DISSERTATIONES
INFORMATICAE
UNIVERSITATIS
TARTUENSIS
49

ABASI-AMEFON OBOT AFFIA

A Framework and Teaching Approach for IoT Security Risk Management

DISSERTATIONES INFORMATICAE UNIVERSITATIS TARTUENSIS

49

DISSERTATIONES INFORMATICAE UNIVERSITATIS TARTUENSIS
49

ABASI-AMEFON OBOT AFFIA

A Framework and Teaching Approach for IoT Security Risk Management

UNIVERSITY OF TARTU
Press
1632

Institute of Computer Science, Faculty of Science and Technology, University of Tartu, Estonia.

Dissertation has been accepted for the commencement of the degree of Doctor of Philosophy (PhD) in Computer Science on 12 October, 2023 by the Council of the Institute of Computer Science, University of Tartu.

Supervisor

Prof. Dr. Raimundas Matulevičius
Institute of Computer Science, University of Tartu, Tartu, Estonia.

Dr. Alexander Nolte
Institute of Computer Science, University of Tartu, Tartu, Estonia.
Carnegie Mellon University, Pittsburgh PA, USA.

Opponents

Prof. Dr. Guttorm Sindre
Department of Computer Science,
Norwegian University of Science and Technology, Norway.

Prof. Dr. Jari Porras
School of Engineering Sciences,
Lappeenranta-Lahti University of Technology LUT, Finland.
Aalto University, Finland.
University of Huddersfield, United Kingdom.

The public defense will take place on 18 December, 2023 at 10:00 in Narva mnt 18 - 2049.

The publication of this dissertation was financed by the Institute of Computer Science, University of Tartu.

ISSN 2613-5906 (print) ISSN 2806-2345 (PDF)
ISBN 978-9916-27-399-9 (print) ISBN 978-9916-27-400-2 (PDF)

University of Tartu Press
http://www.tyk.ee/

To my family and friends

ABSTRACT

With the rise of network-capable devices and their vulnerabilities, attention has focused on security risk management (SRM) in the internet of things (IoT). The potential attack surface expands as the IoT ecosystem grows, underscoring the need for robust security measures against evolving threats. Yet, existing research reveals gaps in fully integrating the IoT architecture into SRM activities. Practical implementation challenges further complicate the intricate nature of IoT systems and the daunting task of SRM. Consequently, the disparity between theory and practice undermines the likelihood of successfully implementing security frameworks in real-world IoT systems, leaving them susceptible to an ever-evolving threat landscape.

To address these challenges, we introduce the IoT Architecture-based Security Risk Management (IoTA-SRM) framework, offering a comprehensive and integrative approach to SRM in IoT systems. By integrating the IoT architecture into security risk management, stakeholders can ensure comprehensive coverage of their IoT system components and interactions. The framework was evaluated involving expert-validated case studies in autonomous vehicle systems to ascertain its practicality, adaptability, and benefits. However, though the practicality of the framework is seen in this case analysis, its theoretical richness may remain abstract for practitioners during application. Given the intricacy of IoT systems and the dynamic nature of threats, traditional instructional approaches alone prove insufficient, to facilitate the framework's transition from theoretical understanding to practical application, necessitating educational methods that transcend conventional instruction. Thus, an intervention-based hackathon approach was developed. Enhanced with tailored interventions, this approach fosters an experiential learning approach. These interventions are carefully crafted, adaptable to diverse hackathon contexts and security learning content, and refined through action research cycles to enable participants to navigate and apply the IoTA-SRM framework.

Our integrated framework and hackathon approach served dual roles: an educational model targeting SRM in IoT systems and an evaluation mechanism for the IoTA-SRM framework through hackathon iterations, thus allowing for its iterative refinement. This thesis contributes to the theoretical understanding of IoT SRM and practical approaches for IoT security risk management. It further offers a method for educating future professionals in this rapidly evolving field. The findings and methodologies presented herein provide a basis for future endeavours in applying and teaching IoT SRM, exemplifying the synergistic relationship between theoretical security concepts and their practical application.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ORIGINAL PUBLICATIONS

Publications included in the thesis

1. **Abasi-amefon Obot Affia**, Raimundas Matulevičius, and Alexander Nolte. 2019. Security Risk Management in Cooperative Intelligent Transportation Systems: A Systematic Literature Review. In: Panetto, H., Debruyne, C., Hepp, M., Lewis, D., Ardagna, C., Meersman, R. (eds) *On the Move to Meaningful Internet Systems: OTM 2019 Conferences*. OTM 2019. Lecture Notes in Computer Science, vol 11877. Springer, Cham. https://doi.org/10.1007/978-3-030-33246-4_18
2. **Abasi-amefon Obot Affia** and Raimundas Matulevičius. 2021. Securing an MQTT-based Traffic Light Perception System for Autonomous Driving. *2021 IEEE International Conference on Cyber Security and Resilience (CSR)*, Rhodes, Greece, 2021, pp. 255-260. https://doi.org/10.1109/CSR51186.2021.9527989
3. **Abasi-amefon Obot Affia**, Raimundas Matulevičius, and Rando Tõnisson. 2021. Security Risk Estimation and Management in Autonomous Driving Vehicles. In: Nurcan, S., Korthaus, A. (eds) *Intelligent Information Systems*. CAiSE 2021. Lecture Notes in Business Information Processing, vol 424. Springer, Cham. https://doi.org/10.1007/978-3-030-79108-7_2
4. **Abasi-amefon Obot Affia**, Alexander Nolte, and Raimundas Matulevičius. 2020. Developing and Evaluating a Hackathon Approach to Foster Cyber Security Learning. In: Nolte, A., Alvarez, C., Hishiyama, R., Chounta, IA., Rodríguez-Triana, M., Inoue, T. (eds) *Collaboration Technologies and Social Computing*. CollabTech 2020. Lecture Notes in Computer Science, vol 12324. Springer, Cham. https://doi.org/10.1007/978-3-030-58157-2_1
5. **Abasi-amefon Obot Affia**, Alexander Nolte, and Raimundas Matulevičius. 2022. Integrating hackathons into an online cybersecurity course. *In Proceedings of the ACM/IEEE 44th International Conference on Software Engineering: Software Engineering Education and Training* (ICSE-SEET '22). Association for Computing Machinery, New York, NY, USA, 134–145. https://doi.org/10.1145/3510456.3514151.
6. **Abasi-amefon Obot Affia**, Alexander Nolte, and Raimundas Matulevičius. 2023. IoT Security Risk Management: A Framework and Teaching Approach, *Informatics in Education*(2023). https://doi.org/10.15388/infedu.2023.30

Other published work of the author

7. **Abasi-amefon Obot Affia** and Raimundas Matulevičius. 2022. Security Risk Management in Shared Mobility Integration. *In Proceedings of the 17th International Conference on Availability, Reliability and Security* (ARES '22). Association for Computing Machinery, New York, NY, USA, Article 145, 1–10. https://doi.org/10.1145/3538969.3543797
8. Malina, Lukas, Petr Dzurenda, Sara Ricci, Jan Hajny, Gautam Srivastava, Raimundas Matulevičius, **Abasi-Amefon O. Affia**, Maryline Laurent, Nazatul Haque Sultan, and Qiang Tang. 2021. Post-quantum era privacy protection for intelligent infrastructures. *IEEE Access* 9 (2021): 36038-36077. https://doi.org/10.1109/ACCESS.2021.3062201
   Author contributions: Performed security and privacy analysis, contributed to case study design and editing of the manuscript.
9. Alexander Nolte, Ei Pa Pa Pe-Than, **Abasi-amefon Obot Affia**, Chalalai Chaihirunkarn, Anna Filippova, Arun Kalyanasundaram, Maria Angelica Medina Angarita, Erik Trainer, and James D. Herbsleb. 2020. How to organize a hackathon–A planning kit. https://doi.org/10.48550/arXiv.2008.08025.
   Author contributions: Contributed research and study results of hackathons for learning.
10. **Abasi-amefon Obot Affia**, Raimundas Matulevičius, and Alexander Nolte. 2020. Security risk management in e-commerce systems: A threat-driven approach. *Baltic Journal of Modern Computing* 8.2 (2020): 213-240. https://doi.org/10.22364/bjmc.2020.8.2.02

LIST OF ABBREVIATIONS

AHP	Analytic hierarchy process
AV	Autonomous vehicle
AVPS	Autonomous vehicle parking system
BPMN	Business process model and notation
BSS	Bike sharing system
CAN	Controller area network
CBA	Cost-benefit analysis
CCTV	Closed-circuit television
CIA	Confidentiality, integrity, and availability
COBIT5	Control objectives for information and related technologies
CPS	Cyber-physical system
CWE	Common weakness enumeration
CTF	Capture the flag
CVA	Central validation authority
CVE	Common vulnerabilities and exposures
DSR	Design science research
DSRC	Dedicated short-range communications
DoS	Denial of service
ECU	Electronic control unit
FAIR	Factor analysis of information risk
FHSS	Frequency-hopping spread spectrum
GPS	Global positioning system
HTTPS	Hypertext transfer protocol secure
HD	High definition
IP	Internet protocol
IEEE	Institute of electrical and electronics engineers
IoT	Internet of things
IoTA-SRM	IoT architecture-based security risk management
IS	Information system
ISSRM	Information system security risk management
IT	Information technology
ITS	Intelligent infrastructure system
LiDAR	Light detection and ranging
LIN	Local interconnect network
LPWAN	Low-power wide-area network
LoWPAN	Low-power wireless personal area networks
LTE	Long-term evolution
LVDS	Low-voltage differential signalling
M2M	Machine-to-machine
MAC	Media access control

MAPE	Monitor, analyze, plan, execute
MOST	Media oriented systems transport
MQTT	Message queuing telemetry transport
NFC	Near-field communication
NIST	National institute of standards and technology
NVD	National vulnerability database
OCTAVE	Operationally critical threat, asset, and vulnerability evaluation
OS	Operating systems
OSI	Open systems interconnection
OT	Operational technology
OTA	Over-the-air
OWASP	Open web application security project
PBL	Project-based learning
PPS	Precision positioning system
PSK	Pre-shared key
PKI	Public key infrastructure
PVN	Plausibility validation network
RBT	Revised bloom's taxonomy
RFID	Radio frequency identification
ROSI	Return on security investment
RQ	Research question
SMS	Short messaging service
SOA	Service-oriented architecture
SRM	Security risk management
STEM	Science, technology, engineering, and mathematics
STRIDE	Spoofing, tampering, repudiation, information disclosure, denial of service, elevation of privilege
SysML	Systems modeling language
TARA	Threat assessment & remediation analysis
TCP	Transmission control protocol
TLS	Transport layer security
TPMS	Tire-pressure monitoring system
TRA	Harmonized threat and risk assessment
UML	Unified modeling language
USB	Universal serial bus
V2I	Vehicle-to-infrastructure
V2V	Vehicle-to-vehicle
VIPER	Vision integrated pseudo-range error removal
WAVE	Wireless access in vehicular environment
WiMAX	Worldwide interoperability for microwave access

1. INTRODUCTION

The Internet of Things (IoT) is a network of physical and virtual objects (devices, things, etc.) with a wide range of data and connectivity capabilities [1, 2]. These devices can collect and exchange data over the internet, forming a vast interconnected system. Often referred to as "smart" or "connected" devices, they enable communication between themselves and humans [3]. IoT devices can vary from small consumer gadgets like smartwatches and home appliances to large-scale industrial systems like smart industries and smart cities [3] consisting of a wide range of configurable objects that gather, process, store, distribute, and utilise data to achieve their objectives [1]. Thus, the core concept of IoT is to seamlessly connect and enable interaction between devices, leading to increased automation, data sharing, and efficiency in various industries and everyday activities [2].

Consequently, the global number of active IoT endpoints experienced an 18% growth in 2022, reaching 14.3 billion connections, with a forecasted 16% increase in 2023 and expansion expected to persist in the future [4]. In early 2023, approximately 54% of organisations were targeted by cyber attacks weekly, with an average of close to 60 attacks aimed at IoT devices per organisation each week [5]. The prevalence of these attacks as IoT devices grow exponentially underscores the crucial need for security and privacy measures to safeguard against cyber threats. Managing security risks in IoT systems requires a comprehensive understanding of the system's assets, vulnerabilities, and potential risks. Due to the variety of IoT components, each with unique design features and needs [6], there isn't a one-size-fits-all strategy for managing IoT security risks. Hence, our initial research objective addresses this gap by proposing an IoT security risk management framework. The framework employs the IoT reference architecture – which outlines the structure and mutual dependencies within IoT systems – as a crucial input to security risk management activities.

However, the alignment of theoretical cybersecurity knowledge with practical implementation for securing critical systems is a challenge that can hinder the practical application of proposed IoT security frameworks [7]. The complex nature of IoT systems and the evolving threat landscape necessitate educational approaches beyond traditional instruction. Thus, bridging this gap requires teaching strategies that encourage the participant's knowledge of the theoretical concepts of IoT security risk management and the expertise needed to secure IoT systems. By employing educational approaches that emphasise hands-on learning, real-world scenarios, and active engagement, participants can develop a deep understanding of IoT security principles and the ability to apply them in practical settings. Effective teaching strategies can also incorporate assessment and feedback mechanisms, foster interaction and knowledge exchange among participants and instructors, and encourage a deeper understanding of IoT security risk management strategies. We thus propose an approach to foster learning and ap-

plication of security concepts through hackathons[1]. Hackathons involve different stakeholders and define activities and goals that allow short-term collaboration to apply knowledge in tackling specific problems [8], providing benefits of learning, knowledge sharing and skill development [8, 9]. Furthermore, hackathons create an immersive learning environment where participants work intensively in teams to solve security challenges within a limited timeframe [9]. This approach enhances participants' technical skills and cultivates their ability to think creatively, adapt and collaborate in dynamic situations [10].

Combining the IoT security risk framework with the hackathon approach can address the gap between theoretical knowledge and practical implementation. This integrated approach allows participants to gain practical experience, deepen their understanding of IoT security risks, and contribute to implementing security risk management measures for IoT systems.

1.1. Problem Statement

The diverse nature of IoT components, each with unique design specifications and system requirements, poses a significant challenge for implementing a one-size-fits-all approach to IoT security risk management. Unlike traditional IT systems, IoT systems consist of interconnected devices with varying processing power, memory, and communication capabilities, making monitoring and security management complex [11]. Furthermore, the security risks associated with IoT systems extend beyond devices and affect people, processes, objects, and data [12, 13, 14]. The IoT architecture provides insight into the interdependencies among components and subsystems within the IoT system [15, 6]. Thus, understanding the IoT architecture and leveraging architectural knowledge can enhance IoT security risk identification and management. While some studies have focused on individual security risks in IoT systems [16, 17, 18], there is a lack of research exploring the benefits of an architectural perspective on security risk management. Additionally, studies that have explored security risk management in the IoT architecture [19, 20] expose research gaps in the disproportionate concentration of security research on IoT architecture layers to the detriment of others. These issues may not have been discovered without considering the architectural perspective.

However, the factors above, combined with the limited practical implementation methods for security risk management, create additional complexities in IoT security risk management for stakeholders [21]. Without proper guidance and effective teaching strategies, learners may struggle to navigate this complexity in implementing security practices [7]. Moreover, the gap between theoretical knowledge and practical implementation is common [7]. Learners may ac-

[1] In this thesis, the term "hackathon" is distinct from "hacking" as it represents legitimate events promoting creative problem-solving and innovation within an ethical framework. Hacking, on the other hand, involves unauthorised and malicious activities, which is not the focus of this work.

quire theoretical knowledge about security concepts and frameworks, but applying this knowledge in real-world scenarios requires hands-on experience and practical skills. Learners may struggle translating theoretical knowledge into practical security implementations without proper guidance and practical learning opportunities. While teaching strategies provide opportunities for learning by doing, learning benefits cannot materialise without careful planning to create a suitable learning environment.

The following research objectives are formulated to address the problems mentioned above:

- *Research objective 1*: To develop a framework for IoT SRM leveraging the IoT architecture as input to analyse assets, security risks, and controls. The framework should address the challenges posed by the diverse nature of IoT systems.
- *Research objective 2*: To bridge the disconnect between theoretical knowledge and practical implementation of security risk management concepts using hackathons. This will be achieved through carefully organised hackathons to serve as an approach to promote knowledge about managing security risks in IoT systems.
- *Research objective 3*: The final objective is to present a cohesive approach to teaching the developed framework for IoT SRM. This combination is intended to provide a well-rounded method for IoT security risk management education.

1.2. Research Questions

We address the identified research problems and achieve the outlined research objectives by answering the main research question:

"*How can an integrated framework and teaching approach for IoT security risk management be realised?*"

The following sub-questions are proffered to produce the expected outcomes:

RQ$_1$. *How to manage security risks in IoT system architectures?*

Answering **RQ**$_1$ involves:

- Identification of the IoT architectural layers of IoT systems and the associated security risks to IoT architecture layers
- Development of the framework for IoT SRM, providing a step-by-step process for identifying, assessing, and treating security risks
- Validation of framework through IoT cases and expert validation

RQ$_2$. *How to use a hackathon-based methodology to teach security risk management in IoT system architectures?*

Answering **RQ**$_2$ involves:

- Exploration of a hackathon-based approach for IoT security education

- Identification and adaptation of hackathon interventions and format to foster understanding and application of security risk management principles.
- Iterative evaluation and refinement of hackathon interventions to support IoT security risk management education, including the framework developed in $\mathbf{RQ}_1$
- Formalisation of the framework and teaching approach integrating the framework for IoT SRM and the refined hackathon approach to teach about IoT security risk management

Answers to our research questions contribute to IoT SRM and cybersecurity education, setting the groundwork for future research and providing practical guidance for educators and practitioners. The hackathon-based teaching strategy promotes security risk management capabilities and awareness among IoT stakeholders and users.

1.3. Research Methodology

Design science principles form the foundation of our research methodology [22, 23, 24]. This approach is deemed appropriate as the objective is to create a framework for managing security risks in IoT architectures and a corresponding teaching approach employing hackathons. The DSR method is adhered to, commencing with problem identification, encompassing a comprehensive literature review and background study to address real-world problems in IoT security risk management. Following this, the design cycle unfolds in four phases: solution objective, development methods, artefact, and evaluation methods. The solution objective targets the development of the framework for IoT SRM and a hackathon approach to teach the framework and provide practical implementation opportunities. Development methods involve literature review, background study, and conceptual analysis to develop our artefacts. Artefacts generated are the framework for IoT SRM and the hackathon teaching approach. The framework is evaluated through case analysis and expert validation. The hackathon teaching approach undergoes iterative evaluation through action research cycles, supported by observation, questionnaires, interviews and content analysis, to gauge its effectiveness in fostering learning benefits. The research culminates in the communication of these artefacts through our contribution publications. The DSR methodology offers flexibility, allowing us to integrate multiple research methods [23, 25] such as literature review, background study, case analysis, and action research, thus enriching our research objectives.

1.3.1. Literature Review and Background Study

Literature review is invaluable for DSR, particularly for creating practical solutions or artefacts [26]. It helps discern key concepts in emerging research areas, thereby setting a solid theoretical foundation for problem identification and solu-

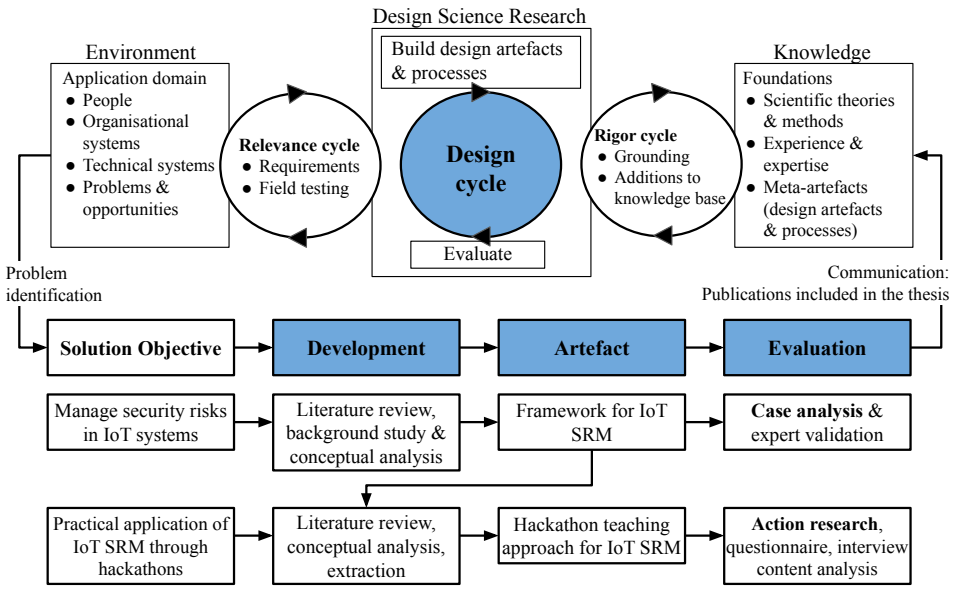


Figure 1. Research methodology: DSR cycle and overall research activities based on [22]

tion design [26, 27, 28]. Existing methods related to security risk management in IoT systems are examined as a starting point. This equips us with a foundational theoretical understanding required for applying established methodologies in IoT security risk management ($\mathbf{RQ}_1$) and provides an avenue for innovative work [29, 27, 28]. Building upon this, our background study focuses on real-world contexts, offering a nuanced understanding beyond the insights gathered from the literature review. Specifically, the architecture and layered model for IoT security risk management are delved into, with security threats identified across various layers ($\mathbf{RQ}_1$). Additionally, the literature review and background study collectively examine instructional strategies, zeroing in on the role of hackathons in enhancing security learning, and provide considerations for hackathon design choices that can optimise the learning potential during such events ($\mathbf{RQ}_2$).

1.3.2. Case Analysis

A case study method provides a focused and thorough investigation of specific situations, entities, or phenomena in real-world contexts [30]. This methodology can be applied for exploratory, descriptive, or explanatory research objectives [31]. It allows us to align our study with our research questions and is consistent with DSR [32, 33, 27]. In qualitative research, case studies investigate complex issues within their natural environments. They emphasise understanding specific characteristics and factors that shape the subject of investigation [30]. A significant criticism of the case study method is its generalisability [30, 34]. However, including multiple data sources like interviews, observations, and content analysis

can enhance credibility and rigor [30, 34]. Furthermore, use cases, which describe specific instances within a broader system, can be integrated within case studies for a more tangible understanding [35].

For this thesis, an explanatory case analysis is applied to assess the framework developed for IoT SRM ($\mathbf{RQ}_1$) within the IoT context and to examine the lessons learned from the framework's application within the intelligent transportation systems (ITS) facet of the IoT ecosystem. A background study to provide context to cases used in this thesis is detailed in Section 2.1.3. Additionally, example use cases representing specific scenarios or applications of IoT, are documented in Appendix A and Appendix B.

1.3.3. Action Research

Our research integrates action research (AR) with the design science research (DSR) methodology, employing a cycle that involves planning, action, observation, and reflection [36, 37, 38, 39, 40]. This iterative process is depicted in Figure 2.

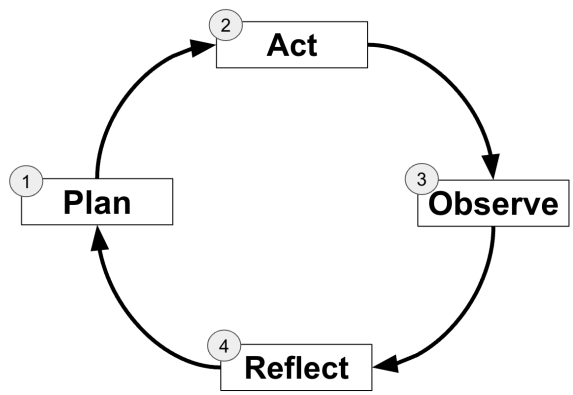


Figure 2. Action research process [39]

The AR method supports the hackathon teaching approach artefact evaluation ($\mathbf{RQ}_2$), adapting our hackathon approach to address real-world problems and improving its practical relevance. The harmonisation of DSR and AR is central to our research method, with iterative cycles unified to bridge theory and practice [41, 42]. While DSR focuses on crafting solutions to tangible problems, AR emphasises learning and enhancement through social interventions [41, 43]. This iterative approach to planning, action, observation, and reflection serves the purpose of refining the hackathon approach, providing substantial insights for improvement beyond techniques like pre- and post-test experiments [44, 45]. Thus, to achieve the second research objective of bridging the gap between theoretical knowledge and practical implementation of security risk management concepts, the hackathon approach is utilised and evaluated through AR in three cycles of planning, action, observation, and reflection.

During these cycles, various research instruments are incorporated into our AR method to evaluate the research objective, aligning with existing works and enhancing the consistency of outcomes [46, 47]. A combination of research instruments, including observation, questionnaires, interviews, and content analysis, is employed within the AR cycles to assess the hackathon approach's effectiveness in fostering security learning and enhancing participants' experiences [48, 49]. Observation serves as a valuable tool for understanding participants' behaviour and is a crucial part of action research that contributes to educational improvements [50, 51, 48, 52]. Questionnaires, particularly those that are carefully designed, offer a broad understanding of participants' experiences and perceptions, including their views on the hackathon approach and learning outcomes [53, 49, 48]. Using closed-ended and open-ended questions facilitates a comprehensive understanding of participants' experiences [48]. Interviews provide deeper insights into participants' learning experiences and challenges and are particularly useful when other methods lack depth [54, 48]. Content analysis is utilised systematically to assess textual or visual data, aiding in identifying patterns, themes, and insights [55, 49]. Questionnaire instruments and hackathon outcome artefacts undergo content analysis within action research cycles to evaluate participants' achievement of expected learning outcomes. The content analysis outcomes also validate both the hackathon approach and the framework within which learning takes place.

For the analysis of participant learning outcomes, the revised Bloom's Taxonomy (RBT) [56], as discussed in Section 1.3.4, is incorporated into content analysis research to enhance the depth and structure of the analysis. Specifically, content analysis is applied to hackathon-generated outcomes compiled into a report. This method provides insights into hackathon participant learning outcomes and the effectiveness of the IoTA-SRM framework and hackathon approach for IoT security risk management.

1.3.4. Revised Blooms Taxonomy (RBT)

Learning outcomes encompass the knowledge, understanding, and skills students are expected to demonstrate or have acquired at the end of a learning period [57, 58, 59]. Learning outcomes can define behaviours within three domains: cognitive, affective, and psychomotor [57]. The cognitive domain encompasses thought processes like understanding, analyzing, and evaluating. The affective domain includes attitudes, feelings, and values, such as appreciation and acceptance. Lastly, the psychomotor domain involves physical skills, including performing, assembling, and dismantling. The cognitive domain is emphasised for evaluating participant achievement regarding the expected learning outcomes derived from applying knowledge provided during our hackathon events. A high level of cognitive engagement is demanded by IoT SRM activities and those outlined in our framework to be applied.

To assess the achievement of learning outcomes, the revised Bloom's taxonomy (RBT) is adopted for our cognitive analysis. Within RBT, cognitive domain levels are categorised into six tiers, as depicted in Figure 3: Remember, Understand, Apply, Analyse, Evaluate, and Create [60, 56]. The definitions of each

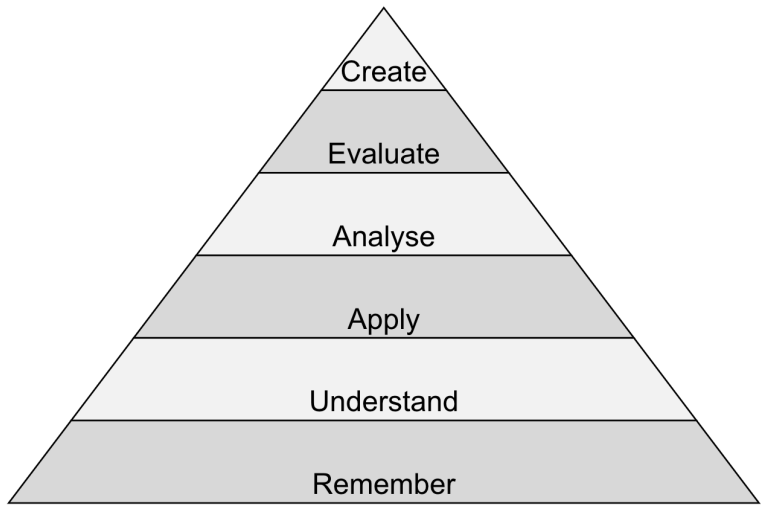


Figure 3. Revised blooms taxonomy (RBT) [56]

RBT level in Table 2 are discussed based on [61]. The three highest tiers (analyse, evaluate, and create) are traditionally linked with higher-order cognitive skills. In comparison, the initial three tiers (remember, understand, and apply) are engaged in lower-order cognitive skills [62, 61]. Given the intricacy and depth of SRM and the derived framework, participants need to engage in higher-order cognitive skills to attain the expected learning outcomes. This proficiency can be evaluated using RBT.

Research has explored evaluating higher-order and lower-order cognitive skills using RBT, previously applied in analogous studies. This taxonomy is widely used to gauge the cognitive level of assessment tasks [61, 63, 62]. Having been Utilised in similar contexts, RBT helps discern the cognitive depth of assessment tasks and verify if participants have reached the requisite thinking level. While RBT is a valuable instrument for evaluating cognitive learning results, its capability to measure non-cognitive outcomes, often referred to as 21st-century or employability skills, can be constrained [64]. Due to the complexity of SRM and the objectives of our research, a deliberate focus was placed on the cognitive domain when evaluating the outcomes of hackathons. While these non-cognitive aspects were not extensively investigated due to our emphasis on cognitive analysis, the importance of non-cognitive skills, especially in hackathons and higher education, is acknowledged.

Table 2. Bloom's taxonomy levels and definitions based on [61]

Level	Definition
Remember	Retrieve relevant knowledge from long-term memory • Recognise: Retrieve relevant knowledge from long-term memory by identifying information consistent with the presented material • Recalling: Retrieve relevant knowledge from long-term memory when prompted **Action verbs**: Choose, define, find, how, label, list, match, name, omit, recall, relate, select, show, spell, tell, what, when, where, which, who, why.
Understand	Construct meaning from instructional messages, including oral, written, and graphic communication • Interpreting: Construct meaning from instructional messages • Exemplifying: Transform information from one form of presentation to another • Classifying: Identify examples or instances of a concept or determine categorisation • Summarizing: Extract a general theme or major point • Inferring: Draw logical conclusions from the provided information • Comparing: Recognise correspondences between two ideas, objects, etc • Explaining: Formulate cause-and-effect models **Action verbs**: Classify, compare, contrast, demonstrate, explain, extend, illustrate, infer, interpret, outline, relate, rephrase, show, summarise, translate.
Apply	Carry out or use a procedure in a given situation • Executing: Execute or employ a procedure in a given situation • Implementing: Utilise a procedure to accomplish a task **Action verbs**: Apply, build, choose, construct, develop, experiment with, identify, interview, make use of, model, organise, plan, select, solve, utilise.
Analyse	Break material into its constituent parts and determine how the parts relate to one another and to an overall structure or purpose • Differentiating: Divide the material into components and ascertain their relationships • Organizing: Differentiate pertinent and significant components of the material • Attributing: Determine how elements fit within a structure **Action verbs**: Analyse, assume, categorise, classify, compare, conclusion, contrast, discover, dissect, distinguish, divide, examine, function, inference, inspect, list, motive, relationships, simplify, survey, take part in, test for, theme.
Evaluate	Make judgments based on criteria and standards • Checking: Formulate judgments based on criteria and standards • Critiquing: Detect inconsistencies or fallacies within a process or product **Action verbs**: Agree, appraise, assess, award, choose, compare, conclude, criteria, criticise, decide, deduct, defend, determine, disprove, estimate, evaluate, explain, importance, influence, interpret, judge, justify, mark, measure, opinion, perceive, prioritise, prove, rate, recommend, rule on, select, support, value.
Create	Put elements together to form a coherent or functional whole; reorganise elements into a new pattern or structure • Generating: Assemble elements to construct a coherent whole or restructure into a new arrangement • Planning: Propose alternative hypotheses and criteria • Producing: Devise methodologies or originate products **Action verbs**: Adapt, build, change, choose, combine, compile, compose, construct, create, delete, design, develop, discuss, elaborate, estimate, formulate, happen, imagine, improve, invent, make up, maximise, minimise, modify, original, originate, plan, predict, purpose, solution, solve, suppose, test, theory.

1.4. Contributions

The thesis provides three incremental contributions, answering our research questions ($\mathbf{RQ}_1$, $\mathbf{RQ}_2$). These contributions culminate in a comprehensive learning

experience that balances theoretical understanding and practical application in security risk management (SRM). Implementing the IoTA-SRM framework in such a context, guided by hackathon interventions, corroborates its applicability and practicality, shedding light on IoT security risk management education and providing actionable recommendations for educators and practitioners alike in IoT security risk management. These contributions are discussed below:

Contribution 1. The thesis presents an IoT Architecture-based Security Risk Management (IoTA-SRM) framework developed and evaluated in three main publications [65, 66, 67] described in Table 3. This framework integrates the IoT ar-

Table 3. Author publications relevant to contribution 1

Publication	Relevance to contribution
[65]	This publication underscores applying the IoTA-SRM framework to the autonomous vehicles (AVs) case, detailing results on testing and refining the IoTA-SRM framework within a practical context. The results and the lessons learned from framework validation guide AV engineers and security analysts. The publication thus substantiates the first contribution of this thesis in exploring the development and validation of the framework for IoT security risk management.
[66]	This publication presents a unique application of the IoTA-SRM framework, showcasing its use securing in a pilot feature – MQTT-based traffic light perception system for autonomous vehicles. This publication underscores the practical utility of the IoTA-SRM framework in analyzing threats, assessing assets and risks, and making risk treatment decisions during the design phase of the new feature.
[67]	This publication formalises the IoTA-SRM framework, underscoring its key contribution to IoT security risk management. It informs on the framework's conceptual model, process and process tasks for managing security risks in IoT systems.

chitecture, enabling a systematic, comprehensive process for assessing and managing security risks in IoT systems. The contribution addresses a gap in prior research that does not incorporate the IoT architecture within SRM activities. The practicality and benefits of the IoTA-SRM framework are further substantiated through case analysis and validation by industry stakeholders, thus answering $\mathbf{RQ}_1$.

Contribution 2. The second contribution presents an intervention-based hackathon approach developed in three main publications [68, 69, 67] and described in Table 4. This approach complements existing hackathon strategies but develops and evaluates specific hackathon interventions to proffer cybersecurity learning benefits. These interventions are adapted to the learning context and systematically evaluated through action research cycles to ensure they maximise the learning potential of hackathon participants, thereby addressing $\mathbf{RQ}_2$. Thus, we can introduce the IoTA-SRM framework ($\mathbf{RQ}_1$) within a hackathon context, where participants

apply the theoretical concepts provided by the framework in a practical setting, thereby enhancing the practicability of the IoTA-SRM framework and equipping learners to address real-world IoT SRM challenges.

Table 4. Author publications relevant to contribution 2

Publication	Relevance to contribution
[68]	This publication focuses on the first action research cycle, developing, adapting and introducing the hackathon interventions within a single hackathon event aimed at fostering cybersecurity learning. The outcome underscores the application of hackathon interventions within a conventional hackathon context.
[69]	This publication delves into the second action research cycle, modifying and implementing the hackathon interventions within an educational setting to promote security risk management learning. The interventions implemented in this context draw insights from the previous cycle [68]. The outcomes demonstrate the effectiveness of hackathon interventions in fostering a practical understanding of security risk management.
[67]	This publication includes the third action research cycle, wherein the hackathon interventions, enriched by the lessons learned from [69], are adapted and implemented to enhance IoT security risk management learning. The outcomes of this cycle affirm the potential of the approach to amplify the practical application of the IoTA-SRM framework and validate both the framework and the intervention-based hackathon approach.

Contribution 3. The third contribution amalgamates the IoTA-SRM framework with the hackathon approach developed in the publication [67] described in Table 5. This amalgamation ensures a comprehensive learning experience that balances theoretical understanding and practical application in security risk management (SRM) by introducing the IoTA-SRM framework ($\mathbf{RQ}_1$) within a hackathon context ($\mathbf{RQ}_2$), the approach allows participants to apply the theoretical concepts in a practical setting, thereby enhancing the practicability of the IoTA-SRM framework and equipping learners to address real-world IoT SRM challenges. Implementing the IoTA-SRM framework in such a context, guided by hackathon interventions, corroborates its applicability and practicality, shedding light on IoT security risk management education and providing actionable recommendations for educators and practitioners alike in IoT security risk management.

Table 5. Author publications relevant to contribution 3

Publication	Relevance to contribution
[67]	This publication substantiates the thesis's contribution by presenting the integration of the framework and the hackathon approach. The publication highlights the potential of this integration to reinforce participants' practical application of the framework, serve as an evidence-based endorsement of both the IoTA-SRM framework and the hackathon approach, and facilitate a practical and repeatable approach to managing IoT security risks in real-world scenarios.

1.5. Thesis Roadmap

This thesis is organised into six (6) chapters. The current chapter outlines the introduction, problem statement, research questions, research methodology and contributions. Chapter 2 introduces IoT systems from an architecture perspective, security risk management, and an approach for security education using hackathons. Chapter 3 discusses the IoT Architecture-based Security Risk Management (IoTA-SRM) framework as the first thesis contribution. Chapter 4 focuses on the intervention-based hackathon approach, where hackathon interventions for security learning are designed and evaluated. This is the second thesis contribution. Chapter 5 presents the integration of the framework and hackathon approach, guiding the practical application of the framework. This chapter is the final contribution of the thesis. Chapter 6 concludes the thesis and highlights future research directions resulting from the thesis contributions.

2. BACKGROUND

This chapter delves into the importance of IoT architecture in security risk management (SRM) and the need for teaching strategies that foster learning and implementation of IoT SRM practices. The purpose is to provide an overview of these major topics relevant to achieving our research objectives and answering our research questions. Section 2.1 presents a description of the IoT system, its architecture, and IoT system cases used in this thesis. In Section 2.2, various SRM methods for IoT SRM are explored, along with an investigation into security threats that span the architecture layers of the IoT use-case. Additionally, this chapter introduces teaching strategies to foster continuous learning in Section 2.3, highlighting the suitability of hackathons for security learning and their potential benefits.

2.1. Internet of Things (IoT) Systems

The Internet of Things (IoT) has significantly influenced technological interaction and changed multiple sectors such as healthcare, manufacturing, and transportation [70]. IoT merges information technology (IT), including cloud technology [71] with operational technology (OT), to form a unified system [72]. The IoT architecture articulates the fundamental principles and properties governing how constituent elements interact within a designated environment. The IoT architecture comprises three IoT concepts: the IoT environment, which comprises all components, systems, and relevant infrastructure; the IoT system, providing value to stakeholders; and the IoT components that collaborate to constitute the IoT system [72]. Figure 4 depicts an IoT environment that encompasses various IoT systems and components. Here, multiple IoT components are situated within this environment with four distinct IoT systems, each constituted by the interaction of at least two IoT components. Two IoT components exist in two different IoT systems. Furthermore, one of the depicted IoT systems also serves as an IoT component within a larger, encompassing system. Each IoT component in this environment possesses capabilities such as sensing or actuating.

IoT Components. IoT components serve as the foundational elements of any IoT system. These components cooperate to fulfil one or more objectives. Each component contributes a specific function necessary for the system's operation. Regarding network interfaces, all IoT components possess at least one, allowing for participation in many-to-many networks [73]. However, a component may not need to interact with more than one other component in a particular system. Additionally, numerous components feature an application interface that facilitates application-level interactions. Data flows among these components can be bidirectional, but roles may differ; some components might only send data, while others may only receive. Components can have capabilities such as sensing or actuating occupy a unique position—they act as the interface between the physical

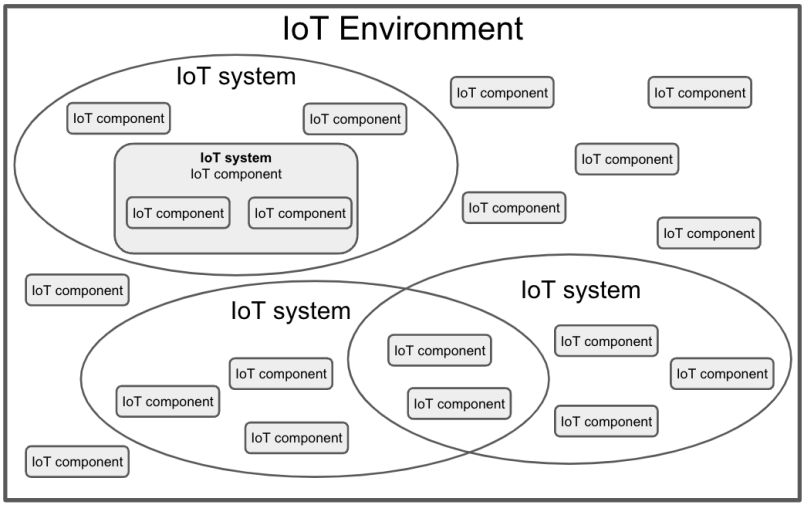


Figure 4. IoT environment and the relationships between IoT components and IoT systems [72]

and digital domains. Such components are commonly termed "IoT devices" [74].

IoT Systems. According to ISO/IEC 15288:2015 [75], a system assembles interrelated elements to achieve one or more specified goals. An IoT system differentiates itself from conventional IT systems by its ability to directly engage with the physical world through sensors and/or actuators embedded in its components. IoT systems automate information processing and task execution from distributed sources, allowing its components to exchange data and influence the physical environment [1, 65]. An individual IoT component with the requisite capabilities can belong to multiple systems simultaneously. Moreover, an IoT system can act as an IoT component within a larger system. People also play significant roles in the functioning of an IoT system, either as end-users, active participants, or entities of interest that the system observes or influences [76].

IoT Environments. An IoT environment comprises IoT components, networks, and ancillary services [72]. The IoT environment is characterised by a high degree of heterogeneity, encompassing IoT components that may have self-configuration capabilities and that operate with limited resources in dynamic conditions [77]. This heterogeneity enables an IoT system to manage a mix of physical and logical entities, facilitating diverse forms of interaction.

Given the diverse nature of IoT systems, as per our understanding, no current approach to managing IoT security risks can adequately accommodate various use cases. The IoT reference architecture highlighted in Figure 5 addresses this gap, playing a crucial role in standardising our comprehension of IoT systems. The IoT reference architecture thus serves as a guiding framework for the design of the IoT system architecture, describing the fundamental components of the IoT system, typically using a layered model [6]. Going beyond the reference architecture,

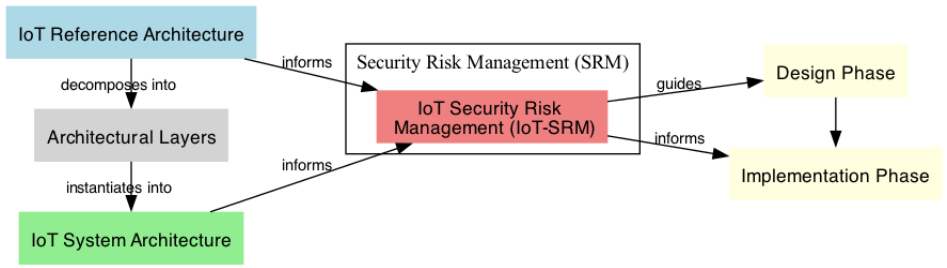


Figure 5. Relationship between reference architecture, system architectures and system implementation

the IoT system architecture furnishes a blueprint for designing and deploying IoT solutions. It extends the reference architecture by presenting specific real-world instantiations of these components and elucidates their interrelationships within the implemented IoT system [78]. Crucially, the IoT reference architecture is vital for identifying potential security risks at each architecture layer, an imperative element for effective risk management. This understanding informs subsequent security risk analyses during the development of the concrete system architecture, leading to an enhanced design that prioritizes security in implementation. Adopting an architectural perspective in SRM enables a tailored risk assessment, aligning with the specific characteristics of the IoT system. This approach fosters security-conscious design practices and facilitates the seamless integration of security protocols throughout the phases of design and implementation [67].

This section explores the IoT reference architectures and introduces a use case of an IoT-based intelligent transportation system. The security landscape for IoT systems has grown more intricate in recent years [79]. This complexity arises as many devices and sensors within the network capture and relay data, thereby becoming susceptible to multiple security vulnerabilities [65]. Identifying these vulnerabilities, potential threats, and resulting risks is critical for ensuring system integrity. As the discussion advances towards SRM in the IoT context, emphasis will be placed on the system's architecture as a fundamental input for security risk assessment activities. This progression allows for more detailed analysis and increased applicability to implementing the IoT system.

2.1.1. IoT Reference Architecture

The perspective of IoT reference architecture highlights the convergence of information and communication technologies in IoT systems. It encompasses how software and hardware components work together to gather, process, store, distribute, and utilise information from various sources to achieve specific objectives that align with their design goals [65, 15]. Several reference architectures have been proposed to address varying application requirements, network topology, protocols, and business models [15]. This includes the three-layer [80], four-layer service-oriented [81], middleware-based IoT or five-layer [15], and Cisco's seven-

layer [82] architecture illustrated in Figure 6. Architectures beyond the three-layered architecture seek to cover, in more detail, aspects of integrating wider technology and application areas in a service-oriented world [6]; however, these architectures are built upon the three-layer model.

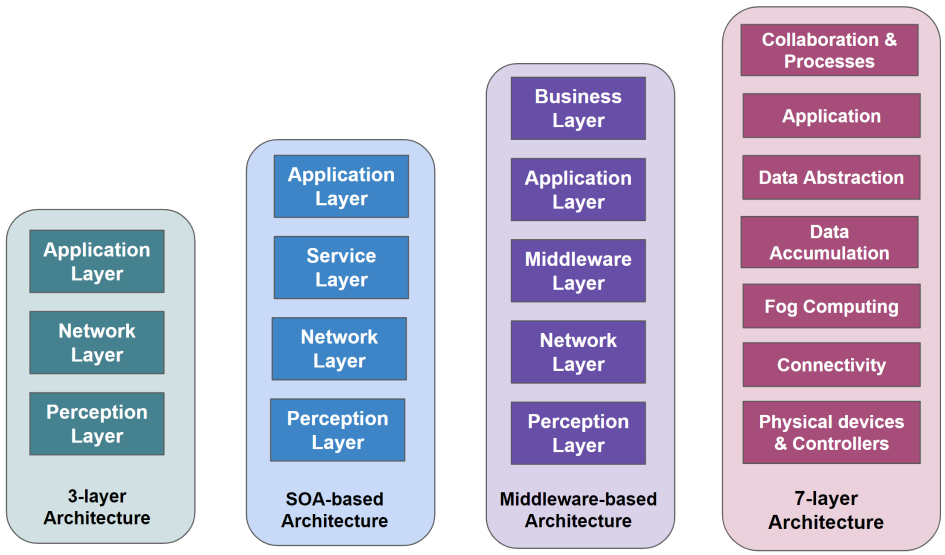


Figure 6. Common IoT reference architectures [80, 15, 6]

Three-layer architecture. The three-layer architecture [80] is a widely accepted standard that defines the structure of an IoT system, consisting of the perception, network, and application layers [80]. The perception layer collects data from sensors and devices, while the network layer handles data transmission and communication. The application layer is responsible for data processing, analysis, and decision-making. This architecture provides a simple and scalable framework for designing, deploying, and managing IoT systems.

Four-layer service-oriented architecture (SOA). The four-layer service-oriented architecture (SOA) [81] extends the three-layer architecture by adding a service layer, which provides a set of standard interfaces for devices and applications to communicate with each other [81, 15]. The four layers are perception (physical), network, service, and application, where the service layer provides data storage, transformation, and security services [15].

Middleware-based IoT or five-layer architecture. The middleware-based IoT architecture [15] extends the three-layer architecture by adding a middleware layer that provides security, data management, and protocol translation services and a business layer responsible for implementing the system's business logic (applications and services that interact with the data collected) [15].

Cisco's seven-layer IoT architecture. Cisco's seven-layer IoT architecture [82] builds on the OSI model to achieve a more detailed and comprehensive architec-

ture for IoT systems than the three-layer one. The seven layers consist of physical devices and controllers, connectivity, edge (fog), computing, data accumulation, data abstraction, application, and collaboration and processes layers [83, 84]. The first three layers of Cisco's architecture (physical devices and controllers, connectivity, and edge computing) are similar to the perception and network layers of the three-layer architecture. Cisco's architecture then expands on the application layer of the three-layer architecture with additional layers that enable more advanced data processing, storage, and application development [83].

2.1.2. IoT Three-layer Architecture Perspective

After the review of IoT reference architecture layers, the three-layer model depicted in Figure 7 is chosen to serve as the theoretical foundation of our SRM approach. The three-layer architecture is a widely recognised standard that clearly

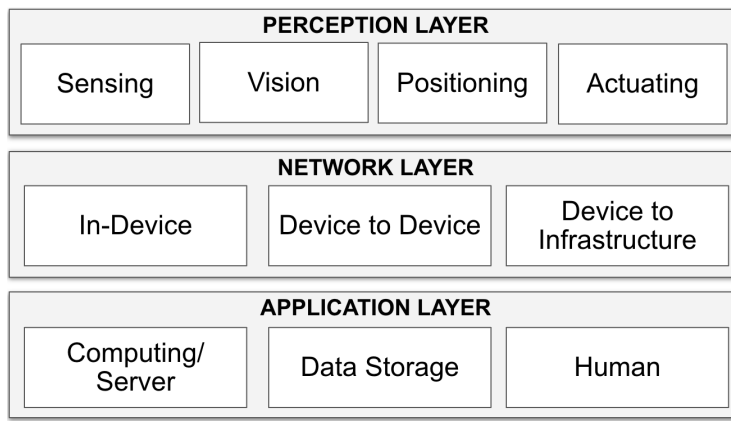


Figure 7. IoT Architecture layers and their components, adapted from [19]

outlines the foundational structure of an IoT system – perception, network, and application layers – each serving distinct roles [80]. The architecture also allows for scalability, making it suitable for IoT systems of varying sizes and complexities to be represented depending on an IoT system's specific needs and requirements. Lastly, while there exist more complex architectures, such as the four-layer service-oriented architecture (SOA), the middleware-based IoT or five-layer architecture, and Cisco's seven-layer architecture [15, 82, 83, 84], they rely on the fundamental structure provided by the three-layer architecture. Thus, there would be relative ease in adapting from a three-layer to more complex architectures as the common foundation between these architectures facilitates the adaptation process. However, it's crucial to remember that this process would require additional considerations of the specific services offered by more complex architectures and their associated security implications.

Due to its wide acceptance, scalability, and simplicity, the three-layer architecture is a great base for developing our IoT SRM framework. Its simplicity allows

straightforward asset identification, functional decomposition, and model-based risk management.

Perception Layer. The perception layer of an IoT system comprises objects/ devices with capabilities for collecting information about the environment, typically without human interaction. This includes sensing, visioning and identification, actuating, and positioning activities [85]. By interacting with the environment, these devices gain intelligence about the environment, such as temperature, humidity, pressure, and light level changes. Sensors (including visioning and positioning sensors) can convert these measurements into digital signals that the IoT system can process while actuators control IoT objects in response to sensor data.

Network Layer. The network layer in an IoT system facilitates the transmission and processing of information within IoT perception devices, between networked devices, and between network devices and network infrastructure. Additionally, the network layer comprises the communication infrastructure and supporting protocols that interact with devices [65]. Various protocols are used in the IoT network layer depending on the use case including CoAP, MQTT, TCP/UDP, IPv4/IPv6, 6LoWPAN, IEEE 802.15.4 (ZigBee, etc.), IEEE 802.15.1 (Bluetooth), LPWAN, RFID, NFC, IEEE 802.11 (WiFi), IEEE 802.3 (Ethernet), and Z-Wave [6, 15]. Each protocol has advantages and disadvantages, and its suitability for specific IoT systems must be evaluated.

Application Layer. The main feature of this layer is to deliver application-specific services to the end-user based on the application type, set business and profit models, and information provided from perception objects. Depending on the use case, this layer can include various computing and data storage service-oriented technologies (e.g., cloud computing, cloud storage, data filtering and aggregation, integration to other applications, etc.), to perform activities required by the end-user [79, 15]. The services provided can be related to business models, profit models, or specific user needs.

2.1.3. Intelligent Transportation Systems

In this research, specific cases from intelligent transportation systems (ITS) are used as practical scenarios to validate our proposed framework and provide hackathon participants with real-world applications. ITS refers to the application of technology and data analysis to increase the efficiency and safety of transportation systems [86]. Based on the IoT paradigm, ITS harnesses the capabilities of IoT, employing cooperative sensing and networking to manage the transportation of people and goods via various means, including road, air, rail, and water [87]. IoT-based ITS rely heavily on real-time data collection from various sources to analyse traffic patterns and optimise transportation systems [88]. However, the security of these systems is of utmost importance, given the potential risks to essential business assets and potential threats to human lives [89, 90]. The complexity of the security threats, which involve transportation, IoT, and distributed

systems, requires comprehensive SRM [89]. This approach ensures the confidentiality, integrity, and availability of the collected data, thereby enabling secure and efficient operation of various transportation functions such as speed management, navigation, and traffic management [91, 92, 88].

This thesis focuses on two rapidly growing sectors within IoT-based ITS: autonomous vehicles (AVs) and micro-mobility solutions. These cases were chosen due to their significance in the IoT sector and their ability to exemplify a range of security considerations. The subsequent sections explore these use cases, providing background information and illustrating how they validate our proposed framework.

Autonomous Vehicles. Autonomous vehicles are intricate information systems that gather, interpret, process and distribute data between vehicular and infrastructure systems, allowing them to function independently and provide mobility, safety, and comfort services [93]. AVs are fitted with advanced sensors that enable them to perceive their surroundings and control their movements with or without human intervention [93]. This thesis includes the study of highly automated components of automotive vehicles, achieving Level 3, 4 and 5 vehicular automation as defined by the SAE J3016 taxonomy [94]. The architecture of autonomous vehicles can be similarly decomposed into perception, network, and application layers [65]. The perception layer employs sensors and cameras to amass environmental data such as road conditions and pedestrians, which is then processed to comprehend the vehicle's surroundings accurately [95]. The network layer facilitates communication between the vehicle and external entities like other vehicles and infrastructure, aiding in real-time decision-making through shared information on traffic congestion and emergencies [95]. Finally, the application layer executes control decisions like acceleration and lane-changing, utilising the data processed in the perception layer. This layer often employs advanced algorithms and artificial intelligence for decision-making [95].

Despite autonomous vehicles' benefits, they face security and privacy concerns [96, 65]. AVs generate, collect and transmit sensitive user data, including where and when the passenger uses the vehicle and the user's environment. Therefore, it is necessary to secure data and information against malicious use and its resulting security risks [97, 19].

Micro-mobility. Micro-mobility refers to using lightweight, electric vehicles for short-distance trips [98], including smart bikes [99], and electric scooters [100]. These vehicles are increasingly becoming popular as a sustainable and convenient means of transportation, and they play a vital role in intelligent transportation systems (ITS) [101, 102]. In this regard, micro-mobility is essential to the last-mile transportation concept. The architecture of micro-mobility vehicles can also be decomposed into perception, network, and application layers [103]. The perception layer includes sensors and cameras that gather data about the surrounding environment, such as road conditions, traffic, and obstacles. The network layer enables communication among micro-mobility vehicles, users, and infrastructure.

The application layer includes the software and algorithms to semi-autonomously control the vehicle's movements, speed, and access for the user. The application layer also provides user interfaces, such as mobile apps, that enable users to locate and rent micro-mobility vehicles. However, micro-mobility solutions like AVs are not immune to security and privacy issues. They generate, gather, and transmit confidential user data, the compromise of which could jeopardise user security [103]. Moreover, their functionality directly affects the physical safety of users [103].

2.2. Security Risk Management

One primary goal of this thesis is to provide a framework for IoT security risk management (SRM). Thus, "*security engineering*" is defined in this thesis as "lowering the risk of intentional unauthorised harm to valuable assets to a level acceptable to the system's stakeholders by preventing and reacting to malicious threats and security risks" [90]. Security in the context of this paper deals with intentional, unauthorised threats and risks that explicitly harm system assets. This differs from safety engineering, where the "*lowering the risk of unintentional unauthorised harm*" [90] is considered. Safety considerations are essential in IoT systems, given their direct impact on the physical world. However, it's crucial to emphasise that security is a prerequisite for ensuring safety in IoT systems [104].

Security risk management is essential to securing IoT systems and is crucial for identifying, assessing, prioritising, and mitigating security risks. To this end, the IoT architecture provides insight into the system components and their interactions, which can help to identify potential vulnerabilities and attack vectors. This section analyses SRM methods applicable to IoT systems to select a foundational method to support the framework with crucial security concepts and relationships covering asset, risk, and risk-treatment concepts at each IoT architecture layer. Additionally, threats at each IoT architecture layer are explored, demonstrating security requirements and security controls to mitigate resulting risks, and showing the architecture perspective's implications for security risk management.

2.2.1. Security Risk Management for IoT Systems

Security risk management in IoT systems requires a comprehensive knowledge of the system's assets, their relationships with each other, and their vulnerabilities and potential risks. Although SRM models [105], best-practices [106], techniques and technologies [107] have been proposed, there are currently no specific SRM methods for IoT systems. However, some well-known SRM methods have been applied to IoT systems [108, 109, 110, 14]. Four (4) of these methods are analysed to select a foundational method that can be applied to our IoT SRM efforts.

1. *EBIOS (Expression of Needs and Identification of Security Objectives)* [111] is a systematic, five-step procedure: (1) context establishment; (2) security

requirements determination, (3) threat identification and analysis, (4) risk identification and security objectives description, (5) security controls determination, highlighting any remaining risk. EBIOS provides a structured approach to identifying risks and aligning security controls with organizational needs for IoT systems.

2. *OCTAVE Allegro (Operationally Critical Threat, Asset, and Vulnerability Evaluation)* [112] offers a risk-oriented strategic evaluation including the following steps: (1) establishing criteria for risk assessment and measurement, (2) identifying and profiling of assets, (3) identification of vulnerabilities and threats of primary assets, and (4) risk assessment and development of mitigation strategies. OCTAVE provides applicability to IoT systems where asset identification and threat profiling are crucial.
3. *ISSRM (Information System Security Risk Management)* [113] is a model-based approach proposing a conceptual reference model for SRM called the domain model, defining asset, risk, and risk treatment-related concepts. The application of ISSRM includes six steps: (1) organisational context and assets identification; (2) determination of security objectives (confidentiality, integrity and availability); (3) risk analysis and assessment; (4) risk treatment decision, which results in, (5) security requirements definition to implement, and (6) security controls. This comprehensive approach is especially relevant to IoT systems, where systematic risk management is crucial.
4. *MITRE's "Threat Assessment & Remediation Analysis" (TARA)* [114] focuses on systematically breaking down attacks and efficiently communicating risks, addressing contemporary security challenges. TARA simplifies prospective attacks into a list of likely attacks and articulates risks and suggestions for remediation. TARA was designed in response to the need to assess security risks in the rapidly evolving threat landscape. In the context of IoT systems, TARA's approach to breaking down complex threats and communicating risk can be particularly beneficial in navigating the often complicated security landscape these systems present.

However, when targeting IoT systems from an architecture perspective, there is a need to support asset identification and functional decomposition of the system and provide a model-based approach to SRM. Asset identification and functional decomposition are significant processes that help break down the system into smaller, manageable components and identify assets requiring protection. Functional decomposition involves breaking down a system into its functional components to understand better how the system works, where potential vulnerabilities might lie, and which assets are at risk. Additionally, a model-based approach to SRM can provide a systematic, repeatable and theoretically grounded method for identifying, evaluating, and mitigating security risks. This approach relies on the creation of abstract representations – models – of the system under considera-

tion [115], capturing essential security aspects and their interrelations within the system [116]. Table 6 highlights the strengths and limitations of these methods for managing security risks with the following criteria: asset identification, functional decomposition, model-based SRM, and IoT-specific considerations (such as scalability and adaptability to rapid change).

Table 6. Comparison of security risk management methods

Criteria	EBIOS	OCTAVE	ISSRM	TARA
Asset identification	++	++	++	+ -
Functional decomposition	+ -	+ -	++	+ -
Model-based support	+ -	+ -	++	- -
IoT specific considerations (i.e scalability and adaptability to rapid change)	+ -	++	++	+ -

*++ Full fulfillment, +- Partial fulfillment, and - - No fulfillment

EBIOS incorporates asset identification in the first step of its process, where the relationship between the business context and the information system is established. However, it might not be as detailed as other methods like OCTAVE Allegro or ISSRM. EBIOS partially addresses functional decomposition, using a systematic approach to identify risks and devise security controls. Still, it does not explicitly focus on decomposing the system into functional elements. EBIOS provides a systematic, five-step procedure but lacks defined model-based support like ISSRM, which could be a limitation for complex IoT systems. The structured approach of EBIOS could be scalable for IoT systems. Still, it may struggle with the rapid change characteristic of IoT because it's more oriented towards tailoring security to organizational needs rather than adapting to dynamic environments. OCTAVE Allegro stands out in asset identification, with an explicit step dedicated to identifying and profiling assets, which is crucial for IoT systems. Like EBIOS, OCTAVE Allegro doesn't explicitly address functional decomposition, but the process does involve identifying vulnerabilities and threats to primary assets, which involves some form of decomposition. OCTAVE Allegro uses a risk-based strategic approach, which, although systematic, doesn't offer the same level of model-based support as ISSRM. The method shows potential scalability for IoT systems due to its focus on primary assets and vulnerabilities. It also shows a degree of adaptability to rapid change due to its risk-oriented nature. While TARA doesn't explicitly focus on asset identification, it strongly emphasises decomposing and understanding attacks, which indirectly includes recognising assets at risk. Unlike ISSRM, TARA does not use a defined model-based approach and functional decomposition is not represented, which could limit its effectiveness for complex systems. TARA can be adapted to IoT considerations due to its approach of breaking down complex threats and focusing on risk communication. Still, it may not be as scalable or adaptable to rapid change as ISSRM. ISSRM involves organizational context and asset identification as the first step, aligning well with the needs of IoT systems. ISSRM stands out for its model-

based support with its domain model that defines asset, risk, and risk treatment concepts. This makes it particularly promising for systematic asset identification and functional decomposition of the system as an input to security risk analysis and treatment. ISSRM's comprehensive approach makes it scalable for IoT systems. Furthermore, its model-based approach ensures systematic risk management and decision-making, allowing for better adaptability to rapid changes in the IoT landscape. However, it may require more effort to implement the ISSRM method due to its model-based approach. ISSRM provides a better basis for building an architecture-based IoT SRM framework. While the other methods have strengths, ISSRM comprehensively addresses all the required criteria.

Based on our analysis, the ISSRM method is selected for further consideration in supporting IoT SRM. However, while ISSRM is preferred for analysing IoT systems from an architectural perspective, other methods covered above can also complement SRM in specific contexts. The ISSRM method covers three major SRM concept groups: asset-related, risk-related, and risk treatment-related concepts [117]. As depicted in Figure 8, these ISSRM concepts and their interconnections are highlighted with the corresponding ISSRM process.

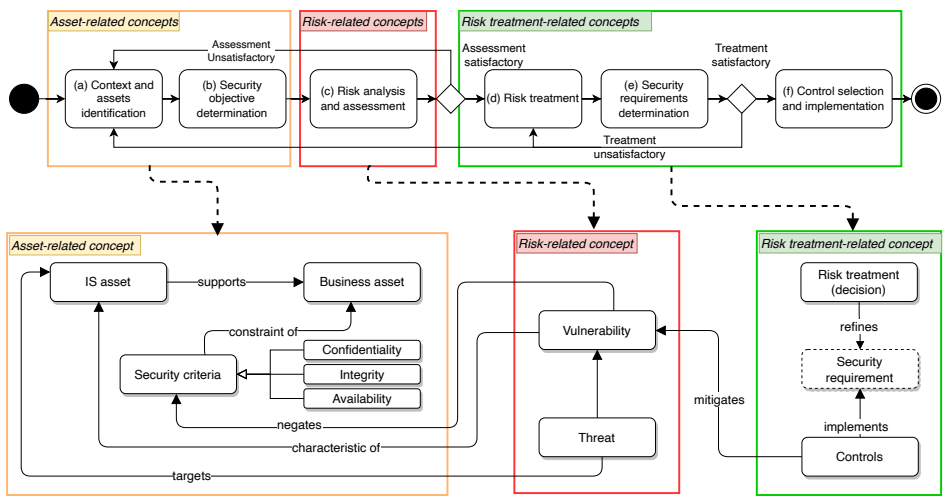


Figure 8. ISSRM process illustrating key concepts and relationships [118, 117, 119]

Asset-related concepts describe constructs for critical business and information system (IS) assets to protect, and the security criteria guarantee a certain level of asset security (in terms of confidentiality, integrity, and availability). *IS assets* are components of the system (e.g., hardware, software, or network) that support *business assets* (i.e., information, data, and processes) that bring business value. *Security criteria* determines the level of asset security (confidentiality, integrity, and availability) defined for each identified *business asset*. Activities *(a), (b)* in Figure 8 cover asset-related concepts.

Risk-related concepts introduce constructs for security risk itself and its com-

ponents (threat, vulnerability, risk impact, etc.). A *vulnerability* constitutes the weakness of the *IS assets*. A *threat* thus exists when an entity with interests can exploit a vulnerability to harm the *IS assets* and negate the *security criteria* of the *business assets*. A security *risk event* is an occurrence where a *threat* exploits one or more *vulnerabilities* of the *IS assets*. The security *risk impact* are the negative consequences of the *risk event* that harms the vulnerable *IS assets* and negates the *security criteria* of the affected *business assets*. Thus, a security *risk* is the likelihood of a *risk event* occurring and the potential *risk impact* of the event. Activity *(c)* in Figure 8 cover risk-related concepts.

Lastly, *risk treatment-related* concepts describe constructs to treat risk, including the risk treatment decision, security requirements, and the controls that implement the defined security requirements. *Security requirements* are the conditions to be reached by mitigating the security risks. Following these requirements, security *controls* to treat the identified security risks are implemented. Activities *(d), (e), (f)* in Figure 8 cover risk treatment-related concepts.

The ISSRM method is used as the theoretical foundation of our framework in Chapter 3 leveraging key concepts and relationships of the ISSRM method to provide a structured and effective approach to IoT SRM based on the system's underlying architecture.

2.2.2. Security Threats in IoT Architecture Layers

This section uses the architecture perspective to uncover security threats in our intelligent transportation system scope. Threat identification is also used as a primary driver for security risk analysis efforts [120], where STRIDE is applied to perform threat analysis.

STRIDE is an industrial-level method used for threat scenario elicitation and analysis [121] that consists of the following elements: **S**poofing – involves pretending to be something or someone you're not, impersonating entities to deceive others. **T**ampering – refers to unauthorised modification of something, altering data or functions you're not permitted to change. **R**epudiation – involves denying an action you've performed, whether the claim is true or false. **I**nformation Disclosure – involves the exposure of information to individuals who are not authorised to access it, potentially leading to breaches of confidentiality. **D**enial of Service – are attacks aimed at obstructing a system's ability to provide its intended services, often causing disruptions or complete shutdowns. **E**levation of Privilege – occurs when a program or user gains access to functions or permissions beyond what they are technically allowed to execute.

STRIDE is suitable as the security threat analysis method due to its industry usage, maturity, and high research concentration and use within the security community, making it beneficial for SRM. The STRIDE method can also be applied for threat analysis at each IoT system layer [19, 122]. Additionally, each threat section within STRIDE offers a deeper explanation of the threats, including details

about the violated security properties. These are;

- Spoofing - *Authentication*
- Tampering - *Integrity*
- Repudiation - *Non-repudiation*
- Information Disclosure - *Confidentiality*
- Denial of Service - *Availability*
- Elevation of privilege - *Authorisation*

These security properties form the security requirements to be fulfilled to defend against the STRIDE threats. These security requirements guide proposed security controls.

In this section, assets in our intelligent transportation system case are summarised, and security threats at each layer are discovered using STRIDE, with suggested security controls at each layer following the STRIDE security requirements. Although not the focus of our analysis, closely related privacy threats are also summarised.

Assets. IS and business assets in ITS can be organised into three layers: perception, network, and application. Table 7 provides a comprehensive overview of the IoT layers and their assets. The perception layer includes sensing, vision, positioning, and actuating assets, which collect and process perception data. The network layer includes assets for in-vehicle, vehicle-to-vehicle (V2V), and vehicle-to-infrastructure (V2I) communication. The application layer includes assets for computing/server, data storage, and human interaction, which enable data processing, storage, and presentation.

Perception layer threats. The perception layer provides capabilities (sensing, visioning, positioning, and actuating) to detect and identify, communicate, collect and gather data about the IoT environment from IoT devices, usually without human interaction [85]. The perception layer is a critical component of an IoT system as it collects and processes sensitive data from connected devices. Table 8 includes security and privacy threats [19, 123] prevalent in the perception layer where STRIDE is applied to discover threats. The perception layer requires physical security for the devices and secure data collection to ensure security. This means security mechanisms must be established to protect the data and the devices from malicious attacks and breaches [19]. However, the sensing layer devices are designed for low power consumption and have limited resources, often resulting in limited connectivity. This wide variety of IoT applications poses various security challenges, such as ensuring device authentication and trusted devices, leveraging the security controls and infrastructure available in the sensing layer, and ensuring timely software updates and security patches without compromising functional safety [124].

Table 9 highlights security controls at the perception layer. One of the key security mechanisms in the perception layer is data integrity and validation. This ensures that the data received from the devices has not been altered in transit,

Table 7. Architecture layer assets [19]

Layer	IS assets		Business assets
Perception	Sensing	Light detection and ranging (LiDAR), visible light communication (VLC), ultrasonic ranging devices (URD), millimeter wave radar, thermometer and infrared ranging	Ultrasonic data, radio frequencies, heat measurement, traffic count, travel time, vehicle weight data
	Vision	Video cameras, HD cameras, stereo vision systems, and Closed-circuit television camera (CCTV)	Surveillance (picture and video) data, 3D imaging data, traffic count
	Positioning	Global positioning system (GPS) receiver and radars (doppler radar speedometers, radar cruise control, and radar-based obstacle detection systems)	pseudo-range measurements, travel speed, radar data, vehicle location data
	Actuating	Vehicle node, ECU, key/remote device, infotainment	Mileage measurement, error codes, event data records, traffic warning messages, key/remote signal, transaction information.
Network	In-vehicle	Controller area network (CAN), automotive Ethernet, byteflight, FlexRay, local interconnect network (LIN), low-voltage differential signalling (LVDS), and media oriented systems transport (MOST)	Perception data (e.g., tire pressure monitoring system (TPMS) messages)
	Vehicle-to-vehicle (V2V)	Dedicated short-range communications (DSRC) / wireless access in vehicular environments (WAVE), LTE/5G, worldwide interoperability for microwave access (WiMAX)	Perception data (e.g., travel direction, vehicle range data)
	Vehicle-to-infrastructure (V2I)		Perception data (e.g., Traffic count, accident data, transaction information, vehicle range data)
Application	Computing /Server	Web application server	Application service, application process, application data, perception data (e.g., key/remote signal, vehicle location data)
	Data Storage	Datacenter, Edge data center (Fog)	Perception data (e.g., vehicle location data)
	Human	User, Driver, Administrator	Application process

is complete, is in the correct format, and falls within the expected range [128]. Additionally, encryption security mechanisms are used to protect the data during storage on the device and to prevent unauthorised access or breaches [131, 130]. Authentication mechanisms are also important in the perception layer, as they help verify the devices' identity before collecting data from them [126], thus preventing unauthorised devices from sending false or malicious data [128]. Access control mechanisms are also used to restrict access to the collected data to authorised parties and implement role-based access control for different users and devices [131, 128]. Logging and auditing mechanisms can be implemented to keep track of all the activities and events in the perception layer. This can be useful for forensic analysis and identifying patterns in the event data collected to detect anomalies or suspicious activity [128, 129]. The perception layer in a cloud-based computational IoT system also includes edge computing capabilities. Thus, it's important to ensure the security of data processing, storage, and transmission at the edge level. This includes securing the communication between the edge device and the cloud and implementing security controls to protect data storage and processing at the edge [124].

Table 8. Perception layer security and privacy threats [19, 123, 124]

Asset	Threats		Perception Layer
Sensing, Positioning, and Vision, Actuating	Security Threats [19]	S	Spoofing, Node Impersonation, Illusion, Replay, Sending deceptive messages, Masquerading
		T	Forgery, Data manipulation, Tampering, Falsification of readings, Message Injection, State manipulation
		R	Bogus message
		I	Stored attacks, Eavesdropping
		D	Message saturation, Jamming, Denial of Service (DoS), Disruption of system, Battery exhaustion DoS/ Distributed denial of service (DDoS)
		E	Backdoor, Unauthorised access, Malware, Elevation of privilege, Remote update of devices
	Privacy Threats		Sensor data leakage (i.e., Inertial Measurement Unit (IMU)), Wearable device privacy leakage, Identification threat, i.e. device owner fingerprinting, Data over-collection, Localization leakage, Inventory attack

Table 9. Perception layer security controls

Sec Req	Perception Layer
Authentication	Spoofing resistant positioning system [125], device level user authentication [126], digital certificates, digital signature of software and sensors [127, 128], challenge/response mechanism [128], encrypted Precise Positioning System (PPS)
Integrity	Restricted physical access [127], challenge/response mechanism [128], use trusted hardware [128]
Non-repudiation	Use trusted hardware [128], event data collection [129]
Confidentiality	Encryption [130, 131]
Availability	Distributing the usage across the spectrum, MAPE architecture [132], Qlearning algorithm [133]
Authorization	Threat modelling [134], hardware and software access control [128], upgrading on-board device ports [135]

Network layer threats. The network layer plays a critical role in the communication and transmission of data between perception objects, other network devices, infrastructure (central or connected), or application layer objects and services. However, it is also susceptible to various security attacks such as eavesdropping, malware, unauthorised access, denial of service attacks, man-in-the-middle (MitM) attacks, etc. These attacks can cause significant disruptions in data transmission and may lead to the leakage of sensitive information [138]. Table 10 includes security and privacy threats [19, 123, 131] prevalent in the network layer where STRIDE is applied to discover threats.

To mitigate these risks, the STRIDE security requirement is used to document security controls in Table 11. The network layer requires security algorithms for confidentiality, user authentication, and data integrity. Cryptographic protocols are also necessary to prevent denial of service attacks and ensure that data

Table 10. Network layer security and privacy threats [19, 123, 131]

Asset	Threats		Network Layer
In-device, Device-to-device, Device-to-Infrastructure	Security Threats	S	Sybil, Spoofing (GPS), Replay attack, Masquerading, RF Fingerprinting, Wormhole, Camouflage attack, Impersonation attack, Illusion attack, Key/Certificate Replication, Tunneling, Position Faking
		T	Timing attacks, Injection (message, command, code, packet), Manipulation/Alteration/ Fabrication/Modification, Routing modification/manipulation, Tampering(broadcast, message transaction, hardware), Forgery, Malicious update (software/firmware)
		R	Bogus messages, Rogue Repudiation, Loss of event traceability
		I	Eavesdropping, Man-in-the-middle, ID disclosure, Location tracking, Data sniffing, Message interception, Information disclosure, Traffic analysis, Information gathering, TPMS tracking, Secrecy attacks
		D	DoS/DDoS, Spam, Jamming, Flooding, Message suppression, Channel interference, Black hole, SYN Flooding, Routing table overflow
		E	Malware, Brute Force, Gaining control, Social engineering, Logical attacks, Unauthorised access, Session Hijack
	Privacy Threats		Device beacon privacy leakage, Identification threat, Traffic data matching [136], Network traffic monitoring [137], Localization leakage, Tracking attack, Lifecycle transitions leakage [123]

is available only to intended devices and users [138]. Security mechanisms for the network layer include using secure protocols such as HTTPS, SSL, or VPN to encrypt the data in transit and prevent eavesdropping or man-in-the-middle attacks [134]. Network segmentation is also used to divide the network into smaller segments to limit the spread of any potential security breaches [128]. Firewalls and intrusion detection systems are implemented to monitor and control access to the network and detect malicious activity [130]. Additionally, mechanisms can be implemented to authenticate and authorise devices and users before allowing them access to the network. Device management is also important in the network layer, as it allows keeping track of the devices connected to the network, revoking access for devices that are no longer authorised and developing secure methods for remotely managing and updating devices, such as over-the-air (OTA) updates [139].

Application layer threats. The application layer of an IoT system plays a crucial role in delivering application-specific services to the end user based on the application type, business and profit models, and information provided by the perception objects. Depending on the use case, this layer can include service-oriented technologies such as cloud computing, storage, and integrations with other applications to perform the necessary activities for the end-user [79, 15]. Table 12 includes security and privacy threats [19, 123, 131] prevalent in the application layer where STRIDE is applied to document threats.

The STRIDE security requirement is used to document security controls in Table 13 to mitigate these risks. This layer provides various services, and as a result, different security mechanisms are beneficial for security, including access con-

Table 11. Network layer security controls

Security Req	**Network Layer**
Authentication	ID authentication [139], radio-frequency identification (RFID) tokens, public key infrastructure [140, 130], WAVE security standard [141], secure routing protocol [134], reputation scoring [142], central validation authority (CVA) [143, 128], secure location verification [128], digital certificates and digital signatures [127, 142, 128], bit commitment and zero-knowledge mechanisms [128], variable MAC and IP addresses, challenge/response mechanism [128]
Integrity	Public key infrastructure (PKI) [144, 139], hashing function, cryptographic primitives [128], security protocol [139], plausibility validation network (PVN) [144]
Non-repudiation	Identity monitoring system [145], auditing and logging [146]
Confidentiality	Vision integrated pseudorange error removal (VIPER) algorithm [128], encryption [130, 139, 128], secure routing protocol [134], key management [134, 139, 135], digital signatures [135, 139], WAVE security standard [141], firewall [130]
Availability	Frequency hopping spread spectrum (FHSS) technique [142, 128, 143], secure routing protocol [134, 142], time stamping mechanism [128], bit commitment and signature based authentication mechanisms [128], WAVE security standard [141], firewall [130]
Authorization	Variable MAC and IP addresses, network segmentation [128], WAVE security standard [141], intrusion detection system, honeypot [134, 135], device management

Table 12. Application layer security and privacy threats [19, 123, 131]

Asset	Threats		Application Layer
Application server, Edge data center, Human	Security Threats	S	Identity spoofing, Sybil, Illusion attack, User impersonation
		T	Malicious Update, Malicious node manipulation
		R	Event log tampering
		I	Eavesdropping, Location tracking, Privacy leakage, Sniffing Attack
		D	DoS
		E	Jail-breaking OS, Social engineering, Rogue Data-center, Malware, Spear-Phishing attack. Malicious users
	Privacy Threats		Localization leakage, Device tracking, Tag tracking

trol, cryptographic services, authentication protocols, and blockchain technology. Additionally, privacy preservation and data protection are also necessary security requirements. Data protection mechanisms can also protect data stored in the application from unauthorised access or breaches. Input validation and sanitization of the input data to prevent injection attacks or other malicious inputs is another security mechanism [138]. Additionally, user management, including managing user accounts and permissions and implementing mechanisms for password recovery and other security features, is also important. Auditing and logging mech-

Table 13. Application layer security controls

Security Req	**Application Layer**
Authentication	Digital certificates and digital signatures [127, 128]
Integrity	Plausibility Validation [20]
Non-repudiation	Blockchain [147, 148], multiparty computation [149], digital signature [148], public key cryptography (PKC) [148] , auditing and logging mechanisms [150]
Confidentiality	Firewall [130], cryptography services [141]
Availability	Cloud computing, microservices [151]
Authorization	Firewall [130]

anisms keep track of all the activities and events in the application layer, which can be useful for forensic analysis. Application security testing is another mechanism to ensure regular application testing for vulnerabilities, implementing patches or updates to fix any issues, and ensuring that the application meets relevant security standards or regulations. Secure third-party integration is also important to ensure that the integration with other systems or third-party services is done securely and without introducing any vulnerabilities. Overall, the application layer plays a critical role in the security of an IoT system to protect the application and its data from malicious attacks and breaches.

2.2.3. Implications of IoT Architecture to Security

An IoT system's complex interactions and dependencies result in a tightly coupled association between its components and subsystems [152]. These dependencies can be direct or indirect, meaning that the security risks associated with IoT systems are not only caused by the compromise of specific components but also by the impact of that compromise on other connected components [20, 153].

The dependencies between IoT subsystems and components can cause security risks to cascade from one affected system or component to another, amplifying the damage resulting from the security risks [20, 153]. For example, the normal functioning of components and services on the application layer typically depends on the normal functioning of their supporting component on the network layer, which relies on a perception layer component or service. Thus, if a component or service on the perception layer is compromised, such an attack can cause a ripple effect, altering the correct functioning of the connected components or services on the network or application layers. These cascading effects of security risk impacts at the application layer also emphasise the need for cooperative security defence across all layers of the IoT architecture [153]. As such, the application layer can defend against threats originating from its layer and associated components and sub-systems in the perception or network layers, providing a last line of defence against harmful effects on IoT end-users [19].

Integrating IoT architecture into SRM can thus improve identifying and miti-

gating security risks associated with IoT systems. Existing IoT SRM approaches that do not consider the IoT architecture perspective may not fully identify relevant assets and their layered dependencies, thus performing inadequate risk assessment and limiting the identification and prioritization of mitigation strategies. Comprehensive multi-layer security risk and risk treatment analysis are necessary to manage security risks in IoT systems.

2.3. Hackathons as a Educational Approach for IoT Security Risk Management

Hackathons are time-bounded events in which participants of different backgrounds form teams and work on projects of interest to them [9], thus encouraging open innovation, awareness, and increased research focus on particular topics, including IoT [154]. Hackathons serve as an innovative educational platform, allowing for hands-on, collaborative learning experiences that align with constructivist and social cognitive theories [9, 155, 156, 157]. Hackathons have the potential to bridge the theoretical-practical gap, enabling real-world application of concepts such as IoT SRM where traditional educational models may not suffice [158, 7]. This section delves into the rationale for utilising hackathons as an educational tool, contrasting them with other teaching methods and offering guidelines for using hackathons to enhance learning in IoT SRM.

2.3.1. Educational Rationale for Hackathons

Teaching IoT SRM involves blending conceptual, practical, and collaborative learning [158, 7]. The subject's complexity, spanning technical and management aspects [7], requires careful planning. A significant aspect is contextual understanding [7], imparted through theoretical groundwork on security principles, IoT technologies, and risk management frameworks. Practical-oriented learning is vital [158, 159], as theoretical knowledge alone might not prepare learners for real-world challenges. Moreover, fostering a collaborative learning environment is essential [7]. Given the dynamic nature of IoT SRM, teaching approaches must be adaptable [158, 160, 161].

Well-known educational approaches like hackathons, lectures, training and workshops, and project-based learning, are evaluated, showing the rationale for using hackathons in cybersecurity education. Table 14 compares the suitability of each approach for IoT SRM learning. These criteria include delivering complex topics, fostering the rapid application of concepts, promoting collaboration and teamwork, enabling application in real-world scenarios, providing mentorship and feedback, and minimising resource requirements. These criteria were selected to encapsulate the essential characteristics needed for cybersecurity and IoT SRM education.

Among the compared methods, hackathons stand out for their effectiveness in fostering collaboration, creativity, and rapid application of security concepts

Table 14. Comparison of teaching strategies for IoT SRM learning

	Hackathons	**Lectures**	**Trainings and Workshops**	**Project-based Learning**
Delivery of complex topics	[++]	[++]	[+−]	[++]
Rapid application of concepts	[++]	[−−]	[−−]	[+−]
Application in real-world scenarios	[++]	[−−]	[−−]	[++]
Collaboration and teamwork	[++]	[−−]	[+−]	[++]
Mentorship and feedback	[++]	[−−]	[+−]	[++]
Minimised resource requirements	[+−]	[++]	[+−]	[+−]

[++] Mostly fulfilled, [+−] Fulfilled with limitations, [−−] Not fulfilled

to real-life scenarios [68, 162]. Hackathons represent learning opportunities promoting collaboration, creativity, and the rapid application of security concepts to real-life scenarios [68, 162, 163, 69]. They are uniquely suited for introducing complex security topics and helping participants devise solutions quickly. Additionally, hackathons provide a platform for mentorship, feedback, and safe experimentation, fostering the development of problem-solving skills, critical thinking, and team effectiveness.

Traditional teaching strategies like lectures offer structured presentations to stimulate learning [164]. However, their main limitation is the lack of active learner engagement and limited opportunities for hands-on activities [165]. Similarly, training and workshops can be customised to meet different learner needs and skill levels [166, 167]. They are easy to introduce and require minimal resources. However, these methods may not provide deep learning or practical experience in real-world scenarios, which is critical for cybersecurity professionals. Project-based learning (PBL) is a longer-term educational approach that typically unfolds over weeks or even months, offering students a flexible learning environment [168, 169, 170, 171, 172] within that academic period. In contrast, hackathons are ideal for simulating real-world applications in a constrained time frame and are adaptable outside the academic environment. The comparison of these learning approaches reveals that hackathons meet key criteria relevant to IoT SRM learning.

2.3.2. Hackathons for Learning IoT SRM

Hackathons have become popular for promoting cybersecurity awareness and training in computer science and software engineering communities [162], allowing organisers to provide learning opportunities that enable participants to apply new skills and deepen their understanding of specific topics. Several studies [173, 174, 175] have demonstrated the effectiveness of using hackathons for education and learning. Thus, hackathons can offer an effective platform for learning about IoT security and SRM.

Among the successful examples of hackathons for IoT security learning is the annual IoT Village at DEF CON[1], which focuses on IoT security and hosts Cap-

[1] https://www.iotvillage.org/

ture the Flag (CTF) competitions that challenge participants to locate vulnerabilities in various IoT devices. This event helps identify weaknesses in IoT devices and allows participants to learn from each other and acquire new skills. Balto *et al.* [176] proposes using a hybrid IoT cyber range as a training ground for IoT security. This hackathon training environment highlighted the importance of education, awareness, demonstrations, and training in improving IoT security. Byrne *et al.* [177] employed week-long hackathon workshops to motivate preuniversity teenagers to pursue careers in STEM. The hackathon event and the learning model effectively increased the students' self-efficacy in emerging technological contexts such as IoT and wearables. Junior *et al.* [178] conducted a study on a CTF competition based on IoT cybersecurity as a learning tool and found that such hands-on activities and interactive competitions significantly contribute to learning about IoT cybersecurity.

Despite these positive insights, the literature reveals a gap in hackathon-like approaches supporting comprehensive knowledge about SRM. CTF or competition-based hackathons primarily target system vulnerability identification and exploitation, secure coding and system defense [179, 180] but lack the granularity required for a thorough asset-oriented system analysis [181]. Additionally, in cybersecurity education, offensive security and network security are typically the focus as opposed to other areas of cybersecurity education [182], including SRM. Addressing this gap provides critical input for IoT SRM and further exploration of hackathons can enhance the effectiveness of hackathons as a learning tool for IoT SRM.

2.3.3. Designing Effective Hackathons for IoT SRM Learning

Designing a hackathon focused on IoT security risk management (SRM) involves careful design. Organisers must consider elements like instilling fundamental cybersecurity knowledge, offering hands-on experiences, promoting both adversarial and system thinking, as well as fostering soft skills and self-directed learning [7, 158, 159]. It's important to note that while organisers establish the framework for the hackathon, the onus of maximising learning within this structure largely falls on the participants [183]. Therefore, as the thesis proceeds, best practices from an organizer's viewpoint will be outlined to ensure an optimal learning environment for participants in IoT SRM.

Nolte *et al.* [184] proposed a hackathon planning kit [2] that outlines 12 major decision points to consider when organising hackathons for specific outcomes. These decision points are domain-agnostic, formulated to enhance the success of a hackathon event [184]. To our knowledge, the hackathon planning kit proposed by Nolte *et al.* [184] offers organisers the most comprehensive guide on organising hackathons. This hackathon planning kit is adapted as the basis of our hackathon learning context, where five (5) of the twelve (12) key decision points, crucial for achieving successful hackathon learning outcomes and facilitating par-

[2] https://hackathon-planning-kit.org/

ticipant learning, are selected. Our selection of decision points is based on their relevance to participant behaviour and learning. For instance, decision points that encourage participant engagement, collaboration, and reflection, which enhance learning outcomes, are prioritised. Conversely, decision points less directly related to participant learning, such as those focused on stakeholder involvement (i.e., in fundraising or marketing), duration/breaks, participant recruitment, etc, are de-prioritised. A hackathon environment that maximizes SRM learning opportunities for participants is created by focusing on decision points that support participant learning, encouraging them to take an active role in their education. It is worth noting that while goals, themes, stakeholder involvement, participant recruitment, duration/breaks, agenda setting, and continuity planning [184] are beneficial to the overall organization and success of a hackathon, they do not constitute design actions to enhance learning. These elements offer broad directives, contextual backdrops, logistical support, and post-event sustainability but do not dictate actionable steps for learning outcomes. However, these decision points are still invaluable as part of a typical hackathon organization.

- **Ideation**: Ideation assists participants in generating engaging and feasible project concepts through conventional brainstorming [185]. Ideation before an event can allow familiarization with the idea and kickstart the learning process [68].
- **Team Formation**: Team formation is based on the participants recruited considering the backgrounds, experience, and interests to create teams that can work collaboratively towards solving security challenges [68]. Participants can develop various skills and knowledge by finding team members with complementary skills and forming teams around the ideas generated [184].
- **Mentoring**: Mentors are critical in improving the learning experience of participants, providing guidance, support, and expertise to help them develop their skills, knowledge, and confidence in tackling security challenges [173].
- **Specialised preparation**: To facilitate effective security learning during a hackathon, organisers may need to provide participants with relevant learning content, such as specific technical or domain knowledge [67]. This could be done before or during the hackathon and should be tailored to align with the learning objectives and the participant's needs. Such an approach fosters a hands-on and experiential learning process, where participants gain knowledge through practical application [186]. However, it's crucial to adjust the complexity of the learning content to match the participants' expertise level. Novices may need more basic security concepts, while advanced participants might benefit from more complex and detailed materials. Hackathons can effectively accomplish their educational objectives by ensuring the learning content aligns with the participants' skill lev-

els. Moreover, this environment encourages participants to learn from one another, share knowledge and skills, and collaboratively tackle projects.

- **Competition /Cooperation**: Organisers decide whether to introduce competition to generate unique solutions under competitive pressure or focus on cooperation to engage participants in a common learning goal or theme. The element of competition can create a sense of urgency and purpose that motivates participants to learn and apply new skills and create unique solutions [68].

These decision points can be translated into actionable design considerations called "hackathon interventions" to optimise the learning outcomes of the hackathon. By developing and adapting these design points, hackathon organisers can create an environment that maximises IoT SRM learning opportunities. These hackathon interventions are formalised in Chapter 4 and validated through multiple hackathon contexts to show learning benefits to IoT SRM.

In the evaluation and study of the hackathon approach, we also acknowledge a growing demographic among hackathon participants, which is the Generation Z (Gen Z)[3]. GenZ have developed unique learning preferences such as the reliance on digital technology and a pattern towards self-directed learning, making them more likely to view hackathons as suitable learning environments [187, 188]. This presents a promising opportunity for the development of IoT SRM-focused hackathon, tailored to meet their specific educational needs and preferences.

2.4. Summary

In this chapter, we explored the significance of IoT system architecture, security risk management, and the use of hackathons as an educational tool. The IoT architecture is seen to offer benefits to facilitate identifying and mitigating associated security risks. As demonstrated, the IoT architecture provided a structural approach to identifying relevant assets in their IoT layers as well as their dependencies. These benefits contribute to improved strategies for tackling security risks in IoT systems. One of such strategy is the multi-layer approach to SRM where each IoT system layer is analysed to discover layer-specific risks and cascading effects of security risks from one layer to another.

However, as such strategies for IoT SRM target complex IoT systems there is a necessity for practical teaching methodologies to enable adoption. Hackathons, are selected amongst other teaching methods as it provided a promising practical approach for teaching and applying complex SRM concepts to real-world IoT systems, while boosting skills such as collaboration and teamwork. Additionally, hackathons for education can be tailored to meet the needs of IoT SRM education through the development and adaptation of "hackathon interventions".

[3]The term "Gen Z" is not uniformly defined in terms of its birth year range, but it generally refers to young people currently in undergraduate university programs, all of whom have been born into a world rich in technology.

3. IOT ARCHITECTURE-BASED SECURITY RISK MANAGEMENT

In this chapter, we draw from publications [67, 65, 66] that inform on the development and application of a framework to manage security risks in IoT systems, answering the research question:

RQ$_1$. *How to manage security risks in IoT system architectures?*

The outcome of this chapter is an IoT architecture-based security risk management (IoTA-SRM) framework to overcome the limitations of existing IoT security risk management approaches by incorporating the IoT architecture perspective into the security risk management process, ensuring that relevant assets and relationships are identified, and security risks are properly analysed and mitigated.

From the analysis of security risk management methods for IoT systems in Chapter 2, the framework's security risk management concepts are guided using the ISSRM method. The conceptual model is defined in Section 3.2, with a primary focus on the architecture perspective, emphasising the asset-oriented nature of IoT systems. This approach allows for systematically exploring IoT assets, providing the necessary inputs into the SRM activity. In Section 3.3, the core activities of the framework are defined to guide risk management, each being supported by a set of guidelines and domain-specific outputs. These activities are iterative and may be revisited as necessary, following the guidance of an ongoing risk management cycle.

The validation of the framework is undertaken through two IoT case studies in Section 3.4, wherein the lessons learned from the application of the framework within these case studies are described. Finally, the implications of this contribution are discussed in Section 3.5 and the chapter concludes in Section 3.7.

3.1. IoTA-SRM Framework Development Overview

In light of the rapid proliferation of IoT devices and applications, managing the security risks associated with these systems has become increasingly crucial. However, given the complexity and dynamic nature of the IoT system, security risk management practices require comprehensive knowledge of the system architecture and its assets as an input to security risk management.

In the initial stages of our framework development, the security focus was broader, centred around security threats in IoT systems Section 2.2.2. The identification and selection of the three-layer architecture for our framework were carried out as described in Section 2.1.2. The architectural perspective aids in comprehending the foundational elements, levels, and interactions within IoT components, thereby assisting in identifying potential vulnerabilities and attack vectors crucial for managing security risks. These formed the initial basis of our framework, providing a structural underpinning that would guide subsequent refinement

and evolution. However, though prior research has emphasised the significance of an architectural viewpoint when assessing IoT systems' security [16, 17, 18], these methods typically do not incorporate this perspective directly into the security risk management process, therefore falling short of fully actualising the principles of security by design. Thus, security risk management principles were incorporated into the subsequent stage of our approach. Informed by the analysis of existing security risk management methods for IoT systems in Section 2.2, the vital concepts of the ISSRM method were integrated into our framework. This resulted in introducing a systematic understanding of risk, thereby enhancing the framework's efficacy. Within this stage, our IoTA-SRM framework was established in Section 3.2, enabling a systematic examination of IoT assets, providing essential inputs for IoT security risk management. Finally, the focus was narrowed to IoT systems and their architecture-based characteristics. This refinement led to the definition of the core activities that guide risk management in Section 3.3. These core activities are supported by guidelines and domain-specific outputs catering to the security needs of IoT systems. The developed framework was validated through case analysis in Section 3.4.

3.2. IoTA-SRM Framework

The IoTA-SRM framework is built upon the principles of the ISSRM method, integrating key elements of asset, risk, and risk treatment-related concepts. It recognises the multi-layered structure of IoT architecture, each encompassing both IS (information system) and business asset-related concepts. The IS assets in IoT represent the physical or virtual objects/things that are instrumental in the functioning of the IoT system. Within each architectural layer of IoT, these objects can be further decomposed. A visual representation is provided through a meta-model of constructs, as shown in Figure 9.

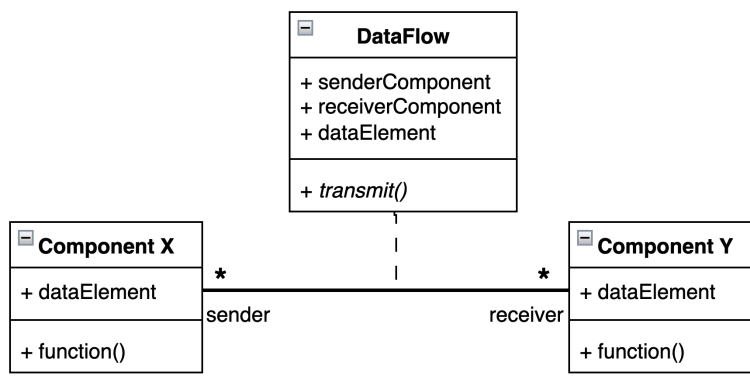


Figure 9. Meta-model of constructs

This model conceptualises IS assets as *component* elements with sub-element relationships. Business assets in the IoT system comprise assets that provide busi-

ness value and are supported by the *IS assets*. These include the data elements, functions, and data flows critical to the IoT system's business functions, conceptualised in the model as $functions$, $dataElements$, and $dataFlows$ that comprise the transit of $dataElement$ between each sender and receiver *component*. Data elements comprise the critical business data the IoT components support to provide the intended IoT business service. Functions are various data processing and transmission tasks executed by the components, and they map to data elements, components, and data flows. Lastly, each data flow comprises the transit of certain data elements between each sender and receiver object within and between the IoT layers. In this context, the sender object is responsible for data transmission, while the receiver object is in charge of receiving and, if necessary, processing the data. Thus, the conceptual model for the IoTA-SRM framework is depicted in Figure 10.

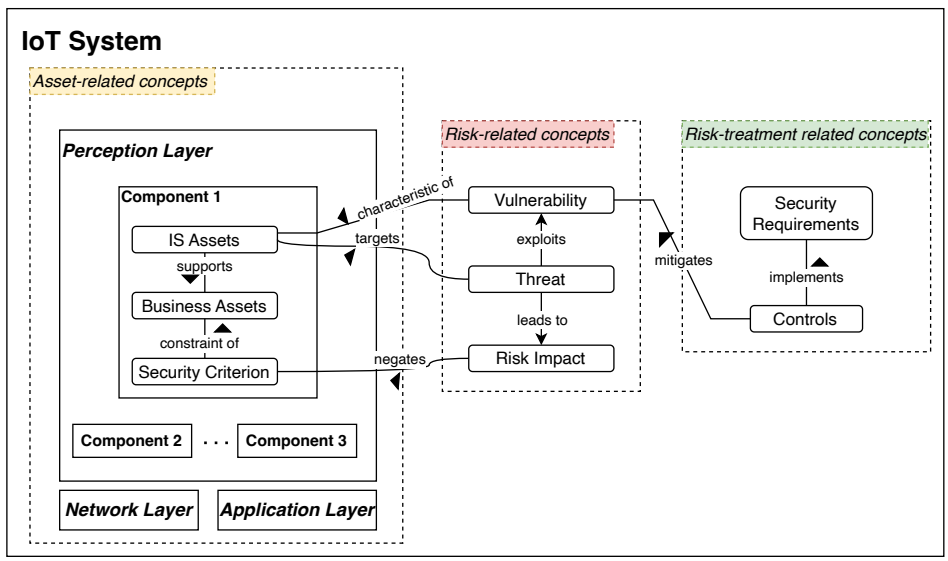


Figure 10. IoTA-SRM conceptual model

Each IS asset (component and sub-elements) can have a *vulnerability*, which presents a potential weak point within the system. When exploited by a *threat* agent, it results in a security *risk impact* at each layer and between layers. These vulnerabilities vary, ranging from simple configuration errors to complex software bugs. Common vulnerabilities per component may include SQL injection, buffer overflows, insecure data storage, or weak authentication, often referred to in vulnerability databases such as the common vulnerabilities and exposures (CVE)[1] databases. When these vulnerabilities are exploited, they evolve into security threats, as detailed in Section 2.2.2. Such threats can be organised and categorised according to methodologies like STRIDE [189], leading to security

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

risk impacts on the IoT system. For risk treatment, *security requirements* for each layer and security *control* implementation according to the security requirements to address the identified security risks within the security budget is defined. Security requirements can be categorised following STRIDE security requirements (see Section 2.2.2) guiding the selection of appropriate security measures (i.e., access control, encryption, intrusion detection systems, or firewalls), each tailored to mitigate security risks. By analysing the IS assets that support data elements, associated risks can be identified, and appropriate security controls can be implemented to safeguard against unauthorised access or manipulation of the data. Similarly, analysing the IS assets that support data functions helps identify risks related to data manipulation and processing functions. It enables the implementation of appropriate security controls and mechanisms to prevent unauthorised access, manipulation, or data leakage. Finally, the security analysis of IS assets that support data flows ensures that the confidentiality, integrity, and availability of transmitted data are maintained.

These constructs and their interactions define a dependency association within each IoT system component and other sub-systems that are either direct or indirect. These dependencies between components show the impact of security risks between layer components. Thus, if a component or service on the perception layer is compromised, such an attack can cause a ripple effect, altering the correct functioning of the connected components or services on the network or application layers. Hence, the IoTA-SRM framework emphasises that IoT security risks from the layered perspective are not only caused by the compromise of a specific IoT component at a specific architecture layer but also by the compromise of other IoT components that are impacted by it. The dependencies between IoT subsystems and components also cause security risks to cascade from one affected system or component to another, amplifying the damage resulting from the security risks [19].

In addition to identifying and analysing potential risks, developing a systematic approach for managing security risks is advantageous. A registry-like approach is proposed by us to ensure efficient management, mainly when dealing with dozens of risks across multiple layers. Each identified risk is assigned a composite identifier featuring a unique ID and pertinent descriptors to facilitate traceability. For instance, risks can be documented and categorised based on their layers, types of threats, or other pertinent criteria, accompanied by corresponding identifiers. In Section 3.3, the process guided by the framework will be explored, with the core activities involved in the effective management of security risks within IoT systems outlined. These components and interactions are considered in the definition of this process.

3.3. IoTA-SRM Process

The IoTA-SRM framework is designed to guide IoT security risk management, founded on integrating several core elements: IS assets, business assets, vulnerabilities, threats, risks, security requirements and security control measures. These elements collectively define the architecture and dynamics of security within the IoT ecosystem. In this section, the process is formalised by the framework, which is structured into four core activities: model the system, identify risks, manage risks, and assess trade-offs. These activities are guided by guidelines and domain-specific outputs, as demonstrated in Figure 11.

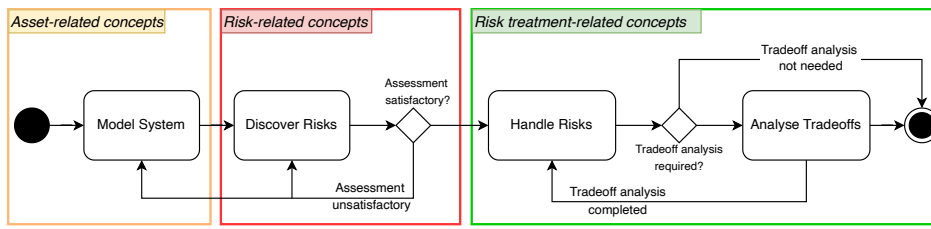


Figure 11. Core processes of IoTA-SRM framework

When applying each framework activity, documentation is essential. A detailed documentation strategy should be defined, outlining what elements should be documented (e.g., asset lists, interactions, security objectives) and how they are documented (e.g., in a structured registry or free-form text). Such documentation must be maintained and regularly updated during the SRM activity, ensuring accuracy and alignment with the IoT system and its requirements. It's important to emphasise the iterative nature of the IoTA-SRM framework. As IoT environments evolve, new risks emerge, and business priorities shift, it might necessitate revisiting and re-evaluating earlier activities. This cyclical approach ensures that risk management stays adaptive and responsive to the dynamic IoT ecosystem. The culmination of this process is a prioritised list that aligns with the IoT system goals and constraints, ensuring that the most critical risks are addressed resource-efficiently. This activity ends the process, integrating all prior activities and providing a clear and actionable path for SRM in the IoT system.

These activities are, therefore, detailed in the subsequent sections, concerning the ongoing example of an AV system (introduced in Section 2.1.3) scenario, considering a perspective based on a three-layer architecture (perception, network, and application layers). In this example, the interaction between a GPS component (IS asset) situated at the perception layer and components at the application layer through the network layer (and its constituent elements) is highlighted [190].

3.3.1. Model System

The model system activity provides an overview of the assets and their interactions that hold business value in the IoT domain (see Table 15). This activity serves as input for threat and risk elicitation by decomposing the IoT system into its constituent parts and components. IoT architecture decomposition is vital to this process since our security risk management analysis is rooted in the architecture perspective. When an IoT system is decomposed into its architectural layers, the system can be further analysed by its components, and the technical interactions between IS and business assets in each architectural layer can be described to achieve IoT application business service. Once the business assets and their corresponding IS assets have been identified, the required level of protection in terms of confidentiality, integrity, and availability (CIA) for the business assets can be determined. In addition, various modelling languages can be used to illus-

Table 15. Model system activity tasks and outcomes

Activity	Activity Tasks	Outcome Artefacts
Model System	Decompose IoT system into IoT layers	-
	Identify IS and business assets for each IoT layer	Asset list
	Define security objectives for business assets per IoT layer	Security objectives
	Model decomposed system	Asset model

trate the IoT IS assets and interactions. Some popular examples include Unified Modeling Language (UML) [191], Systems Modeling Language (SysML) [192], and Business Process Model and Notation (BPMN) [193]. The documentation resulting from this activity encompasses a comprehensive asset list, a system model that highlights the interactions between the assets in the IoT system, and security objectives indicating the significance of the business assets.

This activity can be illustrated using our ongoing example in Figure 12. Com-

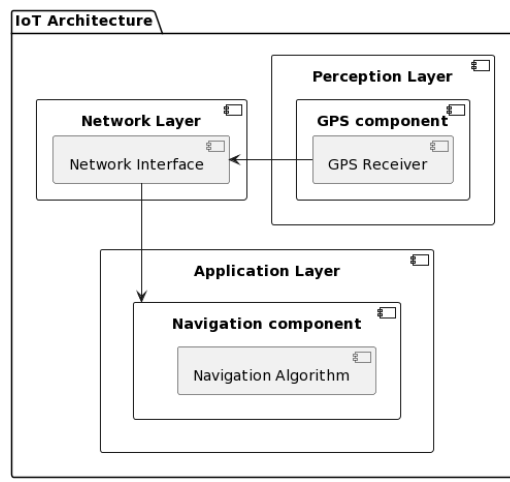


Figure 12. Running example: High-level system decomposition

mencing with the high-level structure of the IoT architecture, the three-layer architecture, encompassing the perception, network, and application layers, is utilised. Identifying IS assets at each layer is facilitated by scrutinising their capabilities, and sub-element relationships are established.

The capabilities of the IoT system were examined to identify the business assets supported by the IS assets at each layer. Business assets, including data elements, functions, and data flows, were defined as assets that create business value and are supported by IS assets. For instance, signals (data element) are collected by the GPS component from GPS satellites to compute the device's location (function), and this information can be transmitted through the network layer to the AV navigation component (data flow) at the application layer, contributing to autonomous navigation. This interaction is illustrated in Figure 13.

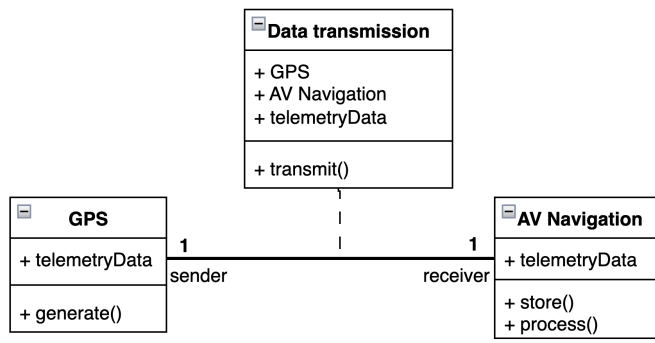


Figure 13. Running example: GPS meta-model constructs

Security objectives are established for each layer to guarantee the security of business assets. In the example scenario, the *integrity* criterion is defined as a security objective for the business asset, ensuring the integrity of the location data collected by the GPS component during transmission. A score of *low*, *medium*, or *high* can be assigned to the criterion based on the system's functionality. In addition to the *integrity* criterion, the *confidentiality* and *availability* security criteria can also be defined for the business assets identified in Table 16.

Table 16. Business assets and security objectives supported by IS assets at each layer of the IoT system

Layer	Business Assets	Supporting IS Assets	Security Criteria	Priority Score
Perception	Data element: GPS signals	GPS receiver sub-component	Integrity	High
	Function: Location computation	GPS receiver sub-component	Integrity	High
Network	Data flow: Transmission of location data	Network component	Integrity, Confidentiality	High, Medium
	Function: Data processing	Network interface sub-component	Integrity	High
Application	Data element: Location data	AV navigation component	Availability	High
	Function: Navigation computation	Navigation algorithm sub-component	Availability	High

3.3.2. Discover Risks

In the discover risks activity, potential security threats could exploit vulnerabilities in the IoT system's components and cause harm to its IS and business assets. This activity involves employing adversarial thinking and creativity to conduct a comprehensive analysis of vulnerabilities and threat modelling from the perspective of a malicious actor.

To perform vulnerability analysis, resources such as vulnerability databases [194, 195, 196] can be utilised. Vulnerabilities can be identified at each layer of the IoT system and within specific components. For example, an *IS asset* like a GPS tracker may have a broken authentication vulnerability (CVE-2022-2141)[2], enabling it to execute SMS-based GPS commands without authentication. For each component identified with a vulnerability, threat analysis is conducted using the STRIDE[3] method to categorise and analyse security threats at each IoT system layer. For instance, at the *perception layer*, the positioning component, a broken authentication vulnerability in the GPS *IS asset* can lead to spoofing threats, where an attacker can send commands while impersonating a legitimate entity. While the STRIDE method is advocated for based on the advantages observed in our threat analysis, as described in Section 2.2.2, other suitable threat modelling methods may also be applied [197]. Thus, the security risk can be defined as the likelihood of a risk event occurring (where the risk event is an aggregation of threats exploiting one or more vulnerabilities) and the risk impact (negative consequences of the risk event).

> Security risk, $R = R_l$ x R_i, where R_l is the risk event likelihood, and R_i is the risk event impact.

While we would use qualitative methods to estimate risk for decision-making, incorporating quantitative metrics would enhance the analysis.

Security risks in each layer can arise from a *threat* within the same layer or from other layers, resulting in a ripple effect of multi-layer risk impacts. For example, an unauthorised GPS command originating from the GPS *IS asset* in the *perception layer* can compromise the analysis of legitimate location data on the computing *IS asset* in the application layer, which can pose a significant security risk, particularly in critical applications like emergency response [19]. These concepts are illustrated in Table 17. The documentation resulting from this activity includes vulnerability and threat lists, as well as information on the impact of risks (see Table 18).

3.3.3. Handle Risks

The handle risks activity addresses the security risks identified in the *Discover Risks* activity. Risk-handling decisions are made based on the identified risks and

[2]https://nvd.nist.gov/vuln/detail/CVE-2022-2141

[3]STRIDE (Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege) [189]

Table 17. GPS Example in the discover risks activity

Layer	Perception
Component (IS Asset)	GPS (Global Positioning System) tracker
Vulnerability	Broken authentication (CVE-2022-2141)
Description	The GPS tracker lacks proper authentication, allowing it to execute SMS-based GPS commands without requiring authentication.
Threats	Spoofing (S)
Impact	• Tampered GPS location data • Unauthorised access to GPS location data • Compromised actions of GPS tracker and connected location-based services
Risk	**SR1**: An attacker sends commands to the GPS while impersonating a legitimate entity, leading to unauthorised access to the GPS tracker, loss of confidentiality of location data, loss of integrity/falsification of GPS location data and unauthorised actions performed by the attacker.

Table 18. Discover risks activity tasks and outcomes

Activity	Activity Tasks	Outcome Artefacts
Discover Risks	Multi-layer vulnerability assessment	Vulnerability list
	Multi-layer threat elicitation	Threat list
	Multi-layer risk impact estimation	Risk impact information

expressed in decision terms (avoidance, reduction, transfer, retention). When a decision is made to reduce risk, *security requirements* are defined to guide the implementation of controls to mitigate the risks. The activity also involves deriving effective remediation plans based on the identified risks and *security requirements*. To systematically elicit risk-based security requirements, the STRIDE method is used. It aligns specific security requirements with each type of threat: Spoofing (Authentication), Tampering (Integrity), Repudiation (Non-repudiation), Information Disclosure (Confidentiality), Denial of Service (Availability), and Elevation of privilege (Authorisation) [122]. However, other suitable security requirement elicitation methods may also be applied [198].

In the case of the GPS example, an *authentication* security requirement would prevent an attacker from gaining unauthorised access to the GPS *IS asset* and falsifying location data. Lastly, *security controls* are implemented to meet the defined security requirements and address the identified security risks. Control selection is guided and validated through relevant security frameworks and best practices recommendations [199, 200]. For instance, a strong password-based authentication control should be implemented for the GPS device *IS asset* to restrict access to its functions [199]. These concepts are illustrated in Table 19. The documentation resulting from this activity are the security requirements to secure the system against the discovered risks and suggested controls to treat the risk (see Table 20).

Table 19. GPS Example in the handle risks activity

Layer	Perception
Component (IS Asset)	GPS (Global Positioning System) tracker
Security Risk	**SR1**: Unauthorised access and falsification of GPS location data
Treatment Decision	Risk reduction
Security Requirement	Authentication
Description	An authentication security requirement is defined to prevent unauthorised access to the GPS IS asset and falsification of tracking data.
Security Control	Strong password-based authentication control for the GPS device IS asset

Table 20. Handle risks activity tasks and outcomes

Activity	Activity Tasks	Outcome Artefacts
Handle Risks	Multi-layer risk treatment decision	Risk decision
	Security requirements elicitation	Security requirements
	Control selection	Selected controls list
	Control implementation (can follow *Analyse Tradeoffs* outcome)	-

3.3.4. Analyse Trade-offs

The required effort to respond to risk and implement the control for risk reduction decisions will likely exceed available resources. Hence, a risk trade-off analysis is required. A security metric and the trade-off analysis procedure are introduced to tackle resource management for security risk treatment. It's crucial to recognise that the trade-off analysis is an iterative process. As more information about the system becomes available, risks evolve, or the effectiveness of controls changes, the trade-off decisions may need revisiting. Periodically reviewing and updating the analysis ensures that the decisions remain optimal in light of the evolving risk landscape. A security metric is a quantitative or qualitative measurement used to assess various security aspects, such as the effectiveness of controls, the impact of risks, or the value of assets. It provides a standardised way to evaluate security-related factors and aid decision-making processes [201, 118]. These are typically quantitative or semi-quantitative methods that involve a more mathematical treatment of risk and cost variables. Various methods are available for conducting trade-off analysis in security risk management. Factor Analysis of Information Risk (FAIR) [202] and Harmonized Threat and Risk Assessment (TRA) Methodology [203] provide structured approaches for risk assessment. To balance the trade-offs between risk and control costs, Cost-Benefit Analysis (CBA) calculates net benefits using metrics such as Return on Security Investment (ROSI) [204, 205]. Analytic Hierarchy Process (AHP) offers a systematic framework for comparing criteria like cost and risk, in a pairwise manner [206, 207]. Monte Carlo Simulation models various scenarios to tackle complexities and variabilities in risk assessment, although it is computationally intensive [208]. Bayesian Networks can also be utilised for probabilistic risk modeling [209].

For example, a cost-benefit trade-off analysis can be conducted to select the

most suitable control by considering the following security metrics and employing appropriate methods. Following the ISSRM metric concepts introduced by Mayer *et al.* [210] to derive security metrics from each IoTA-SRM activity:

- **Model System**: In this activity, the *business asset - location data* is assigned a metric value based on its importance or criticality. These metric values can serve as a guide for the trade-off analysis.
- **Discover Risks**: This activity identifies and evaluates risks associated with the vulnerable components. The impact of each risk, considering factors such as potential harm, consequences, or potential losses, can be estimated. A risk score is then assigned based on the likelihood and impact of an attack at each layer.
- **Handle Risks**: This activity would have proposed specific controls to mitigate or reduce the identified risks. Hence, the cost of implementing any considered control can be determined. This can include the initial implementation cost and recurring maintenance, updates, or operational expenses. Each control would also have an associated risk reduction or control effectiveness level, which can be expressed qualitatively, such as high, moderate, or low, indicating the extent to which the control mitigates the risk [210, 112].

In the *Tradeoff Analysis* activity, calculations can be performed to estimate the risk reduction levels considering the risk event's potentiality, the risk impact level, and the overall risk level metrics. For instance, this can involve determining the cost of controls, assessing the value of the assets, evaluating the likelihood of threats exploiting vulnerabilities at each layer, and analysing the potential impact of attacks at each layer. These security metrics provide valuable insights for the cost-benefit trade-off analysis and aid in selecting the most appropriate controls. The documentation resulting from this activity is the prioritised risks whose selected controls will be implemented to secure the system (see Table 21). Furthermore, the documentation of the outcome of this activity is encouraged.

Table 21. Analyse trade-offs activity tasks and outcomes

Activity	Activity Tasks	Outcome Artefacts
Analyse Tradeoffs	Determine asset values from *Model Risk* activity	Asset metric values
	Estimate risk impact values from *Discover Risks* activity	Risk impact metric values
	Estimate selected controls costs from *Handle Risks* activity	Control cost metric value
	Run cost vs benefit analysis for risk reduction	Prioritised risk list

3.4. Framework Validation: Case Analysis

We apply and demonstrate the proposed framework in two autonomous vehicle (AV) cases. AVs, a.k.a. self-driving cars, perceive, collect, generate and disseminate data within their IoT layers to improve knowledge to act autonomously and provide the required mobility, safety, and comfort services to the human components of its ecosystem. AVs generate sensitive data about customers, including

where and when the passenger is using the car, and collect and disseminate data about the environment and objects like obstacles and traffic signs, which is essential for self-driving. AVs are discussed in Section 2.1.3.

For our case studies (in Section 3.4.1 and Section 3.4.2), the AV is a Lexus RX450h autonomous driving vehicle designed to support AutonomouStuff [4] and used in laboratory settings (details of the laboratory set-up in Appendix B). The AV is part of an autonomous driving project [211], founded in 2019 in cooperation with the Estonian mobility company Bolt. A cross-section of key stakeholders involved in this project was involved in validating our framework application cases. The laboratory setting was designed to mimic real-world conditions closely. While physical tests were not conducted on the vehicle, the AV architecture was derived and scrutinised through discussions with the AV stakeholders and our physical observations of the AV. This process enhances the reliability of our findings, making them less artificial and more applicable to real-world AV ecosystems. It is worth noting that the AV, in this case, is still in the early development phase; thus, the significance of the security risks and controls might change over time.

Our first case in Section 3.4.2 describes the application of our framework to the overall AV system, guided by existing AV-related threats in literature. Our second case in Section 3.4.2 describes an iterative application of our framework on a smaller scope and a pilot feature to be added to the previously assessed AV system. To validate the results obtained at the end of the case analysis, a deliberate selection was made of three experts representing a cross-section of stakeholders and individuals deeply engaged in the AV project. This group included the technical lead on the project, a security specialist from the project partner company, Bolt, and the dedicated project coordinator with extensive business knowledge of the project. After each case, reflections on the framework's application within these case studies were documented, including the formalisation of the framework for broader application through expert validation.

3.4.1. Case 1: Autonomous Vehicle

In our initial case, our framework was applied to establish a multi-layer security risk management approach for the AV within our laboratory setting. Our scope here is directed by basic information provided by the autonomous driving lab[5] about assets in the AV system.

Model System. The AV system can be decomposed into perception, network and application layers, with business assets and their supporting IS assets at the architecture layers illustrated in Table 22. Note that all vehicle components support *autonomous driving* as a service. *Perception layer* includes the system components responsible for collecting business asset data (i.e., *video data, picture data*).

[4] https://autonomoustuff.com/products/astuff-automotive

[5] https://www.cs.ut.ee/en/autonomous-driving-lab

Table 22. Case 1 - Model system: Business assets and their supporting IS assets adapted from [65]

Layer	Business asset	Description	Supporting IS assets
Perception	Video data	Video from surrounding environment	Mako G, Sekonix cameras
	Picture data	Pictures from the surrounding environment	Mako G, Sekonix cameras
	Vehicle location data	Current location of the vehicle	PwrPak7 GPS, IGM-S1 IMU
	Vehicle travel data	Routes used with the time	PwrPak7 GPS, IGM-S1 IMU
	Working vehicle data	Vehicle speed, direction etc.	PwrPak7 GPS, IGM-S1 IMU
	Ultrasonic sensors data	Ultrasonic data about surroundings	Ultrasonic sensors
	Radar data	Data from radar	Delphi ESR 24V
	Surrounding environment data	Data about the surrounding environment and objects	VLP32 LiDAR, Delphi ESR 24V, Ultrasonic sensors
	Inertial measurements	Vehicle speed, angle, location	IGM-S1 IMU
Network	Perception data	Data and messages exchanged by different components	Network
Application	Map data	Map used for autonomous driving	Map storage
	Fused data	Combined data from the perception layer	Perception layer, network, Spectra computer
	Computing data	Results from analysing the fused data	Spectra computer
	Actuation commands data	Commands generated to be sent to actuation module	Spectra computer, PACMod v3.0, ECU
	Decision maker	Software for making driving decisions	Spectra computer
	Driving planner	Software for planning out the route used	Spectra computer
	System software	All software used for autonomous driving	Spectra computer
All	Autonomous driving	Overall process of vehicle self-driving	All components

After collection, the *perception data* is transmitted through the *network layer* to the application layers. Lastly, the *application layer* uses the collected data to perform tasks, i.e., calculate routes, whereas an actuation module uses the results of route calculations to perform autonomous functions. We illustrate the asset relationships in Figure 14 showing the interactions between layer components in the AV system. The business value of each business asset can be estimated based on the impact of the loss of *confidentiality*, *integrity*, or *availability* security criteria on the system. Confidentiality – protecting data from unauthorised access; Integrity – ensuring the data and service remains unaltered and genuine; Availability – ensuring data and services are accessible by authorised users.

Discover Risks. A literature study of known threats to AV systems was undertaken, as documented in prior research [97, 212, 213, 214]. This study provided an understanding of the threat landscape for AV systems, and within our AV system context, specific threats were identified. We identified and mapped threats guided by system stakeholders with knowledge of the AV system. The selected threats were analysed and validated in collaboration with system stakeholders with expertise in the AV system. This allowed us to analyse the selected threats comprehensively, ensuring our findings were academically sound and pertinent to our particular AV system. Stakeholder validation further reinforced the relevance and robustness of our selections. As shown in Table 23, our selection comprises nine (9) threats at the perception layer, four (4) threats at the network layer, and seven

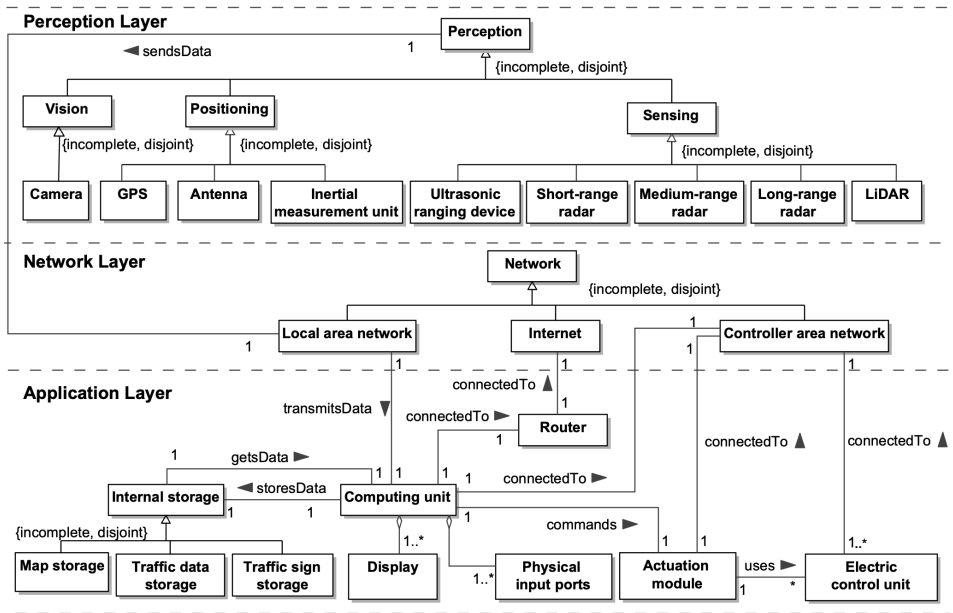


Figure 14. Case 1 - Model system activity: IS assets adapted from [65]

(7) threats at the application layer. Additionally, some threats span multiple layers. T17 is found across all layers, and T18 is present at both network and application layers.

We then analysed these identified threats using our framework, understanding the potential risks that could arise from successful threat exploitation. An example of this analysis is documented in Table 24, illustrating the risk analysis of **T6** – a blinding attack on AV cameras. Here, *an attacker with some expertise and tools sends malicious optical inputs* targeting the AV *cameras* (IS asset) because the *cameras are vulnerable to blinding attacks*. If the risk event occurs, it negates the *integrity of the video and picture data*, leading to unreliable data sensed by the cameras that could provoke wrong decisions when the car is driving/steering.

For each component listed in Table 23, an equivalent analysis can be conducted regarding the associated threats to create a comprehensive risk document. The OCTAVE Allegro worksheets [112] were subsequently employed in our framework due to their capacity for formal documentation and risk estimation, which relies on risk scores determined by the relative impact scores on the affected assets and the likelihood of the threats. The OCTAVE Allegro templates used are documented in Appendix C. Consequently, the impact of each security risk was assessed and documented following the OCTAVE approach. This assessment is elaborated upon in the *Analyse Tradeoffs* activity.

The risk identified as **R6** had its risk impact across multiple AV architecture layers. Risk impacts at the perception layer can trigger a cascade of risks affecting the network and application layers, each contributing to a potential risk escalation

Table 23. Case 1 - Discover risks: Threat list and their associated assets adapted from [65]

ID	Security Threat	Associated Components (IS assets)	Associated Business asset	Affected Layer
T1	Jamming ultrasonic sensors	Ultrasonic ranging devices	Surrounding environment data	Perception
T2	Spoofing ultrasound sensors	Ultrasonic ranging devices	Surrounding environment data	Perception
T3	Acoustic quieting on ultrasound sensors	Ultrasonic ranging devices	Surrounding environment data	Perception
T4	Jamming radar	Delphi ESR 24V radar	Surrounding environment data	Perception
T5	Spoofing radar	Delphi ESR 24V radar	Surrounding environment data	Perception
T6	Blinding cameras	Mako G, Sekonix cameras	Video and Image data	Perception
T7	Confusing car controls using camera inputs	Mako G, Sekonix cameras	Video and Image data	Perception
T8	Relay attack on LiDAR	VLP32 LiDAR	Surrounding environment data	Perception
T9	Spoofing LiDAR	VLP32 LiDAR	Surrounding environment data	Perception
T10	Code modification	ECU, Spectra computer, Software repository	System software	Application
T11	Code injection	ECU, Spectra computer	System software, All perception data	Application
T12	Packet sniffing	Network components (4G)	All perception data	Network
T13	Packet fuzzing	Network components (4G)	All perception data	Network
T14	Inject CAN messages	Controller area network (CAN)	All perception data	Network
T15	Eavesdropping CAN messages	Controller area network (CAN)	All perception data	Network
T16	GPS jamming and spoofing	PwrPak7 GPS	Location data	Perception
T17	EMP attack	All parts	Autonomous driving service	All
T18	Malware injection	Spectra computer	Autonomous driving service	Perception, Application
T19	Manipulate map data	Map storage	Map data	Application
T20	Extract map data	Map storage	Map data	Application
T21	Delete map data	Map storage	Map data	Application
T22	Disable actuation module	PACMod v3.0	Autonomous driving service	Application
T23	Induce bad analysis	Spectra computer	Autonomous driving service	Application

in the overall AV system (see Figure 15). For example, the initial risk impact of **R6** starts at the perception layer - "*Sensor unreliability causes an incorrect perception of the environment*". This is crucial, as perception is foundational for the subsequent functioning of an AV system. The incorrect perception, in turn, compromises the quality of video and picture data collected for the AV's operation. If the perception layer is compromised, data integrity at the network layer becomes questionable. The network layer can transmit corrupted or incomplete data packets across the network. In addition to the disrupted data transmission, there may be an onset of network congestion. This can be due to redundant or excessive data generated by the erroneous perception layer or the result of a targeted attack. These risk impacts from the perception and network layers cascade into the ap-

Table 24. Case 1 - Discover risks: Risk analysis, documented using OCTAVE sheets, adapted from [65]

Allegro – Worksheet 10		**Blinding attack on cameras**							
Threat	**IoT Layer Affected**	Perception							
	Business Asset	Video and picture data							
	Business Asset's Value	Medium – Car can continue driving but can't recognize signs and traffic lights.							
	Area of Concern	An attacker uses their tools to send malicious optical data to the camera, causing unwanted blindness, possible hardware damage and loss of integrity of the video and picture data.							
	Actor Who would exploit the area of concern or threat?	An attacker with some previous experience and tools to send malicious optical inputs (laser etc.).							
	Means How would the actor do it? What would they do?	An attacker uses their knowledge and malicious optical emitters to send and blind cameras, causing unwanted blindness on the cameras and possibly permanently damage the camera sensors.							
	Motive What is the actor's reason for doing it?	Wants to see the car crash and make the company lose its reputation.							
	Outcome (choose one) What would be the resulting effect be?	Disclosure:		Destruction:		Modification:		Interruption:	x
	Security Requirements How would the information asset's security requirements be breached?	Availability and integrity of the AV's visual systems (camera), including automated responses to threats.							
	Likelihood (choose one)	High:	x	Medium:		Low:			

Consequences What are the consequences to the organisation as a result of the risk?	**Severity** How severe are the consequences to the organisation or asset owner by impact area? *3 for highest priority, 2 for medium, 1 for lowest			
1. Blinding attack causing some blind spots on the image recorded by the cameras. 2. Blind spots limit object detection, thus, causing accidents. 3. Sensor unreliability causes an incorrect perception of the environment. 4. Using lasers to carry out the attack can permanently damage the camera's lens.	**Impact area**	**Priority***	**Impact**	**Score**
	Confidentiality	1	Low	1
	Availability	3	High	9
	Integrity	2	High	6
	Relative risk score:			16
	Total Risk Score (Rel x likelihood):			48

plication layer. Algorithms are highly sensitive to the quality and timeliness of data. Incomplete or delayed data can impair the algorithm's performance and lead to suboptimal or hazardous decisions. A compromised perception and network layer could yield incomplete or corrupted data to the decision-making algorithms, disrupting the AV's ability to make safe and effective decisions. In a worst-case scenario, this could lead to accidents or catastrophic failures.

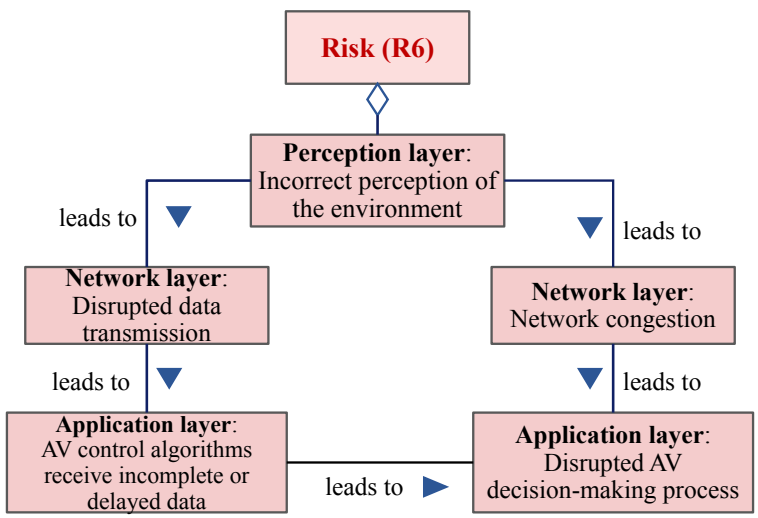


Figure 15. Risk impact of **R6** across architecture layers

Handle Risks. Table 25 suggest controls to each security risk discovered. For example, to mitigate the example risk **R6**, two controls are suggested based on industry best practices [199], literature recommendations [66]: (*i*) multiple sensors for redundancy check, e.g. Overlapping image output with multiple cameras, (*iii*) filter to remove harmful light input, and (*iii*) turn off the auto exposure. Figure 16 also shows the interactions between the proposed controls to security risks at the perception, network and application layers and the existing components to mitigate security risks.

Table 25. Case 1 - Handle Risks: controls adapted from [65]

ID	Security Risks	Layer Affected	Suggested control	Risk score
R1	Jamming ultrasonic sensors	Perception	Noise detection and rejection; Multiple sensors for redundancy check	32
R2	Spoofing ultrasonic sensors	Perception	Noise detection and rejection; Multiple sensors for redundancy check	32
R3	Acoustic quieting on ultrasound sensors	Perception	Multiple sensors for redundancy check	12
R4	Jamming radar	Perception	Noise detection and rejection, Multiple sensors for redundancy check	32
R5	Spoofing radar	Perception	Noise detection and rejection, Multiple sensors for redundancy check	32
R6	Blinding cameras	Perception	Overlapping image output with multiple cameras, Filter to remove harmful light, Turn off auto exposure	48
R7	Confusing car controls using camera inputs	Perception	Overlapping image output with multiple cameras, Filter to remove harmful light, Multiple sensors for redundancy check	32
R8	Relay attack on LiDAR	Perception	Multiple LiDAR inputs, Random probing, Shorten pulse period	16
R9	Spoofing LiDAR	Perception	Multiple LiDAR inputs, Random probing, Shorten pulse period	14
R10	Code modification	Application	Device authentication, Anti-malware, Isolation, Unit tests, Manual code checks	17
R11	Code injection	Application	Device authentication, Anti-malware, Isolation	18
R12	Packet sniffing	Network	Encryption, Device authentication, User authentication	24
R13	Packet fuzzing	Network	Encryption, Device authentication, User authentication, Secure connection, Network isolation	15
R14	Inject CAN messages	Network	Encryption, Device authentication, User authentication	12
R15	Eavesdropping CAN messages	Network	Encryption, Device authentication, User authentication	15
R16	GPS jamming and spoofing	Perception	Nullification, Monitoring signals and identification nodes, GPS data duplication, LiDAR for localisation	32
R17	EMP attack	All	Isolation	12
R18	Malware injection	Perception, Application	Install firewall, Anti-malware, Isolation	36
R19	Manipulate map data	Application	Isolation, Device authentication, User authentication, Data duplication, Testing map data in simulations	12
R20	Extract map data	Application	Isolation, Device authentication, User authentication	12
R21	Delete map data	Application	Isolation, Device authentication, User authentication	12
R22	Disable actuation module	Application	Isolation, Device authentication, User authentication	14
R23	Induce bad analysis	Application	Isolation, Access Control	14

Analyse Tradeoffs. We analyse tradeoffs to respond to risk and make risk mitigation decisions to implement controls for risk reduction. This analysis was based

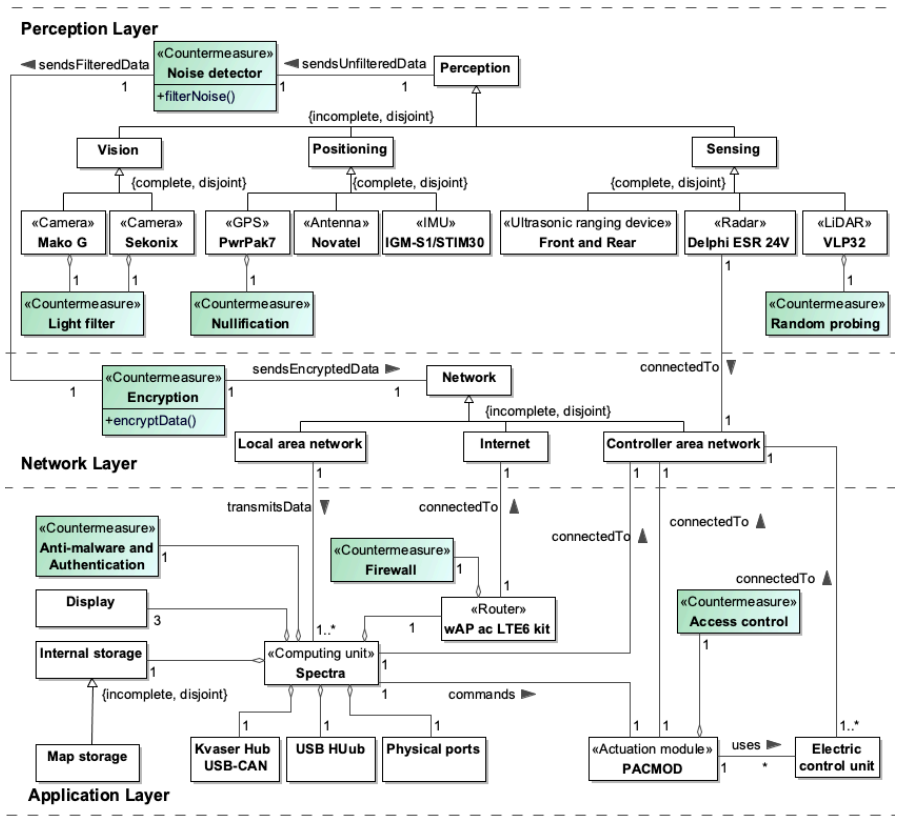


Figure 16. Case 1 - Handle risks: Controls adapted from [65]

on estimates of asset values from the *Model Risk* activity, risk impact values from the *Discover Risks* activity, and control costs from the *Handle Risks* activity, following the OCTAVE approach. The business value of an asset is estimated based on the value it provides to the system. The main concern is to explain what happens if the data is lost or modified. The *low* score is assigned if the system can stay operational without this asset; *medium* - if the system can continue, but there exist some performance issues; and *high* - if the system becomes not operational. Similarly, the threat *likelihood* and *impact* for each security risk can be estimated. The threat likelihood is estimated as *low* if the needed means to perform the attack method are specific, their cost is high, the required knowledge to perform the attack method is high, and the possibility to carry on the attack method requires much time. The likelihood is estimated as *high* if it is easy to obtain the means for executing the attack method; not much knowledge and preparation is required. An example of the risk estimation is given in Table 24 with risk scores for identified risks illustrated in Table 25.

For risk mitigation, addressing risk **R6** was determined as effective in a low-medium cost scenario. The risk score 48, calculated based on impact, likelihood, and security objectives, indicates a significant threat level. This makes

low-medium cost controls favourable under budget constraints, due to its risk reduction level. The importance of **R6**'s risk score becomes evident when compared with other risks in a trade-off analysis (see Table 25). Each control option listed in Table 23 is assessed within the constraints of stakeholder budgets to decide based on the potential risk reduction and cost of implementing the control. To address **R6** controls such as multiple sensors with different cost profiles were suggested for redundancy and data validation. Other low to moderate-cost options include light filters and disabling the auto-exposure feature (see Table 26). These control costs were justified when balanced against the risk mitigation benefits, confirmed through stakeholder discussions.

Table 26. Case 1 - Handle risks: R3 mitigation estimation

Risk Mitigation	**R3: Blinding cameras**							
Choose action to take.	Accept:		Defer:		Mitigate:	x	Transfer:	
For the risk, what actions and controls will be used: **Security requirements:** (*i*) The AV shall provide a minimum of 98% visual coverage to prevent single point of failure. (*ii*) The AV cameras must not degrade image quality below the recommended threshold when exposed to optical attacks. (*iii*) The AV camera shall disable auto exposure settings when exposure to high-intensity light sources is detected.								
Layer where applied	**Description of control or action**					**Estimated cost**		
Perception	Image output (overlapping) with multiple cameras					Moderate - High		
Perception	Filter to remove harmful light					Low - Moderate		
Perception	Turn off auto exposure					Low		

Validation. We applied the IoTA-SRM framework to analyze the AV, focusing on its IoT layers. We interviewed stakeholders to validate the security threats, risks, and control measures elicited, crucial to our validation process. We engaged with three experts closely connected to the autonomous driving project and with knowledge of the AV in our case study.

1. Expert 1 was the technical lead on the AV project. Expert 1 confirmed the presence of ultrasonic sensors in the car (R1, R2, R3) and indicated their potential future applications like automated parking, although the sensor was inactive. Expert 1 also assessed other risks, such as those related to cameras (R6, R7), LiDAR (R8, R9), code modification (R10), packet fuzzing (R13), and GPS jamming (R16). The assessment enriched the understanding of asset importance, feasible attack methods, and security controls.
2. Expert 2 was an information security manager with over 15 years of experience, focusing on the broader security landscape during our interviews. Expert 2 contributed to discussions concerning risks related to radar (R4, R5), cameras (R6, R7), code modification (R10), packet fuzzing (R13), GPS jamming (R16), and EMP attacks (R17).
3. Expert 3 was the project coordinator and provided insights primarily from a managerial and business perspective about our resulting risk analysis. While not directly contributing to the technical aspects, Expert 3 evaluated the proposed risk impacts and discussed how the risk assessment framework outcomes align with the AV project's future objectives.

Thus, the experts provided valuable insights regarding the framework. They reported how well the framework highlighted the potential complexities in understanding OCTAVE templates and recognising their significance. They proposed the inclusion of a concise overview table to simplify comprehension (as updated in the work). These interviews provided technical and business viewpoints to validate our framework, offering a deeper understanding of how the experts perceived our approach. Their perspectives encompassed Although we identified potential threats to validity, such as subjective opinions and the completeness of our questions, the overall support from the experts indicates alignment with the framework's utility. The absence of discussions about alternative security risk management methods could be considered a potential concern, as it may suggest a bias towards the current approach. However, given the experts' endorsement of the existing method, this concern may not be substantial. Overall, the experts' evaluations corroborated the effectiveness and practicality of the IoTA-SRM framework in addressing security risks in AVs. This underscores its value in guiding comprehensive security risk assessments.

Lessons Learned. Key lessons have been derived from examining security risks across different IoT layers of the AV, revealing risk patterns and how they can be addressed. It was observed that security risks at the perception layer significantly impacted the system's *integrity* and *availability*. The risks in this layer highlighted the necessity of employing multiple input sources from diverse or identical sensors for error-checking and cross-validation of data validity. Within the network layer, security risks affected the *confidentiality* and *integrity* of transmitted data. This underscored the requirement for encryption, authentication methods, and access controls to protect data during transit. In the application layer, the identified risks had notable implications for the *integrity* of transmitted data and the *availability* of the autonomous driving service. This emphasizes the importance of input validation, anti-malware strategies, isolation controls, and authentication methods to safeguard data access and system functionality.

The IoTA-SRM framework offered a streamlined strategy for managing IoT security risks. Within this framework, elements from existing risk management methods that align with specific objectives can be incorporated by practitioners. In this case, the application of OCTAVE is instrumental in facilitating risk-based investment decisions concerning security in AVs for stakeholders. Qualitative evaluations of risks and controls were supported, with the utilisation of the OCTAVE Allegro worksheets [112] within our framework. These worksheets provided structured documentation and enabled risk estimation through relative impact and threat likelihood scores. Control cost estimation was also adopted per the approach outlined in [215], which employed quantitative probabilistic methods for control recommendations. However, these costs were transformed into qualitative values, reducing the metric data analysed while still positively influencing decisions regarding control selection. Although the OCTAVE Allegro worksheets were found to be beneficial, consideration could also be given to other methodolo-

gies, such as FAIR (Factor Analysis of Information Risk) [202] or the Harmonized Threat and Risk Assessment (TRA) Methodology [203].

Running a security analysis on the overall AV system can become time-consuming. So, to manage the analysis effectively, the scope was constrained to the static assets within the AV architecture, along with related threats documented in the existing literature. Static assets, in this study, refer to AV components not expected to undergo significant changes during the analysis period. On the other hand, dynamic assets refer to components that are under significant development and are frequently updated or might be replaced. It is important to note that this constrained scope was primarily adopted for testing purposes. It is not implied that a static-only analysis would suffice for a comprehensive security assessment of an entire AV ecosystem. However, our subsequent case analysis (refer to Section 3.4.2) extends this scope by applying the framework to a new feature, illustrating its adaptability. The feasibility of applying our framework in real-life scenarios is substantiated by its design, which aims to balance comprehensiveness and manageability. Our expert-validated outcomes indicate that our framework can be practically applied in real-world environments.

3.4.2. Case 2: Autonomous Traffic Light System

In our second case, our framework was applied to a pilot feature – an MQTT autonomous traffic light system, enabling machine-to-machine (M2M) communication between the AV in laboratory settings[6] and the city traffic lights infrastructure. While the AV included camera sensors to detect the traffic light status, learning from the outcome of security analysis in Case 1 (see Section 3.4.1), the need for multiple sensors to validate inputs used by the system is significant. For example, obstructions to traffic lights or camera blinding attacks could prevent the image recognition algorithms from detecting the correct traffic light status, leading to severe accidents. As part of the IoT ecosystem, traffic light systems have been developed to send traffic light data via the Internet managed by the traffic management system for smart transportation applications, i.e. autonomous driving. Thus, AVs can now have an additional input source to cross-validate traffic light decisions for the autonomous driving service.

We analyse this pilot feature to ensure a secure and reliable perception and communication of traffic light statuses between the traffic light management system and the AV.

Model System. Traffic light systems have been developed to send traffic light data via the internet to the AV to support autonomous driving. This high-level interaction is illustrated in Figure 17. *Business assets* describe important data, processes, and capabilities essential to using MQTT to communicate with the traffic light. *IS assets*, on the other hand, support business assets. *Business* and *system* assets in our scope can be decomposed into network and application layers,

[6] https://www.cs.ut.ee/en/autonomous-driving-lab

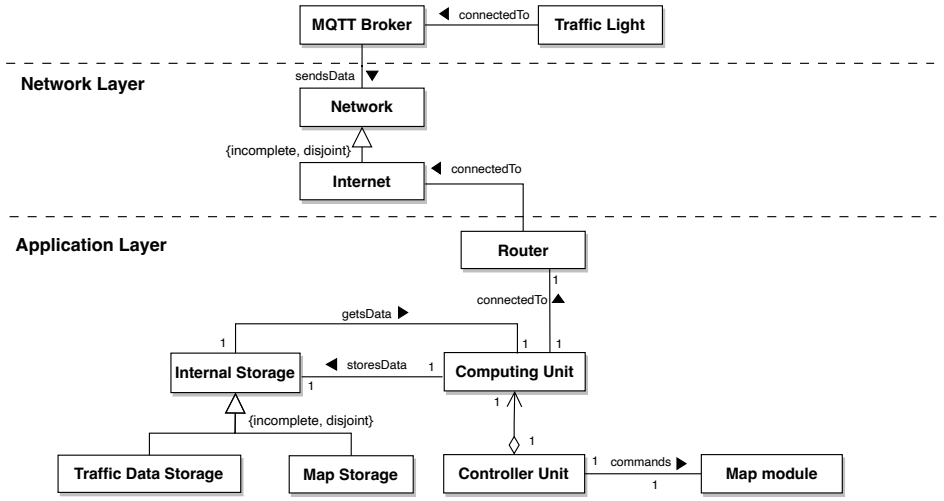


Figure 17. Case 2 - Model system activity: IS assets

illustrated in Table 27.

Table 27. Case 2 - Model system: Business assets and their supporting IS assets

Layer	Business asset	Description	IS assets	CIA
Network	Traffic light status	Data from traffic light infrastructure	3G, MQTT	C, I
Application	Traffic light location	Notify AV controller of traffic light on path	AV map	I, A
	Client credentials	MQTT credentials to connect to broker	MQTT Broker, AV controller, Traffic light client	C, I
	Traffic light status message	Transmitted traffic light status from infrastructure to AV	MQTT Broker, AV controller, Traffic light client	I, A
	Traffic light status topic	Messaging queue for traffic light status messages	MQTT Broker, AV controller	I, A
	Traffic light ID	Identify traffic light and subscribe to corresponding topic	AV map component, Traffic light client	I, A
Both	Get traffic light status	Process of collecting traffic light data and making driving decisions	All components, Spectra computing unit	C, I, A

The business assets important to this case include traffic light messages such as the traffic light status, ID, location, topics on the MQTT broker and other processes that enable the purpose of this MQTT system. The IS assets include the AV controller, the traffic light, the MQTT broker and other components of the AV that support these business assets and the AV driving service. The AV controller is an MQTT client that uses paho-mqtt[7] supporting both MQTT 3.*x* and 5.0. The AV controller connects to the MQTT broker responsible for handling traffic light status topics. The AV controller also uses a special vector map containing data about the traffic lights on the AV's path, which lane it applies, and some meta-

[7] https://pypi.org/project/paho-mqtt/

data attributes (including the traffic light ID). In this system, the AV map module component notifies the AV controller that a traffic light is on its path and provides traffic light ID metadata data. Thus, after connecting to the MQTT broker, the AV uses the provided traffic light ID to subscribe, where the topic name is the traffic light ID. As different traffic lights come to the AV path, the map module notifies the AV controller, which dynamically subscribes and unsubscribes to/from traffic light topics. The traffic light is an MQTT client that connects to the MQTT broker over the network and publishes the traffic light status to the MQTT broker. The publisher thus queries the traffic light infrastructure every 0.1 seconds and publishes the provided data (traffic light's ID, current status and time in seconds from the last change) in JSON format. Lastly, the MQTT broker is a Mosquitto[8] version 1.4.15 MQTT broker that manages topics that are published by the traffic light MQTT client and subscribed by the AV MQTT client. The topic name is the traffic light ID and contains namespaces, e.g. */City/Street/000*.

The security criteria are expressed in Table 27 as *confidentiality* (**C**) – protecting data from being accessed by unauthorised parties, *integrity* (**I**) – ensuring that data or service is not altered or manipulated and that the data source is genuine and *available* (**A**) – that data or service is accessible by authorised users [118], of business assets.

Discover Risks. The MQTT implementation faces security risks as a result of highlighted threats in Table 28. Attacks against confidentiality target sensitive

Table 28. Case 2 - Discover risks: Threat list and their associated assets adapted from [66]

ID	Security Threat	Description	Components (IS assets)	Business asset	Affected Layer
T1	Identity spoofing [216]	Attacker obtains MQTT client credentials to impersonate and connect successfully to the broker.	AV controller, MQTT broker, Traffic light client	Client credentials, Get traffic light status and autonomous driving service	Application
T2	Malware/ botnet [217]	An attacker modifies the requested link to install malicious firmware in the victim devices.	AV controller, MQTT broker, Traffic light client	All business assets	Application, Network
T3	Flooding attack [218, 216]	Attacker sends a large number of connection requests, thereby flooding the broker.	MQTT broker, AV controller	Get traffic light status, Autonomous driving service	Network
T4	DoS attack [219, 220]	Attacker causes many connections with the server to seize all available connections.	MQTT broker, AV controller	Get traffic light status, Autonomous driving service	Network
T5	SYN flooding attack [216]	Attacker crafts a TCP-based attack to create multiple half-opened TCP sessions.	MQTT broker, AV controller	Get traffic light status, Autonomous driving service	Network
T6	Restricted topic access [216]	Attacker subscribes to restricted topics to eavesdrop on all messages.	MQTT broker, AV controller, Traffic light client	Get traffic light status, Autonomous driving service	Application

communications used in the *get traffic-light status* process and MQTT topics used by the system for autonomous driving [216]. Such sensitive data can be used for escalated attacks such as identity spoofing [216]. Attacks against integrity target data modification or destruction of the system as a malicious modification of one or more of the assets could lead to an inevitable failure or unreliability of this MQTT system. Malware attacks target not only the confidentiality of sensitive

[8] https://mosquitto.org/

data but its integrity as well [217]. Availability attacks target the timely and reliable access to and use of collected and generated data. The MQTT broker must be heavily robust against attacks because it handles requests from publishers (traffic light) and subscribers (AV controller) that should be reliable.

Handle Risks. With knowledge of the important security criterion for the business assets and the known risks and possible security controls in MQTT systems, certain design decisions and security requirements are necessary. The security requirements that are applicable to mitigate identified security risks to the system are presented in Table 29.

Table 29. Security requirements for risk mitigation adapted from [66]

Security Requirements
SR.1: The MQTT broker shall identify MQTT clients before allowing them access to its functions.
SR.2: The MQTT broker shall verify MQTT clients' identity before permitting them to use its functions.
SR.3: The MQTT system shall make data in communication between the MQTT broker and clients unreadable.
SR.4: The MQTT broker shall only allow the MQTT traffic light to publish messages to topics.
SR.5: The MQTT broker shall filter incoming data from the MQTT traffic light.
SR.6: The MQTT broker shall filter incoming data from the MQTT AV client.
SR.7: The MQTT broker shall verify the data received from the MQTT traffic light.
SR.8: The MQTT broker shall protect its functions from unauthorised changes.
SR.9: The MQTT broker shall prevent unauthorised corruption of its functions.
SR.10: The MQTT broker shall prevent unauthorised corruption of messages collected from authorised MQTT clients.
SR.11: The MQTT broker shall block abnormal requests from its clients.

Based on the security requirements, security controls can be implemented to mitigate risks illustrated in Table 30.

Table 30. Case 3 - Mapping risks, requirements, and controls

ID	**Security Risks**	**Layer Affected**	**Suggested Control**
R1	Identity spoofing	Application	SR.1, SR.2, SR.3: Digital certificates, Digital signatures, Pre-Shared Key (PSK), Client authentication, Transport layer security (TLS/SSL), Hashing algorithms
R2	Malware/ botnet	Application, Network	SR.8, SR.9: Access control lists, Disable uncontrolled features, Anomaly detection, Distributed MQTT brokers, Encrypt data stored on MQTT broker
R3	Flooding attack	Network	SR.11: Firewall, Application layer firewall, Rate limiting, Anomaly and Intrusion detection systems
R4	DoS attack	Network	SR.9, SR.11: Distributed MQTT brokers, Anomaly and Intrusion detection systems, Firewall policies, Application layer firewall, Rate limiting
R5	SYN flooding attack	Network	SR.11: Firewall policies, Application layer firewall, Rate limiting, Anomaly and Intrusion detection systems
R6	Restricted topic access	Application	SR.4: Access control lists

Analyse Tradeoffs. No tradeoff analysis was implemented for this small-scale study.

Validation. We provided the asset, risk and security requirements analysis to Expert 1, the AV stakeholder, crucial to the system, to guide system design and implementation. The AV stakeholder ensured TLS/SSL implementation to enable the MQTT broker and client to identify and authenticate each other and then com-

municate with confidentiality and data integrity (***SR.1-3***). Also, digital certificates by a certification authority on the MQTT broker enforce identity verification. Similarly, secure client authentication protects the MQTT broker from unauthorised access and malicious changes. The credentials can also be used for authentication to restrict access to published messages in topics as an access control measure. The stakeholder also configured access control lists (ACL) to restrict access to topics on the MQTT broker and provide user activity restrictions (***SR.4***). The implementation of security requirements ***SR.1-4*** as security controls is presented in Figure 3.4.2. As this feature implementation is in its pilot stage, the stakeholder

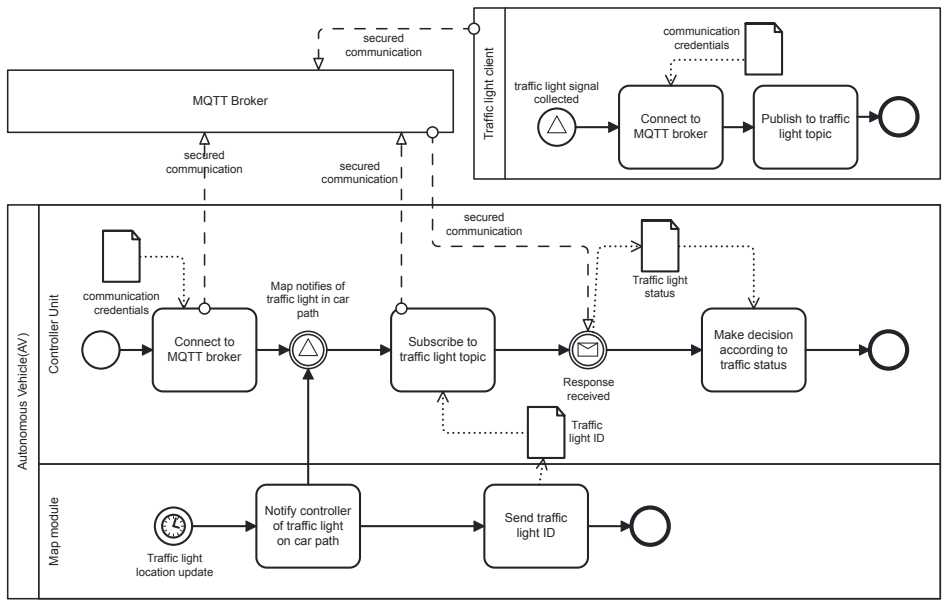


Figure 18. Case 2: Control implementation [66]

could not fulfil all the security requirements suggested. Controls to fulfil ***SR.5-11*** were only partly implemented. However, the requirements for future releases and design considerations were strongly considered when scaling the MQTT system for advanced traffic management systems. As controls to fulfil ***SR.5-11*** were only partly implemented, the stakeholder accepted the security risks that can arise from not implementing a full solution due to the current system scale and the frequency at which the system gets valid data instead of falsely injected data.

Lesson Learned. In this case, an analysis was conducted on a feature in development, specifically, an MQTT autonomous traffic light system comprising assets at the network and application layers of the AV system. We observed the framework's applicability to emerging features implemented on the AV system, noting that this framework could also be employed on a smaller scale and for iterative analyses. Risk estimation for tradeoff analysis was also not significant due to the feature's scope and the stakeholder's decisions to mitigate the risks or accept the

relatively low impact of the risks with the current scale of the feature implementation.

Applying the IoTA-SRM framework in ***Case 1*** (in Section 3.4.1) allowed for a multi-layer analysis of the AV system, which provided a systematic assessment of the system's security risks and helped stakeholders make informed decisions about risk mitigation. By applying the IoTA-SRM framework to ***Case 2***, stakeholders could identify potential security risks and make decisions about risk mitigation even though the system was smaller in scale. We also saw the importance of considering different types of security risks that can affect different system layers. In ***Case 1*** (see Section 3.4.1), security risks at the perception, network, and application layers of the AV system were identified and analysed. Similarly, in ***Case 2***, the network and application layers were analysed for security risks. By considering different system layers, stakeholders can decide which security controls to implement at each layer. Finally, ***Case 2*** learned from ***Case 1*** that the IoTA-SRM framework can be used with other security risk management methods to provide a more comprehensive analysis of IoT systems. In ***Case 1***, the OCTAVE Allegro method was integrated into the framework to provide formal documentation and risk estimation through risk scores. While this method may not be necessary for smaller IoT system scopes like in ***Case 2***, stakeholders can integrate other security risk management methods into the framework to provide a more comprehensive analysis.

3.5. Discussion

In this Chapter, we proposed the IoTA-SRM framework and applied the IoTA-SRM framework to two case studies related to autonomous vehicles (AV). This section summarises lessons learned from both case studies, discussing the implications of our research findings for IoT security risk management.

Firstly, the IoTA-SRM framework provided a systematic approach to security risk management in IoT systems. By decomposing the system into layered architecture and identifying the security risks at each layer, practitioners can discover the attack patterns within layers and pay attention to the cascading effects of risks from these layers. Each layer and component can undergo separate evaluation, allowing for a more detailed analysis of potential risks and their consequences. Consequently, this approach can lead to the development of more effective and robust risk mitigation strategies, ultimately enhancing the overall security of the IoT system. Results from applying the IoTA-SRM framework also serve as a basis for prioritising security controls based on the potential impact of risks at each layer while considering its cascading impacts. Such prioritisation ensures that resources are allocated to address high-impact security risks.

Secondly, applying existing risk management methods within the framework can provide useful output to stakeholders to make risk-based decisions in security return on investment analysis in IoT systems. For example, the AV case em-

ployed the OCTAVE Allegro method [112] to provide formal documentation and risk estimation through risk scores based on the relative impact scores on the affected assets and threat likelihood. Applying OCTAVE within this context also facilitated qualitative risk and control estimations. Table 31 presents methods and resources that can be incorporated to support the framework application.

Table 31. IoT security resources to IoTA-SRM activities

IoTA-SRM Activities	**Relevant Resources**
Model System	NIST SP 800-183 [221], IoTSF Best Practice [222], OWASP IoT Top Ten [223]
Discover Risks	NIST SP 800-30 [224], OWASP IoT Top Ten [223], CWE database [195], CVE database [225]
Handle Risks	CSA IoT Security Controls Framework [199]
Analyse Trade-offs	FAIR [202], TRA [203], ROSI [204], AHP [207], Monte carlo simulation [209]
Documentation	OCTAVE Allegro Worksheets [112]

Thirdly, the IoTA-SRM framework can be applied to new and developing features implemented on IoT systems, even on a smaller scale and for iterative analysis (see Section 3.4.2). The framework provides a lightweight approach to security risk management, allowing for a shorter feedback loop between security analysts and IoT stakeholders. Additionally, it is important to note that the scope of the analysis can affect the outcome of security risk management. While running a security analysis on the overall IoT system can be time-consuming, limiting the scope of the analysis to specific assets and related threats can still provide a systematic assessment of the system, providing outcomes that provide insight into risks to the IoT system and help with rationale when deciding about the controls. The IoTA-SRM framework provides a useful approach to security risk management in IoT systems. The application of the framework in the two case studies demonstrates the usefulness of the framework in providing a systematic and lightweight approach to security risk management, allowing for a shorter feedback loop between security analysts and IoT stakeholders and facilitating qualitative risk and control estimations.

The IoTA-SRM framework was presented and evaluated through its application in two autonomous vehicle (AV) case studies. Initial feedback from several IoT system stakeholders attests to the framework's capability to manage IoT security risks. Nonetheless, there remains a need for improvement in its practical deployment and educational strategies. The IoTA-SRM framework should not be merely presented as a theory; we have provided concrete tasks for each framework activity summarised in Table 32. Framework practitioners can follow these tasks to apply and practice the framework, resulting in tangible outcomes demonstrating its practical utility. Adding real-world IoT case studies to the IoTA-SRM framework enhances its educational value by providing a context for applying theoretical concepts. This combination of theory and practice aids participants in grasping the nuances of IoT security risk management.

Table 32. IoTA-SRM activities, tasks and outcomes

Activity	Activity Tasks	Outcome Artefacts
Model System	Decompose IoT system into IoT layers	
	Identify system and business assets for each IoT layer	Asset list
	Define security objectives for business assets per IoT layer	Security objectives
	Model decomposed system	Asset model
Discover Risks	Multi-layer vulnerability assessment	Vulnerability list
	Multi-layer threat elicitation	Threat list
	Multi-layer risk impact estimation	Risk impact information
Handle Risks	Multi-layer risk treatment decision	Risk decision
	Security requirements elicitation	Security requirements
	Control selection	Selected controls list
	Control implementation (can follow *Analyse Tradeoffs* outcome)	
Analyse Tradeoffs	Determine asset values from *Model Risk* activity	Asset metric values
	Estimate risk impact values from *Discover Risks* activity	Risk impact metric values
	Estimate selected controls costs from *Handle Risks* activity	Control cost metric value
	Run cost-benefit analysis for risk reduction	Prioritised risk list

Hackathons emerge as a viable platform for the real-world application and education of the IoTA-SRM framework. They facilitate practical exercises related to the framework's core activities: Model System, Discover Risks, Handle Risks, and Analyse Trade-offs. For example, hackathons can serve as the context for each framework activity. The expected outcomes, such as a list of identified assets, are part of the hackathon outcomes (see Table 32). While the IoTA-SRM framework is comprehensive, its complexity may pose barriers to those without technical expertise. This limitation could be addressed through targeted educational methods or tools that simplify its application. Hackathons, further elaborated in Chapter 4, are suggested to improve the framework's usability and broaden its reach.

3.6. Related Work

The IoTA-SRM framework addresses the limitations of existing security risk management approaches in IoT systems. Our framework considers the unique characteristics of IoT systems and provides a systematic process for managing risks across multiple architectural layers. Adopting an architecture perspective enables a comprehensive asset-oriented system analysis, multi-layer risk impact analysis, risk treatment, and tradeoff analysis. While the framework was demonstrated using the three-layer IoT architecture, its applicability to any IoT architecture similarly decomposed into layers can also be demonstrated.

The framework is compared to related works in Table 33. The SecIoT framework proposed by [17] covers IoT security requirements, authentication, secure communications, authorisation, and risk indicators. However, it does not recognise the architectural perspective of risk management or explore it systematically. Our framework extends the SecIoT framework by introducing the architecture perspective and performing a systematic security risk management analysis. Sim-

Table 33. Comparison of IoT frameworks for security risk management

Framework	Focus	Asset oriented	Architecture Perspective	Security Risk Management
IoTA-SRM (Our framework)	IoT security risk management	[++]	[++]	[++]
SecIoT [17]	IoT security requirements, authentication, secure communications, authorisation, risk indicators	[++]	[+]	[+−]
COBIT5 [226]	IT risk management	[++]	[+]	[−]
IoT-HarPSecA [227]	Secure IoT design and implementation	[−]	[−]	[−]

[++] Mostly fulfilled, [+−] Fulfilled with limitations, [+] Partially explored, [−] Not fulfilled

ilarly, other related works such as COBIT5 [226] and IoT-HarPSecA [227] discuss security concepts for IoT security risk management but do not provide a systematic approach to applying them nor include the architectural perspective. While COBIT5 is a framework for IT risk management that can be applied to IoT risk management, its application to IoT systems has not been actively explored. IoT-HarPSecA is a security framework that facilitates secure IoT design and implementation but focuses on eliciting security requirements and cryptographic algorithm recommendations. The frameworks compared, such as SecIoT [17], COBIT5 [226], and IoT-HarPSecA [227], have merits in the realm of IoT security but lack a systematic strategy to address security risk management in IoT systems that is flexible across different stages of IoT system development.

However, our IoTA-SRM framework can benefit from best practices and references from related works [17, 227, 226], covering asset management, risk assessment, risk management strategy, governance, and more. By incorporating these concepts and practices, security risk management practices for IoT systems can be improved.

3.7. Summary

Implementing security risk management frameworks for IoT systems can be challenging as IoT systems may require a unique approach to security. However, the IoTA-SRM framework overcomes this challenge by building on the IoT reference architecture that can capture the unique characteristics of different IoT systems. This makes it a versatile and flexible tool that can address the security needs of various IoT systems or specific IoT applications, making it essential in the constantly changing landscape of IoT security.

This chapter answers our research question "$\mathbf{RQ}_1$: *How to manage security risks in IoT architectures?*". In this chapter, we develop and validate a security risk management framework for IoT systems, leveraging the IoT system architecture to propose system decomposition, multi-layer risk analysis and implementing security measures to mitigate these risks at multiple layers of the IoT architecture, reducing the overall risk to the system.

Our proposed framework leverages the three-layered IoT architecture, guiding system decomposition into *components*, *dataElements*, *functions* and *dataFlows*, thereby revealing behavioural interactions and dependencies. Subsequently, we developed a conceptual model for security risk management to demonstrate multi-layer security risk management for IoT systems. Based on these foundations, the framework is formulated into four core activities: *Model System*, *Discover Risks*, *Handle Risks*, *Analyse Tradeoffs* with specific, actionable tasks and expected outcome artefacts. The framework was applied in two case studies, as demonstrated in Section 3.4.1 and Section 3.4.2. The developed framework and the results of the framework application were validated by stakeholders, confirming the framework's benefit to IoT security risk management.

However, the intricate nature of the IoTA-SRM framework may pose barriers to those without technical expertise. Thus, we further elaborate on using hackathons to improve the framework's usability and broaden its applicability in Chapter 4.

4. INTERVENTION-BASED HACKATHON APPROACH TO FOSTER SECURITY LEARNING

This chapter, which draws from the publications [68, 69, 67], explores a hackathon-based methodology as a teaching approach for managing security risks in IoT systems. Although hackathon organizers set up the environment for participants to work on projects, how participants capitalise on the offered learning opportunities is less controllable. Traditional hackathons provide a framework but often lack a structured educational environment conducive to achieving the objectives of IoT security risk management (SRM) instruction. Furthermore, despite the evident educational potential of practical-oriented cybersecurity hackathons, a gap exists in supporting the application of SRM knowledge. This need leads us to our research question:

RQ$_2$. *How to use a hackathon-based methodology to teach security risk management in IoT system architectures?*

To answer our research question, we propose a hackathon approach that integrates a tailored structure to facilitate SRM in IoT architectures. This adapted format should be grounded in structured educational content and enhanced by targeted interventions. Additionally, this format serves as a framework for assessing the attainment of specific learning outcomes related to IoT-SRM. An action research approach is utilised to formulate focused interventions to improve participants' learning experiences [39]. Through the application of the action research method, the assessment of hackathon interventions aimed at enhancing learning in the hackathon context (as outlined in Section 4.1) was conducted via three action research cycles, as detailed in Section 4.2. Our research unfolded in three action research cycles, each serving as a vehicle for developing and refining our hackathon interventions. The cycles facilitated ongoing assessment and adaptation, allowing us to optimise the alignment of the hackathon format with the IoTA-SRM framework. Each cycle incorporated feedback and insights from the preceding one, informing subsequent iterations and refining the interventions and the evaluation methods. In Section 4.3, our action research findings are discussed, encompassing the impact of our hackathon approach and the evaluation of the framework's applicability within the hackathon setting. Implications of the action research results for research are addressed in Section 4.4, and the conclusion of the chapter is presented in Section 4.5.

4.1. Hackathon Interventions for Security Learning

As the saying goes, you can lead a horse to water but can't make it drink [228]. Similarly, hackathon organizers can provide participants with the ideal environment and resources to learn and create but cannot force them to engage and take advantage of the opportunity fully [183]. This is where hackathon interventions

come into play, as intentional design actions that aim to foster learning and maximise the potential of the hackathon setting [68]. With the right interventions, organizers can facilitate a transformative learning experience for participants, but without them, the hackathon may be a missed opportunity.

In this section, lessons are drawn from key hackathon decisions as outlined in Section 2.3.3 to formulate hackathon interventions, which are detailed in Table 34.

Table 34. Proposed intervention design for learning adapted from [184]

Decision	Proposed Design Actions	Formalised Intervention
Ideation	Intervention that is dedicated to guiding ideation towards the learning goals and within the hackathon theme	Idea generation (Section 4.1.1)
	Intervention to help participants better understand the problem space and equip them to generate more informed and targeted ideas	Thematic Input (Section 4.1.3)
Team Formation	Intervention supporting team formation and the subsequent lifecycle of the team, promoting effective collaboration, problem-solving and communication to achieve learning goals.	Collaboration Support (Section 4.1.4)
	Collaboration support to guide task planning and collaboration between team members	
Specialised Preparation	Intervention to introduce thematic input that can foster learning by exposing participants to security concepts and techniques guiding hackathon project plans and prototypes	Thematic Input (Section 4.1.3)
Mentoring	Intervention to encourage team interaction with experts, allowing participants to incorporate this feedback into building their hackathon projects	Targeted Feedback (Section 4.1.2)
	Feedback and mentoring by experts to guide ideation and produce hackathon projects in line with the learning goals	
	Feedback and mentoring by experts to scope projects, suggestions on approaching the security problem and resolving technical issues	
	Expert feedback to help participants assess how their project outcomes compare to others and identify areas for improvement.	
Competition/ Cooperation	Intervention to introduce competition to generate unique solutions under competitive pressure or focus on cooperation to engage participants in a common learning goal or theme	Competition style (Section 4.1.5)

4.1.1. Idea Generation Intervention

The early part of a typical hackathon event is typically devoted to *idea generation*, setting the context of the hackathon projects. The idea generation intervention can foster learning by encouraging participants to think creatively and critically about security challenges, often starting with an open idea generation session [229], where participants express and refine ideas. One way to promote idea generation is by facilitating ideation sessions at the beginning or before the hackathon. During these sessions, participants are encouraged to generate as many ideas as possible, encouraging out-of-the-box thinking and leading to innovative solutions. The facilitator also provides prompts or questions to guide the ideation process,

such as "*What are the most critical security challenges facing an industry today?*" or "*What would be the ideal security solution to address a prevalent security threat?*". Another approach to fostering idea generation is for organisers to provide participants with use-cases or real-world examples of security challenges. Through analysing these use-cases, hackathon organisers familiarise participants with the security landscape and help them identify potential solutions for similar challenges. Additionally, use-cases inspire participants to think creatively and critically about how they can apply their skills and knowledge to solve real-world security problems. The idea generation intervention in security hackathons is essential for fostering learning by promoting creativity and critical thinking and providing participants with real-world examples to analyse and learn from [230].

4.1.2. Targeted Feedback Intervention

Feedback is specified information aimed at improving individuals' competencies or deepening their comprehension, emanating from various sources such as peers, experts, mentors, or even self-evaluation [231]. The medium through which feedback is conveyed is diverse, including but not limited to, oral or written communication in either formal or casual environments. The feedback may be prompted by an explicit inquiry or provided spontaneously. Crucially, the effectiveness of feedback relies on its alignment with three fundamental elements: it should focus on specific goals, assess advancement toward achieving those goals, and suggest actionable strategies for further improvement [232]. Feedback as a mechanism for enhancing learning is heightened when it is focused on particular aims and delivered consistently to facilitate ongoing improvement [231]. This evolving emphasis on feedback as a learner-centric process, wherein the recipients are actively engaged, resonates with contemporary perspectives on feedback design [233, 234, 235, 236]. Consequently, introducing targeted feedback into hackathons equips participants with precise, actionable insights, elevating their learning experiences. Throughout the hackathon, targeted feedback interventions are designed, with mentors or experts serving as the sources, to assist hackathon participants in refining their ideas and acquiring a deeper comprehension of the problem they are endeavouring to address. Such feedback guides and supports participants on technical issues, design, and presentation skills as they work on their projects. Feedback helps teams scope their projects, provide suggestions about how to approach a problem, help with (technical) problems [237], and provide participants with learning-oriented support, especially when mentors perceive their role as that of a traditional (workplace or educational) mentor [173]. Team interaction with mentors allows participants to incorporate this feedback into building their hackathon projects. Mentors also provide feedback at the end of the hackathon and during the final presentations when acting as judges. Teams provide a brief presentation or pitch of their solution to respective judges who evaluate each project and provide feedback on various aspects, including creativity, technical complexity, and

potential impact. This feedback shows the participants how their project compares to others and what they could have done differently.

Mentor feedback during hackathon events is a dialogical process that involves participants actively engaging with mentors, asking questions, challenging insights, and co-constructing the feedback. Such an approach aligns with the modern view of feedback, where participants participate actively in the feedback process [233, 234, 235, 236]. As implemented in our hackathon, mentor feedback embraces the new paradigm of feedback co-construction. Participants engage actively with experts, posing questions, seeking clarifications, and refining their ideas through a collaborative dialogue. This empowers the learners and aligns the feedback process with their specific needs and contexts, enhancing its effectiveness and relevance [238, 173].

4.1.3. Thematic Input Intervention

A hackathon's efficacy in promoting security learning significantly depends on providing learning content, encouraging hands-on learning, and tailoring educational content to align with participants' skills and needs. One approach to this is the thematic input intervention, which presents information about specific themes or issues related to the security problem the hackathon is geared towards addressing. This intervention aims to deepen participants' understanding of the problem domain and direct their idea generation towards specific areas. During the hackathon, thematic input is offered in various ways, such as expert-led talks, workshops, lectures and presentations that supplement learning [68] – for example, inviting cybersecurity experts to discuss topics related to the problem domain, such as common vulnerabilities and attack vectors in a specific system or application. Moreover, sessions teaching skills like penetration testing, cryptography, or threat modelling could be organised. In addition to the aforementioned, thematic input is delivered via pre-hackathon training resources like lectures, online courses or tutorials. These resources help participants gain fundamental knowledge about the problem domain and give them the information necessary for idea generation and task execution at the hackathon. Given the complexity of IoT security risk management, it's crucial to note that learning content may need to be segmented into smaller, digestible components to aid learning and practical application [7, 69]. As a result, the thematic input intervention should be integrated into a hackathon approach that enables organizers to introduce content gradually and in stages, permitting participants to concentrate on each interconnected component of a given topic.

4.1.4. Collaboration Support Intervention

Collaboration is an essential component of hackathons, requiring individuals to work together. Security risk management necessitates a collaborative effort involving multiple stakeholders, each bringing unique perspectives, technical ca-

pabilities, and motivations [239]. Thus, collaboration support interventions at hackathons help to enhance the collaborative power among hackathon participants, fostering learning by working together to solve the developed problem, complete needed tasks and learn new concepts [68]. One way to support collaboration is through team-building activities at the beginning of the hackathon. This allows participants to get to know each other and establish a sense of teamwork. In teams, participants are encouraged to share their knowledge and experiences, contributing to a collaborative learning environment. Providing resources such as collaboration tools and communication channels also fosters collaboration and ensures team members can communicate. Finally, providing opportunities for teams to share their progress and receive feedback from other participants is another way to support collaboration. This can take the form of short presentations or demonstrations, allowing teams to showcase their work and learn from each other.

4.1.5. Competition-style Design Intervention

Competition-style design intervention involves setting up a competitive environment where participants are given a specific security problem to solve or a security feature to the design. At the same time, they work in teams to develop the best solution within a set time frame. Competition-based hackathons (i.e., CTFs) [179, 180] have promoted skill acquisition in identifying and exploiting system vulnerabilities. Such competition-style designs incentivise participants to attempt challenging projects that might even be out of their comfort zone/zone of knowledge [179, 180]. This intervention fosters learning while engaging in challenging projects requiring them to apply their knowledge and skills in the real world [179, 180, 240]. It also encourages collaboration and teamwork, as team members work together to develop the best solution. The competitive aspect of the intervention motivates participants to perform at their best and push their limits. At the same time, the feedback they receive from the judges helps them to identify areas where they can improve.

4.2. Intervention-based Hackathon Approach: Action Research Method

Drawing from the discussions on interventions conducive to promoting learning in hackathon settings (in Section 4.1), these interventions were developed and assessed in three action research cycles, as illustrated in Figure 19. These cycles encompass planning, action combined with observation, and reflection stages (demonstrated in Figure 19).

In the action research cycles (cycle 1 - Section 4.2.1, cycle 2 - Section 4.2.2, and cycle 3 - Section 4.2.3), various interventions were evaluated to enhance security learning through hackathon events and educational settings. Participants,

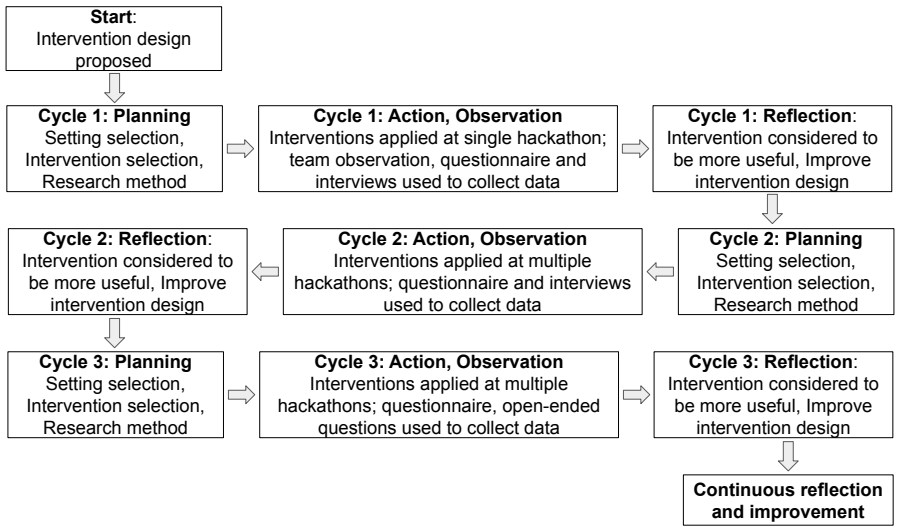


Figure 19. Action research cycles

encompassing diverse backgrounds but sharing a cybersecurity interest, were actively involved in these cycles. In cycle 1, our initial focus was broad, covering general security topics in IoT to provide participants with a foundational knowledge of cybersecurity principles. This phase introduced concepts such as top security risk trends in IoT, threat modelling, and basic security practices. It was a foundational stage to establish a fundamental understanding of cybersecurity's role in system design and implementation. Cycle 2 centred on software system design and security risk management. As the framework's development progressed, it delved into more specialised areas, integrating elements such as asset identification, risk assessment, and mitigation within the context of software system design. This phase empowered participants to apply risk management principles to software systems, introducing a higher level of complexity into their learning journey. In the final phase, cycle 3, the focus narrowed further towards IoT system security risk management. Here, the IoTA-SRM framework assumed a pivotal role as essential learning content. The framework provided specific tasks related to IoT security risk management, enabling participants to apply their knowledge to real-world IoT systems directly.

Throughout these cycles, the learning content evolved progressively, aligning with the refinement of the IoTA-SRM framework, as detailed in Section 3.1. In cycle 1, participants were introduced to general security topics in IoT, with a strong emphasis on fundamental concepts and practices in cybersecurity. Cycle 2 witnessed a shift in focus towards software system design and security risk management, incorporating specialised elements such as asset identification and risk mitigation. Finally, cycle 3 centred on IoT system security risk management, introducing the IoTA-SRM framework as vital learning content. This framework enabled participants to apply their knowledge directly to real-world IoT systems.

These cycles exemplified a progressive approach to customising learning content and interventions for cybersecurity education.

4.2.1. Action Research Cycle 1

Between mid-june and mid-october 2019, the first action research cycle was carried out. During this period, idea generation, thematic input, and targeted feedback interventions were introduced at a 48-hour cybersecurity hackathon event. An analysis assessed how these interventions could promote security learning (refer to Table 35). Comprehensive documentation of the hackathon setting, the interventions employed, the research methodologies applied, and the outcomes of the intervention evaluation is found in [68].

Table 35. Hackathon intervention action research cycle 1 overview

Learning Content	Interventions Used	Learning Context/ Setting	Research Methods
Top security risk trends, security risk management techniques, and general security practices	**Idea generation** as a dedicated pre-event and during the hackathon event **Thematic input** as security talks **Targeted feedback** through free-flowing and dedicated mentors to teams **Competition-style** as prizes to the most innovative team	Single cybersecurity hackathon event, 48 hours	Team observation Post-hackathon questionnaire Interviews

Planning. We selected the hackathon format as a single 48-hour cybersecurity event. We designed learning content to teach participants foundational security concepts that is useful for individuals with varying expertise in the field. The learning content includes general security practices that cover a broad range of security concepts and issues, such as the top security risk trends in IoT and security risk management techniques. The thematic input intervention is utilised to deliver the learning content described. The learning objective is for participants to achieve familiarity with IoT security concepts and apply them in their hackathon projects.

Based on section 4.1, we designed and introduced the following interventions for this action research cycle:

- **Idea generation**: The idea generation intervention consisted of two parts: (*i*) a dedicated 8-hour idea-generation event before the main hackathon, where participants could fully prepare ideas and form preliminary teams, and (*ii*) a typical ideation session for all participants at the beginning of the main hackathon event where they could propose and refine their ideas.
- **Thematic Input**: We also introduced thematic input in the form of *security talks* during the main hackathon and idea generation events. The talks covered top security trends in IoT, security risk management, and the general aspects of security learning.
- **Targeted Feedback**: We introduced the targeted feedback intervention by organising mentors in two ways; (*i*) mentors assigned to teams based on the

team's needs and (*ii*) free-flowing mentors with a broad range of security-related expertise to support multiple teams.

- **Competition Style**: In the competition-style intervention, we gave prizes to teams that were seen to have attempted challenging projects.

We did not introduce the following intervention for this action research cycle:

- **Collaboration Support**: The collaboration support intervention was not implemented during this hackathon as there was no need for such additional support. The hackathon organizers had adeptly recruited participants with diverse skill sets that complemented the learning context. The ideation and mentor support setting also provided an ideal atmosphere for fostering teamwork and collaboration.

For data collection, observation methods, questionnaires and interviews are selected.

Action, Observation. Our action and observation activities are detailed in [68]. We illustrate the timeline of activities in Figure 20. The hackathon preparations

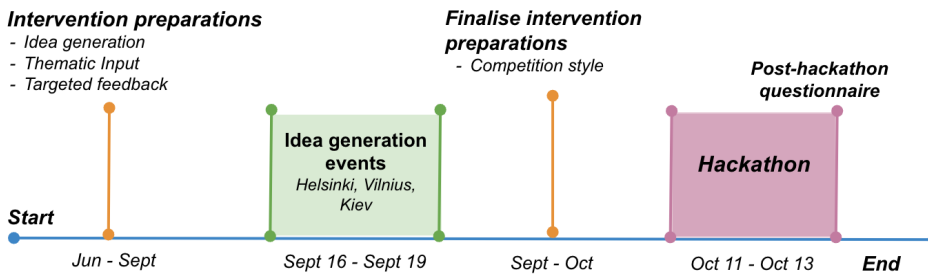


Figure 20. Timeline of activities for action research 1

included designing interventions, organising idea-generation events, and kick-starting participants towards the event's IoT security theme. The main hackathon started with an idea generation session, offering those who hadn't attended the initial ideation events an opportunity to propose and refine their ideas. After ideation, the participants formed teams of 5-8 participants per selected idea and began working on their tasks. Security experts provided presentations to give the participants more security considerations when building their projects. At the end of the hackathon, all teams presented their projects and prototypes for evaluation. The hackathon provided live-streamed presentations of security projects and prototypes to all interested community members. After evaluation, the judges presented prizes to the selected winners.

We collected observational data between the teams and mentors during the hackathon event. After the hackathon event, we provided a post-hackathon questionnaire instrument (see Appendix E - Section E.1), where participants voluntarily provided their responses. Post-hackathon responses were gathered from twelve (12) participants representing seven (7) teams out of ten (10) teams that participated in the hackathon event.

To contextualise the questionnaire findings regarding the perceived benefits of interventions for learning, we analysed team observations and conducted interviews with selected participants (see Appendix E - Section E.1) to inform our findings. We selected three teams (A, B, and C) for our analysis. Our selection was based on the varying levels of participation in the hackathon events where our interventions were implemented, including the idea generation pre-hackathon events. The selected team characteristics are summarised in Table 36. From each team, two participants were identified for interviews, ensuring they were available and consented. Our selection was influenced by two main criteria: either the participant held a pivotal role like a team leader or was notably active and contributive during the event. The interviews, which typically lasted between 25 to 30 minutes, revolved around three core areas: the overall experience of the hackathon, key learnings acquired, and specifics about the security projects they pursued. Once participants provided their consent, we recorded their responses. These voice recordings were later converted into text format using a dedicated transcription service. The textual data underwent analysis using a coding system (see Appendix E - Table 56). This approach facilitated the extraction of common themes and patterns across the interviews, giving us an additional perspective on the security learning benefits of the hackathon event and the participants' experiences.

Table 36. Action research cycle 1: Team characteristics for data analysis [68]

Team	# team members	Interview participants	Selection criteria
A	6	A01 (team lead), A02 (lead developer)	No participation in idea generation pre-hackathon event.
B	6	B01 (team lead), B02 (security expert and developer)	Participation in idea generation pre-hackathon event and continued with the same idea at the hackathon.
C	5	C01 (team lead), C02 (developer)	Participation in idea generation pre-hackathon event but did not continue with the same idea at the hackathon.

Findings of the post-hackathon questionnaire about the perception of learning from interventions by the hackathon participants are illustrated in Figure 21. All responses were given on a 5-point scale anchored between *strongly disagree* (1) and *strongly agree* (5). The participants expressed above-average perceptions of the interventions. The thematic input intervention, introduced as security talks, garnered the highest perceived learning benefits ($M = 4.00, IQR = 0.5$). Furthermore, participants recognised the value of idea generation ($M = 3.00, IQR = 1.0$), targeted feedback ($M = 3.00, IQR = 1.0$), and competition-style ($M = 3.00, IQR = 1.0$) interventions to foster learning benefits. Details on the team's journeys and team properties and findings of the differences between teams concerning their learning process are provided in [68].

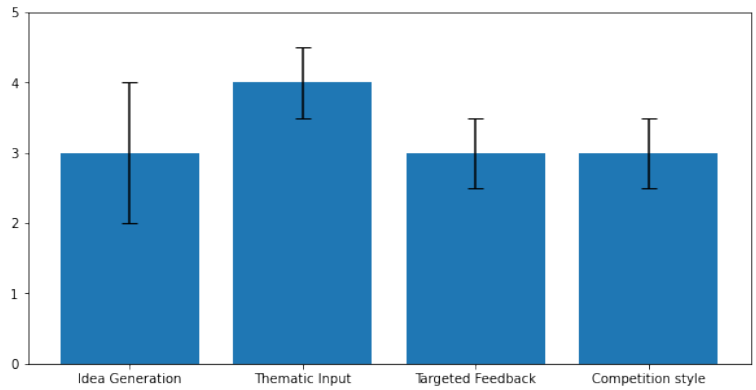


Figure 21. Questionnaire responses by participants after the hackathon about interventions in action research 1

Using team observations and interviews offered insights into how hackathon interventions influenced the learning experiences of the three teams (Teams A, B, and C), supplementing findings from our post-hackathon questionnaire results.

- **Idea generation**: In Team A, the leader (A01) proposed an idea derived from a security problem encountered in their academic studies. However, this initial idea required refinement through discussions within the team and mentor feedback to align with the target audience's needs. In Team B, the leader (B01) attended pre-hackathon events to refine their idea, focusing on making data security more desirable for startups. Team C's idea evolved from a blockchain-based betting platform to an availability insurance smart contract for service providers based on mentor feedback.
- **Thematic Input**: The thematic input introduced through security talk sessions had varying impacts on the teams. Team A members reported high learning gains and an improved understanding of risk management and cybersecurity, which informed the development of their project. Similarly, Team B members reported gaining knowledge on securing systems, contributing to their prototype. In contrast, there were no explicit individual reports of learning experiences from the security talk sessions in Team C, but questionnaire responses indicated that participants did gain security knowledge. Overall, the security talks seemed to be most impactful for Teams A and B in enhancing their understanding of security concepts.
- **Competition Style**: The competition style of the hackathon affected the teams differently. Despite not winning a prize, Team A recognised the educational value of the competition. Team B also reported moderate learning, and their project's uniqueness and usefulness earned them a prize. On the other hand, Team C did not win a prize, and satisfaction with the project outcome was moderate. The competition style motivated Teams A and B to develop prototypes and learn from the experience.
- **Targeted Feedback**: The targeted feedback intervention played a signif-

icant role in all three teams. In Team A, mentors reportedly guided the refinement of the project idea and scoping it to meet the team's needs. In Team B, mentors' multiple visits contributed to scoping and refining the project tasks but disrupted the flow of tasks. In Team C, mentor feedback was essential in shaping their idea into a project adequate for a security hackathon. Additionally, all teams reportedly used mentor feedback to enhance the proposed security prototypes. The mentors were seen to be critical in providing guidance, expertise, and project adjustments.

Reflection. Overall, our findings suggest that participants highly valued the introduction of thematic input sessions, possibly due to the enriched content that facilitated their engagement and understanding of the IoT use case. Still, the participants perceived that other interventions introduced also contributed to participants learning experiences. However, their perceived benefits varied, suggesting that participants may have different preferences and learning styles and that interventions should be tailored to meet diverse learning needs. We discuss the lessons learned from the participant's experience with the introduced interventions, how they benefitted the learning process at this single hackathon event, and summarise suggestions from our findings to improve our interventions for the next cycle (see Table 37). Details on our findings from which we derived these lessons are documented in [68].

Table 37. Suggestions to improve action research cycle 1 interventions

Intervention	Introduced as	Suggestions for Improvement
Idea Generation	Pre-hackathon and early hackathon idea generation events	• Encourage participation at pre-hackathon and early hackathon ideation events • During the hackathon, continue coaching participants on ideas generated before and at the beginning of a hackathon • Ideas proposed or case studies introduced should be real-world problems aligned with the hackathon's theme
Targeted Feedback	Free-flowing and dedicated mentors to teams	• Introduce multiple forms of mentor feedback • Organise mentor-participant interaction to target participants' needs
Thematic Input	Talks delivered by security experts	• Tailor thematic input relevant to the team's project idea and participant skill level • Introduce some thematic input and resources at the pre-hackathon stage to provide the fundamental knowledge needed for the hackathon
Competition Design	Prize incentives to the most innovative solution	• De-emphasise the prizes in a competitive style event, so participants do not over-emphasise competition over learning

- **Idea generation**: The idea generation intervention was instrumental in completing the security project, providing a dedicated event to kick-start

ideation before the main hackathon event and another session at the main hackathon event for participants who did not attend the dedicated ideation event. Participants who took advantage of the idea generation intervention by attending the pre-hackathon ideation event could generate and mature their security project idea before the main hackathon event, allowing for more time to work on their project during the main hackathon. However, we saw that teams who opted out of using the idea generation intervention had less time to mature their idea and focus on hackathon execution and fewer chances to be involved in as much learning. While the observations suggest that participants who attended the pre-hackathon ideation event had more time to develop their projects during the main hackathon, other variables, such as team dynamics and skill level, may contribute more significantly to how much a team can achieve during the hackathon.

- **Thematic input**: The thematic input intervention introduced as security talks provided learning benefits when tailored towards the required security knowledge relevant to the team's project idea and the participant's skill level in completing the hackathon tasks. We saw that providing fundamental knowledge of each participant's idea might be challenging for future use of this intervention. Thus, structuring hackathon events so participants are encouraged to generate ideas within the hackathon context where security talks have been prepared could improve learning benefits. However, to prevent an overly restrictive environment, efforts towards mentor support and targeted feedback can offer targeted security knowledge to participants and allow participants to recognise security aspects within their hackathon artefacts.
- **Targeted feedback**: The mentor feedback intervention also provided learning benefits because of the high interaction with diverse mentors where different mentors visited the team multiple times to provide an expert perspective on work progress. The intervention also provided learning benefits because of the high interaction with diverse mentors, where different mentors visited the team multiple times to provide an expert perspective on work progress. Mentor interaction in idea generation also helps to support the completion of set tasks for the security project. However, it appears crucial that we organise mentoring appropriately to ensure adequate mentor interaction, typically when requested by the teams, based on their needs.
- **Competition style design**: The competition-style intervention encouraged rapid knowledge gathering and application of the security knowledge to product creation, thus winning a prize at the hackathon. However, the perceived learning benefit could have resulted from the culminating factors of idea generation, team formation, and the team dynamic experienced within the competition constraints.

Though beneficial, the typical hackathon format used in this action research

cycle presents two primary obstacles to effective learning: time constraints and the lack of a formal pedagogical framework. B01 highlighted that the time limits of the event are a barrier to deep intellectual engagement: "*hackathon didn't give the possibility to think about learning. This time limit doesn't support this [learning]*". This sentiment echoes A01's input, who expressed the necessity for extended time to engage in learning for producing meaningful outcomes: "*I would need to kind of learn a lot and to do a lot of research to actually do something useful*". Both responses indicate time constraints when using hackathons as a learning approach, highlighting a need to improve the hackathon format to allow for a more comprehensive understanding and application of knowledge. The second issue is the absence of structured pedagogy, which Falk *et al.* [241] noted as a characteristic shortfall of the hackathon format for learning and was seen in this event. In such an unstructured setting, participants may not maximise their learning potential due to a lack of targeted guidance or a well-defined curriculum. And with the additional challenge of time constraints, The combination of these issues could result in participants not using their time effectively for learning. Therefore, incorporating educational components and rethinking the time structure of hackathons could enhance the learning experience in future action research cycles.

4.2.2. Action Research Cycle 2

Between January and early June 2021, we conducted the second action research cycle. Learning from cycle 1, we introduced interventions in an academic course providing a pedagogical basis. Before incorporating the interventions into a course, our initial objective was to evaluate their benefits in a typical hackathon. We achieved this in Section 4.2.1. Learning from cycle 1 (see Table 37), we introduced the thematic input, targeted feedback and collaboration support hackathon interventions within a cybersecurity course. We also extended the time frame of the hackathon events and introduced multiple hackathon events, providing repeated opportunities for applying knowledge gained. Details of the hackathon setting, interventions used, and findings of our intervention evaluation are documented in [69].

Table 38. Hackathon intervention action research cycle 2 overview

Learning Content	Interventions Used	Learning Context/ Setting	Research Methods
Security risk management in software system design – asset, risk and risk-treatment related security concepts.	**Thematic input** as lectures **Targeted feedback** through mentor support with the course instructors as mentors **Collaboration support** introduced as team management plan	Multiple hackathon events integrated into a cybersecurity course	Post-hackathon questionnaires (including open-ended questions)

Planning. We integrated multiple hackathons into a cybersecurity course for this evaluation. The course description is provided in Appendix D. The primary

aim was to introduce practical experience into the curriculum, particularly on security risk management. This course was dedicated to studying secure software system design, emphasising a security risk-aware perspective—making it an ideal setting for our hackathon approach. This perspective aligned with our hackathon approach, and to enhance this, we included an IoT use case for contextual relevance to participants.

We designed learning content introduced during a cybersecurity course to teach about secure software system design from a security risk-aware perspective. The learning content includes topics on the security of software system assets, security requirements engineering and modelling, and the implementation of major security controls, like role-based access control and cryptography, fundamental to secure software design. The learning objectives are for participants to understand the system's context and security risks, master techniques to mitigate and establish security requirements and apply modelling techniques to build secure systems and software. To achieve this using hackathons, the course was split into three (3) aspects: *(i)* asset-related concepts, *(ii)* risk-related concepts, *(iii)* risk-treatment-related concepts where hackathons are introduced to support the learning of these aspects. Unlike traditional educational hackathons, which are usually one-off (single) events at the end of a course, we designed multiple hackathons to reinforce each component. Research indicates that effective learning, particularly in online settings, requires frequent practical application [242]. Therefore, single events at the course's conclusion may not suffice for optimal learning. Consequently, we organised multiple hackathons, each lasting approximately 48 hours and spread over 14 days, to cover the three major aspects of the course. When interventions are applied in an academic setting, a structured task-based hackathon design is typically used. Based on section 4.1 and lessons learned from our application of the interventions at the first evaluation of the interventions summarised in Table 37, we designed and introduced the following interventions for this action research cycle:

- **Thematic Input**: We introduced the thematic input intervention as lecture materials suitable for online instruction, provided as part of the cybersecurity course. Learning from thematic input suggestions for improvement in Table 40, thematic input intervention is introduced before each hackathon event to teach the subject matter before the hackathon events. Lecture resources are also available during the hackathon event for participants to access at their convenience and reflect on their applicability to the hackathon tasks.
- **Targeted Feedback**: The targeted feedback intervention consisted of mentor support interactions to allow hackthon participants gain expert feedback and answer questions regarding hackathon tasks. Learning from targeted feedback suggestions for improvement in Table 37, we organised: (*i*) online feedback/ consultation sessions to provide immediate feedback to partici-

pants' needs during the hackathon event [243]; and (*ii*) asynchronous feedback in written form, containing feedback commentaries on the hackathon outputs [244]. Since we designed closely related hackathon events, we planned the feedback provided during a hackathon to benefit the upcoming hackathon tasks.

- **Collaboration Support**: As the participant's formed teams for the hackathon events, we introduced collaboration support intervention as a team management plan to foster collaboration and ensure team members can collaborate to accomplish their tasks. The team management plan also aimed to support task organization and assignment, or team leadership [69]. However, we did not introduce collaboration support at the first hackathon event. This decision was based on our understanding of hackathons as inherently collaborative environments. We believed that targeted collaboration support might not be immediately necessary until participants established greater collaboration and familiarity within their teams.

We did not introduce the following intervention for this action research cycle:

- **Idea generation**: We decided that the idea generation intervention was unnecessary for this hackathon setting since the typical ideation component of hackathons was not utilised. Instead, participants were provided with pre-defined use cases to apply the security concepts introduced to them.
- **Competition-style design**: We did not implement the competition-style design intervention, as the priority is learning gains and not stimulating innovative or creative solutions. Moreover, the hackathon outcomes offered additional points towards the participants' course grades, incentivising them to complete the hackathon tasks.

For data collection and analysis, post-hackathon questionnaires and open-ended questions are selected.

Action, Observation. We illustrate the timeline of activities in Figure 22. The course started with introductory lectures that explained the learning goals of the course and the hackathon format introduced. After this, the participants formed teams of three (3) or four (4) members to work on hackathon tasks. The course instructors presented two IoT-based intelligent transportation systems, namely the Bike Sharing System (BSS) (see Appendix A - Section A.3) and an Autonomous Vehicle Parking System (AVPS) (see Appendix A - Section A.2), with UML diagrams and textual descriptions as practical scenarios for analysis. The participants were required to choose one of these use cases for analysis. As the course continues, we introduce interventions at each hackathon event and obtain the output of each hackathon as a security report. The setting at each hackathon event is documented in [69].

After each hackathon event, we provided post-hackathon questionnaires to participants by adapting pre-existing instruments with open-ended questions in the questionnaire to provide contextual information (detailed in Appendix E -

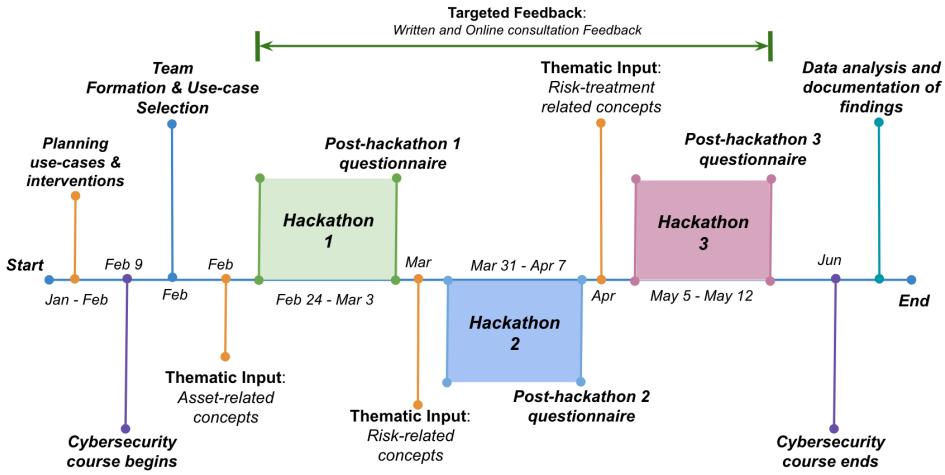


Figure 22. Timeline of activities for action research 2

Table 57). The questionnaire included questions about team familiarity, team process satisfaction, and participant perception of achieving learning outcomes during the hackathons. We also measured the perception of the usefulness of the interventions to foster learning. We analysed responses from six (6) teams with twenty-three (23) participants based on the team size (between three (3) and four (4) members), course grade outcome and teams who provided more complete responses to the questionnaires [69]. The selected team characteristics are summarised in Table 39. Findings of the post-hackathon questionnaire about the

Table 39. Action research cycle 2: Team characteristics for data analysis [69]

Teams	# of team members	Participants
Team A	4	A01, A02, A03, A04
Team B	4	B01, B02, B03, B04
Team C	4	C01, C02, C03, C04
Team D	4	D01, D02, D03, D04
Team E	4	E01, E02, E03, E04
Team F	3	F01, F02, F03

perception of learning from interventions by the hackathon participants are illustrated in Figure 23. All responses were given on a 5-point scale anchored between *strongly disagree* (1) and *strongly agree* (5). Details on the team's journeys, properties, and findings of the differences between teams concerning their learning process are detailed in [69].

Our findings revealed consistent positive perceptions of learning benefits across interventions. The thematic input intervention consistently garnered high perceptions ($M = 3.75, IQR = 0.44$) for the first hackathon, ($M = 3.73, IQR = 0.61$) for the second, and ($M = 4.00, IQR = 0.64$) for the third. This consistency suggested that participants perceived this intervention as valuable throughout the hackathon events, with the highest benefit reported in the third event. Simi-

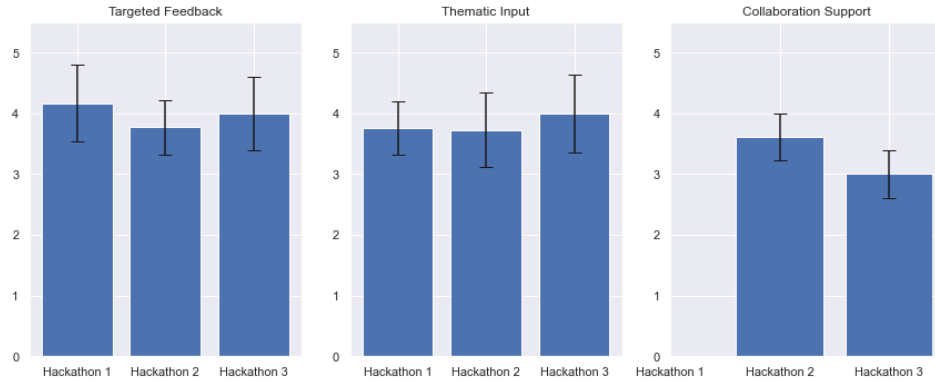


Figure 23. Questionnaire responses by participants about interventions at the hackathons in action research 2

larly, targeted feedback intervention maintained a positive impact on learning ($M = 4.17, IQR = 0.63$), ($M = 3.77, IQR = 0.44$), ($M = 4.00, IQR = 0.60$) across the three hackathons. However, collaboration support intervention exhibited varying perceptions ($M = 3.61, IQR = 0.39$) and ($M = 3.00, IQR = 0.40$) for the second and third hackathons, indicating the need to explore its impact on learning further. However, it is important to note that no data was available for the collaboration support intervention at the first hackathon event. Our analysis of the open-ended questions in the questionnaire also provided contextual insights on the contributions of the hackathon interventions to security learning:

- **Thematic input**: While all teams generally found the lectures useful, the teams saw more value. A01 from Team A mentioned that while the lectures and resources were easy to follow, other team members had different understandings, leading to challenges in task completion due to misunderstandings. This suggests that aligning individual interpretations of lecture content within a team was crucial. B01 from Team B emphasised that the lectures and reading resources were crucial for task completion, highlighting their practical relevance. However, other participants (F01 and C02) expressed that some lecture topics were challenging, and there was a suggestion for refining the difficulty level of the content. Despite challenges, the lectures seemed to help teams comprehend and address the tasks at hand.
- **Targeted Feedback**: The targeted feedback intervention was reported positively across all teams. A03 from Team A mentioned that the written feedback felt vague and rushed, requiring additional clarification. D02 acknowledged the benefits of feedback but suggested that there could be more effective ways to deliver it for both learners and lecturers. Specifically, B01 from Team B felt that receiving real-time explanations during online feedback sessions was more effective, as it allowed for immediate clarifications on comments. Teams used feedback to improve their work and catch inconsistencies or errors before submissions. B01 from Team B, in particu-

lar, credited feedback sessions for identifying and correcting issues in their work.

- **Collaboration Support**: Teams generally positively perceived the collaboration support intervention. The provided team management plan addressed team familiarity, communication, and organization issues. At the first hackathon, challenges with planning and finding time for collaboration were noted. A03 faced team organization and coordination difficulties, and B01 initially reported task division and organization issues. With the introduction of collaboration support in the second hackathon, E01 from Team E noted improved communication and organization within their team, and B01 also appreciated the plan's contribution to task tracking and team progress monitoring. However, D01 from Team D indicated that the plan helped define smaller tasks but didn't significantly enhance their overall work. This variation in perception suggests that the usefulness of this form of collaboration support might depend on factors such as team dynamics and individual working preferences.

Reflection. Overall, the interventions were generally well-received by participants across the hackathon events. The thematic input and targeted feedback interventions consistently demonstrated positive learning benefits. At the same time, the collaboration support interventions exhibited varying levels of effectiveness at different hackathons. We discuss lessons learned from the participant's perception of how the thematic input, targeted feedback, and collaborative support interventions benefited in achieving the learning goals and summarise suggestions from our findings to improve our interventions for the next action research cycle (see Table 40). Details on our findings from which we derived these lessons are documented in [69].

Table 40. Suggestions to improve action research cycle 2 interventions

Intervention	Introduced as	Suggestions for Improvement
Thematic Input	Lectures delivered by course instructors	• Educators can discuss the lecture materials and how they relate to the hackathon tasks during lectures • Consider the balance between the quantity of the theory provided and its applicability in the hackathon tasks
Targeted Feedback	Mentor support providing written and consultation feedback	• Invest more time in consultation sessions (immediate feedback) during the hackathon, where roadblocks are resolved through expert guidance in a short period
Collaboration Support	Team management plan document provided to teams	• Can be made optional as our findings show minimal direct impact on learning • Can be beneficial where there are complex hackathon tasks and where team members have low familiarity

- **Thematic Input**: We designed the lectures to provide in-depth security knowledge on the security concepts needed for the course. The hackathon tasks were then crafted with the course curriculum in mind. The hackathon

participants perceived the thematic input intervention (introduced as *lectures*) as greatly beneficial when understandable and applicable to the hackathon tasks. They thus reported learning by applying security concepts and practices explained in the lectures in their hackathon tasks. However, in the bid to provide as much information as required for learning, the participants also perceived the lectures as having too much content, leading to confusion, affecting the participant's learning process and, in turn, the team's output. Some teams mentioned that they allocated significant time to absorbing the lectures before engaging in the tasks, which limited their ability to fully capitalise on the "learning by doing" approach. To improve this intervention following the hackathon approach, we suggested more consideration of the balance between the quantity of theory provided and its applicability to the hackathon tasks. We also suggested using the targeted feedback intervention, where educators can further discuss the lectures during the feedback sessions and how they relate to the hackathon tasks.

- **Targeted Feedback**: The participants perceived that targeted feedback intervention (introduced as *mentor support*) benefited learning. It allowed their teams to discuss possible misunderstandings and errors with mentors and prevent repeating past mistakes or introducing new errors into the task outcomes of future/subsequent hackathons. We saw that the targeted feedback interventions contributed to the teamwork through the opportunity to clear out misunderstandings within the team about the lectures and the previous and current hackathon tasks. We also observed distinct advantages for immediate and asynchronous feedback within the hackathon. Immediate feedback, mainly through real-time online consultation sessions, allows for dynamic adjustments and roadblock resolution through expert guidance in a shorter period or on the fly. This fosters an agile learning environment, which is particularly beneficial for the fast-paced nature of a hackathon. On the other hand, asynchronous (written) feedback, provided after the submission of hackathon tasks, offers participants the time to reflect on their performance more comprehensively. This is especially useful for complex challenges requiring deeper thought and synthesis. Therefore, we advocate for including immediate and asynchronous feedback mechanisms as complementary components in hackathon education models, each catering to different learning needs and phases of the hackathon.
- **Collaboration Support**: The participants perceived that the collaboration support intervention (introduced as the *team management plan*) benefited the team process. At the first hackathon, the participants perceived a low familiarity between team members. Consequently, they reported that coordinating tasks within the team was challenging and time-wasting as each team member learned to collaborate for the first time. Thus, the participants reported that the team management plan aided in defining and coor-

dinating hackathon tasks, assigning responsibility, and setting deadlines for hackathon tasks. Additionally, some teams who saw minimal contribution from the team management plan reported that using the document did not hinder the participants from their hackathon tasks nor pose an additional workload. We found that overall, the team management plan intervention improved collaborative power for teams, which improves teamwork. However, the participants indicated a low perception of direct learning benefits from this intervention. The collaboration support intervention should instead be introduced in a context that amplifies the ongoing collaborative dynamics and tackles specific challenges that teams encounter during the hackathon's progression. This approach ensures that the intervention aligns with the natural evolution of team dynamics, strategically enhancing their effectiveness at a pivotal juncture of the hackathon process.

Notably, the hackathon format introduced in this cycle was designed to integrate seamlessly with the course content, thus providing a dynamic setting for hands-on practice and feedback. The format demonstrated significant improvements in applying theoretical knowledge to real-world scenarios, thereby augmenting learning outcomes across different phases of the course. This contrasts traditional models, such as the one presented by Gama *et al.* [245], where hackathons typically serve as a culminating experience at the course's end. This format is included in our next action research cycle to complement the IoTA-SRM framework by offering a real-world, controlled environment where theoretical concepts are applied, evaluated, and refined.

4.2.3. Action Research Cycle 3

Between January and early June 2022, we conducted the third action research cycle. We conducted the third cycle of our action research method to assess the benefits of the hackathon learning model in teaching the IoTA-SRM framework and its use within the hackathon approach. This cycle marked the final iteration of our overall action research process (see Figure 19). In this cycle, we implemented interventions in an educational hackathon setting to educate participants on applying the IoTA-SRM framework. The educational hackathon incorporated typical activities adapted to align with a cybersecurity course (see course description in Appendix D). We also incorporated feedback and suggestions for improvement from Table 40 to introduce targeted interventions for learning.

Table 41. Hackathon intervention action research cycle 3 overview

Learning Content	Interventions Used	Learning Context/ Setting	Research Methods
IoTA-SRM framework	**Thematic input** provided as lectures **Targeted feedback** through mentor support with the course instructors as mentors	Multiple hackathon events integrated into a cybersecurity course	Post-hackathon questionnaires (including open-ended questions)

Planning. Like cycle 2, we organised three hackathon events facilitated by the course instructors, designing learning content based on the activities outlined in the IoTA-SRM framework, hackathon interventions and outcome artefacts. The learning objective was for participants to apply the IoTA-SRM framework and accomplish its tasks. Although the IoTA-SRM framework encompasses four activities, the hackathons were only organised around the *Model System*, *Discover Risks*, and *Handle Risks* activities, highlighted in Table 42 to fit within the duration of the course.

Table 42. IoTA-SRM activity tasks and outcomes for action research cycle 3

Activity	Activity Tasks	Outcome Artifacts
Model System	Decompose IoT system into IoT layers	
	Identify system and business assets for each IoT layer	Asset list
	Define security objectives for business assets per IoT layer	Security objectives
	Model decomposed system	Asset model
Discover Risks	Multi-layer vulnerability assessment	Vulnerability list
	Multi-layer threat elicitation	Threat list
	Multi-layer risk impact estimation	Risk impact information
Handle Risks	Multi-layer risk treatment decision	Risk decision
	Security requirements elicitation	Security requirements
	Control selection	Selected controls list
	Control implementation (following *Analyse Tradeoffs* outcome)	

The cybersecurity course took a security risk-aware approach to secure software design, emphasising the protection of software system assets, security requirements engineering and modelling, and understanding key security controls. These course materials align with the concepts in the IoTA-SRM framework introduced in Chapter 3. The learning content of the course encompasses the IoTA-SRM framework activities that participants will practice throughout the course. Additionally, practical tasks (see Table 42) are provided from these activities, allowing participants apply the IoTA-SRM framework to a selected IoT use case and perform each risk management activity, ultimately producing outcome artefacts. We specifically focused on three (3) IoTA-SRM framework activities during the cybersecurity course. The learning content of the course revolves around these activities, providing participants with a comprehensive understanding and practical experience related to these specific aspects of the framework. Thus, introducing the IoTA-SRM framework, the hackathon approach, into the secure software design course establishes a comprehensive educational setting. The IoTA-SRM framework is tailored for IoT security risk management and adds depth to the course's secure software design topics. The hackathon approach facilitates directly applying the IoTA-SRM framework's principles in real scenarios through hands-on and group learning. Meanwhile, the cybersecurity course offers the foundational knowledge and context for integrating the framework and the hackathon technique.

Lessons learned from previous action research cycles and the suggestions out-

lined in Table 37 and Table 40 influenced the implementation of specific interventions in this cycle. The idea generation intervention, similar to our second evaluation in Section 4.2.2, was deemed unnecessary for this hackathon setting. Since the hackathon did not emphasise ideation as a typical component, participants were provided with pre-defined use cases to apply the introduced security concepts. This approach aimed to focus more on the practical application of the IoTA-SRM framework. The collaboration support intervention was not implemented to foster learning during this hackathon. Based on our second cycle in Section 4.2.2, we did not observe a direct impact of collaboration support on learning. Additionally, the mentor support provided an ideal environment for fostering teamwork and collaboration among the participants. Lastly, the competition-style design intervention was not implemented, similar to our second cycle in Section 4.2.2. The primary focus of the hackathon was not to stimulate innovation but to provide participants with an opportunity to accomplish the hackathon tasks. Moving forward, we incorporated insights gained from our experiences with hackathon interventions to introduce two suitable hackathon interventions: thematic input and targeted feedback.

- **Thematic Input**: Throughout the hackathon, participants will receive thematic input through lectures (including lecture materials and relevant resources). These inputs will cover essential cybersecurity topics and concepts related to the IoTA-SRM framework. The learning content will be divided into separate modules to ensure a better understanding and digestion of complex topics at each hackathon iteration. This approach aims to provide participants with the necessary theoretical knowledge to support their practical implementation of the framework.
- **Targeted Feedback**: Course instructors, acting as mentors, will provide targeted feedback to participants. This feedback will be delivered through various channels, such as online consultations, hackathon outcome presentations, and asynchronous written feedback. Mentors will guide participants in effectively applying the IoTA-SRM framework, offer suggestions for improvement, and address any questions or concerns that arise during the hackathon. By leveraging targeted feedback, participants can enhance their learning experience and elevate the quality of their hackathon artefacts.

Action, Observation. We illustrate the timeline of activities in Figure 24. At the beginning of the course, the hackathon format was introduced, requesting participants to form teams of three (3) or four (4) members. Participants in the study were cybersecurity participants with varying levels of prior knowledge in cybersecurity and diverse backgrounds. These hackathons, focused on three of the IoTA-SRM framework activities (see Table 42), were conducted within the duration of the cybersecurity course. To provide a practical scenario for analysis, the course instructors provided a micro-mobility IoT system use case, UML diagrams, and textual descriptions (see Appendix A - Section A.4). The practical tasks derived

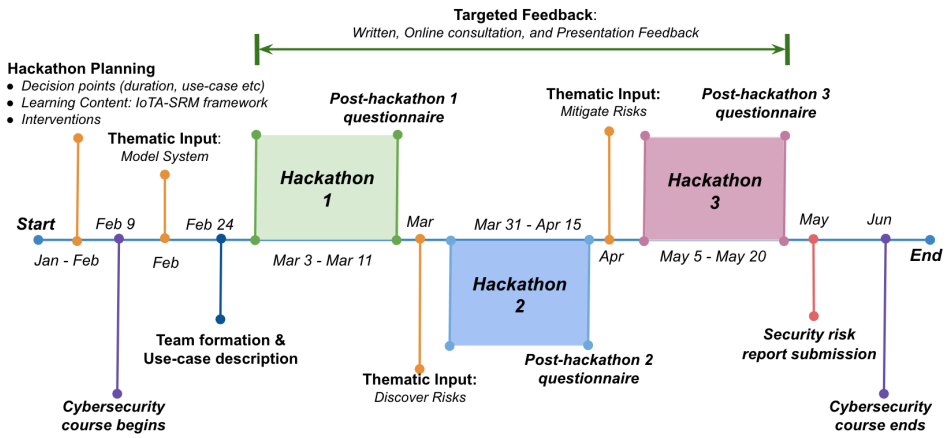


Figure 24. Timeline of activities for action research 3

from the framework were assigned to participants, aiming to apply these tasks to the IoT system use case and generate security outcome artefacts.

In the first hackathon event, named *Hackathon 1* in Figure 24, the focus was on the *Model System* activity from the IoTA-SRM framework. Participants were tasked with defining the system context and scope of the scooter use case (Appendix A - Section A.4), conducting system architecture and asset analysis, and analysing the security objectives of the assets in scope. The outcome was a security asset analysis report for the IoT use case. The second hackathon event, *Hackathon 2* in Figure 24, continued the trajectory by assigning tasks related to security threat elicitation, vulnerability assessment, and risk impact estimation using the analysis outcomes from *Hackathon 1*. Risk documentation was facilitated through provided risk templates, resulting in a security risk analysis report for the IoT use case. The final hackathon event, *Hackathon 3* in Figure 24, shifted focus to handling and treating risks identified during the *Discover Risks* activity. This event involved security requirements elicitation and role-based access control modelling tasks. Participants had to submit an overall security risk management report consolidating analyses from all three hackathon iterations.

Below, we discuss how each intervention was applied at each hackathon event:
Thematic Input. At the *Model System* hackathon, participants were introduced to base knowledge through lectures and lecture resources, aiding their reflection and application of knowledge to hackathon tasks. Lectures guided participants on deriving the IoT system's business assets, system assets, and security objectives. For the *Discover Risks* hackathon, a second set of lectures focused on risk-related concepts in software systems was introduced. The lectures guided vulnerability assessment and threat elicitation as outlined in the framework. Examples from the scooter use case (Appendix A - Section A.4) were also provided. The *Handle Risks* hackathon marked the final round of thematic input with lectures emphasising security requirements elicitation and role-based access control

modelling. These lectures aligned with the corresponding framework activity and tasks, preparing participants for the third hackathon.

Targeted Feedback. The teams submitted hackathon outcome artefacts at the end of the *Model System* hackathon. Mentors provided written feedback to the teams by evaluating hackathon outcome artefacts submitted. This intervention aimed to enhance comprehension of concepts and improve performance in subsequent hackathon events. During the *Discover Risks* hackathon, online consultation sessions were introduced, enabling real-time feedback and interactive engagement between mentors and participants in their teams. Presentation sessions allowed teams to discuss outcomes and receive feedback from mentors and peers. Mentors then provided written feedback to the teams by evaluating their hackathon outcome artefacts. For the final *Handle Risks* hackathon, online consultation sessions encouraged participants in their teams to discuss progress and challenges. Presentation sessions allowed mentors to provide verbal feedback on the team's work and guide the completion of their cumulative hackathon reports. Unlike earlier hackathons, no written feedback was given because this was the final event.

Following each hackathon event, we administered post-hackathon questionnaires to the participants, assessing their perception of the interventions' usefulness, contribution to learning, and satisfaction. We formulated the questionnaire (see Appendix E - Table 57) based on the scales used in the prior action research cycle Section 4.2.2. The questions were adapted to focus on the participants' perception of the hackathon interventions to foster learning, excluding the scales that measured their perception of their teams. Our questionnaire instrument covers various aspects of the post-hackathon evaluation, such as the usefulness of interventions, the level of agreement regarding participants' evaluation of the interventions, their contribution to learning, and learning outcomes. The included open-ended questions also allow participants to provide additional feedback on their intervention and learning experiences. We selected responses from six (7) teams with twenty-three (20) participants based on the team size (between two (2) and four (4) members) and teams who provided more complete responses to the questionnaires [67]. The selected team characteristics are summarised in Table 43.

Table 43. Action research cycle 3: Team characteristics for data analysis [67]

Teams	# of team members	Participants
Team A	2	A01, A02
Team B	3	B01, B02, B03
Team C	4	C01, C02, C03, C04
Team E	2	E01, E02
Team F	4	F01, F02, F03, F04
Team G	2	G01, G02
Team H	3	H01, H02, H03

To analyse data collected from the questionnaire instrument, we transformed

the Likert scales into a numerical format ranging from 1 to 5 to ensure data consistency and reliability. Findings of the post-hackathon questionnaire about the perception of learning from interventions by the hackathon participants are illustrated in Figure 25. All responses were given on a 5-point scale anchored between *strongly disagree* (1) and *strongly agree* (5).

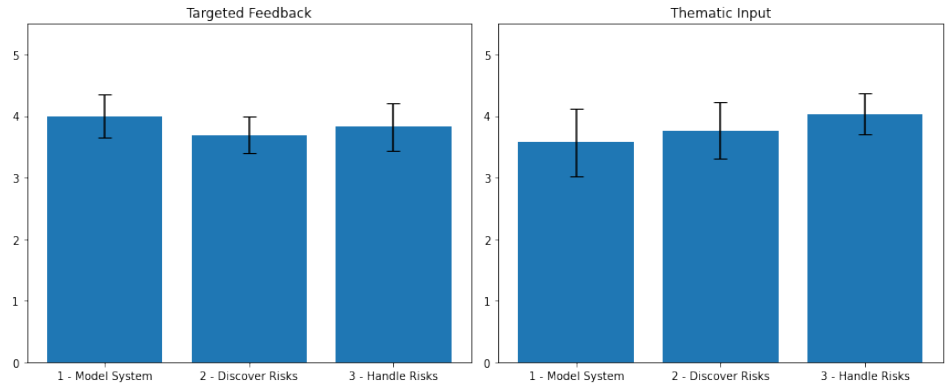


Figure 25. Questionnaire responses by participants about interventions at the hackathons in action research 3

The participants consistently indicated above-average perceptions of learning benefits across interventions. The thematic input intervention showed relatively stable positive perceptions with ($M = 3.58, IQR = 0.55$), ($M = 3.77, IQR = 0.47$), and ($M = 4.04, IQR = 0.33$) for the Model System, Discover Risks, and Handle Risks Hackathons, respectively. The targeted feedback intervention also consistently demonstrated positive perceptions ($M = 4.00, IQR = 0.36$), ($M = 3.69, IQR = 0.30$), ($M = 3.83, IQR = 0.39$) for the respective hackathons. Participants found targeted feedback particularly valuable in the Model System Hackathon, while benefits remained relatively consistent in the Discover Risks and Handle Risks Hackathons. Details on the team's journeys, findings of the differences between teams concerning their learning process, and the hackathon setting are documented in [67].

Our analysis of the open-ended questions in the questionnaire also provided contextual insights on the contributions of the hackathon interventions to security learning:

- **Thematic Input**: During the "Model System" hackathon, participants perceived the lecture intervention as valuable, although concerns about information overload and delivery clarity were raised. Some participants found the practice lectures especially beneficial due to their alignment with design-focused tasks. For instance, participants H03, H04, B04, and C03 appreciated how the lectures improved their knowledge, especially in areas new to them. H03 specifically stated, "*I liked the practice lectures more, especially since we have to design something*" (H03). This point was also reflected in the overall results. However, participants G04 and B03 found the

delivery chaotic and challenging. Similarly, B03 emphasizes that the system models became "*huge uncomprehensive models*" (B03), reinforcing the initial feedback that the content was difficult to manage. Participants in the "Discover Risks" hackathon reported higher learning gains from the lecture intervention. This perception could be attributed to participants recognising the importance of prior knowledge in effectively tackling complex tasks and participating in meaningful mentor-participant discussions. Participants C02 and C01 praised the lectures' rich content and emphasised their reliance on recorded lectures to enhance their understanding. Nonetheless, participant C02 suggested that more flexibility in consuming lecture content at their own pace would be beneficial. Throughout the "Handle Risks" hackathon, the perception of the thematic input intervention learning benefits increased consistently, indicating a consistent upward trend across all three hackathons. However, C02 pointed out the need for more detailed explanations of "*definitions and relations between security requirements and security controls*" (C02). This suggests that while the thematic input is improving, there is room for more in-depth coverage of specific topics.

- **Targeted Feedback**: Feedback was highly valued across hackathons. During the "Model System" hackathon, participants highly valued the detailed written feedback. The targeted feedback intervention received particular appreciation, with participants expressing gratitude for its comprehensive nature. Participants such as B04, C01, C03, and G04 found the feedback immensely helpful in improving their solutions. B04 appreciated the feedback, stating it "*can help improve my solution*" (B04), corroborating our findings. C01's comment that "*it is more important to get a bigger picture of the system*" (C01) adds a layer of nuance to the overall positive reception, suggesting that while detailed feedback is useful, there might be a need for deeper system-wide guidance. However, participants suggested that providing clearer task instructions might diminish the need for such extensive feedback. In the "Discover Risks" hackathon, written feedback, online consultation, and presentation feedback remained valuable. Participants consistently perceived written, online consultation, and presentation feedback as contributing to their learning. The written feedback's value was especially notable due to its detailed critiques and assistance in addressing weaknesses. C02 emphasised how written feedback substantially improved their understanding and helped rectify issues in their report. C02 confirmed this by emphasising how the detailed written feedback helped them "*better understand shortcomings*" (C02) found in their submitted report. However, H01 highlighted a lack of consistency in implementing the written feedback received from the "Model System" hackathon at this hackathon: "*I felt like I was putting more effort and tried to implement the feedback that we got. Others did not really listen to the feedback and thus there was some inconsistency throughout the work*" (H01). C03 commended online con-

sultation feedback for its ability to alleviate confusion. Even presentation feedback was regarded as beneficial, although a few participants recommended extended feedback times to enhance the quality of explanations. Participants found online consultation feedback particularly valuable at the "Handle Risks" hackathon. C03 noted that an online meeting effectively dispelled confusion about the task, while C02 maintained a preference for written feedback. The introduction of online consultation and presentation feedback interventions enhanced participants' learning experiences during this hackathon, something C03 confirms, saying, "*But after online meeting it was clear - answered our questions*" (C03).

Reflection. We discuss the lessons learned from the participant's perception of how the thematic input and targeted feedback interventions benefited the participants' learning process at the hackathon events. Details on our findings from which we derived these lessons are documented in [67]. The participant's perception of the thematic input and targeted feedback interventions remained consistently positive throughout the hackathons. Our findings showed that the thematic input intervention (lectures) is suitable for traditional educational approaches for knowledge transmission, while feedback intervention enables collaboration and knowledge-sharing. However, we found that the effectiveness of different types of interventions varied between hackathons, indicating that tailoring interventions to the specific needs of each hackathon is beneficial. These findings underscore the significant impact of thematic input and targeted feedback interventions in promoting learning within the hackathon setting. However, future implementations could benefit from further refinement of these interventions, including clear task instructions, appropriate pacing of thematic input, and varied forms of feedback to cater to diverse needs (summarised in Table 44).

Table 44. Suggestions to improve action research cycle 3 interventions

Intervention	Introduced as	Suggestions for Improvement
Thematic Input	Lectures delivered by course instructors	• Thematic input interventions should be designed to allow for offline access to lecture materials and encourage a self-paced learning format.
Targeted Feedback	Mentor support providing written, consultation and presentation feedback	• Providing multiple forms of feedback, such as consultation feedback, caters to the different needs of the hackathon participants. • Clarify and review hackathon tasks (i.e., through Q&A sessions) to ensure that mentor interactions are focused on learning-oriented activities • Provide clear instructions and goals for the hackathon tasks to help participants complete them

- **Thematic Input**: Thematic input, in the form of lectures, played a pivotal role across all hackathons, albeit with varying perceived effectiveness. The participants initially felt overwhelmed by the amount of information delivered through thematic input in the "Model System" hackathon. How-

ever, as the hackathons progressed to the "Discover Risks" and "Handle Risks" hackathons, participants began to appreciate the importance of upfront knowledge provided through these lectures. They found this knowledge particularly useful in tackling more complex tasks. Despite these benefits, participants received recurring feedback about the need for clarity in task instructions and goals to approach the hackathon tasks confidently. It was noted that the intensity of thematic input could potentially cause information overload. Therefore, more judicious pacing of thematic input is recommended, encouraged by providing offline access to lecture materials to foster a self-paced learning environment, allowing participants to digest the material at their convenience and as many times as necessary. This helps participants better absorb the content and reduce feeling overwhelmed.

- **Targeted Feedback**: Targeted feedback interventions proved instrumental across all three hackathons. Participants in the initial "Model System" hackathon reported higher learning gains from the targeted feedback, indicating its usefulness in rectifying potential misunderstandings of newly introduced concepts. This feedback strategy remained positively perceived in the following "Discover Risks" hackathon and "Handle Risks" hackathon, reinforcing its overall effectiveness. However, the participants' preference shifted toward thematic input, suggesting that the importance of targeted feedback might be context-dependent relative to specific learning objectives and the nature of the tasks in each hackathon. As with the thematic input, there were suggestions for varied feedback methods to cater to different needs. To improve the intervention and increase learning-oriented support, we suggest introducing more frequent mentor interactions during the hackathons and providing written feedback for each hackathon event. Participants preferred mentor interaction during the first hackathon, where written feedback was not provided until the end of the event. This might be due to the lack of precise scoping and task information, causing the participants to rely heavily on mentor interactions for clarifications. Implementing frequent consultation feedback opportunities could alleviate this issue, allowing mentor interactions to focus more on learning-oriented activities. Organising Q&A sessions after task allocation and before targeted feedback interventions might provide the needed clarification for the hackathon participants.

4.3. Discussion

This section discusses hackathon interventions as they evolve across action research cycles and additional considerations to hackathon learning outcomes following lessons learned from action research cycles (in Section 4.2.1, Section 4.2.2 and Section 4.2.3).

4.3.1. Hackathon Intervention Impact across Action Research Cycles

Across each action research cycle, we implemented several interventions. We tailored them according to the objectives and dynamics of each event, evolving based on participant feedback and the needs of each hackathon. The strategic adjustments and transformations of these interventions and their subsequent impacts indicate an effective optimization of hackathon organization for educational purposes. In cycle 1 (see Section 4.2.1), the idea generation, thematic input, targeted feedback, and competition-style interventions were integrated. The idea generation intervention, held before the main hackathon, provided a forum for brainstorming, team formation, and initial idea refinement. Meanwhile, the thematic input intervention furnished participants with critical knowledge of IoT security trends and practices, forming the cornerstone of their understanding. The targeted feedback and competition-style interventions stimulated participant engagement and offered a platform for learning through mentor guidance and healthy competition. In cycle 2 (see Section 4.2.2), the focus shifted more towards specific interventions that targeted the learning and practical application of security risk management principles. Thematic input was enriched with more specialised lectures on security risk management, and targeted feedback evolved to include online consultation sessions and peer feedback during presentations. However, the idea generation intervention was no longer necessary since use cases were predefined. By cycle 3 (see Section 4.2.3), the interventions had been refined further, focusing on applying knowledge to the specific domain of IoT systems. Thematic input involved targeted lectures on IoT security risk management following the IoTA-SRM framework. The targeted feedback, now a crucial aspect of the learning process, offered additional online consultation sessions and presentation feedback, which helped participants refine their final reports. Thus, we observe an evolution from a broad focus to a specialised and targeted approach to IoT security risk management learning to guide interventions necessary to optimise the educational outcomes of the hackathon events.

These findings provide valuable insights for hackathon organizers, educators, and researchers by demonstrating the value of adapting interventions to optimise hackathon outcomes for educational purposes. It underscores the importance of adapting interventions that cater to participants' evolving needs and learning goals, contributing to developing refined strategies for experiential and outcome-oriented learning experiences.

4.3.2. Hackathons as an Educational Tool: Interventions and Design Considerations

Our comprehensive findings contribute to understanding hackathons as powerful educational platforms, particularly apt for complex subjects like IoT Security Risk Management (SRM). In the initial stage, thematic input and targeted feedback interventions emerged as indispensable, resonating with the pedagogical notion

of feedback as a co-constructed, learner-centred process [246, 247, 248]. These interventions proved crucial in enriching understanding, although they required fine-tuning to prevent participant overwhelm [249]. Furthermore, we recognised the context-dependent utility of additional interventions like idea generation, collaboration support, and competition-style design. Idea generation shines when the emphasis is on fostering creativity and innovation, yet this creativity must be calibrated to maintain rigorous attention to IoT security details. The collaboration support intervention benefits heterogeneous teams, helping to address common challenges like freeriding and uneven work division [250, 251, 252]. It also synergizes well with targeted feedback using mentors [173]. Competition-style design, while motivating, can also intimidate and should thus be modulated according to participants' comfort and expertise levels [253, 254, 255]. Therefore, coupling competition with other interventions like targeted feedback [173] and thematic input to help prepare participants for competitions [179, 180] could provide a balanced and enriching learning experience. The flexibility and adaptability of these interventions allow for customization based on participants' specific needs, including those who prefer interactive, hands-on learning experiences (i.e., Gen Z [188]). Aligning these interventions with broader educational objectives affirms the model's flexibility and relevance across various learning contexts [256, 257, 174].

The introduced hackathon interventions equally shaped the hackathon format suitable for teaching about IoT SRM and incorporating the IoTA-SRM framework across the action research cycles. Given the complexity and volume of learning content, we found benefits in breaking down this content into manageable segments. We then structured individual hackathon events to align with these segmented units of learning [69, 67]. This approach facilitated the implementation of the thematic input intervention and made it easier to provide regular mentor guidance, thereby enhancing the effectiveness of the targeted feedback intervention. The need to accommodate the thematic input and targeted feedback interventions also highlights the need to extend hackathon durations (farther than the typical 24- to 48-hour hackathon timeframe). This is to allow ample time for learning modules and feedback sessions, thereby transforming the hackathon from a sprint into more of a marathon. This longer duration also gives participants the time to digest new information, apply it, receive feedback, and iterate—turning the event into a more comprehensive learning experience. This is similarly encouraged by Maaravi [258] and Wilson *et al.* [259], where recommendations were made to extend hackathon durations beyond the typical short durations to give the participants more time to design and validate their solutions, as well as a more relaxed learning environment. Lastly, evaluating each hackathon artefact is crucial for enhancing the effectiveness of our hackathon's thematic input and targeted feedback interventions, serving as a measure of participants' understanding and application of the IoTA-SRM framework. Furthermore, by incorporating and improving specialised frameworks like IoTA-SRM, the hackathon becomes more than a typ-

ical one-time event but a launchpad for ongoing learning and professional development. To facilitate this sustained learning [260], the hackathon format could include follow-up evaluations and resources, turning it into a stepping stone for continuous education.

4.4. Related Work

Previous research has shown that hackathons have been used for education and learning, and learning has been identified as a key motivator for participants to participate [173, 174, 175, 241]. However, it has also been acknowledged that the extent to which participants take advantage of learning opportunities cannot be controlled [183]. This underscores the need to design hackathon approaches that optimise learning potentials, thereby increasing the likelihood of effective learning outcomes.

In security risk management education, adopting practical-oriented strategies has been recommended to achieve desired learning results [69, 261]. Furthermore, creating opportunities for hackathon participants to engage in real-world scenarios, where security risk management techniques are directly applied, has been identified as a valuable teaching approach [261]. Our intervention-based hackathon learning approach aligns with prior works utilising hackathons to facilitate rapid learning experiences within educational contexts. For instance, La Place *et al.* demonstrated how hackathons foster quick learning-by-doing for engineering participants [262]. Tandon *et al.* explored educational hackathons as tools to enhance interest in STEM education, showcasing positive results in increasing participant interest and knowledge levels [186]. Additionally, Gama *et al.* documented online educational hackathons as resources to engage participants in developing semester projects [245].

Our approach extends these related works by advocating for hackathon interventions tailored to the course curriculum, ensuring participants encounter diverse learning opportunities as they progress through hackathon events and academic courses. We emphasise the need to design interventions deliberately to foster learning and enhance the overall learning potential of hackathon events. As such, our research contributes to the growing body of knowledge on hackathon-based learning approaches and their applicability in educational settings.

4.5. Summary

In this chapter, we addressed ($\mathbf{RQ}_2$) by adapting hackathon interventions to foster security learning and evaluating our intervention-based hackathon approach. It underscores the pivotal role such hackathon interventions play in fostering enhanced learning experiences, specifically for IoT security risk management.

We first adapted hackathon interventions suited to foster learning outcomes. Employing action research methodologies, we evaluated these interventions –

thematic input, targeted feedback, collaboration support, and competition style – across multiple hackathons. We analysed participants' perceptions of the interventions, revealing the beneficial yet nuanced impacts of these interventions. For instance, thematic input introduced as lectures emerged as a crucial learning catalyst, consistently garnered positive feedback across each action research cycle. Targeted feedback, given through mentor interactions and written feedback, facilitated an iterative learning process. Comparative analysis between different action research cycles showed that the benefits of these interventions remained consistently positive, irrespective of the specific hackathon context.

Additionally, we highlight additional factors that impact the hackathon experience. Segmenting complex learning material and prolonging the hackathon duration were identified as strategies that enhance comprehensive learning and project development. We also emphasised the significance of post-hackathon factors like continued access to learning resources and participants' independent motivation for sustaining learning gains. Our findings highlight the interventions' adaptability and applicability, encouraging application in educational contexts.

5. A FRAMEWORK AND TEACHING APPROACH FOR IOT SECURITY RISK MANAGEMENT

Integrating the IoTA-SRM framework with the intervention-based hackathon approach was motivated by a desire to create a comprehensive, effective, and practical educational experience for IoT security risk management. The IoTA-SRM framework, as discussed in Chapter 3, provides a systematic approach for assessing and managing security risks in IoT systems. Integrating this framework with the intervention-based hackathon approach in Chapter 4 can facilitate learning about SRM and better prepare learners for real-world IoT SRM challenges. It ensures that these learners understand the theoretical underpinnings of SRM as provided by the IoTA-SRM framework and know how to apply this knowledge in practice. This chapter is structured as follows: Section 5.1 sets the stage by explaining the rationale behind the integrated framework. We then perform an empirical analysis of learning outcomes in Section 5.2, specifically evaluating the application of the IoTA-SRM framework through security reports from hackathons. An in-depth discussion of these results and their broader research implications is presented in Section 5.4. Based on our findings, Section 5.3 introduces a hackathon teaching model, followed by guidelines for its implementation in Section 5.3.5. Section 5.5 reviews relevant literature, and the chapter concludes with a summary in Section 5.6.

5.1. Rationle for Integrated Framework and Teaching Approach

IoT security risk management (SRM) presents challenges intensified by the constant advancement of technologies and the specific nature of IoT components. Traditional educational methods often fall short of reflecting the real-world intricacies of IoT SRM. Integrating the IoTA-SRM framework with the hackathon approach aims to fill this educational void. The IoTA-SRM framework provides a method to assess and manage security risks in IoT systems. This approach centres on the nuanced understanding of IoT architecture and its unique characteristics [15, 6]. On the other hand, the intervention-based hackathon teaching method is rooted in hands-on experience and active participation. Combined with the IoTA-SRM framework, this approach fosters a learning environment that marries theoretical insights with practical exercises. It guides participants through the intricate landscape of IoT SRM, bridging the gap between theoretical constructs and real-world applications [7]. The fusion of these two methods facilitates a well-rounded learning experience that resonates with the growing needs of the IoT industry. It builds on understanding the complex relationships among devices and the risks that influence people, processes, objects, and data [12, 13, 14]. By merging established risk management principles with engaging, hands-on expe-

riences, this integration lays the groundwork for innovative teaching strategies in cybersecurity education.

5.2. Empirical Analysis of Integrated Approach

In this section, the primary focus is evaluating the participants' ability to apply the introduced IoTA-SRM framework by examining concrete learning outcomes reflected in the security reports produced during the hackathon events. While the insights gained in action research cycle 3 (Section 4.2.3) about the use of the IoTA-SRM framework focused predominantly on participants' perceptions, it lacked concrete analysis of actual learning outcomes. Therefore, we expand upon this by adopting an empirical approach. Specifically, we perform content analysis on the security reports produced by participants and assess the tangible learning outcomes using RBT.

5.2.1. Content Analysis

Our analysis is based on action research cycle 3 (Section 4.2.3). At the end of the hackathon events, teams were instructed to submit security risk reports. These reports were expected to focus on tasks within the IoTA-SRM framework. To preserve confidentiality, all submitted reports were anonymised before analysis. Of the nine (9) teams that submitted reports, we narrowed our analysis to seven (7) based on team sizes ranging from two (2) to four (4) members.

We adopt a content analysis approach to uncover thematic insights within the security reports. These themes were developed around the IoTA-SRM framework's activities, such as risk assessment, vulnerability identification, and threat analysis. Each text corresponding to an IoTA-SRM activity was tagged with its relevant activity code. For instance, segments related to risk assessment were grouped under the "Discover Risks" theme as seen in Table 45. We employed the Revised Bloom's Taxonomy (RBT) to enhance our analysis to categorise tasks based on cognitive thinking levels represented in Table 45.

This approach allowed us to understand the depth of participant engagement and the cognitive demands associated with each task within the IoTA-SRM framework. For instance, the tasks under the "Model System" activity. The **decompose IoT system into IoT layers** task requires participants to analyse the architecture of an IoT system and break it down into its constituent layers. Analysing at this level involves examining the structure and components of a system, which is precisely what this task entails. On the other hand, the **define security objectives for business assets per IoT layer** task falls under the *evaluate* cognitive level, as it involves assessing the suitability of security objectives within each layer. This requires participants to make informed judgments and evaluations based on established criteria. Moving on to the "Discover Risks" activity, the **multi-layer vulnerability assessment** task is categorised as *apply* since participants must use their knowledge to assess vulnerabilities across different layers of the IoT system.

Activity	Tasks	Bloom's Level
Model System	Decompose IoT system into IoT layers	Analyse
	Identify system and business assets for each IoT layer	Apply
	Define security objectives for business assets per IoT layer	Evaluate
	Model decomposed system	Create
Discover Risks	Multi-layer vulnerability assessment	Apply
	Multi-layer threat elicitation	Analyse
	Multi-layer risk impact estimation	Evaluate
Handle Risks	Multi-layer risk treatment decision	Evaluate
	Security requirements elicitation	Apply
	Control selection	Evaluate
	Control implementation	Apply

Table 45. Content analysis: Thematic coding

Similarly, the **multi-layer threat elicitation** task involves participants analysing potential threats for each layer, aligning with the *analyse* cognitive level. The **multi-layer risk impact estimation** task requires participants to evaluate the potential impact of risks, corresponding to the *evaluate* level of cognitive skills. In the "Handle Risks" activity, the **multi-layer risk treatment decision** task falls under the *evaluate* cognitive level, as participants decide, critically assessing and judging the relevance of those risk treatment decisions. The **security requirements elicitation** task falls under the *apply* cognitive level, as participants create a list of security requirements by applying learned concepts. The **control selection** task involves analysing and selecting appropriate controls for risk mitigation and making decisions and assessments on suitability based on criteria corresponding to the *evaluate* level. Finally, **control implementation** task is classified as *apply* since participants directly implement controls based on evaluations from the **control selection** task.

Our thematic coding represented in Table 45 was carefully reviewed and validated to ensure that it accurately reflected the cognitive thinking level of the corresponding IoTA-SRM activities. We analysed each theme to discover patterns and variations supported by relevant quotes and excerpts from the reports.

5.2.2. Results

We evaluate the contents of the participant reports showing learning benefits from the integrated approach in applying the IoTA-SRM framework and accomplishing the hackathon tasks.

Model System. When tasked with the **decomposition of the IoT system into its layers**, five out of the seven teams (Teams B, C, E, G, and H) effectively broke down the given IoT system use case into its core layers: perception, network, and application revealing a comprehension of the IoT system architecture and its application for asset classification. However, Teams A and F adopted an alternative strategy, broadly categorising assets. This approach seemed to stem from potential

confusion between the terms “Intelligent Transportation Systems (ITS)” and IoT, which may have led to reluctance in utilising the IoT architecture layer. Teams A and F seemed to refer to other techniques introduced in the course, such as the system description format presented in [118].

In the subsequent **identification of system and business assets for each IoT layer** task, several teams (Teams B, C, E) demonstrated their ability to recognise and categorise assets within each layer. They delved into these layers and the associated business assets, showcasing their strong grasp of the concept. While Team B focused on the application layer, Team G and H introduced an architecture layer but omitted the detailed asset breakdown. Interestingly, Teams A and F effectively identified assets despite using an alternate methodology. In the **define security objectives for business assets per IoT layer** task, all teams succeeded in establishing security objectives for business assets. However, contextualising these objectives within specific IoT layers was often lacking.

For the final **model decomposed system** task, all teams effectively constructed models that depicted the defined IoT system scope. An example of the model system tasks from the participants’ reports is shown in Figure 26. These models

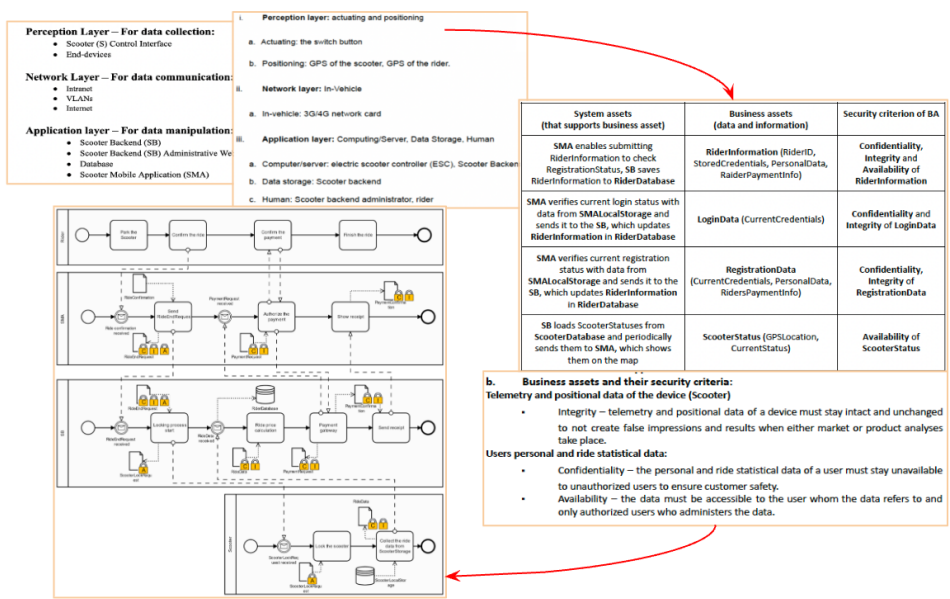
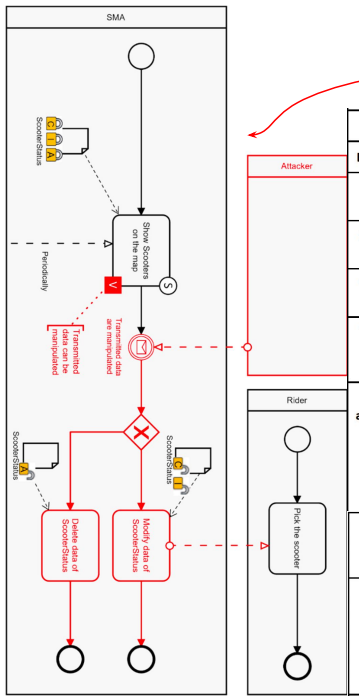


Figure 26. Content analysis: Model system activity example

provided insights into both functional and behavioural aspects. However, some models exhibited syntactic inaccuracies and failed to capture interactions within each IoT layer, leading to the omission of crucial contextual details.

Discover Risk. During the **multi-layer vulnerability assessment** task, teams showcased their capacity to identify vulnerabilities by applying their knowledge, as derived from the model system activity. However, reports indicated an under-

emphasis on assessing vulnerabilities from a multi-layer perspective. In the **multi-layer threat elicitation** and **multi-layer risk estimation** tasks, teams moved beyond merely understanding risk concepts. They applied their knowledge to real-world contexts, examining potential vulnerabilities, threat exploits, and the resulting risk impacts. An illustration of discover risks activity from the teams' reports is provided in Figure 27. The most commonly cited vulnerabilities were a lack of



Risk ID	N2
Business asset	ScooterStatus
Security criteria	**Confidentiality, Integrity and Availability** of ScooterStatus
System asset	**SB** periodical communication with SMA , **SMA** event of Show Scooters on the map, **ScooterDatabase**
Vulnerability	**Incorrect map representation** of the **ScooterStatus** real-time data in the **SMA** can lead to a malfunction of the picking the scooter function by the rider.
Threat agent	**Thief; Undergraduate or Vandal; Script Kiddy** who can capture and modify transmitted data, forward insecure connection using MitM attack or **botnet** which could lead to jamming the communication between SB and SMA.
Threat and attack method	An attacker **can manipulate the transmitted data** between the SB and the SMA using a man in the middle attack to affect real scooters location and availability. 1. Exploring usable vulnerability in the network traffic (used protocols, needed tools and skills) 2. Perform a most suitable attack of jamming or rerouting transmitted data of ScooterStatus. (DDoS attack, MitM with DNS spoofing, Network protocol attack)
Impact	Affecting the **confidentiality, integrity** or **availability** of correct and accurate **ScooterStatus** data to the Rider, which could lead to non-usable SMA for Riders and losing clients of the service.
Risk	An undergraduate thief or script kiddy can manipulate the transmitted data between the SB and the SMA using a different attack scenarios as man in the middle attack or denial of service attack, which can affect the confidentiality, integrity or availability of correct **ScooterStatus** data about the location and state of the Scooter, resulting in a failure of the Rider's ability to pick the scooter and losing clients.

Figure 27. Content analysis: Discover risks example

input validation, weak authentication mechanisms, code-related vulnerabilities, such as insecure code and the lack of protection against Denial of Service (DoS) attacks. Teams likewise defined resulting threats as a result of the vulnerabilities. For example, potential threats regarding information disclosure were mentioned due to unsecured network traffic and lack of data encryption vulnerabilities, as well as threats related to privilege escalation due to poorly implemented access control.

Teams (H, C, B) stood out in recognising threats and understanding how threat agents might exploit them. Their comprehensive risk analyses identified specific assets, vulnerabilities, threat agents, threats, attack methods, and potential risk impacts. However, even among these top performers, assessments often didn't fully assess risks from a multi-layer standpoint. While Teams C and B used a layer-based approach during the model system activity, highlighting their grasp on IoT architecture, they didn't always explore the implications of identified risks across different layers. Although their reports considered selected IoT layers and risk

impacts, they could still enhance their depth of risk analysis. On the other hand, Teams (G, F, E, A) also pinpointed threats, but still, their evaluations could benefit from greater depth and a thorough consideration of multi-layer perspectives. Notably, each team performed risk modelling on their asset models, illustrating the threat posed and the impact of a risk event.

Handle Risks. In the context of the **multi-layer risk treatment decisions** task, Team C uniquely showcased their proficiency in evaluating potential risk treatments, demonstrating their competence in risk mitigation decision-making. In contrast, the other teams did not explicitly engage with this task in their reports. Though teams A, B, and F showcased a solid grasp of foundational concepts by outlining security requirements and control selections, they omitted discussions on risk treatment decisions such as avoidance, reduction, transfer, or retention. Similarly, Teams G, C, and E faced similar challenges and did not explicitly delve into risk treatment decisions. This omission appears to stem from the assumption that risks were generally to be mitigated, implying no further need for explicit risk treatment decision discussions. Team H indicated their decision to mitigate all identified risks through reduction measures. However, they did not provide a rationale for this particular choice.

In the **security requirements elicitation** and **control selection** tasks, participants were evaluated based on their ability to compile a comprehensive list of security requirements. Leveraging their understanding of risks and security needs, teams effectively demonstrated their analytical skills in selecting appropriate controls. Noteworthy is Team G's introduction of a layered perspective; however, they didn't delve into the intricate interplay of how risk treatments might impact different layers. Conversely, Teams C, E, and H encountered challenges when eliciting security requirements and deciding controls, often due to articulation issues. Mistakes in articulating security requirements underscored the need for a deeper understanding, highlighting the importance of distinguishing between these concepts. For instance, Team C's elicitation of security requirements, such as "*The Scooter must not allow Rider to read RideData*," and their corresponding security control, "*The Scooter shall encrypt RideData*," exhibited a tangential aspect. Although potentially fitting, encryption was presented as a security requirement. An appropriate control statement, in this case, would have been "*Implement end-to-end encryption for RideData transmission between the Scooter and Rider's mobile application*". An illustration of security requirements and control selection tasks from a selection of the teams' reports is provided in Figure 28.

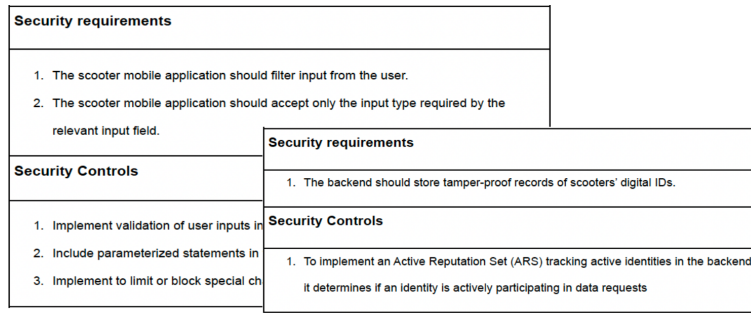


Figure 28. Content analysis: Handle risks example

5.2.3. Lessons Learned

We discuss the lessons derived from our analysis of how well the IoTA-SRM framework worked in a hackathon setting to teach IoT security risk management. To discuss our lessons learned, we incorporate both content analysis results and open-ended questionnaire responses from action research cycle 3 (Section 4.2.3), painting a fuller picture of the team's approach to achieving the IoTA-SRM framework tasks and achieving learning outcomes during the hackathons. We also identify areas for improvement, not only within the framework of the IoTA-SRM but also in the thematic input and targeted feedback interventions.

Table 46 summarises each team's achievement of the cognitive thinking levels for different tasks and activities. It highlights overall trends, areas of strength or improvement for each team, and the cognitive thinking levels they exhibited in their performance. Teams B, C, and H performed better than others, showcasing consistent achievement of higher cognitive thinking levels. The "Discover Risk" framework activity was more comprehensively covered and achieved by all teams. In contrast, the "Handle Risks" framework activity, specifically the aspect of risk treatment decisions, was less emphasised in the reports. However, it's important to note that each team's performance varied across tasks, and some teams excelled in specific tasks while facing challenges in others.

Model System. The content analysis suggests a good understanding of this activity among the teams. A foundational understanding of IoT system decomposition and asset identification was a common success among teams. Teams that effectively analysed the IoT use case to break down systems into layers and *apply* knowledge gained to identify and classify (*evaluate*) assets demonstrated a solid starting point for subsequent tasks. Teams could also *create* asset models indicating their capability to represent analysed constructs. However, participant feedback through open-ended questionnaires exposes certain nuances to the challenges discovered. For instance, participant B03 pointed out the hindrance posed by the activity's complexity, leading to "*huge, uncomprehensive models*" (B03). This calls for tasks introduced with clarity and manageability. Similarly, participant C01 raises the issue that working on a limited scope without focusing on the

Table 46. Analysis of team achievement of RBT cognitive thinking levels

Tasks	Bloom's level	A	B	C	E	F	G	H
Model System								
Decompose IoT system into IoT layers	Analyse	(+_)	(++)	(++)	(++)	(+_)	(++)	(++)
Identify assets for each IoT layer	Apply	(+_)	(++)	(++)	(++)	(+_)	(+_)	(+_)
Define security objectives per IoT layer	Evaluate	(+_)	(++)	(++)	(+_)	(+_)	(+_)	(+_)
Model decomposed system	Create	(+_)	(+_)	(+_)	(+_)	(+_)	(+_)	(+_)
Discover Risk								
Multi-layer vulnerability assessment	Apply	(++)	(++)	(++)	(++)	(++)	(++)	(++)
Multi-layer threat elicitation	Analyse	(+_)	(++)	(++)	(+_)	(+_)	(+_)	(++)
Multi-layer risk estimation	Evaluate	(+_)	(++)	(++)	(+_)	(+_)	(+_)	(++)
Handle Risks								
Multi-layer risk treatment decisions	Evaluate	(–)	(–)	(+_)	(+_)	(–)	(+_)	(++)
Security requirements elicitation	Apply	(++)	(++)	(+_)	(+_)	(++)	(++)	(++)
Control selection	Evaluate	(++)	(++)	(+_)	(+_)	(++)	(++)	(+_)

$[++]$ Fulfilled, $[+-]$ Fulfilled with limitations, $[-]$ Not fulfilled

bigger picture of the system distorts or limits the teams' output. C01 highlighted that "*it is more important to get a bigger picture of the system*" (C01). This implies that even if a team has the technical skills to complete a task, the way they approach it – based on their understanding or perception of its scope – influences the quality and completeness of their output.

Discover Risks. The content analysis suggests a good understanding of this activity among the teams. Teams that excelled in threat analysis, such as Teams B, C, and H, showcased a deeper comprehension of security risks. Their ability to identify specific assets and *apply* knowledge in vulnerability identification, and *analyse* threats based on vulnerability information, threat agent information and attack methods, and *evaluate* potential risk impacts indicated higher expertise. Starting with the multi-layer vulnerability assessment task, it was observed that teams were generally proficient in identifying vulnerabilities, threats and risks. However, the depth of multi-layer analysis was often missing. For instance, F04 mentioned looking for similar examples to the lecture content online to attempt their tasks. This suggests that while the lectures were helpful, they may need more contextual examples for multi-layered analyses. Regarding the multi-layer threat elicitation and risk estimation tasks, Teams H, C, and B exhibited skills in identifying security threats. However, their analysis often fell short of considering the full implications of risks across various layers. Participant C02 noted the detailed feedback about risk definitions was "*especially helpful*" (C02) for improving their report. This feedback might explain why these teams excelled in specific areas. But it also raises a question about whether more targeted feedback focused on multi-layer analysis could have improved performance even further. However, H01's feedback highlights a lack of consistency in implementing the feedback received at this hackathon, suggesting that while some team members might have tried to work on the suggestions identified during the feedback sessions, not everyone did so. This lack of collective action could be one reason the assessments often failed to cover all aspects of the tasks.

Handle Risks. The content analysis suggests a good understanding of this activity among the teams. Some teams demonstrated proficiency in compiling (*apply*) comprehensive security requirements and analysing and choosing (*analyse*) appropriate controls. This skill highlighted their analytical thinking and ability to devise effective risk mitigation strategies. Team C stood out by explicitly engaging with various potential risk treatment decision-making. On the other hand, most other teams failed to address this explicitly, focusing mainly on risk mitigation without considering other forms of risk treatment like avoidance, transfer, or retention. Participant C02's feedback implies that the challenges teams faced may be attributable to a lack of clarity in the lectures (thematic input), leading to less comprehensive reports. Most teams demonstrated their analytical skills regarding security requirements elicitation and control selection tasks. However, Teams C, E, and H struggled with these tasks. Team C, for instance, confused the requirement and control, illustrating a misunderstanding that C02 also identified in their feedback about the lectures. C02 suggested that "*the definitions and relations between security requirements and security controls could be explained in more detail in lectures because [handle risks hackathon] presentation showed that most teams had issues with that*". Team G attempted to introduce a layered perspective in their analysis but did not explore the interplay between risk treatments across different layers. This aligns with the broader issue of teams not fully grasping the multi-layer implications.

Recommendations. Based on lessons learned, we highlight recommendations for key challenges observed. First, teams were often focused on one layer, failing to consider how vulnerabilities and threats propagate through an interconnected system and overlooking risks from a multi-layer perspective. Enhancing multi-layer considerations emerged as a recommendation, urging teams to consider the impact of risks across all three IoT architecture layers – perception, network, and application layers even when the analysis is scoped to one layer. This ensures careful consideration of assets and security considerations, preventing the oversight of potential risks in any layer. Secondly, teams frequently neglected to make explicit risk treatment decisions based on their analyses, focusing solely on risk mitigation decisions. A recommendation for future applications of the framework is to stress the necessity of risk treatment options, such as avoidance, transfer, and mitigation and be guided on how to make these decisions based on their findings. The goal is to guide teams towards making more informed and comprehensive risk treatment decisions. Lastly, the challenges in accurately eliciting security requirements underscored the need for additional support in formulating well-defined requirements. This is addressed through enhanced conceptual understanding. These could include practical examples and additional resources to aid in applying these concepts.

Limitations of the Study. The first limitation pertains to the exclusive reliance on written security risk reports from hackathon teams for analysis. To address this, we standardised language and expression by providing reporting guidelines,

thus reducing the scope for variability and misinterpretation. Moreover, as the hackathon was part of an educational course, participants were expected to adhere to educational standards in their reporting, further enhancing the data's uniformity and reliability. Importantly, the findings of this study are bounded by the specific circumstances under which they were obtained – namely, hackathon events designed to teach the IoTA-SRM framework. Thus, the generalizability of these findings to other educational settings or frameworks remains an open question. Lastly, the content analysis process is subjective, an intrinsic limitation of the methodology and should be considered when evaluating the results.

5.3. Hackathon Teaching Model for IoT Security Risk Management

We outline a hackathon teaching model visually depicted in Figure 29 that offers a structured avenue for teaching IoT security risk management by immersing participants in practical, hands-on experiences. This model formalizes integrating the IoTA-SRM framework with the intervention-based hackathon approach, ensuring a structured and hands-on learning experience. The framework defines the "what" (content), the interventions offer the "how" (delivery method), the hackathon cycles create an environment for framework delivery and application, and the evaluations of learning outcomes confirm if the learning objectives are met. Each core

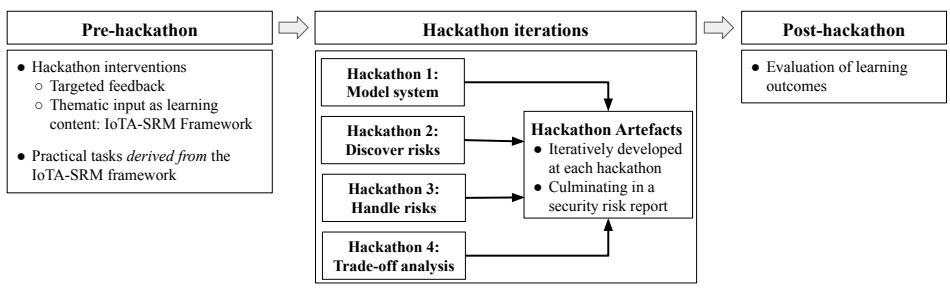


Figure 29. Conceptual model of the hackathon teaching model [67]

activity of the IoTA-SRM framework is delivered as independent modules through multiple hackathon iterations. These activities align with the framework tasks, resulting in improved hackathon outcomes. While our model has been tested in an academic setting with students as hackathon participants, it is equally applicable to a broader audience, including any IoT SRM stakeholders or potential IoTA-SRM framework users. In the subsequent sections, we delve into the key components of the hackathon teaching approach – iterative hackathon format, learning content, interventions (mode of delivery), and hackathon artefacts.

5.3.1. Iterative Hackathon Format

The hackathon model includes multiple iterations, each building upon the previous one. These iterations provide continuous learning benefits to the participants. Participants engage in practical tasks in each iteration aligning with the specific module. The outcomes of these tasks are hackathon artefacts that are iteratively improved and refined over time. The IoT security risk management framework in Chapter 3 covered four major activities: *Model System*, *Discover Risks*, *Handle Risks*, and *Analyse Tradeoffs*. These activities provided a basis to split learning about security risk management into standalone yet connected components, forming the iterative container of our learning process (i.e., an instance of a hackathon). Thus, within each iteration, we learn from the IoT security risk management framework and knowledge of the system context to refine the course lecture content and practical tasks introduced through interventions. Each iteration creates artefacts and contributes to building secure IoT systems (as a learning outcome).

5.3.2. Learning Content (IoTA-SRM Framework)

The IoTA-SRM framework is the foundation for the learning content in the intervention -based hackathon approach. Utilising hackathons as a teaching approach offered an innovative approach to teaching and comprehending intricate IoT security risk management concepts. Table 32 overviews the IoTA-SRM framework activities and their associated tasks. In addition, the IoT domain use cases supplement the IoTA-SRM framework to enrich the learning experience by setting the real-world IoT context where participants apply their knowledge and skills in practical settings and develop a deeper understanding of IoT security risk management challenges and solutions. The iterative nature of the hackathon teaching model aligns well with the IoTA-SRM framework. Each iteration emphasizes specific activities, breaking them into manageable components for in-depth exploration. This approach enhances comprehension and allows participants to build knowledge throughout the hackathon stages, resulting in hackathon artefacts, which are evaluated to assess the achievement of learning outcomes.

Clear learning outcomes are outlined through the IoTA-SRM framework tasks. Evaluation of learning outcomes is systematically aligned with Revised Bloom's Taxonomy (RBT) levels [57], which we introduced in Section 1.3.4. Tasks within the IoTA-SRM framework are mapped to corresponding RBT cognitive levels (Table 47). Priority is assigned to the highest cognitive level engaged by each task, as recommended in prior studies [61]. This categorization not only elucidates the specific cognitive skills required for effective engagement but also allows for a nuanced assessment of participants' learning outcomes. Given the complex nature of security risk management, none of the IoTA-SRM tasks are confined to the lower-order RBT categories of "Remember" and "Understand" levels. Instead, tasks typically require higher-order cognitive skills like analysis, evaluation, and

creative problem-solving. They are ideally suited for the hackathon format, which intrinsically encourages these advanced cognitive engagements.

Table 47. Assessing IoTA-SRM tasks using RBT

Activity	Task	Bloom's Level	Bloom's Assessment
Model System	Decompose IoT system into IoT layers	Analyse	Assess ability to analyse and break down a complex IoT system into its fundamental layers, demonstrating their understanding of system architecture.
	Identify system and business assets for each IoT layer	Apply	Assess capability to apply their understanding to identify essential assets within each layer, demonstrating their comprehension of asset categorization.
	Define security objectives for business assets per IoT layer	Evaluate	Assess proficiency in evaluating and determining security objectives for each identified asset, showing their grasp of risk prioritization.
	Model decomposed system	Create	Assess ability to create or construct a model of the decomposed system, indicating their capability to represent complex structures.
Discover Risks	Multi-layer vulnerability assessment	Apply	Assess skills in applying their knowledge to identify vulnerabilities, gauging their ability to recognise potential weaknesses.
	Multi-layer threat elicitation	Analyse	Assess aptitude to analyse the system and identify potential threats, assessing their comprehension of risk identification.
	Multi-layer risk impact estimation	Evaluate	Assess proficiency in evaluating the potential impact of identified risks, demonstrating their understanding of risk severity.
Handle Risks	Multi-layer risk treatment decision	Evaluate	Assess evaluation of potential risk treatments, assessing their decision-making skills in risk mitigation.
	Security requirements elicitation	Apply	Assess capability to create a list of security requirements, demonstrating their understanding of security needs.
	Control selection	Evaluate	Assess analysis and selection of the most suitable controls, indicating the ability to make informed decisions.
	Control implementation	Apply	Assess proficiency in applying selected controls effectively in the system, showing their application skills.
Analyse Trade-offs	Determine asset values from the Model Risk activity	Analyse	Assess ability to analyse and assign values to assets, gauging their quantitative analysis skills.
	Estimate risk impact values from the Discover Risks activity	Apply	Assess aptitude to apply estimation techniques, demonstrating their analytical skills.
	Estimate selected controls costs from the Handle Risks activity	Analyse	Assess ability to analyse and estimate the cost of implementing chosen controls, indicating their financial analysis skills.
	Run cost vs benefit analysis for risk reduction	Evaluate	Assess capability to evaluate benefits against costs and make informed decisions, showcasing their decision-making skills.

5.3.3. Interventions (Mode of Delivery)

To optimise the learning experience within the hackathon context, hackathon interventions are crucial in delivering the learning content and supporting participants' learning-by-doing approach. In Chapter 4, we introduced intentional design actions as hackathon interventions to enhance participants' learning experi-

ences.

We, thus, incorporate insights from our experiences with hackathon interventions to highlight two hackathon interventions crucial to our model: thematic input and targeted feedback.

- **Thematic Input**: The thematic input intervention is a core component of the hackathon, designed to impart essential knowledge to participants. Thematic input sessions optimise the comprehension of intricate topics across multiple hackathon cycles by breaking the learning content into manageable modules. The thematic input is instrumental in demystifying the IoTA-SRM framework, making it accessible and relatable to a diverse audience, including students, professionals, and other IoT system stakeholders. According to our findings, these thematic sessions serve as a vital link between the theoretical foundations of the IoTA-SRM framework and its practical implications, enhancing the framework's broader applicability in real-world settings.
- **Targeted Feedback**: Besides thematic input, mentors—ranging from course instructors to industry experts in a professional setting—deliver targeted feedback to participants. The knowledge level of mentors chosen by hackathon organizers should ideally be high, particularly in IoT security risk management and the specificities of the IoTA-SRM framework. Mentors should be experts in the security field, be capable of providing insightful, targeted feedback and have the ability to contextualise theoretical knowledge into practical applications. In a professional environment, mentors could be senior cybersecurity experts, industry professionals with hands-on experience in IoT security risk management, or academics who have researched or contributed to developing frameworks similar to IoTA-SRM. These mentors bring real-world perspectives into the learning environment, enhancing the material's practicality and applicability. Additionally, mentors' feedback contributes to assessing hackathon outcomes to understand participant achievement of expected learning outcomes.

  This feedback should be delivered through various channels, such as online consultations, hackathon outcome presentations, and asynchronous written feedback. The objective is to navigate participants through the intricacies of applying the IoTA-SRM framework. Targeted feedback transforms the hackathon from a mere exercise into a tailored, interactive learning journey, equipping diverse learners—from students to professionals and other stakeholders—with the skills and understanding they need to implement the IoTA-SRM framework.

Still, other hackathon interventions can be applied as the organiser deems necessary within the context of the hackathon format.

5.3.4. Hackathon Artefacts

Throughout each iteration of the hackathon, participants actively interact with the IoTA-SRM framework, creating tangible artefacts. These artefacts are concrete representations of their application to the framework's activities. Table 32 offers an overview of the hackathon artefacts that are expected when using the IoTA-SRM framework. In the *model system* activity, the anticipated artefacts encompass an asset list, security objectives for these assets, and an asset model. In the *discover risks* activity, the expected artefacts include lists of vulnerabilities and threats, along with information about the impact of identified risks. Within the *handle risks* activity, participants are expected to produce artefacts like risk decisions, security requirements, and a list of selected controls. Lastly, the *analyse tradeoffs* activity is expected to yield outcomes such as values for assets, risk impact, and control costs, along with a prioritised list of risks to guide control implementation. While the aforementioned artefacts are central and included in Table 32, participants might also generate additional artefacts, such as risk models and updated asset models following control implementation. Nonetheless, the suggested artefacts in Table 32 represent the minimum requirements for the framework. The collection of these artefacts culminates in creating a comprehensive security report. This report serves as a means to evaluate participants' learning accomplishments after the hackathon cycles.

5.3.5. Implementation Guidelines for Organizers

Drawing from our accumulated experiences, guidelines for implementing the hackathon teaching model in IoT security risk management education using the IoTA-SRM framework are formalised in Table 48. These suggested guidelines encompass the pre-hackathon (1), hackathon cycles (2, 3, 4, 5), and post-hackathon (6) components of the hackathon teaching model.

5.4. Discussion

The section presents a comprehensive discussion of contributions from our findings derived from applying the IoTA-SRM framework and hackathon approach in fostering IoT security risk management education, alongside the formalization of the hackathon teaching model.

5.4.1. Applicability of the Hackathon Teaching Model Across Target Groups

While our hackathon teaching model was initially piloted and evaluated on students with intermediate cybersecurity knowledge, its flexible design allows for customization and application across diverse learners. This includes not only students from various educational levels but also professionals and other stakehold-

Table 48. Organiser guidelines to implement the hackathon teaching model

	Activity	Description
1	Pre-hackathon preparation	In this phase, careful planning sets the foundation for a successful hackathon. • Clarify learning objectives: Define the hackathon's goals and learning outcomes, ensuring alignment with the IoTA-SRM framework • Resource preparation: Prepare thematic input and the IoT context to apply the IoTA-SRM framework and relevant resources to support participants during the hackathon • Task design: Design a series of tasks that encompass the IoTA-SRM framework's activities • Targeted feedback preparation: Organise and prepare mentors as experts in the field to guide and provide feedback to participants during the hackathon
2	Hackathon implementation	During this phase, participants are introduced to the hackathon's framework and key concepts. • Thematic input: Introducing the IoTA-SRM framework, explaining its components, layers, and objectives. Conduct thematic sessions to cover each IoTA-SRM activity • Team formation: Where needed, form diverse teams, encouraging a mix of skills and backgrounds
3	Task implementation	As participants transition to the core of each hackathon, they engage with tasks that put their knowledge into practice. • Task distribution: Introduce participants to the hackathon tasks, highlighting their connection to the IoTA-SRM framework • Team collaboration: Where needed, implement interventions such as a team management plan for effective collaboration. • Targeted feedback: Establish a feedback process where mentors or experts offer guidance and clarification as participants work through tasks during each hackathon
4	Reporting and documentation	In this phase, participants document their work for comprehensive reporting. • Report requirements: Instruct participants to document their findings in comprehensive reports. Recommend a structured report format that aligns with the IoTA-SRM framework's activities, ensuring consistency
5	Feedback	Mentors or facilitators offer participants constructive feedback on their solutions. • Feedback criteria: Develop clear criteria to assess the quality of participants' analyses, risk assessments, and solutions, aligning with framework objectives. • Quality and depth: Establish a process where mentors systematically evaluate participants' comprehension and application of the framework, using predefined criteria at the end of each hackathon cycle • Reflection: Encourage participants to reflect on their work based on feedback, enhancing their understanding of how framework activities connect and contribute to IoT security risk management principles
6	Post-hackathon analysis	In the aftermath of the hackathon, a post-event reflection and continuous improvement concludes the hackathon cycle. • Evaluation outcomes: Analyse participants' reports to identify common challenges, successful approaches, and areas for improvement. Review mentors' feedback to understand participants' strengths and weaknesses in applying the IoTA-SRM framework • Lessons learned: Reflect on the hackathon's effectiveness, considering participant feedback and outcomes • Refinement: Use insights from the analysis to refine future hackathon designs, addressing challenges and enhancing learning experiences

ers who may be interested in applying the IoTA-SRM framework in their work contexts. The modular nature of the teaching model lends itself well to adaptations for professional training sessions, lifelong learning courses, or specialised workshops. This makes it a viable educational tool for any prospective user of the IoTA-SRM framework, offering them the hands-on experience and practical skills necessary for effective implementation. The scope and adaptability of the hackathon teaching model are expandable to cater to a broader audience, ensuring that the practical insights and skills offered by the IoTA-SRM framework are accessible to all who seek to use it. Students studying cybersecurity or related fields can benefit from hands-on experience and practical skills in IoT security risk management that traditional learning avenues often can't provide. Those already working in cybersecurity can also deepen their understanding of IoT security risk management by applying the framework practically within their learning context. Lastly, IoT system stakeholders, organizational stakeholders, and decision-makers can comprehensively understand IoT security risk management within this practical and collaborative environment, enhancing their ability to make informed decisions for the IoT system within their business context. Each group gains from comprehending and employing the IoTA-SRM framework in their respective domains.

5.4.2. IoTA-SRM Adaptability

In addition to our initial evaluation of the IoTA-SRM framework in Chapter 3, we evaluated the IoTA-SRM framework's applicability and utility by analysing participant reports and outcomes. It was evident that participants effectively employed a structured approach to identify and analyse security risks and vulnerabilities inherent in IoT systems. This approach facilitated an understanding of the fundamental concepts and equipped participants with practical skills relevant to real-world scenarios. A notable observation was the consistent application of the IoTA-SRM framework across teams during the hackathons. This pattern of consistent utilization not only underscores the framework's repeatability but also highlights its reliability. The generated reports exhibited similarities in structure, content, and analytical approaches, accentuating the repeatable results achieved through the hackathon teaching model. This validation process confirms the framework's practical relevance and reinforces its capacity to guide systematic, consistent, dependable security risk analysis within the IoT domain. The implications of this validation extend to researchers seeking guidance for future IoT security risk management methodologies. By drawing inspiration from the successful outcomes of integrating the IoTA-SRM framework with the hackathon teaching model, researchers can lay the foundation for standardised methodologies in security risk assessment. This has the potential to foster a cohesive and uniform approach to addressing the ever-evolving security challenges presented by IoT systems. The flexibility and adaptability of the IoTA-SRM framework

are significant attributes, particularly in the context of real-life scenarios encountered during hackathons. Given that these scenarios often encompass a range of scopes and diverse requirements, the framework's ability to be domain-agnostic proves invaluable. Furthermore, the framework's integration potential with established methodologies such as STRIDE [189] for threat modelling, OCTAVE Allegro [112] for formal documentation and risk estimation, and Unified Modeling Language (UML) [263] for system modelling highlights its versatility. To ensure the ongoing improvement of the IoTA-SRM framework, the systematic integration within the hackathon teaching model offers an iterative mechanism for improvement. Collecting and incorporating participant feedback into subsequent framework iterations provide a valuable avenue for refining its application. This iterative process fosters a framework that remains adaptive, responsive, and in sync with the evolving landscape of IoT security challenges and solutions.

5.4.3. Advancing Cybersecurity Education Strategies

Using the IoTA-SRM framework with the hackathon teaching model introduces an approach to IoT security education. This shows the usefulness of hands-on learning in IoT security risk management. By showcasing the tangible benefits of experiential learning through hackathons, our research equips educators, institutions, and practitioners with a novel strategy to bridge the longstanding gap between theoretical knowledge and its practical application in the dynamic field of cybersecurity. Our research meets a critical need in IoT security, where the urgency for skill development in securing IoT systems has been recognised [264, 265]. By providing participants with a platform to immerse themselves in real-world scenarios during hackathons, we highlight the pivotal role of experiential learning in nurturing essential competencies for identifying, analysing, and mitigating security risks associated with IoT environments. This valuable insight elevates discussions on curriculum design and pedagogical approaches within cybersecurity education. It beckons educators to embrace more immersive learning experiences that mirror the complexities of the actual challenges faced in practice. While the educational benefits of our integrated approach are evident, the scalability of the hackathon teaching model might encounter some challenges. Participants' diverse backgrounds and proficiency levels in system analysis and security risk management could pose challenges, particularly when introducing the IoTA-SRM framework beyond the academic context. Currently, the framework is most suitable for individuals or stakeholders with a solid foundation in security fundamentals and IoT systems.

5.5. Related Work

We proposed the hackathon teaching model introduced in our contributing publication [67], guiding educators in organising hackathons for IoT security risk management education. Comparing our findings to related works, as summarised

in Table 49, reveals the distinctiveness of the hackathon teaching model – integrating IoTA-SRM framework and intervention-based hackathon approach in teaching IoT security risk management.

Table 49. Related work

Criteria	[69]	[266]	[267]	[268]	**Our approach**
Security learning focus (i.e., security risk management)	+−	+	+	+	++
Hackathon for learning	+	+	+	+	++
Thematic input interventions	+	+−	+	+	++
Feedback interventions	+	+	−	−	++
Multiple hackathon iterations	+	−	+−	−	++

[++] Mostly fulfilled, [+] Partially fulfilled, [+−] Fulfilled with limitations, [−] Not fulfilled

For example, the competition-based hackathon organised by Cheung *et al.* [266] provided a practical cybersecurity scenario for participants to apply their knowledge while working together in a high-pressure environment. However, their approach relies heavily on self-study and peer instruction efforts, which may disadvantage participants who lack the motivation to learn independently. In contrast, our approach demonstrated that introducing thematic input stimulated learning and encouraged self-study through the availability of offline lecture recordings (as lecture resources). In Karagiannis and Magkos [267], capture-the-flag (CTF) challenges were used to help undergraduate participants acquire cybersecurity skills and knowledge. The approach incorporated gamification and self-directed and collaborative learning elements, encouraging teamwork and knowledge-sharing. Our approach used the hackathon teaching model to teach security risk management without relying on pre-existing cybersecurity skills to facilitate self-directed learning. Our selected IoT case and hackathon tasks also provided practical-oriented learning and allowed participants to adopt an adversarial thinking approach to security risk analysis from a hacker's perspective. Finally, O'Connor *et al.* [268] explored the benefits of gamification in a hands-on mobile and wireless cybersecurity course. The authors provided lectures and lab sessions, followed by a hackathon where participants could demonstrate their knowledge of hacking wireless protocols. Similarly, we found benefits in introducing adversarial thinking through hackathons to stimulate a rich threat analysis, facilitating this through our interventions and keeping participants engaged throughout the learning process. While existing cybersecurity hackathon approaches focus on CTF and competition-based hackathons [181], our study provides a unique perspective on the application and suitability of hackathons for teaching security risk management.

We address the learning focus on security risk management, employ hackathons as learning tools, integrate thematic input and feedback interventions, utilise multiple hackathon iterations, and align our approach closely with course curric-

ula. Although existing works have explored hackathons for cybersecurity education [179, 124], our research stands out for its targeted emphasis on security risk management education and intervention-based enhancements. Our approach contributes to the evolution of hackathons and hackathon interventions for educational purposes, emphasising the potential for effective learning in practical domains such as security risk management.

5.6. Summary

This chapter contributes to answering our main research question:

"*How can an integrated framework and teaching approach for IoT security risk management be realised?*"

We proposed our hackathon teaching model, demonstrating its benefits in using hackathons to teach how to apply the IoTA-SRM framework and in encouraging security learning. Our findings showed the benefits of the hackathon teaching model to provide hands-on learning opportunities for IoT SRM concepts and a structured approach to applying IoT SRM. This chapter also evaluated the application of the IoTA-SRM framework to an IoT use case within the hackathon context. Validation results show the usefulness of the IoTA-SRM framework in guiding hackathon participants to practice IoT SRM concepts through the framework's outlined tasks, applicability in different IoT use cases, and its ability to produce consistent and repeatable security outcomes/artefacts. These further validate the IoTA-SRM framework, providing insights into the practical use of the framework in an educational setting and highlighting its potential for broader adoption beyond the academic setting.

Our findings also contribute to the body of knowledge in IoT security risk management education by demonstrating the benefits of the practical application of IoT SRM concepts. This is seen through integrating the IoTA-SRM framework with the hackathon teaching model. Our findings contribute to practical cybersecurity education strategies, especially in security risk management, thus bridging the gap between theory and practice.

6. CONCLUSION

This chapter outlines the contributions and answers to research questions, providing potential for future work.

6.1. Answers to Research Questions

The thesis makes three major contributions to addressing the challenges of IoT security risk management and answering our main research question:

How can an integrated framework and teaching approach for IoT security risk management be realised?

First, we propose an IoT Architecture-based Security Risk Management (IoTA-SRM) framework that integrates the IoT architecture perspective into IoT systems' security risk management process. This framework follows a layered approach to security risk identification, analysis and management in IoT systems. Second, we introduce an intervention-based hackathon approach, providing a basis for practical implementation of the framework and promoting security learning. This approach proposes, adapts and evaluates hackathon interventions that organizers can implement at hackathons to maximise the learning potential of the event for participants. Third, we present a hackathon teaching model formalising our integrated approach, guiding participants on how to apply the IoTA-SRM framework within a hackathon methodology.

We summarise answers to our research questions along with a summary of our findings based on each of these questions below:

RQ_1. How to manage security risks in IoT system architectures?

To address the research question of how to manage security risks in IoT system architectures, we developed and evaluated the IoT architecture-based security risk management (IoTA-SRM) framework. This framework decomposes the IoT system into its layered architecture and performs asset identification and subsequent analysis and management of security risks at each layer.

The IoTA-SRM framework consists of four core activities: *Model System*, *Discover Risks*, *Handle Risks*, and *Analyse Trade-offs*, each featuring specific tasks and expected outcome artefacts. The framework provides avenues for comprehensive asset identification using the layered architecture as an output of the *Model System* activity and as an input to security risk management (*Discover Risks* activity). Subsequent tasks are multi-layer vulnerability, threat, and risk analyses in the *Discover Risks* activity. This provides the input to a multi-layer risk treatment analysis in the *Handle Risks* activity, where the trade-offs are also evaluated in the *Analyse Trade-offs* activity. We tested the framework through two case studies centred on autonomous vehicles (AVs). The first case enabled us to identify various threats and resulting risk impacts across multiple IoT layers of the AV. In the

second case, a pilot feature using an MQTT-based traffic light perception system between the AV and a traffic light system was similarly analysed. After identifying potential risks in both cases, we proposed control measures and provided the results of our analysis to the AV stakeholders. This allowed for stakeholder validation of the results of the framework application and the framework itself.

The IoTA-SRM framework promotes a layered risk assessment in IoT systems, facilitating well-informed risk mitigation strategies. It was applicable in medium and small-scale case study settings and offered an iterative approach for risk-based decision-making. Whilst the initial feedback on the framework was largely positive, there are still opportunities to improve its practical application.

RQ_2: How to use a hackathon-based methodology to teach security risk management in IoT system architectures?

To answer this question, we explored the potential of hackathons as an educational tool. While hackathons naturally create a favourable learning environment, targeted interventions are necessary to maximise their educational value in IoT security risk management. Thus, we applied the action research methodology across three hackathons. Through each hackathon, we followed the planning, action, observation, and reflection phases of the action research methodology, learning from each hackathon to formalise our intervention-based hackathon approach.

In the first cycle, we used a conventional hackathon format augmented by interventions like idea generation, thematic input, targeted feedback, and competition-style design. These interventions yielded promising educational outcomes, setting the stage for their refinement in future cycles. The second cycle extended the hackathon period and included multiple events, focusing on thematic input and targeted feedback. This approach improved the depth of learning, and we found that extending hackathon durations beyond 48 hours enhanced the learning journey significantly. We integrated the IoTA-SRM framework into the hackathon curriculum in the third cycle. Across three hackathon events, instructors facilitated knowledge transfer aligned with the IoTA-SRM activities and tasks, enhancing the educational impact. After each cycle, we evaluated and refined the interventions based on participant feedback and observations, ensuring they aligned more closely with the participants' learning needs and those of the framework.

Our findings consistently showed the learning benefits of introducing thematic input intervention during hackathons for knowledge transfer and the targeted feedback intervention, promoting collaboration and more experiential forms of knowledge sharing and transfer. Other interventions, such as idea generation and competition -style design interventions, still provided learning benefits but were found to be dependent on the specific educational hackathon context or format. Our intervention-based hackathon approach demonstrated benefits to bridging the gap between theoretical and practical aspects of IoT security risk management, alongside participant benefits in cultivating essential 21st-century skills like teamwork and communication.

An integrated framework and teaching approach for IoT security risk management

To address our main research question, we formalised our hackathon teaching model incorporating the IoTA-SRM framework with the intervention-based hackathon approach. The teaching model delivers the IoTA-SRM framework's core activities as distinct modules across multiple hackathons. Each hackathon can be carefully designed with thematic input and targeted feedback interventions. The thematic input through lectures provides domain-specific knowledge, and targeted feedback promotes continuous improvement. Although not explicitly highlighted in the hackathon teaching model, other interventions evaluated in the intervention-based hackathon approach can also be included depending on the hackathon context.

Empirical analysis of the integration of the IoTA-SRM framework and the intervention-based hackathon approach showed participants becoming proficient in the activities of the IoTA-SRM framework to provide comprehensive risk assessments as an outcome. Hackathon participants in their teams could consistently apply IoTA-SRM framework tasks at each hackathon and achieve its corresponding outcome, thus fostering a standardised approach to IoT security risk management. Learning from our experiences applying the hackathon teaching model, we also presented a guide for hackathon organisers and educators who wish to implement educational hackathons to teach IoT SRM using the IoTA-SRM framework. While this hackathon teaching model is formalised and evaluated within an academic setting, its modular nature shows potential for broader applicability, including in professional training and workshops.

The hackathon teaching model for IoT security risk management introduced a hands-on, experiential learning strategy, bridging the gap between theory and practice and addressing the need for skilled cybersecurity professionals. Our findings also open avenues for further discussion on curriculum design and pedagogical approaches in IoT security risk management education.

6.2. Future Work

This section outlines avenues for future research based on its contributions to the IoTA-SRM framework and the hackathon teaching model for IoT security risk management.

6.2.1. IoTA-SRM Framework

The IoTA-SRM framework effectively manages security risks in IoT systems by considering IoT architecture. Despite its benefits, future research still has scope to enhance its capabilities.

- Framework scalability: The IoTA-SRM framework has proven effective in a pilot case study involving autonomous vehicles. However, its scalability

for larger, more complex systems remains untested. Future research could focus on ensuring the framework's adaptability to bigger and more intricate IoT systems.

- Application to architecture variants: The framework was designed around a three-layer IoT architecture. While its applicability to other layered architectures is feasible, it has not been explored. Future work could look at adapting the framework for various architectural models and assess its utility in those contexts.
- IoTA-SRM tool support: Future development could focus on creating a comprehensive tool to support the IoTA-SRM framework. This tool should offer a range of features, including code-based IoT architecture input, asset-based security concept decomposition, threat modelling and risk analysis, control suggestions, and trade-off analysis. Additionally, it should offer reporting and visualization capabilities, compatibility with various IoT architectures, and scalability for systems of different sizes and complexities. It should also be user-friendly, customizable, and secure with features like access controls and data encryption. Developing such a tool would streamline identifying, analysing, and mitigating security risks in IoT systems.

6.2.2. Hackathon Teaching Model for IoT Security Risk Management

Teaching strategies and pedagogical approaches for continuous learning are crucial to ensure the benefits of the IoTA-SRM framework for IoT security risk management. While we developed the hackathon teaching model to deliver the IoTA-SRM framework, there is still room for future work in the following research directions.

- Customised hackathon teaching models: Tailoring the model to cater to different skill levels and professional backgrounds, such as security analysts and developers, would make it more versatile. Different hackathon models could be developed to introduce novices to the basics of the IoTA-SRM framework while offering more advanced challenges to experienced participants.
- Scalability: The current hackathon model has been effective for a select group, but there is a need to examine its scalability for broader adoption. Future studies could examine its applicability specifically for professionals, their challenges, learning outcomes, and relevance in professional settings.
- Sustainability of learning: Another important avenue for future research is examining the long-term sustainability of learning outcomes. Monitoring participants over extended periods could provide insights into how well they retain and apply what they've learned once the structured hackathon environment is no longer present. This would offer valuable feedback for improving future events and educational programmes.

BIBLIOGRAPHY

[1] Hasan Omar Al-Sakran. "Intelligent traffic information system based on integration of Internet of Things and Agent technology". In: *International Journal of Advanced Computer Science and Applications (IJACSA)* 6.2 (2015), pp. 37–43.

[2] Yash Shah and Shamik Sengupta. "A survey on Classification of Cyber-attacks on IoT and IIoT devices". In: *2020 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*. IEEE. 2020, pp. 0406–0413.

[3] Alem Čolaković and Mesud Hadžialić. "Internet of Things (IoT): A review of enabling technologies, challenges, and open research issues". In: *Computer networks* 144 (2018), pp. 17–39.

[4] Knud Lasse Lueth. *State of IoT – Spring 2023*. [Online]. Available at: https://iot-analytics.com/product/state-of-iot-spring-2023/. Accessed: 2023-05-27. 2023.

[5] Check Point Research. *The Tipping Point: Exploring the Surge in IoT Cyberattacks Globally*. [Online]. Available at: https://blog.checkpoint.com/security/the-tipping-point-exploring-the-surge-in-iot-cyberattacks-plaguing-the-education-sector/. Accessed: 2023-07-27. 2023.

[6] Nallapaneni Manoj Kumar and Pradeep Kumar Mallick. "The Internet of Things: Insights into the building blocks, component interactions, and architecture layers". In: *Procedia computer science* 132 (2018), pp. 109–117.

[7] William Crumpler and James A Lewis. "The cybersecurity workforce gap". In: *Center for Strategic and International Studies (CSIS) Washington, DC, USA* (2019).

[8] Nick Taylor and Loraine Clarke. "Everybody's Hacking: Participation and the Mainstreaming of Hackathons". In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM. 2018, p. 172.

[9] Ei Pa Pa Pe-Than et al. "Designing Corporate Hackathons With a Purpose: The Future of Software Development". In: *IEEE Software* 36.1 (2019), pp. 15–22.

[10] Allan Fowler. "Informal stem learning in game jams, hackathons and game creation events". In: *Proceedings of the International Conference on Game Jams, Hackathons, and Game Creation Events*. 2016, pp. 38–41.

[11] Michael J Covington and Rush Carskadden. "Threat implications of the internet of things". In: *2013 5th International Conference on Cyber Conflict (CYCON 2013)*. IEEE. 2013, pp. 1–12.

[12] Fadele Ayotunde Alaba et al. "Internet of Things security: A survey". In: *Journal of Network and Computer Applications* 88 (2017), pp. 10–28.

[13] Hui Li and Xin Zhou. “Study on security architecture for Internet of Things”. In: *International Conference on Applied Informatics and Communication*. Springer. 2011, pp. 404–411.

[14] Raman Shapaval and Raimundas Matulevičius. “Towards the Reference model for security risk management in internet of things”. In: *Databases and Information Systems: 13th International Baltic Conference, DB&IS 2018, Trakai, Lithuania, July 1-4, 2018, Proceedings 13*. Springer. 2018, pp. 58–72.

[15] Marco Lombardi, Francesco Pascale, and Domenico Santaniello. “Internet of things: A general overview between architectures, protocols and applications”. In: *Information* 12.2 (2021), p. 87.

[16] Ivan Cvitić, Miroslav Vujić, et al. “CLASSIFICATION OF SECURITY RISKS IN THE IOT ENVIRONMENT.” In: *Annals of DAAAM & Proceedings* 26.1 (2015).

[17] Xin Huang et al. “SecIoT: a security framework for the Internet of Things”. In: *Security and communication networks* 9.16 (2016), pp. 3083–3094.

[18] Vinita Malik and Sukhdip Singh. “Security risk management in IoT environment”. In: *Journal of Discrete Mathematical Sciences and Cryptography* 22.4 (2019), pp. 697–709.

[19] Abasi-Amefon O Affia, Raimundas Matulevičius, and Alexander Nolte. “Security risk management in cooperative intelligent transportation systems: a systematic literature review”. In: *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*. Springer. 2019, pp. 282–300.

[20] Rim Moalla et al. “Risk Analysis Study of its Communication Architecture”. In: *2012 Third International Conference on The Network of the Future (NOF)*. IEEE. 2012, pp. 1–5.

[21] Gary C Kessler. “Information security: New threats or familiar problems?” In: *Computer* 45.2 (2012), pp. 59–65.

[22] Alan Hevner et al. “Design science research in information systems”. In: *Design research in information systems: theory and practice* (2010), pp. 9–22.

[23] Claes Wohlin and Per Runeson. “Guiding the selection of research methodology in industry–academia collaboration in software engineering”. In: *Information and software technology* 140 (2021), p. 106678.

[24] Göran Goldkuhl and Jonas Sjöström. “Design science in the field: practice design research”. In: *Designing for a Digital and Globalized World: 13th International Conference, DESRIST 2018, Chennai, India, June 3–6, 2018, Proceedings 13*. Springer. 2018, pp. 67–81.

[25] Emelie Engström et al. “How software engineering research aligns with design science: a review”. In: *Empirical Software Engineering* 25 (2020), pp. 2630–2660.

[26] Oliver Gaß et al. “Anatomy of knowledge bases used in design science research: a literature review”. In: *Design Science Research in Information Systems. Advances in Theory and Practice: 7th International Conference, DESRIST 2012, Las Vegas, NV, USA, May 14-15, 2012. Proceedings 7*. Springer. 2012, pp. 328–344.

[27] Weng Marc Lim, Satish Kumar, and Faizan Ali. “Advancing knowledge through literature reviews:‘what’,‘why’, and ‘how to contribute’”. In: *The Service Industries Journal* 42.7-8 (2022), pp. 481–513.

[28] Hannah Snyder. “Literature review as a research methodology: An overview and guidelines”. In: *Journal of business research* 104 (2019), pp. 333–339.

[29] David N Boote and Penny Beile. “Scholars before researchers: On the centrality of the dissertation literature review in research preparation”. In: *Educational researcher* 34.6 (2005), pp. 3–15.

[30] Zaidah Zainal. “Case study as a research method”. In: *Jurnal kemanusiaan* 5.1 (2007).

[31] Julie Kim, Morgan Price, and Francis Lau. “The case study research method: Overview and proposed guidelines for reporting and evaluation illustrated with health informatics case studies”. In: *International Journal of Health Information Management Research* 2.1 (2014), pp. 13–30.

[32] Sudhakar Teegavarapu, Joshua D Summers, and Gregory M Mocko. “Case study method for design research: A justification”. In: *International design engineering technical conferences and computers and information in engineering conference*. Vol. 43284. 2008, pp. 495–503.

[33] Jan Vom Brocke, Alan Hevner, and Alexander Maedche. “Introduction to design science research”. In: *Design science research. Cases* (2020), pp. 1–13.

[34] Eric WK Tsang. “Case studies and generalization in information systems research: A critical realist perspective”. In: *The Journal of Strategic Information Systems* 23.2 (2014), pp. 174–186.

[35] Mary S Morgan. “Exemplification and the use-values of cases and case studies”. In: *Studies in History and Philosophy of Science Part A* 78 (2019), pp. 5–13.

[36] J Spencer Clark et al. *Action research*. 2020.

[37] Ivano Laudonia et al. “Action research in science education–an analytical review of the literature”. In: *Educational action research* 26.3 (2018), pp. 480–495.

[38] Roel Wieringa and Ayşe Moralı. “Technical action research as a validation method in information systems design science”. In: *International Confer-*

ence on Design Science Research in Information Systems. Springer. 2012, pp. 220–238.

[39] Kurt Lewin. "Action research and minority problems". In: *Journal of social issues* 2.4 (1946), pp. 34–46.

[40] Aline Dresch, Daniel Pacheco Lacerda, and Paulo Augusto Cauchick Miguel. "A distinctive analysis of case study, action research and design science research". In: *Revista brasileira de gestão de negócios* 17 (2015), pp. 1116–1133.

[41] Sheriz Khan and Patricia Tzortzopoulos. "Using design science research and action research to bridge the gap between theory and practice in lean construction research". In: *Proceedings of the 26th Annual Conference of the International Group for Lean Construction. Chennai*. 2018, pp. 209–219.

[42] Dalila Cisco Collatto et al. "Is action design research indeed necessary? Analysis and synergies between action research and design science research". In: *Systemic Practice and Action Research* 31 (2018), pp. 239–267.

[43] Anne Burns. "Action research". In: *Qualitative research in applied linguistics: A practical introduction* (2009), pp. 112–134.

[44] Dimiter M Dimitrov and Phillip D Rumrill Jr. "Pretest-posttest designs and measurement of change". In: *Work* 20.2 (2003), pp. 159–165.

[45] John Rogers and Andrea Revesz. "Experimental and quasi-experimental designs". In: *The Routledge handbook of research methods in applied linguistics*. Routledge, 2019, pp. 133–143.

[46] Matt Germonprez and Lars Mathiassen. "The role of conventional research methods in information systems action research". In: *Information Systems Research: Relevant Theory and Informed Practice* (2004), pp. 335–352.

[47] Robert M Davison, Maris G Martinsons, and Julien Malaurent. "Research perspectives: Improving action research by integrating methods". In: *Journal of the Association for Information Systems* 22.3 (2021), p. 1.

[48] Mohammad Zohrabi. "Mixed method research: Instruments, validity, reliability and reporting findings". In: *Theory and practice in language studies* 3.2 (2013), p. 254.

[49] Janette Biares Torrato, Socorro E Aguja, and Maricar S Prudente. "Analysis of the Teachers' Perceptions and Understanding on the Conduct of Action Research". In: *Proceedings of the 2021 12th International Conference on E-Education, E-Business, E-Management, and E-Learning*. 2021, pp. 264–270.

[50] Tony Loughland and Tony Loughland. "Classroom Observation as Method for Research and Improvement". In: *Teacher Adaptive Practices:*

Extending Teacher Adaptability into Classroom Practice (2019), pp. 23–42.

[51] Henryk Dźwigoł and Piotr Barosz. "Observation as a research method in social science". In: *Zeszyty Naukowe. Organizacja i Zarządzanie/Politechnika Śląska* 148 (2020), pp. 141–149.

[52] Matt O'Leary. *Classroom observation: A guide to the effective observation of teaching and learning*. Routledge, 2020.

[53] Hamed Taherdoost. "Validity and reliability of the research instrument; how to test the validation of a questionnaire/survey in a research". In: *How to test the validation of a questionnaire/survey in a research (August 10, 2016)* (2016).

[54] Svetlana Gudkova. "Interviewing in qualitative research". In: *Qualitative Methodologies in Organization Studies: Volume II: Methods and Possibilities* (2018), pp. 75–96.

[55] AJ Kleinheksel et al. "Demystifying content analysis". In: *American journal of pharmaceutical education* 84.1 (2020).

[56] David R Krathwohl and Lorin W Anderson. *A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives*. Longman, 2009.

[57] Liljana Koleva Gudeva et al. "Designing descriptors of learning outcomes for Higher Education qualification". In: *Procedia-Social and Behavioral Sciences* 46 (2012), pp. 1306–1311.

[58] Dong Yub Lee and Eunbae B Yang. "A critical evaluation of the concept and writing of learning outcomes". In: *Korean Medical Education Review* 18.3 (2016), pp. 125–131.

[59] Mark Battersby. "So, What's a Learning Outcome Anyway?." In: (1999).

[60] Benjamin S Bloom et al. "Taxonomy of educational objectives. Vol. 1: Cognitive domain". In: *New York: McKay* (1956), pp. 20–24.

[61] Evelina Johansson. "The assessment of higher-order thinking skills in online EFL courses: A quantitative content analysis". In: *NJES Nordic Journal of English Studies* 19.1 (2020), pp. 224–256.

[62] Jason Stayanchi. "Higher order thinking through Bloom's taxonomy". In: *Kwansei Gakuin University Humanities Review* 22 (2017), pp. 117–124.

[63] Dinçay Köksal and Ömer Gökhan Ulum. "Language assessment through Bloom's Taxonomy". In: *Journal of language and linguistic studies* 14.2 (2018), pp. 76–88.

[64] Rupert Ward et al. "Towards a 21st century personalised learning skills taxonomy". In: *2021 IEEE Global Engineering Education Conference (EDUCON)*. IEEE. 2021, pp. 344–354.

[65] Abasi-Amefon O Affia, Raimundas Matulevičius, and Rando Tõnisson. "Security Risk Estimation and Management in Autonomous Driving Ve-

hicles". In: *International Conference on Advanced Information Systems Engineering*. Springer. 2021, pp. 11–19.

[66] Abasi-Amefon O Affia and Raimundas Matulevičius. "Securing an MQTT-based Traffic Light Perception System for Autonomous Driving". In: *2021 IEEE International Conference on Cyber Security and Resilience (CSR)*. IEEE. 2021, pp. 255–260.

[67] Abasi-amefon O Affia, Alexander Nolte, and Raimundas Matulevičius. "IoT Security Risk Management: A Framework and Teaching Approach". In: *Informatics in Education* (2023). ISSN: 1648-5831. DOI: 10.15388/infedu.2023.30.

[68] Abasi-Amefon O Affia, Alexander Nolte, and Raimundas Matulevičius. "Developing and Evaluating a Hackathon Approach to Foster Cyber Security Learning". In: *International Conference on Collaboration Technologies and Social Computing*. Springer. 2020, pp. 3–19.

[69] Abasi-amefon Obot Affia, Alexander Nolte, and Raimundas Matulevičius. "Integrating Hackathons into an Online Cybersecurity Course". In: *2022 IEEE/ACM 44th International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*. 2022, pp. 134–145. DOI: 10.1109/ICSE-SEET55299.2022.9794183.

[70] Rabeh Morrar, Husam Arman, and Saeed Mousa. "The fourth industrial revolution (Industry 4.0): A social innovation perspective". In: *Technology innovation management review* 7.11 (2017), pp. 12–20.

[71] Max Felser, Markus Rentschler, and Oliver Kleineberg. "Coexistence standardization of operation technology and information technology". In: *Proceedings of the IEEE* 107.6 (2019), pp. 962–976.

[72] Eric Simmon and Eric Simmon. *Internet of Things (IoT) component capability model for research testbed*. US Department of Commerce, National Institute of Standards and Technology, 2020.

[73] Shariq Haseeb et al. "Connectivity, interoperability and manageability challenges in internet of things". In: *AIP conference proceedings*. Vol. 1883. 1. AIP Publishing. 2017.

[74] Claude Baudoin et al. *The Industrial Internet of Things Vocabulary*. [Online]. Accessed: 2020-08-25. 2020.

[75] IEC ISO. "IEEE,"Systems and software engineering-system life cycle process,"" in: *International Organization for Standardization, Geneva, Switzerland, Tech. Rep* (2015).

[76] Doruk Sahinel et al. "Integration of human actors in IoT and CPS landscape". In: *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*. IEEE. 2019, pp. 485–490.

[77] Sarah A Al-Qaseemi et al. "IoT architecture challenges and issues: Lack of standardization". In: *2016 Future technologies conference (FTC)*. IEEE. 2016, pp. 731–738.

[78] Martin Bauer and Joachim W Walewski. “The IoT architectural reference model as enabler”. In: *Enabling Things to Talk: Designing IoT solutions with the IoT Architectural Reference Model* (2013), pp. 17–25.

[79] Eryk Schiller et al. “Landscape of IoT security”. In: *Computer Science Review* 44 (2022), p. 100467.

[80] S Narasimha Swamy and Solomon Raju Kota. “An empirical study on system level aspects of Internet of Things (IoT)”. In: *IEEE Access* 8 (2020), pp. 188082–188134.

[81] GR Sagar and N Jayapandian. “Internet of things: service-oriented architecture opportunities and challenges”. In: *Inventive Communication and Computational Technologies: Proceedings of ICICCT 2019* (2020), pp. 71–78.

[82] Ala Al-Fuqaha et al. “Internet of things: A survey on enabling technologies, protocols, and applications”. In: *IEEE communications surveys & tutorials* 17.4 (2015), pp. 2347–2376.

[83] Aparna Raj and Sujala D Shetty. “IoT eco-system, layered architectures, security and advancing technologies: A comprehensive survey”. In: *Wireless Personal Communications* 122.2 (2022), pp. 1481–1517.

[84] Ryan Smith et al. “Battery draining attacks against edge computing nodes in IoT networks”. In: *Cyber-Physical Systems* 6.2 (2020), pp. 96–116.

[85] Miao Wu et al. “Research on the architecture of Internet of Things”. In: *2010 3rd international conference on advanced computer theory and engineering (ICACTE)*. Vol. 5. IEEE. 2010, pp. V5–484.

[86] P. M. D’Orey and M. Ferreira. “ITS for Sustainable Mobility: A Survey on Applications and Impact Assessment Tools”. In: *IEEE Transactions on Intelligent Transportation Systems* 15.2 (Apr. 2014), pp. 477–493. DOI: 10.1109/TITS.2013.2287257.

[87] Thiyagarajan Manihatty Bojan, Umamaheswaran Raman Kumar, and Viswanathan Manihatty Bojan. “An internet of things based intelligent transportation system”. In: *2014 IEEE International Conference on Vehicular Electronics and Safety*. IEEE. 2014, pp. 174–179.

[88] Asier Perallos et al. *Intelligent Transport Systems: Technologies and Applications*. John Wiley & Sons, 2015.

[89] Ryan Fries, Mashrur Chowdhury, and Jeffrey Brummond. *Transportation Infrastructure Security Utilizing Intelligent Transportation Systems*. John Wiley & Sons, 2009.

[90] Donald Firesmith. *Engineering Safety and Security-Related Requirements for Software-Intensive Systems*. Tech. rep. Carnegie-Mellon University Pittsburg PA Software Engineering Institute, 2007.

[91] Hee-Kyung Kong, Myoung Ki Hong, and Tae-Sung Kim. “Security Risk Assessment Framework for Smart Car Using the Attack Tree Analy-

sis". In: *Journal of Ambient Intelligence and Humanized Computing* 9.3 (2018), pp. 531–551.

[92] Alastair R Ruddle et al. "Cyber Security Risk Analysis for Intelligent Transport Systems and In-Vehicle Networks". In: *Intelligent Transport Systems: Technologies and Applications* 83 (2015).

[93] Asif Faisal et al. "Understanding autonomous vehicles". In: *Journal of transport and land use* 12.1 (2019), pp. 45–72.

[94] SAE International. "Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles". In: (2021). Available at https://www.sae.org/standards/content/j3016_202104/. DOI: 10.4271/j3016_202104.

[95] Jun Wang et al. "Safety of autonomous vehicles". In: *Journal of advanced transportation* 2020 (2020), pp. 1–13.

[96] Muhammad Hataba et al. "Security and Privacy Issues in Autonomous Vehicles: A Layer-Based Survey". In: *IEEE Open Journal of the Communications Society* 3 (2022), pp. 811–829.

[97] V. L. Thing and J. Wu. "Autonomous Vehicle Security: A Taxonomy of Attacks and Defences". In: *IEEE Int. Conf. Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. 2016, pp. 164–170.

[98] Xiaoli Sun et al. "Technology development of electric vehicles: A review". In: *Energies* 13.1 (2019), p. 90.

[99] Paul J DeMaio. "Smart bikes: Public transportation for the 21st century". In: *Transportation Quarterly* 57.1 (2003), pp. 9–11.

[100] Alberica Domitilla Bozzi and Anne Aguilera. "Shared E-scooters: A review of uses, health and environmental impacts, and policy implications of a new micro-mobility service". In: *Sustainability* 13.16 (2021), p. 8676.

[101] Programme 2014 - 2020 INTERREG VB Central Europe. *Shared mobility and Regional transport integrated PLAnning for a better connected Central Europe (SHAREPLACE)*. Nov. 2022. URL: https://keep.eu/projects/18230/Shared-mobility-and-Regiona-EN/.

[102] Community Research and Development Information Service (CORDIS). *Modelling Emerging Transport Solutions for Urban Mobility (MOMENTUM). H2020-EU.3.4. - SOCIETAL CHALLENGES - Smart, Green And Integrated Transport*. Nov. 2022. URL: https://cordis.europa.eu/project/id/815069.

[103] Abasi-amefon Obot Affia and Raimundas Matulevičius. "Security Risk Management in Shared Mobility Integration". In: *Proceedings of the 17th International Conference on Availability, Reliability and Security*. 2022, pp. 1–10.

[104] Marilyn Wolf and Dimitrios Serpanos. “Safety and security in cyber-physical systems and internet-of-things systems”. In: *Proceedings of the IEEE* 106.1 (2017), pp. 9–20.

[105] Traian Mihai Popescu, Alina Madalina Popescu, and Gabriela Prostean. “IoT Security Risk Management Strategy Reference Model (IoTSRM2)”. In: *Future Internet* 13.6 (2021), p. 148.

[106] In Lee. “Internet of Things (IoT) cybersecurity: Literature review and IoT cyber risk management”. In: *Future Internet* 12.9 (2020), p. 157.

[107] John Soldatos. *Security Risk Management for the Internet of Things: Technologies and Techniques for IoT Security, Privacy and Data Protection*. Now Publishers, 2020.

[108] Imad Yassine, Talal Halabi, and Martine Bellaiche. “Security Risk Assessment Methodologies in The Internet of Things: Survey and Taxonomy”. In: *2021 IEEE 21st International Conference on Software Quality, Reliability and Security Companion (QRS-C)*. IEEE. 2021, pp. 668–675.

[109] Kamalanathan Kandasamy et al. “IoT cyber risk: A holistic analysis of cyber risk assessment frameworks, risk vectors, and risk ranking process”. In: *EURASIP Journal on Information Security* 2020.1 (2020), pp. 1–18.

[110] Salim Chehida et al. “Asset-Driven Approach for Security Risk Assessment in IoT Systems”. In: *Risks and Security of Internet and Systems: 15th International Conference, CRiSIS 2020, Paris, France, November 4–6, 2020, Revised Selected Papers 15*. Springer. 2021, pp. 149–163.

[111] ANSSI. “EBIOS 2010–Expression of Needs and Identification of Security Objectives”. In: (2010).

[112] Richard A Caralli et al. *Introducing octave allegro: Improving the information security risk assessment process*. Tech. rep. Carnegie-Mellon Univ Pittsburgh PA Software Engineering Inst, 2007.

[113] Nicolas Mayer. “Model-based management of information system security risk”. PhD thesis. University of Namur, 2009.

[114] Jackson Wynn et al. *Threat assessment & remediation analysis (TARA): methodology description version 1.0*. Tech. rep. MITRE CORP BEDFORD MA, 2011.

[115] Alberto Rodrigues Da Silva. “Model-driven engineering: A survey supported by the unified conceptual model”. In: *Computer Languages, Systems & Structures* 43 (2015), pp. 139–155.

[116] Johannes Geismann and Eric Bodden. “A systematic literature review of model-driven security engineering for cyber–physical systems”. In: *Journal of Systems and Software* 169 (2020), p. 110697.

[117] Éric Dubois et al. “A systematic approach to define the domain of information system security risk management”. In: *Intentional Perspectives on Information Systems Engineering*. Springer, 2010, pp. 289–306.

[118] Raimundas Matulevičius. “Fundamentals of Secure System Modelling”. In: (2017).

[119] Mubashar Iqbal. “Reference Framework for Managing Security Risks Using Blockchain”. PhD thesis. University of Tartu, 2022.

[120] Michael Muckin and Scott C Fitch. “A threat-driven approach to cyber security”. In: *Lockheed Martin Corporation* (2014).

[121] Michael Howard and Steve Lipner. *The security development lifecycle*. Vol. 8. Microsoft Press Redmond, 2006.

[122] Abasi-amefon O Affia, Raimundas Matulevičius, and Alexander Nolte. “Security risk management in e-commerce systems: a threat-driven approach”. In: *Baltic Journal of Modern Computing* 8.2 (2020), pp. 213–240.

[123] Lukas Malina et al. “Post-Quantum Era Privacy Protection for Intelligent Infrastructures”. In: *IEEE Access* 9 (2021), pp. 36038–36077.

[124] Shancang Li, Theo Tryfonas, and Honglei Li. “The Internet of Things: a security point of view”. In: *Internet Research* (2016).

[125] Lotfi ben Othmane et al. “Incorporating Attacker Capabilities in Risk Estimation and Mitigation”. In: *Computers & Security* 51 (2015), pp. 41–61.

[126] Jan Pelzl, Marko Wolf, and Thomas Wollinger. “Automotive Embedded Systems Applications and Platform Embedded Security Requirements”. In: *Secure Smart Embedded Devices, Platforms and Applications*. Springer, 2014, pp. 287–309.

[127] Gonzalo De La Torre, Paul Rad, and Kim-Kwang Raymond Choo. “Driverless Vehicle Security: Challenges and Future Research Opportunities”. In: *Future Generation Computer Systems* (2018).

[128] Mohamed Nidhal Mejri, Ben-Othman Jalel, and Mohamed Hamdi. “Survey on VANET Security Challenges and Possible Cryptographic Solutions”. In: *Vehicular Communications* 1.2 (2014), pp. 53–66.

[129] Joseph Bugeja et al. “IoTSM: an end-to-end security model for IoT ecosystems”. In: *2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. IEEE. 2019, pp. 267–272.

[130] S Tbatou, A Ramrami, and Youness Tabii. “Security of Communications in Connected Cars Modeling and Safety Assessment”. In: *Proceedings of the 2nd international Conference on Big Data, Cloud and Applications*. ACM. 2017, p. 56.

[131] Shantanu Pal et al. “Security requirements for the internet of things: A systematic approach”. In: *Sensors* 20.20 (2020), p. 5897.

[132] Qazi Mamoon Ashraf and Mohamed Hadi Habaebi. “Autonomic schemes for threat mitigation in Internet of Things”. In: *Journal of Network and Computer Applications* 49 (2015), pp. 112–127.

[133] Mohamed A Aref, Sudharman K Jayaweera, and Stephen Machuzak. "Multi-agent reinforcement learning based cognitive anti-jamming". In: *2017 IEEE wireless communications and networking conference (WCNC)*. IEEE. 2017, pp. 1–6.

[134] Yunchuan Sun et al. "Attacks and Countermeasures in the Internet of Vehicles". In: *Annals of Telecommunications* 72.5-6 (2017), pp. 283–295.

[135] Jerry Den Hartog, Nicola Zannone, et al. "Security and Privacy for Innovative Automotive Applications: A Survey". In: *Computer Communications* 132 (2018), pp. 17–41.

[136] Ignacio Sanchez et al. "Privacy leakages in Smart Home wireless technologies". In: *2014 International Carnahan Conference on Security Technology (ICCST)*. IEEE. 2014, pp. 1–6.

[137] Abbas Acar et al. "Peek-a-Boo: I see your smart home activities, even encrypted!" In: *arXiv preprint arXiv:1808.02741* (2018).

[138] Sharmistha Roy, Prashant Pranav, and Vandana Bhattacharjee. "Securing the Internet of Things: Current and Future State of the Art". In: *Smart Healthcare Analytics in IoT Enabled Environment*. Springer, 2020, pp. 227–246.

[139] Richard Gilles Engoulou et al. "VANET Security Surveys". In: *Computer Communications* 44 (2014), pp. 1–13.

[140] Qian Chen, Azizeh Khaled Sowan, and Shouhuai Xu. "A Safety and Security Architecture for Reducing Accidents in Intelligent Transportation Systems". In: *Proceedings of the International Conference on Computer-Aided Design*. ACM. 2018, p. 95.

[141] Christine Laurendeau and Michel Barbeau. "Threats to Security in DSRC/WAVE". In: *International Conference on Ad-Hoc Networks and Wireless*. Springer. 2006, pp. 266–279.

[142] Hamssa Hasrouny et al. "VANet Security Challenges and Solutions: A Survey". In: *Vehicular Communications* 7 (2017), pp. 7–20.

[143] Elyes Hamida, Hassan Noura, and Wassim Znaidi. "Security of Cooperative Intelligent Transport Systems: Standards, Threats Analysis and Cryptographic Countermeasures". In: *Electronics* 4.3 (2015), pp. 380–423.

[144] Mohammed Saeed Al-Kahtani. "Survey on Security Attacks in Vehicular Ad hoc Networks (VANETs)". In: *2012 6th International Conference on Signal Processing and Communication Systems*. IEEE. 2012, pp. 1–9.

[145] Michael W Condry and Catherine Blackadar Nelson. "Using smart edge IoT devices for safer, rapid response with industry IoT control operations". In: *Proceedings of the IEEE* 104.5 (2016), pp. 938–946.

[146] Paolo Nesi et al. "Auditing and assessment of data traffic flows in an IoT architecture". In: *2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC)*. IEEE. 2018, pp. 388–391.

[147] Fei Chen et al. “TrustBuilder: A non-repudiation scheme for IoT cloud applications”. In: *Computers & Security* 116 (2022), p. 102664.

[148] Workneh Y Ayele. “Non-repudiation mechanisms for IoT applications: A systematic literature review”. In: (2021).

[149] KS Divya, HR Roopashree, and AC Yogeesh. “Framework of Multiparty Computation for Higher Non-Repudiation in Internet-of-Things (IoT)”. In: *International Journal of Computer Networks and Applications* 10.1 (2023), pp. 84–94.

[150] Qi Wang et al. “Fear and logging in the internet of things”. In: *Network and Distributed Systems Symposium*. 2018.

[151] Cleber Santana et al. “Increasing the availability of IoT applications with reactive microservices”. In: *Service Oriented Computing and Applications* 15 (2021), pp. 109–126.

[152] Petar Radanliev et al. “Cyber risk management for the internet of things”. In: (2019).

[153] Panayiotis Kotzanikolaou, Marianthi Theoharidou, and Dimitris Gritzalis. “Cascading effects of common-cause failures in critical infrastructures”. In: *International Conference on Critical Infrastructure Protection*. Springer. 2013, pp. 171–182.

[154] Jake Rowan Byrne, Katriona O’Sullivan, and Kevin Sullivan. “An IoT and wearable technology hackathon for promoting careers in computer science”. In: *IEEE Transactions on Education* 60.1 (2016), pp. 50–58.

[155] Tom Goodman and Andreea-Ina Radu. “Learn-apply-reinforce/share learning: hackathons and CTFs as general pedagogic tools in higher education, and their applicability to distance learning”. In: *arXiv preprint arXiv:2006.04226* (2020).

[156] Lyndsey Middleton, Hazel Hall, and Robert Raeside. “Applications and applicability of Social Cognitive Theory in information science research”. In: *Journal of Librarianship and Information Science* 51.4 (2019), pp. 927–937.

[157] Dale H Schunk and Maria K DiBenedetto. “Motivation and social cognitive theory”. In: *Contemporary educational psychology* 60 (2020), p. 101832.

[158] Hwee-Joo Kam and Pairin Katerattanakul. “Enhancing student learning in cybersecurity education using an out-of-class learning approach”. In: *Journal of Information Technology Education. Innovations in Practice* 18 (2019), p. 29.

[159] Seth T Hamman and Kenneth M Hopkinson. “Teaching adversarial thinking for cybersecurity”. In: *Journal of The Colloquium for Information Systems Security Education*. Vol. 4. 1. 2016, pp. 19–19.

[160] Ping Wang and Raed Sbeit. “A comprehensive mentoring model for cybersecurity education”. In: *17th International Conference on Information Technology–New Generations (ITNG 2020)*. Springer. 2020, pp. 17–23.

[161] Yan Bai, Chunming Gao, and Bryan Goda. “Lessons learned from teaching cybersecurity courses during Covid-19”. In: *Proceedings of the 21st Annual Conference on Information Technology Education*. 2020, pp. 308–313.

[162] K Boopathi, S Sreejith, and A Bithin. “Learning cyber security through gamification”. In: *Indian Journal of Science and Technology* 8.7 (2015), pp. 642–649.

[163] Richard S Weiss et al. “Teaching cybersecurity analysis skills in the cloud”. In: *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*. 2015, pp. 332–337.

[164] William Cerbin. “Improving student learning from lectures.” In: *Scholarship of Teaching and Learning in Psychology* 4.3 (2018), p. 151.

[165] Émilie Tremblay-Wragg et al. “The use of diversified teaching strategies by four university teachers: what contribution to their students’ learning motivation?” In: *Teaching in Higher Education* 26.1 (2021), pp. 97–114.

[166] Gertrude H Hildreth. “Evaluation of a Workshop in Education”. In: *Teachers College Record* 46.5 (1945), pp. 1–8.

[167] Ping Xia. “Reform of Network Security Technology Practice Teaching System Based on Virtual Simulation Training Platform”. In: *2020 International Wireless Communications and Mobile Computing (IWCMC)*. IEEE. 2020, pp. 199–203.

[168] Joseph S Krajcik and Phyllis C Blumenfeld. *Project-based learning*. na, 2006.

[169] Dimitra Kokotsaki, Victoria Menzies, and Andy Wiggins. “Project-based learning: A review of the literature”. In: *Improving schools* 19.3 (2016), pp. 267–277.

[170] Alan T Sherman et al. “Project-based learning inspires cybersecurity students: A scholarship-for-service research study”. In: *IEEE Security & Privacy* 17.3 (2019), pp. 82–88.

[171] Gordon W Romney et al. “A teaching prototype for educating IT security engineers in emerging environments”. In: *Information Technology Based Proceedings of the Fifth International Conference on Higher Education and Training, 2004. ITHET 2004.* IEEE. 2004, pp. 662–667.

[172] Abdul Syakur et al. “The effect of project-based learning (PjBL) continuing learning innovation on learning outcomes of english in higher education”. In: *Budapest International Research and Critics in Linguistics and Education (BirLE) Journal* 3.1 (2020), pp. 625–630.

[173] Alexander Nolte, Linda Bailey Hayden, and James D Herbsleb. “How to Support Newcomers in Scientific Hackathons - An Action Research Study

on Expert Mentoring". In: *Proceedings of the ACM on Human-Computer Interaction* 4.CSCW1 (2020), Article 25 (May 2020), 23 pages.

[174] Jari Porras et al. "Code camps and hackathons in education-literature review and lessons learned". In: *Proceedings of the 52nd Hawaii International Conference on System Sciences* (2019).

[175] Hanna Kienzler and Carolyn Fontanesi. "Learning through inquiry: A global health hackathon". In: *Teaching in Higher Education* 22.2 (2017), pp. 129–142.

[176] Karl Edvard Balto et al. "Hybrid IoT Cyber Range". In: *Sensors* 23.6 (2023), p. 3071.

[177] Jake Rowan Byrne, Katriona O'Sullivan, and Kevin Sullivan. "An IoT and Wearable Technology Hackathon for Promoting Careers in Computer Science". In: *IEEE Transactions on Education* 60.1 (2017), pp. 50–58.

[178] Alexandre Oliveira Junior et al. "Learning Cybersecurity in IoT-based Applications through a Capture the Flag Competition". In: *2022 IEEE 20th International Conference on Industrial Informatics (INDIN)*. IEEE. 2022, pp. 560–565.

[179] Valdemar Švábenskỳ et al. "Cybersecurity knowledge and skills taught in capture the flag challenges". In: *Computers & Security* 102 (2021), p. 102154.

[180] Tyler Balon and Ibrahim Baggili. "Cybercompetitions: A survey of competitions, tools, and systems to support cybersecurity education". In: *Education and Information Technologies* (2023), pp. 1–33.

[181] Chengcheng Li and Rucha Kulkarni. "Survey of cybersecurity education through gamification". In: *2016 ASEE Annual Conference & Exposition*. 2016.

[182] Valdemar Švábenskỳ et al. "Applications of educational data mining and learning analytics on data from cybersecurity training". In: *Education and Information Technologies* 27.9 (2022), pp. 12179–12212.

[183] Maria Angelica Medina Angarita and Alexander Nolte. "Does it matter why we hack?–Exploring the impact of goal alignment in hackathons". In: *Proceedings of 17th European Conference on Computer-Supported Cooperative Work*. European Society for Socially Embedded Technologies (EUSSET). 2019.

[184] Alexander Nolte et al. "How to organize a hackathon–A planning kit". In: *arXiv preprint arXiv:2008.08025* (2020).

[185] Anna Filippova, Erik Trainer, and James D Herbsleb. "From diversity by numbers to diversity as process: supporting inclusiveness in software development teams with brainstorming". In: *2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*. IEEE. 2017, pp. 152–163.

[186] James Tandon, Nazzy Pakpour, and Mario Gumina. “Using Hackathons as a Tool in STEM Education”. In: *Journal of Interdisciplinary Studies in Education* 10.2 (2021).

[187] Andrzej Szymkowiak et al. “Information technology and Gen Z: The role of teachers, the internet, and technology in the education of young people”. In: *Technology in Society* 65 (2021), p. 101565.

[188] Anastasija Nikiforova. “Gen Z open data hackathon–civic innovation with digital natives: to hack or not to hack”. In: *Proceedings of ongoing research, practitioners, workshops, posters, and projects of the international conference EGOV-CeDEM-ePart 202. Linkoping, Sweden*. 2022, pp. 251–253.

[189] Adam Shostack. *Threat Modeling: Designing for Security*. John Wiley & Sons, 2014. ISBN: 9781118809990.

[190] Wan Rahiman and Zafariq Zainal. “An overview of development GPS navigation for autonomous car”. In: *2013 IEEE 8th Conference on Industrial Electronics and Applications (ICIEA)*. IEEE. 2013, pp. 1112–1118.

[191] Nenad Medvidovic et al. “Modeling software architectures in the unified modeling language”. In: *ACM Transactions on Software Engineering and Methodology (TOSEM)* 11.1 (2002), pp. 2–57.

[192] Sanford Friedenthal, Alan Moore, and Rick Steiner. *A practical guide to SysML: the systems modeling language*. Morgan Kaufmann, 2014.

[193] Mark von Rosing et al. *Business Process Model and Notation-BPMN*. 2015.

[194] NVD. *National vulnerability database. National Institute of Standards and Technology*. [Online]. Accessed: 2019-05-30. a.

[195] CWE. *Common Weakness Enumeration. A community-developed dictionary of software weakness types*. [Online]. Accessed: 2020-03-20. a.

[196] OWASP. *OWASP Top 10:2021*. Available at https://owasp.org/Top10/. 2021.

[197] Livinus Obiora Nweke and Stephen Wolthusen. “A review of asset-centric threat modelling approaches”. In: (2020).

[198] Guttorm Sindre and Andreas L Opdahl. “Eliciting security requirements with misuse cases”. In: *Requirements engineering* 10 (2005), pp. 34–44.

[199] Cloud Security Alliance. *CSA IoT Security Controls Framework v2*. Available at: https://cloudsecurityalliance.org/artifacts/csa-iot-security-controls-framework-v2/. Accessed: 2022-07-27. 2021.

[200] Center for Internet Security. *CIS Critical Security Controls Version 8*. Available at: https://cloudsecurityalliance.org/artifacts/csa-iot-security-controls-framework-v2/. Accessed: 2023-09-27. 2021.

[201] Wayne Jansen. *Directions in security metrics research*. Diane Publishing, 2010.

[202] Jack Freund and Jack Jones. *Measuring and managing information risk: a FAIR approach*. Butterworth-Heinemann, 2014.

[203] RCMP CSE. "Harmonized Threat and Risk Assessment (TRA) Methodology". In: *TRA-1 Date: October* 23 (2007).

[204] Salvatore Antonio Biancardo et al. "An innovative framework for integrating cost-benefit analysis (cba) within building information modeling (bim)". In: *Socio-Economic Planning Sciences* 85 (2023), p. 101495.

[205] N Thomopoulos et al. "Evaluation methodology and measurement approach". In: (2013).

[206] Jakub Breier and Ladislav Hudec. "New approach in information system security evaluation". In: *2012 IEEE First AESS European Conference on Satellite Telecommunications (ESTEL)*. IEEE. 2012, pp. 1–6.

[207] Thomas L Saaty. "How to make a decision: the analytic hierarchy process". In: *European journal of operational research* 48.1 (1990), pp. 9–26.

[208] Seyed Mojtaba Hosseini Bamakan and Mohammad Dehghanimohammadabadi. "A weighted Monte Carlo simulation approach to risk assessment of information security management system". In: *International Journal of Enterprise Information Systems (IJEIS)* 11.4 (2015), pp. 63–78.

[209] Lingyu Wang et al. "Measuring the overall network security by combining cvss scores based on attack graphs and bayesian networks". In: *Network Security Metrics* (2017), pp. 1–23.

[210] Nicolas Mayer et al. "Towards a Measurement Framework for Security Risk Management." In: *MODSEC@ MoDELS*. 2008.

[211] University of Tartu. *Autonomous Driving Lab*. https://adl.cs.ut.ee/.

[212] C. Yan, W. Xu, and J. Liu. "Can You Trust Autonomous Vehicles: Contactless Attacks Against Sensors of Self-Driving Vehicle". In: *DEFCON*. 2016.

[213] C. Maple et al. "A Connected and Autonomous Vehicle Reference Architecture for Attack Surface Analysis". In: *Applied Sciences* 9.23 (2019).

[214] J. Petit et al. "Remote Attacks on Automated Vehicles Sensors: Experiments on Camera and Lidar". In: *Black Hat Europe*. 2015.

[215] D. Bailey. "Quantitative Cybersecurity Risk Management for Autonomous Vehicle Systems". MA thesis. Technisch Universitat Munchen, 2018.

[216] Syed Naeem Firdous et al. "Modelling and Evaluation of Malicious Attacks Against the IoT MQTT Protocol". In: *2017 IEEE Int. Conf.s on iThings, GreenCom, CPSCom and SmartData*. IEEE. 2017, pp. 748–755.

[217] Syaiful Andy, Budi Rahardjo, and Bagus Hanindhito. "Attack Scenarios and Security Analysis of MQTT Communication Protocol in IoT Sys-

tem". In: *2017 4th Int. Conf. on Electrical Engineering, Computer Science and Informatics (EECSI)*. IEEE. 2017, pp. 1–6.

[218] AP Haripriya and K Kulothungan. "Secure-MQTT: an Efficient Fuzzy Logic-Based Approach to Detect DoS Attack in MQTT Protocol for Internet of Things". In: *EURASIP Journal on Wireless Communications and Networking* 2019.1 (2019), p. 90.

[219] Ivan Vaccari, Maurizio Aiello, and Enrico Cambiaso. "SlowITe, a Novel Denial of Service Attack Affecting MQTT". In: *Sensors* 20.10 (2020), p. 2932.

[220] Ivan Vaccari, Maurizio Aiello, and Enrico Cambiaso. "SlowTT: A Slow Denial of Service Against IoT Networks". In: *Information* 11.9 (2020), p. 452.

[221] J Voas. "NIST Special Publication 800-183. Networks of 'Things'". In: *Gaithersburg, MD: National Institute of Standards and Technology (NIST)* (2016).

[222] IoTSF. *IoT Security Foundation*. Available at: https://iotsecurityfoundation.org/best-practice-guidelines/. Accessed: 2023-09-24.

[223] OWASP. *OWASP Top 10: Internet of Things*. Available at: https://owasp.org/www-pdf-archive/OWASP-IoT-Top-10-2018-final.pdf. Accessed: 2022-10-27.

[224] National Institute of Standards and Technology NIST. "Risk management guide for information technology systems". In: *NIST special publication* 800.30 (2002), pp. 800–30.

[225] MITRE. *CVE Common Vulnerabilities and Exposures database*. [Online]. Accessed: 2023-09-20.

[226] Faride Latifi and Houman Zarrabi. "A COBIT5 Framework for IoT risk management". In: *International Journal of Computer Applications* 170.8 (2017), pp. 40–43.

[227] Musa G Samaila et al. "IoT-HarPSecA: A framework for facilitating the design and development of secure IoT devices". In: *Proceedings of the 14th International Conference on Availability, Reliability and Security*. 2019, pp. 1–7.

[228] Leonard J Waks. "Value judgment and social action in technology studies". In: *International Journal of Technology and Design Education* 4 (1994), pp. 35–49.

[229] Annette Isabel Böhmer, Andreas Beckmann, and Udo Lindemann. "Open innovation ecosystem-makerspaces within an agile innovation process". In: *ISPIM Innovation Summit*. 2015.

[230] Behiye Akcay. "Problem-based learning in science education". In: *Journal of Turkish Science Education* 6.1 (2009), pp. 28–38.

[231] Lindsay Joseph Wexler. “How feedback from mentor teachers sustained student teachers through their first year of teaching”. In: *Action in Teacher Education* 42.2 (2020), pp. 167–185.

[232] John Hattie and Helen Timperley. “The power of feedback”. In: *Review of educational research* 77.1 (2007), pp. 81–112.

[233] David Boud and Elizabeth Molloy. “Rethinking models of feedback for learning: the challenge of design”. In: *Assessment & Evaluation in higher education* 38.6 (2013), pp. 698–712.

[234] David Boud and Elizabeth Molloy. “Feedback in higher and professional education”. In: *Understanding It and Doing It Well* (2013), p. 2013.

[235] Naomi Winstone and David Carless. *Designing effective feedback processes in higher education: A learning-focused approach*. Routledge, 2019.

[236] Margaret Price, Karen Handley, and Jill Millar. “Feedback: Focusing attention on engagement”. In: *Studies in higher education* 36.8 (2011), pp. 879–896.

[237] Miguel Lara and Kate Lockwood. “Hackathons as community-based learning: a case study”. In: *TechTrends* 60.5 (2016), pp. 486–495.

[238] Naomi E Winstone and David Boud. “The need to disentangle assessment and feedback in higher education”. In: *Studies in higher education* 47.3 (2022), pp. 656–667.

[239] Cristin Goodwin, J Paul Nicholas, et al. “A framework for cybersecurity information sharing and risk reduction”. In: ().

[240] Slavi Stoyanov and Paul Kirschner. “Effect of problem solving support and cognitive styles on idea generation: Implications for technology-enhanced learning”. In: *Journal of Research on Technology in Education* 40.1 (2007), pp. 49–63.

[241] Jeanette Falk Olesen and Kim Halskov. “10 years of research with and on hackathons”. In: *Proceedings of the 2020 ACM designing interactive systems conference*. 2020, pp. 1073–1088.

[242] Shivangi Dhawan. “Online learning: A panacea in the time of COVID-19 crisis”. In: *Journal of Educational Technology Systems* 49.1 (2020), pp. 5–22.

[243] Ian Clark. “Formative assessment: Assessment is for self-regulated learning”. In: *Educational psychology review* 24 (2012), pp. 205–249.

[244] Joyce Wangui Gikandi, Donna Morrow, and Niki E Davis. “Online formative assessment in higher education: A review of the literature”. In: *Computers & education* 57.4 (2011), pp. 2333–2351.

[245] Kiev Gama, Carlos Zimmerle, and Pedro Rossi. “Online Hackathons as an Engaging Tool to Promote Group Work in Emergency Remote Learning”. In: *arXiv preprint arXiv:2105.06388* (2021).

[246] Rachelle Esterhazy and Crina Damşa. “Unpacking the feedback process: An analysis of undergraduate students’ interactional meaning-making of feedback comments”. In: *Studies in Higher Education* 44.2 (2019), pp. 260–274.

[247] Stephen Merry et al. *Reconceptualising feedback in higher education: Developing dialogue with students*. Routledge, 2013.

[248] Phillip Dawson et al. “What makes for effective feedback: Staff and student perspectives”. In: *Assessment & Evaluation in Higher Education* 44.1 (2019), pp. 25–36.

[249] Birgit Penzenstadler et al. “Everything is INTERRELATED: teaching software engineering for sustainability”. In: *Proceedings of the 40th International Conference on Software Engineering: Software Engineering Education and Training*. 2018, pp. 153–162.

[250] Jun He. “Counteracting free-riding with team morale—An experimental study”. In: *Project Management Journal* 43.3 (2012), pp. 62–75.

[251] Matthew J Pearsall and Vijaya Venkataramani. “Overcoming asymmetric goals in teams: The interactive roles of team learning orientation and team identification.” In: *Journal of Applied Psychology* 100.3 (2015), p. 735.

[252] Mürüvvet Büyükboyacı and Andrea Robbett. “Collaboration and free-riding in team contests”. In: *Labour Economics* 49 (2017), pp. 162–178.

[253] Ching-Huei Chen, Jun-Han Liu, and Wen-Chuan Shou. “How competition in a game-based science learning environment influences students’ learning achievement, flow experience, and learning behavioral patterns”. In: *Journal of Educational Technology & Society* 21.2 (2018), pp. 164–176.

[254] Jillian Ruhl and Daphne Lordly. “The nature of competition in dietetics education: a narrative review”. In: *Canadian Journal of Dietetic Practice and Research* 78.3 (2017), pp. 129–136.

[255] Yunkai Liu, Jeremy C Cannell, and John H Coffman. “Gannon University Hackathon: A combination of virtual and onsite education event to recruit high-school students within cybersecurity major”. In: *2020 IEEE Frontiers in Education Conference (FIE)*. IEEE. 2020, pp. 1–4.

[256] Kevin Kam Fung Yuen and Amy Ooi Mei Wong. “Designing an Effective Hackathon via University-Industry Collaboration for Data Science Education”. In: *2021 IEEE International conference on engineering, technology & education (TALE)*. IEEE. 2021, pp. 1–6.

[257] Sakhumuzi Mhlongo, Kayode Emmanuel Oyetade, and Tranos Zuva. “The effectiveness of collaboration using the hackathon to promote computer programming skills”. In: *2020 2nd International Multidisciplinary Information Technology and Engineering Conference (IMITEC)*. IEEE. 2020, pp. 1–6.

[258] Yossi Maaravi. “Using hackathons to teach management consulting”. In: *Innovations in Education and Teaching International* 57.2 (2020), pp. 220–230.

[259] Jennifer Wilson, Kimberly Bender, and Jonah DeChants. “Beyond the classroom: the impact of a university-based civic hackathon addressing homelessness”. In: *Journal of Social Work Education* 55.4 (2019), pp. 736–749.

[260] Jeanette Falk et al. “The Future of Hackathon Research and Practice”. In: *arXiv preprint arXiv:2211.08963* (2022).

[261] Janine L Spears. “Gaining real-world experience in information security: A roadmap for a service-learning course”. In: *Journal of Information Systems Education* 29.4 (2018), pp. 183–202.

[262] Cecilia La Place et al. “Engineering students rapidly learning at hackathon events”. In: *2017 ASEE Annual Conference & Exposition*. 2017.

[263] Martin Fowler. *UML distilled: a brief guide to the standard object modeling language*. Addison-Wesley Professional, 2004.

[264] Debasis Dash et al. “Internet of Things (IoT): The New Paradigm of HRM and Skill Development in the Fourth Industrial Revolution (Industry 4.0).” In: *IUP Journal of Information Technology* 15.4 (2019).

[265] Melissa Carlton and Yair Levy. “Cybersecurity skills: Foundational theory and the cornerstone of advanced persistent threats (APTs) mitigation”. In: *Online Journal of Applied Knowledge Management (OJAKM)* 5.2 (2017), pp. 16–28.

[266] Ronald S Cheung et al. “Challenge based learning in cybersecurity education”. In: *Proceedings of the International Conference on Security and Management (SAM)*. Citeseer. 2011, p. 1.

[267] Stylianos Karagiannis and Emmanouil Magkos. “Adapting CTF challenges into virtual cybersecurity learning environments”. In: *Information & Computer Security* (2020).

[268] TJ OConnor and Christopher Stricklan. “Teaching a hands-on mobile and wireless cybersecurity course”. In: *Proceedings of the 26th ACM Conference on Innovation and Technology in Computer Science Education V. 1*. 2021, pp. 296–302.

[269] Alexander Nolte et al. “You Hacked and Now What? -Exploring Outcomes of a Corporate Hackathon”. In: *Proceedings of the ACM on Human-Computer Interaction* 2.CSCW (2018), pp. 1–23.

[270] Anol Bhattacherjee. “Understanding information systems continuance: an expectation-confirmation model”. In: *MIS quarterly* (2001), pp. 351–370.

[271] J Sauro. “MeasuringU: Measuring Usefulness”. In: *Measuringu. com* (2011).

[272] Alién García-Hernández and Teresa González-Ramírez. “Construction and validation of a questionnaire to assess student satisfaction with math-

ematics learning materials". In: *Proceedings of the Sixth International Conference on Technological Ecosystems for Enhancing Multiculturality*. 2018, pp. 134–138.

Appendix A. INTELLIGENT TRANSPORTATION SYSTEM USE-CASES

The descriptions below provides a general overview of the ITS system use-case and, thus, are not an exhaustive explanation of the system-component interaction. You are allowed to assume the existence of lower-level components not explicitly mentioned in the case but are vital to any working software system.

A.1. Autonomous Vehicle Traffic Light Perception System

A.2. Autonomous Vehicle Parking IoT Use-case

The autonomous vehicle parking system is composed of various elements such as the Driver (D), Autonomous Vehicle (AV), Parking Service Provider (PSP), and Parking Lot Terminal (PLT). For example, the Driver can use a mobile device like a phone or tablet to connect with the PSP and send commands to the AV. Meanwhile, the PSP may own and manage multiple PLTs where AVs can be parked.

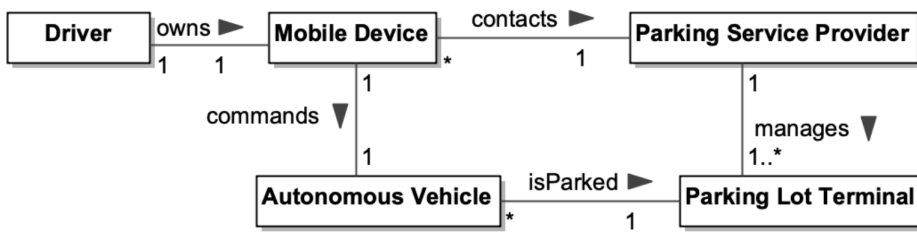


Figure 30. Autonomous vehicle parking IoT use-case

- **Autonomous Vehicle (AV)**: The Autonomous Vehicle (AV) can sense its surroundings and operate without any driver intervention.
- **Parking Service Provider (PSP)**: The Parking Service Provider (PSP) is an online server group that provides on-demand parking services to users subscribed to a parking management company. This includes various services such as checking and finding nearby parking spaces, marking parking space reservations and other premium services, checking parking lot availability, and registering/removing drivers from the service.
- **Parking Lot Terminal (PLT)**: The Parking Lot Terminal (PLT) is a terminal deployed by the parking lot owner responsible for monitoring and managing the parking lot using IoT devices such as cameras and sensors. The PLT manages access to the parking lot for AVs and controls and releases parking permits to drivers via the PSP.

- **Driver (D)**: The Driver (D) is a person who uses the AV and requires parking services with the ability to initiate the parking service.

The parking service encompasses several processes, such as registration with the service provider, checking parking availability and issuing parking permits, parking the vehicle, retrieving the vehicle, and paying for the parking service. To initiate the parking service, the Driver contacts the PSP and registers the AV. Once the parking request is received, the PSP checks the PLTs for available space and issues the parking permit if space is available. The Driver then leaves the AV at the designated drop/pick-up area in the PLT and submits the AV's parking permit. The AV navigates and parks in the assigned space in the PLT. When the Driver wants to retrieve the vehicle, they send a command to the AV, which navigates to the drop/pick-up area. After parking is complete, the PLT informs the PSP, and the PSP issues an invoice and contacts the Driver, who makes the payment. Once payment is made, the PSP is notified.

A.3. Bike Sharing System IoT Use-case

The Smart Bike Share system has a fleet of 750 bikes, including 500 electric-assist bikes and 250 8-speed bikes, providing bike-sharing services to its users. The system is composed of four main components: Smart Bike (SB), Bike Share Website (BWA), Bike Mobile Application (BMA), and Rider (R).

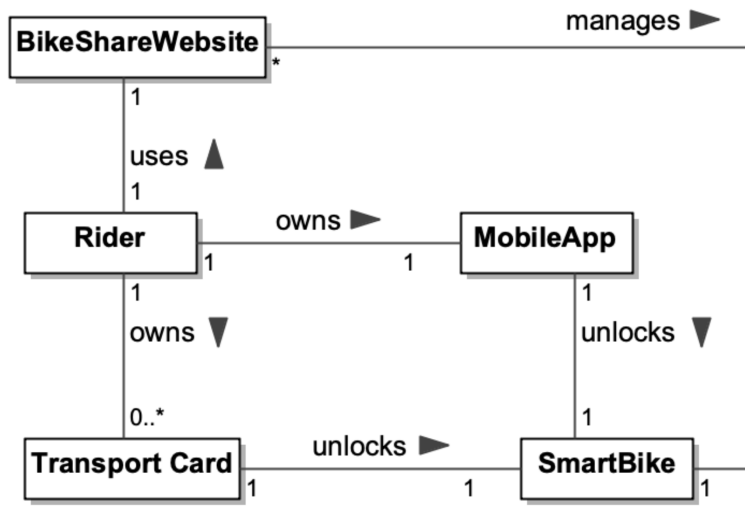


Figure 31. Bike sharing system IoT use-case

- **Smart Bike (SB)**: The smart bikes have smart sensors (such as GPS receiver and RFID sensors) and network capabilities (4G) that allow real-time communication with other system components. The bikes provide information on bike statistics and customer activity and can respond promptly in emergencies.

- **Bike Share Website (BWA)**: The Bike Share Website provides services such as account signup, bike-share membership purchase, and checking bike and dock availability at the nearest station. Users can also link their personalized bus card to their rider account for free membership.
- **Bike Mobile Application (BMA)**: The Bike Mobile Application is available for both Android and iOS devices and enables users to purchase bike-share memberships, unlock bikes at any designated station, access their rider profile, check the bike and dock availability, report defects and emergencies, and make inquiries to customer service.
- **Rider (R)**: The rider is a registered user on the smart bike-share system with a valid and active account, allowing access to the bike-sharing services provided by the system. A rider can only use one smart bike at a time.

This ride service is a crucial functional process of the smart bike-sharing system, comprising several steps, such as finding the bike dock location, confirming bike availability, unlocking the selected bike, riding the bike to the intended destination, and returning the bike to the nearest dock station. Once the ride is completed, the BWA issues an invoice and automatically processes the payment, with all relevant information recorded by the BMA and BWA.

A.4. Scooter Ride-Hailing System IoT Use-case

The system provides micro-mobility services consisting of different components: Scooter(**S**), Scooter Backend (**SB**), Scooter Mobile Application (**SMA**), and Rider(**R**).

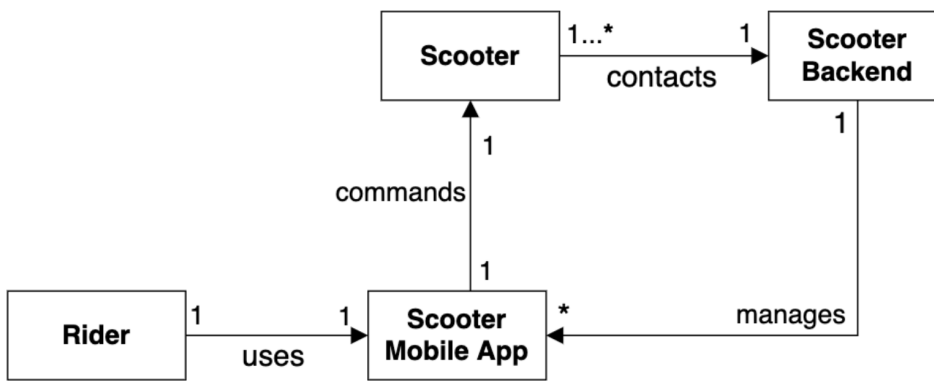


Figure 32. Scooter ride-hailing system IoT use-case

- **Scooter(S)**: The scooter component of the system is used to fulfil commutes. The scooter chassis (external hardware) houses its wheels, lights, batteries, cables, and connectors. Inside its chassis, the scooter contains

various perception (i.e., sensing, positioning, actuating), network, and application (i.e., storage) assets. Below are the important assets classified by their information processing functions.

- **Scooter Backend (SB)**: Scooter backend is comprised of systems that help to monitor the status and location of a fleet of scooters, send commands to the scooter to lock/unlock, manage the user accounts and scooter ride activities. The **SB** can only be accessed through an administrative web interface.
- **Scooter Mobile Application (SMA)**: The scooter mobile application comes in Android and iOS implementations comprising the rider profile, ride-hailing, and billing components.
- **Rider (R)**: A rider is a user registered on the system and possesses a valid and active account. A rider should have access to the system's scooter (**S**) services and cannot use more than one at a time.

Appendix B. AV LABORATORY SET-UP FOR IOTA-SRM CASE ANALYSIS

1. **Hardware**: The primary component of the experimental set-up is a Lexus RX450h, an SUV model equipped with all the necessary sensors for basic autonomy. The vehicle and its hardware were adapted by AutonomouStuff and powered by our in-house software. The following provides a detailed list of the significant hardware and sensors employed:
   - *Vehicle*
     - Lexus RX450h autonomous driving vehicle adapted by AutonomouStuff
   - *Lidar*
     - Ouster OS1-128
     - Velodyne VLP-32C
   - *Cameras*
     - 2x Allied Vision Mako G-319C
     - 4x Sekonix (3x SF3324, 1x SF3325)
     - Comma two devkit (dashcam)
   - *Global Navigation Satellite System Device*
     - NovAtel PwrPak7D-E2
   - *Radar*
     - Delphi ESR 2.5 24V
   - *By-wire Kit*
     - PACMod
   - *Computers*
     - AStuff Spectra
2. **Software**: The software running on our car is Autoware Mini, a minimalistic Python-based autonomy software developed in-house under a permissive MIT license.
   - *Autoware Mini*: Autoware Mini is built on Python and ROS 1, utilizing Autoware messages to define the interfaces between its modules. It is compatible with Autoware and works on ROS Noetic (available on Ubuntu 20.04 or other Linux versions, with the help of Conda RoboStack).
   - *Goals*:
     - Easy to get started with minimal dependencies
     - Simple and pedagogical, using simple Python nodes and ROS 1
     - Easy to implement machine learning-based approaches
   - *Software Architecture*:
     - Localization
     - Global Planner, Local Planner
     - Obstacle Detection
     - Traffic Light Detection

Appendix C. OCTAVE ALLEGRO TEMPLATE FOR RISK DOCUMENTATION

C.1. Template for Risk Documentation

Table 50. Template for risk documentation

Allegro – Worksheet 10											
Threat	**IoT Layer Affected**										
	Business Asset										
	Business Asset's Value										
	Area of Concern										
	Actor Who would exploit the area of concern or threat?										
	Means How would the actor do it? What would they do?										
	Motive What is the actor's reason for doing it?										
	Outcome (choose one) What would be the resulting effect be?	Disclosure:		Destruction:		Modification:		Interruption:			
	Security Requirements How would the information asset's security requirements be breached?										
	Likelihood (choose one)	High:		Medium:		Low:					

Consequences What are the consequences to the organisation as a result of the risk?	**Severity** How severe are the consequences to the organisation or asset owner by impact area? *3 for highest priority, 2 for medium, 1 for lowest			
	Impact area	**Priority***	**Impact**	**Score**
	Confidentiality			
	Availability			
	Integrity			
	Relative risk score:			
	Total Risk Score (Rel x likelihood):			

C.2. Template for Risk Mitigation Documentation

Table 51. Template for risk mitigation documentation

Risk Mitigation								
Choose action to take.	Accept:		Defer:		Mitigate:		Transfer:	
For the risk, what actions and controls will be used:								
Layer where applied	**Description of control or action**					**Estimated cost**		

C.3. Criteria Used in the OCTAVE Worksheets

1. **Value of Business Asset**: The importance of an asset is gauged by its contribution to the system's functionality. Specifically, the focus is on the system's behaviour if the asset's data is compromised. The categorization based on the value of the business asset is:

- Low – The system remains functional even without the asset.
- Medium – The system functions but encounters performance issues.
- High – The system's operations halt without the asset.

2. **Likelihood of an Attack**:
   - **Low**: An attack is considered to have a low chance of occurrence if:
     - Specialized tools are required, and their cost is high;
     - Expertise level needed for the attack is advanced;
     - The opportunity to launch the attack is limited and requires extensive preparation.
   - **Medium**: An attack is seen as moderately probable if:
     - Tools are readily available but might need slight modifications;
     - An attacker with moderate skills can execute the attack;
     - The time frame for the attack and its preparation are average.
   - **High**: An attack has high likelihood if:
     - Tools are either not required or are easy to access;
     - Minimal knowledge is sufficient, allowing even novices to attack;
     - The preparation time for the attack and its setup is minimal.

Table 52. Risk measurement criteria

Impact Area	**Low**	**Medium**	**High**
Confidentiality	Confidentiality remains intact	Minor breaches on low-priority data	Significant breaches on high-priority and classified data
Availability	Minimal component downtime with no system impact	Component briefly unavailable causing minor performance issues	Prolonged component unavailability, system-wide impact
Integrity	Component and data integrity intact	Minor integrity compromise with no performance impact	Severe integrity loss leading to system performance degradation

Appendix D. CYBERSECURITY COURSE DESCRIPTION

In our action research studies in Chapter 4, we introduced the hackathon approach in a cybersecurity course. Table 53 details the course information, description and learning outcomes.

Table 53. Cybersecurity course description

Secure Software Design Course	
Credits	6 ECTS
Curricula	Software Engineering, Cybersecurity, Conversion IT Masters
This course will provide students an overview of the principles for secure software design. The participants will learn what security risk management is, and how to ensure confidentiality, integrity and availability of secure assets. The course will also analyse how to engineer and model security requirements, what are the major security controls, like role-based access control and cryptography (only short introduction). The course also includes the overview of the principles for the model driven security. A number of lectures will be given to understand what are the principles of security development processes and what the security patterns are. A number of invited talks is planned to introduce participants with the practical security solutions and best practices. During the practical seminars, there will be a number of exercises, concentrating on the deepening the theoretical knowledge given during the lectures.	
Learning outcomes	On successful completion of this course, students will able to: 1. Identify causes and consequences of (lack of) system and software security. 2. Master essential techniques to reduce and avoid system and software security problems, to introduce and reason on security requirements and controls. 3. Apply advanced modelling techniques (notations, tools, and processes) to build secure systems and software.

Appendix E. HACKATHON DATA COLLECTION INSTRUMENTS

E.1. Action Research Cycle 1

Table 54: Action research cycle 1: Post-hackathon questionnaire instrument

Team process (based on Nolte *et al.*[269]), anchored between strongly disagree and strongly agree.
I was unclear about the goals and objectives for my work in this team.
I was unsure how my work relates to the overall objectives of my team.
Tasks were mainly distributed based on individual SKILLS.
Tasks were mainly distributed based on individual INTERESTS.
Perceived satisfaction with team process (based on Filippova *et al.*[185]), anchored between strongly disagree and strongly agree.
I am satisfied with the work completed in my project
I am satisfied with the quality of my team's output
My expectations towards my project were met
I intend to continue working on our hackathon project
I intend to continue contributing to the security community
Perceived satisfaction with team process (based on Bhattacherjee[270]), anchored between 1 and 5.
(1) Inefficient to (5) Efficient
(1) Uncoordinated to (5) Coordinated
(1) Unfair to (5) Fair
(1) Confusing to (5) Easy to understand
Perceived learning experience (based on Filippova *et al.*[185]), anchored between strongly disagree and strongly agree.
Working with my team helped me learn more about cybersecurity
Working on my tasks within my team helped me learn more about cybersecurity
Practising previous experience within my project helped me learn more about cybersecurity
The security presentations made at the event helped me learn more about cybersecurity
Talking with my mentors helped me learn more about cybersecurity
Participating in the pre-event Idea Garage helped me learn more about cybersecurity
My expectations towards learning during this event were met
The security presentations made at the event impacted the outcome of my project
Perceived satisfaction with learning experience (based on Filippova *et al.*[185]), anchored between strongly disagree and strongly agree.
I am satisfied with the practices I learned during the event
I am satisfied with the quality of the practices I learned during the event
My expectations towards the practices I learned during the event were met
Overall, the practices I learned during the event will be useful in my future career
I intend to continue learning about security

Preparation for hackathon (based on Nolte *et al.,*[269]), anchored between not at all and completely
I learned about topics that I thought would be useful for our project
I developed a project idea
I formed a team
I met with my team, and we discussed our project
Team familiarity (based on Filippova *et al.*[185]), anchored between not at all and completely
I know my team members well
I have collaborated with some of my team members before
I have been close to some of my team members before
I have socialized/met with some of my team members outside of work/school before

Table 55. Action research cycle 1: Post-hackathon interview instrument

1. How was the hackathon from your perspective in the form of:
    - What did you do after you arrived?
    - How did you see the event play out?
2. Did you attend the idea generation pre-hackathon event?
    - What idea did you develop?
    - How else did you prepare for this hackathon?
3. What were the outcomes as a result of learning? [*mentors, security talks, team members, working on the project*]
4. How do you perceive the outcome of the hackathon?
    - Were you satisfied?
    - How did you see your teamwork?
5. Did you discover new security knowledge during the hackathon?
    - How did you discover this?
6. What about the continuity of your project?
    - Have you used anything learned during the hackathon already?
    - Are you planning to use it in the future?

Table 56. Action research cycle 1: Post-hackathon interview coding system

1. Hackathon Perspective
    - Arrival Actions: First steps upon arrival, e.g., networking, team formation, preliminary research.
    - Event Progression: Observations of event flow, participation levels.
2. Pre-Hackathon Participation
    - Attendance: Yes/No.
    - Idea Development: Nature of idea, continuation with idea, feedback to idea.
    - Preparation Modes: Reading, courses, prior knowledge, team discussions.
3. Learning Outcomes
    - Mentor Interactions: Guidance, feedback.
    - Security Talks: Security relevance, applicability.
    - Team Dynamics: Knowledge sharing, role/task distribution, synergy in team process.
    - Project Work: Challenges, discoveries, milestones.
4. Hackathon Outcome Perception
    - Satisfaction Level: Satisfied, Neutral, Unsatisfied.
    - Teamwork Assessment: Effective, needs improvement, diverse skills, communication quality.
5. Security Knowledge
    - Knowledge Sources: Mentors, peers, security talks, idea generation, presentations, independent research.
    - Nature of Knowledge: Theoretical, practical, tools, techniques.
6. Project Continuity
    - Post-Hackathon Application: Instances of applying hackathon learnings.
    - Future Plans: Continuation of the project, further research, team collaborations.

E.2. Action Research Cycle 2

Table 57. Action research cycle 2: Post-hackathon questionnaire instrument

Team familiarity (based on Filippova *et al.*[185]), anchored between not at all and completely
I know my team members well
I have collaborated with some of my team members before
I have socialized/met with some of my team members outside of work/school before
Team process (based on Bhattacherjee [270]), anchored between 1 and 5
I am satisfied with the work completed in my project
I am satisfied with the quality of my team's output
My ideal outcome coming into my team was achieved
My expectations towards my team were met
Perceived satisfaction with team process (based on Filippova *et al.* [185]), anchored between strongly disagree and strongly agree.
(1) Inefficient to (5) Efficient
(1) Uncoordinated to (5) Coordinated
(1) Unfair to (5) Fair
(1) Confusing to (5) Easy to understand
Team goal clarity (based on Nolte *et al.,* [269]) anchored between strongly disagree and strongly agree.
I was uncertain of my duties and responsibilities in this team
I was unclear about the goals and objectives for my work
I was unsure how my work relates to the overall objectives of my team
Perception of team participation and voice (based on Nolte *et al.* [269]) anchored between strongly disagree and strongly agree.
Everyone had a chance to express her/his opinion.
The team members responded to the comments made by others
The team members participated very actively during our collaboration
Overall, the participation of each team member was effective
Perception of the usefulness of the interventions (based on Sauro [271]) anchored between strongly disagree and strongly agree.
Using the *[intervention]* enabled me to accomplish tasks more quickly
Using the *[intervention]* improved my team's performance
Using the *[intervention]* increased my productivity in the hackathon
Using the *[intervention]* enhanced my effectiveness in my team
Using the *[intervention]* made it easier to complete my *[hackathon]* solution
I found the *[intervention]* useful in my team
Learning outcome measured students' perception of achieving the course's learning outcomes, perceived learning process, and learning through problem-solving; anchored between strongly disagree and strongly agree.
The hackathon events allowed me the opportunity to design secure systems and software
The hackathon activities made my learning experience more productive
There were enough opportunities during the course to find out if I clearly understood the course material
The *[interventions]* given were geared to promote my understanding

E.3. Action Research Cycle 3

Table 58. Action research cycle 3: Post-hackathon questionnaire instrument

Perception of the usefulness of the interventions (based on [271]) anchored between strongly disagree and strongly agree.
Using the *[intervention]* enabled me to accomplish tasks more quickly
Using the *[intervention]* improved my team's performance
Using the *[intervention]* increased my productivity in the hackathon
Using the *[intervention]* enhanced my effectiveness in my team
Using the *[intervention]* made it easier to complete my *[hackathon]* solution
I found the *[intervention]* useful in my team.
Perception of the level of agreement about students' evaluation of the intervention at the hackathon event (based on [272]), anchored between strongly disagree and strongly agree
The [*intervention*] enhanced my satisfaction with the study of [*hackathon*] activity
The [*intervention*] contributed to better learning of [*hackathon*] activity
The [*intervention*] were easy to understand and connected to my learning interests
The [*intervention*] made me forget how difficult the [*hackathon*] activity is
Perception of the level of agreement about the interventions' contribution to learning at the hackathon event (based on [272]) anchored between strongly disagree and strongly agree
The [*intervention*] linked its contents with my security interests
The [*intervention*] made visible the linking of the [*hackathon*] activity with the real world
The [*intervention*] was adapted to my learning rhythm
Learning outcome measured students' perception of the hackathon learning process (based on [69]) anchored between strongly disagree and strongly agree.
The hackathons case study resembled a real-life situation
The hackathons facilitated independent problem-solving
The hackathons allowed me the opportunity to design secure systems and software
The hackathon activities made my learning experience more productive
The hackathons provided enough opportunities during the course to find out if I clearly understood the course material
Open ended questions.
Is there anything else you want to tell us about your [*intervention*] experience?
Is there anything else you want to tell us about your overall learning experience?

ACKNOWLEDGEMENTS

I am profoundly grateful to my supervisors, Raimundas Matulevičius and Alexander Nolte, for their unwavering support, mentorship, and guidance throughout my PhD journey. Their extensive knowledge and invaluable experience have been a constant source of inspiration for my research endeavours.

I offer my sincere thanks to Guttorm Sindre and Jari Porras for their invaluable feedback and constructive criticisms that have significantly shaped and enhanced the quality of my thesis. I am profoundly thankful to Anastasija Nikiforova for all the insightful comments in finalising the thesis.

I wish to convey my profound thanks to my colleagues at the Information Security research group and the Institute of Computer Science, University of Tartu, for the collaborations and discussions that have enriched my doctoral experience. Most special thanks to Kristiina, Danielle, María and Mari for their notable contributions on my PhD journey.

I owe a debt of gratitude to my family: my father, Obot Udo Affia, my mother, Uduak Obot Affia, and my sister, Mmekom Obot Affia, for their love, prayers, and constant encouragement even from afar. To my fiancé, Valdis, I extend my deepest gratitude for his constant support and faith in me. To all my colleagues, friends, and extended family (especially my PUSH group sisters) – your constant encouragement has been invaluable.

Above all, I am grateful to God – in whom I live, move and have my being.

SISUKOKKUVÕTE

IoT turvariskide haldamise raamistik ja õpetamismeetod

Kuna asjade Interneti (IoT) kasutus jätkab kasvamist, suureneb ka sellega seotud küberrünnakute oht vaatamata kiireloomulisele kasvule on teoreetiliste turvameetmete ja IoT-i tegeliku rakendamise vahel endiselt märkimisväärne lõhe. Probleemi juur peitub IoT-süsteemide keerukuses ja praktiliste turvariskide haldamise (SRM) meetodite puudumises. Selle lahendamiseks oleme välja töötanud raamistiku, mida tuntakse kui IoT Arhitektuuripõhist Turvariskide Haldamise süsteemi (IoTA-SRM). Süsteem on loodud teooria mõistmise ja praktilise rakendamise vahelise lõhe täitmiseks. IoTA-SRM-i mitmekülgne lähenemine, mis integreerib IoT arhitektuuri riskide haldamisse. See koosneb neljast peamisest etapist: 1) süsteemi modelleerimine, 2) riskide tuvastamine, 3) riskide käsitlemine ja 4) kompromisside analüüs, millest igaüks hõlmab konkreetseid ülesandeid. IoTA-SRM raamistikku testiti kahe autonoomsetele sõidukitele (AV) keskendunud juhtumiga. Esimesel juhul kasutati raamistikku autonoomse sõiduki mitmekihilise turvariskide haldamise läbiviimiseks laboritingimustes. Tuvastati erinevaid ohte läbi taju-, võrgu- ja rakenduskihtide. Iga oht võib realiseeruda unikaalseteksturvariskideks, mis omakorda võivad omada kaskaadeerivat mõju teistele IoT arhitektuuri kihtidele. Riskihindamise põhjal võisime pakkuda välja tõhusad turvameetmed ja teha teadlikke otsuseid nende riskide vähendamiseks. Teine juhtum hõlmas IoTA-SRM raamistiku rakendamist pilootprojektis, et hõlbustada MQTT-põhist suhtlust AV ja linna liiklusvalgustuse taristu vahel. Taas tuvastati potentsiaalsed riskid ja pärast sidusrühmadega konsulteerimist seati paika sobivad turvameetmed. Mõlemad need stsenaariumid näitasid, et IoTA-SRM raamistik analüüsib IoT-süsteeme kiht-kihilt, võimaldades põhjalikku riskihindamist ja järgnevat riskide leevendamist.

Kuigi IoTA-SRM raamistikul on palju häid omadusi IoT turvalisuse parandamiseks, on selle raamistiku rakendamine väljakutsete rohke. Teoreetiline raamistik on praktilise rakendamise jaoks keeruline. Selle lõhe ületamiseks teooria ja praktika vahel oleme välja töötanud unikaalse häkatoni-põhise meetodi praktilise õppimise hõlbustamiseks. Meetod julgustab raamistiku "käed-küljes" rakendamist, pakkudes samal ajal sihitud sekkumisi, et aidata osalejatel tõhusamalt õppida. Oleme seda lähenemist kolme-tsüklilise juhtumi-uuringu jooksul täiustanud. Alguses keskendusime sekkumistele nagu ideede genereerimine, temaatiline sisend, sihitud tagasiside, koostöötoetus ja võistlusstiil. Juhtumiuuringu kolmanda tsükli ajal keskendusime sihitud tagasisidele ja temaatilisele sisendile, mis näitasid otseseid õppimise eeliseid. Kogu häkaton olistruktureeritud IoTA-SRM raamistiku kasutamisele. Lõpptulemuseks koostasime põhjaliku mudeli häkatoni vormis õpetamiseks, mis ühendab IoTA-SRM raamistiku teoreetilise ranguse praktilise "käed-küljes" õppimisega. Selline integreeritud mudel võeti osalejate poolt hästi vastu, empiirilised õpitulemuste andmed näitasid arusaamise paranemist IoT SRM tavade rakendamiseks. Osalejad suutsid süstemaatiliselt analüüsida IoT-süsteemi kihte, teostada põhjalikke riskihindamisi ja rakendada tõhusaid strateegiaid riskide vähendamiseks.

Kokkuvõttes pakub meie töö üksikasjalikku teekaarti neile, kes on huvitatud IoTA-SRM raamistiku õpetamisest ja rakendamisest professionaalsete koolitusprogrammide, IoT-le spetsialiseerunud töötubade või IoT turvalisusele keskendunud hariduskavade abil. Selle õpetamismudeli modulaarne iseloom võimaldab paindlikkust ja kohandatavust, muutes selle väärtuslikuks vahendiks järgmise põlvkonna küberturbe spetsialistide varustamiseks oskustega, mida nad vajavad IoT turvariskide haldamiseks.

CURRICULUM VITAE

Personal data

Name:	Abasi-amefon Obot Affia
Date of Birth:	04.11.1993
Citizenship :	Nigeria
E-mail:	amefon.affia@ut.ee

Education

2019–2023	Doctor of Philosophy in Computer Science, University of Tartu, Tartu, Estonia
2016–2018	Master of Science in Cyber Security, Tallinn University of Technology, Tallinn, Estonia
2010–2015	Bachelor of Engineering in Computer Engineering, University of Uyo, Uyo, Nigeria

Employment

2018–	Junior Research Fellow, University of Tartu, Tartu, Estonia
2017–2020	Cyber Security Engineer, AFS IT Services Estonia OÜ, Tallinn, Estonia
2017 – 2017	QA Engineer, AFS IT Services Estonia OÜ, Tallinn, Estonia
2014–2016	Technical Manager, Lazarus Care Mission International (LCMI), Port-Harcourt, Nigeria

Scientific work

Main fields of interest:

- Security Risk Management
- Hackathons
- Intelligent Infrastructure

ELULOOKIRJELDUS

Isikuandmed

Nimi:	Abasi-amefon Obot Affia
Sünniaeg:	04.11.1993
Kodakondsus:	Nigeeria
E-post:	amefon.affia@ut.ee

Haridus

2019–2023	Arvutiteaduse filosoofiadoktor, Tartu Ülikool, Tartu, Eesti
2016–2018	Küberkaitse magistrant, Tallinna Tehnikaülikool, Tallinn, Eesti
2010–2015	Arvutitehnika bakalaureusekraad, Uyo Ülikool, Uyo, Nigeeria

Teenistuskäik

2018–	Nooremteadur, Tartu Ülikool, Tartu, Eesti
2017–2020	Küberturbe insener, AFS IT Services Estonia OÜ, Tallinn, Eesti
2017 – 2017	QA insener, AFS IT Services Estonia OÜ, Tallinn, Eesti
2014–2016	Tehniline juht, Lazarus Care Mission International (LCMI), Port-Harcourt, Nigeeria

Teadustegevus

Peamised uurimisvaldkonnad:

- Turvariskide juhtimine
- Hackathons
- Intelligentne infrastruktuur

DISSERTATIONES INFORMATICAE PREVIOUSLY PUBLISHED IN DISSERTATIONES MATHEMATICAE UNIVERSITATIS TARTUENSIS

19. **Helger Lipmaa.** Secure and efficient time-stamping systems. Tartu, 1999, 56 p.
22. **Kaili Müürisep.** Eesti keele arvutigrammatika: süntaks. Tartu, 2000, 107 lk.
23. **Varmo Vene.** Categorical programming with inductive and coinductive types. Tartu, 2000, 116 p.
24. **Olga Sokratova.** Ω-rings, their flat and projective acts with some applications. Tartu, 2000, 120 p.
27. **Tiina Puolakainen.** Eesti keele arvutigrammatika: morfoloogiline ühestamine. Tartu, 2001, 138 lk.
29. **Jan Villemson.** Size-efficient interval time stamps. Tartu, 2002, 82 p.
45. **Kristo Heero.** Path planning and learning strategies for mobile robots in dynamic partially unknown environments. Tartu 2006, 123 p.
49. **Härmel Nestra.** Iteratively defined transfinite trace semantics and program slicing with respect to them. Tartu 2006, 116 p.
53. **Marina Issakova.** Solving of linear equations, linear inequalities and systems of linear equations in interactive learning environment. Tartu 2007, 170 p.
55. **Kaarel Kaljurand.** Attempto controlled English as a Semantic Web language. Tartu 2007, 162 p.
56. **Mart Anton.** Mechanical modeling of IPMC actuators at large deformations. Tartu 2008, 123 p.
59. **Reimo Palm.** Numerical Comparison of Regularization Algorithms for Solving Ill-Posed Problems. Tartu 2010, 105 p.
61. **Jüri Reimand.** Functional analysis of gene lists, networks and regulatory systems. Tartu 2010, 153 p.
62. **Ahti Peder.** Superpositional Graphs and Finding the Description of Structure by Counting Method. Tartu 2010, 87 p.
64. **Vesal Vojdani.** Static Data Race Analysis of Heap-Manipulating C Programs. Tartu 2010, 137 p.
66. **Mark Fišel.** Optimizing Statistical Machine Translation via Input Modification. Tartu 2011, 104 p.
67. **Margus Niitsoo**. Black-box Oracle Separation Techniques with Applications in Time-stamping. Tartu 2011, 174 p.
71. **Siim Karus.** Maintainability of XML Transformations. Tartu 2011, 142 p.
72. **Margus Treumuth.** A Framework for Asynchronous Dialogue Systems: Concepts, Issues and Design Aspects. Tartu 2011, 95 p.
73. **Dmitri Lepp.** Solving simplification problems in the domain of exponents, monomials and polynomials in interactive learning environment T-algebra. Tartu 2011, 202 p.

74. **Meelis Kull.** Statistical enrichment analysis in algorithms for studying gene regulation. Tartu 2011, 151 p.
77. **Bingsheng Zhang.** Efficient cryptographic protocols for secure and private remote databases. Tartu 2011, 206 p.
78. **Reina Uba.** Merging business process models. Tartu 2011, 166 p.
79. **Uuno Puus.** Structural performance as a success factor in software development projects – Estonian experience. Tartu 2012, 106 p.
81. **Georg Singer.** Web search engines and complex information needs. Tartu 2012, 218 p.
83. **Dan Bogdanov.** Sharemind: programmable secure computations with practical applications. Tartu 2013, 191 p.
84. **Jevgeni Kabanov.** Towards a more productive Java EE ecosystem. Tartu 2013, 151 p.
87. **Margus Freudenthal.** Simpl: A toolkit for Domain-Specific Language development in enterprise information systems. Tartu, 2013, 151 p.
90. **Raivo Kolde.** Methods for re-using public gene expression data. Tartu, 2014, 121 p.
91. **Vladimir Šor.** Statistical Approach for Memory Leak Detection in Java Applications. Tartu, 2014, 155 p.
92. **Naved Ahmed.** Deriving Security Requirements from Business Process Models. Tartu, 2014, 171 p.
94. **Liina Kamm.** Privacy-preserving statistical analysis using secure multi-party computation. Tartu, 2015, 201 p.
100. **Abel Armas Cervantes.** Diagnosing Behavioral Differences between Business Process Models. Tartu, 2015, 193 p.
101. **Fredrik Milani.** On Sub-Processes, Process Variation and their Interplay: An Integrated Divide-and-Conquer Method for Modeling Business Processes with Variation. Tartu, 2015, 164 p.
102. **Huber Raul Flores Macario.** Service-Oriented and Evidence-aware Mobile Cloud Computing. Tartu, 2015, 163 p.
103. **Tauno Metsalu.** Statistical analysis of multivariate data in bioinformatics. Tartu, 2016, 197 p.
104. **Riivo Talviste.** Applying Secure Multi-party Computation in Practice. Tartu, 2016, 144 p.
108. **Siim Orasmaa.** Explorations of the Problem of Broad-coverage and General Domain Event Analysis: The Estonian Experience. Tartu, 2016, 186 p.
109. **Prastudy Mungkas Fauzi.** Efficient Non-interactive Zero-knowledge Protocols in the CRS Model. Tartu, 2017, 193 p.
110. **Pelle Jakovits.** Adapting Scientific Computing Algorithms to Distributed Computing Frameworks. Tartu, 2017, 168 p.
111. **Anna Leontjeva.** Using Generative Models to Combine Static and Sequential Features for Classification. Tartu, 2017, 167 p.
112. **Mozhgan Pourmoradnasseri.** Some Problems Related to Extensions of Polytopes. Tartu, 2017, 168 p.

113. **Jaak Randmets.** Programming Languages for Secure Multi-party Computation Application Development. Tartu, 2017, 172 p.
114. **Alisa Pankova.** Efficient Multiparty Computation Secure against Covert and Active Adversaries. Tartu, 2017, 316 p.
116. **Toomas Saarsen.** On the Structure and Use of Process Models and Their Interplay. Tartu, 2017, 123 p.
121. **Kristjan Korjus.** Analyzing EEG Data and Improving Data Partitioning for Machine Learning Algorithms. Tartu, 2017, 106 p.
122. **Eno Tõnisson.** Differences between Expected Answers and the Answers Offered by Computer Algebra Systems to School Mathematics Equations. Tartu, 2017, 195 p.

DISSERTATIONES INFORMATICAE UNIVERSITATIS TARTUENSIS

1. **Abdullah Makkeh**. Applications of Optimization in Some Complex Systems. Tartu 2018, 179 p.
2. **Riivo Kikas**. Analysis of Issue and Dependency Management in Open-Source Software Projects. Tartu 2018, 115 p.
3. **Ehsan Ebrahimi**. Post-Quantum Security in the Presence of Superposition Queries. Tartu 2018, 200 p.
4. **Ilya Verenich**. Explainable Predictive Monitoring of Temporal Measures of Business Processes. Tartu 2019, 151 p.
5. **Yauhen Yakimenka**. Failure Structures of Message-Passing Algorithms in Erasure Decoding and Compressed Sensing. Tartu 2019, 134 p.
6. **Irene Teinemaa**. Predictive and Prescriptive Monitoring of Business Process Outcomes. Tartu 2019, 196 p.
7. **Mohan Liyanage.** A Framework for Mobile Web of Things. Tartu 2019, 131 p.
8. **Toomas Krips.** Improving performance of secure real-number operations. Tartu 2019, 146 p.
9. **Vijayachitra Modhukur.** Profiling of DNA methylation patterns as biomarkers of human disease. Tartu 2019, 134 p.
10. **Elena Sügis.** Integration Methods for Heterogeneous Biological Data. Tartu 2019, 250 p.
11. **Tõnis Tasa.** Bioinformatics Approaches in Personalised Pharmacotherapy. Tartu 2019, 150 p.
12. **Sulev Reisberg.** Developing Computational Solutions for Personalized Medicine. Tartu 2019, 126 p.
13. **Huishi Yin.** Using a Kano-like Model to Facilitate Open Innovation in Requirements Engineering. Tartu 2019, 129 p.
14. **Faiz Ali Shah.** Extracting Information from App Reviews to Facilitate Software Development Activities. Tartu 2020, 149 p.
15. **Adriano Augusto**. Accurate and Efficient Discovery of Process Models from Event Logs. Tartu 2020, 194 p.
16. **Karim Baghery.** Reducing Trust and Improving Security in zk-SNARKs and Commitments. Tartu 2020, 245 p.
17. **Behzad Abdolmaleki.** On Succinct Non-Interactive Zero-Knowledge Protocols Under Weaker Trust Assumptions. Tartu 2020, 209 p.
18. **Janno Siim.** Non-Interactive Shuffle Arguments. Tartu 2020, 154 p.
19. **Ilya Kuzovkin.** Understanding Information Processing in Human Brain by Interpreting Machine Learning Models. Tartu 2020, 149 p.
20. **Orlenys López Pintado.** Collaborative Business Process Execution on the Blockchain: The Caterpillar System. Tartu 2020, 170 p.
21. **Ardi Tampuu.** Neural Networks for Analyzing Biological Data. Tartu 2020, 152 p.

22. **Madis Vasser.** Testing a Computational Theory of Brain Functioning with Virtual Reality. Tartu 2020, 106 p.
23. **Ljubov Jaanuska.** Haar Wavelet Method for Vibration Analysis of Beams and Parameter Quantification. Tartu 2021, 192 p.
24. **Arnis Parsovs.** Estonian Electronic Identity Card and its Security Challenges. Tartu 2021, 214 p.
25. **Kaido Lepik.** Inferring causality between transcriptome and complex traits. Tartu 2021, 224 p.
26. **Tauno Palts.** A Model for Assessing Computational Thinking Skills. Tartu 2021, 134 p.
27. **Liis Kolberg.** Developing and applying bioinformatics tools for gene expression data interpretation. Tartu 2021, 195 p.
28. **Dmytro Fishman.** Developing a data analysis pipeline for automated protein profiling in immunology. Tartu 2021, 155 p.
29. **Ivo Kubjas.** Algebraic Approaches to Problems Arising in Decentralized Systems. Tartu 2021, 120 p.
30. **Hina Anwar.** Towards Greener Software Engineering Using Software Analytics. Tartu 2021, 186 p.
31. **Veronika Plotnikova.** FIN-DM: A Data Mining Process for the Financial Services. Tartu 2021, 197 p.
32. **Manuel Camargo.** Automated Discovery of Business Process Simulation Models From Event Logs: A Hybrid Process Mining and Deep Learning Approach. Tartu 2021, 130 p.
33. **Volodymyr Leno.** Robotic Process Mining: Accelerating the Adoption of Robotic Process Automation. Tartu 2021, 119 p.
34. **Kristjan Krips.** Privacy and Coercion-Resistance in Voting. Tartu 2022, 173 p.
35. **Elizaveta Yankovskaya.** Quality Estimation through Attention. Tartu 2022, 115 p.
36. **Mubashar Iqbal.** Reference Framework for Managing Security Risks Using Blockchain. Tartu 2022, 203 p.
37. **Jakob Mass.** Process Management for Internet of Mobile Things. Tartu 2022, 151 p.
38. **Gamal Elkoumy.** Privacy-Enhancing Technologies for Business Process Mining. Tartu 2022, 135 p.
39. **Lidia Feklistova.** Learners of an Introductory Programming MOOC: Background Variables, Engagement Patterns and Performance. Tartu 2022, 151 p.
40. **Mohamed Ragab.** Bench-Ranking: A Prescriptive Analysis Approach for Large Knowledge Graphs Query Workloads. Tartu 2022, 158 p.
41. **Mohammad Anagreh.** Privacy-Preserving Parallel Computations for Graph Problems. Tartu 2023, 181 p.
42. **Rahul Goel.** Mining Social Well-being Using Mobile Data. Tartu 2023, 104 p.

43. **Anti Ingel.** Algorithms using information theory: classification in brain-computer interfaces and characterising reinforcement-learning agents. Tartu 2023, 142 p.
44. **Shakshi Sharma.** Fighting Misinformation in the Digital Age: A Comprehensive Strategy for Characterizing, Identifying, and Mitigating Misinformation on Online Social Media Platforms. Tartu 2023, 158 p.
45. **Kristiina Rahkema.** Quality Analysis of iOS Applications with Focus on Maintainability and Security Aspects. Tartu 2023, 182 p.
46. **Ivan Slobozhan.** Studying Online Social Media Engagement in CIS Countries during Protests, Mass Demonstrations and War. Tartu 2023, 81 p.
47. **Nurlan Kerimov.** Building a catalogue of molecular quantitative trait loci to interpret complex trait associations. Tartu 2023, 248 p.
48. **Pavlo Tertychnyi.** Machine Learning Methods for Anti-Money Laundering Monitoring. Tartu 2023, 117 p.