ignite Systems Development Method with the NIST 800-82 Standard presents a systematic way to develop projects that incorporate the Security-by-Design requirement.

The advantages of IgniteSec are:

- Allows systematic development of projects considering security as strategic;
- Allows security to be included in all components of the business plan, conceptual modeling, software and hardware design;
- At the end of the design, the project complies with NIST 800-82 widely used in industrial environments.

The eCall case study emphasized the difference between this new approach and the traditional approaches. A system designed with IgniteSec provided a complete view conceived with security to the software design stage. A security diagram generated shows which security control is linked to which component. The inclusion of security analysis increases the required resources and complexity of the project, however, in the long term, this analysis will increase the chances of success of the project throughout its life cycle. In future work, we will continue to detail the recommendations of NIST 800-82 in the remaining stages of the project's life cycle, since, so far, we have only reached the beginning of software design. Another work front concerns the integration of the security workstream with the other workstreams. It is necessary to define which security controls require integration between teams.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Ethical approval This paper does not contain any studies with human participants or animals performed by any of the authors.

References

1. Middleton P, Velosa A, Biscotti F. Forecast analysis: enterprise IoT platforms, worldwide. Gartner Res. 2020. https://www.gartner.com/en/documents/3983783/forecast-analysis-enterprise-iot-platforms-worldwide. Accessed in 18 June 2021.
2. Fahmideh M, Zowghi D. An exploration of IoT platform development. Inf Syst. 2020. https://doi.org/10.1016/j.is.2019.06.005.
3. Hassan QF, Madani SA, Morrish J, Bhatnagar RM. Internet of Things: challenges, advances, and applications. Boca Raton: Taylor & Francis Group; 2018.
4. Tekinerdoğan B, Tüzün E, Giray G. IoT system development methods, Book chapter. Boca Raton: Taylor & Francis Group; 2018. https://doi.org/10.1201/9781315155005.
5. Positive Technologies. ICS vulnerabilities: 2018 in review. Accessed in: https://www.ptsecurity.com/upload/corporate/ww-en/analytics/ICS-vulnerabilities-2019-eng.pdf.
6. Yaacoub JA, Salman O, Noura HN, Kaaniche N, Chehab A, Malli M. Cyber-physical systems security: limitations, issues and future trends. Microprocess Microsyst. 2020;77:103201. https://doi.org/10.1016/j.micpro.2020.103201.
7. Trinca Ann Y, Vishik C, Matsubara M, Plonk A. Key concepts in cyber security: towards a common policy and technology context for cyber security norms. Tallinn: NATO CCD COE Publications; 2016.
8. Thomas RJ, Chothia T, et al. Learning from Vulnerabilities–categorising, understanding and detecting weaknesses in industrial control systems. In: Katsikas S, et al., editors. Computer security. CyberICPS 2020, SECPRE 2020, ADIoT 2020. Lecture notes in computer science, vol. 12501. Cham: Springer; 2020. https://doi.org/10.1007/978-3-030-64330-0_7.
9. Jacobson I, Spence I, Pan-Wei N. Is there a single method for the Internet of Things? Acm Queue. 2017. https://doi.org/10.1145/3121437.3123501.
10. Leszczyna R. Approaching secure industrial control systems. ET Inf Secur. 2014;9(1):81–9. https://doi.org/10.1049/iet-ifs.2013.0159.
11. Zambonelli F. Key abstractions for IoT-oriented software engineering. IEEE Softw IEEE. 2017;34(1):38–45. https://doi.org/10.1109/MS.2017.3.
12. Patel P, Cassou D. Enabling high-level application development for the Internet of Things. J Syst Softw 2015;103:62–84. arXiv:1501.05080.
13. Fortino G, Rango F, Russo W. ELDAMeth design process. In: Cossentino M, Hilaire V, Molesini A, Seidita V, editors. Handbook on AgentOriented Design Processes. Heidelberg: Springer, Berlin; 2014. p. 115–39.
14. Fortino G, Guerrieri A, Russo W, Savaglio C. Towards a development methodology for smart object-oriented IoT systems: a metamodel approach. In: IEEE International Conference on Systems, Man, and Cybernetics, Hong Kong, China, 2015, p. 1297–302. https://doi.org/10.1109/SMC.2015.231.
15. Giray G, Tekinerdogan B. Situational method engineering for constructing Internet of Things development methods. In: Business Modeling and Software Design—8th International Symposium, BMSD 2018, Proceedings, vol. 319. Springer. p. 221–239. https://doi.org/10.1007/978-3-319-94214-8_14.
16. IoT methodology—the Internet of Things project lifecycle guide for creative, technical and business people. http://www.iotmethodology.com/. Accessed in 02 Feb 2022.
17. Merzouk S, Elhadi S, Cherkaoui A, Marzak A, Sael N. Agile software development: comparative study. SSRN Electron J. 2018. https://doi.org/10.2139/ssrn.3186323.
18. Knaster R. SAFe 4.0 distilled: applying the scaled agile framework for lean software and systems engineering. Boston: Addison-Wesley; 2017.
19. Sahoo KS, Tiwary M, Luhach AK, Nayyar A, Choo KKR, Bilal M. Demand-Supply Based Economic Model for Resource

Provisioning in Industrial IoT Traffic. IEEE Internet of Things Journal. 2021;. https://doi.org/10.1109/JIOT.2021.3122255.
20. Merzouk S, Cherkaoui A, Marzak A, Nawal S. IoT methodologies: comparative study. Procedia Comput Sci. 2020;175:585–90. https://doi.org/10.1016/j.procs.2020.07.084 (**ISSN 1877-0509**).
21. International Society of Automation. New ISA/IEC 62443 standard specifies security capabilities for control system components. 2019. https://www.isa.org/intech-plus/2019/may/new-isa-iec-62443-standard-specifies-security-capa.
22. International Organization for Standardization. Information technology—security techniques—information security management systems—overview and vocabulary. Fifth edition, 2018-02. https://www.iso.org/standard/73906.html.
23. Stouffer K, Lightman S, Pillitteri V, Abrams M, Hahn A. Guide to industrial control systems (ICS) security. NIST Special Publication, vol. 800, no. 82 Revision 2, p. 1–247, 2015.
24. Slama D, Puhlmann F, Morrish J, Bhatnagar RM. Enterprise IoT: strategies and best practices for connected products and services. Sebastopol: O'Reilly Media, Inc.; 2016.
25. Object Management Group. Business process model and notation. https://www.bpmnquickguide.com/view-bpmn-quick-guide/.
26. Joint Task Force. Risk management framework for information systems and organizations: a system life cycle approach for security and privacy. NIST Special Publication, vol. 800, no. 37 Revision 2, p. 1–183, 2018. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-37r2.pdf.
27. Rainer R, Marshall T, Knapp K, Montgomery G. Do information security professionals and business managers view information security issues differently? Inf Syst Secur. 2007;16(2):100–8. https://doi.org/10.1080/10658980701260579.
28. Labrado C, Thapliyal H, Prowell S, Kuruganti T. Use of thermistor temperature sensors for cyber-physical system security. Sensors (Basel, Switzerland). 2019;19(18):3905. https://doi.org/10.3390/s19183905.
29. Rocio P, Diego-Mas J, Leon-Medina D. Measuring the project management complexity: the case of information technology projects. complexity, Hindawi, vol 2018, Article ID 6058480, 19 pages. https://doi.org/10.1155/2018/6058480.
30. Sophia Antipolis. European Parliament makes eCall mandatory from 2018. 7 May 2015.https://www.etsi.org/newsroom/news/960-2015-05-european-parliament-makes-ecall-mandatory-from-2018.
31. Economic and Social Council—United Nations, proposal for new regulation no. XXX on accident emergency call systems (AECS). 2017. https://unece.org/DAM/trans/doc/2017/wp29/ECE-TRANS-WP29-2017-132e.pdf.
32. Le VH, den Hartog J, Zannone N. Security and privacy for innovative automotive applications: a survey. Comput Commun. 2018;132:17–41. https://doi.org/10.1016/j.comcom.2018.09.010 (**ISSN 0140-3664**).
33. ŽabenskýRadomír J, Ščurek R, Jeremy Toh WH. Experimental verification of selected risk factors disrupting eCall system function. Trans VŠB Tech Univ Ostrava Saf Eng Ser. 2015. http://dx.doi.org/10.1515/tvsbses-2015-0003.