Computers & Security 129 (2023) 103206

Contents lists available at ScienceDirect

Computers & Security

journal homepage: www.elsevier.com/locate/cose

Identifying and quantifying trade-offs in multi-stakeholder risk evaluation with applications to the data protection impact assessment of the GDPR

Majid Mollaeefar[a,b,*], Silvio Ranise[a,c]

[a] FBK-Center for Cybersecurity, Trento, Italy
[b] DIBRIS, University of Genova, Italy
[c] Department of Mathematics, University of Trento, Trento, Italy

ARTICLE INFO

Article history:
Received 23 June 2022
Revised 19 November 2022
Accepted 25 March 2023
Available online 30 March 2023

Keywords:
Data protection impact assessment
GDPR
Multi-stakeholder risk assessment
Multi-objective optimization
Pareto-optimality

ABSTRACT

Cybersecurity risk management consists of several steps including the selection of appropriate controls to minimize risks. This is a difficult task that requires to search through all possible subsets of a set of available controls and identify those that minimize the risks of all stakeholders. Since stakeholders may have different perceptions of the risks (especially when considering the impact of threats), conflicting goals may arise that require to find the best possible trade-offs among the various needs. In this work, we propose a quantitative and (semi-)automated approach to solve this problem based on the well-known notion of Pareto optimality. For validation, we show how a prototype tool based on our approach can assist in the Data Protection Impact Assessment mandated by the General Data Protection Regulation on a simplified—but realistic—use case scenario. We also evaluate the scalability of the approach by conducting an experimental evaluation with the prototype with encouraging results.

1. Introduction

Cybersecurity risk management, i.e. the identification, evaluation, and prioritization of risks followed by the application of controls to minimize cyber risks, is a vital aspect of the risk management process of any organization. Several approaches are available to identify, evaluate, and prioritize cybersecurity threats such as the NIST Risk Management Framework[1] that consists of several steps including the selection of controls necessary to protect the system and organization commensurate with risk. This is a non-trivial task as it typically requires to (a) search through a large space of possible configurations for controls mitigating a set of threats according to (b) how the various stakeholders (e.g., the organization providing a service and the users using it) perceive risks. Different attitudes to risk by the various stakeholders may give rise to conflicting goals when considering additional constraints such as costs and skills required to deploy controls; for instance, customers of an online banking service may be interested to eliminate all threats for their financial transactions while the bank is willing to provide protection for the most common vulnerabilities while accepting the risk of more sophisticated attacks to maintain costs at an acceptable level. In this paper, we consider the problem of providing automated assistance to the process of selecting the best possible configurations of controls to mitigate risks for all the stakeholders by making the following three contributions:

(C1) we describe a methodology to semi-automatically assist stakeholders in the definition of their objectives that measure how much risks are reduced by adopting a certain configuration of the controls (this addresses point (b) above and is done by extracting crucial information already elicited during the application of the adopted approach to risk management);
(C2) we define a decidable multi-objective optimization problem (based on the objectives previously identified)—called Multi-Stakeholder Risk Minimization Problem (MSRMP)—whose Pareto optimal solutions (see, e.g., Marler and Arora, 2004) are the subsets of the controls for which no stakeholder's risk can be further reduced without increasing the risk of at least one of the other stakeholders (this is a first step towards addressing point (a) above and is done by exploiting automated state-of-the-art tools for computing the set of solutions);

* Corresponding author.
E-mail addresses: mmollaeefar@fbk.eu (M. Mollaeefar), ranise@fbk.eu (S. Ranise).
[1] https://csrc.nist.gov/projects/risk-management/

https://doi.org/10.1016/j.cose.2023.103206
0167-4048/

(C3) designing and experimentally evaluating heuristics to visit the set of all possible configurations and guarantee the scalability of the proposed technique (this complements the previous contribution to address point (a) by identifying appropriate strategies to partition large search spaces to make the approach viable in practice).

The ability to tackle this kind of problem is particularly relevant when considering privacy provisions deriving from national or international regulations. For instance, the General Data Protection Regulation (GDPR) (Regulation, 2016) requires to conduct a Data Protection Impact Assessment (DPIA) to guarantee the protection of personal data and preserve the rights and freedom of individuals. This means that the organization offering a data processing activity should reduce the risk of the user to an acceptable level while controlling costs and other business goals. In this context, being able to compute the subsets of controls that minimize the risks of both the organization of the system and its users is a necessary pre-requisite to identify the most appropriate configuration of the controls that offer the best possible trade-off among the various objectives.

Authors in Mollaeefar et al. (2020) consider a similar—albeit simpler—optimization problem, allowing for finding the best possible solutions among a (finite and small) set of possible RMPs. Indeed, such solutions are not guaranteed to be Pareto optimal, as those of the MSRMP considered in this paper. Additionally, in Mollaeefar et al. (2020), no methodology to identify the set of possible RMPs is provided whereas this work provides a structured approach to the identification of the whole set of RMPs as solutions of the MSRMP.

Outline

In Section 2, we discuss related works. In Section 3, we introduce the Multi-Stakeholder Risk Minimization Problem (MSRMP) and its formalization as a multi-objective optimization problem (cf. contribution (C2) above) together with an approach to reduce the search space (cf. contribution (C3) above). For concreteness, we propose a running example to illustrate the main ideas underlying the problem (Section 3.1). To find all Pareto optimal solutions and assist stakeholders to identify the risk management policies under which the risk exposure is minimized, we propose an automated technique to solve MSRMP instances (Section 3.2). In Section 4, we discuss a methodology to assist stakeholders in the definition of instances of the MSRMP (cf. contribution (C1) above). In Section 5, we describe a tool supporting the definition of MSRMP instances and the computation of their solutions together with a set of experiments aiming to understand the effectiveness of the strategies to reduce the search space and thus improving the scalability of the proposed approach (cf. contribution (C3) above). We conclude the paper with a summary of the main contributions and some hints for future work (Section 6).

2. Related work

This section discusses the related work. First, we look into several techniques for assessing cybersecurity risks that are widely used in the literature (Section 2.1). Then (Section 2.2), we review some privacy impact assessment techniques together with a selection of the most important methodologies, standards, GDPR guidelines (Section 2.3), and available tools (Section 2.4) for assessing the impact of violations of privacy requirements. Finally, we examine different methods for selecting controls and multi-criteria risk assessment techniques (Section 2.5).

2.1. Cybersecurity risk assessment methodologies

In the scope of information security, a wide range of risk assessment approaches have been proposed by standardization institutes and organizations such as NIST SP 800-30, ISO/IEC 27005, etc. Each of these methodologies has its own specific scope, procedures, and assessment techniques (McLaughlin et al., 2016); despite the differences, they all share the same structure consisting of the following phases: (1) plan, (2) execute, and (3) report on the results (Qassim et al., 2019). The process of preparing for an assessment begins with a thorough inventory of the facility's hardware and software, followed by a review of all applicable regulations, policies, procedures, and controls. In the second phase, the assessment is put into action, which entails looking for potential security flaws and software pitfalls. In the third and final phase, it is documented and coordinated that the reported flaws have been remedied.

The National Institute of Standards and Technology (NIST) (Gary et al., 2002) published a special publication in 2002 that reflects the guideline for the process of organizational risk management, and it was revised for the first time in 2012 (NIST, 2012). Framing, assessing, responding, and monitoring risks have all been included in the risk management processes outlined in this guideline. The first process demonstrates how security researchers are framing or constructing a risk context in order to develop a risk management strategy for further assessing, responding to, and observing risks. Following that, the risk is assessed based on the frame of risk to identify external and internal vulnerabilities, as well as threats to the investigated system, thereby preventing potential harm. Meanwhile, responding to hazards demonstrates how security researchers should respond when a risk is identified during the evaluation. As the last process, monitoring risks relates to how organizations keep track of risk throughout time, notably in terms of verifying the effectiveness of reactions to risks and determining shifts in operating systems that are caused by risk.

ISO/IEC 27005 (JTCIJSS, 2013) is the risk analysis standard for the ISO 27000-series. The ISO/IEC 27000 standards are designed to assist businesses in maintaining the security of their information assets. The standard establishes principles for information security risk management and applies to any type of organization that wishes to manage risks effectively in order to prevent jeopardizing the business's information security. ISO/IEC 27001, the information security management system (ISMS) standard, is the most well-known of the ISO 27000 family. Its purpose was to establish requirements for the execution of information security in accordance with a risk management framework.

Octave Allegro is the next generation of the Operationally Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE) methodology for identifying and evaluating information security risks. The OCTAVE risk assessment process is designed to be as efficient as possible when working with limited resources. Generally speaking, it is best suited for smaller to mid-size businesses. This methodology is primarily concerned with information assets in terms of their use, storage, transportation, and processing, as well as their exposure to threats, vulnerabilities, and disruptions as a result (Bieker et al., 2016).

Factor Analysis of Information Risk (FAIR) (Freund and Jones, 2014) is a pragmatic risk management methodology that identifies and quantifies threats to a business's operational and cybersecurity frameworks. The FAIR model, which is compliant with international standards, was developed in 2005 and is widely regarded as the leading Value at Risk (VaR) framework for operational risk and cybersecurity. The FAIR model discovers and aggregates many aspects that may pose a risk to an organization, and it then thoroughly examines how these factors relate to or trigger another potential concern for the organization.

Regardless of the particular processes each of these security risk assessment approaches above have, they all point out to the risk as an unexpected incident that would damage business assets, either

tangible (e.g., organization's hardware infrastructures) or intangible (e.g., organization's services). The ultimate goal of an information security program based on risk management is to augment the organization's output (product and service) while simultaneously limiting the unexpected adverse outcomes generated by potential risks. However, these risk assessment methodologies are restricted in terms of what risks are related to data subjects and how to evaluate them, which is requested by more and more legal frameworks around the world, especially concerning privacy and other fundamental rights. The GDPR and other applicable regulations mandate a risk-based approach and expressly recommend the execution of Data Protection Impact Assessments (DPIA), which is based on an assessment of the privacy impact against the privacy rights of data subjects. Such assessments are discussed in the following.

2.2. Privacy impact assessment

Privacy Impact Assessment (PIA) is a risk management technique that entails assessing the possible impact of systems on privacy as a result of processing operations on personal data (Clarke, 2009). Organizations should anticipate risks associated with their efforts throughout their life-cycle, beginning with the design phase but also during their operational life-cycle through iterative evaluation. Security by Design (SbD) and Privacy by Design (PbD) are principles increasingly identified as necessary for dealing with design faults that may jeopardize security or privacy in the system (Alshammari and Simpson, 2017). Furthermore, their application is also envisaged by the GDPR, as it demands Data Protection by Design (DPbD) for any systems that involve personal data in their processing. Conducting PIA is mandated by data protection authorities (DPAs) and standardization bodies, who have developed legislative frameworks and guidelines. The General Data Protection Regulation (GDPR) asserts that the data controller must do an impact assessment and document it before starting to process the data. This is done to make European citizens more trustful of digital services (Art. 35). The International Organization for Standardization (ISO) released ISO/IEC 29134:2017 gives guidelines for (i) a process on privacy impact assessments, and (ii) a structure and content of a PIA report (International Organization for Standardization, 2017a). Numerous methodologies and frameworks in the context of privacy impact assessment have been proposed, we overview some methodologies, standards, and GDPR guidelines in the following section, and then discuss tool support for PIAs in Section 2.4.

2.3. Methodologies, standards and GDPR guidelines

Several privacy data protection standards, including BS 10012:2017 (Data protection-specification for a personal information management system, 2017), ISO/IEC 29151:2017 (International Organization for Standardization, 2017b), and ISO/IEC 27018:2014 (International Organization for Standardization, 2014), incorporate the PIA as a required step in performing cyber risk assessments. In the absence of a clear methodology, it is impossible to conduct a PIA in conjunction with a risk assessment procedure. Even though, according to the NIST privacy framework (Boeckl et al., 2020), data protection lies at the convergence of cyber security and privacy, the great majority of organizations treat the PIA separately from the cyber risk assessment (Oetzel and Spiekermann, 2014; Wei et al., 2020). Although ISO/IEC 29134:2017 provides detailed guidelines for conducting a PIA, it only outlines the fundamental concepts of impact analysis and provides insufficient information for the risk assessor (Wei et al., 2020). The literature has documented countless privacy metrics, although these often employ criteria of privacy-enhancing technologies (PETs), such as the quantification of leaked information or the number of indistinguishable users, rather than the impact on privacy (Bisztray and Gruschka, 2019).

Prior to the GDPR, PIA was not a legally required assessment. When data processing is likely to result in a high risk to the rights and freedoms of natural people, controllers are mandated to undergo a Data Protection Impact Assessment (DPIA), as provided by article 35 of the GDPR. On the other hand, the GDPR does not prescribe a specific assessment process (Bieker et al., 2016; De and Le Métayer, 2016; 2017; Meis and Heisel, 2015; Van Dijk et al., 2016; van Puijenbroek and Hoepman, 2017) while at the same time mandating a clear understanding of the Personal identifiable information (PII), because any improper management of PIIs may be considered a violation of the GDPR.

The French National Institute of Data Protection (CNIL CNIL (Commission Nationale de l'Informatique et des Libertés) (2018)), the British Information Commissioner's Office (ICO Information Commission's Office (ICO) (2018)), and the Canadian Privacy Act (T. B. of Canada Secretariat, 2010) are just a few of the national authorities who have issued guidance for DPIA. Such instructions have been updated to better serve DPIAs and to provide thorough guidelines on the regulatory standards and processes that they must follow. These guidelines include a variety of techniques and provide a variety of processes for performing a PIA, but they are abstract or vague, making it extremely difficult to conduct such methodologies (Ahmadian et al., 2018). As a result, organizations have difficulty adopting a single methodology, which leads to a lack of support for PIAs (Vemou and Karyda, 2018). To demonstrate the lack of completeness among the most well-known DPIA approaches, a more recent study examined seventeen questions culled from the literature (Vemou and Karyda, 2018).

Along with the regulatory steps described above, academics have recommended improvements to the DPIA processes, which are currently under consideration. To this end, formal modeling techniques for privacy threats are being used to make the DPIA process more systematic and structured. For instance, LINDDUN (Wuyts and Joosen, 2015) is a threat modeling framework to identify privacy threats, and it comprises of six main processes that can assist analysts in systematically eliciting and mitigating privacy threats. It is an acronym for *Linkability, Identifiability, Non-repudiation, Unawareness, Detectability, Disclosure of information,* and *Non-compliance*. Both LINDDUN and the CNIL methodology are based on the same principles. In comparison to the CNIL, however, LINDDUN includes the capability to visualize data flow diagrams and privacy threat tree patterns. From a legal perspective, however, LINDDUN lacks assessment steps and is not integrated into a risk assessment process (Papamartzivanos et al., 2021).

The Standard Data Protection Model (für Datenschutz, 2020) (SDM) provides appropriate measures to transform the regulatory requirements of the GDPR to qualified technical and organizational measures. For this purpose, the SDM first records the legal requirements of the GDPR and then assigns them to the protection goals *Data Minimization, Availability, Integrity, Confidentiality, Transparency, Unlinkability* and *Intervenability*. The SDM thus transposes the legal requirements of the GDPR on protection goals into the technical and organizational measures required by the Regulation, which are described in detail in the SDM's catalog of reference measures. However, this methodology can not be used alone for conducting a risk assessment; indeed, it is a valuable supplement to performing a DPIA where it can help data controllers to specify which GDPR requirements may be at risk in a system.

2.4. PIA tools

Existing PIA tools can be broadly classified as products of the following standardization efforts and their resulting schemes:

ENISA Tool (Evaluating the level of risk for a personal data processing operation, 2020), GS1 EPC/RFID PIA Tool Gs1 (2015), CNIL tool (CNIL (Commission Nationale de l'Informatique et des Libertés), 2020), SPIA Tool (Introduction to the spia program, 2016), and ASPIA Tool (Papamartzivanos et al., 2021). The majority of tools on the market have a limited application scope, with typically a single use case. Existing solutions facilitate the documentation of data processing procedures, the formation of consent templates, and the documentation of privacy and data protection policies in significant numbers. Nonetheless, the cybersecurity posture of the organization performing the impact analysis is largely disregarded (Papamartzivanos et al., 2021).

ENISA Tool. ENISA has provided an online tool for assessing the amount of risk associated with the processing of personal data (Evaluating the level of risk for a personal data processing operation, 2020). This tool is intended to provide direction to small and medium-sized enterprises (SMEs) and help data controllers and processors. The adopted approach includes six steps that give a streamlined approach, steer SMEs toward a data processing operation, and enable them to assess privacy-related security risks. The assessor establishes the context of the processing operation by following the processes that have been suggested, and then manually analyzes how the fundamental rights and freedoms of individuals may be compromised as a result of the potential breach of the security of thepersonal data. Four levels of impact are supported, ranging from Low to Very High. Furthermore, the assessor manually documents both external and internal threats to the system and assesses the likelihood of their occurrence. The final risk assessment is provided following an analysis of the personal data processing operation's impact and the associated threat probability. The tool facilitates the process of adopting new security and privacy measures based on the outcome.

GS1 EPC/RFID PIA Tool. The tool (Gs1, 2015) aids in the assessment of privacy issues associated with RFID implementations and assists in the selection of privacy safeguards to be addressed during application development. The tool is an MS Excel file that facilitates in the calculation of risk level scores using the formula $Risk = Impact \times Likelihood - Controls$. To evaluate the residual risk, the score takes control efficacy into account. The assessor answers specific questions/considerations during the procedure and can establish arbitrary controls and their effectiveness on a scale of 1 to 5. When it comes to privacy issues that can be triggered due to actual attack vectors targeting the deployment, the tool does not focus on identifying technical aspects of the implementations. Furthermore, the score criteria are fairly vague and unspecific for privacy threats (Agarwal, 2015), and the assessment is limited to the technology sector of EPC/RFID applications.

CNIL Tool. The CNIL tool (CNIL (Commission Nationale de l'Informatique et des Libertés), 2020) is designed to help data controllers perform DPIAs using the methodology released by CNIL in CNIL (Commission Nationale de l'Informatique et des Libertés) (2012, 2018). According to CNIL's methodology, a PIA is based on two main aspects: (i) fundamental rights and principles, which are "non-negotiable", mandated by law and which must be respected, regardless of the risk nature, and (ii) management of data subjects'sprivacy risks, which determines the appropriate technical and organizational controls to protect personal data. The following steps must be followed by PIA practitioners:

- Define and document the context of the data processing action under consideration
- Analysis of controls that can protect fundamental principles
- Assessment and management of privacy risks related to data
- Formal documentation and validation of the PIA

The PIA tool assists practitioners in carrying out the actions that were indicated earlier in this paragraph. The evaluation outcome is depicted as a heat map, in which the risks are arranged in a manner that considers both their criticality and likelihood. The CNIL tool supports four levels of severity scales; Negligible, Limited, Significant, Maximum.

SPIA Tool. The Security and Privacy Impact Assessment (SPIA) is a tool developed by the University of Pennsylvania (Introduction to the spia program, 2016) intended to assist organizations in conducting PIA by identifying risk-prone regions and choosing the most appropriate tactics and timetables for risk reduction. This tool concentrates on safeguards while focusing on both security and privacy for the protection of data. The tool has two versions; the first version is an MS Excel file, whereas the second version (SPIA 2.0) is a web-based application. The tool allows organizations to take probability rankings and threat consequences and automatically score risk into categories of High, Significant, Moderate and Low. Additionally, the SPIA Tool is a flexible and adaptable tool that supports various security and privacy threats.

APSIA Tool. The Automated Privacy and Security Impact Assessment (APSIA) is powered by the use of interdependency graph models and data processing flows used to create a digital reflection of the cyber-physical environment of an organization. The methodology presents an extensible privacy risk scoring system for quantifying the privacy impact triggered by the identified vulnerabilities of the ICT infrastructure of an organization. Indeed, APSIA seeks to bridge the gap between the cyber-risk and privacy risk assessments, which are typically handled as separate management processes. In APSIA, the impact level is defined as a combination of three components, namely, (a) the level of impact on the fundamental rights and freedoms of the individuals, (b) the scope of impact to the data processing activities, and (c) the type (i.e., sensitivity) of the processed data. However, in APSIA, the selection of optimal mitigation controls is not considered. Their aim is only to support the decision-makers in making informed decisions during the risk mitigation life cycle.

2.5. Multi-Criteria risk assessment & control selection methodologies

In practice, cost and time constraints, feasibility, and other organizational considerations make it impractical to implement all mitigations (also known as security controls) for every threat. Optimizing mitigation selections has been approached by several researchers, who have taken an extensive list of possible mitigations and narrowed it down to just a few that meet specific criteria or goals (Llansó et al., 2019). The criteria themselves and the analysis methods are the most intriguing dimensions in this area. Based on a grounded theoretical research, authors in Dor and Elovici (2016) present a model of information security investment decision-making. Numerous factors, including policy, competitive advantage, financial considerations, quality, compliance, customer expectations, and strategy, have a profound impact on the manner in which organizations take these decisions. Selection of a security control portfolio for a given circumstance requires taking into account several factors, such as an organization's overarching security concerns, the criteria of individual assets in the environment, potential threats, and the quality of controls. Authors in Llansó et al. (2019), for example, offer a review of the literature that leads to the identification of four criteria: organizational, asset, threat, and control. For instance, in Rees et al. (2011), the authors developed a decision support system for assessing the unknown risk an organization faces during a cyber attack as a function of uncertain threat rates, countermeasure costs, and impacts on its assets. The system employs a genetic algorithm to search for the optimal combination of countermeasures, allowing the user to determine the preferred tradeoff between the portfolio cost and the resulting risk. In Gupta et al. (2006), a Genetic Algorithm (GA)-based method was developed that enables enterprises to select the

lowest-cost security profile with the greatest vulnerability coverage. Authors, in Kavallieratos et al. (2021) developed a technique that permits the best selection of cybersecurity controls for complex cyber-physical systems (CPSs) that contain other CPSs as components. The technique estimates the overall risk by considering the likelihood and impact values for each of the system's components and analyzing how risk propagates across information and control flow components. Then to discover the optimal set of controls for each component, the approach applies a genetic algorithm workflow.

Multi-criteria decision-making (MCDM) (Figueira et al., 2005), commonly known as multiple-criteria decision analysis (MCDA), is widely applied in the selection of security portfolios. MCDM is a method for analyzing multiple conflicting criteria and is used to examine problems in which there are several measurements of costs and benefits that may be traded off to arrive at the optimal solution within the limits that have been specified. Fuzzy set theory (Otero, 2014), multi-attribute utility theory (e.g., value functions, knapsack strategy) (Fielder et al., 2016; Panaousis et al., 2014; Shahpasand et al., 2015; Smeraldi and Malacaria, 2014), and evolutionary multi-objective optimization (EMO), commonly known as genetic algorithms (Kiesling et al., 2016), are some of the MCDM methodologies being investigated by researchers to address this problem.

There are a lot of risk assessment approaches which consider multi-criteria to calculate risk exposure. In Zulueta et al. (2013) risk analysis is modeled as MCDM problem in which experts express their preferences for each risk, over two traditional criteria: probability and impact. A risk-based decision framework (Ganin et al., 2017) is proposed for cybersecurity strategy prioritization. There are a few approaches that have defined risk impact criteria for different stakeholders. For instance, in the context of cloud computing, in Albakri et al. (2014) a security risk assessment framework is proposed that can enable cloud service providers to assess security risks in the cloud computing environment and allow cloud clients with different risk perspectives to contribute to risk assessment. In analyzing the conflict of interest between the risk owner and the risk actors in Rajbhandari and Snekkenes (2012) authors proposed conflicting incentives risk analysis (CIRA) method in which risks are modelled in terms of conflicting incentives. The goal of CIRA is to provide an approach in which the input parameters can be audited more easily. In Wright (2012), the authors provide a seven-step approach to PIA. They have declared that privacy risk shall be assessed from both data subjects and system perspective. The authors recommend privacy controls that can help to minimize, mitigate, or eliminate the identified risk. Similarly, recently, the authors (Iwaya et al., 2019) proposed a privacy risk assessment by considering both perspectives. Their approach is based on the PIA methodology proposed by Wright (2012) in the case of mobile health data collection systems, which proposes a systematic identification and evaluation of privacy risks.

3. Multi-Stakeholder risk minimization problem (MSRMP)

Cyber-risk is a measure of the likelihood and the impact of threats, i.e. circumstances or events with the potential to harm a cyber-system such as the unauthorized disclosure, destruction, modification, or interruption of system assets. Cyber-risk management is the *identification* and *assessment* of risks, followed by the *definition* and *enforcement* of appropriate *mitigation measures* for risk minimization. The identification of risks depends on the assets of the system to be protected and requires to perform threat modeling, i.e. to understand and describe how an adversary might compromise a system. The assessment of risks amounts to evaluating the impact and the likelihood of the various threats. For instance, a backdoor in a certain version of an operating system may have a dramatic impact. The risk may be severe if patches are applied late as the likelihood that an adversary exploits the vulnerability is high whereas the risk becomes small when patches are quickly applied as the time-window during which an attacker can exploit the vulnerability is substantially reduced. The balance between impact and likelihood is key to risk assessment. Once risks have been identified and assessed, suitable Risk Management Policies (RMPs) should be defined and enforced. RMPs comprise both technical (e.g., deploy the latest version of the Transport Layer Security protocol) and organizational (e.g., a cyber security awareness and training program for employees) measures to minimize risks. Indeed, the ultimate goal of risk management is to minimize risks while maximizing the chances to reach business objectives and complying with legal provisions, such as the GDPR. Indeed, failing to do this may bring in additional risks and costs due to an unsatisfactory return on investment or fines for lack of compliance.

Given the increasing complexity of cyber-systems, it is routine that several stakeholders cooperate in their design, development, and deployment. This further complicates risk management. For instance, according to the GDPR, in case a system processes personal data, its data controller shall guarantee that the risk of violating the rights and freedom of the data subjects is low. The data controller must do this by considering state-of-the-art RMPs and budget constraints. When the data controller involves a data processor, the latter may have strict computational constraints for scalability and efficiency that, in turn, guarantee economy of scale. While the various stakeholders may agree on a common set of threats for a given system together with their likelihood, they will have diverging criteria to evaluate the potential impact of the identified threats. For instance, data subjects will favor comprehensive RMPs to reduce the risk of data breaches. In contrast, a data controller or a data processor may be more interested in cheap and easy to enforce RMPs that cover most threats while neglecting those less likely to occur. Besides making the definition of the impact of threats dependent on each stakeholder, this greatly complicates the search for RMPs that minimize risks. Indeed, the search for RMPs that simultaneously minimize the risk level for each stakeholder becomes a non-trivial task in the presence of conflicting objectives and requires the adoption of the notion of Pareto optimality. To understand the problem, consider the situation in which we have two RMPs *rpm1* and *rpm2* with risk vectors $\langle 1, 2, 1\rangle$ and $\langle 1, 1, 2\rangle$, respectively, where the first component is the risk of the data subject, the second is that of the data controller, and the third is that of the data processor. The data subject has no preference between the two RMPs, the data controller prefers *rpm1* over *rpm2*, and the data processor *rpm2* over *rpm1*. In other words, no RMP minimizes the risk for all the stakeholders; so, which one between *rpm1* over *rpm2* should be preferred? According to the notion of Pareto optimality (see, e.g., Marler and Arora, 2004), both *rpm1* and *rpm2* are to be considered optimal and further aspects need to be considered to select one of the two such as the fact that one of the two promises to provide a higher return on investment or that it is easier to show its compliance with the GDPR or other legal provisions. Because vectors cannot be ordered completely, all the Pareto optimal solutions can be regarded as equally desirable in the mathematical sense and we need a decision maker to select the preferred one among them. To enable the decision maker to do this, we need to be able to compute the set of Pareto optimal solutions. Below (Section 3.2), we formalize the problem of finding Pareto optimal configurations of RMPs, i.e., configurations minimizing the risk of stakeholders, in the framework of multi objective optimization and show how it can be solved by using general purpose algorithms under reasonable assumptions. Primarily, we introduce a simplified but realistic running example to better grasp the problem.

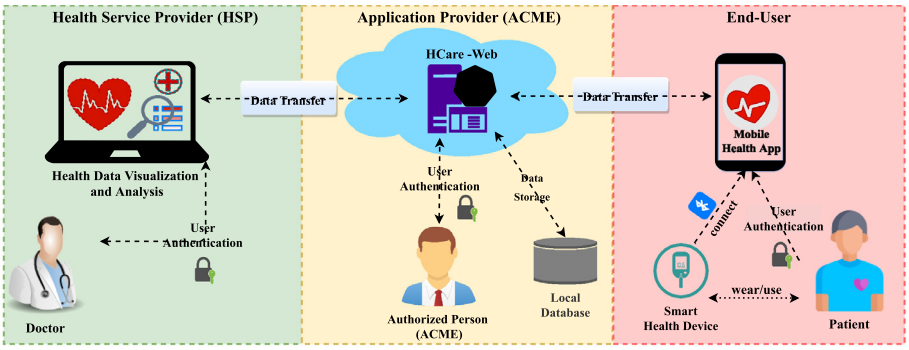

Fig. 1. Overview on the main stakeholders in the scenario and their interaction with the system's components.

3.1. Running example: An application of the GDPR's DPIA

We consider the situation of an Italian company, called ACME below for the sake of anonymity, that must perform a Data Protection Impact Assessment (DPIA) for one of its software applications, as required by Article 35 of the General Data Protection Regulation [2] (GDPR). The goal of a DPIA is to protect the rights and freedoms of EU citizens with particular relevance to those related to their privacy. For this, it is crucial to perform an appropriate privacy risk assessment. There are three main stakeholders involved in the process, namely (i) the *Data Subject*, the patient whose data are being collected, stored and processed by the application, (ii) the *Data controller*, ACME which is responsible for offering the data processing activities implemented by the software application, and (iii) the *Data processor*, a company mandated by the Data Controller to design and implement the application deploying the various data processing activities. The data processor is a third party organization, possibly external to the data controller. In the rest of this section, we focus on the problem of identifying appropriate security controls among a set of available ones that minimize the risks of all three stakeholders. A peculiarity of this risk assessment is that the data controller must perform it to make the risk of the data subjects acceptable. Indeed, this may give rise to conflicts with the data controller's and data processor's requirements on budgets and skills shortage.

ACME develops a software application, called HCare, exposing an API service to allow its clients to work together, as illustrated in Fig. 1. Through the API, HCare connects three main stakeholders: the Health Service Provider (HSP), the API provider (ACME), and the patients which are the data controller, the data processor, and the data subjects in the context of the GDPR, respectively. Notice that an HSP in our case can also be an independent developer who provides IT-only services without offering actual health care support; for example, providing data visualization tools. Finally, the end-user is typically the patient using the app to send biometric data or user-initiated requests and receive responses from the HSP, e.g., prescriptions from a doctor, medical alerts, etc. HSPs use the APIs to perform some operations such as create, read, update, and delete (CRUD operations) in a compliant way—i.e., by considering proper roles and permissions and storing and accessing the data accordingly. The health data is stored in a cloud environment, controlled, and monitored by ACME. Consequently, from a legal perspective, ACME acts as the data processor. However, due to the nature of its offered services, ACME has also to support data controllers to comply suitably. Therefore, it looks at the issue of GDPR compliance from both perspectives, of the data processor and data controllers. This is handled by a service level agreement between ACME and the HSP.

ACME, as data controller, must be aware of how to properly process the patients' data because there could be a variety of harmful or threat events that could put even the patients' life at risk. For instance, data (such as the patient's medical history) could be lost or corrupted due to a hardware failure. Patients may suffer severe consequences as a result of this situation because the healthcare data in question is used to offer healthcare services such as medical prescriptions, and missing or damaged data may result in incorrect diagnoses or the inability to provide the service. For this reason, data storage must be trustworthy, which can be achieved by implementing appropriate data protection controls. For instance, more frequent backups or data replication are potential controls to mitigate possible risks in the case of a hardware failure. However, these solutions change the risk exposure of ACME. Data replication, particularly, introduces the needs and all the associated risks of a sophisticated network architecture. For example, business risks due to the rising costs, but also process risks due to the difficulty of network configuration. This example demonstrates the consequences of the law: given that the data subject has certain fundamental rights, it is the data controller's responsibility to put in place the appropriate technical and organizational means to ensure that the rights of the data subject are respected. The endeavor to reduce the risks for the data subject, on the other hand, may result in an increase in the risk exposure for ACME, which may include risks other than those related to personal data. From these considerations we can see that it is likely that each stakeholder has different preferences for the various RMPs yielding different threat impact levels for each threat. Therefore, we must solve the problem of selecting the optimal risk management policy, which we formalize in the framework of multi-objective optimization in the next section.

To summarize, for the running example, we consider a set $\mathcal{S}$ containing two stakeholders, namely the Data Controller and the Data Subject, a list of 5 threats $T_1, \ldots, T_5$ shown in Table 2 and a list of associated security controls $c_1, \ldots, c_{25}$ shown in (the first two columns of) Table 3. Thus, we have 5 threats and 25 controls; the latter are associated to each threat as follows: $c_1, \ldots, c_5$ to T_1, $c_6, \ldots, c_{15}$ to T_2, $c_{16}, \ldots, c_{19}$ to T_3, $c_{20}, \ldots, c_{22}$ to T_4, and $c_{23}, \ldots, c_{25}$ to T_5. In the next section, we use the running example to illustrate the formal notions we introduce, albeit in a simplified form for the sake of simplicity and space. So, for instance, we will consider only 3 threats instead of 5 and only 5 security controls instead of 25. We observe that we use $c_1, \ldots, c_5$ as identifiers of the security controls in the following section for the sake of simplicity, but they have been renamed in Table 3 where the whole set of controls is listed. The solution of the multi-objective optimization problem in its full generality is discussed later in Section 5.1. A summary of all the variables used in this paper along with a brief description is provided in Table 1.

[2] https://gdpr-info.eu/art-35-gdpr/

Table 1
List of used variables in this paper.

Variable	Description
$\mathcal{T}$	The set of threats
$\mathcal{S}$	The set of stakeholders
$\mathcal{C}$	The set of controls
$\mathcal{P}$	The protection criteria
$\mathcal{G}$	The set of protection goals
$\mathcal{C}_T$	A family of controls associated to $T \in \mathcal{T}$
x_T	The residual risk for threat T
i_s	The impact level for each stakeholder $s \in \mathcal{S}$
$\mu_T(c)$	The impact of T after applying control $c \in \mathcal{C}_T$
PW_p^s	The associated weight to the preference p of stakeholder s
$il_{\max}$	Donates to the maximum impact level
$al_p^s(T)$	The aversion level of threat T for the preference p of stakeholder s
OW_T	The observation weight for threat T
NTC_T	The normalized threat criticality value for threat T
AG_T	The number of goals in $\mathcal{G}$ affected by a threat $T \in \mathcal{T}$
$oir(s)$	The overall impact residue for stakeholder s

3.2. Problem formalization

Let $\mathcal{S}$ be a finite set of stakeholders and $\mathcal{T}$ a finite set of threats. For each stakeholder s in $\mathcal{S}$, we assume a mapping $i_s : \mathcal{T} \to \mathcal{I}$ that computes the impact level of the harmful events generated by a threat T when it occurs, where $\mathcal{I}$ is a sub-set of the reals denoting impact levels, intuitively $il_1 < il_2$ implies that the impact level il_1 is less severe than the impact level il_2.

Example 1. Referring to the example in Section 3.1, the set $\mathcal{S}$ of stakeholders contains $s_1 =$ *Data controller (ACME)* and $s_2 =$ *Data subject (the patient)*. Consider the set $\mathcal{T}$ of threats to contain $T_1 =$ *Unlimited data storage*, $T_2 =$ *Unauthorized access*, and $T_3 =$ *Linkage attack*, as three potential threats. We may define the mappings i_{s_1} and $i_{s_2} : \mathcal{T} \to \mathcal{I}$ by means of a table as follows:

	T_1	T_2	T_3
i_{s_1}	0.6	0.2	0.3
i_{s_2}	0.3	0.5	0.6

The values in the first and second rows of the table denote the impact levels for each threat from the point of view of the data controller (s_1) and data subject (s_2), respectively. For instance, the impact level associated to threat T_1 from the data controller point of view is 0.6 whereas from the data subject point of view is 0.3.

As shown in the example above, i_s is typically specified by using a tabular format. This is also the case for other mappings that we consider below.

Let $\mathcal{C}$ be a finite set of controls and $\{\mathcal{C}_T\}_{T\in\mathcal{T}}$ a family of finite set of controls; intuitively, $\mathcal{C}_T$ is the set of controls that, alone or in combination, may mitigate a threat T.

Example 2. To mitigate the risk of threats in Example 1, we identify a family of set of controls $\{\mathcal{C}_{T_1}, \mathcal{C}_{T_2}, \mathcal{C}_{T_3}\}$ where $\mathcal{C}_{T_1} = \{c_1, c_2\}$, $\mathcal{C}_{T_2} = \{c_3, c_4\}$, and $\mathcal{C}_{T_3} = \{c_5\}$. For instance, c_1 can be (Ensuring data minimization), c_2 (Enabling data deletion), c_3 (Ensuring secure storage), c_4 (Logging access to personal data), and c_5 (Ensuring data anonymization).

For each threat T in $\mathcal{T}$, we assume a mapping $\mu_T : \mathcal{C}_T \to [0.1)$ that quantifies the mitigation by a control in $\mathcal{C}_T$ on the impact of a threat T. Intuitively, $\mu_T(c)$ can have three possible statuses: (i) $\mu_T(c) = 0$ clarifies that the control c is not adopted and thus can not contribute in mitigating threat T, (ii) $0 < \mu_T(c) < 1$ means that the control c is adopted and partially mitigates the threat T, and (iii) $\mu_T(c) = 1$ represents that the control is adopted and fully mitigates T.

We are now in the position to define the impact residue of the threat T under a given mitigation mapping μ_T as:

$$ir_s(T) = i_s(T) \cdot (1 - \frac{\Sigma_{c\in\mathcal{C}_T}\mu_T(c)}{|\mathcal{C}_T|}). \quad (1)$$

We observe that the expression between parentheses is the mitigation obtained by adopting some of the controls in $\mathcal{C}_T$ associated to T and that the degree of effectiveness of a control c in mitigating a threat T is given by $\mu_T(c)$. Because of its importance, we introduce the following abbreviation:

$$m(T) = \frac{\Sigma_{c\in\mathcal{C}_T}\mu_T(c)}{|\mathcal{C}_T|} \quad (2)$$

that depends on the mitigation mapping μ_T (and since $ir_s(T) = i_s(T) \cdot (1 - m(T))$ also $ir_s(T)$ depends on μ_T) but we avoid to make such a dependence explicit to simplify notation. Given a family $\{\mu_T\}_{T\in\mathcal{T}}$ of mitigation mappings, the overall impact residue for a given stakeholder $s \in \mathcal{S}$ is defined as $oir(s) = \Sigma_{T\in\mathcal{T}} ir_s(T)$, where $ir_s(T)$ is evaluated under the mitigation mapping μ_T. In other words, $oir(s)$ is the sum, over the set $\mathcal{T}$ of threats, of all impact residues, each one evaluated under the associated mitigation mapping in $\{\mu_T\}_{T\in\mathcal{T}}$.

Example 3. For simplicity, we consider three possible values in the co-domain of μ_{T_1}, μ_{T_2}, and μ_{T_3}, namely 0 (the control does not mitigate the threat), 0.5 (the control partially mitigates the threat), and 1 (the control eliminates the threat). Continuing the previous examples, the mitigation mappings for T_1, T_2, and T_3 can be defined as follows:

$\langle\mu_{T_1}(c_1), \mu_{T_1}(c_2)\rangle$	$m(T_1)$	$\langle\mu_{T_2}(c_3), \mu_{T_2}(c_4)\rangle$	$m(T_2)$	$\langle\mu_{T_3}(c_5)\rangle$	$m(T_3)$
$\langle 0, 0\rangle$	0	$\langle 0, 0\rangle$	0	$\langle 0\rangle$	0
$\langle 0, 0.5\rangle$	0.25	$\langle 0, 0.5\rangle$	0.25	$\langle 0.5\rangle$	0.5
$\langle 0.5, 0\rangle$	0.25	$\langle 0.5, 0\rangle$	0.25		
$\langle 0.5, 0.5\rangle$	0.5	$\langle 0.5, 0.5\rangle$	0.5		
$\langle 1, 0\rangle$	0.5	$\langle 1, 0\rangle$	0.5		
$\langle 0, 1\rangle$	0.5	$\langle 0, 1\rangle$	0.5		
$\langle 1, 0.5\rangle$	0.75	$\langle 1, 0.5\rangle$	0.75		
$\langle 0.5, 1\rangle$	0.75	$\langle 0.5, 1\rangle$	0.75		

where the first column of each table lists all possible mitigation vectors that are assigned to the controls of $\mathcal{C}_{T_1}$, $\mathcal{C}_{T_2}$, and $\mathcal{C}_{T_3}$, respectively, when considering an arbitrary total order on the controls (in our case c_i comes before c_j if $i < j$ for $i, j \in \{1, \ldots, 5\}$. For instance, the vector $\langle 0.5, 0\rangle$ means that c_1 partially mitigates T_1 whereas c_2 has no mitigation effect on T_1.

Table 2
An example of possible threat scenarios and associated malicious activities in the ACME scenario.

Threats ($\mathcal{T}$)	Possible malicious activity
T_1- Unlimited data storage	Personal data is kept stored longer than necessary for the purposes by ACME.
T_2- Unauthorized access and disclosure	Due to over-privileged or inadequate controls, insiders (i.e., a medical practitioner or an ACME's staff) modify patients' data or disclose by mistake.
T_3- Linkage attack	Patients and their personal data can re-identify in de-identified data sets by outsiders' malicious.
T_4- Denial of service	Attackers can disrupt the communication channel between patients and the healthcare service provider to prevent data from being uploaded to the server.
T_5- Threat to intervenability	ACME does not implement a procedure (technical and /or processes) that allows the patients to rectify, erase, or block individual data.

Table 3
Threats with associated security controls (first two columns) together with a mitigation mapping (third column) and the resulting risk residue (fourth column). Legend: each control is associated to a mitigation level among three possible values ○=0 (the control has not been selected for implementation), ◐=0. (the control has been selected for implementation but it is only partially effective to mitigate T), or ●=1 (the control has been selected for implementation and it is fully effective to mitigate T).

Threats ($\mathcal{T}$)	Controls $\{\mathcal{C}_T\}_{T\in\{T_1,T_2,T_3,T_4,T_5\}}$	Mitigation Mapping μ_T	Risk residue x_T
T_1	c_1) Purpose specification	●	0.4
	c_2) Ensuring limited data processing	●	
	c_3) Ensuring purpose related processing	◐	
	c_4) Ensuring data minimization	◐	
	c_5) Enabling data deletion	○	
T_2	c_6) Ensuring data subject authentication	●	0.35
	c_7) Ensuring staff authentication	●	
	c_8) Ensuring device authentication	◐	
	c_9) Logging access to personal data	◐	
	c_{10}) Performing regular privacy audits	○	
	c_{11}) Ensuring data anonymization	◐	
	c_{12}) Providing confidential communication	●	
	c_{13}) Providing usable access control	◐	
	c_{14}) Ensuring secure storage	●	
	c_{15}) Ensuring physical security	◐	
T_3	c_{16}) Providing confidential communication	●	0.25
	c_{17}) Logging access to personal data	◐	
	c_{18}) Ensuring data subject authentication	●	
	c_{19}) Ensuring data anonymization	◐	
T_4	c_{20}) Enabling offline authentication	○	0.83
	c_{21}) Network monitoring	◐	
	c_{22}) Prevention mechanisms for DoS attacks like firewalls, etc.	○	
T_5	c_{23}) Informing data subjects about data processing	◐	0.66
	c_{24}) Handling data subject's change requests	◐	
	c_{25}) Providing data export functionality	○	

Example 4. From the definitions of $i_s(T)$ and $m(T)$ in Examples 1 and 3, respectively, we can compute the impact residue $ir_s(T) = i_s(T)\cdot(1-m(T))$ for each mitigation vector in Example 3 as follows:

T	$ir_{s_1}(T)$	$ir_{s_2}(T)$
T_1	$0.6\times(1-0)=0.6$	$0.3\times(1-0)=0.3$
	$0.6\times(1-0.25)=0.45$	$0.3\times(1-0.25)=0.225$
	$0.6\times(1-0.25)=0.45$	$0.3\times(1-0.25)=0.225$
	$0.6\times(1-0.5)=0.3$	$0.3\times(1-0.5)=0.15$
	$0.6\times(1-0.5)=0.3$	$0.3\times(1-0.5)=0.15$
	$0.6\times(1-0.5)=0.3$	$0.3\times(1-0.5)=0.15$
	$0.6\times(1-0.75)=0.15$	$0.3\times(1-0.75)=0.06$
	$0.6\times(1-0.75)=0.15$	$0.3\times(1-0.75)=0.06$
T_2	$0.2\times(1-0)=0.2$	$0.5\times(1-0)=0.5$
	$0.2\times(1-0.25)=0.15$	$0.5\times(1-0.25)=0.375$
	$0.2\times(1-0.25)=0.15$	$0.5\times(1-0.25)=0.375$
	$0.2\times(1-0.5)=0.1$	$0.5\times(1-0.5)=0.25$
	$0.2\times(1-0.5)=0.1$	$0.5\times(1-0.5)=0.25$
	$0.2\times(1-0.5)=0.1$	$0.5\times(1-0.5)=0.25$
	$0.2\times(1-0.75)=0.05$	$0.5\times(1-0.75)=0.125$
	$0.2\times(1-0.75)=0.05$	$0.5\times(1-0.75)=0.125$
T_3	$0.3\times(1-0)=0.3$	$0.6\times(1-0)=0.6$
	$0.3\times(1-0.5)=0.15$	$0.6\times(1-0.5)=0.3$

where the second and third columns represent the computed impact residues under all possible mitigation mappings and the corresponding threat (in the rows) for s_1 and s_2, respectively. For instance, the impact residue under the mitigation vector $\langle 0.5, 1\rangle$ for T_1 from the point of view of s_1 is 0.15 whereas it is 0.06 for s_2. Recalling that $oir(s) = \Sigma_{T\in\mathcal{T}} ir_s(T)$, the overall impact residue for s_1 is $oir(s_1) = 0.6+0.2+0.3 = 1.1$ and for s_2 is $oir(s_2) = 0.3+0.5+0.6 = 1.4$ where $\langle \mu_{T_1}(c_1), \mu_{T1}(c_2)\rangle = \langle 0,0\rangle$, $\langle \mu_{T_2}(c_3), \mu_{T_2}(c_4)\rangle = \langle 0,0\rangle$, and $\langle \mu_{T_3}(c_5)\rangle = \langle 0\rangle$.

The *Multi-Stakeholder Risk Minimization Problem* (MSRMP) amounts to solve the following multi-objective optimization problem:

$$min_{\langle \mu_T\rangle_{T\in\mathcal{T}}} \quad \langle oir(s)\rangle_{s\in\mathcal{S}} \tag{3}$$

where $\langle\ \rangle_{T\in\mathcal{T}}$ and $\langle\ \rangle_{s\in\mathcal{S}}$ are the vectors of all mitigation mappings and overall impact residues (under the associated mitigation mappings) according to arbitrary total orders over $\mathcal{T}$ and $\mathcal{S}$, respectively. In other words, the MSRMP consists of finding the vector of mitigation mappings that allows for minimizing the overall impact residues of the stakeholders. A solution of (3) is a vector $\langle \mu_T\rangle_{T\in\mathcal{T}}$ of mitigation mappings that is Pareto optimal (see, e.g., Marler and Arora, 2004), i.e. it is such that if there does not exist another vector $\langle \mu'_T\rangle_{T\in\mathcal{T}}$ of mitigation mappings such that $oir(s) \leq oir'(\bar{s})$ for each $s\in\mathcal{S}$ and $oir'(\bar{s}) < oir(s)$ for at least one $\bar{s}\in\mathcal{S}$ where oir and oir' are the overall impact residues under the family $\{\mu_T\}_{T\in\mathcal{T}}$ and $\{\mu'_T\}_{T\in\mathcal{T}}$ of mitigation mappings, respectively.

We make two observations. First, (3) considers only the impact and not the likelihood since, as already discussed earlier, we assume that the stakeholders in $\mathcal{S}$ agree on both the set $\mathcal{T}$ of threats and their likelihood. As a consequence, minimizing the impact is equivalent to minimizing the risk since the latter is the product of impact and likelihood, and it is a constant and positive value for each stakeholder in $\mathcal{S}$. This is a natural assumption to make in the context of the GDPR whereby the data controller is accountable for the risk assessment and needs to guarantee that the risks of the data subject are kept to a minimum. The second observation is about solving (3). Indeed, it is possible to re-use the cornucopia of techniques available for Multi Objective Optimization Problem (MOOP); see, e.g., Marler and Arora (2004). However, for some of the techniques to be applicable, it is crucial to have a definition of the functions i_s and μ_T for $T\in\mathcal{T}$ in closed form. This is rarely the case for the use case scenarios we have in mind. Instead, experts are typically able to define both i_s and μ_T as discrete functions, i.e. by associating a given impact level with a certain threat for i_s and quantifying the amplitude of the mitigation associated to a given control in $\mathcal{C}_T$ for μ_T. The examples above present this kind of definitions for such functions by using tables.

As a consequence of the two observations above, we make the following assumptions. First, each stakeholder s in $\mathcal{S}$ provides a definition of the mapping i_s as a finite set of pairs of the form (T, il) where T is a threat in $\mathcal{T}$ and il is an impact level in a finite

set $\mathcal{I}$ of values (i.e., $\mathcal{I} = \{0, 1, 2, 3, 4\}$ where 0 denotes a negligible impact, 4 a dramatic impact, and the values in between increasing values). Second, for each threat T in $\mathcal{T}$, the stakeholder in charge of the risk management process (i.e., the data controller in the case of the GDPR) defines the mapping $\mu_T : \mathcal{C}_T \to \mathcal{A}$ with $\mathcal{A}$ a finite set of values in the interval [0.1]; in other words, μ_T is specified as a finite set of pairs of the form (c, p) where c is a control in $\mathcal{C}$ and p is the amplitude of the mitigation of the impact of the threat T when adopting the control c. For instance, we can take $\mathcal{A} = \{0, 0.5, 1\}$, so that $\mu_T(c) = 0$ means that control c has no effect in mitigating the threat T, $\mu_T(c) = 0.5$ has partial effect on T, and $\mu_T(c) = 1$ has full effect. Under these assumptions, we obtain an instance of (3) that belongs to a particular class of MOOP called Multi Objective Combinatorial Optimization Problems (MOCOPs); see, e.g., Klamroth (2009). We observe that finding all Pareto optimal solutions of such instances of (3) requires, in the worst case, to search among $\Pi_{T\in\mathcal{T}}(k^{|\mathcal{C}_T|} - 1)$ candidate sets of controls for $k = |\mathcal{A}|$ the number of distinct real values in the co-domain of the mappings μ_T for all T in $\mathcal{T}$. The -1 in the expression considers that it is never the case that all controls in $\mathcal{C}_T$ will be adopted; this is a reasonable assumption because of multiple reasons including lack of skills to manage several different technologies on which the controls are based and constraints in costs. Indeed, this implies the decidability of the instances of the MSRMP that we consider in the rest of the work. We observe that, despite their decidability, solving these instances of the MSRMP may be quite a challenge from a computational point of view because the number of possible solutions in which to search for the optimal ones is exponential in the size of $\mathcal{C}_T$ for $T \in \mathcal{T}$. In the rest of this section, we describe a strategy to manage this problem and in Section 5.3, we propose an experimental evaluation of some refinements and study the scalability of the proposed approach in practice.

Example 5. As described above, by considering $k = 3$ possible values for the mappings μ_{T_1}, μ_{T_2}, and μ_{T_3} introduced in Example 3, the search for finding optimal solutions is among $\Pi_{T\in\mathcal{T}}(k^{|\mathcal{C}_T|} - 1) = (3^2 - 1) \times (3^2 - 1) \times (3^1 - 1) = 128$ candidates. Note that we do not consider the situation in which all controls are in place as this would yield a risk equal to zero, thereby making the search for optimal solutions trivial. This is reasonable in practice since, as already observed, it is unlikely that the stakeholders will be able to adopt all security controls in $\{\mathcal{C}_T\}_{T\in\mathcal{T}}$ because of other constraints such as those related to budget and required security skills for their deployment.

To simplify the solution of the instances of (3), we consider an associated problem derived from (3), by introducing a variable x_T to replace $1 - m(T)$ and obtain:

$$\begin{array}{l} min_{\langle x_T\rangle_{T\in\mathcal{T}}} \ \ \langle \frac{1}{|\mathcal{T}|} \Sigma_{T\in\mathcal{T}} \ (i_s(T) * x_T)\rangle_{s\in\mathcal{S}} \\ \text{subject to } x_T \in \{1 - m(T)\} \text{ for each } T \in \mathcal{T} \end{array} \quad (4)$$

where $m(T)$ is the expression defined in 2, $\langle x_T\rangle_{T\in\mathcal{T}}$ is the vector of variables representing mitigation amplitudes when considering an arbitrary total order over $\mathcal{T}$. For each threat T in $\mathcal{T}$, we have that $|\{1 - m(T)\}|$ is the number of distinct sum values, divided by the number of controls in $\mathcal{C}_T$, that can be obtained by adding values in $\mathcal{I}$ (that, in our examples, is the set $\{0, 0.5, 1\}$) according to a μ_T that induces a value $m(T)$. The space of solutions of the modified version of (4), is thus $\Pi_{T\in\mathcal{T}}|\{1 - m(T)\}|$ which may be remarkably less than $\Pi_{T\in\mathcal{T}}(k^{|\mathcal{C}_T|} - 1)$. For instance, consider Example 3, the first two tables contain 8 different mitigation vectors with only 4 different values for the function $m(\cdot)$.

Example 6. Recall Example 3, consider only the values of $m(T)$ that are distinct, and derive the values $x_T = 1 - m(T)$ for each $T \in \{T_1, T_2, T_3\}$:

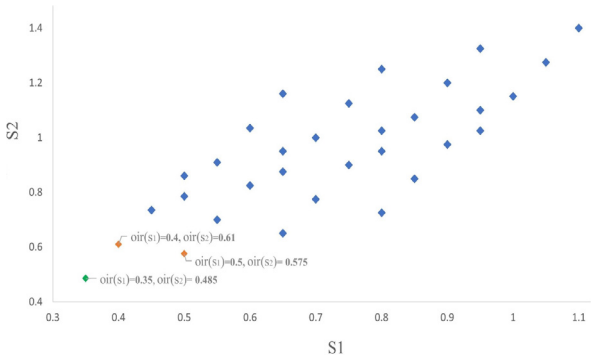

Fig. 2. The solution points.

$m(T_1)$	x_{T_1}	$m(T_2)$	x_{T_2}	$m(T_3)$	x_{T_3}
0	1	0	1	0	1
0.25	0.75	0.25	0.75	0.5	0.5
0.5	0.5	0.5	0.5		
0.75	0.25	0.75	0.25		

The set of possible solutions of (4) is the set of all triples of the form $\langle x_{T_1}, x_{T_2}, x_{T_3}\rangle$ whose values are taken from the three tables above and thus the size of such a set is $4 \times 4 \times 2 = 32$. Observe that this is one-fourth of the size of the set of potential solutions to the original problem (3), namely $\Pi_{T\in\{T_1,T_2,T_3\}}(k^{|\mathcal{C}_T|} - 1) = (3^2 - 1) \cdot (3^2 - 1) \cdot (3^1 - 1) = 128$. For larger problem instances, the reduction is much more substantial as we will see in Section 5.3 below. By considering the 32 triples $\langle x_{T_1}, x_{T_2}, x_{T_3}\rangle$, we can derive the values of the overall impact values for the two stakeholders by recalling that $oir(s) = ir_s(T_1) + ir_s(T_2) + ir_s(T_3)$, $ir_s(T) = i_s(T) \cdot (1 - m(T))$ from (1), (2, and $x_T = 1 - m(T)$ for $s \in \{s_1, s_2\}$ and for $T \in \{T_1, T_2, T_3\}$. Also, recall that the definition of $i_s(\cdot)$ can be found in Example 4. The pairs $(oir(s_1), oir(s_2))$ so computed are plotted in Fig. 2 where the x-axis shows the values of $oir(s_1)$ and the y-axis those of $oir(s_2)$.

It is then immediate to see that the point (0.35,0.485) at the bottom left (in green) is the Pareto optimal solution. We also observe that the two points in orange are not dominated by any other points but the optimal one.

Indeed, it is possible to find solutions of (3) corresponding to those of the simplified version of (4) by adapting the procedure above. Let $\langle x^*_T\rangle_{T\in\mathcal{T}}$ be a solution for (4). By definition and the simplifying assumption above, there must exist μ^*_T such that $x^*_T = 1 - m(T) = 1 - \frac{\Sigma_{c\in\mathcal{C}_T} \ \mu^*_T(c)}{|\mathcal{C}_T|}$ for each $T \in \mathcal{T}$ and it is thus immediate to discover all the solutions of (3).

Example 7. We explain how it is possible to derive the sets of controls associated to a certain triple $\langle x^*_{T_1}, x^*_{T_2}, x^*_{T_3}\rangle$. To illustrate, we consider the (orange) point in Fig. 2 with coordinates (0.4,0.61) that is associated to the triple $\langle x^*_{T_1}, x^*_{T_2}, x^*_{T_3}\rangle = \langle 0.25, 0.5, 0.5\rangle$. From (2) and $x_T = 1 - m(T)$, it is immediate to derive that

$$\frac{\mu_{T_1}(c_1) + \mu_{T_1}(c_2)}{2} = 1 - x^*_{T_1} \quad \frac{\mu_{T_2}(c_3) + \mu_{T_2}(c_4)}{2} = 1 - x^*_{T_2}$$
$$\frac{\mu_{T_3}(c_5)}{1} = 1 - x^*_{T_3}$$

so that we are left with the problem of enumerating all mitigation mappings $\mu_{T_1}(\cdot), \mu_{T_2}(\cdot), \mu_{T_3}(\cdot)$ satisfying the three equalities above. The following table lists all possible such mappings:

	$x^*_{T_1} = 0.25$		$x^*_{T_2} = 0.5$		$x^*_{T_3} = 0.5$
	$\mu_{T_1}(c_1)$	$\mu_{T_1}(c_2)$	$\mu_{T_2}(c_3)$	$\mu_{T_2}(c_4)$	$\mu_{T_3}(c_5)$
$\mathbb{S}_1$	1	0.5	0.5	0.5	0.5
$\mathbb{S}_2$	0.5	1	0.5	0.5	0.5
$\mathbb{S}_3$	1	0.5	1	0	0.5
$\mathbb{S}_4$	0.5	1	1	0	0.5
$\mathbb{S}_5$	1	0.5	0	1	0.5
$\mathbb{S}_6$	0.5	1	0	1	0.5

The obvious question is the computational complexity of enumerating all possible mitigation mappings $\mu_T(\cdot)$ such that

$$\frac{\Sigma_{c\in\mathcal{C}_T}\mu_T(c)}{|\mathcal{C}_T|} = 1 - x^*_T \tag{5}$$

for each $T \in \mathcal{T}$; notice that the three equalities in Example 7 are instances of (5). Indeed, if there exists a (practically) efficient algorithm to enumerate the mitigation mappings satisfying (5), we can hope that solving instances of (4) and then using such an algorithm to derive the corresponding solutions of (3) is an efficient alternative to solving directly the latter as the number of the possible solutions of (4) is smaller (as we have seen in Example 6 and even substantially so as we will see in Section 5.3) than those of (3).

To answer this question, we consider the Subset Sum Problem (SSP) with multiplicities (Cormen et al., 2001), i.e. given a multiset X of integers and an integer s, does any non-empty multisubset of X sum to s? Solving the instances of (5) for each $T \in \mathcal{T}$ is equivalent to solving an instance of the SSP under the natural assumption that x^*_T and the values in $\mathcal{A}$ are real numbers that can be represented as $v \cdot 10^{-d}$ for v and d positive integers such that $0 < v \cdot 10^{-d} < 1$. To see this, observe that all the values in $\mathcal{A} \cup \{1 - x^*_T\}$ can be transformed to integers by multiplying each one by their maximum exponent d when represented as $v \cdot 10^{-d}$, the integers so obtained from the values in $\mathcal{A}$ are added to the multiset X, each one with multiplicity equal to the number of controls in $\mathcal{C}_T$ for $T \in \mathcal{T}$, and the integer obtained from $1 - x^*_T$ is set to s. Several different algorithms are available to solve this problem with different complexities ranging from exponential to (pseudo-)polynomial (see, e.g., Cormen et al., 2001). The most naive algorithm (with exponential worst-case complexity) amounts to cycling through all multisubsets of X and, for each one, check if it sums to s. To solve the SSP, it is possible to stop as soon as one solution is found, but in our case, we need to find all possible solutions. Indeed, the naive algorithm can be trivially adapted to do this, resulting in exponential best-case and worst-case complexity. Despite being in such a complexity class, the naive algorithm turns out to give satisfactory results in practice because the instances derived from (5) are typically small because the cardinality of $\mathcal{C}_T$ is relatively small for each $T \in \mathcal{T}$ or can be reduced by exploiting the knowledge of security experts. We will discuss this issue in Section 5 below.

4. Defining instances of the MSRMP

Our main goal is to assist in the identification of the best possible set of controls to minimize the risk for all stakeholders. This has been formalized as solving an appropriate instance of the MSRMP introduced in Section 3.2. To specify instances of the MSRMP in either statement (3) or (4), we consider additional information that is typically available in many methodologies for risk assessment. In the rest of this section, we first (Section 4.1) consider the problem statement (4) and discuss an approach to derive the risk residue i_s for each stakeholder s that yields a problem with a reduced search space whose solutions can be used to derive optimal mitigation mappings as explained at the end of Section 3.2. We will see that this approach requires the stakeholder s to take several decisions that are highly subjective and this may lead to bias. Then (Section 4.2), we propose an approach that aims to reduce the level of subjectivity in defining the risk residue i_s that requires to consider the general problem statement (4). We will discuss how also in this case it is possible to first solve a problem with a reduced search space and then to derive optimal mitigation mappings. Both approaches require to identify a set $\mathcal{S}$ of stakeholders, a set $\mathcal{T}$ of threats, a family $\{\mathcal{C}_T\}_{T\in\mathcal{T}}$ of sets of controls (each one associated to a threat $T \in \mathcal{T}$), and be able to define the mapping i_s that quantifies the impact level for each stakeholder $s \in \mathcal{S}$ and the residual risk x_T for each threat T that results from applying a certain set of controls (or, equivalently, from selecting a certain mitigation mapping μ_T). The approaches presented in Sections 4.1 and 4.2 differ in the definition of i_s. For this reason, we preliminary consider the definitions of the other parameters, namely $\mathcal{T}$, $\{\mathcal{C}_T\}_{T\in\mathcal{T}}$, and x_T.

As reviewed earlier, the literature lists several approaches (e.g., Shostack, 2014; Wuyts and Joosen, 2015) dealing with threat identification together with appropriate mitigation controls that allow us to define the set $\mathcal{T}$ of threats and the family $\{\mathcal{C}_T\}_{T\in\mathcal{T}}$ of sets of controls associated to the threats in $\mathcal{T}$. The decision to select a method or another depends on the specific needs and specific concerns (see, e.g., the discussion in Shevchenko et al. (2018)). For instance, Microsoft STRIDE (Shostack, 2014) is a well-established threat modeling to identify security threats according to a predefined classification of threat types. It is an acronym for *Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service,* and *Elevation of privilege*. These threat types represent the violation of the primary security properties: authentication, integrity, non-repudiation, confidentiality, availability, and authorization. LINDDUN (Wuyts and Joosen, 2015) is another well-known threat modeling approach to identify privacy threats, and it is an acronym for *Linkability, Identifiability, Non-repudiation, Unawareness, Detectability, Disclosure of information,* and *Non-compliance*. Similar to STRIDE, also, these represent violations of properties characterizing different dimensions of privacy. For concreteness, an instance of the set $\mathcal{T}$ is shown in Table 2 and an instance of the family $\{\mathcal{C}_T\}_{T\in\mathcal{T}}$ can be found in the first two columns of Table 3 (for instance, consider T_4 = Denial of service, $\mathcal{C}_{T_4}$ is associated with three controls, namely *Enabling off-line authentication, Network monitoring,* and *Prevention mechanisms for DoS attacks like firewalls, etc.*); both are related to the running example introduced in Section 3.1. For the applicability of the method proposed in this work, any methodology that allows for the definition of $\mathcal{T}$ and $\{\mathcal{C}_T\}_{T\in\mathcal{T}}$ can be used.

We are left with the problem of defining i_s for $s \in \mathcal{S}$ and x_T for $T \in \mathcal{T}$. Concerning the latter, recall that according to (4) and (1), the risk residue $x_T \in \{1 - m(T)\}$ with $m(T) = 1 - \frac{\Sigma_{c\in\mathcal{C}}\mu_T(c)}{|\mathcal{C}_T|}$, i.e. x_T is the risk residue obtained by applying a certain combination of the security controls available in $\mathcal{C}_T$ for the threat T according to the mitigation mapping μ_T. Recall also that $\mu_T(c)$ measures the impact of T after applying control $c \in \mathcal{C}_T$ and thus $m(T)$ measures the aggregated mitigating effect of selecting a given set of controls in $\mathcal{C}_T$ on the risk of T materializing (under the assumption that the mitigations are independent of each other). The third and fourth columns of Table 3 show a given mitigation mapping μ_T and the associated value x_T of the resulting risk residue. It will be the task of an automated solver to explore the space of all possible values of x_T and find those that are Pareto-optimal solutions of the MSRMP instance (4) so that it is possible to derive the optimal mitigation mappings as described at the end of Section 3.2; see

Table 4
The assigned impacts to each stakeholders' preferences for each threat in our scenario.

Stakeholders ($\mathcal{S}$)	Protection Criteria ($\mathcal{P}$)	Weights PW_p^s	Aversion level (al_p^s)				
			T_1	T_2	T_3	T_4	T_5
Data Subject	Health condition	0.4	0	4	0	3	4
	Individual freedom	0.2	0	2	4	3	3
	Social situation	0.3	1	2	3	0	3
	Financial situation	0.1	0	3	1	0	3
Data Controller	Reputational situation	0.4	1	2	3	2	2
	Financial situation	0.6	2	2	3	3	2

Section 4.1). As already said above, we will see that finding optimal values for x_T is crucial also for solving instances of the general problem statement (3); see Section 4.2.

4.1. Defining impacts levels according to stakeholders: A first attempt

Different stakeholders have different criteria that define what they consider risky. Data controllers (e.g., companies) typically choose business impact criteria, such as financial impact or reputation, whereas data subjects (e.g., individuals) evaluate risk based on impact on their personal sphere. For the running example introduced in Section 3.1, we consider the *social situation, individual freedom, financial situation* (Oetzel and Spiekermann, 2014), and *health condition* as the data subject protection criteria while for the data controller, *reputational situation* and *financial situation* are the protection criteria, which are linked to indirect or direct pecuniary losses. Additionally, each stakeholder has different preferences, which result in different importance given to different criteria; e.g., in the running example, the health condition criterion is more momentous than others for patients. We capture these high-level stakeholder preferences by assigning a weight to each stakeholder's protection criterion. The associations among stakeholders, protection criteria, and weights are shown in the first three columns of Table 4. Formally, we assume the availability of a set $\mathcal{P}$ of protection criteria, a family $\{PW_p^s\}_{p\in\mathcal{P},s\in\mathcal{S}}$ of weights associated to a preference p for each stakeholder s besides the definitions of $\mathcal{T}$, $\{\mathcal{C}\}_{T\in\mathcal{T}}$, and x_T for $T \in \mathcal{T}$ as discussed above in this section.

The additional information in $\mathcal{P}$ and $\{PW_p^s\}_{p\in\mathcal{P},s\in\mathcal{S}}$ are used to define the impact level i_s by giving a quantitative evaluation of the negative influence that a threat $T \in \mathcal{T}$ may have on a preference $p \in \mathcal{P}$ for a certain stakeholder $s \in \mathcal{S}$. The intuition is to characterize how each threat is perceived as more or less dangerous by each stakeholder in relation to his/her own protection criteria. For instance, in the context of the running example, it is very unlikely that excessive storage of patients' health data would damage the data controller's reputation; by increasing stored data, there is financial damage on the data controller cause of cost of storage and management of the IT infrastructure. On the other hand, the reputation of patients is not affected by excessive storage of personal data; indeed, a larger amount of stored data increases the impact of data breaches and leaks on the rights and freedoms of patients. For this, we assign an impact value in $\mathcal{IL}$ (recall that this set typically contains a finite set of integer values from 0 to 4 included) to the level of aversion that each stakeholder s has for a threat T acting on a given protection criterion p. Formally, we assume the definition of an *aversion mapping* $al_p^s : \mathcal{P} \to \mathcal{IL}$ for each preference $p \in \mathcal{P}$ and stakeholder $s \in \mathcal{S}$. At this point, we are in the position to define i_s by combining the weight PW_p^s and the mapping al_p^s as follows:

$$i_s(T) = \frac{1}{|il_{max}|} \sum_{p\in\mathcal{P}} al_p^s(T) \times PW_p^s \tag{6}$$

where $il_{\max} \in \mathcal{IL}$ represents the maximum impact level (in our case, it is 4). The crux to specify i_s is thus to define the family $\{al_p^s\}_{p\in\mathcal{P},s\in\mathcal{S}}$ of aversion mappings. This can be done as shown in the fourth column of Table 4 where each threat $T \in \mathcal{T}$ gets an aversion level al_p^s between 0 and 4 (recall that 0 means no, 1 low, 2 moderate, 3 critical, and 4 catastrophic impact) for each protection criterion p and stakeholder s. Intuitively, the values are assigned by answering the question "For the stakeholder s, what would be the impact level on the criterion p if the threat T happen?" To illustrate, consider Table 4 in which the aversion level of the *health condition* for the second threat (T_2) according to the data subject ($s = DS$) is 4 and thus the value of $i_{DS}(T_2)$ will be $\frac{(0.4\times4)+(0.2\times2)+(0.3\times2)+(0.1\times3)}{4} = 0.725$ according to (6).

To summarize, we have described an approach to define i_s by assuming the capability of identifying protection criteria for each stakeholder (i.e. being able to define the set $\mathcal{P}$), of quantifying the relevance of each such criterion (in a scale between 0 and 1) for each stakeholder (i.e. being able to define the family $\{PW_p^s\}_{p\in\mathcal{P},s\in\mathcal{S}}$), and assigning an aversion level of each stakeholder when a threat impacts a given protection criterion (i.e. defining the family $\{al_p^s\}_{p\in\mathcal{P},s\in\mathcal{S}}$). This allows us to define an instance of the MSRMP (4) which, as we will see in the following, can be solved by using available techniques and then, as described at the end of Section 3.2, to identify the set of Pareto optimal mitigation mappings that minimize the risks with respect the various stakeholders. However, we observe that it may be non-obvious to quantify the weights in $\{PW_p^s\}_{p\in\mathcal{P},s\in\mathcal{S}}$ and the aversion level mappings in $\{al_p^s\}_{p\in\mathcal{P},s\in\mathcal{S}}$ as their definitions are quite subjective for each stakeholder. This is somehow unavoidable because it is up to each stakeholder to define i_s, however it is important to mitigate possible bias that would make the solutions of the corresponding instance of the MSRMP (4) hardly useful in practice or even detrimental because of an over or under estimation of the risk levels with negative business or privacy impacts, respectively, on some stakeholders. We can consider to assign the definitions of $\{PW_p^s\}_{p\in\mathcal{P},s\in\mathcal{S}}$ and $\{al_p^s\}_{p\in\mathcal{P},s\in\mathcal{S}}$ to two independent groups of experts for each stakeholder so to mitigate possible bias. In the next section, we describe a refined approach to define an instance of the MSRMP (3) that aims to further reduce the level of subjectivity of each stakeholder in defining i_s.

4.2. A less subjective definition of impact levels

Our goal is to reduce the level of subjectivity with which i_s is defined. The idea is to refine the definition of i_s given above by in-

troducing a cross-weighting system to reduce bias resulting from stakeholders as much as possible. Besides the availability of a set $\mathcal{P}$ of protection criteria and a family $\{PW_p^s\}_{p\in\mathcal{P},s\in\mathcal{S}}$ of weights associated to a preference p for each stakeholder s, we consider a set $\mathcal{G}$ of protection goals which play a crucial role in identifying appropriate security controls (see, e.g., Zwingelberg and Hansen, 2011). Indeed, Confidentiality, Integrity, and Availability are obvious candidates to be included in the set $\mathcal{G}$ (see, e.g., Brooks et al., 2017). However, these are not enough to consider the complex protection requirements deriving from national and international legal provisions such as those concerning data protection contained in the GDPR. For this reason, in the rest of the paper, we assume the set $\mathcal{G}$ to contain the "data protection goals" introduced by the Standard Data protection Model (SDM) (für Datenschutz, 2020).

To systematize data protection requirements of the GDPR, the SDM employs "protection goals". The data protection requirements seek to ensure legal compliance processing, which technological and organizational safeguards must ensure. The assurance consists in lowering the risk of deviations from legally compliant processes to a suitable degree. Unauthorized processing by third parties and the failure to carry out mandatory processing procedures are examples of deviations to avoid. The data protection goals combine and arrange the criteria for data protection requirements and can be operationalized through integrated, scalable measures (für Datenschutz, 2020). These protection goals are

G1. ***Confidentiality*** refers to the requirement that no person is allowed to access personal data without authorisation.
G2. ***Integrity*** refers, on the one hand, to the requirement that information technology processes and systems continuously comply with the specifications that have been determined for the execution of their intended functions. On the other hand, integrity means that the data to be processed remain intact, complete, and up-to-date.
G3. ***Availability*** is the requirement that personal data must be available and can be used properly in the intended process. Thus, the data must be accessible to authorised parties and the methods intended for their processing must be applied.
G4. ***Unlinkability*** &***Data minimization*** where the *unlinkability* goal refers to the requirement that data shall be processed and analysed only for the purpose for which they were collected, while the *data minimization* goal covers the fundamental requirement under data protection law to limit the processing of personal data to what is appropriate, substantial and necessary for the purpose.
G5. ***Transparency*** refers to the requirement that the data subject as well as the system operators and the competent supervisory authorities can identify to a varying extent, which data are collected and processed for a particular purpose, and which systems and processes are used for this purpose, where the data flow to which purpose, and who is legally responsible for the data and systems in the various phases of data processing.
G6. ***Intervenability*** refers to the requirement that the data subjects are effectively granted the right to notification, information, rectification, blocking and erasure at any time.

The SDM have provided precise mappings between the GDPR requirements and these protection goals (for more details, see the table [3] on pages 28 to 30). These mappings can be interpreted as if threats adversely affecting these protection goals mean non-compliance with the GDPR requirements. Working with protection goals simplifies the modeling of functional requirements in use cases and the visualization of conflicts. They also enable the methodical application of legal requirements into technological and organizational measures and are therefore "optimization requirements". We observe that our approach can be applied with other protection goals, we consider those of für Datenschutz (2020) only for the sake of concreteness.

The goal of the approach discussed below is twofold: (i) identify how many goals each threat is impacting and (ii) measure the amplitude of the impact on each goal of a given threat. We start by considering (i).

For example, a "*Denial of service*" threat will intuitively have more impact on the data availability goal rather than on the integrity goal; an "*Identity theft*" threat will have more impact on the data confidentiality goal. To keep track of this, we use a *Threat-Protection Goals association* as shown in the first two columns in Table 5 where the "×" ("-") mark in a cell means the goal in the column is affected (not affected, respectively) by the threat in the row (the particular instance of the threat-protection goals association is related to the running example of Section 3.1). Intuitively, the more a threat impacts multiple goals, the more it is considered pervasive (e.g., threat T_2 is the most pervasive in Table 5 as it affects 3 goals); the more a goal is impacted by multiple threats, the more it is considered scattered (e.g., goal $G1$ is the most scattered in Table 5 as it impacts 3 threats). The third column of Table 5 shows the so called Observation Weight

$$OW_T = \frac{AG_T}{\sum_{T\in\mathcal{T}} AG_T} \tag{7}$$

that measures how much a threat T is pervasive for the goals in $\mathcal{G}$, where AG_T is the number of goals in $\mathcal{G}$ affected by a threat $T \in \mathcal{T}$. For example, in Table 5, the observation weight OW_{T_1} is 2/10, where $G1$ and $G4$ are the two affected goals by T_1, and the total number of affected goals is 10.

We now consider objective (ii), namely to measure the amplitude of the impact on each goal of a given threat. This is necessary as soon as we realize that the information in Table 5 is not enough alone to define i_s because it may be the case that the impact value can be much higher when a goal is impacted severely by a single threat rather than when this is impacted by many threats but only lightly. We do this in two steps. First, we define the *normalized threat criticality level* as

$$NTC_T = \frac{OW_T \times x_T}{\sum_{T\in\mathcal{T}}(OW_T \times x_T)} \tag{8}$$

to quantify the severity of a threat $T \in \mathcal{T}$ (recall that x_T is the impact residue of the threat T after applying the security controls according to a mitigation mapping μ_T). Intuitively, NTC_T is the level of danger of a threat T among all threats in $\mathcal{T}$, or in other words, the relative importance of T with respect to all other threats in $\mathcal{T}$.

By having obtained the observation weights (in Table 5) and the calculated x_T values (in Table 3), the computed normalized threat criticality values for $T \in \mathcal{T}$ are shown in the second column of Table 6.

The second step to achieve goal (ii) above is to use the normalized threat criticality level to weight the function i_s defined in Section 4.1 when considering a certain protection goal $G \in \mathcal{G}$ for a given stakeholder s so to define the overall impact residue as follows

$$oir(s) = \sum_{G\in\mathcal{G}} \left(\frac{\sum_{T\in\mathcal{T}} \xi_{T,G} \times NTC_T \times i_s(T)}{\#(\mathcal{T}, G)} \right) \tag{9}$$

where $\xi_{T,G}$ is 1 when the threat $T \in \mathcal{T}$ compromises the goal $G \in \mathcal{G}$ and 0 otherwise; $\#(\mathcal{T}, G)$ is the number of threats in $\mathcal{T}$ that have an impact on the goal G (this means that $\#(\mathcal{T}, G) = \sum_{T\in\mathcal{T}} \xi_{T,G}$). Observe that the expression between parentheses in (9) can be seen as the average impact on a given goal G with respect to the threats

[3] The Standard Data Protection Model (SDM), https://www.datenschutzzentrum.de/uploads/sdm/SDM-Methodology_V2.0b.pdf

Table 5
Affected protection goals by each threat and the observation weights in our scenario, G1= Confidentiality, G2= Integrity, G3= Availability, G4= Unlinkability & Data minimization, G5= Transparency, and G6= Intervenability.

Threat	Data Protection Goals						Observation Weights (OW)
	G1	G2	G3	G4	G5	G6	
T_1	×	-	-	×	-	-	2/10
T_2	×	×	×	-	-	-	3/10
T_3	×	-	-	×	-	-	2/10
T_4	-	×	×	-	-	-	2/10
T_5	-	-	-	-	-	×	1/10

Table 6
Threat criticality and impact level values together with the computed protection goals' impacts for each threat for the data subject (DS) and the data controller (DC).

Threats ($\mathcal{T}$)	Normalized Threat Criticality (NTC)	Normalized Impact Level $i_s(T)$		Protection Goals' Impacts									
				G1		G2		G3		G4		G6	
		DS	DC	DS	DC	DS	DC	DS	DC	DS	DC	DS	DC
T_1	0.17	0.075	0.4	0.013	0.068	0.000	0.000	0.000	0.000	0.013	0.068	0.000	0.000
T_2	0.22	0.725	0.5	0.165	0.112	0.165	0.112	0.165	0.112	0.000	0.000	0.000	0.000
T_3	0.11	0.45	0.75	0.048	0.080	0.000	0.000	0.000	0.000	0.048	0.080	0.000	0.000
T_4	0.36	0.45	0.65	0.000	0.000	0.160	0.231	0.160	0.231	0.000	0.000	0.000	0.000
T_5	0.14	0.85	0.5	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.121	0.071
Average impact	-	-	-	0.074	0.087	0.161	0.172	0.161	0.172	0.030	0.074	0.121	0.071

in $\mathcal{T}$ that are relevant to G. For instance, according to Table 5, the *intervenability* goal ($G6$) is affected only by T_5 which means that $\#(\mathcal{T}, G6)$ is 1. According to Table 6, the average impact of the *confidentiality* goal ($G1$) for the data subject is 0.074, while the same value for the data controller is 0.087. Finally, observe that since the *transparency* goal ($G5$) is not affected by anyone of the threats (according to Table 5), it is not mentioned in Table 6 neither used for calculating the overall impact residue. By aggregating the impact average of protection goals, the overall impact residue from the data subject's point of view is $oir(DS) = 0.549$, and for the data controller is $oir(DC) = 0.576$.

At this point, we are in the position to define instances of the MSRMP statement (3) by using (9) as the definition of the overall impact residue rather than those proposed in Section 3.2. We also observe that by substituting the definition (8) to NTC_T in the expression of $oir(s)$, it is easy to see that we can derive a MSRMP similar to (4), i.e. considering x_T as variables rather than μ_T for $T \in \mathcal{T}$, for which it is possible to apply the same technique discussed at the end of Section 3.2 that allows us to solve an optimization problem over a smaller search space and then derive optimal solutions for the original problem.

5. Implementation and experimental evaluation

To validate the applicability of the proposed methodology, we have implemented a tool able to assist in defining an instance of the MSRMP as discussed in Section 4 and performed two sets of tests in order to experimentally evaluate the practicality of our approach [4].

The goal of the tool is two-fold, namely (i) assisting in the definition of an instance of the MSRMP and (ii) automatically solving the resulting instance. The architecture of the tool is illustrated in Fig. 3; the modules are implemented in Java, while the documents use JSON as the data representation format. The tool operates in two phases (see outer boxes in the figure) and assumes the availability of the sets of stakeholders $\mathcal{S}$, threats $\mathcal{T}$, security controls $\mathcal{C}$, protection criteria $\mathcal{P}$ together with their weights $\{PW_p^s\}_{p\in\mathcal{P},s\in\mathcal{S}}$, and goals $\mathcal{G}$; the first three are discussed in Section 3.2, the fourth in Section 4.1, and the last in Section 4.2. The architecture also reports how tabular definitions of the various entities can be given; for instance, the set $\mathcal{T}$ of threats can be defined as in Table 2 and the set $\mathcal{P}$ of protection goals together with their weights $\{PW_p^s\}_{p\in\mathcal{P},s\in\mathcal{S}}$ as in Table 4. We assume that these inputs are derived from the application of available and well-known techniques for risk assessment, as already discussed above; our approach is agnostic with respect to the particular methodology used. The tables specifying the inputs above are encoded in JSON format.

The first phase is semi-automated and a preliminary step to the definition of an instance of the MSRMP. More precisely, it defines the association between controls and threats $\{\mathcal{C}_T\}_{T\in\mathcal{T}}$ (see Section 3.2), the aversion level mapping al_p^s for each protection criteria $p \in \mathcal{P}$ and stakeholder $s \in \mathcal{S}$ (see Table 4 in Section 4.1), and the observation weight OW_T for each threat $T \in \mathcal{T}$; see last column of Table 5 whose value is derived according to (7). The first two outputs of this phase are obtained with human intervention as the user needs to identify which security controls are effective for each threat and which is the level of aversion of each stakeholder for a given protection criteria to be violated, whereas the last one is automatically derived after the user has specified which goals are affected by each threat.

The second phase is fully automated and aims to define and solve an instance of the MSRMP. This requires to use the outputs of the first phase to define the impact level mapping i_s for each stakeholder $s \in \mathcal{S}$; see Section 3.2) along the lines of Section 4.1 and then the overall impact residue oir as discussed in Section 4.2. At this point, the tool has fully defined an instance of the MSRMP (3) and it is left with the task of solving it. For this, it needs to enumerate all risk residues x_T for each threat $T \in \mathcal{T}$ by using the approach in Section 4 to define Table 3, derive the Normalized Threat Criticality values for the various threats, and then adapt the strategy discussed at the end of Section 3.2 to identify the mitigation mappings that are Pareto optimal.

[4] The code of the tool and the material to replicate the experiments are available at https://github.com/stfbk/MSRMP

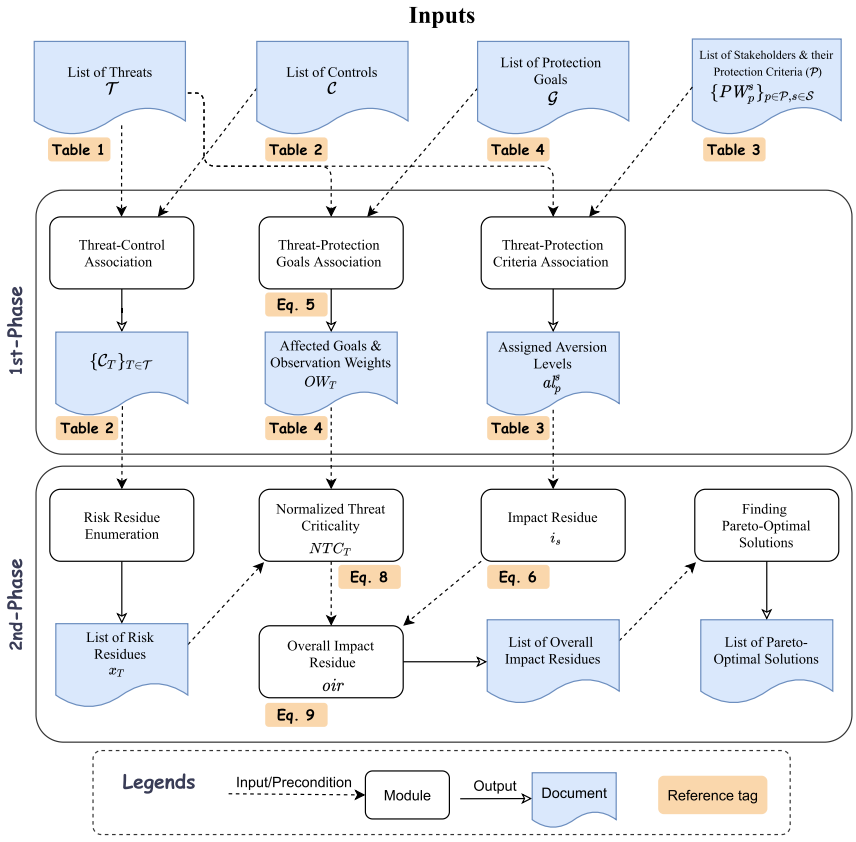

Fig. 3. Architecture of the implemented tool.

We observe that there are multiple possible strategies to combine the enumeration of risk residues and the identification of Pareto optimal values. For instance, one can first compute the entire set of feasible solutions and only after look for Pareto optimal ones or one can imagine to interleave the two activities by computing the Pareto optimal values in different subsets of the whole set of feasible solutions and then select those solutions that are Pareto optimal for the entire search space. Below, we first discuss the computational behavior of the second phase on the running example in Section 3.1 and then design two sets of tests to understand which is the most promising strategy to identify the set of Pareto Optimal risk residues or, equivalently, mitigation mappings.

5.1. Applying the prototype tool on the running example

We discuss the results of applying the second phase of our methodology, as implemented in the prototype tool, on the running example of Section 3.1. First, the tool computes the whole set of possible solutions whose cardinality is 57,600; this is as expected from the formula $\Pi_{T\in\mathcal{T}}|X_T| = |X_{T_1}| \times |X_{T_2}| \times |X_{T_3}| \times |X_{T_4}| \times |X_{T_5}| = 10 \times 20 \times 8 \times 6 \times 6 = 57,600$ presented in Section 3.2 (see Example 6). This takes around 2.1 s on a machine with 16 GB of RAM and a 1.90 GHz CPU. Each solution is a pair containing the risk residue values for the Data Subject (DS) and the Data Controller (DC). Figure 4 shows the set of possible solutions plotted on a Cartesian plane whose x-axis shows the risk residue of DS and the y-axis that of DC. By looking at the figure, it is immediate to see that the optimal solution is that on the bottom left—whose risk residue values are 0.2260 for DS and 0.4168 for DC— as it dominates all other solutions. The tool takes around 2.2 s to identify this point as the best one.

After identifying the risk residue levels, one is left with the problem of computing the set of RMPs that generate such values. A method to do this has been illustrated at the end of Section 3 and implemented in the tool that takes less than 3 s to identify the following tuple

$$\langle x^*_{T_1}, x^*_{T_2}, x^*_{T_3}, x^*_{T_4}, x^*_{T_5}\rangle = \langle 1, 0.05, 0.125, 0.16, 0.16\rangle$$

corresponding to $(0.2260, 0.4168)$ and then to identify all the RMPs associated to the above tuple of x_T values for $T \in \{T_1, T_2, T_3, T_4, T_5\}$. By recalling (5) and that $\mathcal{A} = \{0, 0.5, 1\}$, it is not difficult to see that there are $360 = 1 \times 10 \times 4 \times 3 \times 3$ distinct RMPs associated to the tuple of x_T values above since

- there is just one mitigation mapping satisfying
$$\frac{\Sigma_{c\in\{c_1,\ldots,c_5\}}\mu_{T_1}(c)}{5} = 1 - x^*_{T_1} = 0$$
as the values in $\mathcal{A}$ are non-negative values;
- there are 10 mitigation mappings satisfying
$$\frac{\Sigma_{c\in\{c_6,\ldots,c_{15}\}}\mu_{T_2}(c)}{10} = 1 - x^*_{T_2} = 0.95$$
as the only way to get 9.5 by adding 10 values from $\mathcal{A}$ is to have nine of them equal to 1 and the remaining one to 0.5;
- there are 4 mitigation mappings satisfying
$$\frac{\Sigma_{c\in\{c_{16},\ldots,c_{19}\}}\mu_{T_3}(c)}{4} = 1 - x^*_{T_3} = 0.875$$
as the only way to get $3.5 = 4 \times 0.875$ by adding 3 values from $\mathcal{A}$ is to have three of them equal to 1 and the remaining one to 0.5;

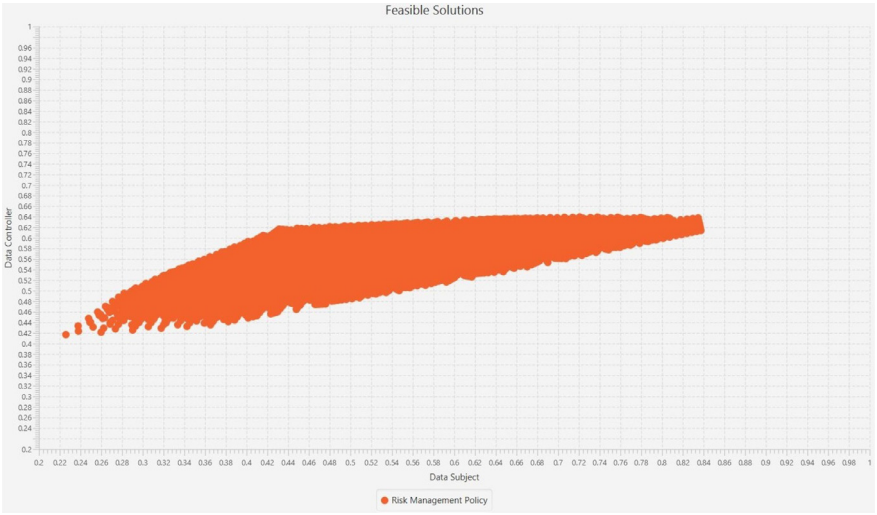

Fig. 4. All feasible solutions (i.e., the search space) in the running example scenario.

- there are 3 mitigation mappings satisfying

$$\frac{\Sigma_{c\in\{c_{20},\ldots,c_{22}\}}\mu_{T_4}(c)}{3} = \frac{\Sigma_{c\in\{c_{23},\ldots,c_{25}\}}\mu_{T_5}(c)}{3} = 1 - x^*_{T_4} = 1 - x^*_{T_5} = 0.84$$

  as the only way to get $3.5 = 3 \times 0.8$ by adding three values from $\mathcal{A}$ is to have two of them equal to 1 and the remaining one to 0.5.

The tool mechanizes the observations above and computes the set of security controls associated to the Pareto optimal solutions by solving a variant of the Sum Subset Problem (SSP) in which multisets are considered instead of sets as explained at the end of Section 3.2. Indeed, this is so because a mitigation mapping μ_T associates a control of $\mathcal{C}_T$ with a value in $\mathcal{A} = \{0, 0.5, 1\}$ for each $T \in \mathcal{T}$ and nothing prevents two or more controls to be mapped to the same value in $\mathcal{A}$. Since all solutions to the SSP should be identified to be able to enumerate all possible mitigation mappings, the algorithm is exponential in the number of security controls associated to each threat, i.e. in the cardinality of $\mathcal{C}_T$ for $T \in \mathcal{T}$. Since such a number is typically low (on average around 5 and at most 10 in our experience), the time consumption is quite reasonable in practice being around half a second at most for a single threat $T \in \mathcal{T}$. To conclude the discussion, the third column of Table 7 reports three mitigation mappings associated to the optimal solution considered above. Notice that all mitigation mappings associated to the optimal solution above suggest to avoid implementing any security control for threat T_1. This is a consequence of the impact defined in Table 4 that makes T_1 relevant only for the social situation of the DS while it is negligible for all other aspects. Given this remark, one may decide to modify the values to increase the impact of T_1 for the DS and then re-run the analysis. This is a clear advantage of having a high level of mechanization of our methodology.

Indeed, the running example is simple and poses no challenges to our prototype implementation. To understand the scalability of the proposed approach, we have designed a set of synthetic optimization problems whose sets of potential solutions is increasingly large and then experiment with two different strategies to generate and visit such a set in the process of identifying Pareto optimal solutions. This is reported in Sections 5.3 below.

5.2. Tuning MSRMP instances by means of constraints

Constraints may allow for the introduction of additional requirements. For instance, security experts may consider a certain set of security controls mandatory for a particular use case scenario and be interested in understanding which additional security controls are optimal once added to those already selected. As an another example, one may introduce risk thresholds for some or all of the stakeholders in the MSRMP. Other constraints can be added to create particular instances of the MSRMP, thereby showing the flexibility of our methodology and the advantages of reducing the MSRMP to a constraint optimization problem. Below, we give more details about how the methodology can handle the two cases discussed above.

Fixed set of controls. Let us consider the situation described in Section 5.1 that is related to the running example of Section 3.1 and assume that the set $\{c_1, c_6, c_{16}, c_{20}, \text{and } c_{23}\}$ of security controls is fixed. This means that a constraint c in such a set should be such that $\mu_T(c) = 1$ in any of the Pareto optimal solutions returned by the solver. This means that for threat T_1, the control c_1 must always be considered, which has a 1/5 contribution (5 is the total number of controls for T1) in reducing the impact of T_1. Therefore, the residual impact level $x^*_{T_1}$ for T_1 will be in the range $[1 - (4.5/5),\ 1 - (1/5)] = [0.1,\ 0.8]$ whereas it was in the range $[0.1, 1]$ when the security control c_1 was not fixed *a priori*. We can reason in a similar way the effect of fixing the remaining controls on the residual impact levels: $x^*_{T_2} \in [0.05,\ 0.9]$, $x^*_{T_3} \in [0.125,\ 0.75]$, $x^*_{T_4} \in [0.16,\ 0.84]$, and $x^*_{T_5} \in [0.16,\ 0.84]$. Notice that fixing security controls reduces the number of feasible solutions among which to search for the optimal ones. Indeed, it is easy to see that the cardinality of the set of feasible solutions is $\Pi_{T\in\mathcal{T}}|X_T| = |X_{T_1}| \times |X_{T_2}| \times |X_{T_3}| \times |X_{T_4}| \times |X_{T_5}| = 8 \times 18 \times 6 \times 4 \times 4 = 13,824$ which is much smaller than 57,600 when no security controls was fixed. The obvious by-product of this is that also the computation time and memory occupation are reduced.

Setting risk thresholds. Let us consider again the same situation described in above and consider that some stakeholders are willing to accept that their residual risk levels are contained in a certain range. This requirement can be specified by adding to the MSRMP instance constraints of the form $l^s_1 \le x^*_T \le l^s_2$ for appropriate lower l^s_1 and upper l^s_2 bounds and stakeholder $s \in \mathcal{S}$ (when the lower bound is not specified, it is possible to take $l^s_1 = -\infty$ and similarly when the upper bound is not specified, set $l^s_2 = \infty$).

Table 7
Examples of mitigation mappings associated to the optimal solution in Fig. 4.

Threats ($\mathcal{T}$)	Controls $\{\mathcal{C}_T\}_{T\in\{T_1,T_2,T_3,T_4,T_5\}}$	Possible Mitigation Combinations			x_T^*
T_1	c_1) Purpose specification	○	○	○	1
	c_2) Ensuring limited data processing	○	○	○	
	c_3) Ensuring purpose related processing	○	○	○	
	c_4) Ensuring data minimization	○	○	○	
	c_5) Enabling data deletion	○	○	○	
T_2	c_6) Ensuring data subject authentication	◐	●	●	0.05
	c_7) Ensuring staff authentication	●	◐	●	
	c_8) Ensuring device authentication	●	●	◐	
	c_9) Logging access to personal data	●	●	●	
	c_{10}) Performing regular privacy audits	●	●	●	
	c_{11}) Ensuring data anonymization	●	●	●	
	c_{12}) Providing confidential communication	●	●	●	
	c_{13}) Providing usable access control	●	●	●	
	c_{14}) Ensuring secure storage	●	●	●	
	c_{15}) Ensuring physical security	●	●	●	
T_3	c_{16}) Providing confidential communication	◐	●	●	0.125
	c_{17}) Logging access to personal data	●	◐	●	
	c_{18}) Ensuring data subject authentication	●	●	◐	
	c_{19}) Ensuring data anonymization	●	●	●	
T_4	c_{20}) Enabling offline authentication	◐	●	●	0.16
	c_{21}) Network monitoring	●	◐	●	
	c_{22}) Prevention mechanisms for DoS attacks like firewalls, etc.	●	●	◐	
T_5	c_{23}) Informing data subjects about data processing	◐	●	●	0.16
	c_{24}) Handling data subject's change requests	●	◐	●	
	c_{25}) Providing data export functionality	●	●	◐	

Table 8
Pareto's solutions with the defined risk threshold.

Pareto Solutions	Data Subject	Data Controller
$\mathbb{S}_1$	0.4514	0.5518
$\mathbb{S}_2$	0.4503	0.5536
$\mathbb{S}_3$	0.4504	0.5530
$\mathbb{S}_4$	0.4520	0.5501
$\mathbb{S}_5$	0.4516	0.5502
$\mathbb{S}_6$	0.4505	0.5525

For example, we set the threshold risk exposure level for the data subject and the data controller to 0.45 and 0.55, respectively, meaning that these stakeholders are willing to accept residual risk level above such values. To compare with the situation without any threshold constraints in which only one Pareto solution is identified (as we described above in Section 5.1), while in this case, 6 Pareto solutions are identified as reported in Table 8.

5.3. Experimental results

This section undertakes some experimental evaluations to examine the scalability of proposed methodology through the implemented tool. Hence, we present two test cases to assess the computational time and resources in the following. Since the instances of the variant of the SSP required to enumerate all possible mitigation mappings corresponding to each Pareto optimal solution of the form $\langle x_T\rangle_{T\in\mathcal{T}}$ are typically small, their solution does not consume a relevant amount of resources (both time and memory) and thus we disregard this activity in the discussion below.

5.3.1. Test 1: Upfront computation of feasible solutions

The goal of the first set of tests is to evaluate the strategy of computing the set of feasible solutions upfront and then identify those that are Pareto optimal. The idea is to understand the time and memory occupation required to do this while increasing the number of threats and the number of security controls per threat. We consider two stakeholders (i.e. $|\mathcal{S}| = 2$), the protection criteria $\mathcal{P}$ are the same as those in Table 4, the number of protection goals are 6 as those introduced in Section 4.2, an increasing number $|\mathcal{T}| = 5, 6, 7, 8$ of threats, and a number $q = 4, 5$ of security control associated with each threat so that $|\mathcal{C}_T| = q*5, q*6, q*7, q*8$. For each one of these configurations, we measure the time (in seconds) and the memory occupation (in GB of heap) taken to compute the entire set of feasible solutions when running our prototype on a cluster with a CPU of 3.2 GHz and 500 GB of RAM. We do not include the time to identify the Pareto Optimal solutions as the resource consumption for computing the feasible set of solutions (see the last two columns of Table 9) clearly shows the exponential behavior for both computation time and memory occupation despite the dramatic reduction in the search space (consider the values in the column Reduction Factor) obtained by using the approach of solving with respect to risk residues in place of mitigation mappings discussed at the end of Section 3.2.

5.3.2. Test 2: Interleaving the computation of feasible and optimal solutions

The first test set clearly shows that the upfront computation of the whole set of feasible solutions does not scale. For this reason, we designed a different approach whereby the two activities are interleaved by computing non-overlapping sub-sets of the feasible solutions and then identify those that are Pareto Optimal. As already observed, this can be done in different ways and we propose two strategies both parameterized by the size d of the sub-set of feasible solutions that are being considered.

- In the first strategy, we collect the Pareto Optimal solutions identified in each sub-set with cardinality d of the set of feasible solutions in a list ℓ and once the entire set of feasible solutions has been covered, the list ℓ is processed to extract the final set of Pareto Optimal solutions.
- The second strategy is similar to the previous one except for the fact that the content of the list ℓ of Pareto Optimal solutions for a given sub-set of the set of feasible solutions is added to the next sub-set of feasible solutions to be considered so that, when considering the last sub-set, we identify the final set of Pareto Optimal solutions.

To study the scalability in terms of resource consumption of these two strategies, we define a second test set with the same parameters of the previous one except for $|\mathcal{T}| = 6, 7, 8, 9$ and the

Table 9
Experimental results of Test 1. Legend: Reduction Factor, Computation Time is in Seconds (S), and the maximum Heap Size is in Gigabyte (GB).

$\lvert\mathcal{T}\rvert$	$\lvert\mathcal{C}_T\rvert$	Solution Set Size		Reduction Factor	Computation Time (S)	Heap Size (GB)
		$\Pi_{T\in\mathcal{T}}(k^{\lvert\mathcal{C}_T\rvert}-1)$	$\Pi_{T\in\mathcal{T}}\lvert X_T\rvert$			
5	20	$32,768\cdot 10^5$	32,768	10^5	0.312	∼0.25
6	24	$262,144\cdot 10^6$	262,144	10^6	1.2	∼1.5
7	28	$2,097,152\cdot 10^7$	2,097,152	10^7	9.7	∼12
8	32	$16,777,216\cdot 10^8$	16,777,216	10^8	237	∼29
5	25	$\sim 8.29\cdot 10^{11}$	100,000	$\sim 8.29\cdot 10^6$	0.626	∼0.5
6	30	$\sim 2.01\cdot 10^{14}$	1,000,000	$\sim 2.01\cdot 10^8$	3.7	∼9
7	35	$\sim 4.86\cdot 10^{16}$	10,000,000	$\sim 4.86\cdot 10^9$	105	∼28
8	40	$\sim 1.17\cdot 10^{19}$		$\sim 1.17\cdot 10^{11}$	2787	∼416
			100,000,000			

Table 10
Experimental results based on the two defined strategies.

Test Case	$\lvert\mathcal{T}\rvert$	$\lvert\mathcal{C}_T\rvert$	Computation Time (Second) and RAM Heap Size (Megabyte)					
			d=8	d=64	d=512	d=4,096	d=32,768	d=262,144
Strategy1	6	24	4.7(S), 308(MB)	2.5(S), 256(MB)	2.7(S), 256(MB)	3.7(S), 320(MB)	11.3(S), 499(MB)	6.8(S), 2,422(MB)
	7	28	95.5(S), 986(MB)	8.1(S), 382(MB)	8.7(S), 256(MB)	10.7(S), 459(MB)	15.8(S), 900(MB)	357(S), 2,509(MB)
	8	32	2,098.4(S), 1,282(MB)	71(S), 308(MB)	52(S), 308(MB)	62.5(S), 497(MB)	159.5(S), 1,004(MB)	317.5(S), 3,500(MB)
	9	36	T/O	7,346.7(S), 533(MB)	541.3(S), 308(MB)	560(S), 522(MB)	575.9(S), 1575(MB)	5,124.9(S), 3,812(MB)
Strategy 2	6	24	2.5(S), 256(MB)	4.5(S), 256(MB)	2.7(S), 256(MB)	3.9(S), 308(MB)	5.6(S), 826(MB)	7.2(S), 2,405(MB)
	7	28	10.7(S), 256(MB)	10.7(S), 256(MB)	12.7(S), 256(MB)	8.9(S), 525(MB)	10.9(S), 1,037(MB)	60.9(S), 2,471(MB)
	8	32	68.7(S), 256(MB)	83.3(S), 256(MB)	70.9(S), 256(MB)	58.3(S), 256(MB)	64.7(S), 1,186(MB)	108.9(S), 4,066(MB)
	9	36	567.3(S), 256(MB)	557.2(S), 308(MB)	507.6(S), 308(MB)	553.7(S), 408(MB)	555.8(S), 1,513(MB)	934(S), 4,066(MB)

number of security controls q associated to each threat is 4. We consider increasing values of $d = 8^h$ for $h = 1, 2, 3, 4, 5, 6$ to understand how the cardinality of the sub-set of the feasible solutions affect performances. As for the previous test set, we measure the timing (in seconds) and the heap occupation (in MB) with a time out (T/O) of 3 h. As the results—obtained on a personal computer with a CPU of 1.90 GHz and 16 GB of RAM—in Table 10 shows, the scalability is much improved with respect to the results of the first test above, regardless of the strategy adopted to identify the Pareto Optimal solutions. It is worth noticing that for this test set we consider a less powerful computer and include the computation for identifying the Pareto Optimal solutions. Although there is no clear winner between the two strategies described above, a closer analysis of the results in Table 10 shows that the second strategy is better than the first one in most cases and in particular for larger instances of the MSRMP; for example, consider the test case with 8 threats and $d = 8$, the computation time and the maximum heap space used by the first strategy are 2,098.4 s and 1,282 MB, whereas those used by the second strategy are 68.7 s and 256 MB. We observe that setting an appropriate value for the parameter d (neither too small nor too large) seems to be crucial for the timing behavior of first strategy while the second strategy seems to be much less independent; unsurprisingly, for the memory occupation, larger values of d corresponds to larger heap sizes but much less than those of the first test set (notice that the numbers in Table 9 are in GB whereas those in Table 10 are in MB).

Discussion on experiments There are two main lessons learned from the experiments discussed above. First, the transformation of the original MSRMP (3) over $\langle\mu_T\rangle_{T\in\mathcal{T}}$ into the one (4) over the $\langle x_T\rangle_{T\in\mathcal{T}}$ allows for a substantial reduction of the search space. To see this, consider the Reduction Factor in Table 9. Second, considering the family $\mathcal{C}_{T\in\mathcal{T}}$ of controls associated to each threat $T\in\mathcal{T}$ is crucial, in practice, to reduce the search space of the problem of transforming back a solution $\langle x^*_T\rangle_{T\in\mathcal{T}}$ of (4) into the set $\{\langle\mu_T\rangle_{T\in\mathcal{T}}\}$ of associated mitigation mappings of the original MSRMP (3). This is so because the cardinality of $\mathcal{C}_T$ is usually low for each $T\in\mathcal{T}$ so that, despite the exponential complexity as discussed at the end of Section 3.2, the time and memory consumption are reasonable in practice.

6. Conclusions and future work

We have introduced the Multi-Stakeholder Risk Minimization Problem (MSRMP) to assist in the definition of the best (with respect to all the stakeholders involved in the system) Risk Management Policies (RMPs)—as an appropriate set of security controls to mitigate the identified set of threats—in the fundamental step of selecting mitigations for risk management. We have formalized the MSRMP as a multi-objective optimization problem that can be solved by using state-of-the-art techniques for Pareto Optimality. On top of such techniques, we have proposed a semi-automated approach to define and solve instances of the MSRMP. We have also discussed strategies to reduce the large search space resulting from real instances of the MSRMP. We have illustrated the main notions of our approach on a simple yet representative running example. An implementation of the proposed approach has allowed us to perform an experimental evaluation whose results confirm the practical viability of the proposed approach.

As future work, we consider four possibilities. First, we plan to further validate the flexibility of our approach by integrating it with a methodology for the risk evaluation of identity proofing solutions introduced in Pernpruner et al. (2021). In that work, the authors present a framework composed to analyze the risks of enrollment solutions at the design time. In particular, they focus on associating security controls with threats deriving from a set of attackers, so to reduce risks at an acceptable level while guaranteeing usability and economy. However, it is left open the problem of determining the optimal set of mitigations, and this is the reason for which the approach presented in this work becomes an interesting complement. The second (medium term) possibility for future work is to identify a comprehensive baseline of controls (such as the one in the Risk Management Framework of NIST[5]) and provide an approach to tailor it to the use case sce-

[5] https://csrc.nist.gov/Projects/risk-management/about-rmf/select-step

nario under consideration in order to lower the barrier of adoption of the approach proposed here by addressing the intricacies of evaluating the trade-offs of security controls including costs and skills required. The third (and longer term) line of future work is to investigate how it is possible to smoothly combine the approach proposed in this work with available methodologies for risk management (e.g., STRIDE).

We also would like to consider the aspect of the complexity of implementing controls as possible future work. Considering the complexity of controls in the risk assessment is challenging. For instance, one form of control may handle one risk source/threat but may define new dangers. Indeed, mutually incompatible controls can be specified by using suitable specification tricks (basically, the idea is to use a logical xor). A more challenging aspect is to consider the fact that adding security controls may end up enlarging the attack surface, thus contributing to increasing the risk level. This is a complex area, and such intrinsic constraints requires improving the optimization approach presented in this work.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Majid Mollaeefar: Methodology, Investigation, Software, Writing – review & editing. **Silvio Ranise:** Supervision, Conceptualization, Methodology, Writing – review & editing.

Data availability

Data will be made available on request.

References

Agarwal, S., 2015. Developing a structured metric to measure privacy risk in privacy impact assessments. In: IFIP International Summer School on Privacy and Identity Management. Springer, pp. 141–155.

Ahmadian, A.S., Strüber, D., Riediger, V., Jürjens, J., 2018. Supporting privacy impact assessment by model-based privacy analysis. In: Proceedings of the 33rd Annual ACM Symposium on Applied Computing, pp. 1467–1474.

Albakri, S.H., Shanmugam, B., Samy, G.N., Idris, N.B., Ahmed, A., 2014. Security risk assessment framework for cloud computing environments. Secur. Commun. Netw. 7 (11), 2114–2124.

Alshammari, M., Simpson, A., 2017. Towards a principled approach for engineering privacy by design.

Bieker, F., Friedewald, M., Hansen, M., Obersteller, H., Rost, M., 2016. A process for data protection impact assessment under the European general data protection regulation. In: Annual Privacy Forum. Springer, pp. 21–37.

Bisztray, T., Gruschka, N., 2019. Privacy impact assessment: comparing methodologies with a focus on practicality. In: Nordic Conference on Secure IT Systems. Springer, pp. 3–19.

Boeckl, K. R., Lefkovitz, N. B., et al., 2020. NIST privacy framework: a tool for improving privacy through enterprise risk management, version 1.0.

Brooks, S., Brooks, S., Garcia, M., Lefkovitz, N., Lightman, S., Nadeau, E., 2017. An introduction to privacy engineering and risk management in federal systems.

Clarke, R., 2009. Privacy impact assessment: its origins and development. Comput. Law Secur. Rev. 25 (2), 123–135.

CNIL (Commission Nationale de l'Informatique et des Libertés), 2012. Methodology for privacy risk management: how to implement the data protection act. https://www.cnil.fr/sites/default/files/typo/document/CNIL-ManagingPrivacyRisks-Methodology.pdf.

CNIL (Commission Nationale de l'Informatique et des Libertés), 2018. Privacy risk assessment (PIA). https://www.cnil.fr/sites/default/files/atoms/files/cnil-pia-1-en-methodology.pdf.

CNIL (Commission Nationale de l'Informatique et des Libertés), 2022. The open source PIA software helps to carry out data protection impact assessment. https://www.cnil.fr/en/open-source-pia-software-helps-carry-out-data-protection-impact-assesment. Accessed: February 2020.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., Stein, C., 2001. 35.5: The subset-sum problem.

Data protection-specification for a personal information management system, 2017. Available at: https://www.bsigroup.com/en-GB/BS-10012-Personal-information-management/.

De, S., Le Métayer, D., 2016. PRiAM: a privacy risk analysis methodology. In: Data Privacy Management and Security Assurance. Springer, pp. 221–229.

De, S.J., Le Métayer, D., 2017. A refinement approach for the reuse of privacy risk analysis results. In: Annual Privacy Forum. Springer, pp. 52–83.

Dor, D., Elovici, Y., 2016. A model of the information security investment decision–making process. Comput. Secur. 63, 1–13.

Evaluating the level of risk for a personal data processing operation., 2020. https://www.enisa.europa.eu/risk-level-tool/risk.

Fielder, A., Panaousis, E., Malacaria, P., Hankin, C., Smeraldi, F., 2016. Decision support approaches for cyber security investment. Decis. Support Syst. 86, 13–23.

Figueira, J., Greco, S., Ehrgott, M., 2005. State of the art surveys.

Freund, J., Jones, J., 2014. Measuring and Managing Information Risk: A FAIR Approach. Butterworth-Heinemann.

für Datenschutz, U. L., 2020. The standard data protection model: a concept for inspection and consultation on the basis of unified protection goals.https://www.datenschutzzentrum.de/uploads/sdm/SDM-Methodology_V2.0b.pdf.

Ganin, A.A., Quach, P., Panwar, M., Collier, Z.A., Keisler, J.M., Marchese, D., Linkov, I., 2017. Multicriteria decision framework for cybersecurity risk assessment and management. Risk Anal..

Gary, S., Alice, G., Alexis, F., 2002. Risk management guide for information technology systems. Special Publication 800–300.

GS1, 2015. EPC/RFID privacy impact assessment tool. https://www.gs1.org/standards/epc-rfid/pia. Accessed: January 2021.

Gupta, M., Rees, J., Chaturvedi, A., Chi, J., 2006. Matching information security vulnerabilities to organizational security profiles: a genetic algorithm approach. Decis. Support Syst. 41 (3), 592–603.

Information Commission's Office (ICO), 2018. Data protection impact assessments. https://ico.org.uk/media/for-organisations/guide-to-the-general-data-protection-regulation-gdpr/data-protection-impact-assessments-dpias-1-0.pdf. (accessed on 6 June 2019.

International Organization for Standardization, 2014. (ISO). ISO/IEC 27018: 2014-information technology–security techniques–code of practice for protection of personally identifiable information (PII) in public clouds acting as PII processors. Available online: https://www.iso.org/obp/ui/#iso:std:iso-iec:27018:ed-1:v1:en.

International Organization for Standardization, 2017a. (ISO). ISO/IEC 29134: 2017-information technology–security techniques–guidelines for privacy impact assessment. Available online: https://www.iso.org/obp/ui/#iso:std:iso-iec:29134:ed-1:v1:en.

International Organization for Standardization, 2017b. (ISO). ISO/IEC 29151:2017-information technology' security techniques' code of practice for personally identifiable information protection. Available online: https://www.iso.org/obp/ui/#iso:std:iso-iec:29151:ed-1:v1:en.

Introduction to the spia program, 2016. https://www.isc.upenn.edu/sites/default/files/introduction_to_spia_program.pdf. Accessed: December 2020.

Iwaya, L.H., Fischer-Hübner, S., Åhlfeldt, R.-M., Martucci, L.A., 2019. Mobile health systems for community-based primary care: identifying controls and mitigating privacy threats. JMIR mHealth and uHealth 7 (3), e11642.

JTCIJSS, 2013. Information technology–security techniques–information security management systems–requirements.

Kavallieratos, G., Spathoulas, G., Katsikas, S., 2021. Cyber risk propagation and optimal selection of cybersecurity controls for complex cyberphysical systems. Sensors 21 (5), 1691.

Kiesling, E., Ekelhart, A., Grill, B., Strauss, C., Stummer, C., 2016. Selecting security control portfolios: a multi-objective simulation-optimization approach. EURO J. Decis. Process. 4 (1-2), 85–117.

Klamroth, K., 2009. Discrete multiobjective optimization.

Llansó, T., McNeil, M., Noteboom, C., 2019. Multi-criteria selection of capability-based cybersecurity solutions.

Marler, R. T., Arora, J. S., 2004. Survey of multi-objective optimization methods for engineering.

McLaughlin, S., Konstantinou, C., Wang, X., Davi, L., Sadeghi, A.-R., Maniatakos, M., Karri, R., 2016. The cybersecurity landscape in industrial control systems. Proc. IEEE 104 (5), 1039–1057.

Meis, R., Heisel, M., 2015. Supporting privacy impact assessments using problem-based privacy analysis. In: ICSOFT. Springer, pp. 79–98.

Mollaeefar, M., Siena, A., Ranise, S., 2020. Multi-stakeholder cybersecurity risk assessment for data protection. 10.5220/0009822703490356

NIST, S., 2012. 800-30 revision 1.

Oetzel, M. C., Spiekermann, S., 2014. A systematic methodology for privacy impact assessments: a design science approach.

Otero, A. R., 2014. An information security control assessment methodology for organizations.

Panaousis, E., Fielder, A., Malacaria, P., Hankin, C., Smeraldi, F., 2014. Cybersecurity games and investments: a decision support approach. In: International Conference on Decision and Game Theory for Security. Springer, pp. 266–286.

Papamartzivanos, D., Menesidou, S.A., Gouvas, P., Giannetsos, T., 2021. A perfect match: converging and automating privacy and security impact assessment on-the-fly. Future Internet 13 (2), 30.

Pernpruner, M., Sciarretta, G., Ranise, S., 2021. A framework for security and risk analysis of enrollment procedures: application to fully-remote solutions based on edocuments10.5220/0010554502220233

Qassim, Q.S., Jamil, N., Daud, M., Patel, A., Ja'affar, N., 2019. A review of security assessment methodologies in industrial control systems. Inf. Comput. Secur. 27 (1), 47–61.
Rajbhandari, L., Snekkenes, E., 2012. Intended actions: risk is conflicting incentives. In: International Conference on Information Security. Springer, pp. 370–386.
Rees, L.P., Deane, J.K., Rakes, T.R., Baker, W.H., 2011. Decision support for cybersecurity risk planning. Decis. Support Syst. 51 (3), 493–505.
Regulation, 2016. (eu) 2016/679 of the EUROPEAN parliament and of the council on the protection of natural persons with regard to the processing of personal data and on the free movement of such data. https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679&from=EN.
Shahpasand, M., Shajari, M., Golpaygani, S.A.H., Ghavamipoor, H., 2015. A comprehensive security control selection model for inter-dependent organizational assets structure. Inf. Comput. Secur..
Shevchenko, N., Chick, T. A., O'Riordan, P., Scanlon, T. P., Woody, C., 2018. Threat modeling: a summary of available methods.
Shostack, A., 2014. Threat modeling: designing for security.
Smeraldi, F., Malacaria, P., 2014. How to spend it: optimal investment for cyber security. In: Proceedings of the 1st International Workshop on Agents and CyberSecurity, pp. 1–4.
T. B. of Canada Secretariat, 2010. Directive of privacy impact assessments. https://www.isc.upenn.edu/sites/default/files/introduction_to_spia_program.pdf. (accessed on 29 December 2020).
Van Dijk, N., Gellert, R., Rommetveit, K., 2016. A risk to a right? Beyond data protection risk assessments. Comput. Law Secur. Rev. 32 (2), 286–306.
van Puijenbroek, J., Hoepman, J.-H., 2017. Privacy impact assessments in practice: Outcome of a descriptive field research in the Netherlands.
Vemou, K., Karyda, M., 2018. An evaluation framework for privacy impact assessment methods. In: MCIS, p. 5.
Wei, Y.-C., Wu, W.-C., Lai, G.-H., Chu, Y.-C., 2020. pISRA: privacy considered information security risk assessment model. J. Supercomput. 76 (3), 1468–1481.
Wright, D., 2012. The state of the art in privacy impact assessment. Comput. Law Secur. Rev. 28 (1), 54–61.
Wuyts, K., Joosen, W., 2015. LINDDUN privacy threat modeling: a tutorial. https://www.linddun.org/linddun.
Zulueta, Y., Martell, V., Martínez, J., Martínez, L., 2013. A dynamic multi-expert multi-criteria decision making model for risk analysis. In: Mexican International Conference on Artificial Intelligence. Springer, pp. 132–143.
Zwingelberg, H., Hansen, M., 2011. Privacy protection goals and their implications for eID systems. Privacy and Identity Management for Life: 7th IFIP WG 9.2, 9.6/11.7, 11.4, 11.6/PrimeLife International Summer School, Trento, Italy, September 5-9, 2011, Revised Selected Papers 7.

Majid Mollaeefar is a PhD candidate at the University of Genova, and he is also a member of the Security&Trust unit of the Cybersecurity research area of Fondazione Bruno Kessler, Trento. He received his MSc degree in Secure Communication from Imam Reza International University, Mashhad, Iran, in 2015. His research focuses on Cybersecurity Risk assessment, Security and Privacy Requirements, and GDPR compliance.

Silvio Ranise is full professor of Computer Science at the University of Trento and Director of the Center for Cybersecurity a Fondazione Bruno Kessler, Trento. Before, he held a research position at INRIA in France and was a visiting professor at the University of Milan, Italy. His research focuses on identity and access management, risk management, and legal compliance checking.